ВЕСТНИК ТОМСКОГО ГОСУДАРСТВЕННОГО УНИВЕРСИТЕТА
2021 Управление, вычислительная техника и информатика № 54

ПРОЕКТИРОВАНИЕ И ДИАГНОСТИКА ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

УДК 004.052.32+681.518.5
DOI: 10.17223/19988605/54/12

Д.В. Ефанов, В.В. Сапожников, Вл.В. Сапожников, Г.В. Осадчий

СИНТЕЗ СХЕМ ВСТРОЕННОГО КОНТРОЛЯ НА ОСНОВЕ МЕТОДА ЛОГИЧЕСКОГО ДОПОЛНЕНИЯ С ПРЕДВАРИТЕЛЬНЫМ СЖАТИЕМ СИГНАЛОВ РАБОЧИХ ФУНКЦИЙ

Работа выполнена при финансовой поддержке гранта № МД-2533.2021.4 Президента Российской Федерации.

Приводится описание разработанного авторами способа синтеза схем встроенного контроля по методу логического дополнения с предварительным сжатием сигналов от выходов объекта диагностирования и последующим контролем полученных сигналов с помощью равновесного кода. При этом использовано схемотехническое решение, позволяющее в блоке коррекции сигналов не преобразовывать один сигнал от одной из пар сжимаемых выходов, а также подключение одного из выходов блока контрольной логики непосредственно к одному из входов тестера. Такое техническое решение хоть и увеличивает на единицу длину контролируемого кодового слова по сравнению с известным решением без использования дополнительного выхода блока контрольной логики, но и позволяет увеличить число обнаруживаемых ошибок на выходах объекта диагностирования. Дается описание предлагаемой структуры организации схемы встроенного контроля по широко известным равновесным кодам «1 из 4» и «2 из 4». Отмечены неоспоримые преимущества применения именно кода «1 из 4» при организации схемы встроенного контроля по предложенному авторами способу. Несмотря на большее число выходов блока контрольной логики тестер кода «1 из 4» имеет более простую структуру, чем тестер кода «1 из 3», взамен которого организуется контроль схем по группам из трех выходов схемы сжатия. Способ легко обобщается на использование равновесных кодов «1 из *n*», что также показано в статье. Описан метод выбора групп сжимаемых выходов для обеспечения обнаружения ошибок на выходах элемента сжатия, подразумевающий выбор групп независимых выходов для сжатия сигналов от них. Применение разработанного авторами метода позволяет при допустимых аппаратурных затратах (менее аппаратурных затрат при дублировании) синтезировать полностью самопроверяемые схемы встроенного контроля логических устройств автоматики и вычислительной техники.

Ключевые слова: схема встроенного контроля; логическое дополнение; схема сжатия сигналов; равновесный код «1 из *n*»; равновесный код «1 из 3»; равновесный код «1 из 4»; равновесный код «2 из 4»; полностью самопроверяемая структура.

Методы синтеза полностью самопроверяемых схем широко используются при разработке надежных цифровых вычислительных систем [1–3] наравне с другими методами аппаратной и программной избыточности [4, 5]. Применение таких схем дает возможность построения устройства, снабженного схемой обнаружения неисправностей и индикации событий их возникновения, без подачи специализированных тестовых воздействий на входы устройства. Рабочие входные воздействия одновременно являются и тестовыми [6, 7].

При разработке самопроверяемого устройства исходный объект снабжается специализированной схемой контроля, которая позволяет косвенно по результатам вычислений функций на его выходах установить, исправен объект или в нем присутствует неисправность. Часто схемы контроля строятся с использованием равномерных двоичных избыточных кодов [8]. Применяются как разделимые коды, так и неразделимые коды [9]. В этом случае существует несколько основных способов организации схемы контроля.

Первый способ связан с дополнением вектора, формируемого на выходах блока основной логики, специальным контрольным вектором, вычисляемым блоком контрольной логики, до кодового слова заранее выбранного кода [10]. Для контроля принадлежности формируемого кодового слова на выходах обоих блоков в схеме контроля устанавливается тестер, снабжаемый одним или двумя выходами для информирования о техническом состоянии самопроверяемого устройства [11, 12]. Второй способ предполагает не дополнение вектора, формируемого на выходах блока основной логики, а его полное или частичное преобразование в кодовое слово заранее выбранного кода [13–15].

Первый способ достаточно полно освещен в мировой литературе [1, 2, 8, 9, 16]. Его основными достоинствами являются возможности синтеза полностью самопроверяемых структур при уменьшенной по сравнению с дублированием избыточностью. Однако существуют и некоторые недостатки. Например, не для любых устройств возможна реализация полностью самопроверяемых структур по определенным избыточным кодам, что связано с невозможностью формирования полного множества проверяющих комбинаций для тестера в структуре схемы контроля [8]. Требуется анализ возможностей применения конкретного кода при синтезе самопроверяемого цифрового устройства.

Как показано в [17, 18], второй способ, именуемый «методом логического дополнения», дает возможность устранить основной недостаток традиционного метода и позволяет для выбранного кода строить большое разнообразие схем встроенного контроля (СВК). Это позволяет не только достигать свойства полной самопроверяемости, но и влиять на избыточность конечного устройства, выбирая наилучший вариант организации СВК.

Данная статья посвящена описанию одного из способов организации СВК на основе метода логического дополнения. При этом подразумеваются первоначальное сжатие сигналов от рабочих выходов блока основной логики и последующее преобразование сформированного вектора в кодовое слово равновесного кода [19].

Такой подход к организации СВК по методу логического дополнения впервые был освещен в работе авторов данной статьи и подразумевает сжатие сигналов в базовой структуре для восьмивыходного логического устройства в четырехразрядный вектор и прямое преобразование данного вектора в кодовое слово равновесного кода «1 из 4» или «2 из 4» [20]. В отличие от указанной работы, в настоящей статье подразумевается организация базовой структуры для шестивыходных логических устройств с преобразованием трехбитного вектора в кодовое слово равновесного кода «1 из 4» или «2 из 4». Новый способ обладает универсальностью и позволяет существенно повысить обнаруживающую способность схемы контроля по сравнению с известными подходами.

1. Варианты организации схем встроенного контроля

Описываемый способ подразумевает организацию СВК по одной из базовых структур, изображенных на рис. 1, для исходного устройства $F(x)$. В СВК выделяется несколько функциональных блоков. Схема сжатия (СЖ) образована каскадом сумматоров по модулю $M = 2$ (элементов *XOR*) и предназначена для формирования трехбитного вектора $<\varphi_3\ \varphi_2\ \varphi_1>$. При этом сжимаются сигналы от пар выходов (f_1, f_4), (f_2, f_5), (f_3, f_6). Эти пары назовем парами сжимаемых выходов (ПСЖ-выходами), а сигналы на них – ПСЖ-сигналами. Блок логического дополнения (БЛД) предназначен для преобразования вектора $<\varphi_3\ \varphi_2\ \varphi_1>$ в кодовое слово равновесного кода. Для этого в нем используются блок контрольной логики $G(x)$ и блок коррекции сигналов (БКС). Блок контрольной логики формирует специальные контрольные функции, предназначенные для формирования сигналов коррекции разрядов вектора $<\varphi_3\ \varphi_2\ \varphi_1>$. Блок коррекции сигналов включает в себя элементы преобразования *XOR*, требующиеся для получения кодового слова выбранного кода. Для контроля соответствия кодового слова выбранному коду устанавливается тестер *TSC* (totally self-checking checker). Тестер имеет два выхода z^0 и z^1, на которых формируется контрольный сигнал: наличие парафазного сигнала <01> или <10> говорит об исправности устройства, нарушение же парафазности и формирование сигнала <00> или <11> свидетельствует о присутствии ошибки в вычислениях.

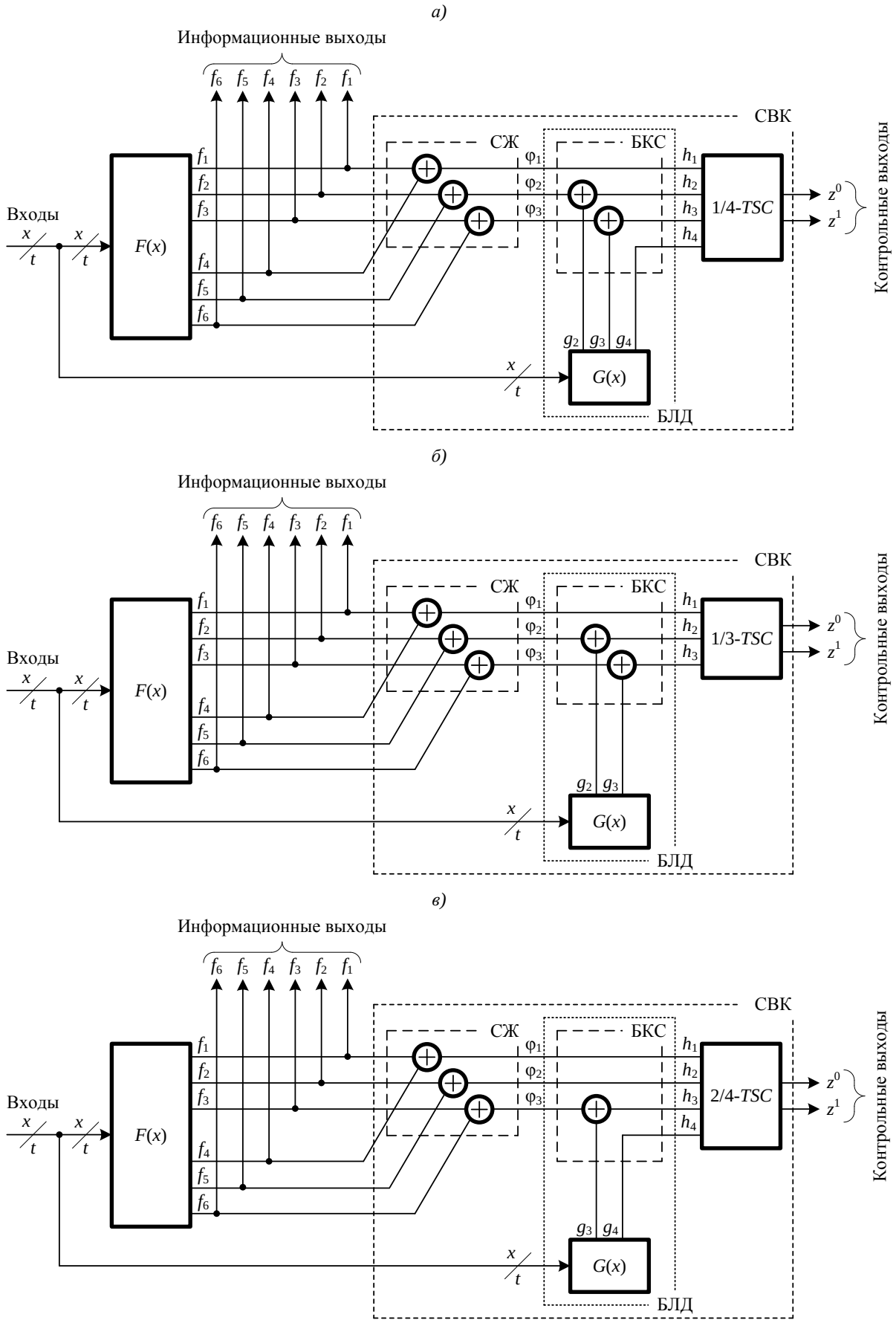


Рис. 1. Базовые структуры организации СВК

Fig. 1. Basic structures of the integrated check circuits organization

На рис. 1, *а* приведен способ преобразования трехбитного вектора <φ_3 φ_2 φ_1> в четырехбитный вектор <h_4 h_3 h_2 h_1>, принадлежащий равновесному коду «1 из 4» (1/4-коду). На рис. 1, *б* дан вариант преобразования трехбитного вектора <φ_3 φ_2 φ_1> в трехбитный вектор <h_3 h_2 h_1>, принадлежащий равновесному 1/3-коду. На рис. 1, *в* приведен способ преобразования трехбитного вектора <φ_3 φ_2 φ_1> в четырехбитный вектор <h_4 h_3 h_2 h_1>, принадлежащий равновесному 2/4-коду. Особенностью СВК, изображенных на рис. 1, *а, в*, является то, что в них подразумевается дополнение вектора <φ_3 φ_2 φ_1> до четырехбитного вектора <h_4 h_3 h_2 h_1> прямым подключением одного из выходов блока контрольной логики к входам тестера [21].

Установим особенности обнаружения ошибок в СВК, организованных по структурам, приведенным на рис. 1, а также преимущества и недостатки каждого из предлагаемых способов.

2. Особенности обнаружения ошибок

Различные варианты преобразований сигналов в СВК на основе равновесных кодов представлены на рис. 2–4. Данные примеры охватывают разнообразные случаи формирования сигналов на входах тестера. Если в СВК на основе 1/3-кода неважно, на каком из входов тестера сформирован сигнал логической единицы, то для схем на основе 1/4- и 2/4-кодов это существенно. Для схемы на основе 1/4-кода существует два случая – формирование единичного значения на входе тестера с преобразованием сигналов от блока $F(x)$ (рис. 3, *а*) и формирование единичного значения на входе тестера, подключенном напрямую к блоку $G(x)$ (рис. 3, *б*). Для схемы на основе 2/4-кода также существует два случая – формирование одного из единичных значений на входе тестера, подключенном напрямую к блоку $G(x)$ (рис. 4, *а*), и формирование на этом входе нулевого сигнала (рис. 4, *б*). Далее поясним, почему это важно.

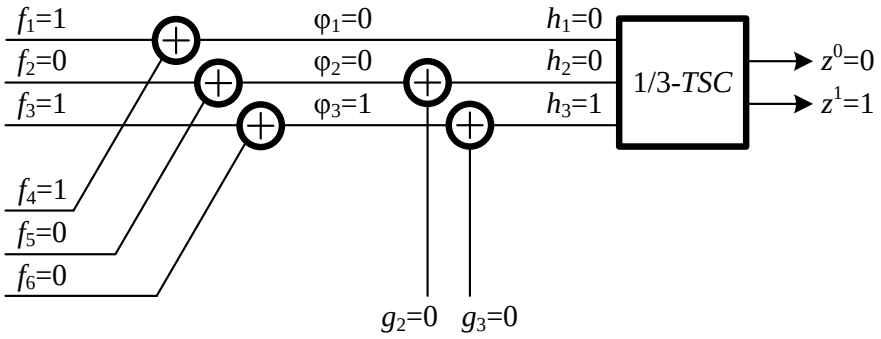


Рис. 2. Преобразования сигналов в СВК на основе 1/3-кода
Fig. 2. Conversion of signals into integrated check circuits based on 1/3-code

Рассмотрим варианты формирования сигналов, показанные на рис. 2, 3, *а*, 4, *а*). Определим, в каких случаях ошибки окажутся обнаруженными в СВК, а в каких – нет.

Заметим, что ошибка будет необнаруживаемой только в том случае, если переведет кодовое слово, принадлежащее выбранному равновесному коду, в кодовое слово, также принадлежащее ему. В рассматриваемых структурах ошибка не будет обнаружена в том случае, если она вызовет двукратное искажение в кодовом векторе <φ_1 φ_2 φ_3>, причем будет искажаться та функция, которая дает на входе тестера единичное значение, и та функция, которая дает на входе тестера нулевое значение. Для всех трех рассматриваемых структур характеристики ошибок с позиции их обнаружения в СВК будут одинаковыми (за исключением особого случая использования 1/4-кода).

В СВК на основе 1/3- и 1/4-кода ошибка не будет обнаружена, если исказятся те функции, которые приводят к трансляции ошибки на два входа тестера, на одном из которых в исправном состоянии формируется нулевое значение, а на втором – единичное. Для СВК на основе 2/4-кода ситуация оказывается аналогичной, поскольку для формирования кодового слова 2/4-кода при ошибках на вы-

ходах блока $F(x)$ и при сформированном единичном значении на входе тестера, подключенном напрямую к блоку $G(x)$, необходима трансляция искажения на два входа тестера, на которых при исправной работе формировались нулевое и единичное значения. Определим, какие именно это ошибки.

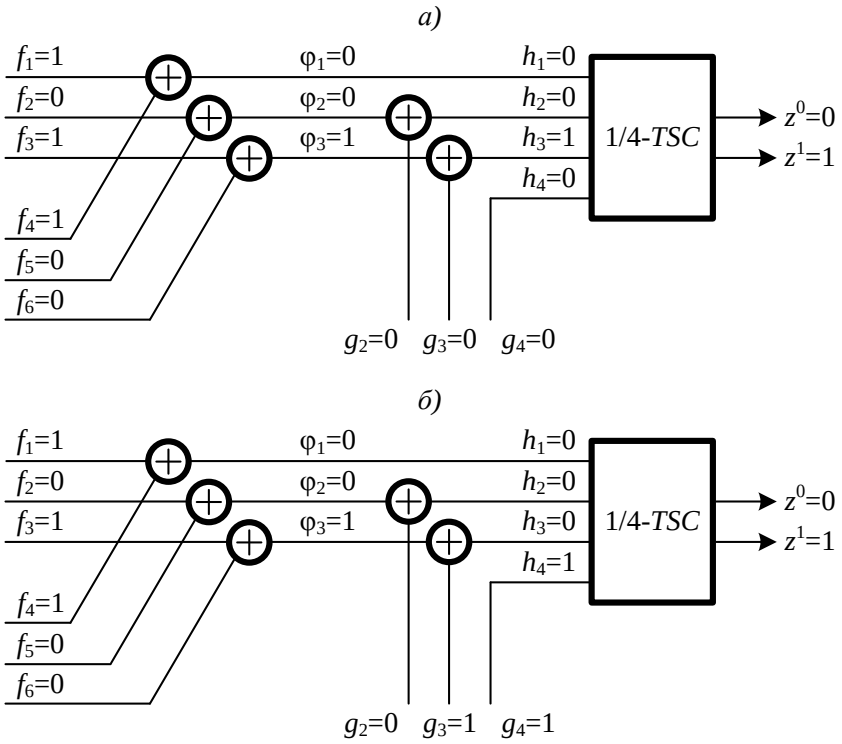


Рис. 3. Преобразования сигналов в СВК на основе 1/4-кода
Fig. 3. Conversion of signals into integrated check circuits based on 1/4-code

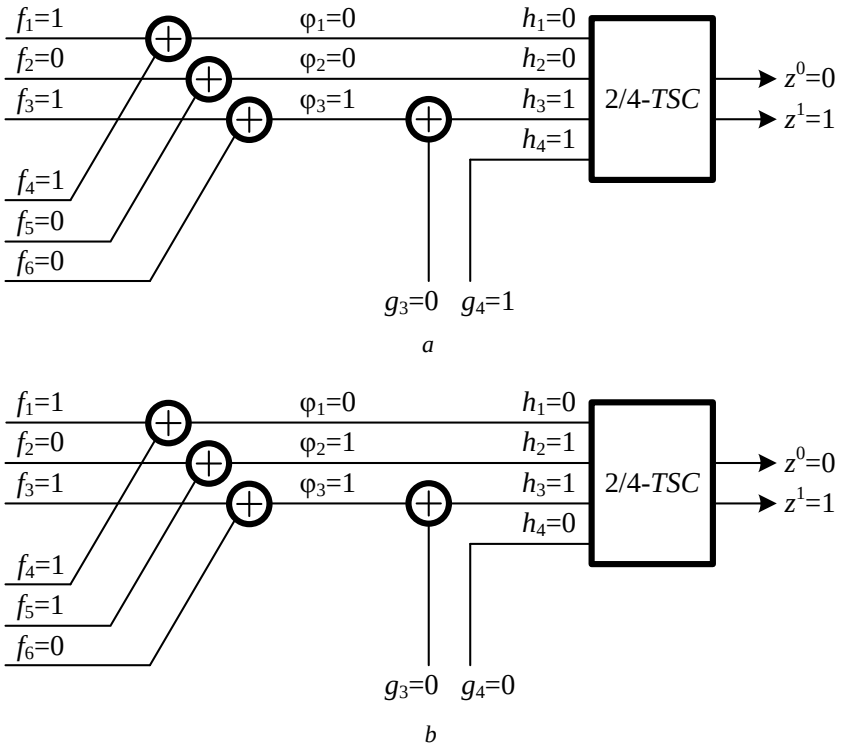


Рис. 4. Преобразования сигналов в СВК на основе 2/4-кода
Fig. 4. Conversion of signals into integrated check circuits based on 2/4-code

Выделим ПСЖ-выходы и ПСЖ-сигналы. Так как выходы сжимаются только парами, то ошибка в информационном векторе может оказаться необнаруживаемой только в том случае, если она имеет четную кратность. Ошибки с нечетными кратностями будут транслироваться на нечетное количество выходов СЖ: если ошибка возникнет на трех выходах первой группы $\{f_1, f_2, f_3\}$ или на трех выходах второй группы $\{f_4, f_5, f_6\}$, то она транслируется в виде трехкратной ошибки в векторе $<\varphi_1\ \varphi_2\ \varphi_3>$ и вызовет трехкратную ошибку в векторе $\{h_1, h_2, h_3\}$; если ошибка возникает на ПСЖ-выходах и на каком-либо из других выходов, то она будет транслирована в вектор $<\varphi_1\ \varphi_2\ \varphi_3>$ в виде либо однократной, либо трехкратной ошибки. Все эти ошибки обнаружатся тестером соответствующего кода. Определим, какие из ошибок четной кратностью не будут обнаружены в СВК.

Рассмотрим множество двукратных ошибок на выходах блока $F(x)$. Число таких ошибок равно $C_6^2 = 15$. Двукратная ошибка не будет обнаружена в СВК, если она возникнет на выходах одной из трех пар ПСЖ-выходов (число вариантов таких ошибок равно $C_3^1 = 3$). Также двукратная ошибка не будет обнаружена в СВК, если возникнет на том выходе блока $F(x)$, с которого сигнал транслируется на вход тестера с единичным значением при исправной работе, а также на одном из остальных выходов блока $F(x)$, не составляющим пару для первого выхода. Число выходов, с которых сигнал транслируется на вход тестера с единичным значением, при исправной работе равно двум (число вариантов искажений $C_2^1 = 2$), а число оставшихся выходов равно четырем. Это дает 2 × 4 = 8 необнаруживаемых двукратных ошибок. Итак, в классе необнаруживаемых оказывается 11 двукратных ошибок. Остальные 4 двукратные ошибки будут обнаружены.

Обратимся к четырехкратным ошибкам на выходах блока $F(x)$. Их число равно $C_6^4 = 15$. Четырехкратная ошибка не будет обнаружена, если она возникнет на выходах двух пар ПСЖ-выходов (число вариантов искажений равно $C_3^2 = 3$). Также четырехкратная ошибка не будет обнаружена, если она возникнет на выходах одной ПСЖ-пары, не дающей единицы на входе тестера при исправной работе (число вариантов выбора пар равно $C_2^1 = 2$), при этом исказится один из выходов, с которых сигнал транслируется на вход тестера с единичным значением (число вариантов равно $C_2^1 = 2$), и один из двух оставшихся выходов, с которых сигнал транслируется на вход тестера с нулевым значением. Число таких ошибок равно 2 × 2 × 2 = 8. Итого имеется 11 необнаруживаемых четырехкратных ошибок и 4 подобные ошибки будут обнаружены.

Шестикратная ошибка на выходах блока $F(x)$ одна ($C_6^6 = 1$). Она не будет обнаружена в СВК, так как полностью скомпенсируется в схеме сжатия.

Таким образом, в СВК не будет обнаружено 23 ошибки четной кратностью (74,19% ошибок четной кратностью) и будет обнаружено 8 ошибок четной кратностью (25,81% ошибок четной кратностью). В табл. 1 дана характеристика всех возможных ошибок четной кратностью.

Таблица 1

Характеристика ошибок на выходах блока основной логики

Двукратные ошибки		
f_1f_2 – О	f_2f_3 – Н	f_3f_5 – Н
f_1f_3 – Н	f_2f_4 – О	f_3f_6 – Н
f_1f_4 – Н	f_2f_5 – Н	f_4f_5 – О
f_1f_5 – О	f_2f_6 – Н	f_4f_6 – Н
f_1f_6 – Н	f_3f_4 – Н	f_5f_6 – Н
Четырехкратные ошибки		
$f_1f_2f_3f_4$ – Н	$f_1f_2f_5f_6$ – Н	$f_2f_3f_4f_5$ – Н
$f_1f_2f_3f_5$ – Н	$f_1f_3f_4f_5$ – Н	$f_2f_3f_4f_6$ – О
$f_1f_2f_3f_6$ – О	$f_1f_3f_4f_6$ – Н	$f_2f_3f_5f_6$ – Н
$f_1f_2f_4f_5$ – Н	$f_1f_3f_5f_6$ – О	$f_2f_4f_5f_6$ – Н
$f_1f_2f_4f_6$ – Н	$f_1f_4f_5f_6$ – Н	$f_3f_4f_5f_6$ – О
Шестикратные ошибки		
$f_1f_2f_3f_4f_5f_6$ – Н		

Аналогичные результаты получаются для структуры на основе 2/4-кода в случае, показанном на рис. 4, *б*. Однако ошибка будет необнаруживаемой в том случае, если она приведет к искажению на входах тестера, на которых при исправной работе формировались нулевое и единичное значения. Число обнаруживаемых ошибок четной кратностью будет равным числу обнаруживаемых ошибок для случая рис. 4, *а*.

Уникальным является случай рис. 3, *б*. В нем единственное единичное значение, формируемое на входах тестера 1/4-кода, появляется на том входе, сигнал на который приходит непосредственно с блока $G(x)$ без преобразования (формируется без работы блока $F(x)$). При такой реализации в СВК будет обнаружена любая ошибка на выходах блока $F(x)$, которая не окажется скомпенсированной в схеме сжатия. Скомпенсируются только $C_3^1 = 3$ двукратные ошибки (ошибки на выходах каждой из пар ПСЖ-выходов), $C_3^2 = 3$ четырехкратные ошибки (ошибки на выходах двух пар ПСЖ-выходов из трех) и единственная ($C_3^3 = 1$) шестикратная ошибка. Таким образом, не будет обнаружено 7 ошибок четной кратностью (22,58% ошибок четной кратностью) и будет обнаружено 24 ошибки четной кратностью (77,42% ошибок четной кратностью).

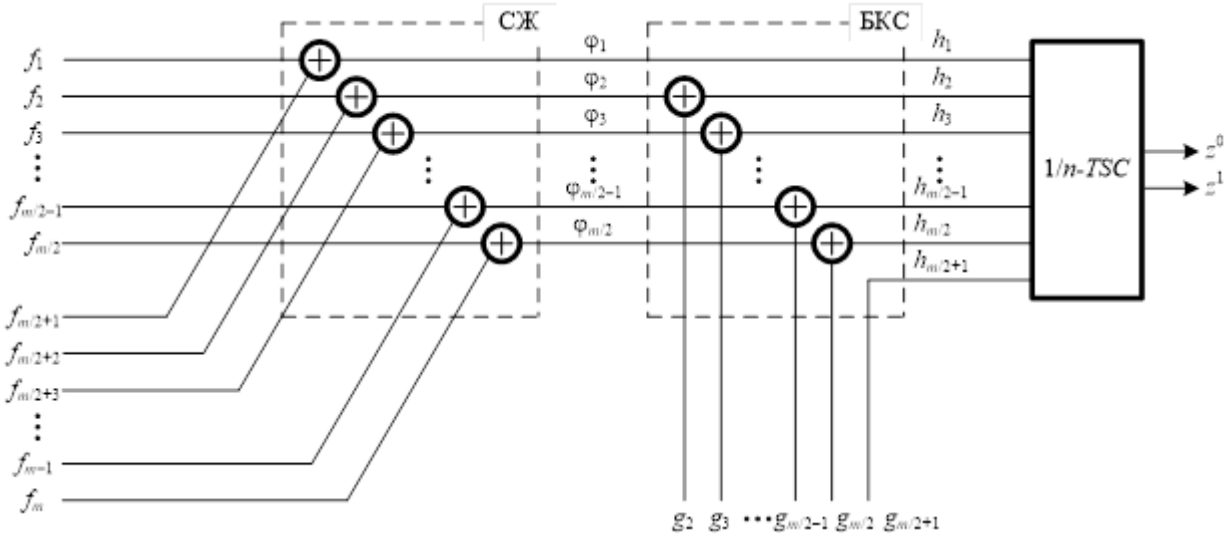
Этот уникальный случай может быть эффективно использован при построении СВК. Для обеспечения полной самопроверяемости тестера 1/4-кода потребуется формирование хотя бы по разу каждого из кодовых векторов множества {0001, 0010, 0100, 1000}. Учитывая, что использование кодового вектора, появляющегося в случае формирования единицы на входе тестера, подключенном напрямую к блоку $G(x)$, дает преимущество в обнаружении ошибок на выходах блока $F(x)$, можно при построении СВК наиболее часто использовать именно этот вектор, а все остальные кодовые векторы 1/4-кода формировать гораздо реже. Это позволит сократить потенциальное число необнаруживаемых в СВК ошибок.

3. Использование 1/*n*-кодов

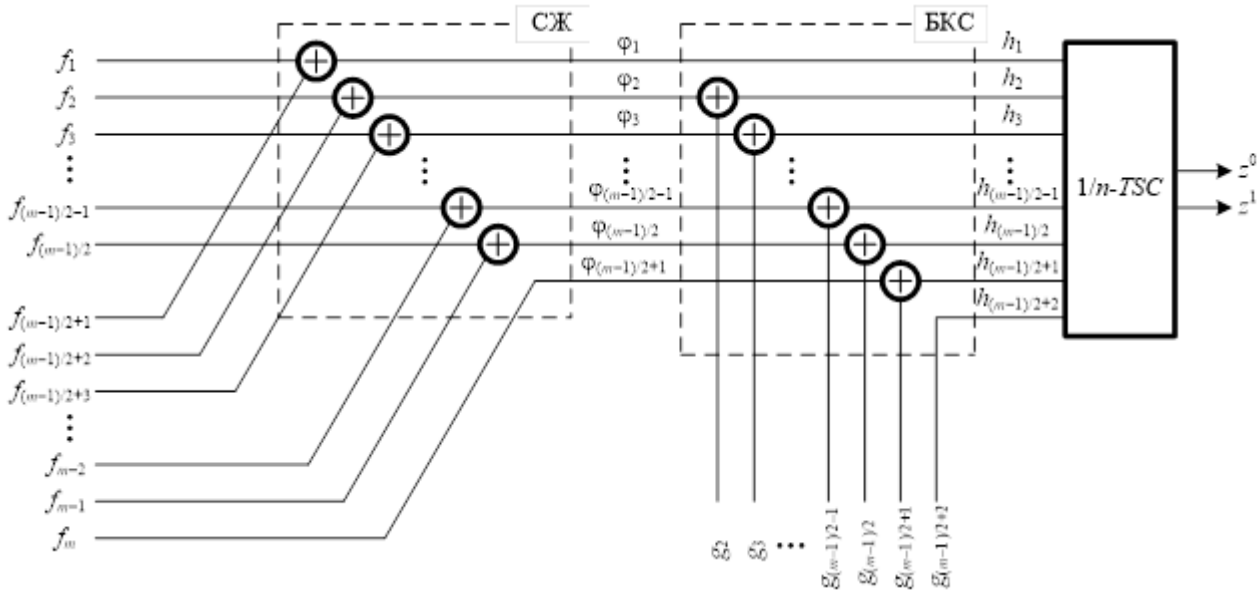
Следует отметить возможность применения при организации предложенным способом СВК, организованных по 1/*n* кодам. В этом случае схема сжатия и блок коррекции сигналов синтезируются по структурам, представленным на рис. 5. В данных структурах при четном числе выходов блока $F(x)$ для сжатия сигналов используется $\frac{m}{2}$ элементов сложения по модулю $M = 2$, и все выходы образуют $\frac{m}{2}$ пар ПСЖ-выходов. При нечетном числе выходов блока $F(x)$ для сжатия сигналов используется $\frac{m}{2} - 1$ элемент сложения по модулю $M = 2$, и все выходы образуют $\frac{m}{2} - 1$ пары ПСЖ-выходов и один свободный выход, который непосредственно подключается к входам блока коррекции сигналов. В блоке коррекции сигналов один из выходов схемы сжатия может не преобразовываться. Поэтому данный блок включает в себя $\frac{m}{2} - 1$ элементов сложения по модулю $M = 2$ в случае четного значения m и $\frac{m-1}{2}$ элементов сложения по модулю $M = 2$ в случае нечетного значения m.

С использованием приведенных на рис. 5 структур можно синтезировать СВК для логических схем с любым числом выходов. В табл. 2 приведены виды кодов для произвольных значений m, а также отмечен состав схем сжатия и блоков коррекции сигналов.

Следует, однако, отметить, что при использовании 1/*n*-кодов для контроля всех выходов существует проблема обеспечения полной самопроверяемости тестера 1/*n*-*TSC* [18]. Упрощение процесса реализации схемы контроля в таком случае достигается путем выделения нескольких контролируемых групп выходов, установкой более простых тестеров равновесных кодов с уменьшенной длиной кодовых слов и объединением контрольных выходов отдельных схем контроля на входах самопроверяемого компаратора [22].



a



b

Рис. 5. Организация СВК по 1/*n*-кодам: *a* – при четном числе выходов контролируемого устройства; *b* – при нечетном числе выходов контролируемого устройства

Fig. 5. Organization of integrated check circuits by 1/*n*-codes: *a* – with an even number of outputs of the controlled device; *b* – with an odd number of outputs of the controlled device

Таблица 2

Характеристика СВК для схем с произвольным числом выходов

m	Код	Число *XOR*sв СЖ	Число *XOR*sв БКС
4	1/3-код	2	1
5	1/4-код	2	2
6	1/4-код	3	2
7	1/5-код	3	3
8	1/5-код	4	3
9	1/6-код	4	4
10	1/6-код	5	4
…	…	…	…
Нечетное *m*	$1/\left(\frac{m+1}{2}+1\right)$-код	$\frac{m-1}{2}$	$\frac{m-1}{2}$
Четное *m*	$1/\left(\frac{m}{2}+1\right)$-код	$\frac{m}{2}$	$\frac{m}{2}-1$

4. Особенности выбора групп сжимаемых выходов

Исследования показывают, что можно схемотехническими способами добиться невозможности компенсации сигналов при сжатии выходов. Следующий способ, основанный на поиске групп независимых выходов, позволяет решить проблему маскировки ошибок на сжимаемых выходах.

Пример 1. *Случай с полным покрытием выходов устройства группами независимых выходов.*

Рассмотрим пример комбинационного устройства, приведенного на рис. 6.

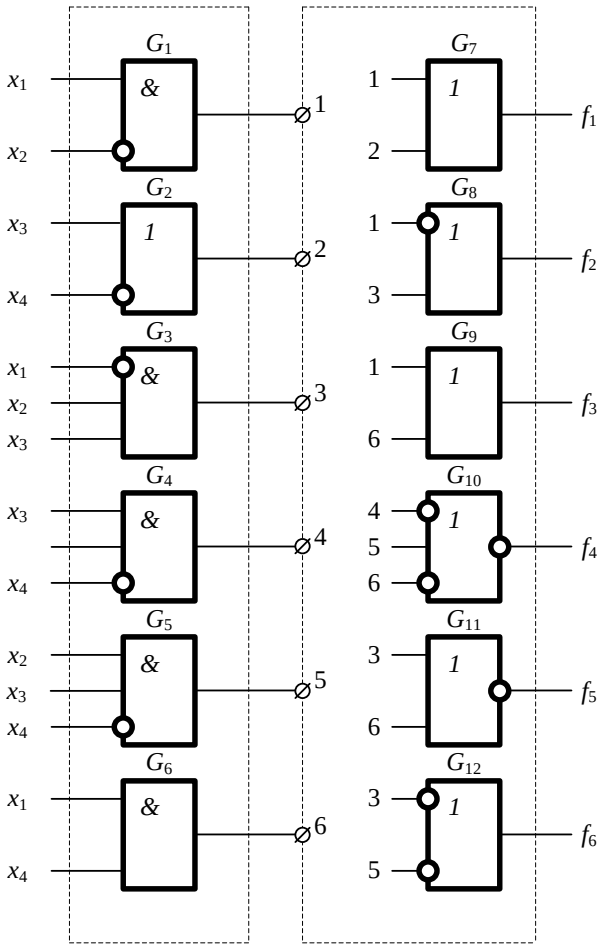


Рис. 6. Исходное комбинационное устройство к примеру 1
Fig. 6. The original combinational device for example 1

Установим пары зависимых и независимых выходов заданной комбинационной схемы, для чего составим матрицу, приведенную в табл. 3. На пересечении столбца и строки ставится знак покрытия (например, знак «×»). Если в одном столбце находится более одного знака покрытия, то выходы, в строках которых стоит данный знак, являются зависимыми, и от соответствующего столбцу логического элемента имеются пути к ним.

Таблица 3

Матрица зависимости элементов первого каскада схемы устройства и его выходов

f_i	G_j					
	G_1	G_2	G_3	G_4	G_5	G_6
f_1	×	×				
f_2	×		×			
f_3	×					×
f_4				×	×	×
f_5			×			×
f_6			×		×	

Из полученной матрицы необходимо найти все пары зависимых и независимых выходов (З-пары и Н-пары). Строго говоря, для решения поставленной задачи требуется найти только Н-пары. Однако удобно все пары классифицировать на две категории при работе с матрицей зависимости выходов и элементов. Для этого из рассмотрения исключаются все столбцы матрицы, имеющие только по одному знаку покрытия, а затем по столбцам анализируются те пары выходов, напротив которых стоит знак покрытия. Например, рассмотрим столбец G_1: из него следует, что З-парами являются следующие пары выходов: (f_1, f_2), (f_1, f_3), (f_2, f_3). Аналогично выделяются З-пары для столбцов G_3, G_5 и G_6. Перечень полученных пар, а также все остальные пары выходов, которые являются Н-парами, приведен в табл. 4.

Таблица 4

Классификация пар выходов комбинационного устройства

З-пары	Н-пары
(f_1, f_2)	(f_1, f_4) – A
(f_1, f_3)	(f_1, f_5) – B
(f_2, f_3)	(f_1, f_6) – C
(f_3, f_4)	(f_2, f_4) – D
(f_3, f_5)	(f_2, f_5) – E
(f_4, f_5)	(f_2, f_6) – F
(f_4, f_6)	(f_3, f_6) – G
(f_5, f_6)	

Здесь следует отметить, что общее количество пар равно C_m^2, где m – число выходов устройства. Среди Н-пар могут присутствовать такие пары выходов, которые образуют «тройки», «четверки» независимых выходов и т.д. Для установления такой особенности сравниваются выходы пар – если в «тройке» выходов все выходы попарно независимы, то и «тройка» выходов будет независима. Далее сравнения повторяются для «четверки» выходов и т.д. Методика аналогична описанной в [23]. Необходимо также обратить внимание читателя и на то, что при выделении трех и более выходов в группы сжимаемых выходов возникают условия для формирования необнаруживаемых ошибок с нечетной кратностью на выходах схемы сжатия. Тем не менее всегда могут быть выделены пары сжимаемых выходов вместо большего их количества.

Далее составляется таблица покрытия выходов заданного устройства Н-парами (табл. 5).

Таблица 5

Таблица покрытий

f_i	Буквы покрытия						
	A	B	C	D	E	F	G
f_1	×	×	×				
f_2				×	×	×	
f_3							×
f_4	×			×			
f_5		×			×		
f_6			×			×	×

Находится покрытие всех выходов минимальным количеством букв, для чего составляется и минимизируется выражение типа «конъюнкция дизъюнкций»: каждая дизъюнкция соответствует строке – записывается дизъюнкция тех букв, для которых в рассматриваемой строке указан знак покрытия. Для табл. 3 имеем:

$$W = (A \vee B \vee C)(D \vee E \vee F)G(A \vee D)(B \vee E)(C \vee F \vee G) =$$
$$= G(A \vee BD \vee CD)(E \vee BD \vee BF) = GAE \vee GABF \vee GBD \vee GCDE.$$

В выражении W, записанном в виде дизъюнкции конъюнкций, каждая конъюнкция соответствует возможному покрытию выходов Н-парами. Выберем вариант покрытия GAE, которому соответствуют пары: G – (f_3, f_6), A – (f_1, f_4) и E – (f_2, f_5).

Следует отметить, что при минимизации может оказаться так, что сразу несколько выходов входит в две и более пары. В этом случае в целях минимизации числа элементов преобразования при синтезе схемы контроля из всех пар, кроме одной, исключаются повторяющиеся выходы. Еще один случай – когда все выходы не удается покрыть Н-парами. Тогда оставшиеся выходы следует обозначить новыми буквами (каждому выходу присвоить свою букву) и добавить в логическое выражение *W*. Соответствующий пример приведем несколько позже.

Обратимся к особенностям структуры схемы встроенного контроля.

Получив покрытие выходов Н-группами, мы можем синтезировать схему сжатия: выходы каждой Н-группы становятся сжимаемыми выходами (СЖ-выходами). В рассматриваемом варианте имеется три пары СЖ-выходов. Далее выбирается способ контроля, например по равновесному коду «1 из 4» (1/4-коду) (рис. 7).

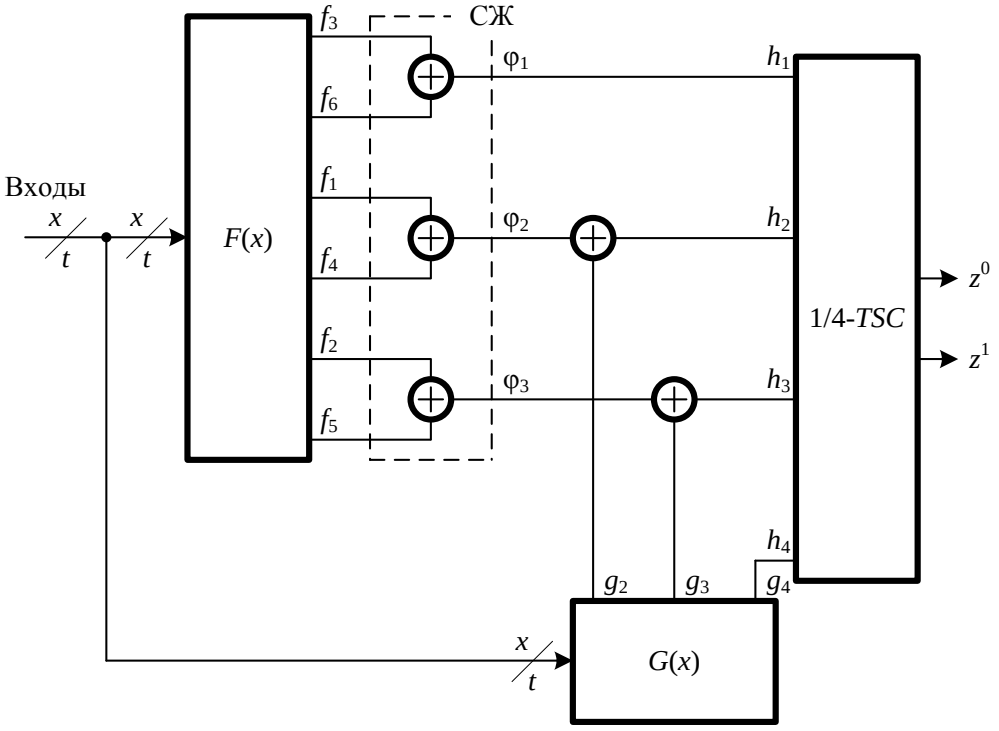


Рис. 7. Схема встроенного контроля по 1/4-коду для примера 1
Fig. 7. Integrated check circuits by 1/4-code for example 1

Так как предложенный способ организации схемы сжатия позволяет сжимать сигналы с независимых выходов, на элементах преобразования никогда не будет происходить компенсации сигналов. Однако возможно проявление искажений сразу же на двух и более выходах самой схемы сжатия, а значит, существует возможность возникновения многократной ошибки на входах тестера. К примеру, ошибка на выходе элемента G_6 приведет к тому, что возникнут ошибки на входах элементов G_9, G_{10} и G_{11}, входы которых являются выходами f_3, f_4 и f_5 устройства. Это гипотетически может приводить к формированию двух- и трехкратной ошибки на входах тестера. Аналогично ошибка на выходе элемента G_5 приведет возникновению ошибки на входах элементов G_{10} и G_{12}, входы которых являются выходами f_4 и f_6 устройства. Это создает условия для возникновения двукратной ошибки на входах тестера. Могут быть приведены и другие примеры.

Пример 2. *Случай с неполным покрытием выходов устройства группами независимых выходов.*

Рассмотрим пример комбинационного устройства, приведенного на рис. 8. Проведение процедуры поиска Н-пар по описанной выше методике позволило выделить две Н-пары – (f_1, f_6) и (f_2, f_5). Выходы же f_3 и f_4 являются зависимыми. Для исключения возможностей компенсации сигналов на элементах «сжатия» эти выходы следует разделять.

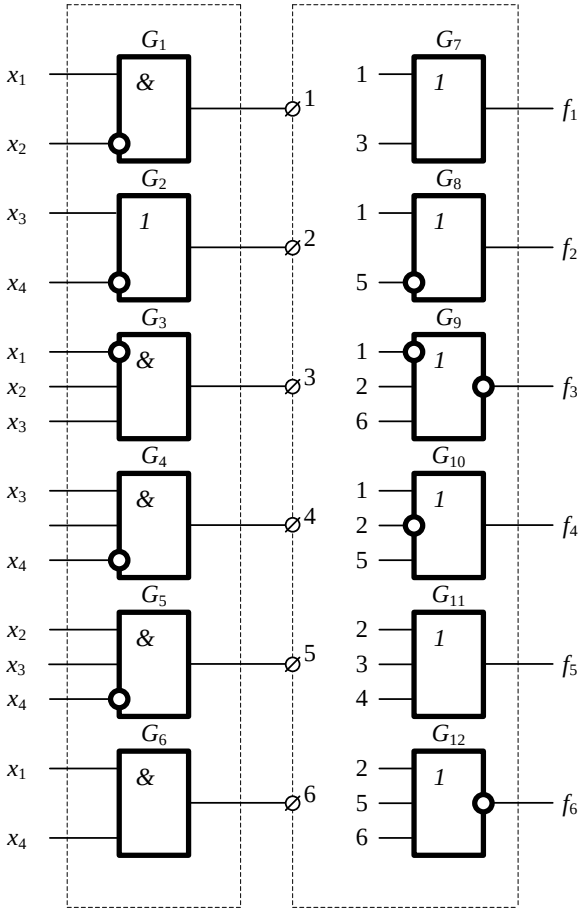


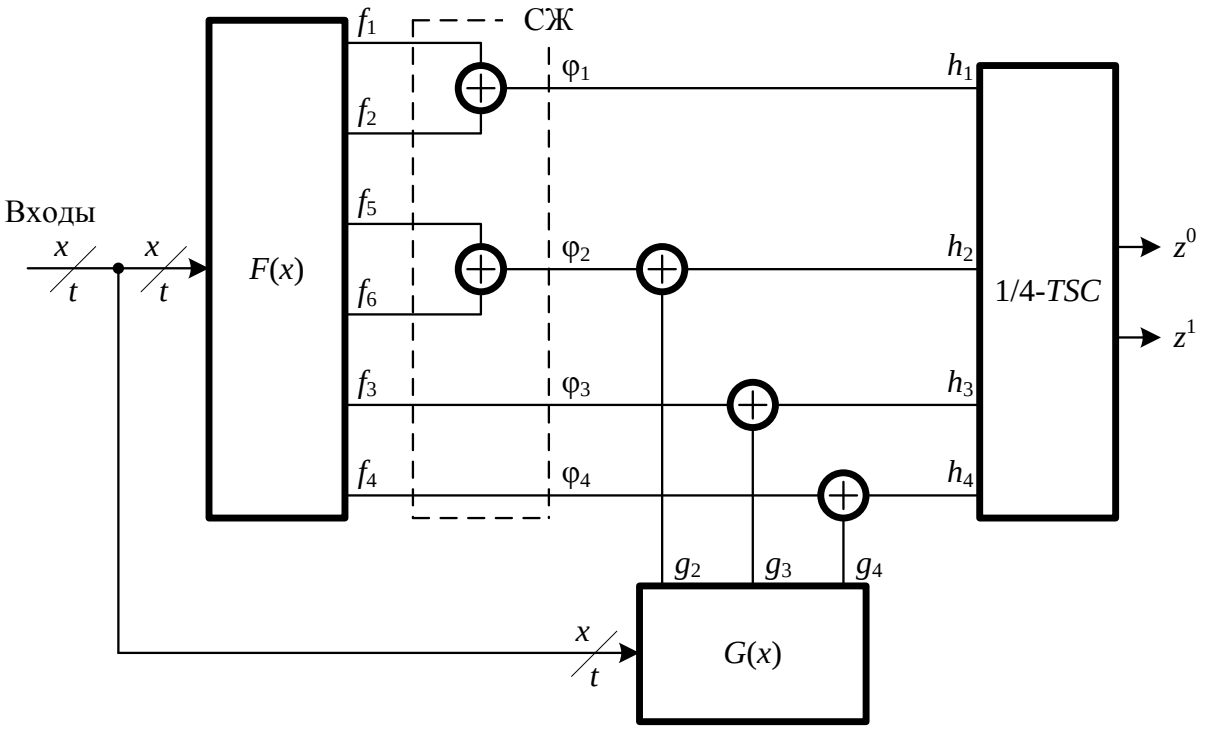
Рис. 8. Исходное комбинационное устройство к примеру 2
Fig. 8. The original combinational device for example 2

Варианты организации схем встроенного контроля для данного случая приведены на рис. 9. Следует отметить, что вариант, приведенный на рис. 9, *а* имеет более простой тестер, чем вариант, приведенный на рис. 9, *б*. Кроме того, блок контрольной логики $G(x)$ для первой структуры имеет три выхода, тогда как для второй – четыре выхода. Однако вариант на рис. 9, *б* позволяет воспользоваться предлагаемой в статье особенностью структуры с использованием сжимаемых выходов и увеличить количество обнаруживаемых на выходах контролируемого объекта ошибок. Эффективность такого подхода, однако, решается отдельно в каждом конкретном случае.

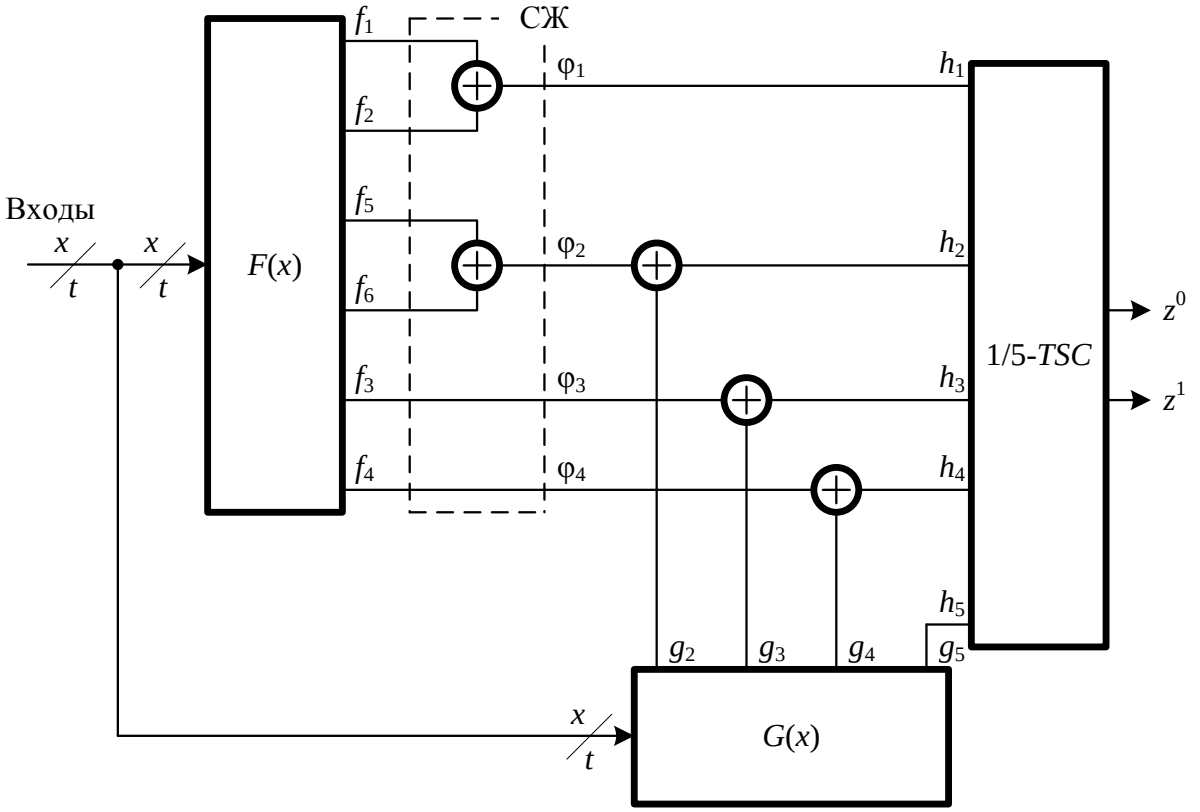
5. Эксперименты с контрольными схемами

Предлагаемый в статье подход во многих случаях дает возможность построения гораздо более простой СВК, чем по известному методу дублирования, что связано с использованием более простого блока для вычисления контрольных функций.

В ходе исследований были проведены эксперименты с рядом комбинационных схем из тестовых наборов LGSunth’89 и MCNCBenchmarks [24]. Для проведения экспериментов число вариаций СВК было ограничено: выбрана базовая структура для сжатия сигналов с 8 выходов попарно в четыре сигнала с последующим преобразованием с помощью трех элементов *XOR* в кодовое слово 1/4-кода (рис. 10). Выходы комбинационных схем $F(x)$ разбивались на q групп по 8 выходов в каждой, остальные выходы дублировались. Результаты экспериментов с контрольными комбинационными схемами с числом выходов $m \geq 8$ представлены в табл. 6. Для каждой схемы $F(x)$ указано число входов (n_{in}) и выходов (n_{out}), а также сложность $L_{F(x)}$ технической реализации в условных единицах системы SIS [25]. Сложность блока контрольной логики приводится в столбце $L_{G(x)}$. В столбце L_{CED} приведены показатели сложности технической реализации устройства с СВК.



a



b

Рис. 9. Варианты схем встроенного контроля для устройства из примера 2
Fig. 9. Variants of integrated check circuits for the device from example 2

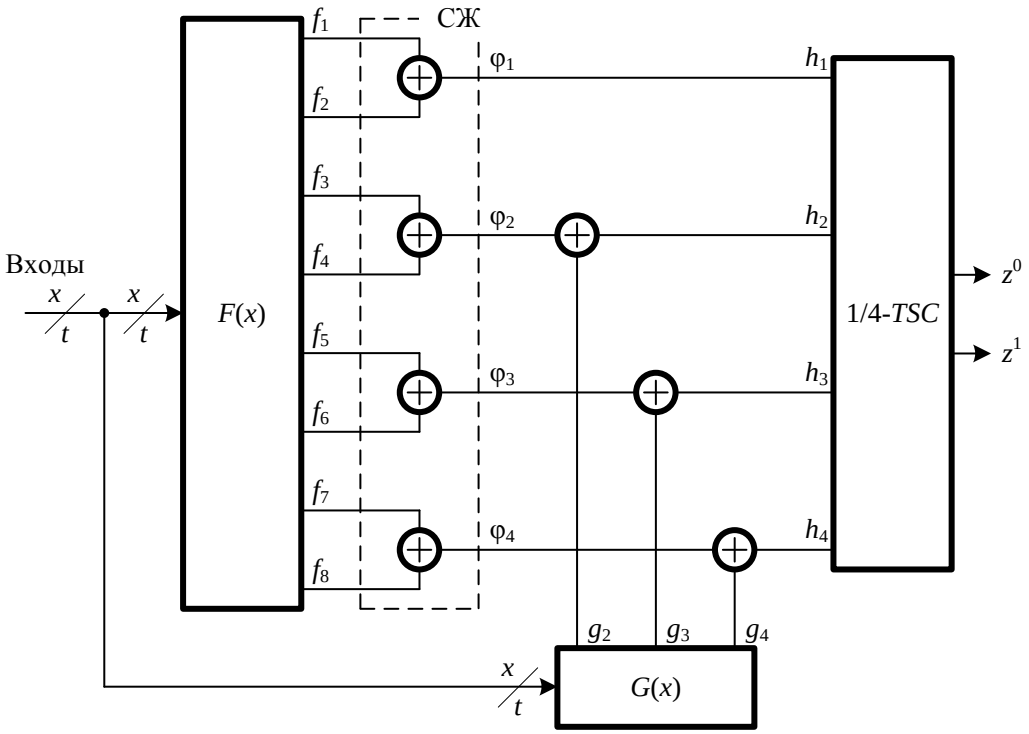


Рис. 10. Базовая структура организации схемы встроенного контроля по 1/4-коду спредварительнымсжатиемсигналов
Fig. 10. The basic structure of the organization of the integrated check circuits by 1/4-code with pre-compression of signals

Таблица 6

Результаты экспериментов с контролем групп выходов на основе 1/4-кода

№	Схема	n_{in}/n_{out}	$L_{F(x)}$	q	$L_{G(x)}$	L_{CED}	μ, %
1	cm138a	6/8	41	1	27,9	105	60,3
2	pcle	19/9	102	1	111,9	262	85,1
3	5xp1	7/10	169	1	154,5	384	84,2
4	cm42a	4/10	41	1	35,1	136	68
5	cu	14/11	90	1	74,3	236	76,1
6	decod	5/16	62	2	28,5	175	54
7	ldd	9/19	114	2	90,6	327	72,4
8	sct	19/15	109	1	96,2	332	82,1
9	unreg	36/16	152	2	154	446	89,3
10	cht	47/36	262	4	290	952	97,2
11	ttt2	24/21	300	2	265	714	83,6
12	tcon	17/16	49	2	60,3	249	84,4
13	term1	34/10	230	1	188	480	83,1
14	count	35/16	210	2	197	491	79,5
15	c8	28/18	214	2	199	536	82,7
16	lal	26/19	139	2	140	457	89,1
17	x1	51/35	456	4	434,5	1109	81,1
18	vda	17/39	810	4	783,3	1812	85,2
19	pm1	16/13	69	1	42	212	71
Среднее значение							79,4

Эффективность применения рассматриваемого подхода оценивалась по сравнению с использованием дублирования для этих целей. В столбцах φ и μ приведены значения относительных показателей для оценки сложности технической реализации СВК:

$$\varphi = \frac{L_{G(x)}}{L_{F(x)}} \cdot 100\%, \quad \mu = \frac{L_{CED}}{L_D} \cdot 100\%,$$

где L_D – показатель сложности технической реализации системы дублирования для заданного устройства.

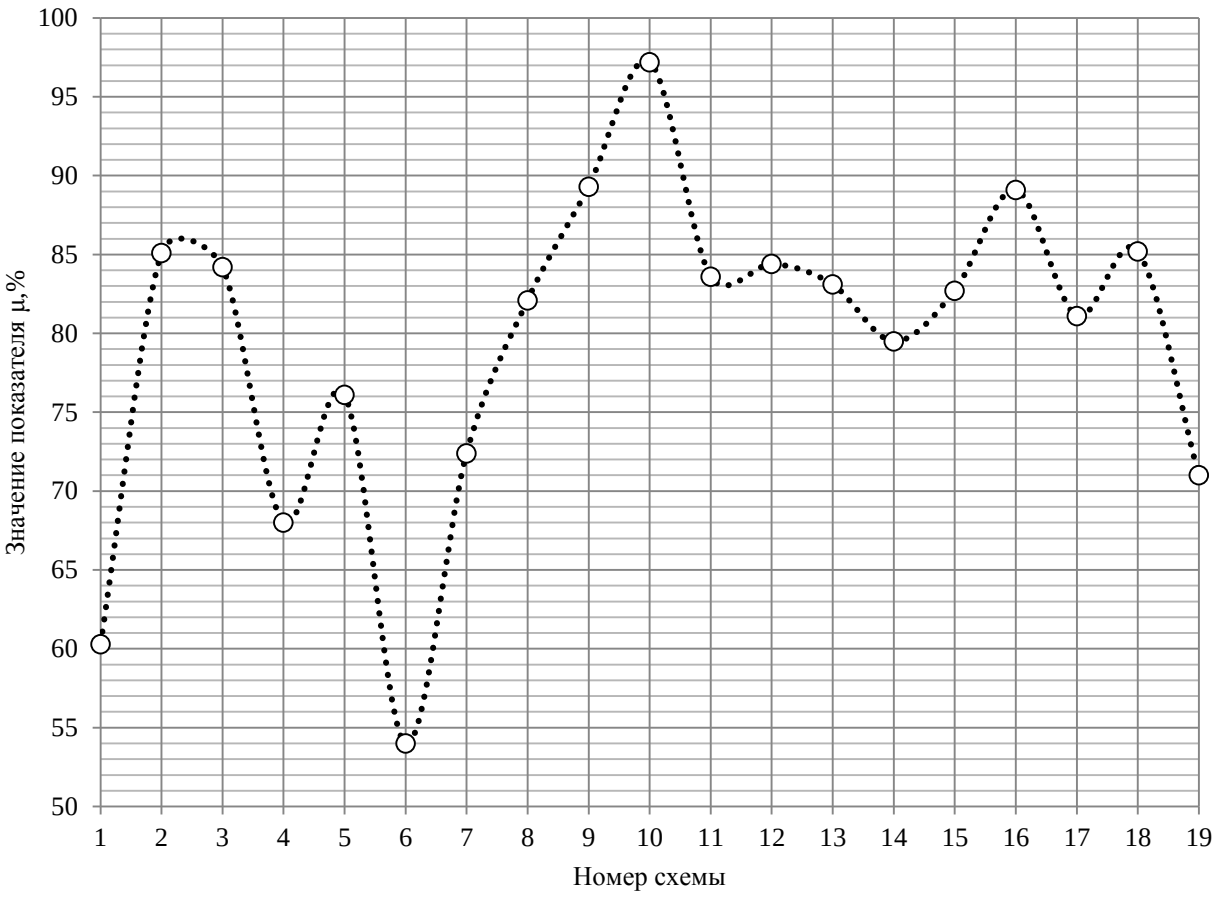


Рис. 11. Сравнение показателей μ для СВК, синтезированных по рассматриваемому методу
Fig. 11. Comparison of indicators μ for integrated check circuits, synthesized by the considered method

Для всех контрольных комбинационных схем получен показатель μ менее 100%, что свидетельствует о более простой реализации, чем по методу дублирования. Среднее значение μ = 79,4%. Для наглядности на рис. 11 показано значение показателя μ для всех рассмотренных схем.

Использование СВК, в которой не преобразуется один из сигналов, может дать еще большее уменьшение показателя μ. Следует, однако, отметить, что не всегда использование предложенного подхода может дать эффект по сравнению с применением дублирования. Функции исходного устройства $F(x)$ могут быть простыми и, соответственно, более простым может оказаться применение стандартного подхода с дублированием. При синтезе устройств с обнаружением неисправностей необходимо предварительно проводить оценку эффективности реализации по предложенному методу в сравнении с дублированием и выбирать наилучший способ.

Заключение

Представленный в настоящей работе подход к организации СВК на основе метода логического дополнения позволяет за счет использования схемы сжатия выбирать равновесный код с меньшей длиной кодовых слов, чем при известном подходе без применения схемы сжатия. Это, в свою очередь, дает возможность уменьшения сложности технической реализации СВК за счет использования более простых тестеров равновесных кодов. Кроме того, следует отметить неоспоримые преимущества предлагаемого подхода к организации СВК, заключающиеся в том, что удается строить СВК, обнаруживающие большее количество ошибок, чем СВК по известному коду паритета. Это следует из того факта, что в предложенных структурах обнаруживается некоторая доля ошибок четной кратностью, тогда как кодом паритета не обнаруживается любая ошибка четной кратностью. При этом

в схеме сжатия и блоке коррекции сигналов используется точно такое же число элементов сложения по модулю $M = 2$. Усложнение предлагаемой схемы по сравнению со схемой контроля по паритету связано только со сложностью тестера равновесного кода.

Использование при организации СВК такого подхода, при котором один из выходов блока контрольной логики $G(x)$ подключается на вход тестера без преобразования, позволяет существенно повысить число обнаруживаемых ошибок на выходах блока основной логики $F(x)$. Особое место среди всех равновесных кодов, как показано в статье, занимает 1/4-код. Его использование при контроле трехбитных векторов, поступающих со схемы сжатия, вместо 1/3-кода позволяет значительно повысить обнаруживающую способность СВК. При этом, если единичное значение формируется на непреобразуемом выходе блока $G(x)$, удается обнаруживать любые ошибки на выходах блока $F(x)$, кроме ошибок на парах ПСЖ-выходов. Таким образом, в схеме контроля по 1/4-коду можно значительно повысить обнаруживающую способность за счет формирования единичного значения преимущественно на выходе блока контрольной логики, подключаемом непосредственно к одному из входов тестера. Для построения полностью самопроверяемого комбинационного устройства при этом необходимо обеспечивать формирование полных множеств тестовых комбинаций для элементов сложения по модулю $M = 2$ в схеме сжатия и блоке коррекции сигналов, а также на входах тестера; кроме того, следует учитывать особенности реализации блока контрольной логики при использовании метода логического дополнения [26].

Представленный в статье подход, по нашему мнению, является перспективным при разработке самопроверяемых цифровых устройств.

ЛИТЕРАТУРА

1. Согомонян Е.С., Слабаков Е.В. Самопроверяемые устройства и отказоустойчивые системы. М. : Радио и связь, 1989. 208 с.
2. Goessel M., Graf S. Error Detection Circuits. London : McGraw-Hill, 1994. 261 p.
3. Mitra S., McCluskey E.J. Which Concurrent Error Detection Scheme to Choose? // Proceedings of International Test Conference, 2000, USA, Atlantic City, NJ, 03–05 October 2000. P. 985–994.
4. Drozd O., Romankevich V., Kuznietsov M., Drozd M., Mertynyuk O. Using Natural Version Redundancy of FPGA Projects in Area of Critical Applications // Proceedings of IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT), 14–18 May 2020. Kyiv, Ukraine. DOI: 10.1109/DESSERT50317.2020.9125050.
5. Babeshko E., Illiashenko O., Kharchenko V., Ruchkiv E. Safety and Reliability Assessment of NPP Instrumentation and Control Systems Considering Different Communication Architectures // Nuclear and Radiation Safety. 2020. Is. 2 (86). P. 38–43.
6. Mertynyuk O., Drozd O., Stepova H., Martynyuk D. Multi-Level Method of Behavioral Online Testing of Distributed Information Systems // Proceedings of 10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), 18–21 September 2019, Metz, France. DOI: 10.1109/IDAACS.2019.8924427.
7. Дрозд А.В., Харченко В.С., Антощук С.Г., Дрозд Ю.В., Дрозд М.А., Сулима Ю.Ю. Рабочее диагностирование безопасных информационно-управляющих систем / под ред. А.В. Дрозда и В.С. Харченко. Харьков : Нац. аэрокосмический ун-т им. Н.Е. Жуковского «ХАИ», 2012. 614 с.
8. Сапожников В.В., Сапожников Вл.В., Ефанов Д.В. Коды с суммированием для систем технического диагностирования. М. : Наука, 2020. Т. 1: Классические коды Бергера и их модификации. 383 с.
9. Piestrak S.J. Design of Self-Testing Checkers for Unidirectional Error Detecting Codes. Wrocław : Oficyna Wydawnicza Politechniki Wrocławskiej, 1995. 111 p.
10. Микони С.В. Общие диагностические базы знаний вычислительных систем. СПб. : СПИИРАН, 1992. 234 с.
11. Nikolos D., Kavousianos X. Modular TSC Checkers for Bose-Lin and Bose Codes // Proceedings of the 17th IEEE VLSI Test Symposium. April 25–29, 1999. Dana Point. USA. P. 354–360.
12. Matrosova A., Ostrovsky V., Levin I., Nikitin K. Designing FPGA based Self-Testing Checkers for m-out-of-n Codes // Proceedings of the 9th IEEE International On-Line Testing Symposium (IOLTS'03). 7–9 July 2003. Kos Island, Greece. P. 49–53.
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.
14. Göessel M., Ocheretny V., Sogomonyan E., Marienfeld D. New methods of concurrent checking: Ed. 1. Dordrecht : Springer Science + Business Media B.V., 2008, 184 p.
15. Das D.K., Roy S.S., Dmitiriev A., Morozov A., Gössel M. Constraint Don't Cares for Optimizing Designs for Concurrent Checking by 1-out-of-3 Codes // Proceedings of the 10th International Workshops on Boolean Problems, Freiberg. Germany, September, 2012. P. 33–40.
16. Lala P.K. Self-Checking and Fault-Tolerant Digital Design. San Francisco : Morgan Kaufmann Publishers, 2001. 216 p.

17. Ефанов Д.В., Сапожников В.В., Сапожников Вл.В., Пивоваров Д.В. Синтез самопроверяемых схем встроенного контроля на основе метода логического дополнения до равновесного кода «2 из 4» // Информатика. 2018. Т. 15, № 4. С. 71–85.
18. Efanov D.V., Sapozhnikov V.V., SapozhnikovVl.V., Pivovarov D.V. Synthesis of built-in self-test control circuits based on the method of boolean complement to constant-weight 1-out-of-n codes // Automatic Control and Computer Sciences. 2019. V. 53, is. 6. P. 481–491.
19. Freiman C.V. Optimal error detection codes for completely asymmetric binary channels // Information and Control. 1962. V. 5, is. 1. P. 64–71.
20. Сапожников В.В., Сапожников Вл.В., Ефанов Д.В. Синтез схемы встроенного контроля для многовыходных комбинационных устройств на основе логического дополнения и сжатия сигналов // Известия вузов. Приборостроение. 2020. Т. 63, № 7. С. 581–597.
21. Сапожников В.В., Сапожников Вл.В., Ефанов Д.В., Пивоваров Д.В. Метод синтеза систем диагностирования на основе логического дополнения с уменьшенным числом элементов преобразования // Известия вузов. Приборостроение. 2019. Т. 62, № 12. С. 1039–1052.
22. Nikolos D. Self-Testing Embedded Two-Rail Checkers // Journal of Electronic Testing: Theory and Applications. 1998. V. 12, is. 1/2. P. 69–79.
23. Ефанов Д.В., Сапожников В.В., Сапожников Вл.В. Синтез самопроверяемых комбинационных устройств на основе выделения специальных групп выходов // Автоматика и телемеханика. 2018. № 9. С. 79–94.
24. Collection of Digital Design Benchmarks. URL: http://ddd.fit.cvut.cz/prj/Benchmarks/
25. Sentovich E.M., Singh K.J., Lavagno L., Moon C., Murgai R., Saldanha A., Savoj H., Stephan P.R., Brayton R.K., Sangiovanni-Vincentelli A. SIS: a System for Sequential Circuit Synthesis / Electronics Research Laboratory, Department of Electrical Engineering and Computer Science, University of California. Berkeley, 1992. 45 p.
26. Efanov D.V., Sapozhnikov V.V., Sapozhnikov Vl.V., Pivovarov D.V. The Synthesis Conditions of Completely Self-Testing Embedded-Control Circuits Based on the Boolean Complement Method to the «1-out-of-m» Constant-Weight Code // Automatic Control and Computer Sciences. 2020. V. 54, is. 2. P. 89–99.

Поступила в редакцию 3 сентября 2020 г.

Efanov D.V., Sapozhnikov V.V., Sapozhnikov Vl.V., Osadchy G.V. (2021) THE SYNTHESIS OF INTEGRATED CHECK CIRCUITS BASED ON THE BOOLEAN COMPLEMENT METHOD WITH THE PRELIMINARY COMPRESSION OF THE OPERATING FUNCTIONS SIGNALS. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie, vychislitelnaja tehnika i informatika* [Tomsk State University Journal of Control and Computer Science]. 54. pp. 97–115

DOI: 10.17223/19988605/54/12

Modern devices and control systems are increasingly implemented on the basis of microelectronic and microprocessor technology, which includes a large number of elementary components with their dense location relative to each other. The natural aging of components, external destabilizing factors, electromagnetic and radiation effects, etc. lead to the occurrence of failures and resistant faults. The use of microelectronic and microprocessor devices in critical application systems requires the fault parry, high fault-tolerance and survivability. The methods of the checkable and self-checking digital devices synthesis are widely used for this purpose in the control systems development.

Often, the synthesis of self-checking digital devices uses methods of coding theory and, directly, uniform block codes with low redundancy. Their use in many cases makes it possible to synthesize simpler self-checking devices than in duplication. The traditional structure of a self-checking device is a structure that includes a source block, an additional block that generates the values of special check functions, as well as a checker that checks the correspondence of the values of the operating functions of the source block and the check functions of the additional block. However, the author's research shows that an alternative structure that implements the idea of Boolean complement has more «flexibility». In this structure, in addition to the source and additional blocks, a cascade of transformation elements (two-input elements of addition by modulo two) is also used. It makes it possible to convert the values of all (or part) of the operating functions of the source block into the code word of the preselected uniform binary code, or into the values of functions belonging to one of the classes of Boolean functions, for example, the class of self-dual functions. The Boolean complement method due to a large number of variations in signal transformations on the correction elements makes it possible to get a large number of the additional block implementations, as well as more simply provide the self-checking of the integrated check circuit itself (conversion elements and the checker).

In this paper, the reader's attention is focused on the application of the preliminary signal compression with subsequent control of the vector of compressed functions based on constant-weight codes in the organization of integrated check circuits using the Boolean complement method. In this case, it is assumed to use the basic structure, which is constructed for six outputs of the controlled device. Signals from six outputs are compressed on three elements of addition by modulo two and make it possible to form a three-bit vector of compressed functions $<\varphi_3\ \varphi_2\ \varphi_1>$. Then the vector $<\varphi_3\ \varphi_2\ \varphi_1>$ is converted using the Boolean complement block to a four-bit vector $<h_4\ h_3\ h_2\ h_1>$, which belongs to the constant-weight code “1-out-of-4” and is fed to the inputs of the checker of this code. In this case, the h_4 function is formed directly by an additional block without any special transformation. This approach to the organization of the integrated check circuit makes it possible not only to minimize the number of check subcircuits from the "six" outputs of the source block, but also to significantly increase the detection ability of the structure. This is achieved by most frequently generating

one value at the input h_4 of the tester and zero values at the other inputs, because in this case any distortion in the vector<φ_3 φ_2 φ_1> will be fixed by the checker.

The authors show why the new structure will have an improved detection ability compared to other methods of the integrated check circuit synthesis using the Boolean complement method.

The article also provides the generalized structures for the organization of integrated check circuits based on constant-weight code "1-out-of-*n*".

The article severally highlights the features of selecting the groups of compressed outputs and a method that makes it possible to select the groups of compressed outputs in such a way that each such group eliminates masking errors on the inputs of elements of addition by modulo two. The method is based on searching for groups of dependent and independent outputs to compress the signals only in groups of independent outputs. The article describes a structured approach to searching for groups of compressed outputs. Examples are provided to illustrate this approach.

According to the authors, the use of the Boolean complement method with the scheme of preliminary compression of the signals is promising for solving problems of synthesis of self-checking digital systems.

Keywords: integrated control circuit; Boolean complement; signals compression scheme; constant-weight code "1-out-of-*n*"; constant-weight code "1-out-of-3"; constant-weight code "1-out-of-4"; constant-weight code "2-out-of-4"; totally self-checking structures.

EFANOV Dmitry Viktorovich (Doctor of Technical Sciences, Professor, Russian University of Transport, Moscow, Russian Federation; Peter the Great St. Petersburg Polytechnic University, St. Petersburg, Russian Federation).
E-mail: TrES-4b@yandex.ru

SAPOZHNIKOV Valery Vladimirovich (Doctor of Technical Sciences, Professor, Emperor Alexander I St. Petersburg State Transport University, St. Petersburg, Russian Federation).
E-mail: port.at.pgups@gmail.com

SAPOZHNIKOV Vladimir Vladimirovich (Doctor of Technical Sciences, Professor, Emperor Alexander I St. Petersburg State Transport University, St. Petersburg, Russian Federation).
E-mail: at.pgups@gmail.com

OSADCHY German Vladimirovich (Emperor Alexander I St. Petersburg State Transport University, St. Petersburg, Russian Federation).
E-mail: osgerman@mail.ru

REFERENCES

1. Sogomonyan, E.S. & Slabakov, E.V. (1989) *Samoproveryaemye ustroystva i otkazoustoychivye sistemy* [Self-checking devices and fault-tolerant systems]. Moscow: Radio i svyaz'.
2. Goessel, M. & Graf, S. (1994) *Error Detection Circuits*. London: McGraw-Hill.
3. Mitra, S. & McCluskey, E.J. (2000) Which concurrent error detection scheme to choose? *Proceedings of International Test Conference.* Atlantic City, NJ. October 3–5, 2000. pp. 985–994.
4. Drozd, O., Romankevich, V., Kuznietsov, M., Drozd, M. & Mertynyuk, O. (2020) Using natural version redundancy of FPGA projects in area of critical applications. *Proceedings of IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT)*. May 14–18, 2020. Kyiv. DOI: 10.1109/DESSERT50317.2020.9125050
5. Babeshko, E., Illiashenko, O., Kharchenko, V. & Ruchkiv, E. (2020) Safety and Reliability Assessment of NPP Instrumentation and Control Systems Considering Different Communication Architectures. *Nuclear and Radiation Safety*. 2(86). pp. 38–43. DOI: 10.32918/nrs.2020.2(86).05
6. Mertynyuk, O., Drozd, O., Stepova, H. & Martynyuk, D. (2019) Multi-Level Method of Behavioral Online Testing of Distributed Information Systems. *Proceedings of 10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*. September 18–21, 2019. Metz, France. DOI: 10.1109/IDAACS.2019.8924427
7. Drozd, A.V., Kharchenko, V.S., Antoshchuk, S.G., Drozd, Yu.V., Drozd, M.A. & Sulima, Yu.Yu. (2012) *Rabochee diagnostirovanie bezopasnykh informatsionno-upravlyayushchikh sistem* [On-line testing of the safe instrumentation and control systems]. Kharkov: National Aerospace University.
8. Sapozhnikov, V.V., Sapozhnikov, Vl.V. & Efanov, D.V.(2020) *Kody s summirovaniem dlya sistem tekhnicheskogo diagnostirovaniya* [Sum codes for technical diagnostics systems]. Vol. 1. Moscow: Nauka.
9. Piestrak, S.J. (1995) *Design of Self-Testing Checkers for Unidirectional Error Detecting Codes*. Wrocław: Oficyna Wydawnicza Politechniki Wrocławskiej.
10. Mikoni, S.V. (1992) *Obshchie diagnosticheskie bazy znaniy vychislitel'nykh sistem* [General Diagnostic Knowledge Bases of Computing Systems]. St. Petersburg: SPIIRAS.
11. Nikolos, D. & Kavousianos, X. (1999) Modular TSC checkers for bose-lin and bose codes. *Proceedings of the 17th IEEE VLSI Test Symposium*. April 25–29, 1999. Dana Point, USA. pp. 354–360.
12. Matrosova, A., Ostrovsky, V., Levin, I. & Nikitin, K. (2003) Designing FPGA based self-testing checkers for m-out-of-n codes. *Proceedings of the 9th IEEE International On-Line Testing Symposium (IOLTS'03)*. July 7–9, 2003. Kos Island, Greece. pp. 49–53.

13. Goessel, M., Morozov, A.V., Sapozhnikov, V.V. & Sapozhnikov, Vl.V. (2003) Logic Complement, a New Method of Checking the Combinational Circuits. *Avtomatika i telemekhanika – Automation and Remote Control*. 1. pp. 167–176. DOI: 10.1023/A:1021884727370
14. Göessel, M., Ocheretny, V., Sogomonyan, E. & Marienfeld, M. (2008) *New Methods of Concurrent Checking*. 1st ed. Dordrecht: Springer Science+Business Media B.V.
15. Das, D.K., Roy, S.S., Dmitiriev, A., Morozov, A. & Gössel, M. (2012) Constraint Don't Cares for Optimizing Designs for Concurrent Checking by 1-out-of-3 Codes. *Proceedings of the 10th International Workshops on Boolean Problems*. Freiberg, Germany. September, 2012. pp. 33–40.
16. Lala, P.K. (2001) *Self-Checking and Fault-Tolerant Digital Design*. San Francisco: Morgan Kaufmann Publishers.
17. Efanov, D.V., Sapozhnikov, V.V., Sapozhnikov, Vl.V. & Pivovarov, D.V. (2020) The Self-Checking Integrated Control Circuits Synthesis Based on The Boolean Complement Method to "2-out-of-4" Constant-Weight Code. *Informatika – Informatics.* 15(4). pp. 71–85.
18. Efanov, D.V., Sapozhnikov, V.V., Sapozhnikov, Vl.V. & Pivovarov, D.V. (2019) Synthesis of Built-in Self-Test Control Circuits Based on the Method of Boolean Complement to Constant-Weight 1-out-of-n Codes. *Automatic Control and Computer Sciences*. 53(6). pp. 481–491. DOI: 10.3103/S014641161906004X
19. Freiman, C.V. (1962) Optimal Error Detection Codes for Completely Asymmetric Binary Channels. *Information and Control*. 5(1). pp. 64–71. DOI: 10.1016/S0019-9958(62)90223-1.
20. Sapozhnikov, V.V., Sapozhnikov, Vl.V. & Efanov, D.V. (2020) Synthesis of a Built-In Control Circuit for Multi-Output Combinational Devices Based on Logical Complement and Signal Compression. *Izvestiya vuzov. Priborostroenie – Journal of Instrument Engineering.* 63(7). pp. 581–597. DOI: 10.17586/0021-3454-2020-63-7-581-597
21. Sapozhnikov, V.V., Sapozhnikov, Vl.V., Efanov, D.V. & Pivovarov, D.V. (2019) Method of Diagnostic System Synthesis Based on Boolean Complement with Reduced Number of Correction Elements. *Journal of Instrument Engineering.* 62(12). pp. 1039–1052. DOI: 10.17586/0021-3454-2019-62-12-1039-1052
22. Nikolos, D. (1998) Self-Testing Embedded Two-Rail Checkers. *Journal of Electronic Testing: Theory and Applications.* 12(1/2). pp. 69–79. DOI: 10.1023/A:1008281822966
23. Efanov, D.V., Sapozhnikov, V.V. & Sapozhnikov, Vl.V. (2018) Synthesis of Self-Checking Combination Devices Based on Allocating Special Groups of Outputs. *Avtomatika i telemekhanika – Automation and Remote Control*. 79(9). pp. 1609–1620. DOI: 10.1134/S0005117918090060
24. *Collection of Digital Design Benchmarks*. [Online] Available from: http://ddd.fit.cvut.cz/prj/Benchmarks/.
25. Sentovich, E.M., Singh, K.J., Lavagno, L., Moon, C., Murgai, R., Saldanha, A., Savoj, H., Stephan, P.R., Brayton, R.K. & Sangiovanni-Vincentelli, A. (1992) *SIS: A System for Sequential Circuit Synthesis*. Technical Report No. UCB/ERL M92/41. May 1992. Electronics Research Laboratory, Department of Electrical Engineering and Computer Science, University of California, Berkeley.
26. Efanov, D.V., Sapozhnikov, V.V., Sapozhnikov, Vl.V. & Pivovarov, D.V. (2020) The Synthesis Conditions of Completely Self-Testing Embedded-Control Circuits Based on the Boolean Complement Method to the "1-out-of-m" Constant-Weight Code. *Automatic Control and Computer Sciences.* 54(2). pp. 89–99.