
УДК 681.518.5:004.052.32

В.В. Сапожников, д-р техн. наук, **Вл.В. Сапожников**, д-р техн. наук,
Д.В. Ефанов, канд. техн. наук, **Д.В. Пивоваров**, аспирант
Петербургский государственный университет
путей сообщения Императора Александра I
(Российская Федерация, 190031, Санкт-Петербург, Московский пр-т, 9,
тел. (+7) 9117092164, (+7) (812) 4578579; e-mail: TrES-4b@yandex.ru)

Метод логического дополнения на основе равновесного кода «1 из 4» для построения полностью самопроверяемых структур систем функционального контроля

Предложен способ, позволяющий формализовать правила вычисления значений функций логического дополнения (ФЛД) в системе функционального контроля (СФК) по равновесному коду «1 из 4». При этом исключается процедура подбора значений ФЛД и обеспечивается свойство полной самопроверяемости структуры, т.е. гарантированно тестируются все элементы сложения по модулю два в блоке логического дополнения и тестер. Установлено число способов доопределения ФЛД в СФК по коду «1 из 4» при дополнении только трех рабочих функций, а также минимально необходимое для обеспечения свойства полной самопроверяемости множество рабочих векторов.

К л ю ч е в ы е с л о в а: система функционального контроля, логическое дополнение, равновесный код, код «1 из 4», полностью самопроверяемая структура.

Запропоновано спосіб, який дозволяє формалізувати правила обчислювання значень функцій логічного доповнення (ФЛД) в системі функціонального контролю (СФК) по рівноваговому коду «1 з 4». При цьому виключено процедуру підбору значень ФЛД і забезпечується властивість повної самоперевіряємості структури, тобто гарантовано тестуються всі елементи складання по модулю два в блоці логічного доповнення і тестер. Встановлено кількість способів до визначення ФЛД в СФК по коду «1 з 4» при добавлянні тільки трьох робочих функцій, а також мінімально необхідна для забезпечення повної самоперевіряємості множина робочих векторів.

К л ю ч о в і с л о в а: система функціонального контролю, логічне доповнення, рівноваговий код, код «1 з 4», повністю самоперевіряема структура.

Для реализации процедуры технического диагностирования логического устройства в процессе выполнения им своих функций организуются системы рабочего (функционального) контроля (СФК) [1—3]. В таких системах объект диагностирования $F(x)$, вычисляющий значения рабочих буле-

© В.В. Сапожников, Вл.В. Сапожников, Д.В. Ефанов, Д.В. Пивоваров, 2017

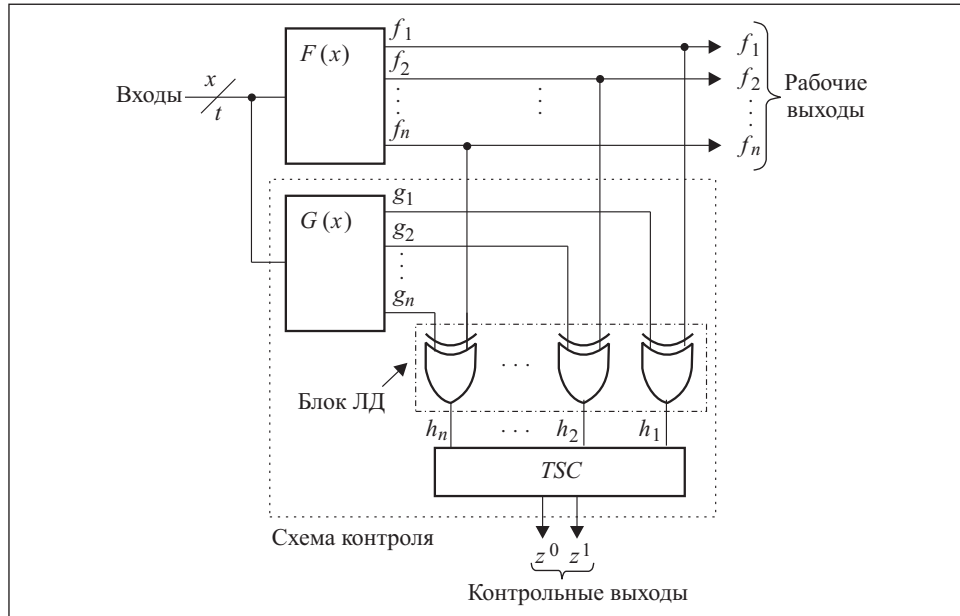


Рис. 1. Структурная схема реализации СФК методом ЛД

вых функций f_1, f_2, \dots, f_n , снабжается специализированной схемой контроля, позволяющей косвенно, по результатам вычислений, фиксировать в нем неисправности [4]. Наиболее часто при организации СФК используют помехоустойчивые коды с небольшой избыточностью (как разделимые, так и неразделимые равновесные коды) [5—14].

Рассмотрим один из способов организации СФК, основанный на идее логического дополнения (ЛД) (рис. 1) [15]. Схема контроля строится так, чтобы информационный вектор $\langle f_n f_{n-1} \dots f_1 \rangle$ при любых входных значениях был преобразован в кодовое слово $\langle h_n h_{n-1} \dots h_1 \rangle$ заранее выбранного кода, что контролируется схемой тестера TSC . В схему контроля кроме тестера помещены блок контрольной логики $G(x)$ и блок ЛД. Блок $G(x)$ вычисляет значения функций дополнения g_1, g_2, \dots, g_n , позволяющих преобразовать значения соответствующих разрядов f_i информационного вектора в значения разрядов h_i заранее выбранного помехоустойчивого кода: $h_i = f_i \oplus g_i, i=1, m$. Сформированное кодовое слово $\langle h_n h_{n-1} \dots h_1 \rangle$ поступает на входы тестера TSC .

Важной задачей при организации СФК является обеспечение свойства полной самопроверяемости ее структуры относительно заданного класса неисправностей (как правило, это одиночные константные неисправности на выходах логических элементов внутренней структуры) [16]. Ключевой

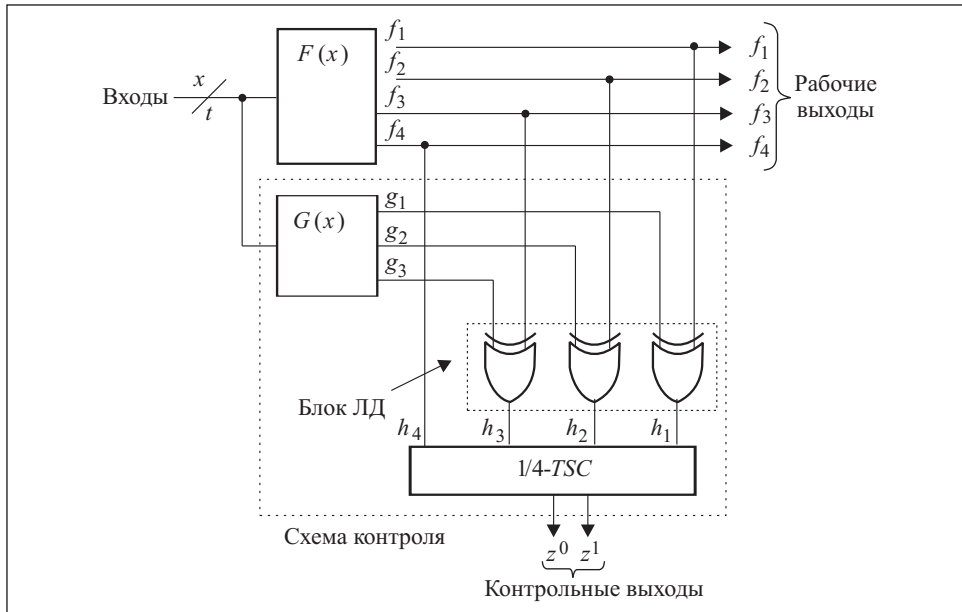


Рис. 2. Базовая структурная схема СФК на основе 1/4-кода

фактор, влияющий на данное свойство, — возможность полной проверки блока ЛД и схемы тестера [17, 18]. Поэтому организовать СФК наиболее просто, используя разделимые коды с небольшой длиной контрольных векторов (например, коды Бергера или коды Боуза—Лина [19, 20]) или неразделимые равновесные коды с небольшим числом разрешенных кодовых слов (например, коды «1 из 3», «1 из 4» или «2 из 4» [21]). Приложения данных кодов при организации СФК на основе метода ЛД рассмотрены во многих публикациях [22—32].

Весьма перспективным в задачах организации СФК является использование равновесного кода «1 из 4» (1/4-кода), имеющего всего четыре разрешенных кодовых слова: {0001; 0010; 0100; 1000} [33]. При их использовании выходы объекта диагностирования f_1, f_2, \dots, f_n (см. рис. 1) делятся на группы по четыре выхода в каждой (одни и те же выходы могут входить в разные группы), а затем для каждой группы выходов организуется отдельная СФК на основе базовой структуры, представленной на рис. 2. Выходы отдельных схем контроля объединяются на входах самопроверяемого компаратора, реализуемого на основе модулей сжатия парафазных сигналов [34, 35].

Для минимизации аппаратных затрат и упрощения процедуры тестирования элементов в базовой структурной схеме СФК на основе 1/4-кода

используется блок ЛД, состоящий из трех элементов сложения по модулю два (элементов *XOR*). Такое количество элементов *XOR* обусловлено необходимостью преобразования значений только трех рабочих функций для получения векторов $\langle h_4 h_3 h_2 h_1 \rangle$, принадлежащих 1/4-коду.

В известных работах при построении СФК методом ЛД на основе 1/4-кода предполагается подбор значений контрольных функций, позволяющих обеспечивать полную самопроверяемость структуры [15, 18, 24]. Рассмотрим способ, позволяющий формализовать правила вычисления контрольных функций g_1 , g_2 и g_3 в структурной схеме, изображенной на рис. 2, и исключить процедуру направленного перебора значений при построении СФК с полностью самопроверяемой структурой.

Постановка задачи формирования функций ЛД (ФЛД). В структурной схеме СФК, изображенной на рис. 2, при преобразовании вектора $\langle f_4 f_3 f_2 f_1 \rangle$ в вектор $\langle h_4 h_3 h_2 h_1 \rangle$ не требуется вносить изменения в значения функции f_4 . Поэтому в блоке ЛД используется только три элемента сложения по модулю два, преобразующих значения функций f_1 , f_2 и f_3 . При преобразовании значений рабочих функций для обеспечения свойства полной самопроверяемости структуры требуется выполнить два условия:

- 1) обеспечить подачу полного множества тестовых наборов для тестера 1/4-кода;
- 2) обеспечить подачу полного множества тестовых наборов для каждого элемента сложения по модулю два в блоке ЛД.

На рис. 3 представлены структуры 1/4-*TSC* [36] и каноническая структура элемента *XOR*. Для полной проверки тестера требуется подача на его входы всех разрешенных кодовых слов 1/4-кода: {0001; 0010; 0100; 1000} [33]. Для тестирования элемента *XOR* с канонической структурой требуется подача всех четырех рабочих комбинаций {00; 01; 10; 11} [37, 38].

Анализируя табл. 1, можно сделать вывод о том, что в нижней ее половине значения всех ФЛД однозначно определены: они позволяют преобразовать любой вектор рабочих функций в вектор 1/4-кода $\langle 1000 \rangle$, при этом на каждый элемент *XOR* подается по две тестовые комбинации: $\langle 00 \rangle$ и $\langle 11 \rangle$. Таким образом, для обеспечения свойства полной самопроверяемости структуры требуется преобразовать векторы рабочих функций в векторы 1/4-кода с учетом необходимости формирования хотя бы по одному разу для каждого элемента *XOR* комбинаций $\langle 01 \rangle$ и $\langle 10 \rangle$, а также комбинаций $\langle 0001 \rangle$, $\langle 0010 \rangle$ и $\langle 0100 \rangle$, необходимых для полной проверки 1/4-*TSC*.

Вычисление ФЛД с учетом необходимости обеспечения полной самопроверяемости структуры. Рассмотрим верхнюю половину табл. 1. Каждый из восьми информационных векторов $\langle f_4 f_3 f_2 f_1 \rangle$ можно преобразовать в вектор $\langle h_4 h_3 h_2 h_1 \rangle$ 1/4-кода тремя различными способами,

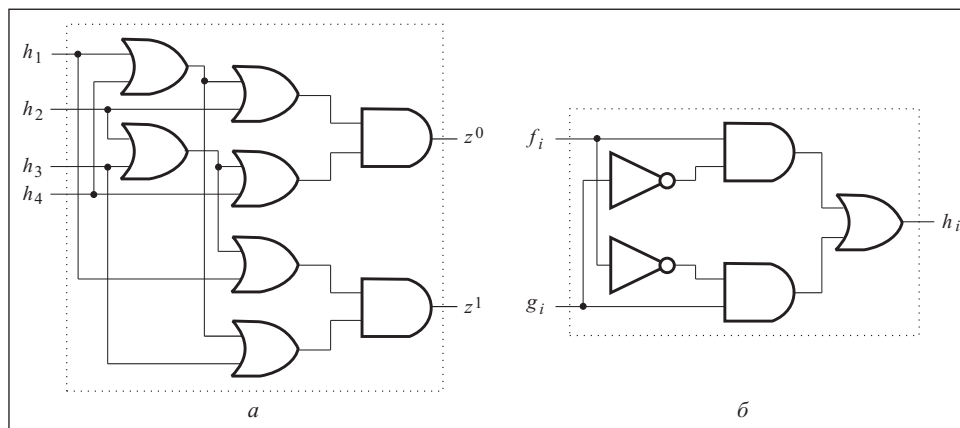


Рис. 3. Структурные схемы: а — 1/4-TSC; б — элемент XOR

Таблица 1. Исходная таблица ФЛД

Информационный вектор				ФЛД				Кодовое слово 1/4-кода			
f_4	f_3	f_2	f_1	g_4	g_3	g_2	g_1	h_4	h_3	h_2	h_1
0	0	0	0	0				0			
0	0	0	1	0				0			
0	0	1	0	0				0			
0	0	1	1	0				0			
0	1	0	0	0				0			
0	1	0	1	0				0			
0	1	1	0	0				0			
0	1	1	1	0				0			
1	0	0	0	0	0	0	0	1	0	0	0
1	0	0	1	0	0	0	1	1	0	0	0
1	0	1	0	0	0	1	0	1	0	0	0
1	0	1	1	0	0	1	1	1	0	0	0
1	1	0	0	0	1	0	0	1	0	0	0
1	1	0	1	0	1	0	1	1	0	0	0
1	1	1	0	0	1	1	0	1	0	0	0
1	1	1	1	0	1	1	1	1	0	0	0

получив, соответственно, векторы $\langle 0001 \rangle$, $\langle 0010 \rangle$ или $\langle 0100 \rangle$. Для обеспечения полной проверки 1/4-TSC требуется на восьми первых векторах табл. 1 сформировать хотя бы по одному разу указанные векторы рабочих функций. Однако ФЛД не могут быть доопределены только с учетом этой особенности.

Необходимо также сформировать для каждого элемента XOR в блоке ЛД хотя бы по одному разу комбинации $\langle 01 \rangle$ и $\langle 10 \rangle$. Тестовая комбинация i -го элемента XOR $\langle f_i g_i \rangle = \langle 01 \rangle$ будет сформирована тогда, когда исходный вектор рабочих функций будет дополнен до вектора 1/4-кода, имеющего единичное значение разряда h_i и нулевые значения всех остальных разрядов. Другие разряды информационного вектора дополняются значениями 0, если они были равны 0, и значением 1, если они были равны 1, т.е. остальные значения в информационном векторе, кроме f_i , преобразуются в нулевые значения вектора 1/4-кода.

Аналогичная ситуация наблюдается при формировании тестовой комбинации i -го элемента XOR $\langle f_i g_i \rangle = \langle 10 \rangle$. Следовательно, для формирования обеих тестовых комбинаций, $\langle 01 \rangle$ и $\langle 10 \rangle$, для i -го элемента XOR потребуется дважды сформировать один и тот же вектор 1/4-кода, имеющий единичное значение i -го разряда. Незаполненные ячейки табл. 1 в столбцах g_1 , g_2 и g_3 должны быть заполнены с учетом этого факта.

В табл. 2 представлены все возможные способы доопределения функций g_1 , g_2 и g_3 для каждого из восьми информационных векторов, позволяющие решать задачу формирования одной из тестовых комбинаций 1/4-TSC при преобразовании соответствующего информационного вектора.

Таблица 2. Способы доопределения ФЛД

Номер строки	Преобразуемый разряд информационного вектора	Вариант доопределения функций преобразования g_3, g_2, g_1	Получаемый вектор 1/4-кода
k	f_3, f_2, f_1	3 / 2 / 1	3 / 2 / 1
1	000	001 / 010 / 100	0001 / 0010 / 0100
2	001	000 / 011 / 101	0001 / 0010 / 0100
3	010	011 / 000 / 110	0001 / 0010 / 0100
4	011	010 / 001 / 111	0001 / 0010 / 0100
5	100	101 / 110 / 000	0001 / 0010 / 0100
6	101	100 / 111 / 001	0001 / 0010 / 0100
7	110	111 / 100 / 010	0001 / 0010 / 0100
8	111	110 / 101 / 011	0001 / 0010 / 0100

Необходимо установить, каким способом следует доопределить функции g_1 , g_2 и g_3 на каждом из восьми векторов рабочих функций верхней половины табл. 1. Введем в рассмотрение условную переменную a_k^j ($j = 1, 2, 3$), определяющую, какой из способов доопределения функций g_1 , g_2 и g_3 следует использовать для k -й строки. Например, запись a_1^1 означает доопределение функций информационного вектора из строки $k = 1$ способом $\langle g_3 \ g_2 \ g_1 \rangle = \langle 100 \rangle$, расположенным на позиции $j = 1$ (см. табл. 2). Рассмотрим также функцию γ_j , определяющую те варианты дополнения каждого информационного вектора, на которых формируются тестовые комбинации $\langle 01 \rangle$ и $\langle 10 \rangle$ элемента XOR , преобразующего j -й разряд информационного вектора (на этих же комбинациях формируется вектор 1/4-кода с j -м единичным и остальными нулевыми разрядами).

Например, для формирования тестовой комбинации $\langle 01 \rangle$ для элемента XOR , преобразующего функцию f_3 , необходимо выполнить условие $(a_1^1 \vee a_2^1 \vee a_3^1 \vee a_4^1)$, т.е. преобразовать хотя бы один из первых четырех информационных векторов первым способом. Аналогично выглядит условие формирования тестовой комбинации $\langle 10 \rangle$ — $(a_5^1 \vee a_6^1 \vee a_7^1 \vee a_8^1)$, что означает преобразование хотя бы одного из следующих четырех информационных векторов (векторов с номерами 5—8 в табл. 2) первым способом. Для обеспечения формирования обеих комбинаций, $\langle 01 \rangle$ и $\langle 10 \rangle$, для тестирования элемента XOR , осуществляющего преобразование функции f_3 , необходимо воспользоваться следующим логическим выражением, определяющим, как и какие строки следует дополнять:

$$\gamma_3 = (a_1^1 \vee a_2^1 \vee a_3^1 \vee a_4^1)(a_5^1 \vee a_6^1 \vee a_7^1 \vee a_8^1). \quad (1)$$

Из (1) следует, что число дополнений строк табл. 1 для обеспечения формирования недостающих тестовых комбинаций элемента XOR , осуществляющего преобразование функции f_3 , равно 16. Все преобразования выполнены первым способом.

Аналогично выглядят условия доопределения ФЛД с учетом обеспечения формирования комбинаций $\langle 01 \rangle$ и $\langle 10 \rangle$ для тестирования элементов XOR , осуществляющих преобразования функций f_2 и f_1 :

$$\gamma_2 = (a_1^2 \vee a_2^2 \vee a_5^2 \vee a_6^2)(a_3^2 \vee a_4^2 \vee a_7^2 \vee a_8^2), \quad (2)$$

$$\gamma_1 = (a_1^3 \vee a_3^3 \vee a_5^3 \vee a_7^3)(a_2^3 \vee a_4^3 \vee a_6^3 \vee a_8^3). \quad (3)$$

Существует также по 16 вариантов дополнения функций g_2 и g_1 с учетом необходимости формирования недостающих тестовых комбинаций для элементов XOR , осуществляющих преобразования функций f_2 и f_1 .

Раскрывая скобки в выражениях (1)—(3) и приводя запись к дизъюнктивной нормальной форме, получаем выражения вида $a_{k_1}^j a_{k_2}^j$, где k_1 и k_2 — номера строк, для которых используется j -й вариант доопределения функций g_1 , g_2 и g_3 . Учесть все возможные варианты ЛД можно, составив специальную таблицу покрытия (табл. 3). В строках табл. 3 указаны пары дополняемых строк, а в столбцах — функции γ_j , определяющие «принадлежность» пары строк к способу доопределения. Как видно из табл. 3, некоторые пары строк могут быть заполнены различными способами. Всего восемь строк можно заполнить количеством способов $C_8^2 = 28$, каждый из которых присутствует в табл. 3. Знак покрытия «x» поставлен на пересечении тех строк и столбцов, которые удовлетворяют соответствующему выражению для γ_j . Учитывая тот факт, что $a_k^1 a_k^2 = 0$, $a_k^1 a_k^3 = 0$ и $a_k^2 a_k^3 = 0$ (поскольку элементы a_k^1 , a_k^2 и a_k^3 определяют различные способы заполнения одной и той же строки), можно выделить однозначные (существенные) способы E дополнения строк, которые указаны в последнем столбце табл. 3.

Анализируя столбец E , можно выделить все возможные способы доопределения функций g_1 , g_2 и g_3 . Запишем выражение вида «конъюнкция дизъюнкций» тех подмножеств $a_{k_1}^j a_{k_2}^j$, для которых в столбце E стоит знак *:

$$\Psi = \gamma_3 \gamma_2 \gamma_1 = (a_1^3 a_2^3 \vee a_3^3 a_4^3 \vee a_5^3 a_6^3 \vee a_7^3 a_8^3) \times \\ \times (a_1^2 a_3^2 \vee a_2^2 a_4^2 \vee a_5^2 a_7^2 \vee a_6^2 a_8^2) (a_1^1 a_5^1 \vee a_2^1 a_6^1 \vee a_3^1 a_7^1 \vee a_4^1 a_8^1). \quad (4)$$

Раскроем в выражении (4) скобки и представим его в дизъюнктивной нормальной форме:

$$\Psi = \gamma_3 \gamma_2 \gamma_1 = a_3^1 a_7^1 a_6^2 a_8^2 a_1^3 a_2^3 \vee a_4^1 a_8^1 a_5^2 a_7^2 a_1^3 a_2^3 \vee a_1^1 a_5^1 a_6^2 a_8^2 a_3^3 a_4^3 \vee \\ \vee a_2^1 a_6^1 a_5^2 a_7^2 a_3^3 a_4^3 \vee a_3^1 a_7^1 a_2^2 a_4^2 a_5^3 a_6^3 \vee a_4^1 a_8^1 a_1^2 a_3^2 a_5^3 a_6^3 \vee \\ \vee a_1^1 a_5^1 a_2^2 a_4^2 a_7^3 a_8^3 \vee a_2^1 a_6^1 a_1^2 a_3^2 a_7^3 a_8^3. \quad (5)$$

Каждая конъюнкция в выражении (5) определяет способ заполнения шести из восьми строк верхней половины табл. 1 с учетом обеспечения условия полной самопроверяемости как элементов блока ЛД, так и тестера 1/4-кода. Две неопределенные конкретной конъюнкцией строки могут быть заполнены любым из трех способов, указанных в табл. 2. Исходя из этого сформулируем следующее утверждение.

Утверждение 1. Существует 72 способа построения СФК с полностью самопроверяемой структурой методом ЛД по 1/4-коду и преобразованием только трех рабочих функций.

Предложенные способы доопределения ФЛД позволяют как минимум по два раза сформировать тестовые комбинации <0001>, <0010> и <0100>, а также восемь раз комбинацию <1000>. Все необходимые для полной проверки элементов *XOR* в блоке ЛД комбинации формируются при подаче шести комбинаций, определяемых выбранной конъюнкцией в выражении (4). Указанное обстоятельство приводит к следующему утверждению.

Таблица 3. Множество способов доопределения строк табл. 1

k_1, k_2	γ_3	γ_2	γ_1	E
1, 2			×	*
1, 3		×		*
1, 4		×	×	
1, 5	×			*
1, 6	×		×	
1, 7	×	×		
1, 8	×	×	×	
2, 3		×	×	
2, 4		×		*
2, 5	×		×	
2, 6	×			*
2, 7	×	×	×	
2, 8	×	×		
3, 4			×	*
3, 5	×	×		
3, 6	×	×	×	
3, 7	×			*
3, 8	×		×	
4, 5	×	×	×	
4, 6	×	×		
4, 7	×		×	
4, 8	×			*
5, 6			×	*
5, 7		×		*
5, 8		×	×	
6, 7		×	×	
6, 8		×		*
7, 8			×	*

Утверждение 2. Минимальное число информационных векторов, необходимых для обеспечения свойства полной самопроверяемости структуры СФК, составляет семь наборов, включая шесть определяемых выбранной конъюнкцией в (5), и один набор из нижней половины табл. 1.

Пример. Рассмотрим один из вариантов ЛД информационного вектора до вектора 1/4-кода, позволяющий строить полностью самопроверяемые структуры СФК. Согласно первой конъюнкции выражения (5), $a_3^1 a_7^1 a_6^2 a_8^2 a_1^3 a_2^3$, строки с номерами 3 и 7 заполняются первым способом, строки 6 и 8 — вторым способом, а строки 1 и 2 — третьим способом (см. табл. 2). Строки с номерами 4 и 5 остаются пустыми, однако они должны быть также доопределены, любым способом, указанным в табл. 2. Доопределим их так, чтобы на входах тестера формировалось кодовое слово <0100>.

В табл. 4 представлен полученный вариант доопределения ФЛД. Выписывая и минимизируя функции g_1 , g_2 и g_3 , получаем

$$\begin{aligned} g_3 &= \overline{f_4} \overline{f_3} f_2 \vee f_3 f_1 \vee f_4 f_3, \\ g_2 &= \overline{f_4} f_3 \overline{f_2} f_1 \vee f_2 \overline{f_1} \vee \overline{f_3} f_2 \vee f_4 f_2, \\ g_1 &= \overline{f_4} \overline{f_3} \overline{f_2} \overline{f_1} \vee f_3 f_1 \vee f_4 f_1 \vee f_2 f_1, \end{aligned}$$

или

$$\begin{aligned} g_3 &= \overline{f_4} \overline{f_3} f_2 \vee f_3 (f_4 \vee f_1), \\ g_2 &= \overline{f_4} f_3 \overline{f_2} f_1 \vee f_2 (f_4 \vee \overline{f_3} \vee \overline{f_1}), \\ g_1 &= \overline{f_4} \overline{f_3} \overline{f_2} \overline{f_1} \vee f_1 (f_4 \vee f_3 \vee f_2). \end{aligned} \tag{6}$$

Результаты экспериментов. В ходе исследований был проведен эксперимент с контрольными комбинационными схемами, имеющими четыре выхода [39], для изучения влияния перестановки выходов в контролируемой группе на показатели структурной избыточности СФК, построенной по схеме на рис. 2. С использованием специально разработанного программного комплекса получены файлы-описания блоков СФК, что позволило оценить сложность их технической реализации в условном показателе площади, занимаемой устройством на кристалле. Результаты получены с применением интерпретатора SIS в библиотеке функциональных элементов *stdcell2_2.genlib* [40].

Поскольку ФЛД в системе (6) являются различными и ими дополняются различные рабочие функции контрольной комбинационной схемы, имеет значение, в какой последовательности расположены разряды в информационном векторе. По этой причине в эксперименте синтезирована

лись все возможные варианты СФК, построенной методом ЛД (4!=24 варианта) для каждой комбинационной схемы. В табл. 5 приведены результаты эксперимента для схемы «*sao2*», имеющей 10 входов и 4 выхода. Для оценки эффективности предлагаемого варианта ЛД были рассчитаны относительные показатели структурной избыточности (%):

$$\delta_{F(x)} = 100 \frac{L_{1/4}}{L_{F(x)}}, \delta_D = 100 \frac{L_{1/4}}{L_D}.$$

Коэффициент $\delta_{F(x)}$ характеризует долю площади СФК, построенной методом ЛД ($L_{1/4}$), от площади контролируемой комбинационной схемы ($L_{F(x)}$), а коэффициент δ_D — аналогичное отношение величины $L_{1/4}$ к площади системы дублирования (L_D). От того, как расположены выходы в контролируемой группе («четверке»), зависит сложность технической реализации СФК: величина показателя $\delta_{F(x)}$ колеблется от 165,707 до 198,691, а показатель δ_D — от 75 до 89,929.

Таблица 4. Вариант ФЛД

Информационный вектор				ФЛД				Кодовое слово 1/4-кода			
f_4	f_3	f_2	f_1	g_4	g_3	g_2	g_1	h_4	h_3	h_2	h_1
0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	1	0	0	0	0	0	0	0	1
0	0	1	0	0	1	1	0	0	1	0	0
0	0	1	1	0	1	1	1	0	1	0	0
0	1	0	0	0	0	0	0	0	1	0	0
0	1	0	1	0	1	1	1	0	0	1	0
0	1	1	0	0	0	1	0	0	1	0	0
0	1	1	1	0	1	0	1	0	0	1	0
1	0	0	0	0	0	0	0	1	0	0	0
1	0	0	1	0	0	0	1	1	0	0	0
1	0	1	0	0	0	1	0	1	0	0	0
1	0	1	1	0	0	1	1	1	0	0	0
1	1	0	0	0	1	0	0	1	0	0	0
1	1	0	1	0	1	0	1	1	0	0	0
1	1	1	0	0	1	1	0	1	0	0	0
1	1	1	1	0	1	1	1	1	0	0	0

Таблица 5. Результаты экспериментов с контрольной схемой «sao2»

Выходы в группе	Площадь, у.е.				Показатель структурной избыточности, %					
	$F(x)$	$G(x)$	1/4-TSC	Система дублирования	Система контроля по 1/4-коду	$\delta_{F(x)}$	δ_D			
$f_1 f_2 f_3 f_4$	3056	2200	256	6752	5632	184,293	83,412			
$f_1 f_2 f_4 f_3$		2512			5944	194,503	88,033			
$f_1 f_3 f_2 f_4$		2640			6072	198,691	89,929			
$f_1 f_3 f_4 f_2$		2176			5608	183,508	83,057			
$f_1 f_4 f_2 f_3$		2296			5728	187,435	84,834			
$f_1 f_4 f_3 f_2$		2192			5624	184,031	83,294			
$f_2 f_1 f_3 f_4$		2168			5600	183,246	82,938			
$f_2 f_1 f_4 f_3$		2352			5784	189,267	85,664			
$f_2 f_3 f_1 f_4$		2080			5512	180,366	81,635			
$f_2 f_3 f_4 f_1$		1920			5352	175,131	79,265			
$f_2 f_4 f_1 f_3$		1936			5368	175,654	79,502			
$f_2 f_4 f_3 f_1$		1984			5416	177,225	80,213			
$f_3 f_1 f_2 f_4$		2176			5608	183,508	83,057			
$f_3 f_1 f_4 f_2$		2088			5520	180,628	81,754			
$f_3 f_2 f_1 f_4$		2392			5824	190,576	86,256			
$f_3 f_2 f_4 f_1$		2392			5824	190,576	86,256			
$f_3 f_4 f_1 f_2$		1936			5368	175,654	79,502			
$f_3 f_4 f_2 f_1$		1976			5408	176,963	80,095			
$f_4 f_1 f_2 f_3$		2152			5584	182,723	82,701			
$f_4 f_1 f_3 f_2$		2384			5816	190,314	86,137			
$f_4 f_2 f_1 f_3$		2048			5480	179,319	81,161			
$f_4 f_2 f_3 f_1$		2144			5576	182,461	82,583			
$f_4 f_3 f_1 f_2$		1632			5064	165,707	75			
$f_4 f_3 f_2 f_1$		1656			5088	166,492	75,355			
Среднее значение		3056			2143	256	6752	5575	182,428	82,568

Во всех случаях для схемы «sao2» получено уменьшение площади технической реализации СФК по 1/4-коду по сравнению с использованием стандартной структуры дублирования, что свидетельствует о высокой эффективности предлагаемого метода. Для других комбинационных схем получены аналогичные результаты: наблюдается разброс значений показателей $\delta_{F(x)}$ и δ_D , а подбором варианта расположения разрядов в информационном векторе можно добиться максимального уменьшения структурной избыточности до значения, меньшего, чем у структуры дублирования.

Выводы

Описанный подход к формализации правил доопределения ФЛД при организации СФК на основе равновесного 1/4-кода дает возможность формирования полного множества формул дополнения, позволяющих на практике организовывать структуры, обладающие свойством полной самопроверяемости. При этом, в отличие от предложенных в [15, 18, 24] способов логического дополнения, правила, полученные на основе описанного подхода, гарантируют подачу всех проверочных комбинаций как на входы элементов *XOR* блока логического дополнения, так и на входы 1/4-*TSC* без процедуры целенаправленного перебора.

Возможны 72 варианта ЛД информационных векторов до векторов 1/4-кода при преобразовании только трех рабочих функций. Каждый из вариантов подразумевает для полной самопроверяемости структуры СФК формирование как минимум семи определенных рабочих комбинаций. Данное ограничение, однако, не является жестким, так как для каждого конкретного случая могут быть подобраны соответствующие варианты ЛД. Если решить задачу указанным способом не удастся, всегда можно увеличить число элементов *XOR* в блоке ЛД, решив задачу при дополнении всех четырех рабочих функций, а для многовыходных схем — перегруппировать выходы в «четверках».

Предложенный подход может быть обобщен и на случай использования других равновесных кодов, например 1/*n*-кодов. Однако с увеличением значения *n* число способов ЛД существенно возрастает, как возрастает и сложность 1/*n*-*TSC* и блока ЛД, а кроме того, увеличивается и множество необходимых для обеспечения свойства полной самопроверяемости структуры СФК формируемых рабочих векторов.

Полученные результаты расширяют теорию функционального контроля логических устройств автоматики и вычислительной техники и позволяют ожидать хороших перспектив для приложения равновесного 1/4-кода при организации систем диагностирования.

СПИСОК ЛИТЕРАТУРЫ

1. *Согомонян Е.С., Слабаков Е.В.* Самопроверяемые устройства и отказоустойчивые системы. — М. : Радио и связь, 1989. — 208 с.
2. *Дрозд А.В., Харченко В.С., Антощук С.Г. и др.* Рабочее диагностирование безопасных информационно-управляющих систем / Под ред. А.В. Дрозда и В.С. Харченко. — Харьков: Национальный аэрокосмический университет им. Н.Е. Жуковского «ХАИ», 2012. — 614 с.
3. *Kharchenko V., Kondratenko Yu., Kasprzyk J.* Green IT Engineering: Concepts, Models, Complex Systems Architectures // Springer Book series «Studies in Systems, Decision and Control». — 2017. — Vol. 74. — 305 p. — DOI: 10.1007/978-3-319-44162-7.

4. *Mitra S., McCluskey E.J.* Which Concurrent Error Detection Scheme to Choose? // Proc. of Intern. Test Conference, 2000. — USA, Atlantic City, NJ, 03-05 October 2000. — P. 985—994.
5. *Слабаков Е.В., Согомонян Е.С.* Построение полностью самопроверяемых комбинационных устройств с использованием равновесных кодов // Автоматика и телемеханика. — 1980. — № 9. — С. 173—181.
6. *Слабаков Е.В., Согомонян Е.С.* Самопроверяемые вычислительные устройства и системы (обзор) // Там же. — 1981. — № 11. — С. 147—167.
7. *Piestrak S.J.* Design of Self-Testing Checkers for Unidirectional Error Detecting Codes. — Wroclaw: Oficyna Wydawnicza Politechniki Wroclawskiej, 1995. — 111 p.
8. *Das D., Touba N.A.* Synthesis of Circuits with Low-Cost Concurrent Error Detection Based on Bose-Lin Codes // Journal of Electronic Testing: Theory and Applications. — 1999. — Vol. 15, Issue 1-2. — P. 145—155. — DOI: 10.1023/A:1008344603814.
9. *Nicolaidis M., Zorian Y.* On-Line Testing for VLSI — A Compendium of Approaches // Ibid. — 1998. — № 12. — P. 7—20.
10. *Das D., Touba N.A.* Weight-Based Codes and Their Application to Concurrent Error Detection of Multilevel Circuits // Proceedings of 17th IEEE Test Symposium. — USA, California, 1999. — P. 370—376.
11. *Matrosova A.Yu., Levin I., Ostanin S.A.* Self-Checking Synchronous FSM Network Design with Low Overhead // VLSI Design. — 2000. — Vol. 11, Issue 1. — P. 47—58. — DOI:10.1155/2000/46578.
12. *Das D., Touba N.A., Seuring M., Gossel M.* Low Cost Concurrent Error Detection Based on Modulo Weight-Based Codes // Proc. of IEEE 6th International On-Line Testing Workshop (IOLTW). — Spain, Palma de Mallorca, July 3-5, 2000. — P. 171—176.
13. *Matrosova A., Levin I., Ostanin S.* Survivable Self-Checking Sequential Circuits // Proc. of 2001 IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems (DFT 2001). — USA, CA, San Francisco, October 24-26, 2001. — P. 395—402.
14. *Блюдов А.А., Ефанов Д.В., Сапожников В.В., Сапожников Вл.В.* О кодах с суммированием единичных разрядов в системах функционального контроля // Автоматика и телемеханика. — 2014. — № 8. — С. 131—145.
15. *Сапожников В.В., Сапожников Вл.В., Дмитриев А.В. и др.* Организация функционального контроля комбинационных схем методом логического дополнения // Электрон. моделирование. — 2002. — 24, № 6. — С. 52—66.
16. *Пархоменко П.П., Согомонян Е.С.* Основы технической диагностики (оптимизация алгоритмов диагностирования, аппаратурные средства). — М.: Энергоатомиздат, 1981. — 320 с.
17. *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.
18. *Гессель М., Морозов А.В., Сапожников В.В., Сапожников Вл.В.* Контроль комбинационных схем методом логического дополнения // Там же. — 2005. — № 8. — С. 161—172.
19. *Berger J.M.* A Note on Error Detecting Codes for Asymmetric Channels // Information and Control. — 1961. — Vol. 4, Issue 1. — P. 68—73. — DOI: 10.1016/S0019-9958(61)80037-5.
20. *Bose B., Lin D.J.* Systematic Unidirectional Error-Detection Codes // IEEE Transactions on Computers. — 1985. — Vol. C-34. — P. 1026—1032.
21. *Сапожников В.В., Сапожников Вл.В.* Самопроверяемые тестеры для равновесных кодов // Автоматика и телемеханика. — 1992. — № 3. — С. 3—35.
22. *Goessel M., Saposhnikov Vl., Saposhnikov V., Dmitriev A.* A New Method for Concurrent Checking by Use of a 1-out-of-4 Code // Proc. of the 6th IEEE International On-line Testing Workshop. — Spain, Palma de Mallorca, 3-5 July, 2000. — P. 147—152.

23. *Morozov A., Saposhnikov V.V., Saposhnikov V.I., Goessel M.* New Self-Checking Circuits by Use of Berger-codes // *Ibid.* — Spain, Palma De Mallorca, 3-5 July, 2000. — P. 171—176.
24. *Saposhnikov V.V., Saposhnikov V.I., Morozov A. et al.* Design of Totally Self-Checking Combinational Circuits by Use of Complementary Circuits // *Proc. of East-West Design & Test Workshop.* — Ukraine, Yalta, 2004. — P. 83—87.
25. *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.
26. *Sen S.K.* A Self-Checking Circuit for Concurrent Checking by 1-out-of-4 code with Design Optimization Using Constraint Don't Cares // *National Conf. on Emerging trends and advances in Electrical Engineering and Renewable Energy (NCEEERE 2010).*— Sikkim Manipal Institute of Technology, Sikkim, 22-24 December, 2010.
27. *Das D.K., Roy S.S., Dmitiriev A. et al.* Constraint Don't Cares for Optimizing Designs for Concurrent Checking by 1-out-of-3 Codes // *Proc. of the 10th International Workshops on Boolean Problems.* — Germany, Freiberg, September, 2012. — P. 33—40.
28. *Сапожников В.В., Сапожников В.В., Ефанов Д.В.* Метод функционального контроля комбинационных логических устройств на основе кода «2 из 4» // *Известия вузов. Приборостроение.* — 2016. — **59**, № 7. — С. 524—533. — DOI: 10.17586/0021-3454-2016-59-7-524-533.
29. *Efanov D., Sapozhnikov V., Sapozhnikov V.I.* Methods of Organization of Totally Self-Checking Concurrent Error Detection System on the Basis of Constant-Weight «1-out-of-3»-Code // *Proc. of 14th IEEE East-West Design & Test Symposium (EWDTS' 2016).* — Armenia, Yerevan, 14-17 October, 2016. — P. 117—125.
30. *Sapozhnikov V., Sapozhnikov V.I., Efanov D.* Concurrent Error Detection of Combinational Circuits by the Method of Boolean Complement on the Base of «2-out-of-4» Code // *Ibid.*— Armenia, Yerevan, 14-17 October, 2016. — P. 126—133.
31. *Сапожников В.В., Сапожников В.В., Ефанов Д.В.* Построение полностью самопроверяемых структур систем функционального контроля с использованием равновесного кода «1 из 3» // *Электрон. моделирование.* — 2016. — **38**, № 6. — С. 25—43.
32. *Сапожников В.В., Сапожников В.В., Ефанов Д.В.* Построение полностью самопроверяемых структур систем функционального контроля на основе равновесного кода «2 из 4» // *Проблемы управления.* — 2017. — № 1. — С. 57—64.
33. *Сапожников В.В., Сапожников В.В.* Самопроверяемые дискретные устройства. — СПб: Энергоатомиздат, 1992. — 224 с.
34. *Carter W.C., Duke K.A., Schneider P.R.* Self-Checking Error Checker for Two-Rail Coded Data // *United States Patent Office*, filed July 25, 1968, ser. No. 747,533, patented Jan. 26, 1971.— N. Y.— 10 p.
35. *Huches J.L.A., McCluskey E.J., Lu D.J.* Design of Totally Self-Checking Comparators with an Arbitrary Number of Inputs // *IEEE Transactions on Computers.* — 1984. — Vol. C-33, No. 6. — P. 546—550.
36. *Сапожников В.В., Рабара В.* Универсальный алгоритм синтеза $1/n$ -тестеров // *Проблемы передачи информации.* — 1982. — **18**, № 3. — С. 62—73.
37. *Аксенова Г.П.* Необходимые и достаточные условия построения полностью проверяемых схем свертки по модулю 2 // *Автоматика и телемеханика.* — 1979. — № 9. — С. 126—135.
38. *Аксенова Г.П.* О функциональном диагностировании дискретных устройств в условиях работы с неточными данными // *Проблемы управления.* — 2008. — № 5. — С. 62—66.
39. *Collection of Digital Design Benchmarks* [Режим доступа: <http://ddd.fit.cvut.cz/prj/Benchmarks/>].

40. 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 // Electronics Research Laboratory, Department of Electrical Engineering and Computer Science. — University of California, Berkeley, 4 May, 1992. — 45 p.

Поступила 23.01.17

REFERENCES

1. Sogomonyan, E.S. and Slabakov, E.V. (1989), *Samoproveryaemye ustroystva i otkazoustoichivyye sistemy* [Self-checking devices and fault-tolerant systems], Radio i svyaz, Moscow, USSR.
2. Drozd, A.V., Kharchenko, V.S., Antoshchuk, S.G. and et al. (2012), *Rabocheye diagnostirovanie bezopasnykh informatsionno-upravlyayushchikh sistem* [On-line testing for safe information-control systems], N.E. Zhukovsky Natsionalny Aerokosmichesky Universitet «KhAI». Kharkov, Ukraine.
3. Kharchenko, V., Kondratenko, Yu., and Kacprzyk, J. (2017), Green IT engineering: concepts, models, complex systems architectures, Springer Book series: Studies in Systems, Decision and Control, DOI: 10.1007/978-3-319-44162-7.
4. Mitra, S. and McCluskey, E.J. (2000), “Which concurrent error detection scheme to choose?”, *Proceedings of International Test Conference 2000*, Atlantic City, NJ, USA, October 03-05, 2000, pp. 985-994.
5. Slabakov, E.V. and Sogomonyan, E.S. (1980), “Formation of totally self-checking combinational devices with the use of constant-weight codes”, *Avtomatika i telemekhanika*, no. 9, pp. 173-181.
6. Slabakov, E.V. and Sogomonyan, E.S. (1981), “Self-checking computing devices and systems” (Review), *Avtomatika i telemekhanika*, no. 11, pp. 147-167.
7. Piestrak, S.J. (1995), Design of self-testing checkers for unidirectional error detecting codes, Oficyna Wydawnicza Politechniki Wrociavskiej, Wrociaw, Poland.
8. Das, D. and Touba, N.A. (1999), “Synthesis of circuits with low-cost concurrent error detection based on Bose-Lin codes”, *Journal of Electronic Testing: Theory and Applications*, Vol. 15, iss. 1-2, pp. 145-155, DOI: 10.1023/A:1008344603814.
9. Nicolaidis, M., and Zorian, Y. (1998), “On-line testing for VLSI - a compendium of approaches”, *Journal of Electronic Testing: Theory and Applications*, no. 12, pp. 7-20.
10. Das, D. and Touba, N.A. (1999), “Weight-based codes and their application to concurrent error detection of multilevel circuits”, *Proceedings of 17th IEEE Test Symposium*, California, USA, pp. 370-376.
11. Matrosova, A.Yu., Levin, I. and Ostanin, S.A. (2000), “Self-checking synchronous FSM network design with low overhead”, *VLSI Design*, Vol. 11, iss. 1, pp. 47-58, DOI:10.1155/2000/46578.
12. Das, D., Touba, N.A., Seuring, M. and Gossel, M. (2000), “Low cost concurrent error detection based on modulo weight-based codes”, *Proceedings of IEEE 6th International On-Line Testing Workshop (IOLTW)*, Palma de Mallorca, Spain, July 3-5, 2000, pp. 171-176.
13. Matrosova, A., Levin, I. and Ostanin, S. (2001), “Survivable Self-Checking Sequential Circuits”, *Proceedings of 2001 IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems (DFT 2001)*, CA, San Francisco, USA, October 24-26, 2001, pp. 395-402.
14. Blyudov, A.A., Efanov, D.V., Sapozhnikov, V.V. and Sapozhnikov, V.I. (2014), “On codes with summation of data bits in concurrent error detection systems”, *Avtomatika i telemekhanika*, no. 8, pp. 131-145.
15. Sapozhnikov, V.V. and et al. (2002), “Organization of functional checking of combinational circuits by the logic complement method”, *Elektronnoe modelirovanie*, Vol. 24, no. 6, pp. 52-66.

16. Parkhomenko, P.P. and Sogomonyan, E.S. (1981), *Osnovy tekhnicheskoy diagnostiki (optimizatsiya algoritmov diagnostirovaniya, apparaturnyye sredstva)* [Basics of technical diagnostics (optimization of diagnostic algorithms and equipment)], Energoatomizdat, Moscow, Russia.
17. Goessel, M., Morozov, A.V., Sapozhnikov, V.V. and Sapozhnikov, VI.V. (2003), “Logic complement, a new method of checking the combinational circuits”, *Avtomatika i telemekhanika*, no. 1, pp. 167-176.
18. Goessel, M., Morozov, A.V., Sapozhnikov, V.V. and Sapozhnikov, VI.V. (2005), “Checking combinational circuits by the method of logic complement”, *Avtomatika i telemekhanika*, no. 8, pp. 161-172.
19. Berger, J.M. (1961), “A note on error detecting codes for asymmetric channels”, *Information and Control*, Vol. 4, iss. 1, pp. 68-73, DOI: 10.1016/S0019-9958(61)80037-5.
20. Bose, B. and Lin, D.J. (1985), “Systematic unidirectional error-detection codes”, *IEEE Transactions on Computers*, vol. C-34, pp. 1026-1032.
21. Sapozhnikov, V.V. and Sapozhnikov, VI.V. (1992), “Self-checking checkers for constant-weight codes”, *Avtomatika i telemekhanika*, no. 3, pp. 3-35.
22. Goessel, M., Sapozhnikov, VI., Sapozhnikov, V. and Dmitriev, A. (2000), “A new method for concurrent checking by use of a 1-out-of-4 code”, *Proceedings of the 6th IEEE International On-line Testing Workshop*, Palma de Mallorca, Spain, July 3-5, 2000, pp. 147-152.
23. Morozov, A., Sapozhnikov, V.V., Sapozhnikov, VI.V. and Goessel, M. (2000), “New self-checking circuits by use of Berger-codes”, *Proceedings of the 6th IEEE International On-line Testing Workshop*, Palma de Mallorca, Spain, July 3-5, 2000, pp. 171-176.
24. Sapozhnikov, V.V. and et al. (2004), “Design of totally self-checking combinational circuits by use of complementary circuits”, *Proceedings of East-West Design and Test Workshop*, Yalta, Ukraine, 2004, pp. 83-87.
25. Göessel, M., Ocheretny, V., Sogomonyan, E. and Marienfeld, D. (2008), *New methods of concurrent checking: Edition 1*, Springer Science+Business Media B.V., Dodrecht, Germany.
26. Sen, S.K. (2010), “A self-checking circuit for concurrent checking by 1-out-of-4 code with design optimization using constraint don't cares”, *Proceedings of National Conf. on Emerging Trends and Advances in Electrical Engineering and Renewable Energy* (NCEEERE 2010), December 22-24, 2010, Manipal Institute of Technology, Sikkim, India.
27. Das, D.K., Roy, S.S., Dmitriev, A., Morozov, A. and Güssel, 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*, September, 2012, Freiberg, Germany, pp. 33-40.
28. Sapozhnikov, V.V., Sapozhnikov, VI.V. and Efanov, D.V. (2016), “Method of logical devices concurrent error detection system based on “2-out-of-4” code”, *Izvestiya vuzov. Priboroostroenie*, Vol. 59, no. 7, pp. 524-533, DOI 10.17586/0021-3454-2016-59-7-524-533.
29. Efanov, D., Sapozhnikov, V. and Sapozhnikov, VI. (2016), “Methods of organization of totally self-checking concurrent error detection system on the basis of constant-weight «1-out-of-3»-code”, *Proceedings of 14th IEEE East-West Design and Test Symposium (EWDTS'2016)*, October 14-17, 2016, Armenia, Yerevan, pp. 117-125, DOI: 10.1109/EWDTS.2016.7807622.
30. Sapozhnikov, V., Sapozhnikov, VI. and Efanov, D. (2016), “Concurrent error detection of combinational circuits by the method of Boolean complement on the base of «2-out-of-4» code”, *Proceedings of 14th IEEE East-West Design and Test Symposium (EWDTS'2016)*, October 14-17, 2016, Yerevan, Armenia, pp. 126-133, DOI: 10.1109/EWDTS.2016.7807677.
31. Sapozhnikov, V.V., Sapozhnikov, VI.V. and Efanov, D.V. (2016), “Formation of totally self-checking structures of concurrent error detection systems with use of constant-weight code “1-out-of-3””, *Elektronnoe modelirovanie*, Vol. 38, no. 6, pp. 25-43.

32. Sapozhnikov, V.V., Sapozhnikov, Vl.V. and Efanov, D.V. (2017), "Design of self-checking concurrent error detection systems based on «2-out-of-4» constant-weight code", *Problemy upravleniya*, iss. 1, pp. 57-64.
33. Sapozhnikov, V.V. and Sapozhnikov, Vl.V. (1992), *Samoproveryaemye diskretnye ustroystva* [Self-checking discrete devices], Energoatomizdat, St. Petersburg, Russia.
34. Carter, W.C., Duke, K.A. and Schneider, P.R. (1971), Self-checking error checker for two-rail coded data, United States Patent Office, filed July 25, 1968, ser. No. 747, 533, patented Jan. 26, N. Y., USA.
35. Huches, J.L.A., McCluskey, E.J. and Lu, D.J. (1984), "Design of totally self-checking comparators with an arbitrary number of inputs", *IEEE Transactions on Computers*, Vol. C-33, no. 6, pp. 546-550.
36. Sapozhnikov, V.V. and Rabara, B. (1982), "Universal algorithm of 1/n-checkers synthesis", *Problemy peredachi informatsii*, Vol. 18, iss. 3, pp. 62-73.
37. Aksyonova, G.P. (1979), "Necessary and sufficient conditions for the design of totally checking circuits of compression by modulo 2", *Avtomatika i telemekhanika*, no. 9, pp. 126-135.
38. Aksyonova, G.P. (2008), "On functional diagnosis of discrete devices under imperfect data processing conditions", *Problemy upravleniya*, iss. 5, pp. 62-66.
39. Collection of Digital Design Benchmarks [URL: <http://ddd.fit.cvut.cz/prj/Benchmarks/>].
40. Sentovich, E.M., Singh, K.J., Lavagno, L., Moon, C., Murgai, R., Saldanha, A., Savoj, H., Stephan, P.R., Brayton, R.K. and Sangiovanni-Vincentelli, A. (1992), SIS: A system for sequential circuit synthesis, Electronics Research Laboratory, Department of Electrical Engineering and Computer Science, University of California, Berkeley, USA.

Received 23.01.17

V.V. Sapozhnikov, Vl.V. Sapozhnikov, D.V. Efanov, D.V. Pivovarov

BOOLEAN COMPLEMENT METHOD BASED ON CONSTANT-WEIGHT CODE "1-OUT-OF-4" FOR FORMATION OF TOTALLY SELF-CHECKING CONCURRENT ERROR DETECTION SYSTEMS

Authors offer the way of formalization of Boolean complement functions calculation rules in concurrent error detection systems based on constant-weight code "1-out-of-4". Herewith the procedure of complement function values selection is excluded and the property of totally self-checking is provided – all the XOR gates in Boolean complement module and checker are checked by guarantee. The number of ways of completion of Boolean complement functions of "1-out-of-4" code with complement of three operational functions only is established as well as the minimum number of operational vectors needed for totally self-checking provision.

Keywords: concurrent error detection system, Boolean complement, constant-weight code, "1-out-of-4" code, totally self-checking structure.

САПОЖНИКОВ Валерий Владимирович, д-р техн. наук, профессор кафедры «Автоматика и телемеханика на железных дорогах» Петербургского государственного университета путей сообщения Императора Александра I. В 1963 г. окончил Ленинградский ин-т инженеров железнодорожного транспорта. Область научных исследований — надежность синтез дискретных устройств, синтез безопасных систем, синтез самопроверяемых схем, техническая диагностика дискретных систем.

САПОЖНИКОВ Владимир Владимирович, д-р техн. наук, профессор кафедры «Автоматика и телемеханика на железных дорогах» Петербургского государственного университета путей сообщения Императора Александра I. В 1963 г. окончил Ленинградский ин-т инженеров железнодорожного транспорта. Область научных исследований — надежность и синтез дискретных устройств, синтез безопасных систем, синтез самопроверяемых схем, техническая диагностика дискретных систем.

ЕФАНОВ Дмитрий Викторович, канд. техн. наук, доцент кафедры «Автоматика и телемеханика на железных дорогах» Петербургского государственного университета путей сообщения Императора Александра I. В 2007 г. окончил Петербургский государственный университет путей сообщения. Область научных исследований — дискретная математика, надежность и техническая диагностика дискретных систем.

ПИВОВАРОВ Дмитрий Вячеславович, аспирант кафедры «Автоматика и телемеханика на железных дорогах» Петербургского государственного университета путей сообщения Императора Александра I, который окончил в 2016 г. Область научных исследований — техническая диагностика дискретных систем, математическое моделирование.

