

# **Оценка сбоестойчивости топологии СФ-блока на разных этапах оптимизации комбинационной логики логического синтеза**

**Е. К. Эмин<sup>1</sup>, К. А. Петров<sup>2</sup>, В. В. Азаров<sup>3</sup>, А. П. Скоробогатов<sup>4</sup>,  
А. А. Антонов<sup>5</sup>**

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

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

<sup>3</sup>ФГУ ФНЦ НИИСИ РАН, Москва, azarov\_v@cs.niisi.ras.ru;

<sup>4</sup>ФГУ ФНЦ НИИСИ РАН, Москва, skorobog\_a@cs.niisi.ras.ru;

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

**Аннотация.** Проведен анализ сбоестойчивости полученных схем СФ-блока с их реальной топологической оценкой. Предложена оценка выходных характеристик полученной схемы при помощи сигмоидальной функции. Данная функция может использоваться для сравнения различных схем, а также для поиска оптимальной схемы заданной логической функции в эвристических алгоритмах, и алгоритмах машинного обучения.

**Ключевые слова:** сложно-функциональный блок, логический синтез, оптимизация, сбоестойчивость, одиночные сбои, проектирование СБИС

## **1. Введение**

Развитие электронной компонентной базы для космических аппаратов приводит к возрастанию требований к ее характеристикам, таким как вычислительная производительность, радиационная стойкость и сбоестойкость. В современных условиях повышение производительности часто осуществляется как за счет увеличения архитектурной сложности (увеличение количества регистров общего назначения, увеличение операций типа регистр-регистр и память-память, появление дополнительных векторных со-процессоров и т. д.), так при помощи увеличения объема внутренней памяти. Это приводит к необходимости увеличения степени интеграции современных систем на кристалле и переходу на более низкие технологические нормы производства электронно-компонентной базы, что свою очередь, повышает уязвимость интегральных микросхем к воздействию ионизирующего излучения в космическом пространстве. В настоящее время уже на этапе разработки необходимо учитывать требования к сбоестойчивости систем на кристалле, а дальнейшее снижение технологических норм производства приводит к необходимости поиска новых методов и средств для создания более устойчивых к внешним воздействиям интегральных схем [1].

## **2. Постановка задачи**

Одним из наиболее распространенных методов повышения сбоестойчивости является тройное модульное резервирование (ТМР) – этот метод позволяет эффективно маскировать сбои в элементах памяти. Существует множество различных методов троирования, имеющих свои достоинства и недостатки [2,3]. Однако, для современных технологических процессов ни один из существующих методов аппаратного троирования не гарантирует полную стойкость к одиночным эффектам по причине возникновения множественных сбоев в нескольких элементах одновременно, что приводит к сбоям как в комбинационных путях, так и дереве распространения синхросигналов [4]. Предыдущее исследование [5] показало, что ни один из исследованных вариантов тройного модульного резервирования не является абсолютно универсальным.

Различают несколько механизмов маскирования сбоев в комбинационных схемах [6]:

- логическое маскирование – сбой будет маскирован, если он затрагивает часть схемы, которая логически не влияет на конечный результат;
- электрическое маскирование – сбой будет маскирован, если импульс, создаваемый попаданием частицы, затухает до того, как он достигнет входа триггера.
- временное маскирование – сбой

маскирован, если индуцированный импульс не достигнет входа триггера в момент окна фиксации.

Логический синтез является ключевым компонентом современных процессов автоматизации проектирования микросхем. Основную задачу логического синтеза можно сформулировать следующим образом: Данна логическая функция  $f: B^n \rightarrow B$  (где  $B$  обозначает булево множество  $\{0, 1\}$ ). Логическая функция  $f$  может быть реализована множеством эквивалентных логических схем. Необходимо найти логическую схему, реализующую заданную функцию  $f$  с минимальным числом логических элементов.

На практике полный перебор всех возможных решений невозможен. Исторически первый эвристический алгоритм минимизации был предложен Р. Брайнтом на основе диаграмм двоичного выбора (BDD) [7].

Данный алгоритм направлен на минимизацию систем булевых функций, что позволяет уменьшать количество элементов в комбинационных путях, тем самым снижая максимальную занимаемую площадь, и как следствие, повышать максимально возможное быстродействие синтезируемых логических функций. Сейчас в задаче логического синтеза применяют как эвристические алгоритмы (Espresso, MIS II и т. д.), так и алгоритмы машинного обучения [8].

Исследование влияния структуры схемы показало, что ее маскирующие свойства могут изменяться в зависимости от выбранных методов синтеза [9]. Поэтому в рамках данной работы предлагается рассматривать только механизм логического маскирования и оценить сбоестойчивость на разных этапах алгоритма перебора всех возможных решений.

Существуют как коммерческие САПР логического синтеза (Design Compiler (Synopsys), Encounter RTL Compiler (Cadence Design Systems) и др.), так и проекты с открытым исходным кодом: Yosys. В коммерческих САПР при логическом синтезе учитываются параметры временной спецификации для анализа быстродействия схемы, однако их алгоритм их работы не находится в открытом доступе. В рамках данного исследования в качестве инструмента логического синтеза использовался Yosys.

На практике особую важность при моделировании последствий множественных сбоев имеет реальное топологическое размещение стандартных ячеек. Кроме того, при проектировании топологии микросхемы для достижения заданного быстродействия комбинационные пути могут подвергнуться изменениям (добавлены буферы, инверторы, меняться ветвление по входам и выходам), а также соблюдения правил проектирования. Таким образом, оценку их эффективности необходимо осуществлять с учетом точного положения стандартных ячеек на топологии микросхемы.

В рамках работ, проводимых в ФГУ ФНЦ НИИСИ РАН, был создан инструмент для моделирования последствий сбоев в ячейках с учетом топологии [10]. Он представляет собой набор классов SystemVerilog и позволяет моделировать последствия сбоев во время моделирования нет-листа. Цели для внесения сбоев могут задаваться при помощи выбора случайной координаты – инжектор генерирует квадратную область заданного размера и на основе топологической информации определяет какие ячейки оказались в этой области. Для всех пораженного последовательного элемента моделируется переключение его в противоположное логическое состояние. Сбой в комбинационном элементе (SET) – как временное изменение логического состояния (длительностью 100 пс) на выходе элемента.

## 2.1. Тестируемое устройство

В качестве объекта исследования был выбран 32-разрядный блок целочисленного умножения/деления, входящий в состав микропроцессорного ядра. Функционально он содержит логику для выполнения операции деления – самой длительной в трактах инструкций в составе АЛУ, во время которой не происходит промежуточного сохранения результатов в регистрах или ОЗУ. Кроме того, его структура достаточно неоднородна, в составе нет макроблоков и элементов памяти. Это позволит адекватно оценить влияние оптимизации комбинационной логики на сбоестойчивость. Дальнейшее исследование проводилось без изменения исходного RTL-кода.

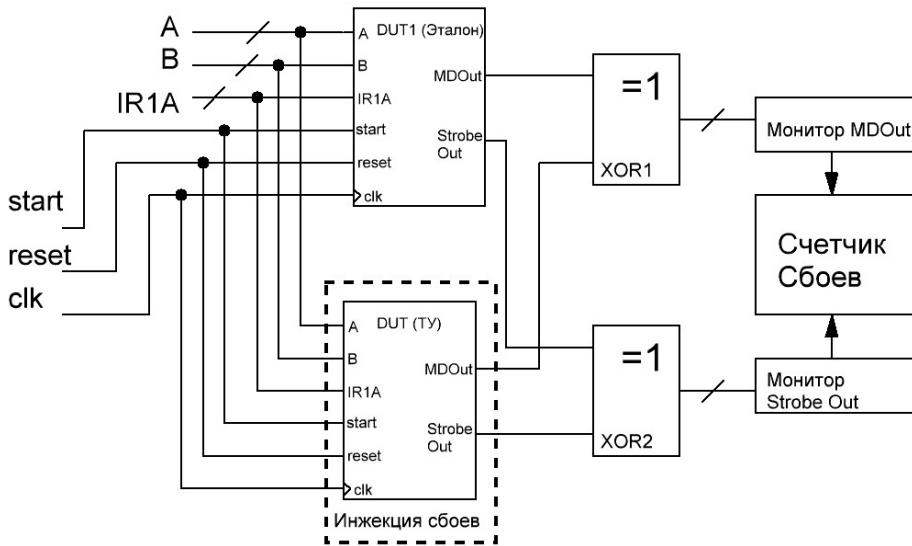


Рис. 1. Схема тестового окружения

## 2.2. Методика моделирования

Для тестирования каждого из вариантов была разработана тестовая схема, которая состоит из двух версий блока делителя/умножителя: эталонной и тестовой, в которую инжектировались сбои. Сначала в рамках тестовой задачи на входы A, B, IR1A (выбор соответствующего режима) обеих версий подаются случайные значения, после чего запускается тест, длительностью 35 тактов (максимальная длительность алгоритма деления). В случайный момент выполнения теста инжектируется один сбой и в пораженных ячейках моделируется одиночных эффектов. В конце теста значения выходов обеих версий сравниваются при помощи операции ИСКЛ-ИЛИ, и при любом различии выходов фиксируется ошибка. Тестирование повторяется циклически  $10^4$  раз для набора статистических данных.

Таблица 1. Характеристики СФ-блоков деления/умножения

N	Период, нс	Площадь, мкм <sup>2</sup>	P <sub>f</sub> , мВт	$\sigma$ , 10 <sup>-2</sup> мм <sup>2</sup>	Score
0	2,4	147946,8	46,3	0,51	3,15
10	2,2	147946,8	46,6	0,80	3,21
20	2,2	148407,2	49,9	2,39	3,44
30	2,3	148038,9	49,10	1,34	3,32
40	2,3	148038,9	49,3	1,46	3,34
50	2,2	148038,9	44,9	0,44	3,13
55	2,3	147854,8	47,7	0,57	3,16

За основу был взят маршрут логического синтеза при помощи инструмента с открытым

исходным кодом Yosys, т. к. из рассмотренных инструментов синтеза только он позволяет получать промежуточные итерации минимизации логических функций.

В рамках стандартного маршрута синтез комбинационных путей выполняет другой инструмент ABC, который использует алгоритм упрощения диаграмм двоичного выбора ESPRESSO. Алгоритм является итерационным и для данного исследуемого СФ-блока занимает 55 итераций.

Для упрощения задачи проектирования топологии было решено выбрать 6 этапов с интервалом 10 итераций и последнюю итерацию. Далее была получена топология для каждой из этих схем. Характеристики этих схем приведены в таблице 1.

## 2.3. Результаты моделирования

Для оценки сбоестойчивости разных вариантов использовалось сечение событий  $\sigma$  (в мм<sup>2</sup>). Сечение событий – это отношение числа тестов, в которых была зафиксирована ошибка к флюенсу (1):

$$\sigma = \frac{N}{\Phi}, \quad (1)$$

где  $N$  – число тестов, завершившихся с ошибкой,  $\Phi$  – значение флюенса.

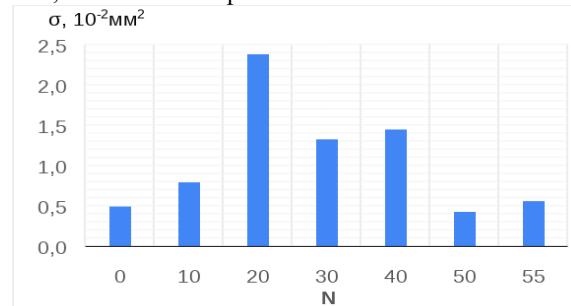


Рис. 2. Диаграмма сечения событий на разных этапах оптимизации

При проектировании топологии использовался единый изначальный коэффициент заполнения 50%. Т.к. площадь изменялась незначительно – это позволило набрать одинаковый флюенс для всех вариантов при помощи одного количества тестов  $0,68 \times 10^{-5} \text{ мм}^2$ .

Сечения событий для вариантов тестируемого блока представлены на диаграмме. Как можно убедиться, с количеством итераций значение сечения событий изменяется немонотонно. Можно отметить снижение стойкости к одиночным сбоям с увеличением количества используемых функций для итераций с 1 по 20. Таким образом, можно констатировать, что увеличение количества элементов в схеме и создание подобной избыточности не всегда ведет к улучшению стойкости к одиночным сбоям, а в случае 20 итерации даже наоборот – к увеличению количества элементов, затронутых многократным сбоем и увеличению сечения событий в 4,7 раза.

Также следует отметить, что последняя итерация также не является наиболее сбоестойчивой из рассматриваемых. А в сравнении с исходной схемой характеризуется 12% увеличением сечения событий. Самым низким значением сечения событий обладает вариант, соответствующий 50-ой итерации –  $0,44 \times 10^{-2} \text{ мм}^2$ . Помимо этого, необходимо отметить, что он обладает наилучшим быстродействием.

Таким образом, возникает задача выбора оптимальной схемы, обладающей наилучшими характеристиками. При чем для получения оптимальной схемы необходимо учитывать начальные параметры уже на этапе логического синтеза.

В качестве решения предлагается определение численной оценки спроектированных вариантов на основе функции сигмоиды. Для формирования оценки предлагается использовать следующую функцию (2):

$$S = \alpha \text{sigm}(P) + \beta \text{sigm}(S) + \gamma \text{sigm}(P_f) + \eta \text{sigm}(\sigma), \quad (2)$$

где  $\alpha$ ,  $\beta$ ,  $\gamma$  и  $\eta$  – коэффициенты влияния периода  $P$ , площади  $S$ , полной потребляемой мощности  $P_f$  и сечения событий  $\sigma$ . В рамках данной работы  $\alpha$ ,  $\beta$ ,  $\gamma$  и  $\eta$  приняты равными 1. Данная оценка позволяет оценить “качество” полученной схемы – меньшему значению функции соответствует схема с лучшими характеристиками.

В таблице 1 для каждого из вариантов также указаны рассчитанные значения данной функции. Как можно убедиться, наилучший вариант из исследуемых – это вариант, соответствующий 50 итерации.

### 3. Заключение

В статье рассмотрена оценка сбоестойчивости схем, полученных на разных этапах работы алгоритма минимизации логических функций во время операции логического синтеза. Показано, что последняя итерация соответствующая минимальной с точки зрения количества логических элементов не является наиболее сбоестойчивой из рассматриваемых.

Самым низким значением сечения событий и наилучшим быстродействием обладает вариант, соответствующий 50-ой итерации.

Предложена оценка выходных характеристик полученной схемы при помощи сигмоидальной функции. Использование данной функции позволяет как сравнивать различные варианты схем, так и применять ее для поиска оптимальной схемы заданной логической функции в эвристических алгоритмах и в алгоритмах машинного обучения.

Публикация выполнена в рамках государственного задания ФГУ ФНЦ НИИСИ РАН по теме FNEF-2022-0008.

## Fault Tolerance Evaluation of IP-Block Topology at Different Stages of Combinational Logic Synthesis

E. K. Emin, K. A. Petrov, V. V. Azarov, A. P. Skorobogatov, A. A. Antonov

**Abstract.** The analysis of the fault stability of the obtained schemes with a real topological assessment is represented. Fault tolerance evaluation of the resulting circuit using a sigmoidal function is proposed. This function can be used to compare different circuits, and also used to find the optimal circuits of a given logical function in heuristic and machine learning algorithms.

**Keywords:** IP-block, logical synthesis, optimization, fault tolerance, single events, VLSI design

## Литература

1. R. E. Bryant et al., Limitations and challenges of computer-aided design technology for CMOS VLSI // Proceedings of the IEEE, vol. 89, no. 3, pp. 341-365, March 2001.
2. Mahatme N.N., Jagannathan S., Loveless T.D., Massengill L.W., Bhava B.L., Wen S-J. et al., Comparison of combinational and sequential error rates for a deep submicron process // IEEE Trans Nucl. Sci. (IEEE T NUCL SCI). 2011. Vol. 58, No.6. P. 2719-2725.
3. V. Petrovic and M. Krstic, Design Flow for Radhard TMR Flip-Flops // 2015 IEEE 18th International Symposium on Design and Diagnostics of Electronic Circuits & Systems, Belgrade, Serbia, 2015, pp. 203-208.
4. P. E. Dodd, M. R. Shaneyfelt, J. R. Schwank, and J. A. Felix, Current and future challenges in radiation effects on cmos electronics // IEEE Transactions on Nuclear Science, vol. 57, no. 4, pp. 1747–1763, Aug. 2010.
5. Власов А.О., Клишин А.В., Желудков Н.В., Эмин Е.К., Горбунов М.С. Сравнительная характеристика методов повышения сбоестойчивости топологии блоков целочисленного умножения/деления в проектных нормах 65нм // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС). 2020. Выпуск 3. С. 188-193.
6. Mukherjee, S. (2008). Architecture Design for Soft Errors.
7. R. E. Bryant, Graph-Based Algorithms for Boolean Function Manipulation // IEEE Transactions on Computers, Vol. C-35, No. 8 (August, 1986), pp. 677–691. Reprinted in M. Yoeli, Formal Verification of Hardware Design, IEEE Computer Society Press, 1990, pp. 253–267.
8. S. Rai et al., Logic Synthesis Meets Machine Learning: Trading Exactness for Generalization // 2021 Design, Automation & Test in Europe Conference & Exhibition (DATE), Grenoble, France, 2021, pp. 1026-1031.
9. Стемпковский А.Л., Соловьев Р.А., Тельпухов Д.В. Повышение сбоестойчивости логических схем на основе частичного ресинтеза схемы // Информационные технологии. 2016. Т. 22. №7. С. 515-522.
10. P. Chernyakov et al., "Comparative Analysis of Layout-Aware Fault Injection on TMR-based DMA Controllers," // 2019 IEEE 31st International Conference on Microelectronics (MIEL), Nis, Serbia, 2019, pp. 289-292.