

УДК 004.052.42+681.518.5

## КОДЫ С СУММИРОВАНИЕМ ВЕСОВЫХ КОЭФФИЦИЕНТОВ РАЗРЯДОВ ИНФОРМАЦИОННЫХ ВЕКТОРОВ В КОЛЬЦЕ ВЫЧЕТОВ ПО ПРОИЗВОЛЬНОМУ МОДУЛЮ ДЛЯ СИНТЕЗА ЦИФРОВЫХ ВЫЧИСЛИТЕЛЬНЫХ УСТРОЙСТВ

**ЕФАНОВ Дмитрий Викторович**, д-р техн. наук, доцент, член Института инженеров электротехники и электроники (IEEE member), действительный член Международной академии транспорта, заместитель генерального директора по научно-исследовательской работе<sup>1</sup>, профессор<sup>2,3</sup>;  
e-mail: TrES-4b@yandex.ru

**ПАШУКОВ Артем Валерьевич**, старший преподаватель, аспирант кафедры<sup>3</sup>; e-mail: art\_pash@mail.ru

<sup>1</sup>ООО НТЦ «Комплексные системы мониторинга», Санкт-Петербург

<sup>2</sup>Санкт-Петербургский политехнический университет Петра Великого, Высшая школа транспорта, Института машиностроения, материалов и транспорта, Санкт-Петербург

<sup>3</sup>Российский университет транспорта (МИИТ), кафедры «Автоматика, телемеханика и связь на железнодорожном транспорте», Москва

Рассматриваются особенности кодов с суммированием весовых коэффициентов разрядов информационных векторов в кольце вычетов по произвольному модулю. Установлены характеристики обнаружения ошибок данным классом кодов и некоторые закономерности, связанные с использованием конкретных значений модулей и весовых коэффициентов. Показано, что число различных взвешенных кодов с суммированием ограничено, однако существует большое число способов их построения для каждого значения числа разрядов в информационных векторах. Приводится алгоритм получения разрядов контрольных векторов рассматриваемых кодов, примеры их построения, описан способ подсчета числа необнаруживаемых ошибок в информационных векторах. Определены свойства кодов с суммированием весовых коэффициентов информационных векторов в кольце вычетов по произвольному модулю, учет которых может быть полезен при решении задач технической диагностики, синтеза самопроверяемых и отказоустойчивых цифровых устройств и устройств с контролепригодными структурами. Описан подход к синтезу кодеров любых взвешенных кодов с суммированием на основе сумматоров двоичных чисел по установленному модулю. Приводятся некоторые результаты экспериментов по анализу обнаруживающих способностей рассматриваемых кодов в схемах встроенного контроля тестовых комбинационных устройств. Полученные в работе результаты носят универсальный характер и не ориентированы на применение только с одной элементной базой реализуемых устройств, что делает их полезными не только в настоящее время, но и в будущем.

**Ключевые слова:** отказоустойчивые цифровые системы; контролепригодные устройства; самопроверяемые схемы встроенного контроля; код с суммированием; взвешенный код с суммированием; суммирование в кольце вычетов по произвольному модулю; обнаружение ошибок в информационных разрядах; свойства взвешенного кода с суммированием.

**DOI:** 10.20295/2412-9186-2022-8-02-198-217

### ▼ Введение

При разработке устройств и систем критического применения как в промышленности, так и на транспорте широко применяются методы синтеза контролепригодных, самопроверяемых и отказоустойчивых структур [1–5]. Все эти методы базируются фактически на внесении

структурной, информационной или временной избыточности для получения того или иного свойства разрабатываемого устройства или системы. Часто это делается с применением методов теории информации и кодирования.

Одними из повсеместно используемых при синтезе надежных цифровых систем кодами

являются коды с суммированием [6–10]. Их широкое применение обусловлено простотой реализации кодирующего и декодирующего оборудования, а также сравнительно малой избыточностью.

Существует большое разнообразие кодов с суммированием [11, 12], особое место среди которых занимают коды, при формировании которых информационным разрядам ставятся в соответствие весовые коэффициенты (веса), — некоторые натуральные числа, что позволяет установить неравноправие между этими разрядами. Такие коды относятся к взвешенным кодам с суммированием.

Одной из первых работ, в которой описаны взвешенные коды с суммированием, является статья Джея Бергера [13]. В ней операция взвешивания введена для получения кода с суммированием, более эффективно обнаруживающего ошибки в области малой их кратности, чем классический код с суммированием (код Бергера). Информационным разрядам ставятся в соответствие веса из натурального ряда чисел за исключением степеней числа 2: то есть числа из последовательности 3, 5, 6, 7, 9, 10 и т. д. Такой взвешенный код обнаруживает все однонаправленные (монотонные) ошибки, а также все одно- и двукратные ошибки. Однако взвешенный код Бергера обладает высокой избыточностью, что ограничивает его применимость на практике. Например, при построении устройств с контролепригодными структурами и при построении схем встроенного контроля логических устройств использование взвешенного кода Бергера приводит к внесению весомой аппаратной избыточности, сравнимой с дублированием (и даже превосходящей ее!) [11, 12]. В [14] Джейм Бергером описан еще один способ построения кода, подразумевающий взвешивание коэффициентами из последовательности возрастающих степеней числа 2 подряд расположенных разрядов, что дает код, обнаруживающий пакеты ошибок. Длину пакета обнаруживаемых ошибок определяют на этапе построения кода. Такой код также имеет большую избыточность, а его применение при решении задач синтеза цифровых устройств из-за этого ограничено.

В технической диагностике цифровых вычислительных устройств и систем применяются коды с суммированием с различными, но наперед заданными, значениями весовых коэффициентов [15] и коды с суммированием весовых коэффициентов в кольце вычетов по заданному модулю [16].

Важными для задач технической диагностики цифровых устройств являются результаты исследований обнаруживающих характеристик взвешенных кодов с суммированием в информационных векторах. Например, при синтезе СВК по взвешенным кодам с суммированием часто информационные разряды отождествляют с выходами объекта диагностирования, а контрольный вектор формируют в самих технических средствах диагностирования, чем физически разделяют информационный и контрольный векторы [11, 12]. Наиболее вероятная одиночная неисправность может в таком случае возникать только в объекте диагностирования либо только в технических средствах диагностирования. Последние строятся самопроверяемыми, что позволяет обнаруживать неисправности сразу же в то время, когда они возникают [17]. Неисправности же объекта диагностирования приводят к возникновению ошибок в информационных векторах. Ошибки могут иметь разную кратность (число искажаемых разрядов) и различные виды (по числу сочетаний искажений нулевых и единичных разрядов) [11]. На основании характеристик обнаружения ошибок в разрядах информационных векторов определяют и принципы синтеза технических средств диагностирования для построения устройств с обнаружением неисправностей [18–22].

Ранее в научных публикациях были освещены результаты исследований особенностей обнаружения ошибок в информационных векторах кодами с суммированием, при построении которых разрядам приписывались произвольные числа с ограничением избыточности кода избыточностью кодов Бергера [23, 24] и разнообразных кодов с суммированием с натуральной последовательностью весовых коэффициентов [25–28]. В [29] описаны особенности обнаружения ошибок кодами с суммированием в кольце вычетов по модулю  $M \in \{2^2, 2^3, \dots, 2^{\lceil \log_2(m+1) \rceil}\}$ ,

при построении которых неединичным весовым коэффициентом взвешивался только один информационный разряд. Анализ показывает, что неизученными оказались коды, при построении которых применяют суммирование значений весов в кольце вычетов по произвольному, наперед заданному, модулю. Изложению результатов их исследований и посвящена настоящая статья.

**1. Особенности взвешивания разрядов при построении кодов с суммированием**

Процедура взвешивания разрядов информационного вектора позволяет существенно расширить число способов построения кодов с суммированием по сравнению с операциями над невзвешенными разрядами [11, 12].

Взвешенный код с суммированием строится по следующим правилам:

1. Для заданного значения длины информационного вектора  $m$  определяется последовательность весовых коэффициентов, приписываемых

информационным разрядам:  $[w] = [w_m, w_{m-1}, \dots, w_2, w_1]$  (весовой коэффициент является натуральным числом).

2. Устанавливается значение модуля  $M \in \{2, 3, \dots, W_{\max} + 1\}$ , где  $W_{\max} = \sum_{i=1}^m w_i$ .

3. Подсчитывается число  $W = \sum_{i=1}^m w_i f_i$ , где  $f_i$  — значение  $i$ -го информационного разряда,  $i = 1, m$ .

4. Вычисляется наименьший неотрицательный вычет числа  $W$  по модулю  $M$ :  $W(\text{mod}M)$ .

5. Значение  $W(\text{mod}M)$  записывается в  $k$  разрядах контрольного вектора в двоичном виде.

Введем обозначение рассматриваемого класса кодов —  $WSM(m, k)$ -коды. При этом будем отдельно указывать последовательность весовых коэффициентов, используемую при их построении. Число контрольных разрядов в  $WSM(m, k)$ -кодах определяется значением модуля:  $k = \lceil \log_2 M \rceil$ . В табл. 1 приведены все значения наименьших неотрицательных

Таблица 1. Значения  $W(\text{mod}M)$  для различных вариантов построения  $WSM(4, k)$ -кодов

Десятичный номер вектора	Значения весовых коэффициентов				$W(\text{mod}M)$								
	$w_4 = 3$	$w_3 = 2$	$w_2 = 2$	$w_1 = 1$									
	Значения информационных разрядов				$W(\text{mod}9)$	$W(\text{mod}8)$	$W(\text{mod}7)$	$W(\text{mod}6)$	$W(\text{mod}5)$	$W(\text{mod}4)$	$W(\text{mod}3)$	$W(\text{mod}2)$	
	$f_4$	$f_3$	$f_2$	$f_1$									
0	0	0	0	0	0	0	0	0	0	0	0	0	
1	0	0	0	1	1	1	1	1	1	1	1	1	
2	0	0	1	0	2	2	2	2	2	2	2	0	
3	0	0	1	1	3	3	3	3	3	3	0	1	
4	0	1	0	0	2	2	2	2	2	2	2	0	
5	0	1	0	1	3	3	3	3	3	3	0	1	
6	0	1	1	0	4	4	4	4	4	0	1	0	
7	0	1	1	1	5	5	5	5	0	1	2	1	
8	1	0	0	0	3	3	3	3	3	3	0	1	
9	1	0	0	1	4	4	4	4	4	0	1	0	
10	1	0	1	0	5	5	5	5	0	1	2	1	
11	1	0	1	1	6	6	6	0	1	2	0	0	
12	1	1	0	0	5	5	5	5	0	1	2	1	
13	1	1	0	1	6	6	6	0	1	2	0	0	
14	1	1	1	0	7	7	0	1	2	3	1	1	
15	1	1	1	1	8	0	1	2	3	0	2	0	

вычетов, получаемые при построении  $WSM(4,k)$ -кодов со значениями весовых коэффициентов  $[w_4, w_3, w_2, w_1] = [3, 2, 2, 1]$ .

Количество способов построения  $WSM(m,k)$ -кодов зависит от длины  $m$  информационного вектора и максимального значения суммарного веса коэффициентов (числа  $W_{max} = \sum_{i=1}^m w_i$ ). Оно равно  $W_{max}$ , поскольку значение  $M$  может быть выбрано из множества  $M \in \{2, 3, \dots, W_{max} + 1\}$ . К примеру, в табл. 1 приведены все варианты формирования вычетов  $W(\text{mod}M)$  при построении  $WSM(4,k)$ -кодов, для которых значения весовых коэффициентов равны  $[w_4, w_3, w_2, w_1] = [3, 2, 2, 1]$ . Для данных параметров могут быть построены  $W_{max} = 8$  кодов. Фактически значения вычетов определяют конкретные контрольные векторы для каждого из информационных векторов.

Обнаруживающие свойства кода связаны с распределением информационных векторов между контрольными векторами (контрольными группами, соответствующими всем возможным вычетам) [11, 12]. Для рассматриваемых кодов такие распределения приведены в табл. 2.

Необнаруживаемой ошибка будет при условии возникновения перехода информационных векторов, принадлежащих одной и той же контрольной группе, друг в друга.

Ошибки характеризуются кратностью и видом [11]. Кратность  $d$  ошибки — это количество искажаемых при возникновении ошибки разрядов. Вид ошибки характеризуется соотношением числа искажаемых нулевых и единичных разрядов вектора. Выделяют одиночные, монотонные, симметричные и асимметричные ошибки. Одиночная, или однократная, ошибка возникает при искажении только одного разряда. Монотонная, или однонаправленная, ошибка происходит при искажении только нулевых или только единичных разрядов. Симметричная — при искажении равного числа нулевых и единичных разрядов. Асимметричная — при искажении и нулевых, и единичных разрядов, но неравного их количества. Для установления особенностей обнаружения ошибок различных кратностей и видов требуется анализ взаимных переходов друг в друга

Таблица 2. Распределение информационных векторов по значениям  $W(\text{mod}M)$

0	1	2	3	4	5	6	7	8
<b>WS9(4,4)</b>								
0000	0001	0010	0011	0110	0111	1011	1110	1111
		0100	0101	1001	1010	1101		
			1000		1100			
<b>WS8(4,3)</b>								
0000	0001	0010	0011	0110	0111	1011	1110	
1111		0100	0101	1001	1010	1101		
			1000		1100			
<b>WS7(4,3)</b>								
0000	0001	0010	0011	0110	0111	1011		
1110	1111	0100	0101	1001	1010	1101		
			1000		1100			
<b>WS6(4,3)</b>								
0000	0001	0010	0011	0110	0111			
1011	1110	0100	0101	1001	1010			
1101		1111	1000		1100			
<b>WS5(4,3)</b>								
0000	0001	0010	0011	0110				
0111	1011	0100	0101	1001				
1010	1101	1110	1000					
1100			1111					
<b>WS4(4,2)</b>								
0000	0001	0010	0011					
0110	0111	0100	0101					
1001	1010	1011	1000					
1111	1100	1101	1110					
<b>WS3(4,2)</b>								
0000	0001	0010						
0011	0110	0100						
0101	1001	0111						
1000	1110	1010						
1011		1100						
1101		1111						
<b>WS2(4,1)</b>								
0000	0001							
0010	0011							
0100	0101							
0110	0111							
1001	1000							
1011	1010							
1101	1100							
1111	1110							

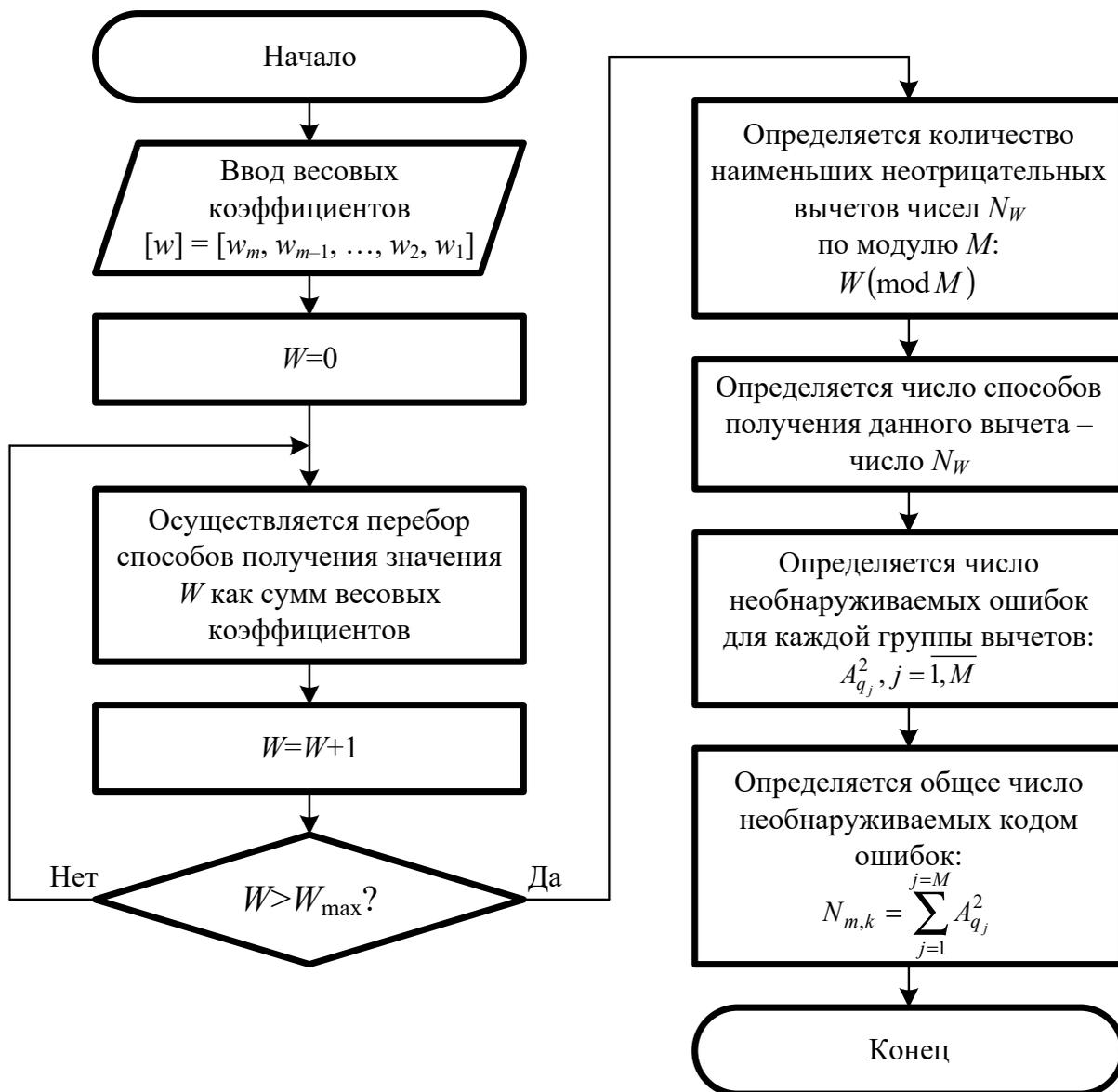


Рис. 1. Алгоритм подсчета числа необнаруживаемых  $WSM(m,k)$ -кодами ошибок в информационных векторах

информационных векторов, имеющих одинаковые контрольные векторы. Такой анализ позволяет подсчитать точное число ошибок в информационных векторах  $WSM(m,k)$ -кодов, классифицируя их по видам и кратностям.

Важной особенностью  $WSM(m,k)$ -кодов является то, что коды, при построении которых использован хотя бы один весовой коэффициент, значение которого кратно значению модуля  $M$ , не будут обнаруживать некоторую долю одиночных ошибок (не будут помехозащищенными). Это вытекает из используемого способа построения кода, так как  $iM \pmod M = 0$ , где  $i$  — произвольное целое число. Например, читатель может обратить

внимание на контрольную группу №0 для  $WS3(4,2)$ -кода и сравнить информационные векторы  $\langle 0000 \rangle$  и  $\langle 1000 \rangle$ . Расстояние Хэмминга между ними  $d = 1$ , а значит, в число необнаруживаемых добавляются две однократные ошибки. Вообще из рассматриваемых в примере кодов не будут помехозащищенными  $WS2(4,1)$  и  $WS3(4,2)$  коды.

Число необнаруживаемых  $WSM(m,k)$ -кодами ошибок можно подсчитать по алгоритму, приведенному на рис. 1.

Продemonстрируем работу алгоритма подсчета числа необнаруживаемых ошибок на примере  $WS5(4,3)$ -кода с весами  $[w_4, w_3, w_2, w_1] = [3, 2, 2, 1]$ .

Вес  $W=0$  может быть получен только для вектора  $\langle 0000 \rangle$ . Вес  $W=1$  формируется также только для одного вектора, для которого единице равен разряд с весом  $w_1$ . Вес  $W=2$  может быть получен двумя вариантами: при равенстве единице разрядов с весами  $w_2$  или  $w_3$ . Вес  $W=3$  получается несколькими способами: 1) как сумма коэффициентов  $w_1 + w_2$  и  $w_1 + w_3$ ; 2) как коэффициент  $w_4$ . Вес  $W=4$ :  $w_1 + w_4$ , либо  $w_2 + w_3$ . Вес  $W=5$ :  $w_1 + w_2 + w_3$ , либо  $w_2 + w_4$ , либо  $w_3 + w_4$ . Вес  $W=6$ :  $w_1 + w_2 + w_4$ , либо  $w_1 + w_3 + w_4$ . Вес  $W=7$ :  $w_2 + w_3 + w_4$ . Вес  $W=8$ :  $w_1 + w_2 + w_3 + w_4$ . Таким образом, перебраны все варианты сумм весовых коэффициентов, число которых равно

$$\sum_{i=0}^m C_m^i = 2^m.$$

Число вариантов формирования каждого значения  $W$  приведено в табл. 3.

Для подсчета общего числа необнаруживаемых ошибок требуется определить только число вычетов по установленному модулю  $M$ . В данном случае  $M=5$ . С учетом того, что  $5(\bmod 5) = 0$ ,  $6(\bmod 5) = 1$ ,  $7(\bmod 5) = 2$ ,  $8(\bmod 5) = 3$ , требуется добавить число вариантов получения вычетов из строк 5, 6, 7 и 8 соответственно в строки 0, 1, 2 и 3. Конечный вариант определения числа различных наименьших неотрицательных вычетов представлен в табл. 4.

Фактически каждая строка табл. 4 соответствует контрольной группе рассматриваемого  $WS5(4,3)$ -кода. Число необнаруживаемых ошибок для каждой контрольной группы определяется числом размещений из  $q_j$  по 2 ( $A_{q_j}^2$ ), где  $q_j$  — число вычетов в  $j$ -ой строке,  $j=1, M$ , поскольку для каждой пары информационных векторов рассматривается двунаправленный переход (переход одного вектора в другой и наоборот). Общее же число необнаруживаемых ошибок будет равно:

$$N_{m,k} = \sum_{j=1}^{j=M} A_{q_j}^2 = \sum_{j=1}^{j=M} q_j (q_j - 1).$$

Для  $WS5(4,3)$ -кода имеем:  $N_{4,3} = 4 \cdot 3 + 3 \cdot 2 + 3 \cdot 2 + 4 \cdot 3 + 2 \cdot 1 = 12 + 6 + 6 + 12 + 2 = 38$  необнаруживаемых в информационных векторах ошибок.

Аналогично можно определить число необнаруживаемых ошибок по их видам (монотонных,

Таблица 3. Число вариантов подсчета каждого числа  $W$

Номер строки, $j$	$W$	Число способов формирования
1	0	1
2	1	1
3	2	2
4	3	3
5	4	2
6	5	3
7	6	2
8	7	1
9	8	1

Таблица 4. Число вариантов подсчета каждого числа  $W(\bmod 5)$

Номер строки, $j$	$W(\bmod 5)$	Число способов формирования
1	0	4
2	1	3
3	2	3
4	3	4
5	4	2

симметричных и асимметричных). Для подсчета числа необнаруживаемых ошибок требуется не только анализ числа способов формирования того или иного веса, но и расположения единичных разрядов для каждого вектора с одинаковым суммарным весом. Здесь его приводить не будем. Далее все подсчеты были автоматизированы и реализованы в специализированном программном модуле, реализованном при участии авторов.

## 2. Характеристики обнаружения ошибок $WSM(m,k)$ -кодами

С увеличением значения модуля  $M$  существенно возрастает число способов взвешивания разрядов информационного вектора. К примеру, при  $M=2$  имеется один способ взвешивания разрядов; при  $M=3$  их становится больше —  $m+1$  и т. д.

В табл. 5–8 приведены характеристики обнаружения ошибок некоторыми  $WSM(m,k)$ -кодами со значением длины информационного вектора  $m=4$ . Представлены данные для

Таблица 5. Характеристики  $WS2(4, 1)$ -кода

[w]	$\gamma_{m,k}$	$\xi_{m,k}$	$\nu_{m,k}$	$\sigma_{m,k}$	$\alpha_{m,k}$	$N_{m,k}$	Распределение необнаруживаемых ошибок по кратностям $d$			
							1	2	3	4
[1,1,1,1]	46,667	100	44,643	48,214	7,143	112	0	96	0	16
						50 / 54 / 8	0 / 0 / 0	48 / 48 / 0	0 / 0 / 0	2 / 6 / 8

Таблица 6. Характеристики  $WS3(4, 2)$ -кода

[w]	$\gamma_{m,k}$	$\xi_{m,k}$	$\nu_{m,k}$	$\sigma_{m,k}$	$\alpha_{m,k}$	$N_{m,k}$	Распределение необнаруживаемых ошибок по кратностям $d$			
							1	2	3	4
[1,1,1,1]	29,167	68,571	22,857	77,143	0	70	0	48	16	6
						16 / 54 / 0	0 / 0 / 0	0 / 48 / 0	16 / 0 / 0	0 / 6 / 0
[1,1,1,2]	29,167	68,571	40	34,286	25,714	70	0	48	16	6
						28 / 24 / 18	0 / 0 / 0	24 / 24 / 0	4 / 0 / 12	0 / 0 / 6
[1,1,2,2]	29,167	68,571	48,572	28,571	22,857	70	0	48	16	6
						34 / 20 / 16	0 / 0 / 0	32 / 16 / 0	0 / 0 / 16	2 / 4 / 0
[1,2,2,2]	29,167	68,571	40	34,286	25,714	70	0	48	16	6
						28 / 24 / 18	0 / 0 / 0	24 / 24 / 0	4 / 0 / 12	0 / 0 / 6
[2,2,2,2]	29,167	68,571	22,857	77,143	0	70	0	48	16	6
						16 / 54 / 0	0 / 0 / 0	0 / 48 / 0	16 / 0 / 0	0 / 6 / 0

$WS2(4,1)$ ,  $WS3(4,2)$ ,  $WS4(4,2)$ ,  $WS5(4,3)$  кодов.  $WS2(4,1)$ -код — один.  $WS3(4,2)$ -кодов — пять.  $WS4(4,2)$ -кодов — 15.  $WS5(4,3)$  — 35.

Увеличение модуля приводит к существенному росту числа способов взвешивания [30]. При этом необходимо заметить, что рассмотрены коды с учетом формирования полного множества информационных векторов, и при этом важно учитывать конкретное сочетание весовых коэффициентов, а не то, какие именно разряды взвешены какими числами. К примеру, последовательность [1, 2, 2, 3] дает точно такой же по характеристикам взвешенный код, что и последовательность [1, 3, 2, 2]. Всего имеется 12 способов взвешивания четырехразрядного вектора указанными числами. Таким образом, заведомо учтены особенности последовательностей весовых коэффициентов при построении кода с суммированием.

В характеристических таблицах приведены абсолютные величины и относительные. Рассчитано количество необнаруживаемых ошибок с классификацией их по кратностям и видам. В каждой ячейке таблицы для конкретной последовательности весов приведены: в верхней строке — количество ошибок данной

кратностью, в нижней строке — через косые черты числа необнаруживаемых монотонных, симметричных и асимметричных ошибок соответственно. Также рассчитаны относительные показатели обнаружения ошибок.

Показатель  $\gamma_{m,k}$  — доля необнаруживаемых кодом ошибок ( $N_{m,k}$ ) от общего их количества в информационных векторах ( $N_m$ ). Оно рассчитывается по формуле:

$$\gamma_{m,k} = 100\% \cdot \frac{N_{m,k}}{N_m} = 100\% \cdot \frac{N_{m,k}}{2^m (2^m - 1)}.$$

Показатель  $\xi_{m,k}$  говорит о том, насколько близок рассматриваемый код к коду с наименьшим общим числом необнаруживаемых ошибок при заданных  $m$  и  $k$  [11]:

$$\xi_{m,k} = 100\% \cdot \frac{N_{m,k}^{\min}}{N_{m,k}} = 100\% \cdot \frac{2^m (2^{m-k} - 1)}{N_{m,k}}.$$

Чем ближе значение  $\xi_{m,k}$  к 100 %, тем ближе рассматриваемый код к оптимальному по критерию минимума числа  $N_{m,k}$  при заданных  $m$  и  $k$ . Между коэффициентами  $\gamma_{m,k}$  и  $\xi_{m,k}$  имеется зависимость, однако далее будем их

**Таблица 7. Характеристики WS4(4,2)-кода**

[w]	$\gamma_{m,k}$	$\xi_{m,k}$	$\upsilon_{m,k}$	$\sigma_{m,k}$	$\alpha_{m,k}$	$N_{m,k}$	Распределение необнаруживаемых ошибок по кратностям $d$			
							1	2	3	4
[1,1,1,1]	23,333	85,714	3,571	96,429	0	56	0	48	0	8
						2 / 54 / 0	0 / 0 / 0	0 / 48 / 0	0 / 0 / 0	2 / 6 / 0
[1,1,1,2]	20	100	25	50	25	48	0	24	24	0
						12 / 24 / 12	0 / 0 / 0	0 / 24 / 0	12 / 0 / 12	0 / 0 / 0
[1,1,1,3]	23,333	85,714	42,857	42,857	14,286	56	0	48	0	8
						24 / 24 / 8	0 / 0 / 0	24 / 24 / 0	0 / 0 / 0	0 / 0 / 8
[1,1,2,2]	20	100	33,333	41,667	25	48	0	24	16	8
						16 / 20 / 12	0 / 0 / 0	8 / 16 / 0	8 / 0 / 8	0 / 4 / 4
[1,1,2,3]	20	100	41,667	16,666	41,667	48	0	24	24	0
						20 / 8 / 20	0 / 0 / 0	16 / 8 / 0	4 / 0 / 20	0 / 0 / 0
[1,1,3,3]	23,333	85,714	60,714	39,286	0	56	0	48	0	8
						34 / 22 / 0	0 / 0 / 0	32 / 16 / 0	0 / 0 / 0	2 / 6 / 0
[1,2,2,2]	20	100	50	50	0	48	0	48	0	0
						24 / 24 / 0	0 / 0 / 0	24 / 24 / 0	0 / 0 / 0	0 / 0 / 0
[1,2,2,3]	20	100	37,5	20,833	41,667	48	0	24	16	8
						18 / 10 / 20	0 / 0 / 0	16 / 8 / 0	0 / 0 / 16	2 / 2 / 4
[1,2,3,3]	20	100	41,667	16,666	41,667	48	0	24	24	0
						20 / 8 / 20	0 / 0 / 0	16 / 8 / 0	4 / 0 / 20	0 / 0 / 0
[1,3,3,3]	23,333	85,714	42,857	42,857	14,286	56	0	48	0	8
						24 / 24 / 8	0 / 0 / 0	24 / 24 / 0	0 / 0 / 0	0 / 0 / 8
[2,2,2,2]	46,667	42,857	44,643	48,214	7,143	112	0	96	0	16
						50 / 54 / 8	0 / 0 / 0	48 / 48 / 0	0 / 0 / 0	2 / 6 / 8
[2,2,2,3]	20	100	50	50	0	48	0	48	0	0
						24 / 24 / 0	0 / 0 / 0	24 / 24 / 0	0 / 0 / 0	0 / 0 / 0
[2,2,3,3]	20	100	33,333	41,667	25	48	0	24	16	8
						16 / 20 / 12	0 / 0 / 0	8 / 16 / 0	8 / 0 / 8	0 / 4 / 4
[2,3,3,3]	20	100	25	50	25	48	0	24	24	0
						12 / 24 / 12	0 / 0 / 0	0 / 24 / 0	12 / 0 / 12	0 / 0 / 0
[3,3,3,3]	23,333	85,714	3,571	96,429	0	56	0	48	0	8
						2 / 54 / 0	0 / 0 / 0	0 / 48 / 0	0 / 0 / 0	2 / 6 / 0

рассматривать наравне, так как они позволяют наглядно демонстрировать особенности обнаружения ошибок и эффективность использования контрольных разрядов.

Показатели  $\upsilon_{m,k}$ ,  $\sigma_{m,k}$  и  $\alpha_{m,k}$  характеризуют долю монотонных, симметричных и асимметричных ошибок от общего числа необнаруживаемых кодом ошибок. Из анализа характеристических таблиц следует, что оптимальный по критерию минимума необнаруживаемых

ошибок при конкретных  $m$  и  $k$  код, для которого  $\xi_{m,k}=100\%$ , строится только при значении модуля  $M \in \{2^1, 2^2, 2^3, \dots, 2^{\lceil \log_2(m+1) \rceil}\}$ . Во всех остальных случаях код оптимальным не будет. Подобная особенность WSM( $m,k$ )-кодов связана с возможностями представления чисел в двоичной системе счисления и возможностями получения полного множества контрольных векторов для конкретного числа  $k$ . При значениях модулей  $M \notin \{2^1, 2^2, 2^3, \dots, 2^{\lceil \log_2(m+1) \rceil}\}$

Таблица 8. Характеристики WS5(4,3)-кода

[w]	$\gamma_{m,k}$	$\xi_{m,k}$	$\nu_{m,k}$	$\sigma_{m,k}$	$\alpha_{m,k}$	$N_{m,k}$	Распределение необнаруживаемых ошибок по кратностям d			
							1	2	3	4
[1,1,1,1]	22,5	29,63	0	100	0	54	0	48	0	6
						0 / 54 / 0	0 / 0 / 0	0 / 48 / 0	0 / 0 / 0	0 / 6 / 0
[1,1,1,2]	15,833	42,105	5,263	63,158	31,579	38	0	24	12	2
						2 / 24 / 12	0 / 0 / 0	0 / 24 / 0	0 / 0 / 12	2 / 0 / 0
[1,1,1,3]	15,833	42,105	31,579	63,158	5,263	38	0	24	12	2
						12 / 24 / 2	0 / 0 / 0	0 / 24 / 0	12 / 0 / 0	0 / 0 / 2
[1,1,1,4]	22,5	29,63	44,444	44,444	11,112	54	0	48	0	6
						24 / 24 / 6	0 / 0 / 0	24 / 24 / 0	0 / 0 / 0	0 / 0 / 6
[1,1,2,2]	15	44,444	22,222	55,556	22,222	36	0	16	16	4
						8 / 20 / 8	0 / 0 / 0	0 / 16 / 0	8 / 0 / 8	0 / 4 / 0
[1,1,2,3]	15	44,444	33,333	22,222	44,445	36	0	16	16	4
						12 / 8 / 16	0 / 0 / 0	8 / 8 / 0	4 / 0 / 12	0 / 0 / 4
[1,1,2,4]	15,833	42,105	42,105	21,053	36,842	38	0	24	12	2
						16 / 8 / 14	0 / 0 / 0	16 / 8 / 0	0 / 0 / 12	0 / 0 / 2
[1,1,3,3]	15	44,444	22,222	55,556	22,222	36	0	16	16	4
						8 / 20 / 8	0 / 0 / 0	0 / 16 / 0	8 / 0 / 8	0 / 4 / 0
[1,1,3,4]	15,833	42,105	52,631	26,316	21,053	38	0	24	12	2
						20 / 10 / 8	0 / 0 / 0	16 / 8 / 0	4 / 0 / 8	0 / 2 / 0
[1,1,4,4]	22,5	29,63	62,963	37,037	0	54	0	48	0	6
						34 / 20 / 0	0 / 0 / 0	32 / 16 / 0	0 / 0 / 0	2 / 4 / 0
[1,2,2,2]	15,833	42,105	31,579	63,158	5,263	38	0	24	12	2
						12 / 24 / 2	0 / 0 / 0	0 / 24 / 0	12 / 0 / 0	0 / 0 / 2
[1,2,2,3]	15,833	42,105	52,631	26,316	21,053	38	0	24	12	2
						20 / 10 / 8	0 / 0 / 0	16 / 8 / 0	4 / 0 / 8	0 / 2 / 0
[1,2,2,4]	15	44,444	33,333	22,222	44,445	36	0	16	16	4
						12 / 8 / 16	0 / 0 / 0	8 / 8 / 0	4 / 0 / 12	0 / 0 / 4
[1,2,3,3]	15,833	42,105	42,105	21,053	36,842	38	0	24	12	2
						16 / 8 / 14	0 / 0 / 0	16 / 8 / 0	0 / 0 / 12	0 / 0 / 2
[1,2,3,4]	15	44,444	50	5,556	44,444	36	0	16	16	4
						18 / 2 / 16	0 / 0 / 0	16 / 0 / 0	0 / 0 / 16	2 / 2 / 0
[1,2,4,4]	15,833	42,105	52,631	26,316	21,053	38	0	24	12	2
						20 / 10 / 8	0 / 0 / 0	16 / 8 / 0	4 / 0 / 8	0 / 2 / 0
[1,3,3,3]	15,833	42,105	5,263	63,158	31,579	38	0	24	12	2
						2 / 24 / 12	0 / 0 / 0	0 / 24 / 0	0 / 0 / 12	2 / 0 / 0
[1,3,3,4]	15	44,444	33,333	22,222	44,445	36	0	16	16	4
						12 / 8 / 16	0 / 0 / 0	8 / 8 / 0	4 / 0 / 12	0 / 0 / 4
[1,3,4,4]	15,833	42,105	42,105	21,053	36,842	38	0	24	12	2
						16 / 8 / 14	0 / 0 / 0	16 / 8 / 0	0 / 0 / 12	0 / 0 / 2
[1,4,4,4]	22,5	29,63	44,444	44,444	11,112	54	0	48	0	6
						24 / 24 / 6	0 / 0 / 0	24 / 24 / 0	0 / 0 / 0	0 / 0 / 6
[2,2,2,2]	22,5	29,63	0	100	0	54	0	48	0	6
						0 / 54 / 0	0 / 0 / 0	0 / 48 / 0	0 / 0 / 0	0 / 6 / 0

Окончание таблицы 8

[w]	$\gamma_{m,k}$	$\xi_{m,k}$	$\nu_{m,k}$	$\sigma_{m,k}$	$\alpha_{m,k}$	$N_{m,k}$	Распределение необнаруживаемых ошибок по кратностям $d$			
							1	2	3	4
[2,2,2,3]	22,5	29,63	44,444	44,444	11,112	54	0	48	0	6
						24 / 24 / 6	0 / 0 / 0	24 / 24 / 0	0 / 0 / 0	0 / 0 / 6
[2,2,2,4]	15,833	42,105	5,263	63,158	31,579	38	0	24	12	2
						2 / 24 / 12	0 / 0 / 0	0 / 24 / 0	0 / 0 / 12	2 / 0 / 0
[2,2,3,3]	22,5	29,63	62,963	37,037	0	54	0	48	0	6
						34 / 20 / 0	0 / 0 / 0	32 / 16 / 0	0 / 0 / 0	2 / 4 / 0
[2,2,3,4]	15,833	42,105	42,105	21,053	36,842	38	0	24	12	2
						16 / 8 / 14	0 / 0 / 0	16 / 8 / 0	0 / 0 / 12	0 / 0 / 2
[2,2,4,4]	15	44,444	22,222	55,556	22,222	36	0	16	16	4
						8 / 20 / 8	0 / 0 / 0	0 / 16 / 0	8 / 0 / 8	0 / 4 / 0
[2,3,3,3]	22,5	29,63	44,444	44,444	11,112	54	0	48	0	6
						24 / 24 / 6	0 / 0 / 0	24 / 24 / 0	0 / 0 / 0	0 / 0 / 6
[2,3,3,4]	15,833	42,105	52,631	26,316	21,053	38	0	24	12	2
						20 / 10 / 8	0 / 0 / 0	16 / 8 / 0	4 / 0 / 8	0 / 2 / 0
[2,3,4,4]	15	44,444	33,333	22,222	44,445	36	0	16	16	4
						12 / 8 / 16	0 / 0 / 0	8 / 8 / 0	4 / 0 / 12	0 / 0 / 4
[2,4,4,4]	15,833	42,105	31,579	63,158	5,263	38	0	24	12	2
						12 / 24 / 2	0 / 0 / 0	0 / 24 / 0	12 / 0 / 0	0 / 0 / 2
[3,3,3,3]	22,5	29,63	0	100	0	54	0	48	0	6
						0 / 54 / 0	0 / 0 / 0	0 / 48 / 0	0 / 0 / 0	0 / 6 / 0
[3,3,3,4]	15,833	42,105	31,579	63,158	5,263	38	0	24	12	2
						12 / 24 / 2	0 / 0 / 0	0 / 24 / 0	12 / 0 / 0	0 / 0 / 2
[3,3,4,4]	15	44,444	22,222	55,556	22,222	36	0	16	16	4
						8 / 20 / 8	0 / 0 / 0	0 / 16 / 0	8 / 0 / 8	0 / 4 / 0
[3,4,4,4]	15,833	42,105	5,263	63,158	31,579	38	0	24	12	2
						2 / 24 / 12	0 / 0 / 0	0 / 24 / 0	0 / 0 / 12	2 / 0 / 0
[4,4,4,4]	22,5	29,63	0	100	0	54	0	48	0	6
						0 / 54 / 0	0 / 0 / 0	0 / 48 / 0	0 / 0 / 0	0 / 6 / 0

формирование всех возможных контрольных векторов для конкретного числа  $k$  оказывается невозможным. Это же обстоятельство усложняет процедуру построения полностью самопроверяемого кодера  $WSM(m,k)$ -кода, так как для полной его проверки потребуется подача всех возможных контрольных векторов. Эта задача, однако, решается схемотехнически.

$WSM(m,k)$ -код со значением модуля  $M \in \{2^1, 2^2, 2^3, \dots, 2^{\lceil \log_2(m+1) \rceil}\}$  не при любых сочетаниях весовых коэффициентов будет иметь значение коэффициента  $\xi_{m,k} = 100\%$ . Например, для  $WS4(4,2)$ -кода (см. табл. 7) оптимальный

код строится только для следующих сочетаний весовых коэффициентов:  $[1, 1, 1, 2]$ ,  $[1, 1, 2, 2]$ ,  $[1, 1, 2, 3]$ ,  $[1, 2, 2, 2]$ ,  $[1, 2, 2, 3]$ ,  $[1, 2, 3, 3]$ ,  $[2, 2, 2, 3]$ ,  $[2, 2, 3, 3]$  и  $[2, 3, 3, 3]$ . Отмеченная особенность объясняется возможностью получения равномерного числа вычетов сумм всех весовых коэффициентов по модулю  $M = 4$  (включая число 0 для вектора  $\langle 0000 \rangle$ ). Например, в табл. 2 задан в том числе и аналог  $WS4(4,2)$ -кода с  $[w] = [1, 2, 2, 3]$  (такая последовательность равнозначна последовательности  $[3, 2, 2, 1]$ ). Для  $WS4(4,2)$ -кода формируются вычеты из множества  $W \pmod{4} = \{0, 1, 2, 3\}$ . При данном

сочетании весовых коэффициентов каждый вычет формируется по 4 раза. Для других же сочетаний весовых коэффициентов вычеты формируют неравномерно, что и приводит к уменьшению значения коэффициента  $\xi_{m,k}$ .

При построении  $WSM(m,k)$ -кода могут быть подобраны такие сочетания весовых коэффициентов, которые позволят построить код с наименьшим общим числом необнаруживаемых ошибок конкретного вида (монотонных, симметричных и асимметричных). Из табл. 4, например, следует, что  $WS4(4,2)$ -код будет иметь наименьшее количество необнаруживаемых монотонных ошибок при значениях весовых коэффициентов  $[1, 1, 1, 1]$  и  $[3, 3, 3, 3]$ :  $\upsilon_{m,k} = 3,571 \%$ . При весовых коэффициентах  $[1, 1, 1, 1]$ ,  $[1, 1, 3, 3]$ ,  $[1, 2, 2, 2]$ ,  $[2, 2, 2, 3]$  и  $[3, 3, 3, 3]$   $WS4(4,2)$ -код обнаруживает все асимметричные ошибки:  $\alpha_{m,k} = 0 \%$ .

Возможность выбора способа взвешивания разрядов при построении кода на практике может оказаться полезной, например, при организации схем встроенного контроля для устройств с заранее известной структурой и с конкретным распределением возможных ошибок на их выходах или для синтеза устройств с контролепригодными структурами. При этом, естественно, должна учитываться рассматриваемая модель неисправностей.

Могут быть установлены и более сложные закономерности в характеристиках  $WSM(m,k)$ -кодов, наблюдающиеся при варьировании весов.

### 3. Синтез кодеров взвешенных кодов с суммированием

Кодеры для  $WSM(m,k)$ -кодов могут быть синтезированы непосредственно по таблице истинности, отражающей зависимость между векторами  $\langle f_m f_{m-1} \dots f_2 f_1 \rangle$  и  $\langle g_k g_{k-1} \dots g_2 g_1 \rangle$ . Однако такой способ эффективен только при малой длине информационных векторов. Другим вариантом, эффективным при синтезе кодеров, например, на программируемых логических интегральных схемах (ПЛИС), является запись каждой функции в дизъюнктивной совершенной нормальной форме (ДСНФ) [31] и реализация кодера в виде двухуровневой схемы.

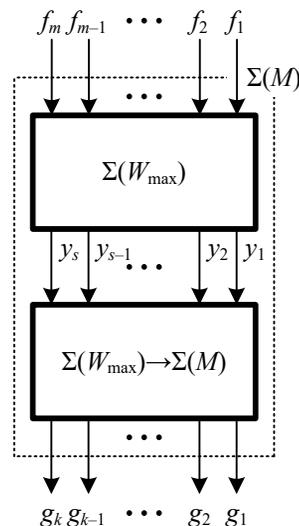
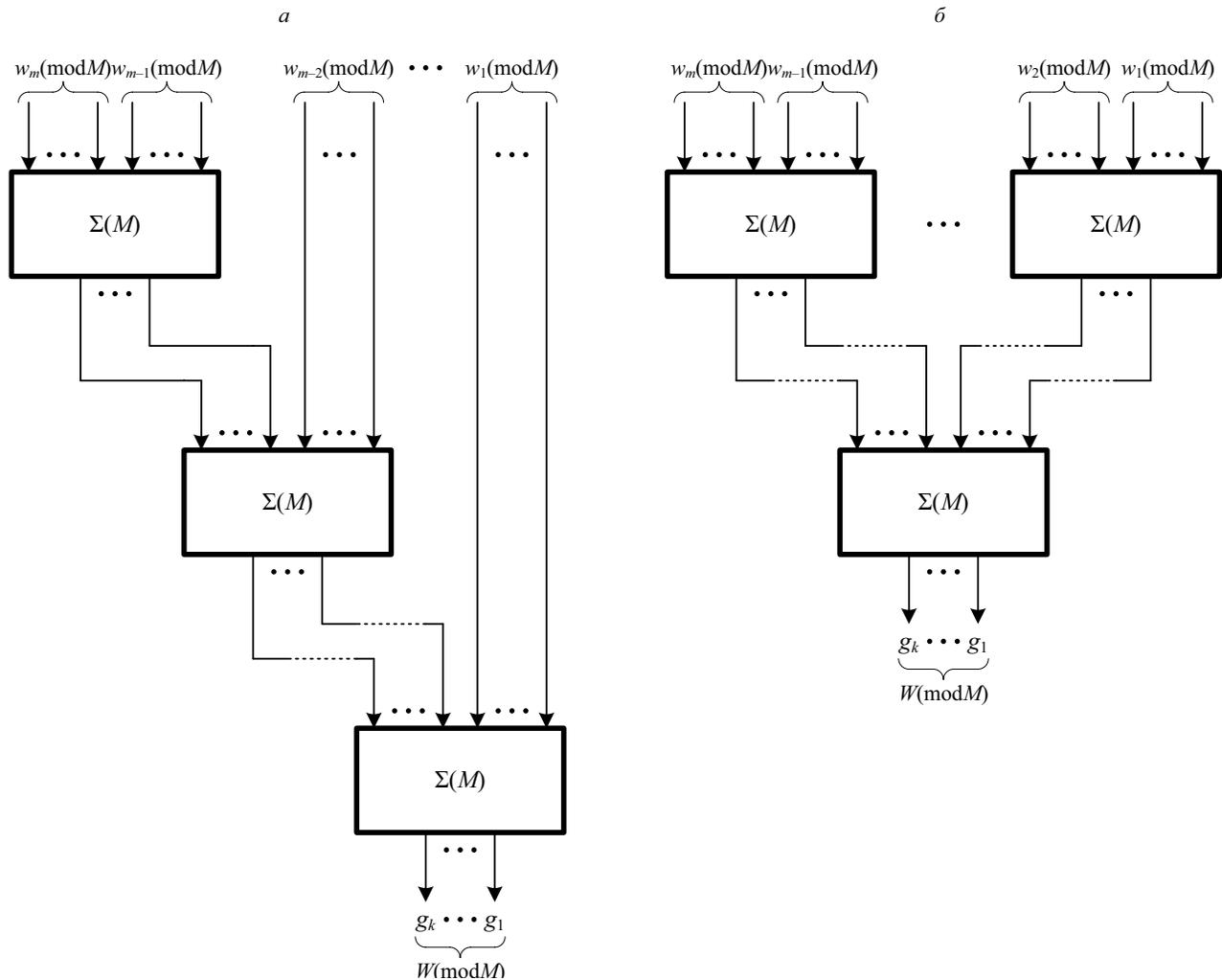


Рис. 2. Обобщенная структура кодеров  $WSM(m,k)$ -кодов

Возможны и другие подходы к синтезу кодеров  $WSM(m,k)$ -кодов. Может использоваться способ реализации кодеров  $WSM(m,k)$ -кодов на основе структуры, представленной на рис. 2. Данная структура подразумевает синтез кодера в виде каскадного соединения двух устройств — сумматора по модулю  $M = W_{\max}$  (устройства  $\Sigma(W_{\max})$ ) и преобразователя получаемого двоичного числа в двоичное число, соответствующее вычету по установленному модулю  $M$  (устройства  $\Sigma(W_{\max}) \rightarrow \Sigma(M)$ ). Особенности синтеза подобных устройств описаны в работах [32–35]. Сумматор по модулю  $M = W_{\max}$  представляет собой кодер взвешенного кода с суммированием в кольце вычетов по максимальному модулю и синтезируется любым из известных методов [12]. Он осуществляет преобразование входного вектора  $\langle f_m f_{m-1} \dots f_2 f_1 \rangle$  в вектор  $\langle y_s y_{s-1} \dots y_2 y_1 \rangle$ , где  $s = \lceil \log_2 (W_{\max} + 1) \rceil$ . Устройство  $\Sigma(W_{\max}) \rightarrow \Sigma(M)$  преобразует вектор  $\langle y_s y_{s-1} \dots y_2 y_1 \rangle$  в вектор  $\langle g_k g_{k-1} \dots g_2 g_1 \rangle$ , где  $k = \lceil \log_2 M \rceil$ .

Другой способ реализации кодера вытекает из особенностей представления вычетов весовых коэффициентов:

$$\begin{aligned} (w_1 + w_2 + \dots + w_{m-1} + w_m) \pmod{M} &= \\ &= w_1 \pmod{M} + w_2 \pmod{M} + \dots + \\ &+ w_{m-1} \pmod{M} + w_m \pmod{M}. \end{aligned}$$



**Рис. 3.** Структуры кодеров  $WSM(m,k)$ -кодов, реализованных в виде сумматоров двоичных чисел по заданному модулю:

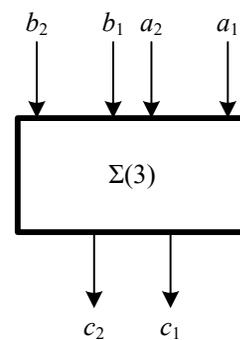
а) структура с последовательным соединением блоков; б) структура с минимальным числом уровней

Из представленного выражения непосредственно следует, что кодер может быть реализован в виде соединения функциональных блоков сложения двоичных чисел по заданному модулю (рис. 3).

Сумматоры в кольце вычетов по заданному модулю, реализованные в базе простейших функциональных элементов, широко известны в литературе [36–41]. Для реализации таких модулей на программируемой элементной базе можно синтезировать устройства сложения двоичных чисел по заданному модулю в виде преобразователей кодовых слов. Покажем данный процесс на примере реализации сумматора по модулю  $M = 3$ .

Множество наименьших неотрицательных вычетов по модулю  $M = 3$  включает в себя следующие числа  $\{0,1,2\}$ . Эти числа в двоичной

логике представляются с использованием двух разрядов. Таким образом, сумматор по модулю  $M = 3$  (устройство  $\Sigma(3)$ ) складывает два двухразрядных двоичных числа  $A = \langle a_2 a_1 \rangle$  и  $B = \langle b_2 b_1 \rangle$  и на выходе формирует двоичное число  $C = \langle c_2 c_1 \rangle$  (рис. 4). Данный сумматор реализуется непосредственно по табл. 9.



**Рис. 4.** Сумматор по модулю  $M = 3$

Таблица 9. Описание работы  $\Sigma(3)$

$b_1$	$b_2$	$a_2$	$a_1$	$\Sigma$	$\Sigma(3)$	$c_2$	$c_1$	$E$
0	0	0	0	0	0	0	0	
0	0	0	1	1	1	0	1	
0	0	1	0	2	2	1	0	
0	0	1	1	3	0	0	0	*
0	1	0	0	1	1	0	1	
0	1	0	1	2	2	1	0	
0	1	1	0	3	0	0	0	
0	1	1	1	4	1	0	1	*
1	0	0	0	2	2	1	0	
1	0	0	1	3	0	0	0	
1	0	1	0	4	1	0	1	
1	0	1	1	5	2	1	0	*
1	1	0	0	3	0	0	0	*
1	1	0	1	4	1	0	1	*
1	1	1	0	5	2	1	0	*
1	1	1	1	6	0	0	0	*

В таблице представлены все значения двоичных чисел  $A$  и  $B$ , их сумма в десятичном виде в столбце  $\Sigma$  и сумма в кольце вычетов по модулю  $M = 3$  (столбец  $\Sigma(3)$ ), а также число  $C$  в двоичном виде. Так как числа  $A = B = \langle 11 \rangle$  не подаются на входы сумматора, то на этих входных комбинациях значения разрядов числа  $C$  могут быть безразличными. В последнем столбце таблицы (столбец  $E$ ) значения числа  $C$  на наборах, включающих в себя числа  $A = B = \langle 11 \rangle$ , отмечены знаком «\*». Функции, описывающие разряды числа  $C$ , получаются непосредственно в виде ДСНФ либо путем минимизации с учетом возможностей использования безразличных состояний. Сумматор реализуется в выбранном базисе функциональных элементов.

Для реализации сумматора по модулю  $M = 3$  на ПЛИС используем Quartus Prime [42]. Выписывая из таблицы ДСНФ (не учитывая безразличные состояния) и реализуя функции графически, получаем структуру сумматора, представленную на рис. 5 (скриншот с Quartus Prime). Альтернативным способом реализации сумматора является использование специальных языков описания, таких как Verilog HDL, VHDL и другие [43, 44]. В данной работе рассмотрим описание схемы с помощью языка SystemVerilog. Так как схема небольшая, используем описание

в виде функций алгебры логики. Листинг программы выглядит следующим образом:

```

module sum (output c1, c2, input a1,a2, b1, b2);
assign c1 = b1&&!b2&&!a1&&!a2 ||
b1&&!b2&&a1&&a2 || b1&&b2&&!a1&&a2 ||
!b1&&b2&&a1&&!a2 || !b1&&!b2&&!a1&&a2;
assign c2 = b1&&!b2&&!a1&&a2 ||
b1&&b2&&a1&&!a2 || !b1&&b2&&a1&&a2 ||
!b1&&b2&&!a1&&!a2 || !b1&&!b2&&a1&&!a2;
endmodule
    
```

Для оценки сложности реализации сумматора по модулю  $M = 3$  была использована программируемая логическая интегральная схема семейства MAX II типа EPM240F100C4 [45]. Описание данного устройства можно найти на сайте [46]. Были получены следующие результаты: количество занимаемых ячеек логических элементов — 2, количество занимаемых портов входа/выхода — 6; всего ячеек — 8. Аналогичный результат получается при учете безразличных состояний.

Диаграмма на рис. 6 демонстрирует корректность работы сумматора по модулю  $M = 3$ .

#### 4. Эксперименты с контрольными схемами

В ходе исследований были проведены эксперименты по обнаружению ошибок на выходах тестовых комбинационных схем из набора LGSynth'89 [47, 48] при внесении одиночных константных неисправностей (stuck-at faults) на выходы логических элементов в их структурах.

Эксперимент проводился следующим образом. Выбирался  $WSM(m,k)$ -код с конкретной последовательностью весовых коэффициентов  $[w]$ . Модуль выбирался последовательно из множества  $M \in \{2, 3, \dots, W_{\max} + 1\}$ , а значения весовых коэффициентов  $w_i, i = 1, m$  — из множества  $w_i \in \{1, 2, \dots, M - 1\}$ . Для каждого кода выполнялись такие процедуры. При внесении каждой одиночной неисправности на входы подавалось полное множество входных комбинаций и определялось наличие/отсутствие ошибки в выходном векторе, а также, при ее наличии, кратность и вид. Ошибки классифицировались на обнаруживаемые и необнаруживаемые выбранным кодом. Затем определялось общее количество необнаруживаемых ошибок по видам и кратностям.

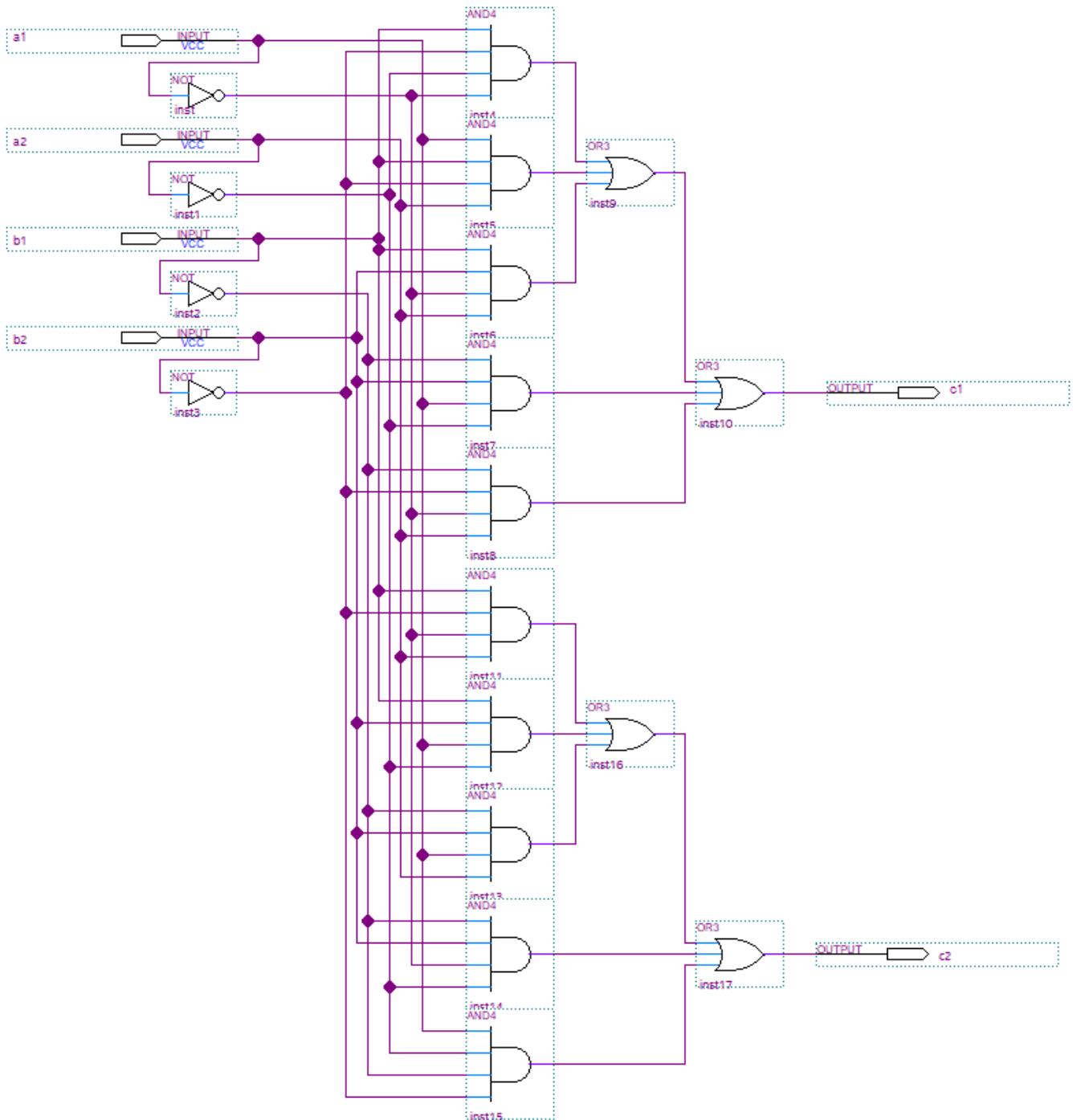


Рис. 5. Схема сумматора по модулю  $M = 3$ , реализованная в Quartus Prime

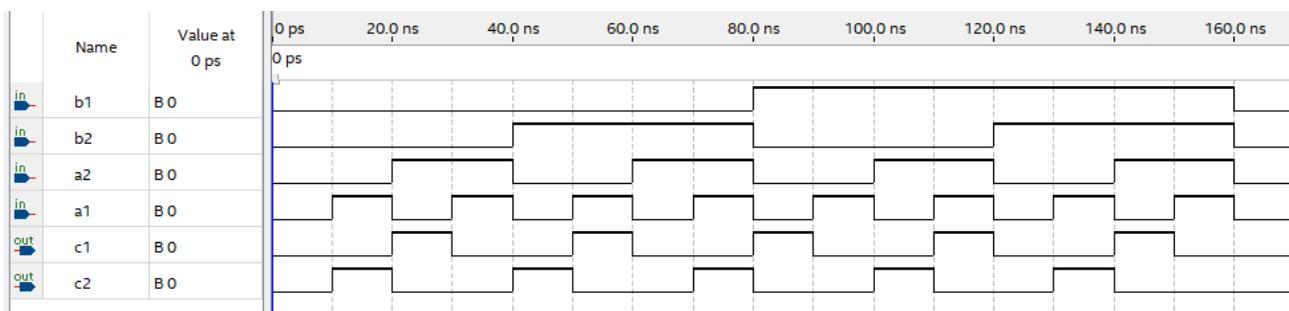
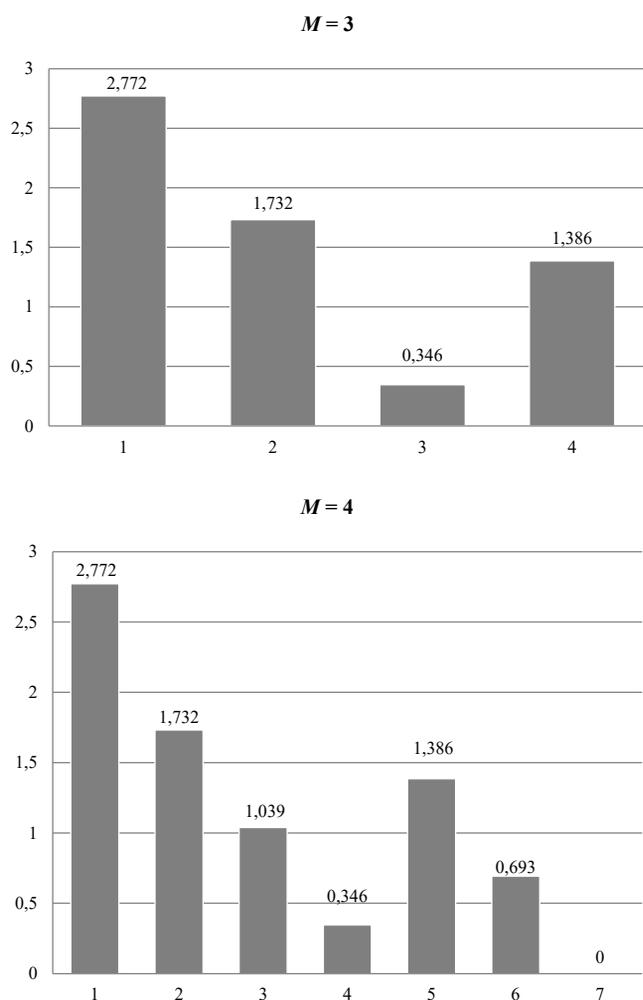


Рис. 6. Диаграмма работы сумматора по модулю  $M = 3$



**Рис. 7.** Доли необнаруживаемых ошибок на выходах схемы «z4ml» от общего их количества

Были рассмотрены различные комбинационные схемы, записанные в формате \*.netblif. В данном формате схемы реализованы на элементах NOR с числом входов от 1 до 4. Результаты экспериментов для различных схем схожи. В табл. 10 приведены характеристики обнаружения ошибок WSM(m,k)-кодами на выходах схемы «z4ml». Она снабжена 7-ю входами и 4-мя выходами. В формате \*.netblif она содержит 70 логических элементов. Общее количество ошибок на ее выходах равно 4618, включая 4008 монотонных, 128 двукратных симметричных и 32 трехкратных асимметричных ошибки. Различные WSM(m,k)-коды по-разному обнаруживают данные ошибки на выходах. Максимальное число необнаруживаемых ошибок равно 128, что составляет 2,772 % от общего их количества (показатель  $\gamma$ ). Использование различных WSM(m,k)-кодов с различными

значениями модулей и последовательностями весовых коэффициентов позволяет снизить эту долю (рис. 7). В табл. 10 также приведены показатели  $\nu$ ,  $\sigma$ ,  $\alpha$  и  $\nu^*$ ,  $\sigma^*$ ,  $\alpha^*$ . Первые характеризуют доли необнаруживаемых монотонных, симметричных и асимметричных ошибок от общего числа необнаруживаемых ошибок, а вторые — доли необнаруживаемых монотонных, симметричных и асимметричных ошибок от общего числа ошибок соответствующего вида. По данным показателям можно определить, за счет каких видов ошибок достигается эффект в применении конкретного WSM(m,k)-кода при организации контроля схемы.

Из данных табл. 10 и рис. 7 следует, что ряд способов построения WSM(m,k)-кодов позволяет обнаруживать любые ошибки на выходах рассматриваемой схемы. Для этого требуется установить модуль  $M = 4$  и выбрать один из следующих способов взвешивания разрядов: [1, 1, 3, 2], [1, 3, 1, 2], [2, 1, 3, 2], [2, 3, 1, 2], [3, 1, 3, 2], [3, 3, 1, 2]. Другие сочетания весовых коэффициентов и значений модуля не дают возможности идентификации полного множества ошибок на выходах рассматриваемой схемы. Тем не менее показатель  $\gamma$  оказывается крайне низким и для ряда способов построения взвешенного кода составляет менее 2 %.

Выбор возможности взвешивания для обнаружения ошибок на выходах цифровых устройств позволяет на практике выбирать наилучший способ с учетом и критерия вносимой структурной избыточности. Это требует также определения структурной избыточности схемы встроенного контроля, которая в рассматриваемом примере не синтезировалась.

Следует отметить, что при различных модулях существует возможность обнаружения наибольшего числа ошибок на выходах схем, а сам выбор модуля и значений весовых коэффициентов индивидуален для каждого контролируемого устройства.

### Заключение

При построении кодов с суммированием могут быть использованы различные варианты взвешивания разрядов. При этом для ограничения числа контрольных разрядов могут применяться принципы модулярной

Таблица 10. Характеристики обнаружения ошибок в контрольной схеме «z4m!»

Номер семейства	[w]	Число необнаруживаемых ошибок по видам				Всего	Показатели обнаружения ошибок, %								
		Монотонных	Симметричных, d = 2		Асимметричных, d = 3		u	σ	α	γ	U*	σ*	α*		
			M = 2												
1	1111	0	128	0	128	0	100	0	2,772	0	100	0	0	0	0
<b>M = 3</b>															
1	[1,1,1,1], [1,2,2,2], [2,1,1,1], [2,2,2,2]	0	128	0	128	0	100	0	2,772	0	100	0	0	0	0
2	[1,1,1,2], [1,2,2,1], [2,1,1,2], [2,2,2,1]	0	48	32	80	0	60	40	1,732	0	37,5	100	0	0	0
3	[1,1,2,1], [1,2,1,2], [2,1,2,1], [2,2,1,2]	0	16	0	16	0	100	0	0,346	0	12,5	0	0	0	0
4	[1,1,2,2], [1,2,1,1], [2,1,2,2], [2,2,1,1]	0	64	0	64	0	100	0	1,386	0	50	0	0	0	0
<b>M = 4</b>															
1	[1,1,1,1], [1,2,2,2], [1,3,3,3], [2,1,1,1], [2,2,2,2], [2,3,3,3], [3,1,1,1], [3,2,2,2], [3,3,3,3]	0	128	0	128	0	100	0	2,772	0	100	0	0	0	0
2	[1,1,1,2], [1,3,3,2], [2,1,1,2], [2,3,3,2], [3,1,1,2], [3,3,3,2]	0	48	32	80	0	60	40	1,732	0	37,5	100	0	0	0
3	[1,1,1,3], [1,2,2,1], [1,2,2,3], [1,3,3,1], [2,1,1,3], [2,2,2,1], [2,2,2,3], [2,3,3,1], [3,1,1,3], [3,2,2,1], [3,2,2,3], [3,3,3,1]	0	48	0	48	0	100	0	1,039	0	37,5	0	0	0	0
4	[1,1,2,1], [1,1,3,1], [1,2,1,2], [1,2,3,2], [1,3,1,3], [1,3,2,3], [2,1,2,1], [2,1,3,1], [2,2,1,2], [2,2,3,2], [2,3,1,3], [2,3,2,3], [3,1,2,1], [3,1,3,1], [3,2,1,2], [3,2,3,2], [3,3,1,3], [3,3,2,3]	0	16	0	16	0	100	0	0,346	0	12,5	0	0	0	0
5	[1,1,2,2], [1,1,3,3], [1,2,1,1], [1,2,3,3], [1,3,1,1], [1,3,2,2], [2,1,2,2], [2,1,3,3], [2,2,1,1], [2,2,3,3], [2,3,1,1], [2,3,2,2], [3,1,2,2], [3,1,3,3], [3,2,1,1], [3,2,3,3], [3,3,1,1], [3,3,2,2]	0	64	0	64	0	100	0	1,386	0	50	0	0	0	0
6	[1,1,2,3], [1,2,1,3], [1,2,3,1], [1,3,2,1], [2,1,2,3], [2,2,1,3], [2,2,3,1], [2,3,2,1], [3,1,2,3], [3,2,1,3], [3,2,3,1], [3,3,2,1]	0	0	32	32	0	0	100	0,693	0	0	100	0	0	100
7	[1,1,3,2], [1,3,1,2], [2,1,3,2], [2,3,1,2], [3,1,3,2], [3,3,1,2]	0	0	0	0	0	0	0	0	0	0	0	0	0	0

арифметики. Значение модуля при формировании кода может выбираться из множества  $M \in \{2, 3, \dots, W_{\max} + 1\}$ , а значения весовых коэффициентов — из множества  $w_i \in \{1, 2, \dots, M - 1\}$ . Исследования показывают, что такие ограничения позволяют строить огромное количество кодов с суммированием с конкретным фиксированным числом контрольных разрядов. Данные коды могут эффективно использоваться при решении большого числа задач, в том числе обеспечения контролепригодности структур цифровых устройств и обнаружения ошибок в вычислениях блоками и составляющими вычислительных систем.

Подбор весовых коэффициентов дает возможность построения  $WSM(m, k)$ -кодов с различными характеристиками обнаружения ошибок, в том числе получать коды с максимальным смещением числа необнаруживаемых ошибок в сторону большей их кратности, коды с обнаружением полного множества монотонных ошибок и коды с малым количеством необнаруживаемых монотонных ошибок, коды с обнаружением полного множества ошибок с нечетными кратностями и т. д. При этом могут устанавливаться ограничения на избыточность кода, на сложность кодирующего оборудования и условия обеспечения его самопроверяемости.

Одним из перспективных приложений  $WSM(m, k)$ -кодов является использование их при реализации цифровых устройств, наделяемых свойствами контролепригодности, самопроверяемости и отказоустойчивости [49, 50]. Здесь интересным может быть исследование методов синтеза схем встроенного контроля, а также методов реализации устройств в виде контролепригодных конечных автоматов при учете различных моделей неисправностей. Решению этих и других задач посвящены дальнейшие исследования авторов. ▲

### Библиографический список

1. Сапожников В. В. Методы построения безопасных микроэлектронных систем железнодорожной автоматики / В. В. Сапожников, Вл. В. Сапожников, Х. А. Христов, Д. В. Гавзов; под ред. Вл. В. Сапожникова. — М.: Транспорт, 1995. — 272 с.
2. Ubar R. Design and Test Technology for Dependable Systems-on-Chip (Premier Reference Source) / R. Ubar, J. Raik, H.-T. Vierhaus. — Information Science Reference, Hershey. — New York, IGI Global, 2011. — 578 p.
3. Drozd O. Particularities of Sync Monitoring in FPGA Components of Safety-Related Systems / O. Drozd, A. Sachenko, R. Hiromoto et al. // Proceedings of 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS'2021). — Vol. 2. — Cracow, Poland, September 22–25. — 2021. — Pp. 979–983.
4. Hahanov V. Vector Simulation of Logic Faults Based on XOR-Relations / V. Hahanov, S. Chumachenko, E. Litviniva, H. Khakhanova // Proceedings of 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS'2021). — Vol. 2. — Cracow, Poland, September 22–25. — 2021. — Pp. 1041–1044.
5. Сапожников Вл. В. Синтез систем управления движением поездов на железнодорожных станциях с исключением опасных отказов / Вл. В. Сапожников. — М.: Наука, 2021. — 229 с.
6. McCluskey E. J. Logic Design Principles: With Emphasis on Testable Semicustom Circuits / E. J. McCluskey. — New Jersey: Prentice Hall PTR, 1986. — 549 p.
7. Abramovici M. Digital System Testing and Testable Design / M. Abramovici, M. A. Breuer, A. D. Friedman. — Computer Science Press, 1998. — 652 p.
8. Lala P. K. Self-Checking and Fault-Tolerant Digital Design / P. K. Lala. — San Francisco: Morgan Kaufmann Publishers. — 2001. — 216 p.
9. Fujiwara E. Code Design for Dependable Systems: Theory and Practical Applications / E. Fujiwara // John Wiley & Sons. — 2006. — 720 p.
10. Göessel M. New Methods of Concurrent Checking: Edition 1 / M. Göessel, V. Ocheretny, E. Sogomonyan, D. Marienfeld. — Dordrecht: Springer Science+Business Media B.V. — 2008. — 184 p.
11. Сапожников В. В. Коды с суммированием для систем технического диагностирования / В. В. Сапожников, Вл. В. Сапожников, Д. В. Ефанов. — Т. 1: Классические коды Бергера и их модификации. — М.: Наука, 2020. — 383 с.
12. Сапожников В. В. Коды с суммированием для систем технического диагностирования / В. В. Сапожников, Вл. В. Сапожников, Д. В. Ефанов. — Т. 2: Взвешенные коды с суммированием. — М.: Наука, 2021. — 455 с.
13. Berger J. M. A Note on Error Detection Codes for Asymmetric Channels / J. M. Berger // Information and Control. — 1961. — Vol. 4. — Iss. 1. — Pp. 68–73. — DOI: 10.1016/S0019-9958(61)80037-5.
14. Berger J. M. A Note on Burst Detection Sum Codes / J. M. Berger // Information and Control. — 1961. — Vol. 4. — Iss. 2–3. — Pp. 297–299. — DOI: 10.1016/S0019-9958(61)80024-7.
15. Das D. Weight-Based Codes and Their Application to Concurrent Error Detection of Multilevel Circuits / D. Das, N. A. Touba // Proceedings of 17th IEEE Test Symposium. — California, USA. — 1999. — Pp. 370–376. — DOI: 10.1109/VTEST.1999.766691.
16. Das D. Low Cost Concurrent Error Detection Based on Modulo Weight-Based Codes / D. Das, N. A. Touba, M. Seuring, M. Gossel // Proceedings of the IEEE 6th International On-Line Testing Workshop (IOLTW). — Spain, Palma de Mallorca. — July 3–5, 2000. — Pp. 171–176. — DOI: 10.1109/OLT.2000.856633.

17. *Согомонян Е. С.* Самопроверяемые устройства и отказоустойчивые системы / Е. С. Согомонян, Е. В. Слабаков. — М.: Радио и связь, 1989. — 208 с.
18. *Sogomonyan E. S.* Design of Self-Testing and On-Line Fault Detection Combinational Circuits with Weakly Independent Outputs / E. S. Sogomonyan, M. Gössel // *Journal of Electronic Testing: Theory and Applications*. — 1993. — Vol. 4. — Iss. 4. — Pp. 267–281. — DOI: 10.1007/BF00971975.
19. *Busaba F. Y.* Self-Checking Combinational Circuit Design for Single and Unidirectional Multibit Errors / F. Y. Busaba, P. K. Lala // *Journal of Electronic Testing: Theory and Applications*. — 1994. — Vol. 5. — Iss. 5. — Pp. 19–28. — DOI: 10.1007/BF00971960.
20. *Morosow A.* Self-Checking Combinational Circuits with Unidirectionally Independent Outputs / A. Morosow, V. V. Sapozhnikov, V. V. Sapozhnikov, M. Goessel // *VLSI Design*. — 1998. — Vol. 5. — Iss. 4. — Pp. 333–345. — DOI: 10.1155/1998/20389.
21. *Matrosova A. Yu.* Self-Checking Synchronous FSM Network Design with Low Overhead / A. Yu. Matrosova, I. Levin, S. A. Ostanin // *VLSI Design*. — 2000. — Vol. 11. — Iss. 1. — Pp. 47–58. — DOI: 10.1155/2000/46578.
22. *Efanov D. V.* Organization of a Fully Self-Checking Structure of a Combinational Device Based on Searching for Groups of Symmetrically Independent Outputs / D. V. Efanov, V. V. Sapozhnikov, V. V. Sapozhnikov // *Automatic Control and Computer Sciences*. — 2020. — Vol. 54. — Iss. 4. — Pp. 279–290. — DOI: 10.3103/S0146411620040045.
23. *Сапожников В. В.* Взвешенные коды с суммированием для организации контроля логических устройств / В. В. Сапожников, Вл. В. Сапожников, Д. В. Ефанов // *Электронное моделирование*. — 2014. — Т. 36. — № 1. — С. 59–80.
24. *Efanov D. V.* Using Codes with Summation of Weighted Bits to Organize Checking of Combinational Logical Devices / D. V. Efanov, V. V. Sapozhnikov, V. V. Sapozhnikov // *Automatic Control and Computer Sciences*. — 2019. — Vol. 53. — Iss. 1. — Pp. 1–11. — DOI: 10.3103/S0146411619010061.
25. *Дмитриев В. В.* О двух способах взвешивания и их влиянии на свойства кодов с суммированием взвешенных переходов в системах функционального контроля логических схем / В. В. Дмитриев // *Известия Петербургского университета путей сообщения*. — 2015. — № 3. — С. 119–129.
26. *Mehov V.* Concurrent Error Detection Based on New Code with Modulo Weighted Transitions between Information Bits / V. Mehov, V. Sapozhnikov, V. Sapozhnikov, D. Urganskov // *Proceedings of 7th IEEE East-West Design & Test Workshop (EWDTW'2007)*. — Erevan, Armenia. — September 25–30, 2007. — Pp. 21–26.
27. *Мехов В. В.* Контроль комбинационных схем на основе модифицированных кодов с суммированием / В. В. Мехов, В. В. Сапожников, Вл. В. Сапожников // *Автоматика и телемеханика*. — 2008. — № 8. — С. 153–165.
28. *Efanov D.* On One Method of Formation of Optimum Sum Code for Technical Diagnostics Systems / D. Efanov, V. Sapozhnikov, V. Sapozhnikov // *Proceedings of 14th IEEE East-West Design & Test Symposium (EWDT'S'2016)*. — Yerevan, Armenia. — October 14–17, 2016. — Pp. 158–163. — DOI: 10.1109/EWDT'S.2016.7807633.
29. *Ефанов Д. В.* Коды с суммированием с фиксированными значениями кратностей, обнаруживаемых монотонных и асимметричных ошибок для систем технического диагностирования / Д. В. Ефанов, В. В. Сапожников, Вл. В. Сапожников // *Автоматика и телемеханика*. — 2019. — № 6. — С. 121–141.
30. *Efanov D. V.* The Weight-Based Sum Codes in the Residue Ring by Arbitrary Modulus for Synthesis of Self-Checking Digital Computing Systems / D. V. Efanov, A. V. Pashukov // *Proceedings of 19th IEEE East-West Design & Test Symposium (EWDT'S'2021)*. — Batumi, Georgia. — September 10–13, 2021. — Pp. 170–179. — DOI: 10.1109/EWDT'S52692.2021.9581032.
31. *Закревский А. Д.* Логические основы проектирования дискретных устройств / А. Д. Закревский, Ю. В. Поттосин, Л. Д. Черемисинова. — М.: Физматлит, 2007, 592 с.
32. *Сапожников В. В.* Универсальные структуры двоичных счетчиков единиц по произвольному модулю счета / В. В. Сапожников, Вл. В. Сапожников, Д. И. Ургансков // *Электронное моделирование*. — 2002. — Т. 24. — № 4. — С. 65–81.
33. *Сапожников В. В.* Блочная структура двоичного счетчика единиц по произвольному модулю счета / В. В. Сапожников, Вл. В. Сапожников, Д. И. Ургансков // *Электронное моделирование*. — 2005. — Т. 27. — № 4. — С. 65–81.
34. *Sapozhnikov V. V.* Composite Structure of Binary Counter of Ones Arbitrary Modulo / V. V. Sapozhnikov, V. V. Sapozhnikov, D. I. Urganskov // *Proceedings of East-West Design & Test Workshop (EWDTW'05)*. — 15–19 September 2005. — Odessa, Ukraine. — Pp. 102–106.
35. *Sapozhnikov V. V.* Multistage Regular Structure of Binary Counter of Ones Arbitrary Modulo / V. V. Sapozhnikov, V. V. Sapozhnikov, D. I. Urganskov // *Proceedings of East-West Design & Test Workshop (EWDTW'06)*. — 15–19 September 2006. — Sochi, Russia. — Pp. 287–290.
36. *Супрун В. П.* Сумматор по модулю пять. — Авторское свидетельство СССР № 1388850, SU 1803911 A1 / В. П. Супрун, А. Л. Дорожинский. — 1986. — 3 с.
37. *Авгуль Л. Б.* Сумматор по модулю семь / Л. Б. Авгуль. — Авторское свидетельство РФ RU 2028660 C1, опубликовано 09.02.1995, подано 21.05.1992. — 7 с.
38. *Suprun V. P.* Realization of Addition and Multiplication Operations in Unitary Codes / V. P. Suprun, D. A. Gorodecky // *Automatic Control and Computer Sciences*. — 2010. — Vol. 44. — Iss. 5. — Pp. 292–301. — DOI: 10.3103/S014641161005007X.
39. *Suprun V. P.* Single-Level Schematic Realization of Basic Operations of Modular Arithmetic in Unitary Codes / V. P. Suprun // *Automatic Control and Computer Sciences*. — 2011. — Vol. 45. — Iss. 2. — Pp. 70–79. — DOI: 10.3103/S0146411611020088.
40. *Дрозд А. В.* Рабочее диагностирование безопасных информационно-управляющих систем / А. В. Дрозд, В. С. Харченко, С. Г. Антошук и др.; под ред. А. В. Дрозда и В. С. Харченко. — Харьков: Национальный аэрокосмический университет им. Н. Е. Жуковского «ХАИ», 2012. — 614 с.
41. *Балака Е. С.* Разработка высокоскоростных сумматоров по модулю на базе комбинационных сумматоров с параллельным переносом / Е. С. Балака, Д. А. Городецкий, В. С. Рухлов, А. Н. Щелоков // *Известия ЮФУ. Технические науки*. — 2016. — № 6. — С. 158–169.
42. *Intel® Quartus® Prime Download* — Intel® Quartus® Prime Software. — URL: <https://www.intel.ru/content/www/ru/ru/software/programmable/quartus-prime/download.html> (дата обращения: 08.11.2021).

43. Хаханов В. И. Проектирование и тестирование цифровых систем на кристаллах / В. И. Хаханов, Е. И. Литвинова, О. А. Гузь. — Харьков: ХНУРЭ, 2009. — 484 с.
44. Navabi Z. Digital System Test and Testable Design: Using HDL Models and Architectures / Z. Navabi. — Springer Science+Business Media, LLC 2011. — 435 p.
45. MAX II Device Handbook. — Vol. 1. — Altera Corporation, 2007. — 107 p.
46. ПЛИС семейства MAX II. — URL: <http://altera.ru/plis-max-II.html> (дата обращения: 08.11.2021).
47. Collection of Digital Design Benchmarks. — URL: <http://ddd.fit.cvut.cz/www/prj/Benchmarks/>.
48. Sentovich E. M. SIS: A System for Sequential Circuit Synthesis / E. M. Sentovich, K. J. Singh, L. Lavagno, C. Moon et al. // Electronics Research Laboratory, Department of

- Electrical Engineering and Computer Science. — University of California, Berkeley, 4 May 1992. — 45 p.
49. Drozd A. Checkability of the Digital Components in Safety-Critical Systems: Problems and Solutions / A. Drozd, V. Kharchenko, S. Antoshchuk et al. // Proceedings of 9th IEEE East-West Design & Test Symposium (EWDTS'2011). — Sevastopol, Ukraine, 2011. — Pp. 411–416. — DOI: 10.1109/EWDTS.2011.6116606.
  50. Drozd O. Resilient Development of Models and Methods in Computing Space / O. Drozd, A. Rucinski, K. Zashcholkin et al. // Proceedings of 19th IEEE East-West Design & Test Symposium (EWDTS'2021). — Batumi, Georgia, September 10–13 2021. — Pp. 70–75. — DOI: 10.1109/EWDTS52692.2021.9581002.

TRANSPORT AUTOMATION RESEARCH, 2022, Vol. 8, No. 2, pp. 198–217  
DOI: 10.20295/2412-9186-2022-8-02-198-217

### Codes With Summation of Weight Coefficients of Data Vector Bits in the Ring of Residues Modulo an Arbitrary for Digital Computing Devices

#### Information about authors

**Efanov D. V.**, Doctor of Science, Assistant Professor, IEEE (Institute of Electrical and Electronics Engineers) Member, Full-member of the International Transport Academy, General Director Deputy on Scientific Research Work<sup>1</sup>, Professor<sup>2,3</sup>. E-mail: [TrES-4b@yandex.ru](mailto:TrES-4b@yandex.ru)

**Pashukov A. V.**, Senior Lecturer, PhD Student<sup>3</sup>.  
E-mail: [art\\_pash@mail.ru](mailto:art_pash@mail.ru)

<sup>1</sup>Scientific-Technical Center Integrated Monitoring Systems Ltd, Saint Petersburg

<sup>2</sup>Peter the Great Saint Petersburg Polytechnical University, Transport Higher School, Mechanical Engineering. Material and Transport Institute, Saint Petersburg

<sup>3</sup>Russian University of Transport (MIIT), Department of Automation, Remote Control and Communication on Railway Transport, Moscow

**Abstract:** We consider the specificities of codes with the summation of weight coefficients of data vector bits in the ring of residues modulo an arbitrary. There're established the characteristics of error detection by the given code class and some legitimacies related to the use of modules of particular values and of weight coefficients. It's shown that the number of various weight-based sum codes is limited, nevertheless, large number of various ways of their formation exists for each value of bit number in data vectors. There're given the algorithm to obtain bits of check vectors of being considered codes and the examples of their upbuilding; the way to calculate the number of not being revealed errors in data vectors is described. There're defined the properties of weight-based sum codes of data vectors in the ring of residues modulo an arbitrary accounting for which may be useful while the settlement of technical diagnostics task, the synthesis of being self-checking and fault-tolerant digital devices and devices with testability structures. The approach to the synthesis of coders of any weight-based sum codes on the base of binary number summators according to established module is described. There're given some results of experiments on the analysis of revealing capacities of being considered codes in the circuits of inbuilt control of combinational benchmarks. Obtained in the work results are of universal character and not oriented to application just with one element base of being realized devices that make them useful not just at the moment but in the future.

**Keywords:** fault-tolerant digital systems; testability devices; self-checking circuits of in-built control; sum code; weight-based sum codes; summation in the ring of residues with arbitrary modulo; error detection in data bits; properties of weight-based sum codes.

#### References

1. Sapozhnikov V. V., Sapozhnikov V. V., Khristov Kh. A., Gavzov D. V. *Metody postroeniya bezopasnykh mikroelektronnykh sistem zheleznodorozhnoy avtomatiki* [Methods for constructing safe microelectronic systems for railway automation]. Moscow: Transport Publ., 1995. 272 p. (In Russian)
2. Ubar R., Raik J., Vierhaus H.-T. Design and Test Technology for Dependable Systems-on-Chip (Premier Reference Source). Information Science Reference, Hershey. New York, IGI Global, 2011, 578 p.
3. Drozd O., Sachenko A., Hiromoto R., Zashcholkin K., Drozd M. Particularities of Sync Monitoring in FPGA Components of Safety-Related Systems. Proceedings of 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS'2021), vol. 2, Cracow, Poland, September 22–25, 2021, pp. 979–983.
4. Hahanov V., Chumachenko S., Litviniva E., Khakhanova H. Vector Simulation of Logic Faults Based on XOR-Relations. Proceedings of 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS'2021), vol. 2, Cracow, Poland, September 22–25, 2021, pp. 1041–1044.
5. Sapozhnikov V. V. *Sintez sistem upravleniya dvizheniem poezdov na zheleznodorozhnykh stantsiyakh s isklyucheniem opasnykh otkazov* [Synthesis of train traffic control systems at railway stations with the exception of dangerous failures]. Moscow: Nauka Publ., 2021. 229 p. (In Russian)
6. McCluskey E. J. Logic Design Principles: With Emphasis on Testable Semicustom Circuits. New Jersey: Prentice Hall PTR, 1986, 549 p.
7. Abramovici M., Breuer M.A., Friedman A.D. Digital System Testing and Testable Design. Computer Science Press, 1998, 652 p.
8. Lala P. K. Self-Checking and Fault-Tolerant Digital Design. San Francisco: Morgan Kaufmann Publishers, 2001, 216 p.
9. Fujiwara E. Code Design for Dependable Systems: Theory and Practical Applications. — John Wiley & Sons, 2006, 720 p.
10. 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.
11. Sapozhnikov V. V., Sapozhnikov V. V., Efanov D. V. *Kody s summirovaniem dlya sistem tekhnicheskogo diagnostirovaniya* [Summation codes for technical diagnostic systems]. *Klassicheskie kody Bergera i ikh modifikatsii* [Classic Berger codes and their modifications]. Moscow: Nauka Publ., 2020, vol. 1, 383 p. (In Russian)
12. Sapozhnikov V. V., Sapozhnikov V. V., Efanov D. V. *Kody s summirovaniem dlya sistem tekhnicheskogo diagnostirovaniya* [Summation codes for technical diagnostic systems]. *Vzveshennyye kody s summirovaniem* [Weighted Summation Codes]. Moscow: Nauka Publ., 2021, vol. 2, 455 p. (In Russian)
13. Berger J. M. A Note on Error Detection Codes for Asymmetric Channels. Information and Control. 1961, vol. 4, I. 1, pp. 68–73. DOI: 10.1016/S0019-9958(61)80037-5.
14. Berger J. M. A Note on Burst Detection Sum Codes. Information and Control. 1961, vol. 4, I. 2-3, pp. 297–299. DOI: 10.1016/S0019-9958(61)80024-7.
15. Das D., Touban N. A. Weight-Based Codes and Their Application to Concurrent Error Detection of Multilevel Circuits. Proceedings of 17th IEEE Test Symposium, California, USA, 1999, pp. 370–376. DOI: 10.1109/VTEST.1999.766691.

16. Das D., Toubia N. A., Seuring M., Gossel M. Low Cost Concurrent Error Detection Based on Modulo Weight-Based Codes. Proceedings of the IEEE 6th International On-Line Testing Workshop (IOLTW), Spain, Palma de Mallorca, July 3-5, 2000, pp. 171–176. DOI: 10.1109/IOLT.2000.856633.
17. Sogomonyan E. S., Slabakov E. V. *Samoproveryaemye ustroystva i otkazoustoychivye sistemy* [Self-checking devices and fault-tolerant systems]. Moscow: Radio i svyaz' Publ., 1989. 208 p. (In Russian)
18. Sogomonyan E. S., Gossel M. Design of Self-Testing and On-Line Fault Detection Combinational Circuits with Weakly Independent Outputs. Journal of Electronic Testing: Theory and Applications. 1993, vol. 4, I. 4, pp. 267–281. DOI: 10.1007/BF00971975.
19. Busaba F. Y., Lala P. K. Self-Checking Combinational Circuit Design for Single and Unidirectional Multibit Errors. Journal of Electronic Testing: Theory and Applications. 1994, vol. 5, I. 5, pp. 19–28. DOI: 10.1007/BF00971960.
20. Morosov A., Sapozhnikov V. V., Sapozhnikov V. V., Goessel M. Self-Checking Combinational Circuits with Unidirectionally Independent Outputs. VLSI Design. 1998, vol. 5, I. 4, pp. 333–345. DOI: 10.1155/1998/20389.
21. Matrosova A. Yu., Levin I., Ostanin S. A. Self-Checking Synchronous FSM Network Design with Low Overhead. VLSI Design. 2000, vol. 11, I. 1, pp. 47–58. DOI: 10.1155/2000/46578.
22. Efanov D. V., Sapozhnikov V. V., Sapozhnikov V. V. Organization of a Fully Self-Checking Structure of a Combinational Device Based on Searching for Groups of Symmetrically Independent Outputs. Automatic Control and Computer Sciences. 2020, vol. 54, I. 4, pp. 279–290. DOI: 10.3103/S0146411620040045.
23. Sapozhnikov V. V., Sapozhnikov V. V., Efanov D. V. Vzveshennyye kody s summirovaniem dlya organizatsii kontrolya logicheskikh ustroystv [Weighted codes with summation for organizing the control of logical devices]. *Elektronnoe modelirovaniye* [Electronic modeling]. 2014, vol. 36, I. 1, pp. 59–80. (In Russian)
24. Efanov D. V., Sapozhnikov V. V., Sapozhnikov V. V. Using Codes with Summation of Weighted Bits to Organize Checking of Combinational Logical Devices. Automatic Control and Computer Sciences. 2019, vol. 53, I. 1, pp. 1–11. DOI: 10.3103/S0146411619010061.
25. Dmitriev V. V. O dvukh sposobakh vzheshivaniya i ikh vliyaniy na svoystva kodov s summirovaniem vzheshennykh perekhodov v sistemakh funktsional'nogo kontrolya logicheskikh skhem [On two methods of weighting and their influence on the properties of codes with the summation of weighted transitions in functional control systems for logic circuits]. *Izvestiya Peterburgskogo universiteta putey soobshcheniya* [Izvestiya of the Petersburg University of Communications]. 2015, I. 3, pp. 119–129. (In Russian)
26. Mehov V., Sapozhnikov V., Sapozhnikov V. V., Urganov D. Concurrent Error Detection Based on New Code with Modulo Weighted Transitions between Information Bits. Proceedings of 7th IEEE East-West Design & Test Workshop (EWDTW'2007), Erevan, Armenia, September 25-30, 2007, pp. 21–26.
27. Mekhov V. B., Sapozhnikov V. V., Sapozhnikov V. V. Kontrol' kombinatsionnykh skhem na osnove modifitsirovannykh kodov s summirovaniem [Control of combinational circuits based on modified codes with summation]. *Avtomatika i telemekhanika* [Automation and Telemechanics]. 2008, I. 8, pp. 153–165. (In Russian)
28. Efanov D., Sapozhnikov V., Sapozhnikov V. V. On One Method of Formation of Optimum Sum Code for Technical Diagnostics Systems. Proceedings of 14th IEEE East-West Design & Test Symposium (EWDT'S'2016), Yerevan, Armenia, October 14-17, 2016, pp. 158–163, DOI: 10.1109/EWDT'S.2016.7807633.
29. Efanov D. V., Sapozhnikov V. V., Sapozhnikov V. V. Kody s summirovaniem s fiksirovannymi znacheniyami kratnostey, obnaruzhivaemykh monotonnykh i asimmetrichnykh oshibok dlya sistem tekhnicheskogo diagnostirovaniya [Summation codes with fixed multiplicity values, detectable monotonic and asymmetric errors for technical diagnostic systems]. *Avtomatika i telemekhanika* [Automation and Telemechanics]. 2019, I. 6, pp. 121–141. (In Russian)
30. Efanov D. V., Pashukov A. V. The Weight-Based Sum Codes in the Residue Ring by Arbitrary Modulus for Synthesis of Self-Checking Digital Computing Systems. Proceedings of 19th IEEE East-West Design & Test Symposium (EWDT'S'2021), Batumi, Georgia, September 10-13, 2021, pp. 170–179, DOI: 10.1109/EWDT'S52692.2021.9581032.
31. Zakrevskiy A. D., Pottosin Yu. V., Cheremisina L. D. *Logicheskie osnovy pro-ektirovaniya diskretnykh ustroystv* [Logical foundations for the design of discrete devices]. Moscow: Fizmatlit Publ., 2007. 592 p. (In Russian)
32. Sapozhnikov V. V., Sapozhnikov V. V., Urganov D. I. Universal'nye struktury dvoichnykh schetchikov edinit po proizvol'nomu modulyu scheta [Universal Structures of Binary Counters of Units by an Arbitrary Counting Modulus]. *Elektronnoe modelirovaniye* [Electronic modeling]. 2002, V. 24, I. 4, pp. 65–81. (In Russian)
33. Sapozhnikov V. V., Sapozhnikov V. V., Urganov D. I. Blochnaya struktura dvoichnogo schetchika edinit po proizvol'nomu modulyu scheta [Block structure of a binary counter of units by an arbitrary modulus of count]. *Elektronnoe modelirovaniye* [Electronic modeling]. 2005, V. 27, I. 4, pp. 65–81. (In Russian)
34. Sapozhnikov V. V., Sapozhnikov V. V., Urganov D. I. Composite Structure of Binary Counter of Ones Arbitrary Modulo. Proceedings of East-West Design & Test Workshop (EWDTW'05), 15-19 September 2005, Odessa, Ukraine, pp. 102–106.
35. Sapozhnikov V. V., Sapozhnikov V. V., Urganov D. I. Multistage Regular Structure of Binary Counter of Ones Arbitrary Modulo. Proceedings of East-West Design & Test Workshop (EWDTW'06), 15-19 September 2006, Sochi, Russia, pp. 287–290.
36. Suprun V. P., Dorozhinskiy A. L. *Summator po modulyu pyat'*. – *Avtorskoe svidetel'stvo SSSR № 1388850, SU 1803911 A1* [Adder modulo five. – Author's certificate of the USSR No. 1388850, SU 1803911 A1]. 1986. 3 p. (In Russian)
37. Avgul' L. B. *Summator po modulyu sem'*. – *Avtorskoe svidetel'stvo RF RU 2028660 C1, opublikovano 09.02.1995, podano 21.05.1992* [Adder modulo seven. – Author's certificate of the Russian Federation RU 2028660 C1, published on 02/09/1995, filed on 05/21/1992]. 7 p. (In Russian)
38. Suprun V. P., Gorodecky D. A. Realization of Addition and Multiplication Operations in Unitary Codes. Automatic Control and Computer Sciences. 2010, vol. 44, I. 5, pp. 292–301. DOI: 10.3103/S014641161005007X.
39. Suprun V. P. Single-Level Schematic Realization of Basic Operations of Modular Arithmetic in Unitary Codes. Automatic Control and Computer Sciences. 2011, vol. 5, I. 2, pp. 70–79. DOI: 10.3103/S0146411611020088.
40. Drozd A. V., Kharchenko V. S., Antoshchuk S. G., Drozd Yu. V., Drozd M. A., Sulima Yu. Yu. *Rabochee diagnostirovaniye bezopasnykh informatsionno-upravlyayushchikh sistem* [Working diagnostics of safe information and control systems]. Khar'kov: Natsional'nyy aerokosmicheskii univer-sitet im. N.E. Zhukovskogo «KhAI» Publ., 2012. 614 p. (In Russian)
41. Balaka E. S., Gorodetskiy D. A., Rukhlov V. S., Shchelokov A. N. Razrabotka vysokoskorostnykh summatorov po modulyu na baze kombinatsionnykh summatorov s parallel'nym perenosom [Development of high-speed modulo adders based on combinational adders with parallel transfer]. *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Technical science]. 2016, I. 6, pp. 158–169. (In Russian)
42. Intel® Quartus® Prime Download – Intel® Quartus® Prime Software. Available at: <https://www.intel.ru/content/www/ru/ru/software/programmable/quartus-prime/download.html> (accessed: November 08, 2021).
43. Khakhanov V. I., Litvinova E. I., Guz' O. A. *Proektirovaniye i testirovaniye tsifrovyykh sistem na kristallakh* [Design and testing of digital systems-on-chips]. Khar'kov: KhNUR Publ., 2009. 484 p. (In Russian)
44. Navabi Z. Digital System Test and Testable Design: Using HDL Models and Architectures. Springer Science+Business Media, LLC 2011, 435 p.
45. MAX II Device Handbook, vol. 1. Altera Corporation, 2007, 107 p.
46. PLIS semeytva MAX II. Available at: <http://altera.ru/plis-max-ii.html> (accessed: November 08, 2021).
47. Collection of Digital Design Benchmarks. Available at: <http://ddd.fit.cvut.cz/www/prj/Benchmarks/>.
48. 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.
49. Drozd A., Kharchenko V., Antoshchuk S., Sulima J., Drozd M. Checkability of the Digital Components in Safety-Critical Systems: Problems and Solutions. Proceedings of 9th IEEE East-West Design & Test Symposium (EWDT'S'2011), Sevastopol, Ukraine, 2011, pp. 411–416, DOI: 10.1109/EWDT'S.2011.6116606.
50. Drozd O., Rucinski A., Zashcholkina K., Martynuk O., Drozd J. Resilient Development of Models and Methods in Computing Space. Proceedings of 19th IEEE East-West Design & Test Symposium (EWDT'S'2021), Batumi, Georgia, September 10-13, 2021, pp. 70–75, DOI: 10.1109/EWDT'S52692.2021.9581002.