

СИНТЕЗ САМОПРОВЕРЯЕМЫХ ДИСКРЕТНЫХ УСТРОЙСТВ НА ОСНОВЕ ЛОГИЧЕСКОЙ КОРРЕКЦИИ СИГНАЛОВ И КОМПОЗИЦИИ РАВНОВЕСНЫХ КОДОВ «1 ИЗ 4» И «3 ИЗ 4».

Ч. 1. Метод синтеза с преобразованием всех сигналов от объекта диагностирования

Д. В. Ефанов*, Е. И. Елина**

***Санкт-Петербургский политехнический университет Петра Великого, г. Санкт-Петербург;

*Институт проблем транспорта им. Н. С. Соломенко РАН, г. Санкт-Петербург

*✉ TrES-4b@yandex.ru, **✉ eseniya-elina@mail.ru

Аннотация. Предложено при синтезе самопроверяемых дискретных устройств на основе логической коррекции сигналов использовать композицию равновесных кодов «1 из 4» и «3 из 4». Для данной композиции равновесных кодов можно реализовать простую и компактную самопроверяемую схему тестера, требующую для полной проверки всего четырех тестовых комбинаций. Описана структура самопроверяемого дискретного устройства. Подразумевается преобразование всех четырех сигналов от объекта диагностирования при синтезе для него схемы встроенного контроля. Разработан простейший алгоритм синтеза самопроверяемой схемы встроенного контроля на основе логической коррекции сигналов и композиции равновесных кодов «1 из 4» и «3 из 4». Приведены результаты экспериментов с тестовыми комбинационными схемами, демонстрирующие преимущество предлагаемого метода перед известным методом дублирования и перед методом, основанным на применении равновесного кода «1 из 4» совместно с логической коррекцией сигналов, по показателям структурной избыточности конечного самопроверяемого дискретного устройства. Использование разработанного авторами метода перспективно при синтезе самопроверяемых дискретных устройств на разной элементной базе в различных сферах применения.

Ключевые слова: самопроверяемое дискретное устройство, схема встроенного контроля, логическая коррекция сигналов, композиция равновесных кодов «1 из 4» и «3 из 4», структурная избыточность, контролепригодная структура.

ВВЕДЕНИЕ

Часто при синтезе самопроверяемых дискретных устройств применяют методы, основанные на внесении избыточности не в сам исходный объект (объект диагностирования), а в специализированную схему встроенного контроля (СВК) [1–3]. Такая схема строится с учетом особенностей объекта диагностирования и выбираемого диагностического признака для фиксации неисправностей и ошибок в вычислениях. По сути, реализуется система

рабочего диагностирования для заданного объекта, которая косвенно, по результатам вычисления им своих функций, фиксирует факт наличия или отсутствия неисправностей [4]. Схемы встроенного контроля реализуются самопроверяемыми, что позволяет исключить проблему «сторожа над сторожем», когда разработчику требуется задумываться над тем, как обеспечить контроль исправности самой СВК. Полностью самопроверяемые СВК являются самотестируемыми и защищенными от неисправностей [5]. Первое свойство подразумева-



ет существование для всех неисправностей СВК из заданного класса хотя бы одного набора значений аргументов, при подаче которого на ее входы устанавливается контрольный сигнал ошибки. Второе свойство говорит о том, что при возникновении любой неисправности в СВК из заданного класса на ее выходах либо вычисляются верные значения, либо фиксируется сигнал ошибки.

Существуют два основных подхода к синтезу СВК, каждый из которых порождает множество методов. Первый из них является классическим и известен с середины прошлого столетия [6]. Он подразумевает дополнение информационных сигналов, формируемых на параллельных выходах объекта диагностирования при подаче на входы каждого из наборов значений аргументов, в СВК контрольными сигналами таким образом, чтобы в исправном состоянии объекта диагностирования было установлено некое соответствие между информационными и контрольными сигналами. Зачастую это делается с использованием диагностического признака принадлежности кодовых слов, образуемых информационными и контрольными сигналами, некоторому наперед заданному равномерному блоковому коду [7–9]. Неисправности объекта диагностирования в процессе его эксплуатации вызывают ошибки в вычислениях, что приводит к нарушению соответствия между значениями информационных и контрольных сигналов, что, в свою очередь, и фиксируется в СВК. Второй подход известен с конца прошлого столетия – начала нынешнего [10–13]. Он подразумевает не дополнение информационных сигналов контрольными, а преобразование всех или части сигналов таким образом, чтобы на каждом наборе значений аргументов было зафиксировано требуемое кодовое слово с заданными диагностическими свойствами.

Оба подхода имеют свои преимущества и недостатки. Например, подход, основанный на дополнении сигналов, позволяет учитывать напрямую свойства обнаружения ошибок равномерными блоковыми кодами и использовать особые схемотехнические способы, позволяющие всегда обнаруживать ошибки определенных видов в СВК [14, 15]. При этом, однако, для заданного элементного базиса и способа построения компонентов СВК ее структура оказывается единственной, и разработчик не может повлиять на характеристики эффективности реализации конечного самопроверяемого дискретного устройства (например, на показатели структурной избыточности, контролепригодности, энергопотребления и пр.). Второй же подход, основанный на логической коррекции сигналов (ЛКС), позволяет на каждом наборе значений ар-

гументов выбрать кодовое слово, в которое преобразуется двоичный вектор с выходов объекта диагностирования, из некоторого множества, что определяет широкие возможности для построения различных СВК даже для заданных элементной базы и методов их реализации [16]. При этом можно решать оптимизационные задачи для того или иного показателя эффективности СВК. Но из-за использования ЛКС для вектора, формируемого на выходах объекта диагностирования, оказывается невозможным применение напрямую известных схемотехнических приемов для покрытия отдельных видов ошибок в СВК. Требуются иные подходы, например, выделение подмножеств выходов с особыми используемыми при организации контроля свойствами (например, на которых невозможно возникновение ошибок с заданной их кратностью d (одно-, двух-, трехкратных и т. д.)) [17]. Во всех этих случаях существуют два варианта построения СВК, первый из которых подразумевает полное покрытие ошибок на выходах объекта диагностирования, вызванных неисправностями в его внутренней структуре, а второй – проявление неисправностей хотя бы на одном наборе значений аргументов [18, 19].

Наиболее просто при использовании второго подхода реализуются СВК, основанные на применении неразделимых блоковых кодов. К ним относятся равновесные коды [20], коды Плоткина (Адамара) [21], коды Бордена [22] и другие разнообразные композиции равновесных кодов [23]. Особенностью методов, использующих такие коды, является то, что при организации СВК можно легко схемотехническими приемами добиться исключения ошибок определенного вида в кодовых словах, тогда как для разделимых кодов приходится еще учитывать потенциальные одновременные искажения и информационных, и проверочных символов [24].

Исследования авторов настоящей статьи показали, что помимо традиционных равновесных кодов существует особая их композиция, образованная путем объединения кодовых слов, принадлежащих кодам «1 из 4» и «3 из 4» [23]. Для нее может быть построен тестер с простой и компактной структурой, для полной проверки которого требуются всего четыре тестовые комбинации. Это минимальное значение мощности множества кодовых слов, образующих проверяющий тест, для самопроверяемых тестеров [5].

Данная работа посвящена описанию методов синтеза СВК на основе ЛКС с применением композиции равновесных кодов «1 из 4» и «3 из 4». Она состоит из двух частей: в первой авторы представляют структуру организации СВК на основе

ЛКС с преобразованием сигналов со всех выходов объекта диагностирования и простейший алгоритм синтеза самопроверяемой СВК, а во второй приводятся методы, позволяющие снижать структурную избыточность путем использования свойств композиции равновесных кодов «1 из 4» и «3 из 4» и уменьшения числа элементов преобразования в СВК.

1. ОРГАНИЗАЦИЯ СХЕМЫ ВСТРОЕННОГО КОНТРОЛЯ НА ОСНОВЕ КОМПОЗИЦИИ РАВНОВЕСНЫХ КОДОВ «1 ИЗ 4» И «3 ИЗ 4»

Структура организации СВК на основе композиции равновесных кодов «1 из 4» и «3 из 4» изображена на рис. 1. Она строится для множества из четырех выходов объекта диагностирования. Выбор множества выходов мощностью 4 связан с длиной кодового вектора композиции равновесных кодов «1 из 4» и «3 из 4».

Объектом диагностирования является комбинационная часть дискретного устройства, или просто комбинационное дискретное устройство $F(X)$, которое вычисляет значения булевых функций $f_1(X), f_2(X), f_3(X), f_4(X)$ при подаче на входы заданного множества наборов значений аргументов $\langle X \rangle = \langle x_i x_{i-1} \dots x_2 x_1 \rangle$ (далее полагается, что это множество полное). Таким образом, на каждом наборе значений аргументов формируется булев вектор $\langle f_4(X) f_3(X) f_2(X) f_1(X) \rangle$. Для контроля корректности вычислений обустраивается специализированная СВК. Она состоит из трех функциональных блоков: блока $G(X)$ вычисления значений функций коррекции, непосредственно блока коррекции сигналов (БКС) и самопроверяемого тесте-

ра композиции равновесных кодов «1 из 4» и «3 из 4» TSC (*totally self-checking checker*).

Блок $G(X)$ предназначен для вычисления значений функций $g_1(X), g_2(X), g_3(X), g_4(X)$ для коррекции сигналов от объекта диагностирования $F(X)$ при подаче на входы обоих блоков одинаковых наборов значений аргументов. Каждая из функций коррекции сигналов предназначена для реализации в БКС преобразования вида

$$h_i(X) = f_i(X) \oplus g_i(X), \quad i = \overline{1, 4}, \quad (1)$$

где $h_i(X)$ – скорректированное значение на выходе БКС, получаемое при подаче на входы конкретного набора значений аргументов.

Для коррекции сигналов используются двухвходовые элементы сложения по модулю два (элементы XOR). Их использование позволяет преобразовать любое значение 0 (1) в любое из значений 0 (1)¹. Тем самым, на каждом наборе значений аргументов можно преобразовать кодовый вектор $\langle f_4(X) f_3(X) f_2(X) f_1(X) \rangle$ в кодовый вектор $\langle h_4(X) h_3(X) h_2(X) h_1(X) \rangle$ с заданными диагностическими свойствами. В представленной на рис. 1 структуре осуществляется преобразование в кодовое слово, принадлежащее композиции равновесных кодов «1 из 4» и «3 из 4». Для проверки принадлежности кодового слова $\langle h_4(X) h_3(X) h_2(X) h_1(X) \rangle$ заданной композиции устанавливается блок TSC . Он снабжен двумя выходами $z^0(X)$ и $z^1(X)$, которые функционируют в парафазной логике: комбинации $\langle 01 \rangle$ и $\langle 10 \rangle$ свидетельствуют об отсутствии ошибок в вычислениях, а комбинации $\langle 00 \rangle$ и $\langle 11 \rangle$ говорят о присутствии ошибки и, косвенно, о наличии неисправности в объекте диагностирования.

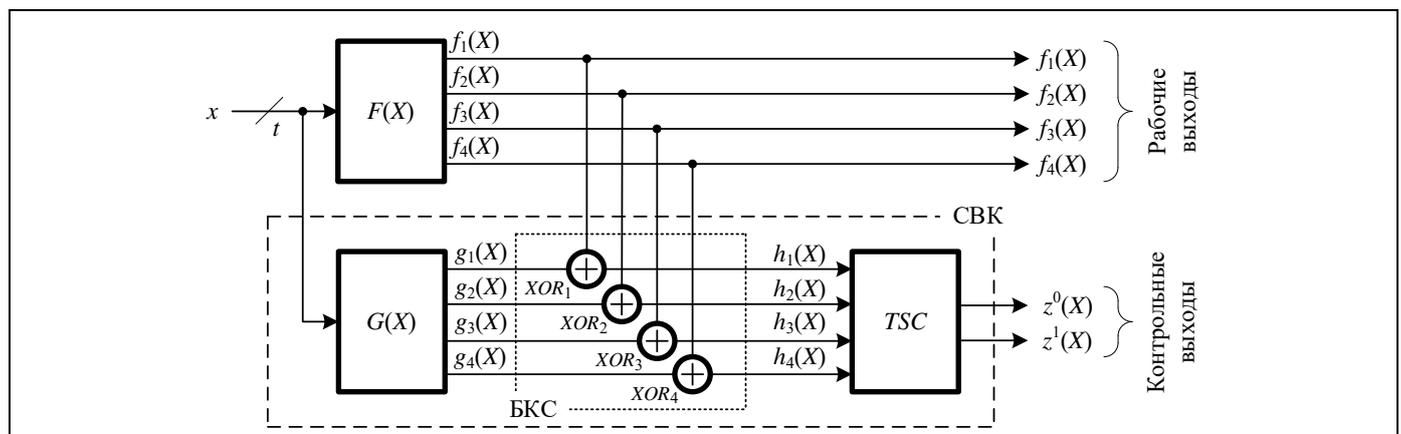


Рис. 1. Структура организации СВК

¹ Аналогичным свойством обладает функция равнозначности, реализуемая элементом $XNOR$. Ее также можно было бы использовать при синтезе СВК на основе ЛКС. Другие элементарные булевы функции можно применять при ЛКС только с рядом ограничений.



На рис. 2 приведена структура простейшего тестера композиции равновесных кодов «1 из 4» и «3 из 4».

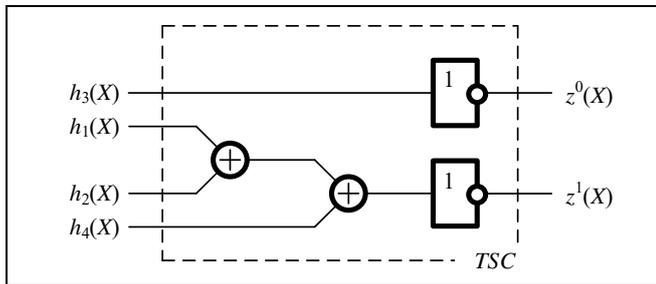


Рис. 2. Структура тестера композиции равновесных кодов «1 из 4» и «3 из 4»

Проверяющий тест T_n для изображенной на рис. 2 структуры тестера содержит комбинации из следующего множества:

$$T_n = \{1000, 0010, 1101, 0111\} \cup \{0001, 1011, 0100, 1110\}. \quad (2)$$

При синтезе СВК достаточно обеспечить формирование хотя бы одного из подмножеств из множества (2). Это позволит в процессе эксплуатации самопроверяемого устройства осуществлять полную проверку тестера TSC.

Структура, приведенная на рис. 1, таким образом, будет самопроверяемой.

2. ПРОСТЕЙШИЙ АЛГОРИТМ ПОСТРОЕНИЯ СХЕМЫ ВСТРОЕННОГО КОНТРОЛЯ НА ОСНОВЕ КОМПОЗИЦИИ РАВНОВЕСНЫХ КОДОВ «1 ИЗ 4» И «3 ИЗ 4»

Преобразование вида (1) осуществляется на каждом наборе значений аргументов. Преобразование вектора $\langle f_4(X) f_3(X) f_2(X) f_1(X) \rangle$ в кодовый вектор $\langle h_4(X) h_3(X) h_2(X) h_1(X) \rangle$ можно осуществить восемью разными способами, так как мощность множества кодовых слов композиции равновесных кодов «1 из 4» и «3 из 4» равна $C_4^1 + C_4^3 = 8$. Отсюда следует, что при t аргументах преобразование вида (1) может быть осуществлено 8^2 способами. С учетом необходимости формирования множества T_n для тестера это $8^2 - 3$ способа. Уже при $t = 3$ это дает 16 777 213 способов синтеза самопроверяемых СВК. Среди этих способов можно выбрать наилучший, например, по критерию минимума структурной избыточности самопроверяемого дискретного устройства.

Предложим, однако, здесь простой алгоритм построения СВК на основе ЛКС и композиции

равновесных кодов «1 из 4» и «3 из 4», обеспечивающий формирование множества T_n для тестера, но требующий постпроверки формирования тестовых комбинаций для элементов XOR. Будем исходить из того, что все функции объекта диагностирования определены полностью и на входы поступают все 2^t наборов значений аргументов. Отметим также, что на начальном этапе известны структуры БКС и TSC, а также задана логика работы $F(X)$; значения же функций, реализуемых блоком $G(X)$, не установлены. Основная цель синтеза – получить значения этих функций на каждом наборе значений аргументов, а также обеспечить формирование проверяющих тестов для элементов БКС и TSC.

Алгоритм 1. Синтез СВК для четверки выходов объекта диагностирования:

1. Определить число $\delta = \frac{2^t}{8} = 2^{t-3}$, характеризующее количество каждого из используемых слов рассматриваемой композиции равновесных кодов $\langle h_4(X) h_3(X) h_2(X) h_1(X) \rangle$.

Это позволит построить СВК, в которой будет обеспечено равномерное формирование кодовых слов композиции равновесных кодов «1 из 4» и «3 из 4» на входах TSC.

2. Рассмотреть наборы значений аргументов в лексикографическом порядке от набора с десятичным эквивалентом 0 до набора с десятичным эквивалентом $2^t - 1$. На наборах значений аргументов с десятичными эквивалентами $0 \dots (2^{t-3} - 1)$ доопределить разряды векторов $\langle h_4(X) h_3(X) h_2(X) h_1(X) \rangle$ до кодового слова $\langle 0001 \rangle$; на наборах с номерами $2 \cdot 2^{t-3} \dots (2 \cdot 2^{t-3} - 1)$ – до кодового слова $\langle 0010 \rangle$; на наборах с номерами $2 \cdot 2^{t-3} \dots (3 \cdot 2^{t-3} - 1)$ – до кодового слова $\langle 0100 \rangle$; на наборах с номерами $3 \cdot 2^{t-3} \dots (4 \cdot 2^{t-3} - 1)$ – до кодового слова $\langle 1000 \rangle$; на наборах с номерами $4 \cdot 2^{t-3} \dots (5 \cdot 2^{t-3} - 1)$ – до кодового слова $\langle 1110 \rangle$; на наборах с номерами $5 \cdot 2^{t-3} \dots (6 \cdot 2^{t-3} - 1)$ – до кодового слова $\langle 1101 \rangle$; на наборах с номерами $6 \cdot 2^{t-3} \dots (7 \cdot 2^{t-3} - 1)$ – до кодового слова $\langle 1011 \rangle$; наконец, на наборах с номерами $7 \cdot 2^{t-3} \dots (8 \cdot 2^{t-3} - 1)$ – до кодового слова $\langle 0111 \rangle$. Тем самым обеспечивается формирование каждого из кодовых слов композиции равновесных кодов «1 из 4» и «3 из 4» на входах TSC ровно по δ раз.

3. Определить значения функций $g_i(X)$ по известным значениям функций $h_i(X)$ на каждом наборе значений аргументов:

$$g_i(X) = f_i(X) \oplus h_i(X) \Leftrightarrow h_i(X) = f_i(X) \oplus g_i(X), \quad i = \overline{1, 4}. \quad (3)$$

4. Произвести проверку формирования проверяющего теста для каждого из элементов XOR в СВК: в проверяющий тест при канонической ре-

лизации XOR входят все четыре рабочих комбинации {00, 01, 10, 11} [25], каждая из которых должна быть сформирована хотя бы на одном наборе значений аргументов. Если тесты формируются для всех элементов БКС, то перейти к п. 5 алгоритма; в противном случае использовать переопределение значений разрядов в векторе $\langle h_4(X) h_3(X) h_2(X) h_1(X) \rangle$ на выборочном числе наборов значений аргументов в соответствии с методом, изложенным в работе [26].

5. Оптимизировать функции $g_4(X)$, $g_3(X)$, $g_2(X)$, $g_1(X)$ [27].

6. Синтезировать самопроверяемое дискретное устройство в выбранном элементном базисе.

Обратим внимание на п. 2 алгоритма, в котором подразумевается использование всех восьми кодовых слов, принадлежащих заданной композиции. Однако, учитывая выражение (2), можно изменить этот пункт и доопределять векторы $\langle h_4(X) h_3(X) h_2(X) h_1(X) \rangle$ только до четырех кодовых слов заданной композиции, которые обеспечат полную проверку тестера.

Пример 1. Синтезируем СВК для устройства, логика работы которого задана в первых девяти столбцах табл. 1.

В соответствии с п. 1 алгоритма 1 определяем число $\delta = 2^{4-3} = 2$, характеризующее количество каждого из

используемых слов композиции равновесных кодов «1 из 4» и «3 из 4». Затем, согласно п. 2 алгоритма 1, рассматриваем последовательно наборы значений аргументов и фиксируем на них следующие кодовые слова композиции равновесных кодов «1 из 4» и «3 из 4» в векторе $\langle h_4(X) h_3(X) h_2(X) h_1(X) \rangle$: на наборах 0, 1 – слово $\langle 0001 \rangle$, на наборах 2, 3 – слово $\langle 0010 \rangle$, на наборах 4, 5 – слово $\langle 0100 \rangle$, на наборах 6, 7 – слово $\langle 1000 \rangle$, на наборах 8, 9 – слово $\langle 1110 \rangle$, на наборах 10, 11 – слово $\langle 1101 \rangle$, на наборах 12, 13 – слово $\langle 1011 \rangle$, на наборах 14, 15 – слово $\langle 0111 \rangle$ (см. столбцы, соответствующие разрядам вектора $\langle h_4(X) h_3(X) h_2(X) h_1(X) \rangle$ в табл. 1). Пользуясь п. 3 алгоритма 1 и формулой (3), получаем значения функций, реализуемых на выходах блока $G(X)$, на каждом наборе значений аргументов (см. столбцы, соответствующие разрядам вектора $\langle g_4(X) g_3(X) g_2(X) g_1(X) \rangle$ в табл. 1). Следуя за п. 4 алгоритма 1, определяем, формируются ли проверяющие тесты для каждого из элементов XOR_4 , XOR_3 , XOR_2 , XOR_1 . Все комбинации, формирующиеся на их входах, приведены в соответствующих столбцах табл. 1. Анализ комбинаций, поступающих на входы каждого элемента коррекции, говорит о том, что проверяющий тест для каждого из них формируется.

Далее следовало бы выполнить п. 5 и 6 алгоритма 1, но они представляются тривиальными: оптимизируются функции $g_4(X)$, $g_3(X)$, $g_2(X)$, $g_1(X)$ и реализуется самопроверяемое дискретное устройство в выбранном элементном базисе. ♦

Таблица 1

Описание работы устройства с четырьмя выходами и СВК для него

№	Наборы значений аргументов				Значения на выходах $F(X)$				Значения на выходах БКС				Значения на выходах $G(X)$				Тестовые комбинации БКС			
	x_4	x_3	x_2	x_1	$f_4(X)$	$f_3(X)$	$f_2(X)$	$f_1(X)$	$h_4(X)$	$h_3(X)$	$h_2(X)$	$h_1(X)$	$g_4(X)$	$g_3(X)$	$g_2(X)$	$g_1(X)$	XOR_4	XOR_3	XOR_2	XOR_1
0	0	0	0	0	1	1	0	0	0	0	0	1	1	1	0	1	11	11	00	01
1	0	0	0	1	0	0	1	0	0	0	0	1	0	0	1	1	00	00	11	01
2	0	0	1	0	1	0	1	0	0	0	1	0	0	0	1	0	10	00	11	00
3	0	0	1	1	0	0	0	0	0	0	1	0	1	0	0	0	01	00	00	00
4	0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	00	10	00	00
5	0	1	0	1	0	0	1	0	0	1	0	0	0	1	1	0	00	01	11	00
6	0	1	1	0	0	1	0	0	1	0	0	0	0	1	1	0	00	11	01	00
7	0	1	1	1	0	0	0	1	1	0	0	0	0	0	1	1	00	00	01	11
8	1	0	0	0	1	0	1	0	1	1	1	0	1	1	1	0	11	01	11	00
9	1	0	0	1	1	0	0	0	1	1	1	0	1	1	0	0	11	01	00	00
10	1	0	1	0	0	1	1	0	1	1	0	1	1	1	1	0	01	11	11	00
11	1	0	1	1	0	0	0	1	1	1	0	1	1	1	0	0	01	01	00	10
12	1	1	0	0	1	0	1	1	1	0	1	1	0	0	0	0	10	00	10	10
13	1	1	0	1	1	0	1	0	1	0	1	1	0	0	0	1	10	00	10	01
14	1	1	1	0	1	1	1	1	0	1	1	1	1	0	0	0	11	10	10	10
15	1	1	1	1	0	1	0	1	0	1	1	1	0	0	1	0	00	10	01	10



При синтезе СВК для устройства с $n > 4$ выходами используется

Алгоритм 2. Синтез СВК для объекта диагностирования с числом выходов $n > 4$:

1. Упорядочить выходы устройства и разбить их на $q = \left\lceil \frac{n}{4} \right\rceil$ подмножеств (запись $\lceil \dots \rceil$ обозначает целое сверху от вычисляемого значения). Причем, при $n(\bmod 4) = 0$, в каждое из q подмножеств войдут уникальные выходы; если же $n(\bmod 4) \neq 0$, то в последнем подмножестве будут присутствовать $n(\bmod 4)$ уникальных выходов и $4 - n(\bmod 4)$ уже использованных в $q - 1$ подмножестве.

2. Синтезировать СВК по алгоритму 1 для каждого из q подмножеств выходов. При этом для сокращения избыточности отдельные блоки $G(X)$ каждой СВК следует синтезировать совместно.

3. Подключить выходы q СВК к входам самопроверяемого компаратора $qTRC1$, осуществляющего сжатие q парафазных сигналов в один и строящегося на основе $q - 1$ элементарного модуля сжатия парафазных сигналов TRC (*two-rail checker*) [28–30].

Пример 2. Привести структуру самопроверяемого дискретного устройства для организации СВК для исходного устройства с $n = 10$ выходами.

В соответствии с п. 1 алгоритма 2 упорядочиваем выходы и разбиваем их на следующее количество подмножеств: $q = \left\lceil \frac{10}{4} \right\rceil = 3$. Первые два подмножества не пересекаются: I – $\{f_1(X), f_2(X), f_3(X), f_4(X)\}$, II – $\{f_5(X), f_6(X), f_7(X), f_8(X)\}$. Так как $10(\bmod 4) = 2$, в последнее подмножество включаются два выхода из второго подмножества и два оставшихся ранее не использованных выходов: III – $\{f_7(X), f_8(X), f_9(X), f_{10}(X)\}$.

На рис. 3 приведена структура самопроверяемого дискретного устройства в соответствии с данными примера 2. Для ее реализации потребовалось фактически выделить три СВК с отдельными БКС и тестерами. Контрольные выходы отдельных СВК подключены к входам самопроверяемого компаратора, реализованного на основе двух модулей TRC . Так как выходы $f_7(X)$ и $f_8(X)$ использованы дважды, соответствующие функции в СВК отмечены знаками «*» и «**». ♦

Основное преимущество разработанных алгоритмов состоит в том, что они максимально просто позволяют синтезировать СВК для исходных объектов диагностирования, а процедуры, реализуемые в алгоритмах 1 и 2, можно легко автоматизировать.

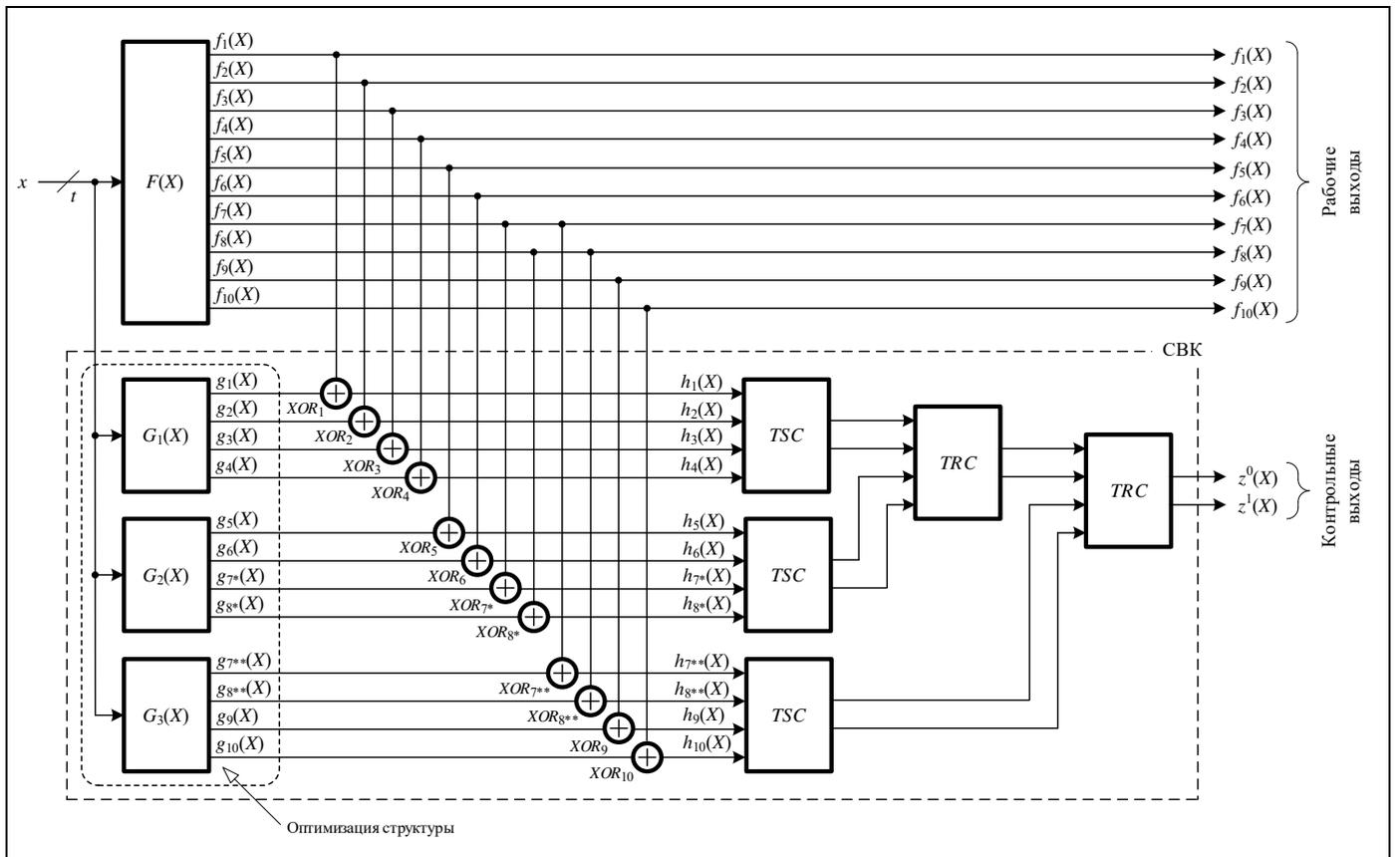


Рис. 3. Структура самопроверяемого дискретного устройства к примеру 2

ровать для использования в системах автоматизированного проектирования самопроверяемых дискретных устройств.

3. РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТОВ

Важной задачей исследования было экспериментальное подтверждение эффективности применения структуры, приведенной на рис. 1, и алгоритмов 1 и 2 при синтезе самопроверяемых дискретных устройств для набора тестовых комбинационных схем MCNC Benchmarks [31, 32]. В ходе экспериментов оценивались показатели структурной избыточности самопроверяемых дискретных устройств, синтезируемых для каждой тестовой схемы. Для сравнения применялись также два метода, в которых подразумевается такое же суммарное количество выходов у блоков $G(X)$, что и у объекта диагностирования: первый – классический метод дублирования, где все блоки $G(X)$ фактически заменены единственной копией объекта диагностирования или устройством, вычисляющим эквивалентные булевы функции [6] (данный метод широко применяется в настоящее время при синтезе самопроверяемых реализаций дискретных устройств [33]), а второй – это метод, основанный на ЛКС и применении равновесного кода «1 из 4» с коррекцией всех сигналов от объекта диагностирования [34].

Для выбранных тестовых схем синтезировались СВК по трем методам. Были выбраны тестовые комбинационные схемы, представленные в формате *.pla. В нем схема задана описанием булевых функций в интервальной форме в виде фактически сжатой таблицы истинности, представляющей собой троичную матрицу с элементарными конъюнкциями [35]. Процедуры обработки тестовых схем, а также разработанные алгоритмы были автоматизированы и добавлены в виде кода в программный модуль DM Coding², что позволило получить описания в формате *.pla для функциональных блоков СВК. Далее с использованием известного интерпретатора SIS [36, 37] и библиотеки функциональных элементов stdcell2_2.genlib были синтезированы самопроверяемые устройства и определены некоторые их параметры. При синтезе

для блоков $G(X)$ использовалась процедура оптимизации булевых функций full_simplify (остальные блоки СВК являются типовыми). Данная процедура позволяет оптимизировать систему булевых функций с применением бинарных диаграмм решений (англ. *binary decision diagram*, BDD) [38]. Затем осуществлялась процедура map -s, позволяющая произвести структурный синтез устройств в заданной библиотеке функциональных элементов и вывести основные их параметры. Для сравнения рассматривался параметр, который характеризует сложность технической реализации устройства, – параметр условной площади, занимаемой устройством на кристалле (в условных единицах библиотеки элементов). Для каждого компонента СВК были определены показатели сложности реализации, а также итоговые значения показателя сложности реализации самопроверяемого устройства (по аналогии с методикой, описанной в работе [39]). Для метода дублирования это величина L_D , для метода на основе ЛКС и кода «1 из 4» – $L_{1/4}$, для метода, описанного в настоящей статье, – $L_{1/4+3/4}$. Для удобства сравнения всех трех методов между собой были рассчитаны следующие относительные показатели:

$$\mu = \frac{L_{1/4+3/4}}{L_D} \cdot 100 \%, \quad (4)$$

$$\eta = \frac{L_{1/4+3/4}}{L_{1/4}} \cdot 100 \%. \quad (5)$$

Показатели (4) и (5) характеризуют самопроверяемое устройство, синтезированное по предложенному авторами методу, в сравнении с самопроверяемыми устройствами, синтезированными на основе дублирования и кода «1 из 4». Если $\mu > 100 \%$, то разработанный в настоящей статье метод эффективнее дублирования, если $\eta > 100 \%$, то разработанный метод эффективнее применения кода «1 из 4». Также для удобства анализа эффективности применения нового метода были рассчитаны показатели, характеризующие выигрыш в показателе сложности по сравнению с тем или иным методом:

$$\Delta\mu = \left(1 - \frac{L_{1/4+3/4}}{L_D}\right) \cdot 100 \%,$$

$$\Delta\eta = \left(1 - \frac{L_{1/4+3/4}}{L_{1/4}}\right) \cdot 100 \%.$$

Значение $\Delta\mu > 0 \%$ свидетельствует о выигрыше в показателе сложности реализации перед дублированием; значение $\Delta\eta > 0 \%$ – о выигрыше в показателе сложности реализации перед использованием ЛКС и кода «1 из 4».

² Программный модуль DM Coding разработан коллективом научной группы авторов настоящей статьи и канд. техн. наук В. В. Дмитриевым, является наращиваемым и предназначен для постановки экспериментов по разрабатываемым методам технической диагностики. Он написан на языке C# и позволяет реализовывать некоторые функции по анализу тестовых схем в соответствии с разрабатываемыми алгоритмами, в том числе дает возможность получения функционального описания блоков СВК, синтезируемых по различным методам.



В табл. 2 сведены результаты экспериментов для ряда тестовых комбинационных схем. Для каждой тестовой схемы определен показатель сложности реализации $L_{F(X)}$, а также показатели L_D , $L_{1/4}$ и $L_{1/4+3/4}$. Кроме того, для методов, основанных на ЛКС, приведены также данные по значениям показателя сложности реализации блока $G(X)$ – показателя $L_{G(X)}$. При этом для схем с числом выходов $n \geq 5$ блок $G(X)$ получался путем объединения отдельных блоков вычисления значений функций коррекции сигналов каждой из подсхем контроля и синтеза в виде единой схемы. Сравнивая между собой данные показатели для разных методов, основанных на ЛКС, можно оценить влияние показателя сложности реализации блока вычисления функций коррекции на итоговое значение показателя сложности реализации для самопроверяемого дискретного устройства. Отдельно на рис. 4 демонстрируются значения показателей $\Delta\mu$ и $\Delta\eta$ в виде столбчатых диаграмм.

Анализируя данные табл. 2 и рис. 4, можно установить следующее. Для 9-ти тестовых схем из 28-ми оказалось эффективнее использовать метод дублирования. Однако для большинства схем – 19-ти (это составляет примерно 68 % всех схем) – с использованием разработанного метода были получены самопроверяемые дискретные устройства с

меньшими значениями показателей сложности реализации. Причем выигрыш для некоторых схем составил $\Delta\mu > 20\%$. В сравнении с методом, изложенным в работе [34], новый метод дал эффект для 16-ти схем (это составляет примерно 57 % всех схем). Улучшение же по показателю $\Delta\eta$ составило для большинства схем не более 10 %. Это объясняется близостью друг к другу разработанного в настоящем исследовании метода и метода из работы [34] по количеству корректируемых сигналов. Эффект достигался в основном из-за выигрыша в показателе $L_{G(X)}$ при применении того или иного метода. Как следует из рис. 4, для трех тестовых схем получен существенный проигрыш в показателе сложности реализации – свыше 20 % по обоим показателям $\Delta\mu$ и $\Delta\eta$. Это связано со сложностью булевых функций, реализуемых блоками $F(X)$ и $G(X)$ [40]. В отдельно взятых случаях сложность булевых функций, реализуемых блоком $G(X)$, оказалась значительно превышающей сложность булевых функций, реализуемых блоком $F(X)$ (читатель может построчно сравнить данные столбцов $L_{F(X)}$ и $L_{G(X)}$). Но, несмотря на это, для большинства тестовых схем получен положительный эффект при использовании разработанного метода, на основе чего можно судить о его применимости на практике.

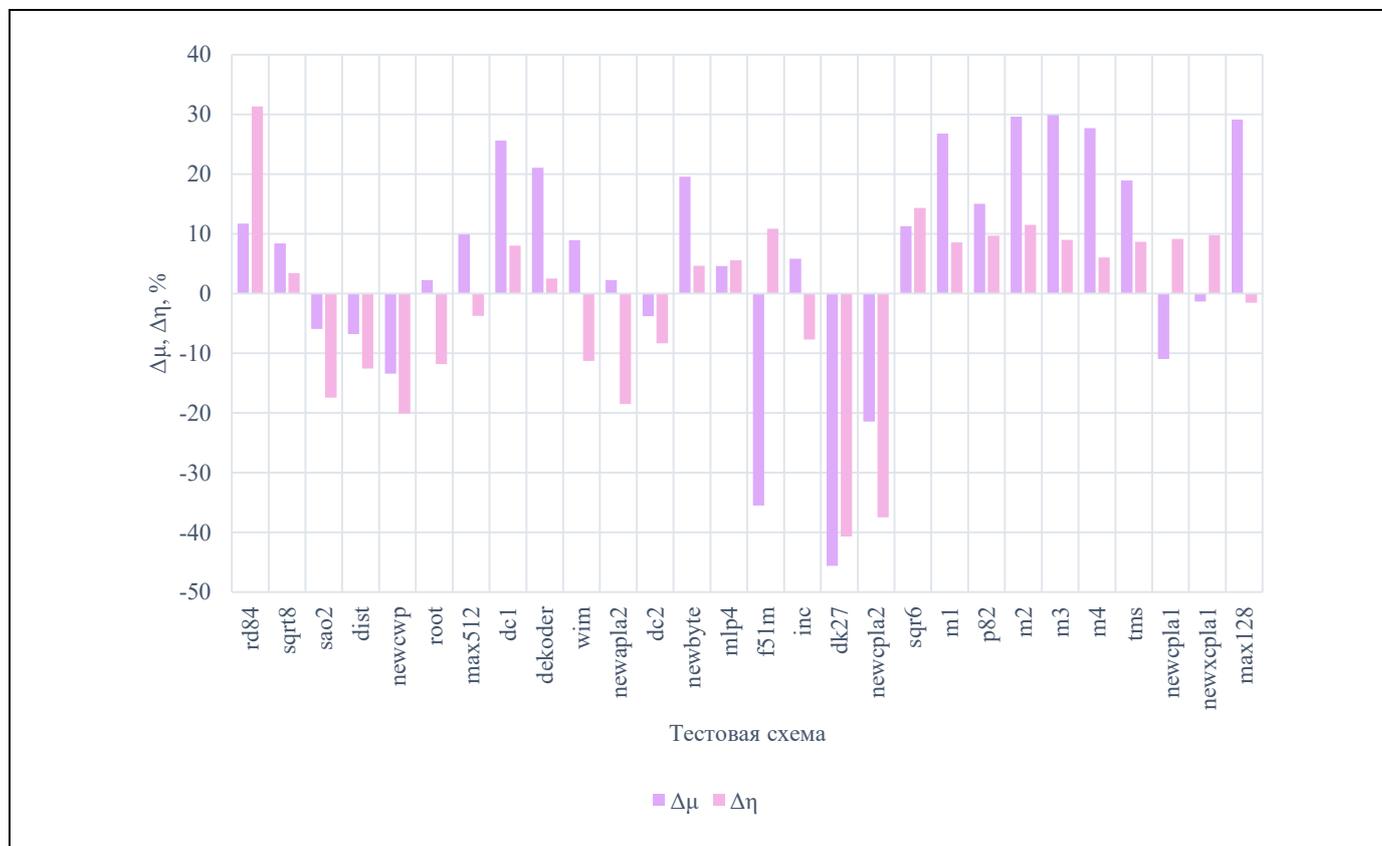


Рис. 4. Значения показателей $\Delta\mu$ и $\Delta\eta$ для различных тестовых схем



Результаты экспериментов

Тестовая схема	n	q	$L_{F(x)}$	L_D	Разработанный метод		Метод из работы [34]		μ	η	$\Delta\mu$	$\Delta\eta$
					$L_{G(x)}$	$L_{1/4+3/4}$	$L_{G(x)}$	$L_{1/4}$				
rd84	4	1	4912	10464	4056	9240	8128	13456	88,303	68,668	11,697	31,332
sqrt8	4	1	1160	2960	1280	2712	1232	2808	91,622	96,581	8,378	3,419
sao2	4	1	2992	6624	3752	7016	2568	5976	105,918	117,403	-5,918	-17,403
dist	5	2	6968	14784	8080	15784	6448	14024	106,764	112,55	-6,764	-12,55
newcwp	5	2	440	1728	784	1960	584	1632	113,426	120,098	-13,426	-20,098
root	5	2	3496	7840	3432	7664	2752	6856	97,755	111,785	2,245	-11,785
max512	6	2	9632	20320	7944	18312	7224	17656	90,118	103,715	9,882	-3,715
dc1	7	2	976	3216	680	2392	632	2600	74,378	92	25,622	8
dekoder	7	2	736	2736	688	2160	488	2216	78,947	97,473	21,053	2,527
wim	7	2	712	2688	1000	2448	496	2200	91,071	111,273	8,929	-11,273
newapla2	7	2	600	2464	1072	2408	480	2032	97,727	118,504	2,273	-18,504
dc2	7	2	2424	6112	3184	6344	2440	5856	103,796	108,333	-3,796	-8,333
newbyte	8	2	592	2656	808	2136	384	2240	80,422	95,357	19,578	4,643
mlp4	8	2	7224	15920	7232	15192	7520	16088	95,427	94,431	4,573	5,569
f51m	8	2	2272	6016	5144	8152	5528	9144	135,505	89,151	-35,505	10,849
Inc	9	3	2376	6432	2480	6056	1712	5624	94,154	107,681	5,846	-7,681
dk27	9	3	528	2736	2256	3984	768	2832	145,614	140,678	-45,614	-40,678
newcpla2	10	3	1896	5680	3800	6896	1392	5016	121,408	137,48	-21,408	-37,48
sqr6	12	3	2648	7600	2896	6744	2672	7872	88,737	85,671	11,263	14,329
m1	12	3	3064	8432	1912	6176	1096	6752	73,245	91,469	26,755	8,531
p82	14	4	2368	7456	2304	6336	1712	7016	84,979	90,308	15,021	9,692
m2	16	4	10096	23328	4656	16416	4288	18544	70,37	88,525	29,63	11,475
m3	16	4	13464	30064	5952	21080	5536	23160	70,117	91,019	29,883	8,981
m4	16	4	18704	40544	8952	29320	8344	31208	72,316	93,95	27,684	6,05
tms	16	4	6784	16704	5096	13544	3928	14832	81,082	91,316	18,918	8,684
newcpla1	16	4	2520	8176	4888	9072	3304	9984	110,959	90,865	-10,959	9,135
newxcpla1	23	6	3760	12112	5920	12272	3216	13600	101,321	90,235	-1,321	9,765
max128	24	6	20192	45184	9232	32016	4528	31536	70,857	101,522	29,143	-1,522



ЗАКЛЮЧЕНИЕ

При синтезе самопроверяемых дискретных устройств на основе ЛКС можно эффективно использовать композицию равновесных кодов «1 из 4» и «3 из 4». Тестер данного кода имеет простую и компактную структуру, для полной проверки которой достаточно четырех комбинаций. Это позволяет довольно просто синтезировать самопроверяемые СВК для дискретных устройств, например, по представленным в статье алгоритмам.

Анализируя результаты экспериментов, полученные с применением использованных программных модулей и продуктов, по показателям структурной избыточности, можно сделать вывод о преимуществах описанного в настоящей статье метода по сравнению с некоторыми известными методами синтеза самопроверяемых дискретных устройств. Более того, как будет показано во второй части статьи, учет свойств композиции равновесных кодов «1 из 4» и «3 из 4» позволяет сократить количество преобразуемых сигналов от объекта диагностирования и тем самым уменьшить количество выходов блоков $G(X)$. Такая процедура модификации структуры организации СВК, изображенной на рис. 1, позволяет добиться еще большего выигрыша в показателях сложности технической реализации самопроверяемых дискретных устройств по сравнению с известными методами. Здесь следует отметить, что для определения особенностей и характеристик разработанного метода также могут использоваться другие метрики, синтезаторы и наборы функциональных элементов.

Достоинства использования ЛКС и композиции равновесных кодов «1 из 4» и «3 из 4» состоят в том, что существует большое число способов синтеза СВК, что позволяет решать оптимизационные задачи и строить самопроверяемые дискретные устройства, например, с наименьшими аппаратными затратами. В качестве недостатка использования разработанного метода следует отметить необходимость учета числа сочетаний искажений на выходах объекта диагностирования при выборе среди них четверок. Здесь существуют два варианта в построении СВК: классический вариант с полным покрытием любых ошибок на выходах объекта диагностирования и вариант с покрытием неисправностей при их проявлении хотя бы на одном наборе значений аргументов. Также может быть использован первый вариант реализации СВК с покрытием любых ошибок на выходах объекта диагностирования с увеличенным числом контролируемых подмножеств выходов по сравнению с ми-

нимально необходимым для разработанного метода, но по показателям сложности технической реализации имеющий преимущество по сравнению с иными методами. О возможности такой реализации говорят некоторые результаты экспериментов и, в частности, анализ выигрышей в показателях сложности.

ЛИТЕРАТУРА

1. Согомонян Е.С., Слабаков Е.В. Самопроверяемые устройства и отказоустойчивые системы. – М.: Радио и связь, 1989, 208 с. [Sogomonyan, E.S., Slabakov, E.V. Samoproveryaemye ustroystva i otkazoustoichivyye sistemy. – Moscow: Radio i svyaz', 1989. – 208 s. (In Russian)].
2. Mitra, S., McCluskey, E.J. Which Concurrent Error Detection Scheme to Choose? // Proceedings of International Test Conference. – Atlantic City, NJ, USA, 2000. – P. 985–994. – DOI: 10.1109/TEST.2000.894311
3. Göessel, M., Ocheretny, V., Sogomonyan, E., Marienfeld, D. New Methods of Concurrent Checking: Edition 1. – Dordrecht: Springer Science+Business Media B.V., 2008. – 184 p.
4. Дрозд А.В. Нетрадиционный взгляд на рабочее диагностирование вычислительных устройств // Проблемы управления. – 2008. – № 2. – С. 48–56. [Drozd, A.V. An Untraditional View on Operational Diagnostics of Computing Devices // Control Sciences. – 2008. – No. 2. – P. 48–56. (In Russian)].
5. Сапожников В.В., Сапожников Вл.В. Самопроверяемые дискретные устройства. – СПб.: Энергоатомиздат, 1992. – 224 с. [Sapozhnikov, V.V., Sapozhnikov, Vl.V. Samoproveryaemye diskretnyye ustroystva. – St. Petersburg: Energoatomizdat, 1992. – 224 s. (In Russian)].
6. Goessel, M., Graf, S. Error Detection Circuits. – London: McGraw-Hill, 1994. – 261 p.
7. Сапожников В.В., Сапожников Вл.В., Ефанов Д.В. Коды Хэмминга в системах функционального контроля логических устройств. – СПб.: Наука, 2018. – 151 с. [Sapozhnikov, V.V., Sapozhnikov, Vl.V., Efanov, D.V. Kody Khemminga v sistemakh funktsional'nogo kontrolya logicheskikh ustroystv. – St. Petersburg: Nauka, 2018. – 151 s. (In Russian)].
8. Сапожников В.В., Сапожников Вл.В., Ефанов Д.В. Коды с суммированием для систем технического диагностирования. Том 1: Классические коды Бергера и их модификации. – М.: Наука, 2020. – 383 с. [Sapozhnikov, V.V., Sapozhnikov, Vl.V., Efanov, D.V. Kody s summirovaniem dlya sistem tekhnicheskogo diagnostirovaniya. Tom 1: Klassicheskie kody Bergera i ikh modifikatsii. – Moscow: Nauka, 2020. – 383 s. (In Russian)].
9. Сапожников В.В., Сапожников Вл.В., Ефанов Д.В. Коды с суммированием для систем технического диагностирования. Том 2: Взвешенные коды с суммированием. – М.: Наука, 2021, 455 с. [Sapozhnikov, V.V., Sapozhnikov, Vl.V., Efanov, D.V. Kody s summirovaniem dlya sistem tekhnicheskogo diagnostirovaniya. Tom 2: Vzveshennyye kody s summirovaniem. – Moscow: Nauka, 2021. – 455 s. (In Russian)].
10. Goessel, M., Sapozhnikov, V., Sapozhnikov, Vl., Dmitriev, A. A New Method for Concurrent Checking by Use of a 1-out-of-4 Code // Proceedings of 6th IEEE International On-Line Testing Workshop. – Palma de Mallorca, Spain, 2000. – P. 147–152. – DOI: 10.1109/OLT.2000.856627

11. *Dmitriev, A., Saposhnikov, V., Saposhnikov, V., Goessel, M.* New Self-Dual Circuits for Error Detection and Testing // *VLSI Design*. – 2000. – Vol. 11, iss. 1. – P. 1–21. – DOI: 10.1155/2000/84720
12. *Сапожников В.В., Сапожников Вл.В., Дмитриев А.В.* и др. Организация функционального контроля комбинационных схем методом логического дополнения // *Электронное моделирование*. – 2002. – Т. 24, № 6. – С. 52–66. [*Sapozhnikov, V.V., Sapozhnikov, Vl.V., Dmitriev, A.V., et al.* Organization of Functional Control of Combinational Circuits Using the Boolean Complement Method // *Electronic Modeling*. – 2002. – Vol. 24, no. 6. – P. 52–66. (In Russian)].
13. *Гессель М., Морозов А.В., Сапожников В.В., Сапожников Вл.В.* Логическое дополнение – новый метод контроля комбинационных схем // *Автоматика и телемеханика*. – 2003. – № 1. – С. 167–176. [*Gessel, M., Morozov, A.V., Sapozhnikov, V.V., Sapozhnikov, Vl.V.* Logic Complement, a New Method of Checking the Combinational Circuits // *Automation and Remote Control*. – 2003. – Vol. 64, no. 1. – P. 153–161].
14. *Saposhnikov, V.V., Morosov, A., Saposhnikov, Vl.V., Göessel, M.* A New Design Method for Self-Checking Unidirectional Combinational Circuits // *Journal of Electronic Testing: Theory and Applications*. – 1998. – Vol. 12, iss. 1-2. – P. 41–53. – DOI: 10.1023/A:1008257118423
15. *Матросова А.Ю., Митрофанов Е.В.* Синтез легко тестируемых последовательностных схем // *Вестник Томского государственного университета. Управление, вычислительная техника и информатика*. – 2013. – № 2 (23). – С. 140–147. [*Matrosova, A.Yu., Mitrofanov, E.V.* Delay Testable Sequential Circuit Design // *Tomsk State University Journal of Control and Computer Science*. – 2013. – No. 2 (23). – P. 140–147. (In Russian)].
16. *Ефанов Д.В., Елина Е.И.* Исследование способов синтеза схем встроенного контроля на основе логической коррекции сигналов с применением равномерных разделимых кодов // *Микроэлектроника*. – 2024. – Т. 53, № 5. – С. 413–426. – DOI: 10.31857/S0544126924050079. [*Efanov, D.V., Yelina, Y.I.* Investigation of Ways of Synthesizing Concurrent Error-Detection Circuits Based on Boolean Signal Correction Using Uniform Separable Codes // *Russian Microelectronics*. – 2024. – Vol. 53, no. 5. – P. 471–482. – DOI: 10.1134/S1063739724600456].
17. *Ефанов Д.В.* Синтез самопроверяемых вычислительных устройств на основе полной системы особых групп выходов объекта диагностирования // *Известия высших учебных заведений. Приборостроение*. – 2023. – Т. 66, № 5. – С. 355–372. – DOI: 10.17586/0021-3454-2023-66-5-355-372. [*Efanov, D.V.* Synthesis of Self-Checking Computing Devices Based on a Complete System of Special Groups of the Diagnostic Object Outputs // *Journal of Instrument Engineering*. – 2023. – Vol. 66, no. 5. – P. 355–372. – DOI: 10.17586/0021-3454-2023-66-5-355-372 (In Russian)].
18. *Гессель М., Согомонян Е.С.* Построение самотестируемых и самопроверяемых комбинационных устройств со слабозависимыми выходами // *Автоматика и телемеханика*. – 1992. – № 8. – С. 150–160. [*Goessel, M., Sogomonyan, E.S.* Design of Self-Testing and Self-Checking Combinational Circuits with Weakly Independent Outputs // *Automation and Remote Control*. – 1992. – Vol. 53, no. 8. – P. 1264–1272.]
19. *Sogomonyan, E.S., Gössel, M.* Design of Self-Testing and On-Line Fault Detection Combinational Circuits with Weakly Independent Outputs // *Journal of Electronic Testing: Theory and Applications*. – 1993. – Vol. 4, iss. 4. – P. 267–281. – DOI: 10.1007/BF00971975
20. *Freiman, C.V.* Optimal Error Detection Codes for Completely Asymmetric Binary Channels // *Information and Control*. – 1962. – Vol. 5, iss. 1. – P. 64–71. – DOI: 10.1016/S0019-9958(62)90223-1
21. *MacWilliams, F.J., Sloane, N.J.A.* The Theory of Error-Correcting Codes. – Amsterdam: North-Holland, 1977. – 785 p.
22. *Borden, J.M.* Optimal Asymmetric Error Detecting Codes // *Information and Control*. – 1982. – Vol. 53, iss. 1-2. – P. 66–73. – DOI: 10.1016/S0019-9958(82)91125-1
23. *Ефанов Д.В.* Композиции двух равновесных кодов с ортогональными по всем разрядам комбинациями для синтеза самопроверяемых дискретных устройств // *Проблемы управления*. – 2025. – № 3. – С. 49–62. [*Efanov, D.V.* Compositions of Two Constant-Weight Codes with Orthogonal Combinations over All Bits for Self-Checking Discrete Device Design // *Control Sciences*. – 2025. – No. 3. – P. 41–61].
24. *Efanov, D., Osadchy, G., Zueva, M.* Special Aspects of Errors Definition via Sum Codes within Embedded Control Schemas Being Realized by Means of Boolean Complement Method // *Proceedings of 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS'2021)*. – Cracow, Poland, 2021. – Vol. 1. – P. 424–431. – DOI: 10.1109/IDAACS53288.2021.9660837
25. *Аксёнова Г.П.* Необходимые и достаточные условия построения полностью проверяемых схем свертки по модулю два // *Автоматика и телемеханика*. – 1979. – № 9. – С. 126–135. [*Aksenova, G.P.* Necessary and Sufficient Conditions for Design of Completely Checkable Modulo 2 Convolution Circuits // *Automation and Remote Control*. – 1979. – Vol. 40, no. 9. – P. 1362–1369].
26. *Ефанов Д.В.* Синтез самопроверяемых комбинационных устройств на основе метода логической коррекции сигналов с применением кодов Боуза–Лина // *Информационные технологии*. – 2023. – Т. 29, № 10. – С. 503–511. – DOI: 10.17587/it.29.503-511. [*Efanov D.V.* Self-Checking Combinational Devices Synthesis Based on the Boolean Signal Correction Method Using Bose–Lin Codes // *Information Technologies*. – 2023. – Vol. 29, no. 10. – P. 503–511. – DOI: 10.17587/it.29.503-511. (In Russian)].
27. *Закревский А.Д., Поттосин Ю.В., Черемисинова Л.Д.* Логические основы проектирования дискретных устройств. – М.: Физматлит, 2007. – 592 с. [*Zakrevskii, A.D., Pottosin, Yu.V., Cheremisinova, L.D.* Logicheskie osnovy proektirovaniya diskretnykh ustroystv. – Moscow: Fizmatlit, 2007. – 592 s. (In Russian)].
28. *Пархоменко П.П., Согомонян Е.С.* Основы технической диагностики. Книга вторая. Оптимизация алгоритмов диагностирования, аппаратные средства. – М.: Энергоатомиздат, 1981. – 320 с. [*Parkhomenko, P.P., Sogomonyan, E.S.* Osnovy tekhnicheskoi diagnostiki. Tom 2: optimizatsiya algoritmov diagnostirovaniya, apparaturnye sredstva. – М.: Ehnergoatomizdat, 1981. – 320 s. (In Russian)].
29. *Lala, P.K.* Self-Checking and Fault-Tolerant Digital Design. – San Francisco: Morgan Kaufmann Publishers, 2001. – 216 p.
30. *Chioktour, V., Kakarountas, A.* Adaptive BIST for Concurrent On-Line Testing on Combinational Circuits // *Electronics*. – 2022. – Vol. 19, no. 11. – P. 1–20. – DOI: 10.3390/electronics11193193



31. *McElvain, K.* IWLS'93 Benchmark Set: Version 4.0: distributed as a part of IWLS'93 benchmark set. – 1993.
32. *Collection of Digital Design Benchmarks.* – URL: <https://ddd.fit.cvut.cz/www/prj/Benchmarks/> (дата обращения: 26.08.2025 г.) [Accessed August 26, 2025].
33. *Hahanov, V., Litvinova, E., Chumachenko, S., et al.* Vector Logic Modeling Self-Tested Circuits // Proceedings of 21st IEEE East-West Design & Test Symposium (EWDTS'2025). – Tbilisi, Georgia, 2025. – DOI: 10.1109/EWDTS67441.2025.11303706
34. *Ефанов Д.В.* Использование равновесного кода «1 из 4» при синтезе самопроверяемых схем встроенного контроля на основе логической коррекции сигналов // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. – 2025. – № 72. – С. 114–133. – DOI: 10.17223/19988605/72/12 [Efanov, D.V. Using the “1-Out-Of-4” Constant-Weight Code in the Synthesis of Self-Checking Concurrent Error-Detection Circuit Based on Boolean Signal Correction // Tomsk State University Journal of Control and Computer Science. – 2025. – No. 72. – P. 114–133. – DOI: 10.17223/19988605/72/12 (In Russian)].
35. *Закревский А.Д., Торопов Н.Р.* Минимизация булевых функций многих переменных в классе ДНФ – итеративный метод и программная реализация // Прикладная дискретная математика. – 2009. – № 1. – С. 5–14. [Zakrevskij, A.D., Toropov, N.R. Minimization of Boolean Functions of Many Variables - Iterative Method and Program Realization // Applied Discrete Mathematics. – 2009. – No. 1. – P. 5–14. (In Russian)].
36. *Sentovich, E.M., Singh, K.J., Moon, C., et al.* Sequential Circuit Design Using Synthesis and Optimization // Proceedings of IEEE International Conference on Computer Design: VLSI in Computers & Processors. – Cambridge, MA, USA, 1992. – P. 328–333. – DOI: 10.1109/ICCD.1992.276282
37. *SIS: A System for Sequential Circuit Synthesis / E.M. Sentovich, K.J. Singh, L. Lavagno, C. Moon, R. Murgai, A. Saldanha, H. Savoj, P. R. Stephan, R.K. Brayton, A. Sangiovanni-Vincentelli.* – Berkeley: Electronics Research Laboratory, Department of Electrical Engineering and Computer Science, University of California, 1992. – 45 p.
38. *Бибило П.Н.* Бинарные диаграммы решений в логическом проектировании. – М.: ЛЕНАНД, 2024. – 560 с. [Bibilo, P.N. Binarnye diagrammy reshenii v logicheskom proektirovanii. – Moscow: LENAND, 2024. – 560 s. (In Russian)].
39. *Ефанов Д.В., Елина Е.И.* Синтез самопроверяемых цифровых устройств на основе логической коррекции сигналов с применением взвешенных кодов Боуза – Лина // Проблемы управления. – 2024. – № 4. – С. 26–43. – DOI: <http://doi.org/10.25728/ru.2024.4.3> [Efanov D.V., Yelina Y.I. Design of Self-Checking Digital Devices with Boolean Signals Correction Using Weight-Based Bose–Lin Codes // Control Sciences. – 2024. – No. 4. – P. 22–36. – DOI: <http://doi.org/10.25728/ru.2024.4.3>].
40. *Коршунов А.Д.* Сложность вычислений булевых функций // Успехи математических наук. – 2012. – Т. 67, № 1 (403). – С. 97–168. – DOI: 10.4213/rm9459. [Korshunov, A.D. Computational Complexity of Boolean Functions // Russian Mathematical Surveys. – 2012. – Vol. 67, no. 1 (403). – P. 93–165. – DOI: <https://doi.org/10.1070/RM2012v067n01ABEH004777>]

Статья представлена к публикации членом редколлегии Л. Ю. Филimonюком.

Поступила в редакцию 14.09.2025,
после доработки 27.11.2025.
Принята к публикации 27.11.2025.

Ефанов Дмитрий Викторович – д-р техн. наук, Санкт-Петербургский политехнический университет Петра Великого, г. Санкт-Петербург; Институт проблем транспорта им. Н. С. Соломенко РАН, г. Санкт-Петербург, ✉ TrES-4b@yandex.ru
ORCID iD: <https://orcid.org/0000-0002-4563-6411>

Елина Есения Игоревна – аспирант, Санкт-Петербургский политехнический университет Петра Великого, г. Санкт-Петербург, ✉ eseniya-elina@mail.ru
ORCID iD: <https://orcid.org/0009-0004-4167-3591>

© 2026 г. Ефанов Д. В., Елина Е. И.



Эта статья доступна по [лицензии Creative Commons «Attribution» \(«Атрибуция»\) 4.0 Всемирная](https://creativecommons.org/licenses/by/4.0/).

**DESIGN OF SELF-CHECKING DISCRETE DEVICES
BASED ON BOOLEAN SIGNALS CORRECTION AND COMPOSITION
OF CONSTANT-WEIGHT CODES OF THE “1-OUT-OF-4” AND “3-OUT-OF-4” TYPES.
PART I: The Design Method with Conversion
of All Signals from the Object under Diagnosis**

D. V. Efanov* and Y. I. Yelina**

***Peter the Great Saint Petersburg Polytechnic University, St. Petersburg, Russia
*Solomenko Institute of Transport Problems, Russian Academy of Sciences, St. Petersburg, Russia

*✉ TrES-4b@yandex.ru, **✉ eseniya-elina@mail.ru

Abstract. It is proposed to use the composition of constant-weight codes of the “1-out-of-4” and “3-out-of-4” types in the design of self-checking discrete devices based on Boolean signals correction. For this composition of constant-weight codes, a simple and compact self-checking checker can be implemented, requiring only four test combinations for a complete check. The structure of a self-checking discrete device is described. The conversion of all four signals from an object under diagnosis is considered when designing a concurrent error-detection circuit for the object. The simplest algorithm for building a self-checking concurrent error-detection circuit based on the Boolean signals correction and the composition of 1-out-of-4 and 3-out-of-4 codes is developed. The results of experiments with combinational benchmarks are provided. They demonstrate the advantage of the proposed approach over the well-known duplication method and the one involving the 1-out-of-4 code together with Boolean signals correction, in terms of structural redundancy of the final self-checking discrete device. The method developed in this paper is promising for self-checking discrete device design with different element bases in various fields of application.

Keywords: self-checking discrete device, concurrent error-detection circuit, Boolean signals correction, composition of constant-weight codes, 1-out-of-4 and 3-out-of-4 codes, structural redundancy, controllable structure.