

ТРОИЧНЫЙ КОД С СУММИРОВАНИЕМ В НЕСИММЕТРИЧНОЙ СИСТЕМЕ СЧИСЛЕНИЯ ДЛЯ СИНТЕЗА КОНТРОЛЕПРИГОДНЫХ И САМОПРОВЕРЯЕМЫХ ЦИФРОВЫХ СИСТЕМ

Д. В. ЕФАНОВ

*Российский университет транспорта, 127994, Москва, Россия,
Санкт-Петербургский политехнический университет Петра Великого, 195251, Санкт-Петербург, Россия
E-mail: TrES-4b@yandex.ru*

Предложен подход к построению кода с суммированием разрядов информационных векторов в троичной несимметричной системе счисления. Подобные коды ориентированы на обнаружение ошибок в информационных векторах и имеют сравнительно невысокую избыточность, что позволяет применять их эффективно при разработке контролепригодных цифровых устройств и систем с обнаружением неисправностей. Проанализированы особенности обнаружения ошибок представленным кодом с суммированием в сравнении с известным кодом с суммированием, подразумевающим отдельный контроль разрядов, равных единице, и разрядов, равных двум. Показано, что код с суммированием непосредственно разрядов информационных векторов более эффективно использует свои контрольные разряды, чем рассмотренный ранее код; кроме того, он обладает меньшей избыточностью. При этом, однако, предложенный код не обнаруживает гораздо больше ошибок в информационных векторах, которые могут быть как композиционными, так и монотонными и асимметричными. Эти обстоятельства следует учитывать при синтезе цифровых устройств, функционирующих в троичной логике.

Ключевые слова: *техническая диагностика цифровых устройств, троичный код с суммированием, обнаружение ошибок в информационных векторах, троичная несимметричная система счисления, характеристики обнаружения ошибок*

Введение. В современном мире широко используются цифровые устройства и системы, функционирующие в двоичной логике, что фактически определено тотальным применением принстонской архитектуры вычислительных машин (архитектурой фон Неймана). Группой ученых из Института перспективных исследований (Institute for Advanced Study, IAS) в 1946 г. опубликована фундаментальная работа в области компьютерной техники и технологий [1], в которой, в том числе, демонстрируются преимущества двоичной системы счисления в технической реализации, удобстве и простоте выполнения арифметических и логических операций. Тем не менее в настоящее время активно обсуждается передел современной электроники, в частности, обращается внимание на возможность использования недвоичной логики для ее работы. Учеными всего мира ведутся дискуссии о возможности использования систем счисления больших порядков для формирования новой элементной базы.

Широко известно, что оптимальным основанием для представления данных является $e \approx 2,718$: число 3 ближе к этому значению, чем используемое в настоящее время для представления данных в вычислительных системах основание 2.

Многими специалистами отмечаются преимущества троичной логики [2] и троичной системы счисления перед двоичной [3] и предпринимаются попытки разработки троичных вычислительных систем [4, 5]. Первой такой системой стала электронная троичная вычислительная система „Сетунь“, разработанная группой инженеров МГУ (Москва, СССР) в 1959 г. на основе троичной ферритодиодной ячейки Н. П. Брусенцова и функционирующая в троичной симметричной системе счисления. Большого распространения система не получила. В современном мире также обращают внимание на троичную логику. Известна и более современная реализация троичного компьютера: система ТСА 2 (версия v2.0), построенная в 2008 г. Дж. Коннели, К. Пателем и А. Чавезом при поддержке Ф. Нико в Калифорнийском государственном политехническом университете (Сан-Луис-Обиспо, США) [6]. Рядом специалистов отмечаются преимущества троичной логики перед двоичной и при реализации квантовых компьютеров [7].

До сих пор ввиду массового производства микроэлектронных компонентов, функционирующих в двоичной логике, и непосредственного использования действующих принципов реализации вычислительной техники троичная логика не распространена. Более того, физическая реализация кодирующих устройств для кодов троичной логики более сложна, чем для устройств, функционирующих в двоичной логике. Это в полной мере касается и представления сигналов в двоичной логике: есть сигнал/нет сигнала. В троичной логике сигналы интерпретируются иначе. Тут следует указать на возможности использования различных троичных систем счисления (симметричной и несимметричной) и соответственно различной интерпретации сигналов. Тем не менее во всем мире существуют целые направления в исследованиях, охватывающие как особенности технической реализации устройств, функционирующих в троичной логике [8—12], так и способов представления и защиты информации, представленной в троичной логике [13—15]. Это в полной мере касается и развития вычислительных технологий в области реализации технологий ближайшего будущего — квантовых компьютеров [16, 17].

Важным направлением исследований является разработка способов защиты данных, представленных в троичной логике. Это направление развивается в основном путем создания помехоустойчивых кодов, ориентированных на коррекцию ошибок [18, 19]. Однако известны и работы в области построения кодов, ориентированных только на обнаружение ошибок [20, 21]. Такие коды менее избыточны, чем корректирующие, и могут применяться при построении контролепригодных и самопроверяемых цифровых систем, функционирующих в троичной логике, подобно тому как это используется в современных устройствах автоматики, реализующих принципы двоичной логики [22—27].

В настоящей статье анализируются особенности обнаружения ошибок в информационных векторах троичных кодов с суммированием разрядов информационных векторов, что актуально при разработке схем встраиваемого контроля компонентов цифровых троичных вычислительных систем.

Троичный код с обнаружением любых некомпозиционных ошибок. Зачастую в избыточный код, применяемый в задачах синтеза цифровых контролепригодных систем и систем с обнаружением неисправностей, при построении закладываются определенные особенности. К таким особенностям следует отнести возможности обнаружения любых ошибок определенных кратностей и видов (по совокупности искажений разрядов, равных 0, 1 и 2) в информационных векторах [28]. Естественно, существует ограничение, связанное с числом используемых для этого контрольных разрядов. Понятно, что может быть построен такой код, который будет обнаруживать максимальное число ошибок всех видов и кратностей — код с повторением. Этот код будет иметь двойную избыточность. Возникает задача построения такого троичного избыточного кода, который будет обладать свойством обнаружения ошибок определенных видов и

кратностей (но не всех) при уменьшенной по сравнению с дублированием избыточности. Такой код, например, предложен в [29, 30]. Он относится к классу кодов с суммированием.

Для построения троичного кода с суммированием в информационном векторе длиной m отдельно считается число разрядов, равных 1 (число r_1), и число разрядов, равных 2 (r_2). Полученные числа r_1 и r_2 записываются соответственно в $k_1 = \lceil \log_3(m+1) \rceil$ старших разрядах контрольного вектора (запись $\lceil \dots \rceil$ обозначает целое сверху от вычисляемого значения) и в $k_2 = \lceil \log_3(m+1) \rceil$ младших разрядах контрольного вектора. Предложенный $\Sigma(m,k)$ -код будет обнаруживать любые виды ошибок, кроме композиционных, в информационных векторах, при которых в искаженном векторе число нулей, единиц и двоек сохраняется таким же, как в исходном.

По своим свойствам троичный $\Sigma(m,k)$ -код схож с известным двоичным кодом Бергера [31]. При этом его избыточность равна $k = k_1 + k_2 = 2 \lceil \log_3(m+1) \rceil$, а информационные векторы распределяются между контрольными векторами крайне неравномерно. Таким образом, эффективность использования контрольных разрядов $\Sigma(m,k)$ -кодом весьма низка.

Эффективность использования контрольных разрядов оценивается следующим отношением [32]:

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

где $N_{m,k}$ — число не обнаруживаемых кодом ошибок, а $N_{m,k}^{\min} = 3^m (3^{m-k} - 1)$ — минимальное число необнаруживаемых ошибок для кода с длиной информационных и контрольных векторов m и k .

Согласно [29, 30], при $m=5-8$ эффективность $\Sigma(m,k)$ -кода $\xi_{m,k}=11-25\%$, а при $m=9-15$ — $3-5\%$. Это говорит о крайне низкой эффективности использования кодами своих контрольных разрядов. Тем не менее $\Sigma(m,k)$ -коды, как и их двоичные аналоги, могут широко применяться при синтезе контролепригодных цифровых систем и технических средств диагностирования компонентов.

Построим новый троичный код с суммированием, который будет иметь больший коэффициент эффективности $\xi_{m,k}$.

Правила построения троичного кода с суммированием. Для построения кода с суммированием:

- 1) значения разрядов отождествляются с весом каждого разряда $w_i \in \{0, 1, 2\}$, $i = \overline{1, m}$;
- 2) определяется суммарный вес разрядов информационного вектора:

$$W = \sum_{i=1}^m w_i; \quad (2)$$

3) полученное число W представляется в троичном виде и записывается в разряды контрольного вектора.

Получаемый по представленному алгоритму троичный код с суммированием обозначим $\Sigma^*(m,k)$. В отличие от $\Sigma(m,k)$ -кода, который строится независимо от используемой системы счисления (симметричной или несимметричной [29, 30]), $\Sigma^*(m,k)$ -код подразумевает использование только несимметричной системы счисления (важны абсолютные значения разрядов информационного вектора).

$\Sigma^*(m,k)$ -код будет иметь $k = \lceil \log_3(W_{\max} + 1) \rceil$ контрольных разрядов, где $W_{\max} = 2m$ (максимальный вес получается путем суммирования всех максимальных весов разрядов). Например, для случая $m=10$ $W_{\max} = 2 \cdot 10 = 20$, $k = \lceil \log_3(20 + 1) \rceil = 3$ и строится $\Sigma^*(10,3)$ -код. Для сравнения, троичный код [29, 30] будет иметь $k=6$ контрольных разрядов в этом случае.

Сфокусируем внимание на свойствах $\Sigma^*(m,k)$ -кода, сравнивая его с $\Sigma(m,k)$ -кодом.

Особенности обнаружения ошибок в информационных векторах. Для определения числа необнаруживаемых ошибок требуется представить код в табличной форме, где указать распределение информационных векторов между всеми контрольными векторами (распределение по контрольным группам). Ошибка не будет обнаружена тогда и только тогда, когда она будет переводить информационный вектор одной контрольной группы в информационный вектор этой же контрольной группы. Тогда общее число не обнаруживаемых кодом ошибок будет складываться из числа необнаруживаемых ошибок в каждой контрольной группе, определяемых числом двунаправленных переходов каждого информационного вектора в каждый:

$$N_{m,k}^p = 2C_{q_p}^2 = 2 \cdot \frac{q_p!}{2!(q_p-2)!} = q_p(q_p-1), \quad (3)$$

где p — номер контрольной группы, q_p — число информационных векторов в контрольной группе.

Для $\Sigma(m,k)$ -кода число контрольных групп определяется количеством способов подсчета r_1 и r_2 . Именно эти числа однозначно идентифицируют конкретную композицию разрядов в информационном векторе. Таким образом, для представления $\Sigma(m,k)$ -кода в табличной форме все информационные векторы должны быть разбиты на контрольные группы r_1 - r_2 , соответствующие контрольным векторам. Такая таблица имеет достаточно внушительные размеры. Например, для случая $\Sigma(5,4)$ -кода имеется 21 контрольная группа и 243 информационных вектора, которые распределены в контрольные группы. В табл. 1 приведены все контрольные группы рассматриваемого кода, для каждой указано по одному представителю, число информационных векторов, а также число необнаруживаемых ошибок. Как отмечено ранее, все эти ошибки являются композиционными.

Таблица 1

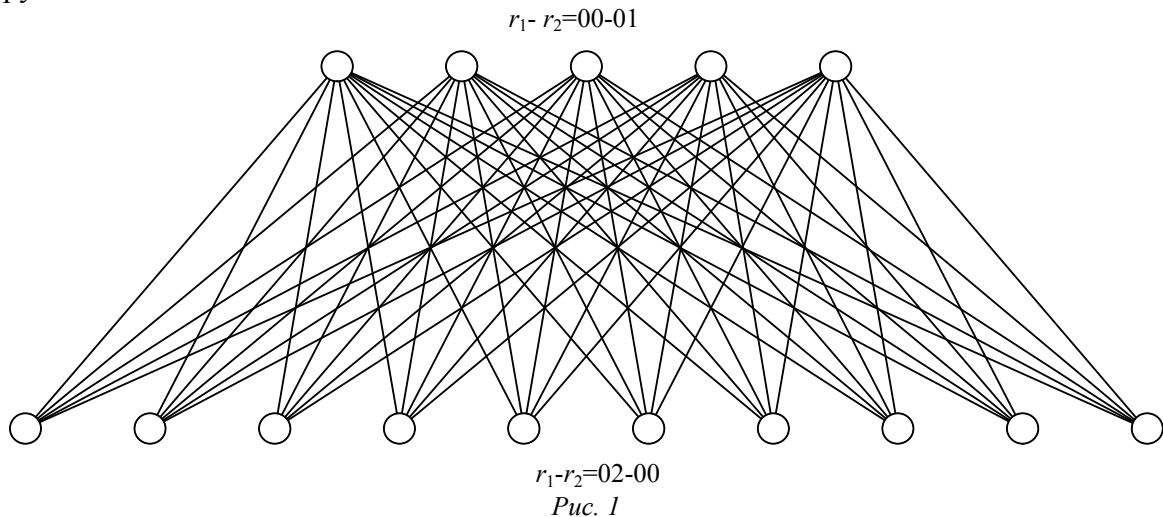
Контрольные группы $\Sigma(5,4)$ -кода

r_1 - r_2	Пример информационного вектора	Число информационных векторов	Число необнаруживаемых ошибок в контрольной группе
00-00	00000	1	0
00-01	00002	5	20
00-02	00022	10	90
00-10	00222	10	90
00-11	02222	5	20
00-12	22222	1	0
01-00	00001	5	20
01-01	00012	20	380
01-02	00122	30	870
01-10	01222	20	380
01-11	12222	5	20
02-00	00011	10	90
02-01	00112	30	870
02-02	01122	30	870
02-10	11222	10	90
10-00	00111	10	90
10-01	01112	20	380
10-02	11122	10	90
11-00	01111	5	20
11-01	11112	5	20
12-00	11111	1	0
Всего		243	4410

Установим различия контрольных групп кодов $\Sigma(m,k)$ и $\Sigma^*(m,k)$.

Состав контрольных групп у $\Sigma^*(m,k)$ -кода определяется числом W . Следует отметить, что это число одинаково для всех информационных векторов с одинаковой композицией (табл. 2). Отсюда следует, что каждая контрольная группа $\Sigma(m,k)$ -кода r_1-r_2 всегда будет полностью включена в одну из групп $\Sigma^*(m,k)$ -кода. При этом каждой группе r_1-r_2 однозначно не соответствует число W . К примеру, группа $r_1-r_2=00-01$ $\Sigma(5,4)$ -кода соответствует группе $W=2$ кода $\Sigma^*(5,2)$, но этой же группе соответствует и группа $r_1-r_2=02-00$ кода $\Sigma(5,4)$. В табл. 3 приведен состав контрольных групп $\Sigma^*(5,4)$ -кода. Отсюда следует, что к необнаруживаемым композиционным ошибкам кода $\Sigma^*(m,k)$ добавляются „новые“ некомпозиционные ошибки (монотонные и асимметричные [28]), связанные с переходами информационных векторов разных контрольных групп r_1-r_2 внутри контрольной группы W .

В контрольных группах $\Sigma^*(5,4)$ -кода $W=0, 1, 9, 10$ присутствует только по одной контрольной группе r_1-r_2 $\Sigma(5,4)$ -кода — дополнительных ошибок эти группы не дают. В группах, соответствующих 2 и 8, находятся по две контрольные группы r_1-r_2 $\Sigma(5,4)$ -кода: первая содержит пять информационных векторов, а вторая — десять. Каждый вектор первой группы может преобразоваться в каждый вектор второй группы и наоборот (рис. 1), что дает для каждой из групп суммарно 100 необнаруживаемых ошибок. Аналогично группы 3 и 7 содержат по две группы r_1-r_2 $\Sigma(5,4)$ -кода, включающие 20 и 10 информационных векторов, что для каждой из групп W дает по 400 необнаруживаемых ошибок. Группы 4 и 6 содержат по три группы r_1-r_2 $\Sigma(5,4)$ -кода, имеющие 10, 30 и 5 векторов. Суммарно это 1000 необнаруживаемых ошибок для каждой группы W . И наконец, группа 5 содержит три группы r_1-r_2 $\Sigma(5,4)$ -кода, имеющие 30, 20 и 1 вектор, что вносит еще 1300 необнаруживаемых ошибок некомпозиционного типа. Таким образом, в $\Sigma^*(5,4)$ -коде дополнительно к имеющимся необнаруживаемым 4410 ошибкам $\Sigma(5,4)$ -кода есть 4300 ошибок некомпозиционного типа: всего 8710 необнаруживаемых ошибок.



Число „дополнительных“ необнаруживаемых ошибок в $\Sigma^*(m,k)$ -коде по отношению к $\Sigma(m,k)$ -коду оказывается значительным. Например, при $m=4$ оно равно 468, что составляет 83,871 % от общего числа не обнаруживаемых $\Sigma(4,4)$ -кодом ошибок; 121,137 % от ошибок, не обнаруживаемых $\Sigma(6,4)$ -кодом, и 193,166 % — $\Sigma(12,6)$ -кодом. Следует отметить, что избыточность $\Sigma^*(m,k)$ -кода при $m=12$ вдвое меньше избыточности $\Sigma(m,k)$ -кода.

В табл. 4 и 5 приведены показатели обнаружения ошибок кодами $\Sigma(m,k)$ и $\Sigma^*(m,k)$ при длине информационных векторов $m=3—12$ ($\gamma_{m,k}$ — доля необнаруживаемых ошибок от общего числа ошибок в троичных информационных векторах, $\xi_{m,k}$ — эффективность; для сравнения кодов на рис. 2 и 3 представлены графики указанных величин).

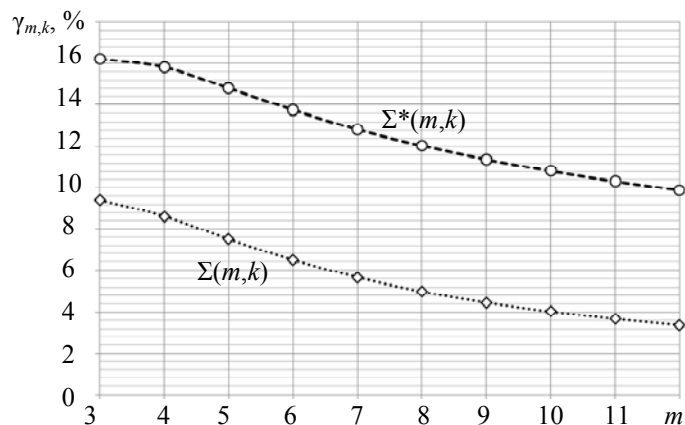


Рис. 2

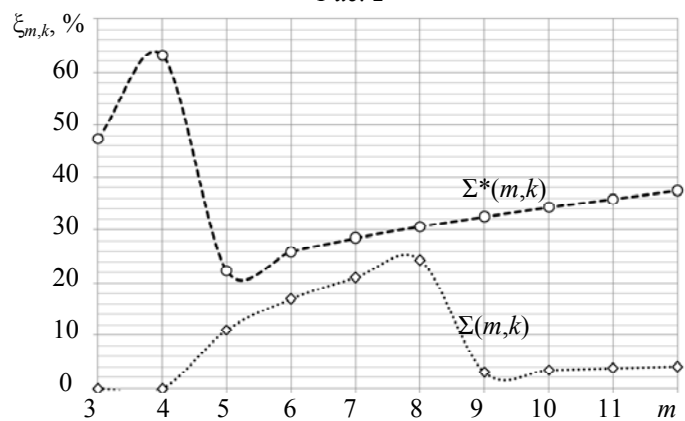


Рис. 3

Таблица 2

Значение W
для контрольных групп $\Sigma(5,4)$ -кода

r_1-r_2	W
00-00	0
00-01	2
00-02	4
00-10	6
00-11	8
00-12	10
01-00	1
01-01	3
01-02	5
01-10	7
01-11	9
02-00	2
02-01	4
02-02	6
02-10	8
10-00	3
10-01	5
10-02	7
11-00	4
11-01	6
12-00	5

Таблица 3

Состав контрольных групп $\Sigma^*(5,4)$ -кода

W										
0	1	2	3	4	5	6	7	8	9	10
00-00	01-00	00-01 02-00	01-01 10-00	00-02 02-01 11-00	01-02 10-01 12-00	00-10 02-02 11-01	01-10 10-02	00-11 02-10	01-11	00-12

Таблица 4

Показатели обнаружения ошибок $\Sigma(m,k)$ -кодом

m	k	N_m	$N_{m,k}^{\min}$	$N_{m,k}$	$\gamma_{m,k}$	$\xi_{m,k}$
3	4	702	0	66	9,402	0
4	4	6480	0	558	8,611	0
5	4	58806	486	4410	7,499	11,02
6	4	530712	5832	34440	6,489	16,934
7	4	4780782	56862	270648	5,661	21,01
8	4	43040160	524880	2151198	4,998	24,399
9	6	387400806	511758	17300154	4,466	2,958
10	6	3486725352	4723920	140609016	4,033	3,36
11	6	31380882462	42869574	1153285848	3,675	3,717
12	6	282429005040	386889048	9533107584	3,375	4,058

Таблица 5

Показатели обнаружения ошибок $\Sigma^*(m,k)$ -кодом

m	k	N_m	$N_{m,k}^{\min}$	$N_{m,k}$	$\gamma_{m,k}$	$\xi_{m,k}$
3	2	702	54	114	16,239	47,368
4	2	6480	648	1026	15,833	63,158
5	3	58806	1944	8710	14,811	22,319
6	3	530712	18954	73060	13,766	25,943
7	3	4780782	174960	614040	12,844	28,493
8	3	43040160	1587762	5190066	12,059	30,592
9	3	387400806	14329224	44133126	11,392	32,468
10	3	3486725352	129081114	377320320	10,822	34,21
11	3	31380882462	1162084320	3240958380	10,328	35,856
12	3	282429005040	10459821762	27947804940	9,896	37,426

Проанализировав рис. 2 и 3, можно сделать вывод о том, что $\Sigma^*(m,k)$ -код не обнаруживает существенно больше ошибок в информационных векторах, чем $\Sigma(m,k)$. Например, при $m=4$ $\Sigma^*(m,k)$ -кодом не обнаруживается в 1,839 раза больше ошибок в информационных векторах, чем $\Sigma(m,k)$ -кодом, а при $m=12$ — в 2,932 раза. При этом $\Sigma^*(m,k)$ -код более эффективно использует свои контрольные разряды: при $m=3-4$ $\xi_{m,k}=45-65$ %, при $m=5-12$ $\xi_{m,k}=20-40$ %. С увеличением m в каждом диапазоне коэффициент $\xi_{m,k}$ постепенно возрастает, уменьшаясь при переходе между диапазонами длин информационных векторов, соответствующих различным значениям k .

Заключение. Описанный в статье код с суммированием разрядов информационных векторов в троичной несимметричной системе счисления может применяться в задачах синтеза контролепригодных цифровых систем, функционирующих в троичной логике, а также технических средств их диагностирования. Следует отметить, что в отличие от предложенного

ранее кода с суммированием, подразумевающего отдельный контроль единиц и двоек, представленный $\Sigma^*(m,k)$ -код помимо всех композиционных ошибок не обнаруживает некоторую долю монотонных и асимметричных ошибок. При этом доля не обнаруживаемых $\Sigma^*(m,k)$ -кодом ошибок от общего числа ошибок в информационных векторах невелика и при $m > 3$ составляет менее 16 %, уменьшаясь с увеличением длины информационного вектора. $\Sigma^*(m,k)$ -код менее избыточен, чем $\Sigma(m,k)$ -код, и гораздо эффективнее использует свои контрольные разряды. Тем не менее при реализации контролепригодных и самопроверяемых цифровых устройств следует учитывать особенности обнаружения различных видов ошибок представленным в статье кодом с суммированием.

При построении кодов с суммированием, которые целесообразно учитывать при создании цифровых вычислительных систем, следует учесть также возможности установления неравноправия между разрядами информационных векторов путем приписывания им весовых коэффициентов [33, 34], а также принципы модификации, изложенные, например, в [35]. Это даст разработчику возможность выбора способа реализации контролепригодной вычислительной системы с самопроверяемыми компонентами.

Необходимо отметить, что исследования в области представления данных и кодирования информации могут стать основой для перехода к использованию более экономичной троичной системы счисления вместо традиционной двоичной при реализации цифровых вычислительных устройств и систем ближайшего будущего.

СПИСОК ЛИТЕРАТУРЫ

1. *Burks A. W., Goldstine H. H., Neumann J.* Preliminary Discussion of the Logical Design of an Electronic Computing Instrument. Princeton, NJ: Institute for Advanced Study, July 1946. P. 1—33.
2. *Карпенко А. С.* Развитие многозначной логики. М.: Изд-во ЛКИ, 2010. 448 с.
3. *Кушнеров А.* Троичная цифровая техника. Ретроспектива и современность. Беэр-Шева, Израиль: Университет им. Бен-Гуриона, 28.10.05.
4. *Брусенцов Н. П., Маслов С. П., Розин В. П., Тишулина А. М.* Малая цифровая вычислительная машина „Сетунь“. М.: Изд-во МГУ, 1962. 140 с.
5. *Roy D., Merril Jr.* Ternary Logic in Digital Computers // Proc. of the SHARE design automation project (DAC '65). ACM New York, NY, USA. P. 6.1—6.17. DOI: 10.1145/800266.810759.
6. *Connely J.* Ternary Computing Testbed 3-Trit Computer Architecture. California Polytechnic State University of San Luis Obispo, 29 August 2008. 184 p.
7. *Lanyon B. P., Barbieri M., Almeida M. P., Jennewein T., Ralph T. C., Resch K. J., Pryde G. J., O'Brien J. L., Gilchrist A., White A. G.* Simplifying Quantum Logic Using Higher-Dimensional Hilbert Spaces // Nature Physics. 2009. Vol. 5, is. 2. P. 134—140. DOI: 10.1038/nphys1150.
8. *Hu M., Smith K. C.* Self-Checking Binary Logic Systems Using Ternary Logic Circuits // Canadian Electrical Engineering Journal. 1984. Vol. 9, is. 3. P. 100—104. DOI: 10.1109/CEEJ.1984.6593793.
9. *Wu J.* Ternary Logic Circuit for Error Detection and Error Correction // Proc. of 19th Intern. Symp. on Multiple-Valued Logic. Guangzhou, China, 29—31 May 1989. P. 94—99. DOI: 10.1109/ISMVL.1989.37766.
10. *Zhuang N., Wu H.* Novel Ternary JKL Flip-Flop // Electronics Letters. 1990. Vol. 26, is. 15. P. 1145—1146. DOI: 10.1049/el:19900741.
11. *Vudadha C., Katragadda S., Phaneendra P. S.* 2:1 Multiplexer Based Design for Ternary Logic Circuits // IEEE Asia Pacific Conf. on Postgraduate Research in Microelectronics and Electronics (PrimeAsia). Visakhapatnam, India, 19—21 December 2013. P. 46—51. DOI: 10.1109/PrimeAsia.2013.6731176.
12. *Kim S., Lim T., Kang S.* An Optimal Gate Design for the Synthesis of Ternary Logic Circuits // 23rd Asia and South Pacific Design Automation Conf. (ASP-DAC). Jeju, South Korea, 22—25 January 2018. P. 476—481. DOI: 10.1109/ASPDAC.2018.8297369.

13. *Brouwer A. E., Hamalainen H. O., Ostergard P. R. J., Sloane N. J. A.* Bounds on Mixed Binary/Ternary Codes // IEEE Transactions on Information Theory. 1988. Vol. 44, is. 1. P. 140—161. DOI: 10.1109/18.651001.
14. *Bitouze N., Graell i Amat A., Rosnes E.* Error Correcting Coding for a Nonsymmetric Ternary Channel // IEEE Transactions on Information Theory. 2010. Vol. 56, is. 11. P. 5715—5729. DOI: 10.1109/TIT.2010.2069211.
15. *Laaksonen A., Östergård P. R. J.* New Lower Bounds on Error-Correcting Ternary, Quaternary and Quinary Codes // Lecture Notes in Computer Science. Coding Theory and Applications, 5th Intern. Castle Meeting, ICMCTA 2017. Vihula, Estonia, 28—31 August 2017. P. 228—237.
16. *Giesecke N.* Ternary quantum logic // Dissertation and Theses. Paper 4092. 2006. 265 p. <https://doi.org/10.15760/etd.5975>.
17. *Majumdar R., Basu S., Ghosh S., Sur-Kolay S.* Quantum Error Correcting Code for Ternary Logic // Phys. Rev. A. 2018. Vol. 97, is. 5. P. 1—7. DOI: 10.1103/PhysRevA.97.052302.
18. *Hosany M. A., Khodabux N. U. S.* Novel Ternary Line Codes Employing Trellis Coded Modulation // 4th Intern. Conf. on Signal Processing and Communication Systems. Gold Coast, QLD, Australia, 13—15 December 2010. DOI: 10.1109/ICSPCS.2010.5709696.
19. *Abdelaziz M., Gulliver T. A.* Ternary Convolutional Codes for Ternary Phase Shift Keying // IEEE Communications Letters. 2016. Vol. 20, is. 9. P. 1709—1712. DOI: 10.1109/LCOMM.2016.2587698.
20. *Svanström M.* A Lower Bound for Ternary Constant Weight Codes // IEEE Transactions on Information Theory. 1997. Vol. 43. P. 1630—1632.
21. *Svanström M., Östergård P. R. J., Bogdanova G. T.* Bounds and Constructions for Ternary Constant-Composition Codes // IEEE Transactions on Information Theory. 2002. Vol. 48. P. 101—111.
22. *Согомонян Е. С., Слабаков Е. В.* Самопроверяемые устройства и отказоустойчивые системы. М.: Радио и связь, 1989. 208 с.
23. *Микони С. В.* Общие диагностические базы знаний вычислительных систем. СПб: СПИИРАН, 1992. 234 с.
24. *Piestrak S. J.* Design of Self-Testing Checkers for Unidirectional Error Detecting Codes. Wrocław: Oficyna Wydawnicza Politechniki Wrocławskiej, 1995. 111 p.
25. *Fujiwara E.* Code Design for Dependable Systems: Theory and Practical Applications. John Wiley & Sons, 2006. 720 p.
26. *Сапожников В. В., Сапожников Вл. В., Ефанов Д. В.* Коды Хэмминга в системах функционального контроля логических. СПб: Наука, 2018. 151 с.
27. *Сапожников В. В., Сапожников Вл. В., Ефанов Д. В.* Коды с суммированием для систем технического диагностирования. Т. 1. Классические коды Бергера и их модификации. М.: Наука, 2020. 383 с.
28. *Ефанов Д. В.* Ошибки в троичных кодовых векторах, их классификация и обнаружение с помощью помехозащищенного кодирования // Изв. вузов. Приборостроение. 2020. Т. 63, № 5. С. 391—404. DOI: 10.17586/0021-3454-2020-63-5-391-404.
29. *Ефанов Д. В.* Троичные коды с суммированием для контроля цифровых схем // Проблемы управления. 2020. № 4. С. 63—71. DOI: <http://doi.org/10.25728/pu.2020.4.6>.
30. *Efanov D.* Ternary Sum Codes // Proc. of 18th IEEE East-West Design & Test Symposium (EWDTS'2020). Varna, Bulgaria, September 4—7, 2020. P. 92—99. DOI: 10.1109/EWDTS50664.2020.9225033.
31. *Berger J. M.* A Note on Error Detecting Codes for Asymmetric Channels // Information and Control. 1961. Vol. 4, is. 1. P. 68—73. DOI: 10.1016/S0019-9958(61)80037-5.
32. *Ефанов Д. В.* Троичный код паритета в системах рабочего диагностирования устройств автоматики и вычислительной техники // Информационные технологии. 2019. Т. 25, № 7. С. 426—434. DOI: 10.17587/it.25.426-434.
33. *Das D., Toubia N. A.* Weight-Based Codes and Their Application to Concurrent Error Detection of Multilevel Circuits // Proc. of 17th IEEE Test Symposium. California, USA, 1999. P. 370—376. DOI: 10.1109/VTEST.1999.766691.
34. *Das D., Toubia N. A., Seuring M., Gossel M.* Low Cost Concurrent Error Detection Based on Modulo Weight-Based Codes // Proc. of the IEEE 6th Intern. On-Line Testing Workshop (IOLTW). Spain, Palma de Mallorca, July 3—5, 2000. P. 171—176. DOI: 10.1109/OLT.2000.856633.

35. Ефанов Д. В. Троичные коды с суммированием и их модификации // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС). 2020. № 1. С. 119—125. DOI: 10.31114/2078-7707-2020-1-119-125.

Сведения об авторе

Дмитрий Викторович Ефанов — д-р техн. наук, доцент; Российский университет транспорта, кафедра автоматизации, телемеханики и связи на железнодорожном транспорте; Санкт-Петербургский политехнический университет Петра Великого; Высшая школа транспорта Института машиностроения, материалов и транспорта; профессор; E-mail: TrES-4b@yandex.ru

Поступила в редакцию
18.10.2020 г.

Ссылка для цитирования: Ефанов Д. В. Троичный код с суммированием в несимметричной системе счисления для синтеза контролепригодных и самопроверяемых цифровых систем // Изв. вузов. Приборостроение. 2021. Т. 64, № 6. С. 433—443.

TERNARY CODE WITH SUMMATION IN AN ASYMMETRIC NUMBER SYSTEM FOR THE SYNTHESIS OF CONTROLLABLE AND SELF-CHECKING DIGITAL SYSTEMS

D. V. Efanov

Russian University of Transport, 127994, Moscow, Russia,
Peter the Great St. Petersburg Polytechnic University, 195251, St. Petersburg, Russia
E-mail: TrES-4b@yandex.ru

An approach to the construction of a code with the summation of the bits of information vectors in a ternary asymmetric number system is proposed. Such codes are focused on the detection of errors in information vectors and have a relatively low redundancy, which makes it possible to use them effectively in the development of traceable digital devices and systems with fault detection. The features of error detection by the presented code with summation are analyzed in comparison with the known code with summation, which implies separate control of digits equal to one and digits equal to two. It is shown that a code with direct summation of the bits of information vectors uses its control bits more efficiently than the code considered earlier; moreover, it has less redundancy. At the same time, however, the proposed code does not reveal much more errors in information vectors, which can be both compositional and monotonic and asymmetric. The above circumstances are recommended to be taken into account when synthesizing digital devices operating in ternary logic.

Keywords: digital devices testing, ternary sum code, error detection in data vectors, ternary asymmetrical system of the numeration, error detection characteristics

REFERENCES

- Burks A.W., Goldstine H.H., Neumann J. *Preliminary Discussion of the Logical Design of an Electronic Computing Instrument*, Institute for Advanced Study, Princeton, N J, July 1946, pp. 1–33.
- Karpenko A.S. *Razvitiye mnogoznachnoy logiki* (Development of Multi-Valued Logic), Moscow, 2010, 448 p. (in Russ.)
- Kushnerov A. *Troichnaya tsifrovaya tekhnika. Retrospektiva i sovremennost'* (Ternary Digital Technology. Retrospective and Modernity), University. Ben Gurion, Beer Sheva, Israel. 28.10.05.
- Brusentsov N.P., Maslov S.P., Rozin V.P., Tishulina A.M. *Malaya tsifrovaya vychislitel'naya mashina "Setun"* (Small Digital Computer "Setun"), Moscow, 1962, 140 p. (in Russ.)
- Roy D., Merril Jr. *Proceedings of the SHARE design automation project (DAC '65)*, ACM New York, NY, USA, pp. 6.1–6.17, DOI: 10.1145/800266.810759.
- Connely J. *Ternary Computing Testbed 3-Trit Computer Architecture*, California Polytechnic State University of San Luis Obispo, August 29th, 2008, 184 p.
- Lanyon B.P., Barbieri M., Almeida M.P., Jennewein T., Ralph T.C., Resch K.J., Pryde G.J., O'Brien J.L., Gilchrist A., White A.G. *Nature Physics*, 2009, no. 2(5), pp. 134–140, DOI: 10.1038/nphys1150.
- Hu M., Smith K.C. *Canadian Electrical Engineering Journal*, 1984, no. 3(9), pp. 100–104, DOI: 10.1109/CEEJ.1984.6593793.
- Wu J. *Proceedings of 19th International Symposium on Multiple-Valued Logic*, 29–31 May 1989, Guangzhou, China, pp. 94–99, DOI: 10.1109/ISMVL.1989.37766.
- Zhuang N., Wu H. *Electronics Letters*, 1990, no. 15(26), pp. 1145–1146, DOI: 10.1049/el:19900741.
- Vudadha C., Katragadda S., Phaneendra P.S. *Microelectronics and Electronics (PrimeAsia)*, 19–21 December 2013, Visakhapatnam, India, pp. 46–51, DOI: 10.1109/PrimeAsia.2013.6731176.
- Kim S., Lim T., Kang S. *23rd Asia and South Pacific Design Automation Conference (ASP-DAC)*, 22–25 January 2018, Jeju, South Korea, pp. 476–481, DOI: 10.1109/ASPAC.2018.8297369.

13. Brouwer A.E., Hamalainen H.O., Ostergard P.R.J., Sloane N.J.A. *IEEE Transactions on Information Theory*, 1988, no. 1(44), pp. 140–161, DOI: 10.1109/18.651001.
14. Bitouze N., Graell i Amat A., Rosnes E. *IEEE Transactions on Information Theory*, 2010, no. 11(56), pp. 5715–5729, DOI: 10.1109/TIT.2010.2069211.
15. Laaksonen A., Östergård P.R.J. *Lecture Notes in Computer Science 10495*, Springer: Coding Theory and Applications, 5th International Castle Meeting, ICMCTA 2017, Vihula, Estonia, August 28–31, 2017, pp. 228–237.
16. Giesecke N. *Ternary quantum logic. – Dissertation and Theses*, Paper 4092, 2006, 265 p. <https://doi.org/10.15760/etd.5975>.
17. Majumdar R., Basu S., Ghosh S., Sur-Kolay S. *Physical Review A*, 2018, no. 5 (97), pp. 1–7, DOI: 10.1103/PhysRevA.97.052302.
18. Hosany M.A., Khodabux N.U.S. *4th International Conference on Signal Processing and Communication Systems*, 13–15 December 2010, Gold Coast, QLD, Australia, DOI: 10.1109/ICSPCS.2010.5709696.
19. Abdelaziz M., Gulliver T.A. *IEEE Communications Letters*, 2016, no. 9(20), pp. 1709–1712, DOI: 10.1109/LCOMM.2016.2587698.
20. Svanström M. *IEEE Transactions on Information Theory*, 1997, vol. 43, pp. 1630–1632.
21. Svanström M., Östergård P.R.J., Bogdanova G.T. *IEEE Transactions on Information Theory*, 2002, vol. 48, pp. 101–111.
22. Sogomonyan E.S., Slabakov E.V. *Samoproveryaemye ustroystva i otkazoustoychivye sistemy* (The Self-Checked Devices and Failure-Safe Systems), Moscow, 1989, 208 p. (in Russ.)
23. Mikoni S.V. *Obshchiye diagnosticheskiye bazy znaniy vychislitel'nykh sistem* (General Diagnostic Knowledge Bases of Computing Systems), St. Petersburg, 1992, 234 p. (in Russ.)
24. Piestrak S.J. *Design of Self-Testing Checkers for Unidirectional Error Detecting Codes*, Wrocław, Oficyna Wydawnicza Politechniki Wrocławskiej, 1995, 111 p.
25. Fujiwara E. *Code Design for Dependable Systems: Theory and Practical Applications*, John Wiley & Sons, 2006, 720 p.
26. Sapozhnikov V.V., Sapozhnikov VI.V., Efanov D.V. *Kody Khemminga v sistemakh funktsional'nogo kontrolya logicheskikh* (Hamming Codes in Functional Control Systems of Logical), St. Petersburg, 2018, 151 p. (in Russ.)
27. Sapozhnikov V.V., Sapozhnikov VI.V., Efanov D.V. *Kody s summirovaniyem dlya sistem tekhnicheskogo diagnostirovaniya. T. 1. Klassicheskiye kody Bergera i ikh modifikatsii* (Summed Codes for Technical Diagnostic Systems. Vol. 1. Classical Berger Codes and Their Modifications), Moscow, 2020, 383 p. (in Russ.)
28. Efanov D.V. *Journal of Instrument Engineering*, 2020, no. 5(63), pp. 391–404, DOI: 10.17586/0021-3454-2020-63-5-391-404. (in Russ.)
29. Efanov D.V. *Control Sciences*, 2020, no. 4, pp. 63–71, DOI: <http://doi.org/10.25728/pu.2020.4.6>. (in Russ.)
30. Efanov D. *Proceedings of 18th IEEE East-West Design & Test Symposium (EWDTS'2020)*, Varna, Bulgaria, September 4–7, 2020, pp. 92–99, DOI: 10.1109/EWDTS50664.2020.9225033.
31. Berger J.M. *Information and Control*, 1961, no. 1(4), pp. 68–73, DOI: 10.1016/S0019-9958(61)80037-5.
32. Efanov D.V. *Information Technologies*, 2019, no. 7(25), pp. 426–434, DOI: 10.17587/it.25.426-434. (in Russ.)
33. Das D., Touba N.A. *Proceedings of 17th IEEE Test Symposium*, California, USA, 1999, pp. 370–376, DOI: 10.1109/VTEST.1999.766691.
34. Das D., Touba N.A., Seuring M., Gossel M. *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.
35. Efanov D.V. *Problemy razrabotki perspektivnykh mikro- i nanoelektronnykh sistem (MES)*, 2020, no. 1, pp. 119–125, DOI: 10.31114/2078-7707-2020-1-119-125. (in Russ.)

Data on author

Dmitry V. Efanov —

Dr. Sci., Associate Professor; Russian University of Transport; Department of Automation, Remote Control, and Communication in Railway Transport; Peter the Great St. Petersburg Polytechnic University, Higher School of Transport of the Institute of Machinery, Materials and Transport, Professor; E-mail: TrES-4b@yandex.ru

For citation: Efanov D. V. Ternary code with summation in an asymmetric number system for the synthesis of controllable and self-checking digital systems. *Journal of Instrument Engineering*. 2021. Vol. 64, N 6. P. 433–443 (in Russian).

DOI: 10.17586/0021-3454-2021-64-6-433-443