

Логическое дополнение до модульных кодов с суммированием для синтеза схем встроенного контроля комбинационных устройств автоматики и вычислительной техники

Д. В. Ефанов, М. В. Зуева

Российский университет транспорта, г. Москва

TrES-4b@yandex.ru, marina-seo-media@yandex.ru

Аннотация — Обсуждаются особенности синтеза схем встроенного контроля по методу логического дополнения до модульных кодов с суммированием. Показано, что необнаруживаемые ошибки в описываемой структуре, в отличие от традиционной структуры организации схемы встроенного контроля, могут возникать не только в информационном векторе, но и одновременно в информационном и в контрольном векторах. Ошибки же в контрольных векторах обнаруживаются всегда. Определены характеристики обнаружения ошибок в модульных кодах с суммированием со значениями модулей $M=2, 4, 8$. Установлено, что для кодов со значениями модуля $M=4$ и 8 число необнаруживаемых ошибок, связанных с искажениями и информационных и контрольных разрядов существенно больше, чем число необнаруживаемых ошибок, возникающих только в информационных векторах. Предложен схемотехнический способ исключения одновременно возникающих ошибок в информационном и контрольном векторах модульных кодов с суммированием. Приведены результаты экспериментов по обнаружению ошибок на выходах контрольных комбинационных схем с помощью схемы встроенного контроля, синтезированной по методу логического дополнения до модульных кодов с суммированием.

Ключевые слова — схема встроенного контроля; кодовые методы синтеза схем контроля; модульный код с суммированием; обнаружение ошибок на выходах устройств; синтез схемы встроенного контроля; самопроверяемое устройство.

I. ВВЕДЕНИЕ

Кодовые методы широко применяются для организации контроля вычислений в современных микроэлектронных и микропроцессорных системах автоматики [1 – 6]. Их применяют и при синтезе самопроверяемых вычислительных устройств, снабжаемых схемами встроенного контроля (СВК) [7 – 12].

Наиболее распространенной является структура организации СВК, приведенная на рис. 1 [13 – 15]. Объектом диагностирования является комбинационное устройство $F(x)$, формирующее булевы функции $f_1, f_2, \dots, f_{m-1}, f_m$. Для организации косвенного контроля возникающих в нем неисправностей по результатам

вычислений (рабочего диагностирования) используется СВК. В СВК рабочие выходы $f_1, f_2, \dots, f_{m-1}, f_m$ дополняются контрольными выходами $g_1, g_2, \dots, g_{k-1}, g_k$, значения которых формируются с использованием специального блока контрольной логики $G(x)$. На этапе проектирования системы диагностирования между рабочими и контрольными выходами устанавливается однозначное соответствие. Как правило, рабочие выходы отождествляются с информационными разрядами, а контрольные – с контрольными разрядами заранее выбранного равномерного блочного кода. Контроль соответствия разрядов в процессе эксплуатации системы диагностирования производится с помощью специального устройства – тестера TSC . Тестер реализуется в виде самопроверяемого комбинационного устройства и снабжается двумя выходами z^0 и z^1 , функционирующими в парафазной логике. Нарушения в работе самого устройства $F(x)$ и элементов СВК должны приводить к нарушению парафазности сигнала на выходах z^0 и z^1 тестера, что является сигналом ошибки в вычислениях.

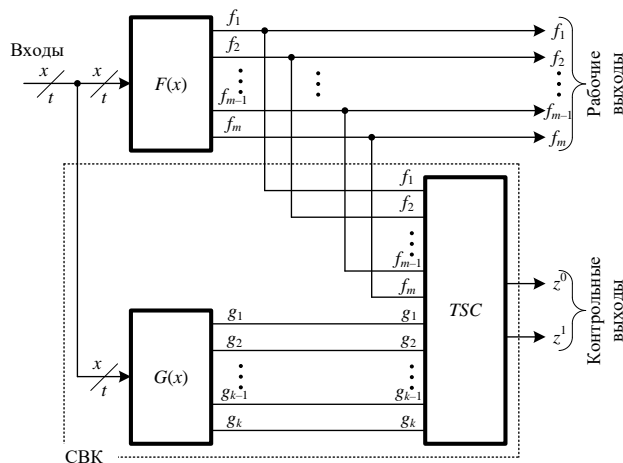


Рис. 1. Традиционная структура организации СВК

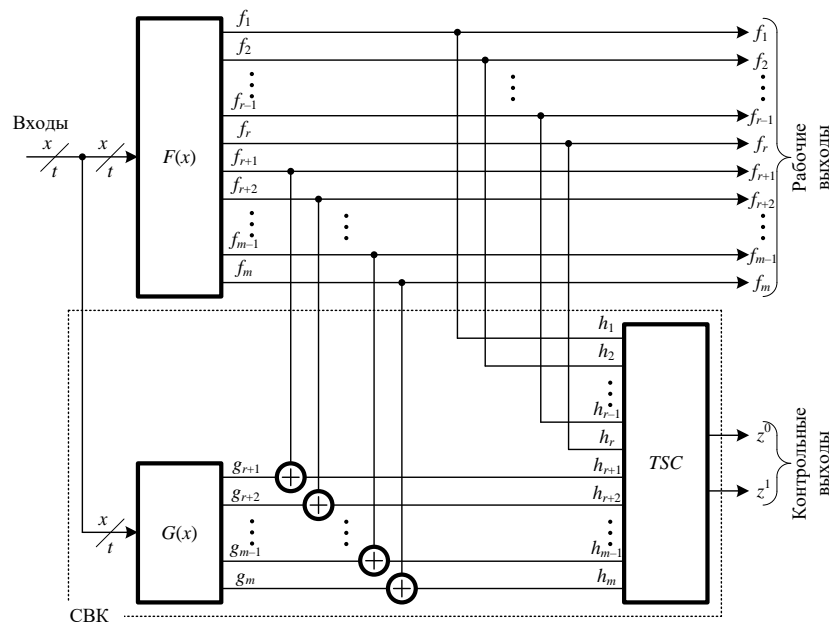


Рис. 2. Структура организации СВК на основе логического дополнения

На начальном этапе синтеза структуры рис. 1 для конкретного устройства $F(x)$ определяются модели неисправностей, относительно которых она будет самопроверяемой. Модель неисправностей позволяет покрыть с некоторой полнотой множество реальных дефектов. Например, модель константных неисправностей (stuck-at faults) покрывает до 80-95 % реальных дефектов в комбинационных устройствах в зависимости от технологии их изготовления [16]. Известны и другие модели [17 – 19].

Структура, изображенная на рис. 1, обычно сравнивается с классической структурой дублирования [1, 2] по показателям аппаратной избыточности и покрытию ошибок, возникающих на выходах объекта диагностирования в результате возникновения неисправности. Известно [7, 11, 12], что применение кодовых методов позволяет значительно снизить аппаратную избыточность системы диагностирования по сравнению с дублированием. Кроме того, при оговоренной модели неисправностей возможен синтез такой СВК, в которой обнаруживаются любые комбинации ошибок на выходах устройства $F(x)$.

Недостатком описанной структуры является то, что для выбранного равномерного блочного кода она может быть построена только в единственной реализации (исключая, конечно, разнообразные способы синтеза тестеров). В ряде случаев это приводит к невозможности обеспечения условий формирования тестовых комбинаций на входах тестера, что делает невозможным учет любых неисправностей из заданной модели в нем и, соответственно, приводит к нарушению самопроверяемости всей системы диагностирования.

Большой гибкостью в плане синтеза СВК обладает метод логического дополнения [20, 21]. Он позволяет строить СВК иначе. Фактически все или часть рабочих

функций блока $F(x)$ не дополняются, а преобразуются в СВК в разряды, отождествляемые с разрядами кодового слова заранее выбранного равномерного блочного кода (рис. 2). Для этих целей в СВК используется блок корректирующих элементов – элементов сложения по модулю $M=2$ (XOR). Все (или часть) рабочих выходов подключаются к входам двухвходовых элементов XOR, а на вторые входы этих элементов подаются сигналы коррекции от блока $G(x)$.

На рис. 2 часть рабочих функций не преобразуется и отождествляется с информационным вектором длиной r . Функции же $f_{r+1}, f_{r+2}, \dots, f_{m-1}, f_m$ корректируются по правилу: $f_i \oplus g_i = h_i$, $i = \overline{r+1, m}$. Функции же $f_1, f_2, \dots, f_{r-1}, f_r$ приравниваются к функциям, описывающим информационные разряды выбранного кода: $f_i = h_i$, $i = \overline{1, r}$.

Выше описано применение именно разделимых кодов для организации СВК по методу логического дополнения. Однако наиболее часто в литературе рассматривалось применение для этих целей неразделимых кодов [22 – 26]. Это связано с тем, что неисправность, возникающая в блоке $F(x)$, может приводить к искажению любого числа разрядов в кодовом слове, поступающем на вход тестера. При использовании разделимого кода возможно искажение как в информационном, так и в контрольном векторе. Такая ошибка может оказаться необнаруженной, а вероятность ее возникновения выше, чем вероятность возникновения ошибки только в информационном или только в контрольном векторе. В работе [27], например, нами показано, что при использовании классических кодов Бергера [28] в структуре,

приведенной на рис. 2, число одновременных искажений и информационных и контрольных векторов в разы превышает число искажений в информационном векторе. Ошибки, возникающие в контрольных разрядах в СВК, обнаруживаются всегда, так как нарушают соответствие между информационными и контрольными векторами. Ошибки, возникающие в информационных и контрольных векторах одновременно, а также ошибки, возникающие только в информационных векторах, могут оказаться не обнаруженными. Известны методы синтеза СВК с полным обнаружением ошибок в информационных векторах [7, 11, 12, 23]. Для синтеза же СВК с обнаружением ошибок и в информационном и в контрольном векторах одновременно требуется учитывать особенности их возникновения.

Целью данной работы является исследование возможностей использования модульных кодов с суммированием (кодов с суммированием единичных разрядов в кольце вычетов по заданному модулю M) при организации СВК по методу логического дополнения.

II. МОДУЛЬНЫЕ КОДЫ С СУММИРОВАНИЕМ И ОСОБЕННОСТИ ИХ ПРИМЕНЕНИЯ ПРИ СИНТЕЗЕ СВК ПО МЕТОДУ ЛОГИЧЕСКОГО ДОПОЛНЕНИЯ

Модульный код с суммированием строится следующим образом [11]. Число единичных разрядов в информационном векторе подсчитывается в кольце вычетов по заданному модулю M . Обозначим модульный код как $SM(r, m-r)$ -код, где M – значение модуля, r – число информационных разрядов, а $m-r$ – число контрольных разрядов.

Отметим, что в качестве модуля при построении $SM(r, m-r)$ -кода может выступать любое число от 2 до $m+1$, однако наиболее эффективными для синтеза самопроверяемых СВК являются модули $M \in \{2^1, 2^2, \dots, 2^{\lceil \log_2(m+1) \rceil - 2}, 2^{\lceil \log_2(m+1) \rceil - 1}\}$ [11]. В этом случае для модульного кода формируется полное множество контрольных векторов, что облегчает процедуру обеспечения самопроверяемости тестеров данных кодов.

С использованием разработанного программного модуля авторы статьи определили для трех основных $SM(r, m-r)$ -кодов со значением модуля $M=2, 4, 8$ характеристики обнаружения ошибок в кодовых словах. Рассчитанные данные для кодов с числом информационных разрядов $r=4\dots 16$ представлены в таблицах 1 – 3. Общее количество ошибок, возникающих в информационных векторах кодовых слов, в каждой таблице занесено в столбец N_r , а общее количество ошибок, возникающих во всех разрядах кодовых слов, – в столбец N_m . Необнаруживаемыми являются только те ошибки, которые приводят к трансформации искажаемого кодового слова в кодовое слово, принадлежащее $SM(r, m-r)$ -коду. Число таких о занесено в таблицах в столбец $N_{r,k}$. Следует отметить, что оно для каждого r совпадает с числом N_r . Необнаруживаемые ошибки складываются из ошибок, возникающих только в информационном векторе (они занесены в столбец N_D каждой таблицы), и из ошибок, возникающих и в информационном и в контрольном векторах (они занесены в столбец N_{DC} каждой таблицы). В последнем столбце каждой таблицы рассчитан показатель χ , характеризующий долю величины N_{DC} от величины N_D . Для различных $SM(r, m-r)$ -кодов показатель χ изменяется по-разному с увеличением числа r (рис. 3). Для $S2(r, 1)$ -кода с увеличением r его значение постепенно уменьшается и стремится к величине $\chi=1$, что говорит о примерном равенстве числа необнаруживаемых искажений только в информационном векторе и в информационном и контрольном векторе. Для $S4(r, 2)$ -кода с увеличением r значение величины χ , наоборот, уменьшается и стремится к величине $\chi=3$. Другими словами, число одновременно возникающих необнаруживаемых ошибок и в информационных и в контрольных векторах $S4(r, 2)$ -кода существенно больше числа необнаруживаемых ошибок, возникающих только в информационном векторе. Для $S8(r, 3)$ -кода с увеличением r значение величины χ растет и в рассмотренном диапазоне длин информационных векторов изменяется от величины $\chi=3,444$ до величины $\chi=5,904$.

Таблица 1

Характеристика необнаруживаемых ошибок в кодовых словах $S2(r, 1)$ -кода

r	$m-r$	m	N_r	N_m	N_D	N_{DC}	$N_{r,k}$	χ
4	1	5	240	992	112	128	240	1,143
5	1	6	992	4032	480	512	992	1,067
6	1	7	4032	16256	1984	2048	4032	1,032
7	1	8	16256	65280	8064	8192	16256	1,016
8	1	9	65280	261632	32512	32768	65280	1,008
9	1	10	261632	1047552	130560	131072	261632	1,004
10	1	11	1047552	4192256	523264	524288	1047552	1,002
11	1	12	4192256	16773120	2095104	2097152	4192256	1,001
12	1	13	16773120	67100672	8384512	8388608	16773120	1
13	1	14	67100672	268419072	33546240	33554432	67100672	1
14	1	15	268419072	1073709056	134201344	134217728	268419072	1
15	1	16	1073709056	4294901760	536838144	536870912	1073709056	1
16	1	17	4294901760	17179738112	2147418112	2147483648	4294901760	1

Таблица 2

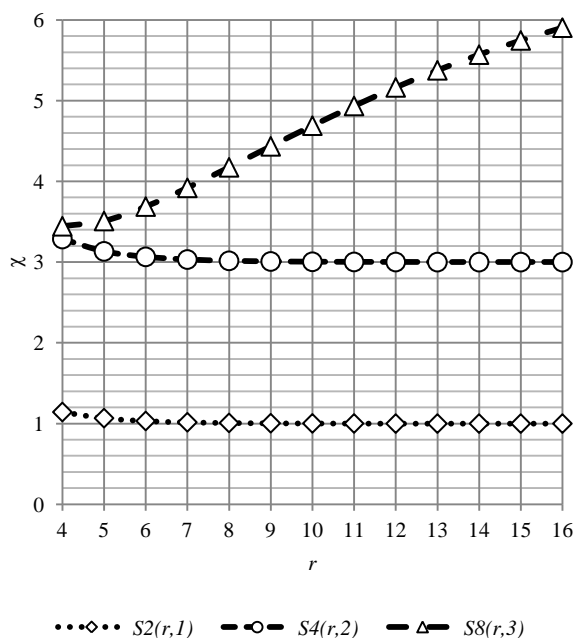
Характеристика необнаруживаемых ошибок в кодовых словах $S4(r,2)$ -кода

r	$m-r$	m	N_r	N_m	N_D	N_{DC}	$N_{r,k}$	χ
4	2	6	240	4032	56	184	240	3,286
5	2	7	992	16256	240	752	992	3,133
6	2	8	4032	65280	992	3040	4032	3,065
7	2	9	16256	261632	4032	12224	16256	3,032
8	2	10	65280	1047552	16256	49024	65280	3,016
9	2	11	261632	4192256	65280	196352	261632	3,008
10	2	12	1047552	16773120	261632	785920	1047552	3,004
11	2	13	4192256	67100672	1047552	3144704	4192256	3,002
12	2	14	16773120	268419072	4192256	12580864	16773120	3,001
13	2	15	67100672	1073709056	16773120	50327552	67100672	3
14	2	16	268419072	4294901760	67100672	201318400	268419072	3
15	2	17	1073709056	17179738112	268419072	805289984	1073709056	3
16	2	18	4294901760	68719214592	1073709056	3221192704	4294901760	3

Таблица 3

Характеристика необнаруживаемых ошибок в кодовых словах $S8(r,3)$ -кода

r	$m-r$	m	N_r	N_m	N_D	N_{DC}	$N_{r,k}$	χ
4	3	7	240	16256	54	186	240	3,444
5	3	8	992	65280	220	772	992	3,509
6	3	9	4032	261632	860	3172	4032	3,688
7	3	10	16256	1047552	3304	12952	16256	3,92
8	3	11	65280	4192256	12616	52664	65280	4,174
9	3	12	261632	16773120	48144	213488	261632	4,434
10	3	13	1047552	67100672	184112	863440	1047552	4,69
11	3	14	4192256	268419072	706464	3485792	4192256	4,934
12	3	15	16773120	1073709056	2721312	14051808	16773120	5,164
13	3	16	67100672	4294901760	10523968	56576704	67100672	5,376
14	3	17	268419072	17179738112	40853696	227565376	268419072	5,57
15	3	18	1073709056	68719214592	159156352	914552704	1073709056	5,746
16	3	19	4294901760	274877382656	622051456	3672850304	4294901760	5,904

Рис. 3. Зависимость χ от r

Практическим выводом из таблиц 1 – 3 и рис. 3 следует то, что при организации СВК по методу

логического дополнения до $SM(r, m-r)$ -кодов целесообразно выделять группы преобразуемых и не преобразуемых выходов таким образом, чтобы одновременные искажения в информационном и контрольном векторах исключались. Это может быть сделано схемотехническим образом путем анализа структуры объекта диагностирования $F(x)$.

На множестве выходов устройства $F(x)$ определяются группы независимых друг от друга выходов.

Определение 1. Группа выходов $(f_{i_1}, f_{i_2}, \dots, f_{i_r})$, $\{i_1, i_2, \dots, i_r\} \in \{1, 2, \dots, m\}$, структурно независима от группы выходов $(f_{j_1}, f_{j_2}, \dots, f_{j_p})$, $\{j_1, j_2, \dots, j_p\} \in \{1, 2, \dots, m\} \setminus \{i_1, i_2, \dots, i_r\}$, если в структуре цифрового устройства не существует такого элемента, пути от которого ведут одновременно хотя бы к одному из выходов группы $(f_{i_1}, f_{i_2}, \dots, f_{i_r})$ и хотя бы к одному из выходов группы $(f_{j_1}, f_{j_2}, \dots, f_{j_p})$.

Если имеется группа выходов $(f_{i_1}, f_{i_2}, \dots, f_{i_q})$ структурно независимая от группы выходов

$(f_{j_1}, f_{j_2}, \dots, f_{j_p})$ то в качестве преобразуемых выходов могут быть выбраны выходы как первой, так и второй группы. Среди таких групп осуществляется поиск тех выходов, которые следует преобразовывать с учетом влияния на сложность технической реализации блока $G(x)$ и необходимости обеспечения контролепригодности тестера выбранного кода и элементов сложения по модулю два (XOR's).

Определение 2. Группа выходов $(f_{i_1}, f_{i_2}, \dots, f_{i_r})$, $\{i_1, i_2, \dots, i_r\} \in \{1, 2, \dots, m\}$, функционально независима от группы выходов $(f_{j_1}, f_{j_2}, \dots, f_{j_p})$, $\{j_1, j_2, \dots, j_p\} \in \{1, 2, \dots, m\} \setminus \{i_1, i_2, \dots, i_r\}$, если для любого элемента G_q с выходом y_q выполняется условие:

$$\left(\frac{\partial f_{i_1}}{\partial y_q} \vee \frac{\partial f_{i_2}}{\partial y_q} \vee \dots \vee \frac{\partial f_{i_r}}{\partial y_q} \right) \left(\frac{\partial f_{j_1}}{\partial y_q} \vee \frac{\partial f_{j_2}}{\partial y_q} \vee \dots \vee \frac{\partial f_{j_p}}{\partial y_q} \right) = 0.$$

Выражение в первых скобках определяет условия искажения хотя бы одного выхода из группы $(f_{i_1}, f_{i_2}, \dots, f_{i_r})$, $\{i_1, i_2, \dots, i_r\} \in \{1, 2, \dots, m\}$, а выражение во вторых скобках – условия искажения хотя бы одного выхода из группы $(f_{j_1}, f_{j_2}, \dots, f_{j_p})$, $\{j_1, j_2, \dots, j_p\} \in \{1, 2, \dots, m\} \setminus \{i_1, i_2, \dots, i_r\}$.

III. ЭКСПЕРИМЕНТАЛЬНЫЕ РЕЗУЛЬТАТЫ

В ходе исследований структуры, приведенной на рис. 2, были проведены эксперименты с контрольными комбинационными схемами из набора LGSynth'89 [29, 30]. Эксперименты состояли в оценке числа необнаруживаемых на выходах схем ошибок в СВК, организованной по $SM(r, m-r)$ -кодам по структуре рис. 2.

Схемы в LGSynth'89 представлены, в том числе, в формате *.netblif, описывающем их структуры в базисе И-НЕ (с использованием 2-, 3-х и 4-х входных элементов И-НЕ). Последовательно в схему вносились одиночные константные неисправности и подавались все входные комбинации. Затем фиксировалось, сколько из ошибок, транслируемых на выходы схемы, не обнаруживалось. Данные расчетов занесены в табл. 4 и проиллюстрированы на рис. 4 в виде столбчатой диаграммы, на которой показаны значения коэффициента ζ (в %), характеризующего долю необнаруживаемых на выходах схемы ошибок от общего их количества.

Как и ожидалось, наиболее эффективными являются модульные коды с наибольшим значением числа контрольных разрядов. Напомним, что k разрядов не преобразуются, а ошибки на них обнаруживаются в СВК. $S2(r, 1)$ -кодами для рассматриваемых схем не обнаруживалось менее 15% ошибок на выходах. $S4(r, 2)$ -кодами для

рассматриваемых схем не обнаруживалось менее 7% ошибок на выходах. $S8(r, 3)$ -кодами для рассматриваемых схем не обнаруживалось менее 3% ошибок на выходах.

Следует отметить, что для некоторых схем использование кода с меньшим числом контрольных разрядов никак не повлияло на характеристики обнаружения ошибок. Например, такой схемой является «cm42a» и для нее из двух кодов $S2(r, 1)$ и $S4(r, 2)$ можно выбрать первый, так как он дает схему с меньшим числом разрядов блока $G(x)$ и элементов XOR. Для схемы «f51m», к примеру, значение показателя ζ при использовании $S4(r, 2)$ и $S8(r, 3)$ кодов отличаются в сотых долях. Это также может характеризовать выбор кода с меньшим числом контрольных разрядов.

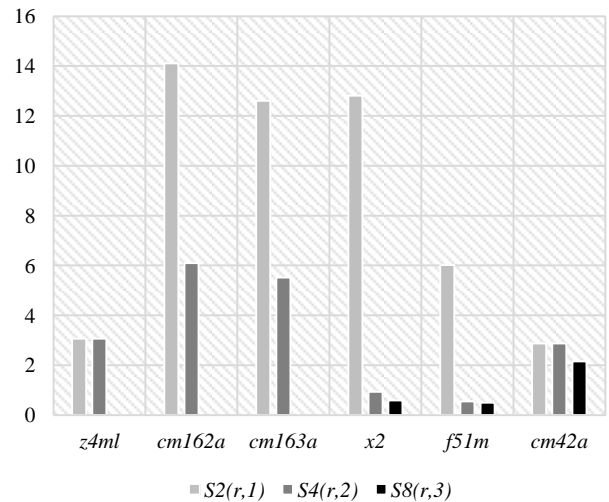


Рис. 4. Зависимость ζ от r

IV. ЗАКЛЮЧЕНИЕ

При синтезе СВК для комбинационных устройств на основе метода логического дополнения могут эффективно использоваться модульные коды с суммированием. При этом, однако, следует учитывать особенности самого метода, связанные с преобразованием всех или части рабочих функций объекта диагностирования в СВК с помощью схемы коррекции сигналов. Указанная особенность метода логического дополнения приводит к возможности возникновения ошибок как в информационных, так и в контрольных разрядах модульных кодов одновременно. Такие ошибки могут оказаться необнаруженными, если приведут к формированию кодового слова, принадлежащего выбранному модульному коду с суммированием. Исследования показали, что число таких ошибок гораздо больше числа необнаруживаемых ошибок, возникающих только в информационных векторах. Целесообразно на этапе проектирования СВК исключить схемотехническими методами возможность возникновения необнаруживаемых ошибок как в информационном, так и в контрольном векторах

одновременно. Это возможно за счет разделения выходов объекта диагностирования на группы функционально независимых выходов.

Таблица 4

Результаты экспериментов

Схема	Число выходов схемы	Всего ошибок по кратностям d					Всего	$\zeta, \%$
		1	2	3	4	5		
$S2(r,1)$								
z4ml	4	0	128	0	–	–	128	3,071
		4008	128	32			4168	
cm162a	5	0	37886	0	6877	0	44763	14,106
		248348	37886	23996	6877	224	317331	
cm163a	5	0	132096	0	21824	0	153920	12,603
		987648	132096	79680	21824	64	1221312	
x2	7	0	2428	0	96	0	2524	12,807
		16624	2428	544	96	16	19708	
f51m	8	0	781	0	2	–	783	6,019
		12169	781	56	2		13008	
cm42a	10	0	8	–	–	–	8	2,878
		270	8				278	
$S4(r,2)$								
z4ml	4	0	128	0	–	–	128	3,071
		4008	128	32			4168	
cm162a	5	0	2592	16517	0	224	19333	6,092
		248348	37886	23996	6877	224	317331	
cm163a	5	0	10240	57152	0	64	67456	5,523
		987648	132096	79680	21824	64	1221312	
x2	7	0	80	80	24	0	184	0,934
		16624	2428	544	96	16	19708	
f51m	8	0	71	0	1	–	72	0,554
		12169	781	56	2		13008	
cm42a	10	0	8	–	–	–	8	2,878
		270	8				278	
$S8(r,3)$								
x2	7	0	58	58	0	0	116	0,589
		16624	2428	544	96	16	19708	
f51m	8	0	65	0	0	–	65	0,5
		12169	781	56	2		13008	
cm42a	10	0	6	–	–	–	6	2,158
		270	8				278	

В заключение следует отметить, что, по нашему мнению, несмотря на обозначенные в статье особенности применения метода логического дополнения при синтезе СВК, он не заслуженно остается в стороне и может эффективно применяться на практике, значительно расширяя поле выбора способа обеспечения самопроверяемости комбинационных устройств автоматики и вычислительной техники.

ПОДДЕРЖКА

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

ЛИТЕРАТУРА

[1] Согомонян Е.С., Слабаков Е.В. Самопроверяемые устройства и отказоустойчивые системы. М.: Радио и связь, 1989, 208 с.

[2] Микони С.В. Общие диагностические базы знаний вычислительных систем. – СПб.: СПИИРАН, 1992, 234 с.
 [3] Abramovici M., Breuer M.A., Friedman A.D. Digital System Testing and Testable Design. – New Jersey: IEEE Press, 1998, 652 p.
 [4] Lala P.K. Self-Checking and Fault-Tolerant Digital Design. – San Francisco: Morgan Kaufmann Publishers, 2001, 216 p.
 [5] Fujiwara E. Code Design for Dependable Systems: Theory and Practical Applications, John Wiley & Sons, 2006, 720 p.
 [6] Дрозд А.В., Харченко В.С., Антошук 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: Национальный аэрокосмический университет им. Н.Е. Жуковского «ХАИ», 2012, 614 с.
 [7] Сапожников В.В., Сапожников Вл.В., Ефанов Д.В. Коды Хэмминга в системах функционального контроля логических устройств, СПб.: Наука, 2018, 151 с.
 [8] Тельпухов Д.В., Жукова Т.Д., Деменева А.И., Гуров С.И. Схема функционального контроля для

- комбинационных схем на основе R-кода // Проблемы разработки перспективных микро- и нанoeлектронных систем (МЭС). 2018. Вып.4. С. 98-104, doi: 10.31114/2078-7707-2018-4-98-104.
- [9] Telpuhov D.V., Zhukova T.D., Demeneva A.I., Gurov S.I. Automatic Synthesis of Fault-Tolerant CED Circuits Based on R-Code // Проблемы разработки перспективных микро- и нанoeлектронных систем (МЭС). 2019. Вып. 3. С.54-59. doi: 10.31114/2078-7707-2019-3-54-59.
- [10] Жукова Т.Д. Разработка системы автоматизированного проектирования СФК на основе методов избыточного кодирования // Проблемы разработки перспективных микро- и нанoeлектронных систем (МЭС). 2020. Вып.4. С. 119-125, doi: 10.31114/2078-7707-2020-4-51-57.
- [11] Сапожников В.В., Сапожников Вл.В., Ефанов Д.В. Коды с суммированием для систем технического диагностирования. Том 1: Классические коды Бергера и их модификации, М.: Наука, 2020, 383 с.
- [12] Сапожников В.В., Сапожников Вл.В., Ефанов Д.В. Коды с суммированием для систем технического диагностирования. Том 2: Взвешенные коды с суммированием, М.: Наука, 2021, 456 с.
- [13] Nicolaidis M., Zorian Y. On-Line Testing for VLSI – A Compendium of Approaches // Journal of Electronic Testing: Theory and Application, 1998, Vol. 12, Issue 1-2, Pp. 7-20, DOI: 10.1023/A:1008244815697.
- [14] 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, pp. 985-994, doi: 10.1109/TEST.2000.894311.
- [15] Гаврилов С.В., Гуров С.И., Жукова Т.Д., Рыжова Д.И. Применение теории кодирования для повышения помехозащищенности комбинационных схем // Информационные технологии, 2016, Т. 22, №12, С. 931-937.
- [16] Сапожников В.В., Сапожников Вл.В., Ефанов Д.В. Основы теории надежности и технической диагностики, Санкт-Петербург: Издательство «Лань», 2019, 588 с.
- [17] Багхдади А.А.А., Хаханов В.И., Литвинова Е.И. Методы анализа и диагностирования цифровых устройств (аналитический обзор) // Автоматизированные системы управления и приборы автоматки, 2014, №166, С. 59-74.
- [18] Матросова А.Ю., Андреева В.В., Тычинский В.З., Гошин Г.Г. Использование ROBDD-графов для тестирования задержек логических схем // Известия высших учебных заведений. Физика, 2019, Т. 62, № 5 (737), С. 86-94.
- [19] Матросова А.Ю., Тычинский В.З., Андреева В.В. Построение тестовых последовательностей для робастно тестируемых неисправностей задержек путей с низкой потребляемой мощностью с использованием SAT-решателей и ROBDD-графов // Проблемы разработки перспективных микро- и нанoeлектронных систем (МЭС). 2020. Вып.2. С. 43-49, doi: 10.31114/2078-7707-2020-2-43-49.
- [20] 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.
- [21] Гессель М., Морозов А.В., Сапожников В.В., Сапожников Вл.В. Контроль комбинационных схем методом логического дополнения // Автоматика и телемеханика, 2005, №8, С. 161-172.
- [22] Saposhnikov V.V., Saposhnikov V.I.V., Morozov A., Goessel M. Osadchy G. Design of totally self-checking combinational circuits by use of complementary circuits // Proceedings of 2th IEEE East-West Design & Test Symposium (EWDTS'2004), Crimea, Ukraine, September 15 – 17, 2004, pp. 83-87.
- [23] Gössel M., Ocheretny V., Sogomonyan E., Marienfeld D. New Methods of Concurrent Checking: Edition 1, Dordrecht: Springer Science+Business Media B.V., 2008, 184 p.
- [24] Sen S.K., Roy S.S. An Optimized Concurrent Self-Checker Using Constraint-Don't Cares and 1-out-of-4 Code // National Conference (AECDISC-2008) in Asansol Engineering College, held during 1-2 August 2008.
- [25] Das D.K., Roy S.S., Dmitriev 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, pp. 33-40.
- [26] Пивоваров Д.В. Построение систем функционального контроля многовыходных комбинационных схем методом логического дополнения по равновесным кодам // Автоматика на транспорте, 2018, Том 4, №1, С. 131-149.
- [27] Ефанов Д.В., Осадчий Г.В., Зуева М.В. Коды Бергера в схемах встроенного контроля, реализованных на основе метода логического дополнения // Информатика и системы управления, 2021, №1, С. 75-89, doi: 10.22250/isu.2021.67.75-89.
- [28] Berger J.M. A Note on Error Detection Codes for Asymmetric Channels // Information and Control, 1961, Vol. 4, Issue 1, Pp. 68-73, DOI: 10.1016/S0019-9958(61)80037-5.
- [29] Collection of Digital Design Benchmarks [Режим доступа: <http://ddd.fit.cvut.cz/prj/Benchmarks/>].
- [30] Sequential Circuit Design Using Synthesis and Optimization / E.M. Sentovich, K.J. Singh, C. Moon, H. Savoj, R.K. Brayton, A. Sangiovanni-Vincentelli // Proceedings IEEE International Conference on Computer Design: VLSI in Computers & Processors, 11-14 October 1992, Cambridge, MA, USA, USA pp. 328-333, doi: 10.1109/ICCD.1992.276282.

Boolean complement to modular sum codes for the concurrent error-detection systems synthesis for combinational devices of automation and computer technology

D. V. Efanov, M. V. Zueva

Russian University of Transport, Moscow

TrES-4b@yandex.ru, marina-seo-media@yandex.ru

Abstract — The article discusses the features of the concurrent error-detection (CED) systems synthesis by the method of Boolean complement to modular sum codes. The authors provide the structure of the CED systems organization by the method of Boolean complement to modular sum codes. This structure is described in detail. We show that undetectable errors in the described structure, in comparison with the traditional structure of the CED systems organization, can occur not only in the data vector, but also simultaneously in data and check vectors. Errors in check vectors are always detected. The article determines the characteristics of error detection in modular sum codes with modulus values $M=2, 4, 8$. It is found that for codes with modulus values $M=4$ and 8 the number of undetectable errors associated with distortions of both data and check bits is significantly higher than the number of undetectable errors occurring in data vectors only. It is shown that when considering the characteristics of the full codeword, the well-known property of modular codes for detecting any errors with even multiplicities in data vectors is not preserved. The authors proposed the circuit engineering method for eliminating simultaneously occurring errors in the data and check vectors of modular sum codes. This method makes it possible to significantly reduce the number of undetectable errors at the outputs of combinational circuits. The article provides the results of experiments on detecting errors at the outputs of the reference combinational circuits using the CED system synthesized by the method of Boolean complement to modular sum codes. The experiment did not distinguish the independent groups of outputs, which made it possible to evaluate the effectiveness of the proposed method directly, without analyzing the structure of the diagnostic object. The method of Boolean complement to modular sum codes is advisable to use in practice.

Keywords — CED system; code methods for the CED systems synthesis; modular sum code; error detection on device outputs; CED systems synthesis; self-checking device.

REFERENCES

- [1] Sogomonyan E.S., Slabakov E.V. Self-checking devices and fail-safe systems. Moscow: Radio and communication, 1989, 208 p.
- [2] Mikoni S.V. General diagnostic knowledge base of computing systems. – St. Petersburg: SPIIRAN, 1992, 234 p.
- [3] Abramovici M., Breuer M.A., Friedman A.D. Digital System Testing and Testable Design. – New Jersey: IEEE Press, 1998, 652 p.
- [4] Lala P.K. Self-Checking and Fault-Tolerant Digital Design. – San Francisco: Morgan Kaufmann Publishers, 2001, 216 p.
- [5] Fujiwara E. Code Design for Dependable Systems: Theory and Practical Applications, John Wiley & Sons, 2006, 720 p.
- [6] Drozd A.V., Kharchenko V.S., Antoshchuk S.G., Drozd Yu.V., Drozd M.A., Sulima Yu.Yu. Working diagnostics of safe information and control systems. – Ed. A.V. Drozd and V.S. Kharchenko. – Kharkiv: National Aerospace University. NOT. Zhukovsky "KhAI", 2012, 614 p.
- [7] Sapozhnikov V.V., Sapozhnikov V.I., Efanov D.V. Hamming codes in functional control systems of logical devices, St. Petersburg: Nauka, 2018, 151 p.
- [8] Telpuhov D.V., Zhukova T.D., Demeneva A.I., Gurov S.I. Circuit of Functional Control for Combinational Circuits Based on R-Code // Problems of Perspective Micro- and Nanoelectronic Systems Development. 2018. Issue 4. P. 98-104, doi: 10.31114/2078-7707-2018-4-98-104.
- [9] Telpuhov D.V., Zhukova T.D., Demeneva A.I., Gurov S.I. Automatic Synthesis of Fault-Tolerant CED Circuits Based on R-Code // Problems of Perspective Micro- and Nanoelectronic Systems Development. . 2019. Issue 3, P. 54-59, doi: 10.31114/2078-7707-2019-3-54-59.
- [10] Zhukova T.D. Functional Control Circuits CAD System Based on Redundant Coding Methods // Problems of Perspective Micro- and Nanoelectronic Systems Development. 2020. Issue 4. P. 119-125, doi: 10.31114/2078-7707-2020-4-51-57.
- [11] Sapozhnikov V.V., Sapozhnikov V.I., Efanov D.V. Sum codes for technical diagnostics systems. Volume 1: Classical Berger codes and their modifications, Moscow: Nauka, 2020, 383 p.
- [12] Sapozhnikov V.V., Sapozhnikov V.I., Efanov D.V. Sum codes for technical diagnostic systems. Volume 2: Weighted codes with summation, Moscow: Nauka, 2021, 456 p.
- [13] Nicolaidis M., Zorian Y. On-Line Testing for VLSI – A Compendium of Approaches // Journal of Electronic Testing: Theory and Application, 1998, Vol. 12, Issue 1-2, Pp. 7-20, DOI: 10.1023/A:1008244815697.
- [14] 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, pp. 985-994, doi: 10.1109/TEST.2000.894311.
- [15] Gavrilov S.V., Gurov S.I., Zhukova T.D., Ryzhova D.I. Application of coding theory to improve noise immunity of combinational circuits // Information technologies, 2016, V. 22, No. 12, pp. 931-937.
- [16] Sapozhnikov V.V., Sapozhnikov V.I., Efanov D.V. Fundamentals of the theory of reliability and technical

- diagnostics, St. Petersburg: Publishing house "Lan", 2019, 588 p.
- [17] Baghdadi A.A.A., Khakhanov V.I., Litvinova E.I. Methods of analysis and diagnostics of digital devices (analytical review) // Automated control systems and automation devices, 2014, No. 166, pp. 59-74.
- [18] Matrosova A.Yu., Andreeva V.V., Tychinsky V.Z., Goshin G.G. The use of ROBDD graphs for testing the delays of logic circuits. Physics, 2019, V. 62, No. 5 (737), pp. 86-94.
- [19] Matrosova A.Yu., Tychinskiy V.Z., Andreeva V.V. Deriving Low Power Robust PDFs Based on Applying SAT-Solvers and Operations on ROBDDs // Problems of Perspective Micro- and Nanoelectronic Systems Development. 2020. Issue 2. P. 43-49, doi: 10.31114/2078-7707-2020-2-43-49.
- [20] Gessel M., Morozov AV, Sapozhnikov VV, Sapozhnikov VI.V. Boolean complement – a new method of control of combinational circuits // Automation and remote control, 2003, No. 1, pp. 167-176.
- [21] Gessel M., Morozov AV, Sapozhnikov VV, Sapozhnikov VI.V. Control of combinational circuits by the method of Boolean complement // Automation and remote control, 2005, No. 8, pp. 161-172.
- [22] Saposhnikov V.V., Saposhnikov VI.V., Morozov A., Goessel M. Osadchy G. Design of totally self-checking combinational circuits by use of complementary circuits // Proceedings of 2th IEEE East-West Design & Test Symposium (EWDTS'2004), Crimea, Ukraine, September 15 – 17, 2004, pp. 83-87.
- [23] Gössel M., Ocheretny V., Sogomonyan E., Marienfeld D. New Methods of Concurrent Checking: Edition 1, Dordrecht: Springer Science+Business Media B.V., 2008, 184 p.
- [24] Sen S.K., Roy S.S. An Optimized Concurrent Self-Checker Using Constraint-Don't Cares and 1-out-of-4 Code // National Conference (AECDISC-2008) in Asansol Engineering College, held during 1-2 August 2008.
- [25] Das D.K., Roy S.S., Dmitriev 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, pp. 33-40.
- [26] Pivovarov D.V. Construction of functional control systems for multi-output combinational circuits by the method of Boolean complement by constant-weight codes // Automation in Transport, 2018, Volume 4, No. 1, pp. 131-149.
- [27] Efanov D.V., Osadchiy G.V., Zueva M.V. Berger Codes in Built-In Control Circuits Implemented on the Basis of the Boolean Complement Method // Informatics and Control Systems, 2021, No. 1, pp. 75-89, DOI: 10.22250 / isu.2021.67.75-89.
- [28] Berger J.M. A Note on Error Detection Codes for Asymmetric Channels // Information and Control, 1961, Vol. 4, Issue 1, Pp. 68-73, doi: 10.1016/S0019-9958(61)80037-5.
- [29] Collection of Digital Design Benchmarks [Режим доступа: <http://ddd.fit.cvut.cz/prj/Benchmarks/>].
- [30] Sequential Circuit Design Using Synthesis and Optimization / E.M. Sentovich, K.J. Singh, C. Moon, H. Savoj, R.K. Brayton, A. Sangiovanni-Vincentelli // Proceedings IEEE International Conference on Computer Design: VLSI in Computers & Processors, 11-14 October 1992, Cambridge, MA, USA, USA pp. 328-333, doi: 10.1109/ICCD.1992.276282.