

ПРОЕКТИРОВАНИЕ И ДИАГНОСТИКА ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

УДК 004.312

DOI: 10.17223/19988605/38/11

Н.Б. Буторина

О СВОЙСТВАХ ПРОГРАММИРУЕМЫХ ЛОГИЧЕСКИХ БЛОКОВ, РЕАЛИЗУЮЩИХ КОДОВЫЕ СЛОВА РАВНОВЕСНЫХ КОДОВ

Исследование поддержано Российским научным фондом (исследовательский проект № 14-19-00218).

Рассматривается проблема синтеза самотестируемого детектора для подмножества кодовых слов. Выделяются свойства тестируемости программируемых логических блоков, на входы которых поступают подмножества кодовых слов специального вида.

Ключевые слова: ПЛБ; самотестируемый детектор; равновесный код.

Увеличивающиеся сложность и значимость дискретных (цифровых) систем требуют их высокой надежности. Сложность систем повышает вероятность возникновения неисправностей в них. Обнаружение неисправности в первый же момент ее проявления на выходах устройства позволяет защитить систему, в которую устройство встроено. Такое обнаружение неисправности может достигаться, например, за счет использования самопроверяемых схем, позволяющих обнаруживать неисправности в режиме нормального функционирования схемы [1]. Обнаружение осуществляется с помощью детектора кодов, который обычно является самотестируемым.

Если при появлении неисправности из некоторого класса неисправностей в процессе нормального функционирования схемы (в рабочей области функционирования) данная неисправность либо обнаружится на выходах схемы в момент ее первого проявления, либо никак не влияет на функционирование схемы (остается необнаружимой в рабочей области функционирования), то такую схему будем называть самопроверяемой относительно рассматриваемого класса неисправностей.

Самотестируемость же означает, что в рамках рассматриваемого класса неисправностей для каждой из неисправностей существует тестовый набор среди множества кодовых слов детектора. В частности, если мы имеем дело с детектором (m, n) -кода (детектором равновесных кодовых слов), то множество кодовых слов включает в себя C_n^m булевых векторов длины n , содержащих m единичных компонент.

Автором рассматриваются детекторы, построенные с использованием программируемых логических блоков.

Ранее [2] предлагался метод проектирования самотестируемого детектора для множества всех кодовых слов равновесного кода (детектора (m, n) -кодов), основанный на использовании программируемых логических блоков (ПЛБ). Предполагалось, что программируемые логические блоки реализованы в рамках LUT-технологий, т.е. могут быть запрограммированы на реализацию любых интересующих нас булевых функций заданного числа переменных.

Рассматривалось множество V неисправностей детектора, которое включает в себя все кратные константные неисправности на входах ПЛБ. При этом в схеме детектора неисправным может быть только один ПЛБ. Предполагалось, что в системе самопроверяемая схема, детектор (m, n) -кодов неисправными могут быть либо схема, либо детектор, но не оба вместе.

К самотестируемому детектору предъявлялись следующие требования:

1) при появлении на выходе схемы (или на входе детектора) в некоторый момент времени t кодового слова детектор должен выдать соответствующий сигнал;

2) в самом детекторе может произойти неисправность из рассматриваемого множества неисправностей V , которая должна быть обнаружима в рабочей области функционирования детектора, т.е. на множестве всех кодовых слов. Это означает, что должен существовать входной набор (кодированное слово (m, n) -кода), на котором эта неисправность проявляется на выходах детектора.

Самотестируемый детектор имеет два выхода, причём комбинации значений сигналов имеют следующие интерпретации:

а) (01) или (10) означают, что входной набор является кодовым словом (m, n) -кода и детектор исправен;

б) (00) или (11) означают, что либо входной набор не является кодовым, либо детектор неисправен.

Кодовые слова (m, n) -кода могут быть представлены дизъюнкцией конъюнкций ранга n . Обозначим эту ДНФ $D_n^m(X)$, где $X = \{x_1, x_2, \dots, x_n\}$ – множество переменных. Для представления всевозможных (m, n) -кодированных слов ранее [2] была предложена специальная формула разложения

$$D_n^m(X) = \sum_{i=0}^m D_g^i(X^1) D_s^{m-i}(X^2). \quad (1)$$

Назовем некоторое число k *основой разложения* (k может быть числом входов в ПЛБ), ДНФ D_g^i, D_s^{m-i} – *функциями разложения*. Первоначальное множество переменных X мощности n делится на два подмножества X^1 и X^2 : $|X^1| = g = \lceil n/2 \rceil$, $|X^2| = s = n - g$. Если $g > k$ и $s > k$, то формула (1) снова используется для каждой функции разложения D_g^i, D_s^{m-i} , $i = \overline{0, m}$, и т.д. В результате мы имеем формулу A для всех кодовых слов (m, n) -кода, для наглядного представления которой удобно использовать соответствующее дерево.

Рассмотрим пример. Получим формулу A для D_{14}^7 , $k = 7$:

$$\begin{aligned} X^1 &= \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}, X^2 = \{x_8, x_9, x_{10}, x_{11}, x_{12}, x_{13}, x_{14}\}. \\ D_{14}^7 &= D_7^0(X^1)D_7^7(X^2) \vee D_7^1(X^1)D_7^6(X^2) \vee D_7^2(X^1)D_7^5(X^2) \vee D_7^3(X^1)D_7^4(X^2) \vee \\ &\vee D_7^4(X^1)D_7^3(X^2) \vee D_7^5(X^1)D_7^2(X^2) \vee D_7^6(X^1)D_7^1(X^2) \vee D_7^7(X^1)D_7^0(X^2). \end{aligned}$$

При выбранных значениях m, n и k достаточно одного шага разложения с использованием формулы (1).

Обычно детектор строится на той же элементной базе, что и схема. Часто построение детектора основано на определении веса входных кодовых слов. Для этой цели применяются схемы, основанные либо на пороговых схемах, либо на параллельных счетчиках [3, 4]. Такие подходы концептуально исключают построение детектора, ориентированного на подмножества кодовых слов, в то время как на выходах самопроверяемого устройства, к которому подключен детектор равновесных кодовых слов, не всегда достигаются всевозможные кодовые слова. Число l достигаемых кодовых слов может быть существенно меньше числа всевозможных кодовых слов. Из сказанного следует, что некоторые неисправности самотестируемого детектора, подключенного к выходам самопроверяемого устройства, могут оказаться необнаружимыми, так как обнаруживающие неисправности кодовые слова не поступают на входы детектора во время работы самопроверяемого устройства. Необнаруженная неисправность детектора может привести в дальнейшем к тому, что детектор не обнаружит некодовое слово, поступающее на него с выходов самопроверяемого устройства, вследствие неисправности, возникшей в устройстве. Эта неисправность появилась после неисправности в детекторе. Такая ситуация в дискретной системе, состоящей из самопроверяемого устройства и детектора кодов, недопустима.

Выходом из нее может быть дублирование детекторов и сравнение их реакций. Предполагается, что каждый раз неисправность появляется либо в одной из копий детектора, либо в самопроверяемом устройстве. Поскольку возникновение одинаковых неисправностей в копиях детектора маловероятно, такая система способна надежно работать в условиях, когда не все кодовые слова достигаются на выходах самопроверяемого устройства. Недостатком дублирования являются большие аппаратные затраты (более 100%).

Итак, если мы не хотим дублировать детекторы равновесных кодов, то необходимо обеспечение свойства самотестируемости на тех кодовых словах, которые достигаются в процессе работы самопроверяемого дискретного устройства. Реализация такого подхода возможна при синтезе самопроверяемого последовательностного дискретного устройства на этапе кодирования его состояний. Зная, для каких именно кодовых слов построен самотестируемый детектор, мы можем закодировать состояния именно этими кодовыми словами.

Построение самотестируемого детектора (m, n) -кодов для произвольного числа l кодовых слов основано на решении двух подзадач:

1. Представление множества кодовых слов мощности l объединением подмножеств равновесных кодовых слов, так что каждое из подмножеств задается существенным поддеревом дерева формулы А.
2. Построение самотестируемого детектора на базе полученного представления.

Метод представления множества мощности l описан в [5, 6]. Метод построения самотестируемого детектора для подмножества кодовых слов тот же, что и метод построения самотестируемого детектора для полного множества кодовых слов, и основан на покрытии дерева разложения программируемыми логическими блоками [1]. В данной статье выделяются свойства программируемых логических блоков, на входы которых поступают подмножества кодовых слов специального вида.

В дальнейшем будем предполагать, что число k входов в ПЛБ не меньше 6.

1. Выявление свойств программируемых логических блоков и подсхем из них, реализующих кодовые слова равновесного кода

Определение 1. Назовем функцию, представляемую ДНФ D_p^q , $0 < q < p$ функцией типа 1.

Пример такой функции для D_3^2 приведен в табл. 1. В ней столбцы сопоставлены с булевыми переменным, а строки представляют единичные наборы булевой функции (булевы векторы).

Т а б л и ц а 1

Пример функции типа 1

x_1	x_2	x_3
0	1	1
1	0	1
1	1	0

Определение 2. Функцией типа 2 назовём функцию, которая удовлетворяет следующим условиям: в каждом столбце таблицы, представляющей множество ее единичных наборов, единичная компонента присутствует один и только один раз, и в каждой строке число единичных компонент одинаково.

Пример функции типа 2 приведен в табл. 2, которая построена аналогично табл. 1.

Т а б л и ц а 2

Пример функции типа 2

x_1	x_2	x_3	x_4	x_5	x_6
0	0	0	0	1	1
0	0	1	1	0	0
1	1	0	0	0	0

Определение 3. Функцией типа 3 назовём функцию, область единичных значений которой представлена двумя взаимно инверсными векторами.

Пример функции типа 3 приведен в табл. 3, построенной аналогично таблице функции типа 1.

Т а б л и ц а 3

Пример функции типа 3

x_1	x_2	x_3	x_4
0	0	0	0
1	1	1	1

Теорема 1. Для выхода одновыходного ПЛБ, реализующего функцию типа 1, 2 или 3, и кратной неисправности на входных и выходном полюсах ПЛБ либо существует тестовый набор из области единичных значений функции типа 1, 2 или 3, либо кратная неисправность проявляет себя как неисправность «константа 1» на выходе ПЛБ. Доказательство теоремы приведено в [7].

Теорема 2. Если на одном из выходов двухвыходного блока реализуется функция типа 1, 2 или 3, то какой бы ни была вторая функция, сопоставляемая блоку, кратная неисправность на входных полюсах этого блока либо обнаруживается на соответствующем функции типа 1, 2 или 3 тестовом наборе из области единичных значений функции, либо проявляется на этом выходе как константа 1. Доказательство теоремы приведено в [7].

Рассмотрим систему из двух булевых функций с непересекающимися множествами наборов их единичных значений. Будем говорить, что система состоит из ортогональных булевых функций. Обозначим через M объединение множеств единичных наборов функций этой системы.

Теорема 3. Если множество M системы ортогональных булевых функций, реализуемой двухвыходным блоком, представляется таблицей функции типа 1 (2), то кратная неисправность на входах этого блока либо обнаруживается на одном из его выходов при поступлении на входы блока тестового набора из M , либо проявляется на одном из выходов как константная неисправность 1. Доказательство теоремы приведено в [7].

Отметим, что при покрытии программируемыми блоками дерева формулы A [1] с целью построения схемы самотестируемого детектора для всех кодовых слов (покрытие выполняется при просмотре дерева от концевых вершин к корню) сначала используются ПЛБ, реализующие функции D_k^q для соответствующих значений k и q . Здесь k – количество входов в программируемый логический блок, а q – число единичных компонент вектора из области единичных значений функции, реализуемой блоком. Затем используются ПЛБ, реализующие функции типа 2, и только они.

При покрытии подмножества существенных поддеревьев дерева формулы A программируемыми блоками, реализующими функции D_k^q , может оказаться, что на входы блока поступают только некоторые кодовые слова из множества, представляемого D_k^q . В этом случае мы не можем гарантировать свойство самотестируемости построенной схемы детектора, опираясь на приведенные выше теоремы и метод синтеза [1]. В такой ситуации приходится исследовать подмножества кодовых слов, поступающих на входы ПЛБ. Будем использовать как одновыходные, так и двухвыходные ПЛБ.

Обозначим через M всё множество кодовых слов (q, k) -кода, реализуемого программируемым логическим блоком, а через M_l – его подмножество мощности l , такое, что в каждом столбце таблицы, содержащей все векторы подмножества M_l , присутствуют как нулевые, так и единичные значения.

Определение 4. Назовем функцию, область единичных значений которой совпадает с подмножеством M_l , функцией типа 1.

Пример такой функции приведен в табл. 4.

Т а б л и ц а 4

Пример функции типа 1' ($l = 11$)

x_1	x_2	x_3	x_4	x_5	x_6
0	0	0	1	1	1
0	0	1	0	1	1
0	1	0	1	0	1
0	1	0	1	1	0
0	1	1	0	0	1
1	0	1	0	1	0
1	0	1	1	0	0
1	1	0	0	0	1
1	1	0	0	1	0
1	1	0	1	0	0
1	1	1	0	0	0

Пусть ПЛБ реализует функцию D_k^q . Здесь k – количество входов в программируемый логический блок, а q – число единичных компонент, $0 < q < k$. Пусть на вход программируемого логического блока приходит некоторое подмножество M_l кодовых слов (q, k) -кода.

Теорема 4. Если на входы ПЛБ, реализующего множество M всех кодовых слов (q, k) -кода, $0 < q < k$, подаётся их подмножество M_l , то для кратной неисправности на входах ПЛБ либо существует тестовый набор из M_l , либо кратная неисправность проявляет себя как неисправность «константа 1» на выходе ПЛБ.

Доказательство теоремы приведено в [8].

Теорема 5. Если на одном из выходов двухвыходного блока реализуется функция типа 1', то какой бы ни была вторая функция, сопоставляемая блоку, для кратной неисправности на входах ПЛБ либо существует тестовый набор из M_l , такой что неисправность проявляется на выходе блока, реализующего функцию типа 1', либо неисправность проявляется на этом выходе как константа 1.

Доказательство теоремы приведено в [8].

Рассмотрим систему из двух ортогональных булевых функций, в которой M есть объединение множеств единичных наборов функций этой системы.

Теорема 6. Если множество M системы ортогональных булевых функций, реализуемой двухвыходным блоком, представляется таблицей функции типа 1', то для кратной неисправности на входах ПЛБ либо существует тестовый набор из M , такой что неисправность проявляется на одном из выходов блока, либо неисправность проявляется на одном из выходах блока как константа 1.

Доказательство. Утверждение следует из того факта, что множества переменных функций одного и того же блока совпадают, а сами функции реализуются на отдельных блоках памяти (LUT). На втором выходе блока кратная неисправность может вообще не проявляться. Это зависит от вида функции, реализуемой вторым выходом.

Теорема доказана.

Воспользуемся приведенными выше теоремами для анализа заданного множества кодовых слов на возможность построения для него самотестируемого детектора.

Алгоритм анализа подмножества кодовых слов.

1) Разбиваем множество n переменных на w подмножеств так, как это делается при построении формулы A для заданного k . В результате подмножество кодовых слов разбивается на w множеств фрагментов булевых векторов. Длина каждого фрагмента булева вектора не более k . Анализируем векторы каждого фрагмента X^i , $i = \overline{1, w}$.

2) Рассмотрим фрагмент, сопоставляемый X^i . Разбиваем его векторы на части по весу.

3) Для каждой части, которая не совпадает ни с функцией типа 1, ни с функцией типа 1', находим такую часть, что пара представляющих обе части ортогональных функций задает единичные наборы функции типа 1'. Каждая часть может входить только в одну пару.

4) Если необходимое количество пар построить не удастся, то самотестируемость детектора рассматриваемым методом синтеза невозможно обеспечить. **ВЫХОД.**

5) Иначе $i = i + 1$.

6) Если $i < w$, переходим в п. 2.

7) При $i = w$ самотестируемость детектора обеспечивается. **ВЫХОД.**

Покажем, что для приведенного выше подмножества кодовых слов, $l = 11$, можно построить детектор, используя данный алгоритм. Это подмножество кодовых слов является частью кодовых слов, представляемых ДНФ D_6^3 . Воспользуемся разложением по формуле (1):

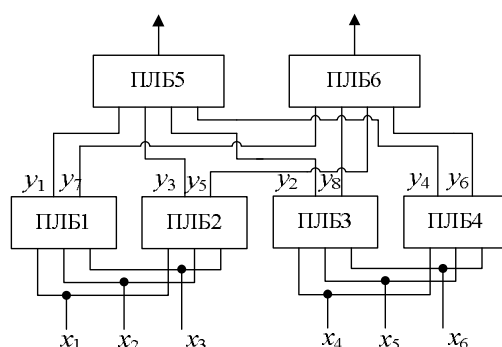
$$D_6^3(x_1, x_2, x_3, x_4, x_5, x_6) = D_3^0(x_1, x_2, x_3)D_3^3(x_4, x_5, x_6) \vee D_3^1(x_1, x_2, x_3)D_3^2(x_4, x_5, x_6) \vee D_3^2(x_1, x_2, x_3)D_3^1(x_4, x_5, x_6) \vee D_3^3(x_1, x_2, x_3)D_3^0(x_4, x_5, x_6). \quad (2)$$

Анализируя табл. 5, видим, что на некоторые ПЛБ, реализующие кодовые слова ДНФ D_k^q , поступают не все кодовые слова.

Описание элементов самотестируемого детектора для подмножества мощности $l = 11$

№ ПЛБ	Функция, реализуемая на 1-м выходе	Функция, реализуемая на 2-м выходе	Обозначение 1-го выхода	Обозначение 2-го выхода	Обеспечение самотестируемости
1	$D_3^0(x_1, x_2, x_3)$	$D_3^3(x_1, x_2, x_3)$	y_1	y_7	Система функций, реализуемая ПЛБ, удовлетворяет теореме 6
2	$D_3^1(x_1, x_2, x_3)$	$D_3^2(x_1, x_2, x_3)$	y_3	y_5	На вход ПЛБ, реализующего ДНФ $D_3^1(x_1, x_2, x_3)$, приходит неполное множество кодовых слов. На вход ПЛБ, реализующего ДНФ $D_3^2(x_1, x_2, x_3)$, приходит полное множество кодовых слов, т.е. реализуется система функций, удовлетворяющая теореме 2
3	$D_3^3(x_4, x_5, x_6)$	$D_3^0(x_4, x_5, x_6)$	y_2	y_8	Система функций, реализуемая ПЛБ, удовлетворяет теореме 6
4	$D_3^2(x_4, x_5, x_6)$	$D_3^1(x_4, x_5, x_6)$	y_4	y_6	На обоих выходах ПЛБ реализуются функции типа 1 (теорема 2)
5	$\overline{y_1 y_2 y_3 y_4} \vee \overline{y_1 y_2 y_3 y_4}$	–	Выход детектора		ПЛБ реализует функцию типа 2 (теорема 1)
6	$\overline{y_5 y_6 y_7 y_8} \vee \overline{y_5 y_6 y_7 y_8}$	–	Выход детектора		ПЛБ реализует функцию типа 2 (теорема 1)

Схема детектора представлена на рис. 1.

Рис. 1. Схема самотестируемого детектора для подмножества мощности $l = 11$

Однако не для любого подмножества кодовых слов можно обеспечить самотестируемость детектора предложенным подходом. Рассмотрим подмножество кодовых слов в табл. 6. Мощность l -го подмножества также равна 11. Для этого подмножества нельзя построить самотестируемый детектор.

Таблица 6

Пример подмножества, на котором нельзя обеспечить свойство самотестируемости детектора ($l = 11$)

x_1	x_2	x_3	x_4	x_5	x_6
0	0	0	1	1	1
0	0	1	0	1	1
0	0	1	1	0	1
0	0	1	1	1	0
0	1	0	0	1	1
0	1	0	1	0	1
0	1	0	1	1	0
1	0	0	0	1	1
1	0	0	1	0	1
1	0	0	1	1	0
1	1	0	1	0	0

Очевидно, свойство самотестируемости нарушается по пункту 4 алгоритма анализа подмножества кодовых слов.

Заключение

Выявлены свойства ПЛБ, позволяющие построить самотестируемый детектор для заданного подмножества кодовых слов. Предложен алгоритм анализа подмножества кодовых слов с целью построения самотестируемого детектора, реализующего это подмножество.

ЛИТЕРАТУРА

1. Lala P.K. Self-Checking and Fault-Tolerant Digital Design. Morgan Kaufmann Pub. 2000.
2. Матросова А.Ю., Никитин К.В. Синтез самотестируемого детектора (m, n) -кодов на программируемых логических блоках // Вестник Томского государственного университета. Приложение. 2003. № 6. С. 124–136.
3. Anderson D.A., Metze G. Design of totally self-checking check circuits for m -out-of- n codes // IEEE Trans. Computers, C-22. 1973. P. 263–269.
4. Marouf M.A., Friedman A.D. Efficient design of sel-checking checker for any m -out-of- n code // IEEE Trans. Computers. 1978. C-27. P. 482–490.
5. Butorina N., Burkatovskaya J. The properties of the essential subtrees of a graph representation of the (m, n) -codewords // Proceedings of the 7th International Forum on Strategic Technology (IFOST 2012) IEEE Computer Society, 2012. P. 472–476.
6. Butorina N., Ostanin S. Implementation by the Special Formula of an Arbitrary Subset of Code Words of (m, n) -code for Designing of a Self-Testing Checker // Proceeding of IEEE East-West Design & Test Symposium. IEEE. 2011. P. 255–258.
7. Буркатовская Ю.Б., Буторина Н.Б., Матросова А.Ю. Синтез самотестируемых детекторов произвольного числа равновесных кодов // Вестник Томского государственного университета. Приложение. 2006. № 17. С. 190–197.
8. Буторина Н.Б. Свойства программируемых логических блоков, реализующих кодовые слова равновесных кодов // Динамика сложных систем – XXI век. 2015. Т. 9, № 4. С. 50–53.

Буторина Наталья Борисовна. E-mail: nnatta07@mail.ru

Национальный исследовательский Томский государственный университет

Поступила в редакцию 21 сентября 2016 г.

Butorina Natalia B. (National Research Tomsk State University, Russian Federation).

The properties of programmable logic blocks realizing code words of (m, n) -code.

Keywords: self-checking circuit; (m, n) -code; self-testing checker; CLB.

This paper deals with a problem of self-testing checker (STC) design for a subset of code words of (m, n) -code.

A self-testing (m, n) -code checker is used in concurrent error detection (CED) technique under suggestion that all code words appear on its inputs. It is supposed that either the circuit or the checker may be fault but not both. In practice only some code words are reachable on outputs of self-checking circuit and consequently on inputs of self-testing checker. In that case some faults of the checker can be undetectable as the proper code words do not appear on the circuit outputs. As a result, wrong code word on the fault self-checking circuit may be not detected by the checker. That is why it is necessary to provide the self-testing property of the checker for the corresponding sub-set of all code words of the same weight.

A method of a self-testing (m, n) -code checker design based on using of CLBs was proposed before. CLBs were supposed to be realized in the frame of the LUT-technology and, hence, could be programmed to implement any Boolean functions of a fixed number of variables. The set V of the checker faults includes all multiple stuck-at faults at the CLB inputs. Only one CLB in the checker can be fault.

The self-testing checker has to satisfy the following conditions:

- 1) if a non-code word appears at the output of the circuit (or at the input of the checker) then the checker should give the appropriate signal;
- 2) any fault from the set V should be detected in the set of all code words. It means that there should be a code word for which the fault manifests itself at the checker outputs.

The self-testing checker has two outputs with the following meanings of signals combinations:

- a) (01) or (10) mean that the input word is a word of the (m, n) -code and the checker is faultless;
- b) (00) or (11) mean that either the input word is non code one or the checker is fault.

To represent all possible (m, n) -code words, a special decomposition formula was proposed

$$D_n^m(X) = \sum_{i=0}^m D_g^i(X^1) D_s^{m-i}(X^2).$$

The decomposition formula is used several times. The result of its application is represented by the corresponding tree. The design of a self-testing checker for arbitrary number l of code words of (n, m) -code is based on the decision of two tasks:

1. Getting representation of l code words by sum of code words presented by proper sub-tree of the tree.
2. Deriving the self-testing checker based on the received representation.

Here the second task of the problem is considered.

The properties of programmable logic blocks which inputs receive a sub-set of code words of special type are described. Based on the properties the algorithm of analysis of the given sub-set is developed. It allows finding out if it is possible to design a self-testing checker or not.

REFERENCES

1. Lala, P.K. (2000) *Self-Checking and Fault-Tolerant Digital Design*. Morgan Kaufmann.
2. Matrosova, A.Yu., Nikitin, K.V. (2003) Sintez samotestiruемого детектора (m, n)-кодов на программируемых логических блоках [Design of a self-testing (m,n)-code checkers using configurable logic blocks]. *Vestnik Tomskogo gosudarstvennogo universiteta. Prilozhenie – Tomsk State University Journal. Supplement*. 6. pp. 124–136.
3. Anderson, D.A. & Metze, G. (1973) Design of totally self-checking check circuits for m-out-of-n codes. *IEEE Trans. Computers*. C-22. pp. 263–269. DOI: 10.1109/T-C.1973.223705
4. Marouf, M.A. & Friedman, A.D. (1978) Efficient design of sel-checking checker for any m-out-of-n code. *IEEE Trans. Computers*. C-27. pp. 482–490. DOI: 10.1109/TC.1978.1675138
5. Butorina, N. & Burkatovskaya, J. (2012) The properties of the essential subtrees of a graph representation of the (m,n) –codeword. *Proceedings of the 7th International Forum on Strategic Technology (IFOST 2012)*. IEEE Computer Society. pp. 472–476
6. Butorina, N. & Ostanin, S. (2011) Implementation by the Special Formula of an Arbitrary Subset of Code Words of (m, n)-code for Designing of a Self-Testing Checker. *Proceeding of IEEE East-West Design&Test Symposium*. IEEE. pp. 255–258
7. Burkatovskaya, Yu.B., Butorina, N.B. & Matrosova, A.Yu. (2006) Sintez samotestiruемых детекторов произвольного числа равновесных кодов [Design of self-testing checkers of an arbitrary number of (m,n)-code words]. *Proceedings of the Vth Siberian scientific school-seminar with international participation "Computer Security and Cryptography" - SIBECRYPT'06*. Tomsk. pp. 190-197. (In Russian).
8. Butorina, N.B. & Sidendirzieva, S.P. (2008) Svoystva программируемых логических блоков, realizuyushchikh kodovye slova ravnoesnykh kodov [Design of self-testing checkers of (m,n)-code words]. *Dinamika slozhnykh sistem–XXI vek – Dynamics of Dif-ficult Systems–XXI*. pp. 44–44.