

14. Kuliev E.V., Kureychik V.V., Kursitya I.O. Prinyatie resheniy v zadache razmeshcheniya komponentov SBIS na osnove modeli povedeniya stai volkov [Decision-making in the problem of placement of VLSI components based on the behavior model of a pack of wolves], *Mezhdunarodnaya konferentsiya po myagkim vychisleniyam i izmereniyam* [International conference on soft computing and measurements], 2018, Vol. 1, pp. 712-715.
15. Kuliev E.V., Shcheglov S.N., Pantelyuk E.A., Kulieva N.V. Adaptivnyy algoritm stai serykh volkov dlya resheniya zadach proektirovaniya [The adaptive algorithm of the pack of gray wolves for the decision of tasks of design], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2017, No. 7 (192), pp. 28-38.
16. Kuliev E.V., Lezhebokov A.A. O gibridnom algoritme razmeshcheniya komponentov SBIS [On hybrid algorithm of VLSI components placement], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2012, No. 11 (136), pp. 188-192.
17. Kuliev E.V., Lezhebokov A.A. Issledovanie kharakteristik gibridnogo algoritma razmeshcheniya [The study of the characteristics of hybrid positioning algorithm], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2013, No. 3 (140), pp. 255-261.
18. IBM-PLACE 2.0 benchmark suits. <http://er.cs.ucla.edu/benchmarks/ibm-place2/bookshelf/ibm-place2-all-bookshelf-nopad.tar.gz>.
19. Adya, S.N., Markov, I.L. Combinatorial techniques for mixed-size placement, *ACM Transactions on Design Automation of Electronic Systems*, 2005, No. 10 (1), pp. 58-90.
20. Wang M., Yang X., Sarrafzadeh M. Dragon2000: Standard-cell Placement Tool for Large Industry Circuits – ICCAD2000, pp. 260-263.

Статью рекомендовал к опубликованию д.т.н., профессор Н.И. Витиска.

**Кулиев Эльмар Валерьевич** – Южный федеральный университет; e-mail: ekuliev@sfnu.ru; 347928, г. Таганрог, пер. Некрасовский, 44; тел.: 88634371651; кафедра систем автоматизированного проектирования; доцент.

**Запорожец Дмитрий Юрьевич** – e-mail: duzaporozhets@sfnu.ru; кафедра систем автоматизированного проектирования; доцент.

**Терешенко Дмитрий Юрьевич** – e-mail: tereshenko@sfnu.ru; кафедра систем автоматизированного проектирования; аспирант.

**Kuliev Elmar Valerievich** – Southern Federal University; e-mail: ekuliev@sfnu.ru; 44, Nekrasovskiy lane, Taganrog, 347928, Russia; phone: +78634371651; the department of computer aided design; associate professor.

**Zaporozhets Dmitriy Yurievich** – e-mail: duzaporozhets@sfnu.ru; the department of computer aided design; associate professor.

**Tereshenko Dmitriy Yurievich** – e-mail: tereshenko@sfnu.ru; the department of computer aided design; graduate student.

УДК 519.725, 621.3.049.771.14

DOI 10.23683/2311-3103-2019-4-195-206

**А.Л. Стемповский, Д.В. Тельпухов, С.И. Гуров, Т.Д. Жукова, А.Н. Щелоков,  
А.Д. Новиков**

### **СИНТЕЗ СФК НА ОСНОВЕ LDPC КОДА С ИСПОЛЬЗОВАНИЕМ МАЖОРИТАРНОГО ДЕКОДИРОВАНИЯ**

*Ионизирующее излучение приводит к возникновению кратковременных нарушений работоспособности электронной аппаратуры. Данный тип сбоя в основном рассматривался в контексте запоминающих устройств и элементов памяти. Однако, интенсивное развитие микроэлектронной промышленности приводит к росту числа сбоев в комбинационных участках, и в скором времени может привести к тому, что частота возникновения сбоев в данных участках будет сопоставима с частотой в незащищенных элементах памяти. На сегодняшний день существует множество различных методов борьбы с последствиями*

возникновения случайных сбоев: традиционные методы  $N$ -кратного модульного резервирования, методы, позволяющих усиливать маскирующие свойства схемы, использование различных средств контроля на основе теории помехоустойчивого кодирования и т.д. Но в основном большинство из представленных методов приводят к появлению большой структурной избыточности. Вследствие этого возникает необходимость в разработке новейших методов борьбы с последствиями случайных сбоев. В рамках данной статьи рассматривается применение особых средств контроля – схем функционального контроля (СФК) на основе низкоплотностного кода с целью повышения сбоеустойчивости комбинационных схем. Использование такой СФК позволяет выполнить исправление однократной ошибки за счет метода мажоритарного декодирования. При сравнении с методом тройного модульного резервирования по таким параметрам как логическая чувствительность и структурная избыточность, применение полученной СФК может стать некоторым компромиссным решением проблемы устойчивости схемы к возникновению случайных сбоев.

*Сбоеустойчивость комбинационных схем; информационная избыточность; избыточное кодирование; низкоплотностный код.*

**A.L. Stempkovskiy, D.V. Telpuhov, S.I. Gurov, T.D. Zhukova, A.N. Schelokov,  
A.D. Novikov**

### **SYNTHESIS METHOD OF FAULT-TOLERANT COMBINATION CIRCUITS WITH CED BASED ON LDPC CODE**

*Ionizing radiation leads to the occurrence of short-term disturbances in the performance of electronic equipment, so-called soft errors. This type of failure was mainly considered in the context of storage devices and memory elements. However, the intensive development of the microelectronic industry leads to an increase in the number of soft errors in combinational areas, and may soon lead to the fact that the frequency of occurrence of soft errors in these areas will be comparable to the frequency in unprotected memory elements. Today, there are many different methods to deal with the consequences of soft errors: traditional methods of  $N$ -fold modular redundancy, methods that enhance the masking properties of the circuit, the use of various control tools based on the theory of noise-resistant coding, etc. However, basically, most of the methods presented lead to the appearance of large structural redundancy. As a result, there is a need to develop the new methods to deal with the consequences of soft errors. This article discusses the use of special control tools – concurrent error detection (CED) circuits based on a low-density code in order to increase the fault tolerance of combination circuits. The use of such a CED allows single error correction due to the majority decoding method. When compared with the triple modular redundancy method in terms of parameters such as logical sensitivity and structural redundancy, the application of the obtained CED can be some compromise solution to the problem of the circuit fault tolerance to the occurrence of soft errors.*

*Fault tolerance of combinational circuits; information redundancy; redundancy coding; low-density code.*

**Введение.** Существует множество разнообразных методов борьбы с последствиями случайных сбоев, не зависящих от причины и природы их возникновения. Данные методы, основываются на использовании кратного резервирования, механизмов маскирования и средств контроля.

Использование методов кратного резервирования является традиционным подходом к борьбе с последствиями возникновения сбоев [1–2]. Данный подход реализуется с помощью  $N$ -кратного копирования основной схемы с последующим объединением полученных копий схем и проверки правильности функционирования результирующего выхода с помощью схемы голосования. Обычно используют метод тройного модульного резервирования (TMR, Triple Modular Redundancy), позволяющий в случае возникновения не более чем одиночной ошибки в одном из экземпляров резервируемого модуля и безошибочной работы схемы мажоритарного исправления эту ошибку. Основным недостатком данного подхода является внесение в схему большой структурной избыточности.

На сегодняшний день основным методом повышения сбоеустойчивости комбинационной схемы является подход, связанный с генерацией средств контроля [3]. Как известно, эти средства делятся на следующие группы в зависимости от вида контроля:

- ◆ *тестовый контроль* (offline testing) – проводится вне штатной работы устройства;
- ◆ *функциональный контроль* (online testing) – проводится непрерывно в рабочем режиме.

Предлагаемый подход на основе синтеза схемы функционального контроля (СФК) [4] с использованием низкоплотного (LDPC) кода способен исправлять одиночные ошибки выхода комбинационной схемы [5]. В данной работе показано, что такой подход является компромиссом между надежностью комбинационной схемы и аппаратными затратами.

**1. Низкоплотный код.** Низкоплотный код (LDPC, low-density parity check code) – частный случай блочного линейного  $(n, k)$ -кода с проверкой на четность, в котором проверочная матрица  $H_{m \times n}$  является сильно разреженной (содержит большое число 0) [6–7]. Малое число единиц в данной проверочной матрице обеспечивает простоту проверки на четность.

Если каждый столбец проверочной матрицы содержит одинаковое число 1 и каждая строка – ровно  $j < n$  единиц, то код является регулярным –  $(n, j, k)$ .

Важной характеристикой проверочной матрицы низкоплотного кода является отсутствие циклов определенного размера. Для более четкого понимания что такое цикл в матрице рассмотрим цикл длиной 4. Под ним понимают образование в  $H_{m \times n}$  прямоугольника, в углах которого стоят единицы. Ниже приведена проверочная матрица низкоплотного несистематического  $(6, 3, 9)$ -кода с минимальным циклом длины 8. Его образуют единицы, выделенные жирным шрифтом.

$$H_{6 \times 9} = \begin{bmatrix} 1 & \mathbf{1} & \mathbf{1} & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & \mathbf{1} & 0 & 0 & \mathbf{1} & 1 & 0 & 0 & 0 \\ 0 & 0 & \mathbf{1} & 0 & 0 & 0 & \mathbf{1} & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{1} & 0 & \mathbf{1} & 0 & 1 \end{bmatrix}$$

В общем случае определение кодового расстояния  $d(C)$  низкоплотного кода – сложная задача.

Считается, что LDPC коды обладают плохим минимальным расстоянием, но, тем не менее, обладает хорошими корректирующими свойствами.

Для осуществления кодирования линейным кодом необходимо задать порождающую матрицу  $G$ . Если проверочная матрица имеет вид

$$H = [I_m \quad P]$$

то порождающую получают следующим образом:

$$G = \begin{bmatrix} P \\ I_m \end{bmatrix}.$$

В общем случае  $G$  можно получить из соотношения  $HG = \mathbf{0}$ , где  $\mathbf{0}$  – нулевая матрица размера  $m \times n$ .

В настоящее время используется два подхода к построению  $H$  – с помощью:

- 1) псевдослучайного генератора (случайные, random-like codes) [6–8].
- 2) специальных алгебраических методов, основанных на теории групп конечных полей (структурированные) [9–11].

Наилучшими по корректирующим свойствам являются проверочные матрицы, полученные с помощью псевдослучайных генераторов. Однако полученные с помощью алгебраических методов позволяют использовать методы оптимизации процедур хранения, кодирования и декодирования.

Сложность декодирования низкоплотных кодов в большей степени зависит от наличия коротких циклов в матрице  $H$ , увеличиваясь при их наличии. Это приводит к необходимости использования алгоритмов для определения и удаления данных циклов [12].

Для декодирования низкоплотных кодов можно использовать мажоритарное декодирование, методы с использованием аппарата быстрого преобразования Фурье или лексикографического метода, итерационного распространения доверия (IBP, Iterative Belief-Propagation) [13] и др. [14].

**2. Принцип мажоритарного декодирования.** Декодирование низкоплотного кода сводится к поиску вектора ошибок  $e$  – кодового слова длины  $n$ , для которого выполняется повышение апостериорных вероятностей полученных бит.

Для СФК на основе низкоплотного кода в рамках данной работы в качестве метода декодирования был выбран метод мажоритарного декодирования [15]. Данный метод был выбран из-за легкости и простоты его выполнения.

Основная идея метода мажоритарного декодирования заключается в составлении для каждого информационного разряда системы проверки последующем сравнении, полученных в результате значений по методу большинства, мажоритарными элементами. В качестве уравнений проверок используются линейные комбинации строк проверочной матрицы  $H$ .

Число уравнений системы проверок вычисляется по формуле  $2\alpha s + 1$ , где  $\alpha$  – связность системы проверок (число повторений в системе для информационного разряда разрядов, через которые он проверяется),  $s$  – кратность исправляемой ошибки.

Системы проверок по параметру связности бывают связанные и отдельные. Системы связанных проверок – это тип системы проверок, при которой:

- ◆ информационный разряд, для которого выполняется проверка, входит в каждое из уравнений системы;
- ◆ любые другие информационные разряды, через которые выполняется проверка входят в уравнения системы проверки не более  $\alpha$  раз.

Системы отдельных проверок отличаются от системы связанных тем, что  $\alpha = 1$ , т.е. информационные разряды, через которые выполняется проверка, входят в уравнения системы только один раз.

Рассмотрим для примера составление систем проверок для проверочной матрицы  $H$ :

$$H = \begin{vmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{vmatrix}.$$

Данная проверочная матрица является систематической, потому можно вычисление проверочных разрядов выполняется довольно просто:

$$\begin{cases} r_1 = y_1 \oplus y_2 \oplus y_3 \\ r_2 = y_2 \oplus y_3 \oplus y_4, \\ r_3 = y_1 \oplus y_2 \oplus y_4 \end{cases}$$

где  $r_1, r_2, r_3$  – проверочные разряды (вычисляются кодером),  $y_1, y_2, y_3, y_4$  – информационные разряды.

В состав декодера при данном методе декодирования входят подсистемы, реализующие системы проверок для каждого из информационных разрядов. Система проверок, как было написано ранее, составляется путем сложения по модулю 2 строк проверочной матрицы  $H$ .

В результате получены следующие системы проверок, в которые входят как тривиальные (например,  $y_1 = y_1$  проверки), так и нетривиальные проверки:

$$\begin{cases} y_1 = y_1 \\ y_1 = y_4 \oplus r_2 \oplus r_3 \\ y_1 = y_3 \oplus r_1 \oplus r_3, \\ y_1 = y_3 \oplus r_2 \oplus r_3 \\ y_1 = y_2 \oplus y_4 \oplus r_3 \end{cases} \quad \begin{cases} y_2 = y_2 \\ y_2 = r_5 \oplus r_2 \oplus r_3 \\ y_2 = y_1 \oplus y_3 \oplus r_1, \\ y_2 = y_3 \oplus y_4 \oplus r_2 \\ y_2 = y_1 \oplus y_4 \oplus r_3 \end{cases}$$

$$\begin{cases} y_3 = y_3 \\ y_3 = y_1 \oplus y_2 \oplus r_1 \\ y_3 = y_4 \oplus r_1 \oplus r_3, \\ y_3 = y_2 \oplus y_4 \oplus r_1 \\ y_3 = y_1 \oplus r_1 \oplus r_2 \end{cases} \quad \begin{cases} y_4 = y_4 \\ y_4 = y_1 \oplus r_1 \oplus r_2 \\ y_4 = y_2 \oplus y_3 \oplus r_2. \\ y_4 = y_1 \oplus y_2 \oplus r_3 \\ y_4 = y_3 \oplus r_1 \oplus r_3 \end{cases}$$

Можно заметить, что некоторые информационные разряды в полученных системах проверок входят в систему более 1 раза. Следовательно, эта система связанных проверок. Поэтому для исправление однократной ошибки необходимо уравнений:

$$2as+1 = 2 \cdot 2 \cdot 1 + 1 = 5.$$

Рассмотрим процесс декодирования проверкой следующего кодового слова:

$y_1$	$y_2$	$y_3$	$y_4$	$r_1$	$r_2$	$r_3$
1	1	0	1	0	0	1

По полученным системам произведем проверку информационных разрядов:

$$\begin{cases} y_1 = 1 \\ y_1 = 1 \oplus 0 \oplus 1 = 0 \\ y_1 = 0 \oplus 1 \oplus 0 = 1, \\ y_1 = 0 \oplus 1 \oplus 0 = 1 \\ y_1 = 1 \oplus 1 \oplus 1 = 1 \end{cases} \quad \begin{cases} y_2 = 1 \\ y_2 = 0 \oplus 0 \oplus 1 = 1 \\ y_2 = 0 \oplus 1 \oplus 0 = 1, \\ y_2 = 0 \oplus 1 \oplus 0 = 1 \\ y_2 = 1 \oplus 1 \oplus 1 = 1 \end{cases}$$

$$\begin{cases} y_3 = 0 \\ y_3 = 0 \oplus 1 \oplus 1 = 0 \\ y_3 = 0 \oplus 1 \oplus 1 = 0, \\ y_3 = 0 \oplus 1 \oplus 1 = 0 \\ y_3 = 0 \oplus 1 \oplus 1 = 0 \end{cases} \quad \begin{cases} y_4 = 1 \\ y_4 = 1 \oplus 0 \oplus 1 = 0 \\ y_4 = 0 \oplus 1 \oplus 0 = 1. \\ y_4 = 1 \oplus 1 \oplus 1 = 1 \\ y_4 = 0 \oplus 1 \oplus 0 = 1 \end{cases}$$

Полученные значения для каждого информационного разряда передаются на мажоритарный элемент:

$y_1$	$y_2$	$y_3$	$y_4$
1	1	0	1

Так как ошибки не возникло, полученные в результате значения полностью совпадают с значениями, поданными на декодер.

Рассмотри тот же информационный вектор, но с однократной ошибкой. Пусть ошибка возникнет во втором разряде  $y_2$ :

$y_1$	$y_2$	$y_3$	$y_4$	$r_1$	$r_2$	$r_3$
1	0	0	1	0	0	1

Тогда по системе проверок будут получены следующие значения:

$$\begin{cases} y_1 = 1 \\ y_1 = 1 \oplus 0 \oplus 1 = 0 \\ y_1 = 0 \oplus 1 \oplus 0 = 1, \\ y_1 = 0 \oplus 1 \oplus 0 = 1 \\ y_1 = 1 \oplus 0 \oplus 1 = 0 \end{cases} \quad \begin{cases} y_2 = 0 \\ y_2 = 0 \oplus 0 \oplus 1 = 1 \\ y_2 = 0 \oplus 1 \oplus 0 = 1, \\ y_2 = 0 \oplus 0 \oplus 1 = 1 \\ y_2 = 1 \oplus 1 \oplus 1 = 1 \end{cases}$$

$$\begin{cases} y_3 = 0 \\ y_3 = 0 \oplus 1 \oplus 0 = 1 \\ y_3 = 0 \oplus 1 \oplus 1 = 0, \\ y_3 = 0 \oplus 0 \oplus 1 = 1 \\ y_3 = 0 \oplus 1 \oplus 1 = 0 \end{cases} \quad \begin{cases} y_4 = 1 \\ y_4 = 0 \oplus 0 \oplus 1 = 1 \\ y_4 = 0 \oplus 0 \oplus 0 = 0. \\ y_4 = 1 \oplus 1 \oplus 0 = 0 \\ y_4 = 0 \oplus 1 \oplus 0 = 1 \end{cases}$$

$y_1$	$y_2$	$y_3$	$y_4$
1	1	0	1

Однократная ошибка полностью исправлена.

**3. Синтез СФК на основе низкоплотного кода.** СФК на основе низкоплотного LDPC кода состоит из основной схемы и корректирующей схемы, в состав которой входят подсхема кодера, состоящего из копии основной схемы, а также блока вычисления проверочных разрядов, и подсхема декодера, в котором выполняется исправление однократной ошибки. На рис. 1 представлена структурная схема СФК для LDPC кода.

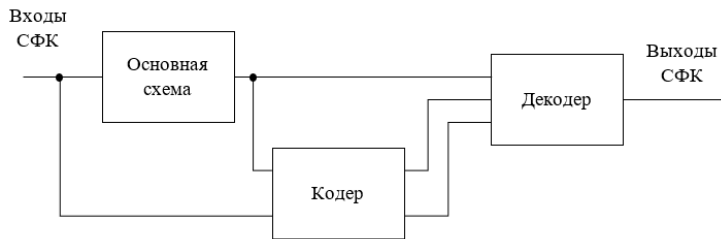


Рис. 1. СФК на основе низкоплотного LDPC кода

На рис. 2 представлена более подробная структура СФК с указанием функции каждой из подсхем, входящих в ее состав, где:



Рис. 2. Структура СФК на основе низкоплотного кода с подробным указанием подсхем входящих в состав корректирующей схемы

$x_1, \dots, x_p$  – входы основной и ее копии схем,  $y_1, \dots, y_k$  – выходы основной схемы,  $y_1^1, \dots, y_k^1$  – выходы копии основной схемы, входящей в схему кодера СФК,  $r_1^1, \dots, r_m^1$  – проверочные разряды копии основной схемы,  $y_1, \dots, y_k$  – выходы схемы функционального контроля,  $m, k$  – число проверочных и информационных разрядов.

Как было представлено в предыдущем разделе декодер для СФК на основе низкоплотного кода реализуется методом мажоритарного декодирования. Рассмотрим в качестве примера синтез кодера и декодера для схемы с 5 выходами.

Пусть для данной схемы будут сгенерированы следующие порождающая и проверочная матрицы:

$$G = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix}, \quad H = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

В состав кодера как было сказано ранее подсхема, ответственная за вычисления для копии основной схемы проверочных разрядов. Данная подсхема реализуется с помощью порождающей матрицы, а именно подматрицы  $P$ :

$$\begin{cases} r''_1 = y''_1 \oplus y''_2 \oplus y''_3 \\ r''_2 = y''_1 \oplus y''_4 \oplus y''_5 \\ r''_3 = y''_2 \oplus y''_4 \\ r''_4 = y''_3 \oplus y''_5 \end{cases}.$$

Подсхема исправления ошибки реализуется с помощью систем проверок для каждого из информационных разрядов с последующим выводом результатов на мажоритарный элемент. Число уравнений входящих в систему проверок влияет на размерность мажоритарного элемента. На рис. 3 и 4 представлены логические схемы мажоритарных элементов для системы отдельных и связанных проверок.

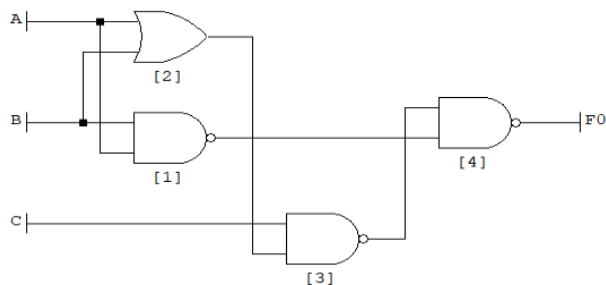


Рис. 3. Логическая схема мажоритарного элемента для системы отдельных проверок информационного разряда

Система проверок:

$$y'_1: \begin{cases} y_1 = y'_1 \\ y_1 = r''_1 \oplus y'_2 \oplus y'_3 \\ y_1 = r''_2 \oplus y'_4 \oplus y'_5 \end{cases}$$

$$y'_2: \begin{cases} y_2 = y'_2 \\ y_2 = r''_1 \oplus y'_1 \oplus y'_3 \\ y_2 = r''_3 \oplus y'_4 \end{cases}$$

$$y'_3: \begin{cases} y_3 = y'_3 \\ y_3 = r''_1 \oplus y'_1 \oplus y'_2, \\ y_3 = r''_4 \oplus y'_5 \end{cases}$$

$$y'_4: \begin{cases} y_4 = y'_4 \\ y_4 = r''_2 \oplus y'_1 \oplus y'_5, \\ y_4 = r''_3 \oplus y'_2 \end{cases}$$

$$y'_5: \begin{cases} y_5 = y'_5 \\ y_5 = r''_2 \oplus y'_1 \oplus y'_4, \\ y_5 = r''_4 \oplus y'_3 \end{cases}$$

Структура подсхем, отвечающих за систему проверок для информационных разрядов  $y'_1$  и  $y'_5$  представлена на рис. 5 и 6.

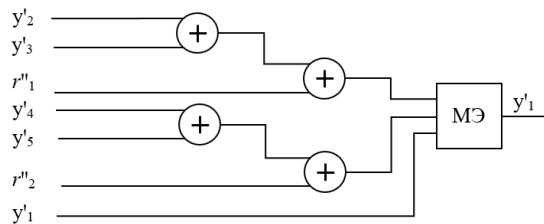


Рис. 4. Логическая схема системы проверок для информационного разряда основной схемы  $y'_1$

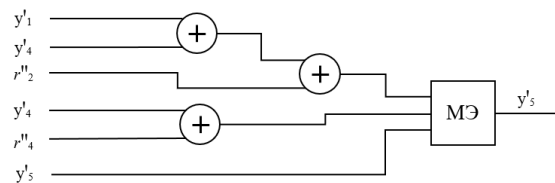


Рис. 5. Логическая схема системы проверок для информационного разряда основной схемы  $y'_5$

**4. Результаты численных экспериментов.** Для оценки эффективности применения схемы функционального контроля для повышения сбоеустойчивости комбинационных схем был проведен ряд вычислительных экспериментов на схемах из тестовых наборов ISCS'85 [16] и LGSynth'89 [17]. По результатам данных экспериментов был проведен сравнительный анализ СФК и схем на основе метода тройного модульного резервирования (ТМР) по структурным затратам и по значению надежности. Надежность схемы определялась по коэффициенту логической чувствительности [18] и коэффициенту средней наблюдаемости [19].

В табл. 1 представлены результаты сравнительного анализа по структурным затратам, а именно по числу элементов, СФК на основе низкоплотностного кода до и после оптимизации, и схемы, полученной в результате использования традиционного метода ТМР. Оптимизация схемы функционального контроля проводилась с помощью программы Yosys [20], которая позволила достичь уменьшения СФК в среднем на 27,3 %.



Таблица 1

**Результаты сравнительного анализа по числу элементов СФК на основе низкоплотностного кода и схемы на основе метода TMR, для схем из тестовых наборов ISCAS'85 и LGSynth'89**

Benchmark	Число элементов				$\frac{n_{TMR}}{n_{OC}}$	$\frac{n_{СФК}}{n_{OC}}$	$\frac{n_{OCСФК}}{n_{OC}}$
	OC, $n_{OC}$	TMR, $n_{TMR}$	СФК, $n_{СФК}$	СФК (опт.), $n_{OCСФК}$			
c1355	590	1930	1404	1036	3,27	2,38	1,76
c1908	1057	3296	2289	1508	3,12	2,17	1,43
c3540	1983	6059	4120	2418	3,06	2,08	1,22
c499	246	898	716	462	3,65	2,91	1,88
c8_synth	152	546	430	402	3,59	2,83	2,65
rd53	64	207	149	129	3,23	2,33	2,02
seq	2176	6703	4597	4240	3,08	2,11	1,95
vg2	215	685	486	428	3,19	2,26	1,99

По данным, представленным в табл. 1, было выявлено, что СФК на основе низкоплотностного кода обладает меньшей структурной избыточностью средним на 26,8 % (46,5 % для СФК после оптимизации) по сравнению со схемой, полученной в результате использования метода тройного модульного резервирования. Так же стоит обратить внимание на то, что при среднем увеличении схемы в 3,3 раз при использовании метода TMR, увеличение при использовании СФК на основе низкоплотностного кода составило в среднем 2,4 раза (для оптимизированной СФК – 1,8 раза).

Также был проведен сравнительный анализ для исследуемых схем по значениям коэффициентов логической чувствительности и средней наблюдаемости. Коэффициент логической чувствительности схемы  $\alpha$  представляет собой сумму наблюдаемостей вентилях и характеризует среднее число ненадежных элементов в схеме, ошибка в которых приводит к ошибке на выходе схемы. Наблюдаемость вентиля стоит рассматривать как вероятность того, что ошибка на вентиле не будет маскирована и повлияет на выход логической схемы при условии, что на остальных вентилях ошибки не возникало. Результаты проведенного анализа представлены в табл. 2.

Таблица 2

**Результаты сравнительного анализа по коэффициенту логической чувствительности  $\alpha$  СФК на основе низкоплотностного кода и схемы на основе метода TMR, для схем из тестовых наборов ISCAS'85 и LGSynth'89**

Benchmark	Коэффициент логической чувствительности				Коэффициент средней наблюдаемости			
	$\alpha_{OC}$	$\alpha_{TMR}$	$\alpha_{СФК}$	$\alpha_{OCСФК}$	$\frac{\alpha_{OC}}{n_{OC}}$	$\frac{\alpha_{TMR}}{n_{TMR}}$	$\frac{\alpha_{СФК}}{n_{СФК}}$	$\frac{\alpha_{OCСФК}}{n_{OCСФК}}$
c1355	201,1	96,0	267,3	161,1	0,341	0,050	0,190	0,156
c3540	445,5	79,1	558,5	347,3	0,225	0,013	0,136	0,144
c499	98,6	95,9	165,3	91,1	0,401	0,107	0,231	0,197
c5315	864,9	391,5	1148,3	162,2	0,291	0,041	0,169	0,234
c8_synth	96,1	55,5	135,7	108,1	0,632	0,102	0,315	0,269
rd53	28,3	9,8	35,7	20,9	0,443	0,047	0,240	0,162
rot_synth	412,9	356,2	651,5	523,5	0,525	0,123	0,281	0,241
vg2	48,9	32,7	79,3	69,6	0,227	0,048	0,163	0,163

Из представленной таблицы видно, что СФК на основе низкоплотностного кода проигрывает методу тройного модульного резервирования по значению коэффициента логической чувствительности. Но при этом значение коэффициента средней наблюдаемости СФК за счет меньшей структурной избыточности будет ниже в среднем на 38,8 % (42,7 % для оптимизированной СФК) при сравнении с основной схемой и на 70,2 % ниже – при сравнении с методом TMR. Данный факт приводит к выводу о эффективности применения СФК на основе низкоплотностного кода в качестве компромисса.

**Заключение.** Из приведенных в данной работе результатов были сделаны следующие выводы:

При использовании программы Yosys оптимизация СФК на основе низкоплотностного кода составила в среднем на 27,3 %. СФК на основе низкоплотностного кода обладает меньшей структурной избыточностью в среднем на 26,8 % (46,5 % для СФК после оптимизации) по сравнению со схемой, полученной в результате использования метода тройного модульного резервирования.

Так же стоит обратить внимание на то, что при среднем увеличении схемы в 3,3 раз при использовании метода TMR, увеличение при использовании СФК на основе низкоплотностного кода составило в среднем 2,4 раза (для оптимизированной СФК – 1,8 раза).

СФК на основе низкоплотностного кода проигрывает методу тройного модульного резервирования по значению коэффициента логической чувствительности. Но при этом значение коэффициента средней наблюдаемости СФК за счет меньшей структурной избыточности будет ниже в среднем на 38,8 % (42,7 % для оптимизированной СФК) при сравнении с основной схемой и на 70,2 % ниже – при сравнении с методом TMR. Данный факт приводит к выводу о эффективности применения СФК на основе низкоплотностного кода в качестве компромисса.

#### БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Стемковский А.Л., Тельпухов Д.В., Соловьев Р.А., Мячиков М.В.* Повышение отказоустойчивости логических схем с использованием нестандартных мажоритарных элементов // Информационные технологии. – 2015. – Т. 21, № 10. – С. 749-756.
2. *El-Maleha A.H., Oughalia F.C.* A generalized modular redundancy scheme for enhancing fault tolerance of combinational circuits // Microelectronics Reliability. – 2014. – Vol. 54, No. 1. – P. 316-326.
3. *Согомонян Е.С., Слабаков Е.В.* Самопроверяемые устройства и отказоустойчивые системы. – М.: Радио и связь. 1989. – 208 с.
4. *Mitra S., McCluskey E.J.* Which concurrent error detection scheme to choose? // Proceedings International Test Conference 2000. – IEEE, 2000. – P. 985-994.
5. *Хетагуров Я.А., Руднев Ю.П.* Повышение надёжности цифровых устройств методами избыточного кодирования. – М.: Энергия, 1974. – 270 с.
6. *Галлагер Р.* Коды с малой плотностью проверок на чётность. – М.: Мир, 1966. – 144 с.
7. Кодирование информации (двоичные коды). Справочник / под ред. проф. Н.Т. Березнюка. – Харьков: Вища школа, 1978.
8. *Косолапов Ю.В.* О применении схемы Озарова-Вайнера для защиты информации в беспроводных многоканальных системах передачи данных // Информационное противодействие угрозам терроризма. – 2007. – Т. 10. – С. 111-120.
9. *Sridhara D.* Low density parity check codes from permutation matrices // Proc. Conference on Information Sciences and Systems, John Hopkins University, USA, 2001. – P. 127-132.
10. *Овинников А.А.* Анализ свойств и параметров низкоплотностных кодов, синтезированных по алгоритму Таннера // Фундаментальные проблемы радиоэлектронного приборостроения. – 2014. – Т. 14, № 5. – С. 80-83.
11. *Иванов Ф.И., Зяблов В.В., Потапов В.Г.* Коды с малой плотностью проверок на чётность, основанные на полях Галуа // Информационные процессы. – 2012. – Т. 12, № 1. – С. 68-83.

12. McGowan J.A., Williamson R.C. Loop removal from LDPC codes // *Proceedings 2003 IEEE Information Theory Workshop*. – IEEE, 2003. – P. 230-233.
13. Fossorier M.P.C., Mihaljevic M., Imai H. Reduced complexity iterative decoding of low-density parity check codes based on belief propagation // *IEEE Transactions on communications*. – May 1999. – Vol. 47(5). – P. 673-680.
14. Белоголовый А.В., Крук Е.А. Многопороговое декодирование кодов с низкой плотностью проверок на четность // *Информационно-управляющие системы*. – 2005. – № 1.
15. Солтанов А.Г. Схемы декодирования и оценка эффективности LDPC-кодов. Применение, преимущества и перспективы развития // *Безопасность информационных технологий*. – 2010. – Т. 17, № 2. – С. 61-67.
16. URL: [http://icdm.ippm.ru/w/Схемы\\_ISCAS85/](http://icdm.ippm.ru/w/Схемы_ISCAS85/) (дата обращения: 20.05.2019).
17. URL: <https://people.engr.ncsu.edu/brglez/CBL/benchmarks/LGSynth89/> (дата обращения: 20.05.2019).
18. Тельпухов Д.В., Соловьев, Р.А., Тельпухова Н.В., & Щелоков А.Н. Оценка параметра логической чувствительности комбинационной схемы к однократным ошибкам с помощью вероятностных методов // *Известия ЮФУ. Технические науки*. – 2016. – № 7 (180). – С. 149-158.
19. Стемковский А.Л., Тельпухов Д.В., Соловьев Р.А., Мячиков М.В., Тельпухова Н.В. Разработка технологически независимых метрик для оценки маскирующих свойств логических схем // *Вычислительные технологии*. – 2016. – Т. 21 (2). – С. 53-62.
20. URL: <http://www.clifford.at/yosys/> (дата обращения: 20.05.2019).

## REFERENCES

1. Stempkovskiy A.L., Tel'pukhov D.V., Solov'ev R.A., Myachikov M.V. Povyshenie otkazoustoychivosti logicheskikh skhem s ispol'zovaniem nestandartnykh mazhoritarnykh elementov [Improving the fault tolerance of logic circuits using non-standard majority elements], *Informatsionnye tekhnologii* [Information technology], 2015, Vol. 21, No. 10, pp. 749-756.
2. El-Maleha A.H., Oughalia F.C. A generalized modular redundancy scheme for enhancing fault tolerance of combinational circuits, *Microelectronics Reliability*, 2014, Vol. 54, No. 1, pp. 316-326.
3. Sogomonyan E.S., Slabakov E.V. Samoproveryaemye ustroystva i otkazoustoychivye sistemy [Self-checking devices and fault-tolerant systems]. Moscow: Radio i svyaz', 1989, 208 p.
4. Mitra S., McCluskey E.J. Which concurrent error detection scheme to choose?, *Proceedings International Test Conference 2000*. IEEE, 2000, pp. 985-994.
5. Khetagurov Ya.A., Rudnev Yu.P. Povyshenie nadezhnosti tsifrovyykh ustroystv metodami izbytochnogo kodirovaniya [Improving the reliability of digital devices methods of redundant coding]. Moscow: Energiya, 1974, 270 p.
6. Gallager R. Kody s maloy plotnost'yu proverok na chetnost' [Codes with low density parity checks]. Moscow: Mir, 1966, 144 p.
7. Kodirovanie informatsii (dvoichnye kody). Spravochnik [Information encoding (binary codes). Handbook], under the ed. prof. N.T. Bereznyuka. Khar'kov: Vishcha shkola, 1978.
8. Kosolapov Yu.V. O primeneni skhemy Ozarova-Vaynera dlya zashchity informatsii v besprovodnykh mnogokanal'nykh sistemakh peredachi dannykh [About application of the scheme Azarova-Weiner for information protection in wireless multi-channel data transmission systems], *Informatsionnoe protivodeystvie ugrozam terrorizma* [Information counteraction to threats of terrorism], 2007, Vol. 10, pp. 111-120.
9. Sridhara D. Low density parity check codes from permutation matrices, *Proc. Conference on Information Sciences and Systems, John Hopkins University, USA, 2001*, pp. 127-132.
10. Ovinnikov A.A. Analiz svoystv i parametrov nizkplotnostnykh kodov, sintezirovannykh po algoritmu Tannera [Analysis of properties and parameters of low-density codes synthesized by the Tanner algorithm], *Fundamental'nye problemy radioelektronnoy priborostroeniya* [Fundamental problems of radioelectronic instrumentation], 2014, Vol. 14, No. 5, pp. 80-83.
11. Ivanov F.I., Zyablov V.V., Potapov V.G. Kody s maloy plotnost'yu proverok na chetnost', osnovannye na polyakh Galua [Low-density parity-check codes based on Galois field], *Informatsionnye protsessy* [Information process], 2012, Vol. 12, No. 1, pp. 68-83.
12. McGowan J.A., Williamson R.C. Loop removal from LDPC codes, *Proceedings 2003 IEEE Information Theory Workshop*. IEEE, 2003, pp. 230-233.

13. Fossorier M.P.C., Mihaljevic M., Imai H. Reduced complexity iterative decoding of low-density parity check codes based on belief propagation, *IEEE Transactions on communications*, May 1999, Vol. 47 (5), pp. 673-680.
14. Belogolovyy A.V., Kruk E.A. Mnogoporogovoe dekodirovanie kodov s nizkoy plotnost'yu proverok na chetnost' [Multithreshold decoding of codes with low density of parity checks], *Informatsionno-upravlyayushchie sistemy* [Information and control systems], 2005, No. 1.
15. Soltanov A.G. Skhemy dekodirovaniya i otsenka effektivnosti LDPC-kodov. Primenenie, preimushchestva i perspektivy razvitiya [Decoding schemes and efficiency evaluation of LDPC codes. Application, advantages and prospects of development], *Bezopasnost' informatsionnykh tekhnologiy* [Information technology security], 2010, Vol. 17, No. 2, pp. 61-67.
16. Available at: [http://icdm.ippm.ru/w/Схемы\\_ISCAS85/](http://icdm.ippm.ru/w/Схемы_ISCAS85/) (accessed 20 May 2019).
17. Available at: <https://people.engr.ncsu.edu/brglez/CBL/benchmarks/LGSynth89/> (accessed 20 May 2019).
18. Tel'pukhov D.V., Solov'ev, R.A., Tel'pukhova N.V., & ShChelokov A.N. Otsenka parametra logicheskoy chuvstvitel'nosti kombinatsionnoy skhemy k odnokratnym oshibkam s pomoshch'yu veroyatnostnykh metodov [Evaluation of parameter sensitivity a Boolean combinational circuit to single errors by using probabilistic methods], *Izvestiya YuFU. Tekhnicheskije nauki* [Izvestiya SFedU. Engineering Sciences], 2016, No. 7 (180), pp. 149-158.
19. Stempkovskiy A.L., Tel'pukhov D.V., Solov'ev R.A., Myachikov M.V., Tel'pukhova N.V. Razrabotka tekhnologicheskikh nezavisimyykh metrik dlya otsenki maskiruyushchikh svoystv logicheskikh skhem [Development of technology-independent metrics for evaluating masking properties of logical schemes], *Vychislitel'nye tekhnologii* [Computing technology], 2016, Vol. 21 (2), pp. 53-62.
20. Available at: <http://www.clifford.at/yosys/> (accessed 20 May 2019).

Статью рекомендовал к опубликованию к.т.н. А.В. Коршунов.

**Стемпковский Александр Леонидович** – Институт проблем проектирования в микроэлектронике РАН (ИППМ РАН); e-mail: stal09@ippm.ru; 124365, Москва, Зеленоград, ул. Советская, 3; тел.: +74997299890; научный руководитель института; профессор; академик РАН; д.т.н.

**Тельпухов Дмитрий Владимирович** – e-mail: nofrost@inbox.ru; отдел методологии проектирования интегральных схем; руководитель отдела; д.т.н.

**Жукова Татьяна Дмитриевна** – e-mail: zhukova\_t@ippm.ru; отдел систем автоматизированного проектирования интегральных схем; инженер–исследователь.

**Щелоков Альберт Николаевич** – e-mail: schan@ippm.ru; зам. директора; к.ф.-м.н.

**Новиков Александр Дмитриевич** – e-mail: a13x.student@yandex.ru; отдел методологии проектирования интегральных схем; стажер-исследователь.

**Гуров Сергей Исаевич** – Московский государственный университет им. М.В. Ломоносова; e-mail: sgur@cs.msu.ru; 119991, Москва, Ленинские горы, 1; тел.: +74959391000; к. ф.-м.н.; доцент.

**Stempkovskiy Alexandr Leonidovich** – The Institute for Design Problems in Microelectronics (IPPM RAS); e-mail: stal09@ippm.ru; 124365, Moscow, Zelenograd, Sovetskaya street, 3; phone: +74997299890; academic supervisor; professor, academician of RAS; dr. of eng. sc.

**Telpukhov Dmitry Vladimirovich** – e-mail: nofrost@inbox.ru; the department of integrated circuits design methodology; head of department; dr. of eng. sc.

**Zhukova Tatiana Dmitrievna** – e-mail: zhukova\_t@ippm.ru; the department of computer-aided IC design; research engineer.

**Schelokov Albert Nikolaevich** – e-mail: schan@ippm.ru; deputy director; cand. of ph.-m. sc.

**Novikov Alexander Dmitrievich** – e-mail: a13x.student@yandex.ru; the department of integrated circuits design methodology; research assistant.

**Gurov Sergey Isaevich** – Lomonosov Moscow State University; e-mail: sgur@cs.msu.ru; 119991, Moscow, Leninskie Gory, 1; phone: +74959391000; cand. of ph.-m. sc.; associate professor.