



<http://dx.doi.org/10.35596/1729-7648-2021-19-4-43-51>

Оригинальная статья
Original paper

УДК 004.33.054

НЕРАЗРУШАЮЩЕЕ ТЕСТИРОВАНИЕ ЗАПОМИНАЮЩИХ УСТРОЙСТВ НА БАЗЕ ДВОЙНЫХ АДРЕСНЫХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ

В.Н. ЯРМОЛИК¹, И. МРОЗЕК², В.А. ЛЕВАНЦЕВИЧ¹, Д.В. ДЕМЕНКОВЕЦ¹

¹Белорусский государственный университет информатики и радиоэлектроники
(г. Минск, Республика Беларусь)

²Белостоцкий технический университет (г. Белосток, Республика Польша)

Поступила в редакцию 3 февраля 2021

© Белорусский государственный университет информатики и радиоэлектроники, 2021

Аннотация. Анализируется эффективность применения классических неразрушающих тестов для тестирования запоминающих устройств (ЗУ) и их основные недостатки, среди которых выделяют большую временную сложность и низкую диагностическую способность. Определяется понятие двойной адресной последовательности 2A и приводятся примеры их формирования на базе счетчиковых адресных последовательностей и последовательностей кода Грея. Синтезируется базовый элемент неразрушающих тестов с применением двойных адресных последовательностей и исследуются его обнаруживающая и диагностическая способности для различных неисправностей ЗУ. Приводятся два новых неразрушающих теста ЗУ *March_2A_1* и *March_2A_2*, для которых оценивается их временная сложность и эффективность обнаружения неисправностей ЗУ. Показывается существенно меньшая временная сложность предложенных тестов и высокая диагностическая способность по сравнению с классическими неразрушающими тестами.

Ключевые слова: тестирование вычислительных систем, запоминающие устройства, маршевые тесты памяти, многократное тестирование, неразрушающее тестирование.

Конфликт интересов. Авторы заявляют об отсутствии конфликта интересов.

Для цитирования. Ярмолик В.Н., Мрозек И., Леванцевич В.А., Деменковец Д.В. Неразрушающее тестирование запоминающих устройств на базе двойных адресных последовательностей. Доклады БГУИР. 2021; 19(4): 43-51.

TRANSPARENT MEMORY TESTING BASED ON DUAL ADDRESS SEQUENCES

VYACHESLAV N. YARMOLIK¹, IRENEUSZ MROZEK², VLADIMIR A. LEVANTSEVICH¹,
DENIS V. DEMENKOVETS¹

¹Belarusian State University of Informatics and Radioelectronics (Minsk, Republic of Belarus)

²Bialystok University of Technology (Bialystok, Poland)

Submitted 3 February 2021

© Belarusian State University of Informatics and Radioelectronics, 2021

Abstract. An effectiveness of the application of classical non-destructive tests for testing storage devices and their main disadvantages, among which there are great time complexity and low diagnostic ability, are analysed. The concept of double address sequence $2A$ is defined and the examples of their formation based on counter address sequences and Gray code are provided. The basic element of non-destructive tests with the use of double address sequences is synthesized and its detecting and diagnostic abilities for different storage devices defects are explored. There are two new non-destructive tests of memory devices *March_2A_1* and *March_2A_2* and an estimation of their time complexity and efficiency of failure detection are given. A significantly lower time complexity of the proposed tests and their high diagnostic ability in comparison with classical non-destructive tests are shown.

Keywords: testing of computing systems, storage devices, march memory tests, repeated testing, transparent testing.

Conflict of interests. The authors declare no conflict of interest.

For citation. Yarmolik V.N., Mrozek I., Levantsevich V.A., Demenkovets D.V. Transparent memory testing based on dual address sequences. Доклады БГУИР. 2021; 19(4): 43-51.

Введение

Важным достижением в развитии функционального диагностирования запоминающих устройств (ЗУ) является разработка и применение методов неразрушающего тестирования [1–3]. Основное свойство данных методов заключается в сохранении данных, хранимых в памяти, после проведения процедуры тестирования. Первые системы неразрушающего периодического тестирования ЗУ использовали резервные модули ЗУ для временного хранения данных, что требовало больших временных затрат на перезаписывание содержимого ЗУ. Приведенные ограничения привели к появлению новых подходов к реализации неразрушающего тестирования ЗУ. Одной из первых является предложенная Б. Конеманом технология, основанная на применении сигнатурного анализа, что приводило к маскированию некоторых, в том числе и простейших, неисправностей [1]. Развитием неразрушающего тестирования ЗУ стала методика, предложенная М. Николаидисом (M. Nicolaidis), которая основывается на применении классических маршевых тестов [3, 4]. Однако реализация подобных неразрушающих тестов требует существенного увеличения их временной сложности, которое достигает 40–50%. Кроме того, данная методика из-за эффекта маскирования не позволяет получить 100% покрывающую способность даже для однократных неисправностей и имеет невысокую диагностическую способность из-за сложности получения информации о виде и месте возникшей неисправности ЗУ [1, 5].

Представленная статья посвящена новому подходу к построению неразрушающих тестов, основанному на применении адресных последовательностей с двукратным повторением адресов.

Анализ эффективности тестов запоминающих устройств

Следует выделить две особенности неразрушающих тестов. Во-первых, неразрушающие тесты строятся на базе классических маршевых тестов в силу того, что только в случае маршевых тестов достигается приемлемая временная сложность процедуры тестирования [1, 2]. Во-вторых, все существующие тесты, в том числе и неразрушающие, рассматриваются для ЗУ, содержащих N однобитных запоминающих элементов (ЗЭ), где $N = 2^m$.

В общем случае маршевый тест включает в себя коллекцию маршевых элементов, которые называются фазами теста [4]. Так, например, тест *March Y*: $\{\uparrow\downarrow(w0); \uparrow(r0, w1, r1); \downarrow(r1, w0, r0); \uparrow\downarrow(r0)\}$ состоит из четырех фаз.

Для формального описания неисправных состояний ЗУ используются математические модели их неисправностей, отражающие реальные физические дефекты ЗУ [2–4, 8, 9]. Наиболее сложными моделями неисправностей, обнаруживаемых маршевыми тестами и покрывающих более простые неисправности, являются кодочувствительные неисправности – *PSFk* (pattern sensitive faults) и их модификации в виде граничных кодочувствительных неисправностей *NPSFk*, в которых могут участвовать любые $k \leq 9$ из N ячеек ЗУ. Среди неисправностей *NPSFk* выделяют пассивные *NPSFk* (*PNPSFk*), в которых содержимое базовой ячейки нельзя изменить в зависимости от определенного набора в любых k из N ячеек [3, 4].

Число всевозможных *PNPSFk* для памяти емкостью N бит определяется согласно выражению [3]

$$Q_{TN}(PNPSFk) = k \cdot 2^k \cdot \binom{N}{k}. \quad (1)$$

А максимально возможное число $Q_{MAX}(PNPSFk)$ обнаруживаемых неисправностей *PNPSFk* при применении однократного маршевого теста равняется [3]

$$Q_{MAX}(PNPSFk) = (8 \cdot (k - 2) + 2 \cdot 4) \times \binom{N}{k} = 8 \cdot (k - 1) \cdot \binom{N}{k}. \quad (2)$$

Соответственно, однократные маршевые тесты не могут превосходить максимально возможную полноту покрытия *PNPSFk* неисправностей, определяемую выражением [3]

$$FC_{MAX}(PNPSFk) = \frac{Q_{MAX}(PNPSFk)}{Q_{TN}(PNPSFk)} \cdot 100 \% = \frac{k - 1}{k \cdot 2^{k-3}} 100 \%. \quad (3)$$

Последнее соотношение показывает предельные возможности классических маршевых тестов в части обнаружения сложных кодочувствительных неисправностей *PNPSFk*.

На примере теста *March Y* рассмотрим эффективность его оригинальной реализации и неразрушающей модификации по методике Николаидиса, приведенной в табл. 1 [3, 4].

Таблица 1. Две версии реализации теста *March Y*
Table 1. Transparent *March Y* test implementation

Описание теста Test description	Тест Test	Сложность теста The complexity of the test
Неразрушающий тест Transparent test	$\{\uparrow(rb, r\bar{b}); \downarrow(r\bar{b}, rb); \uparrow\downarrow(rb)\}$ $\{\uparrow(rb, w\bar{b}, r\bar{b}); \downarrow(r\bar{b}, wb, rb); \uparrow\downarrow(rb)\}$	12N

В описании неразрушающего теста, представленного в табл. 1, b принимает произвольное значение 0 или 1, а \bar{b} инверсное по отношению к b значение. Сам тест состоит из двух частей, а именно, *начального неразрушающего теста* $\{\uparrow(rb, r\bar{b}); \downarrow(r\bar{b}, rb); \uparrow\downarrow(rb)\}$, необходимого для получения эталонной сигнатуры S_F , и *базового неразрушающего теста* $\{\uparrow(rb, w\bar{b}, r\bar{b}); \downarrow(r\bar{b}, wb, rb); \uparrow\downarrow(rb)\}$ [1, 2]. После реализации базового теста формируется реальная сигнатура S_R , которая сравнивается с ранее полученной эталонной S_F . По результату сравнения принимается решение о наличии либо отсутствии неисправности в ЗУ. Несовпадение реальной сигнатуры S_R с ее эталонным значением S_F свидетельствует лишь о неисправном состоянии

памяти. Получение уточняющей информации о неисправности требует дополнительных временных затрат [2].

Для классической реализации теста *March Y* была показана 100 % полнота покрытия относительно простейших неисправностей типа *SAF* и *TF* [1, 2]. Однако для неисправности взаимного влияния *CFid* обнаруживаемыми являются только четыре из восьми их видов, а именно: $\langle \uparrow, 0 \rangle$, $\langle \downarrow, 0 \rangle$, $\langle 1, \uparrow \rangle$, $\langle 1, \downarrow \rangle$, что составляет только 50 % полноты покрытия неисправностей *CFid*. Еще меньшая полнота покрытия теста *March Y* достигается в классе сложных кодочувствительных неисправностей. Обнаруживаемыми неисправностями *PNPSFk* являются только два их вида: $\langle 0, 0, 0, \dots, 0, \uparrow, 1, 1, 1, \dots, 1 \rangle$ и $\langle 0, 0, 0, \dots, 0, \downarrow, 1, 1, 1, \dots, 1 \rangle$. Например, при $k = 3$ тест *March Y* обнаруживает $\langle 0, 0, \uparrow \rangle$, $\langle 0, \uparrow, 1 \rangle$, $\langle \uparrow, 1, 1 \rangle$, $\langle \downarrow, 1, 1 \rangle$, $\langle 0, \downarrow, 1 \rangle$ и $\langle 0, 0, \downarrow \rangle$, что составляет 6 из $k \cdot 2^k = 3 \cdot 2^3 = 24$ неисправностей в k произвольных ячейках ЗУ. В процентном исчислении, полнота покрытия *PNPSF3* составляет только 25 %. Для произвольного k полнота покрытия неисправностей *PNPSFk* определяется выражением

$$FC_{March Y}(PNPSFk) = \frac{Q_{March Y}(PNPSFk)}{Q_{TN}(PNPSFk)} \cdot 100 \% = \frac{1}{2^{k-1}} \cdot 100 \% \quad (4)$$

Количественно эффективность обнаружения неисправностей *PNPSFk* неразрушающей версией теста *March Y* также оценивается выражением (4). Однако в отличие от классической реализации будут обнаруживаться *PNPSFk* неисправности вида $\langle b, b, b, \dots, b, \downarrow, \bar{b}, \bar{b}, \bar{b}, \dots, \bar{b} \rangle$ и $\langle \bar{b}, \bar{b}, \bar{b}, \dots, \bar{b}, \uparrow, b, b, b, \dots, b \rangle$, где символ \downarrow означает инвертирование текущего значения b ячейки ЗУ. Аналогичное утверждение справедливо и для неисправностей взаимного влияния *CF*, однако и в первом, и во втором случае возможно необнаружение указанных неисправностей из-за их отображения в конфигурации многократных ошибок, которые могут быть замаскированы при получении реальной сигнатуры [5].

Неразрушающие маршевые тесты на базе двойных адресных последовательностей

Для повышения эффективности применения маршевого теста используют подход, при котором тест повторяется несколько раз, но с разной последовательностью адресов на каждой итерации маршевого теста [11]. Использование различных адресных последовательностей исследовалось в рамках многократных тестов [2, 3, 11]. Основное внимание уделялось выбору вида адресных последовательностей и различных их модификаций [3, 11]. В настоящей работе предлагается использование модифицированных адресных последовательностей для построения нового класса неразрушающих тестов ЗУ, а также повышения эффективности неразрушающего тестирования ЗУ. Первоначально рассмотрим общие свойства адресных последовательностей и их модификацию для реализации неразрушающих тестов. Под счетчиковой адресной последовательностью $A_c = A_c(0)A_c(1)A_c(2)\dots A_c(N-2)A_c(N-1)$, где $A_c(j) \in \{0, 1, 2, \dots, N-1\}$, $j \in \{0, 1, 2, \dots, N-1\}$ и $N = 2^m$, понимают последовательность, в которой адреса $A_c = c_{m-1}c_{m-2}\dots c_2c_1c_0$, $c_i \in \{0, 1\}$, $i \in \{0, 1, 2, \dots, m-1\}$, формируются по правилам функционирования m -разрядного двоичного суммирующего счетчика [3].

Следует отметить, что в произвольной адресной последовательности $A = a_{m-1}a_{m-2}\dots a_2a_1a_0$, где $a_i \in \{0, 1\}$, $i \in \{0, 1, 2, \dots, m-1\}$, количество двоичных неповторяющихся комбинаций $a_{m-1}a_{m-2}\dots a_{i+1}a_{i-1}\dots a_2a_1a_0$ при $a_i = 0$ и количество комбинаций $a_{m-1}a_{m-2}\dots a_{i+1}a_{i-1}\dots a_2a_1a_0$ при $a_i = 1$ одинаково и равно 2^{m-1} [3].

Данное свойство можно обобщить в виде следующего утверждения.

Утверждение 1. Произвольная совокупность любых $m-1$ разрядов $a_{m-1}a_{m-2}\dots a_{i+1}a_{i-1}\dots a_2a_1a_0$ из m разрядов $a_{m-1}a_{m-2}\dots a_2a_1a_0$ исходной адресной последовательности A формирует адресную последовательность, в которой каждый $m-1$ разрядный адрес генерируется дважды.

В качестве примера в табл. 2 приведены подобные адресные последовательности для случая исходной счетчиковой последовательности $A_C = c_3c_2c_1c_0$ и последовательности кода Грея $A_G = g_3g_2g_1g_0$ для $m = 4$.

Таблица 2. Адресные последовательности A_C и A_G , и их $2A_C$ и $2A_G$ модификации
Table 2. Address sequences A_C and A_G , and their $2A_C$ and $2A_G$ modifications

$A_C =$ $= c_3c_2c_1c_0$	$2A_C =$ $= c_3c_2c_1$	$2A_C =$ $= c_3c_2c_0$	$2A_C =$ $= c_3c_1c_0$	$2A_C =$ $= c_2c_1c_0$	$A_G =$ $= g_3g_2g_1g_0$	$2A_G =$ $= g_3g_2g_1$	$2A_G =$ $= g_3g_2g_0$	$2A_G =$ $= g_3g_1g_0$	$2A_G =$ $= g_2g_1g_0$
0000 (0)	000 (0)	000 (0)	000 (0)	000 (0)	0000 (0)	000 (0)	000 (0)	000 (0)	000 (0)
0001 (1)	000 (0)	001 (1)	001 (1)	001 (1)	0001 (1)	000 (0)	001 (1)	001 (1)	001 (1)
0010 (2)	001 (1)	000 (0)	010 (2)	010 (2)	0011 (3)	001 (1)	001 (1)	011 (3)	011 (3)
0011 (3)	001 (1)	001 (1)	011 (3)	011 (3)	0010 (2)	001 (1)	000 (0)	010 (2)	010 (2)
0100 (4)	010 (2)	010 (2)	000 (0)	100 (4)	0110 (6)	011 (3)	010 (2)	010 (2)	110 (6)
0101 (5)	010 (2)	011 (3)	001 (1)	101 (5)	0111 (7)	011 (3)	011 (3)	011 (3)	111 (7)
0110 (6)	011 (3)	010 (2)	010 (2)	110 (6)	0101 (5)	010 (2)	011 (3)	001 (1)	101 (5)
0111 (7)	011 (3)	011 (3)	011 (3)	111 (7)	0100 (4)	010 (2)	010 (2)	000 (0)	100 (4)
1000 (8)	100 (4)	100 (4)	100 (4)	000 (0)	1100 (12)	110 (6)	110 (6)	100 (4)	100 (4)
1001 (9)	100 (4)	101 (5)	101 (5)	001 (1)	1101 (13)	110 (6)	111 (7)	101 (5)	101 (5)
1010 (10)	101 (5)	100 (4)	110 (6)	010 (2)	1111 (15)	111 (7)	111 (7)	111 (7)	111 (7)
1011 (11)	101 (5)	101 (5)	111 (7)	011 (3)	1110 (14)	111 (7)	110 (6)	110 (6)	110 (6)
1100 (12)	110 (6)	110 (6)	100 (4)	100 (4)	1010 (10)	101 (5)	100 (4)	110 (6)	010 (2)
1101 (13)	110 (6)	111 (7)	101 (5)	101 (5)	1011 (11)	101 (5)	101 (5)	111 (7)	011 (3)
1110 (14)	111 (7)	110 (6)	110 (6)	110 (6)	1001 (9)	100 (4)	101 (5)	101 (5)	001 (1)
1111 (15)	111 (7)	111 (7)	111 (7)	111 (7)	1000 (8)	100 (4)	100 (4)	100 (4)	000 (0)

В дальнейшем подобные адресные последовательности будем называть двойными адресными последовательностями ($2A$) в силу того, что каждый адрес, как это видно из табл. 2, формируется дважды. Возрастающую последовательность подобных адресов обозначим как $2\uparrow$, а убывающую – как $2\downarrow$. Для каждой адресной последовательности в табл. 2 приведены бинарные значения адресов и десятичные эквиваленты, представленные в скобках. Вид двойной адресной последовательности зависит как от выбранных разрядов исходной адресной последовательности A , так и от их перестановок, и, соответственно, общее количество последовательностей $2A$, полученных из исходной последовательности A , равняется $m!$. Очевидно, что число подобных последовательностей для реальных значений m велико, также как и разнообразие их свойств.

Основная идея неразрушающих маршевых тестов на базе двойных адресных последовательностей основана на том, что при двукратном инвертировании содержимого ячейки ЗУ его значение останется прежним. В соответствии с этим простейшим свойством операции инвертирования построим базовый элемент неразрушающего маршевого теста на базе двойной адресной последовательности $2A$. Так же, как и в классических неразрушающих тестах, маршевый элемент должен начинаться с операции чтения rb содержимого b текущей ячейки ЗУ. Это необходимо для однозначных действий с текущей ячейкой ЗУ, которые основываются на знании значения ее содержимого. Следующей операцией должна быть операция записи инверсного значения \bar{b} по отношению к только что прочитанному содержимому b , так как подобная операция является необходимым условием активизации неисправностей ЗУ. За операцией записи следует операция чтения этой же текущей ячейки ЗУ для проверки правильности выполнения операции инвертирования ее содержимого. Далее переходят к следующему запоминающему элементу, который соответствует следующему адресу адресной последовательности. Использование двойных адресных последовательностей $2A$ обеспечивает повторное инвертирование каждой ячейки ЗУ, сохраняя в итоге его исходное состояние. Таким образом, базовый элемент имеет следующий вид:

$$2 \uparrow (rb, \bar{wb}, rb). \tag{5}$$

Отметим, что использование в базовом элементе двойной адресной последовательности ($2\uparrow$) приводит к тому, что каждая ячейка ЗУ последовательно выполнит два перехода из b в \bar{b} , и наоборот, из \bar{b} в b , сохранив, таким образом, свое начальное значение. Правильность выполнения обоих переходов (\uparrow) и (\downarrow), а также операций чтения нулевых и единичных

значений обеспечивает вторая операция чтения rb базового элемента (5). Для иллюстрации реализации базового элемента (5), рассмотрим его применение для процедуры тестирования ЗУ, содержащего $N = 8$ запоминающих элементов с исходным содержимым $0\ 1\ 1\ 1\ 0\ 1\ 0\ 0$. В качестве двойной адресной последовательности используются последовательности адресов $2A_C = c_3c_1c_0$ и $2A_G = g_3g_1g_0$, приведенные в табл. 2. Пошаговое изменение содержимого ЗУ для обоих случаев двойной адресации приведено в табл. 3. На каждом шаге реализации базового элемента только одна запоминающая ячейка (выделена жирным шрифтом и курсивом) изменяет свое состояние на противоположное. После выполнения базового элемента (5) начальное состояние ЗУ осталось неизменным.

Базовый элемент на основе двойных адресных последовательностей позволяет синтезировать два неразрушающих маршевых теста (6).

$$\begin{aligned} \text{March}_{2A_1} &: \{\uparrow\downarrow(rb); 2\uparrow(rb, \overline{wb}, rb); \uparrow\downarrow(rb)\}, (8N); \\ \text{March}_{2A_2} &: \{\uparrow\downarrow(rb); 2\uparrow(rb, \overline{wb}, rb); 2\downarrow(rb, \overline{wb}, rb) \uparrow\downarrow(rb)\}, (14N); \end{aligned} \quad (6)$$

Таблица 3. Процедура реализации базового элемента (5) неразрушающего теста для двух видов адресации
Table 3. The procedure for implementing the basic element (5) of a transparent test for two types of addressing

ЗУ	Адреса Addresses	0	1	2	3	4	5	6	7	ЗУ	Адреса Addresses	0	1	2	3	4	5	6	7
	Содержимое Content	0	1	1	1	0	1	0	0		Содержимое Content	0	1	1	1	0	1	0	0
$2A_C =$ $= c_3c_1c_0$	000 (0)	1	1	1	1	0	1	0	0	$2A_G =$ $= g_3g_1g_0$	000 (0)	1	1	1	1	0	1	0	0
	001 (1)	1	0	1	1	0	1	0	0		001 (1)	1	0	1	1	0	1	0	0
	010 (2)	1	0	0	1	0	1	0	0		011 (3)	1	0	1	0	0	1	0	0
	011 (3)	1	0	0	0	0	1	0	0		010 (2)	1	0	0	0	0	1	0	0
	000 (0)	0	0	0	0	0	1	0	0		010 (2)	1	0	1	0	0	1	0	0
	001 (1)	0	1	0	0	0	1	0	0		011 (3)	1	0	1	1	0	1	0	0
	010 (2)	0	1	1	0	0	1	0	0		001 (1)	1	1	1	1	0	1	0	0
	011 (3)	0	1	1	1	0	1	0	0		000 (0)	0	1	1	1	0	1	0	0
	100 (4)	0	1	1	1	1	1	0	0		100 (4)	0	1	1	1	1	1	0	0
	101 (5)	0	1	1	1	1	0	0	0		101 (5)	0	1	1	1	1	0	0	0
	110 (6)	0	1	1	1	1	0	1	0		111 (7)	0	1	1	1	1	0	0	1
	111 (7)	0	1	1	1	1	0	1	1		110 (6)	0	1	1	1	1	0	1	1
100 (4)	0	1	1	1	0	0	1	1	110 (6)	0	1	1	1	1	0	0	1		
101 (5)	0	1	1	1	0	1	1	1	111 (7)	0	1	1	1	1	0	0	0		
110 (6)	0	1	1	1	0	1	0	1	101 (5)	0	1	1	1	1	1	0	0		
111 (7)	0	1	1	1	0	1	0	0	100 (4)	0	1	1	1	0	1	0	0		

В обоих тестах произвольный порядок адресов $\uparrow\downarrow$ для первой и последней операций чтения должен быть одинаков, возрастающий либо убывающий. Это связано с тем, что первая фаза тестов March_{2A_1} и March_{2A_2} используется для сжатия исходного состояния ЗУ и получения эталонной сигнатуры S_F , а их последняя фаза – для получения реального значения сигнатуры S_R после выполнения предыдущих базовых элементов. В случае проявления неисправностей в ходе выполнения базовых элементов их наличие будет определяться выполнением неравенства $S_F \neq S_R$.

Анализ эффективности неразрушающих тестов March_{2A_1} и March_{2A_2}

При допущении, что при реализации теста March_{2A_1} исходное содержимое ЗУ нулевое, т. е. для всех ячеек $b = 0$, а базовый элемент $2\uparrow(rb, \overline{wb}, rb)$ представляется двумя последовательными элементами $\uparrow(rb, \overline{wb}, rb)$ и $\downarrow(rb, \overline{wb}, rb)$, можно заключить об эквивалентности тестов March_{2A_1} и March_Y . Их эквивалентность заключается как во временной сложности, равной $8N$, так и в покрывающей способности различных типов неисправностей.

Как уже отмечалось ранее, базовый элемент (5) обеспечивает активизацию и обнаружение всех простейших неисправностей типа SAF и TF . Операция записи \overline{wb} и двойная адресация $2A$ обеспечивают условие активизации данных неисправностей, а их обнаружение

обеспечивает вторая операция чтения *rb*. Таким образом, для простейших неисправностей предложенный тест *March_2A_1*, в отличие от известных неразрушающих тестов, имеет максимальную диагностическую способность.

Аналогично максимально возможная диагностическая способность теста *March_2A_1* достигается и для случая сложных кодочувствительных неисправностей *PNPSFk*. Выполнение базового элемента в случае *PNPSFk* позволяет идентифицировать адрес базовой ячейки, которая не может выполнить один из переходов в этой ячейке для конкретного содержимого в соседних ячейках. Кроме того, тест *March_2A_1* позволяет достичь такого же значения полноты покрытия для *PNPSFk*, как и тест *March_Y*. И в данном случае обнаруживаются только два их вида $\langle u, u, u, \dots, u, \uparrow, u, u, u, \dots, u \rangle$ и $\langle d, d, d, \dots, d, \downarrow, d, d, d, \dots, d \rangle$, где $u, d \in \{0, 1\}$. Значения содержимого соседних ячеек *u* и *d* зависят как от их начального состояния, так и от вида последовательности адресов *2A*. Например, для случая ЗУ с 8 ячейками и неисправностей *PNPSF3* в ячейках с адресами 1, 3 и 5, применив адресную последовательность $2AC = c3c1c0$, базовый элемент (5) позволяет обнаруживать следующие неисправности: $\langle 1, 0, \uparrow \rangle$, $\langle 1, 1, \downarrow \rangle$, $\langle 1, \uparrow, 1 \rangle$, $\langle 1, \downarrow, 0 \rangle$, $\langle \uparrow, 1, 1 \rangle$ и $\langle \downarrow, 1, 1 \rangle$ (см. табл. 3). При тех же условиях изменение двойной адресной последовательности $2AC = c3c1c0$ на $2AG = g3g1g0$ приводит к обнаружению уже другого множества *PNPSF3*: $\langle 1, 1, \uparrow \rangle$, $\langle 1, 1, \downarrow \rangle$, $\langle 1, \uparrow, 0 \rangle$, $\langle 1, \downarrow, 0 \rangle$, $\langle \uparrow, 1, 1 \rangle$ и $\langle \downarrow, 1, 1 \rangle$ (см. табл. 3). Таким образом, однократное применение теста *March_2A_1* позволяет достичь полноты покрытия, равной 25 %, для *PNPSF3*.

Для обнаружения неисправностей взаимного влияния необходимо выполнить анализ состояния ячейки-жертвы после активизации конкретной неисправности, что невозможно в рамках базового элемента (5). Поэтому данные неисправности обнаруживаются по факту изменения четного количества инвертирования содержимого ячейки-жертвы базовым элементом (5) на нечетное количество. Это приводит к тому, что конечное состояние ЗУ будет отличаться от его исходного состояния, что приведет к выполнению неравенства $SF \neq SR$. Количественно полнота покрытия тестом *March_2A_1* таких неисправностей равняется полноте покрытия теста *March_Y*, как это видно, например, для *CFid* из экспериментальных данных, приведенных в табл. 4.

Таблица 4. Полнота покрытия неисправностей *CFid* тестом *March_2A_1* в процентах (%)
Table 4. Faults coverage of *CFid* faults by *March_2A_1* test in percent (%)

<i>CFid</i>	$2Ac_0$	$2Ac_1$	$2Ac_2$	$2Ac_3$	$2Ac_4$	$2Ac_5$	$2Ac_6$	$2Ac_7$	$2Ac_8$
$(0, \uparrow)$	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
$(1, \uparrow)$	100,00	100,00	100,00	100,00	100,00	100,00	100,00	100,00	100,00
$(0, \downarrow)$	0,00	0,39	1,18	2,75	5,88	12,16	24,71	49,80	100,00
$(1, \downarrow)$	100,00	99,61	98,82	97,25	94,12	87,84	75,29	50,20	0,00
$(\uparrow, 0)$	0,00	0,39	1,18	2,75	5,88	12,16	24,71	49,80	100,00
$(\uparrow, 1)$	100,00	99,61	98,82	97,25	94,12	87,84	75,29	50,20	0,00
$(\downarrow, 0)$	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
$(\downarrow, 1)$	100,00	100,00	100,00	100,00	100,00	100,00	100,00	100,00	100,00
<i>Total</i>	50,00	50,00	50,00	50,00	50,00	50,00	50,00	50,00	50,00

Как видно из приведенной таблицы, для памяти емкостью $N = 256$ общее (*Total*) количество обнаруживаемых неисправностей тестом *March_2A_1*, независимо от адресной последовательности *2A*, всегда равняется 50 %.

Неразрушающий маршевый тест *March_2A_2* отличается от *March_2A_1* наличием второго базового элемента с обратным порядком адресов *2A*, что расширяет его возможности обнаруживать сложные неисправности, сохраняя эффективность *March_2A_1* для более простых неисправностей. Действительно, для случая *PNPSFk*, дополнительно к $2k$ неисправностям вида $\langle u, u, u, \dots, u, \uparrow, u, u, u, \dots, u \rangle$ и $\langle d, d, d, \dots, d, \downarrow, d, d, d, \dots, d \rangle$, где $u, d \in \{0, 1\}$, тест *March_2A_2* в пределе будет обнаруживать еще $2k$ неисправностей вида $\langle u, u, u, \dots, u, \downarrow, u, u, u, \dots, u \rangle$ и $\langle d, d, d, \dots, d, \uparrow, d, d, d, \dots, d \rangle$. Например, тест *March_2A_1* с адресной последовательности $2Ac = c3c1c0$ обнаруживает *PNPSF3*: $\langle 1, 1, \uparrow \rangle$, $\langle 0, 0, \downarrow \rangle$, $\langle 0, \uparrow, 0 \rangle$, $\langle 1, \downarrow, 1 \rangle$, $\langle \uparrow, 1, 0 \rangle$ и $\langle \downarrow, 0, 1 \rangle$ для запоминающих ячеек 5, 6 и 7 (см. табл. 3). В то же время первый базовый элемент теста *March_2A_2* обнаруживает те же *PNPSF3*, а второй базовый элемент дополнительно

обеспечивает обнаружение следующих неисправностей: $\langle 0,0,\uparrow \rangle$, $\langle 1,1,\downarrow \rangle$, $\langle 1,\uparrow,1 \rangle$, $\langle 0,\downarrow,0 \rangle$, $\langle \uparrow,0,1 \rangle$ и $\langle \downarrow,1,0 \rangle$. Отметим, что удвоение обнаруживаемых тестом *March_2A_2* неисправностей *PNPSFk* достижимо только для случая, когда первый базовый элемент обнаруживает такие неисправности $\langle u,u,u,\dots, u,\uparrow,u,u,u,\dots,u \rangle$ и $\langle d,d,d,\dots,d,\downarrow,d,d,d,\dots,d \rangle$, для которых состояния соседних ячеек различны. В противном случае второй базовый элемент будет обнаруживать те же две неисправности $\langle u,u,u,\dots, u,\uparrow,u,u,u,\dots,u \rangle$ и $\langle d,d,d,\dots,d,\downarrow,d,d,d,\dots,d \rangle$, но в обратной последовательности. В то же время при увеличении значения k вероятность совпадения состояний ячеек – соседей в неисправностях $\langle u,u,u,\dots,u,\uparrow,u,u, u,\dots,u \rangle$ и $\langle d,d,d,\dots,d,\downarrow,d,d,d,\dots,d \rangle$ заметно уменьшается, что позволяет констатировать близость полноты покрытия *PNPSFk* к 50 %.

Заключение

Приведенный выше анализ свидетельствует о высокой покрывающей способности новых неразрушающих тестов, которая сравнима с покрывающей способностью *March Y* для *March_2A_1* и эффективностью двукратного теста *March Y* по отношению к *March_2A_2*. Кроме того, наличие базового элемента (5) в обоих тестах обеспечивает максимальную диагностическую способность, достижимую в рамках маршевых тестов для неисправностей *SAF*, *TF* и *PNPSFk*. Третьим несомненным достоинством предложенных маршевых тестов является их невысокая временная сложность.

Список литературы

1. Nicolaidis M. Theory of transparent BIST for RAMs. *IEEE Transactions on Computers*. 1996;45(10):1141-1156.
2. Ярмолик В.Н., Мурашко И.А., Куммерт А., Иванюк А.А. *Неразрушающее тестирование запоминающих устройств*. Минск: Бестпринт; 2005.
3. Ярмолик С.В., Занкович А.П., Иванюк А.А. *Маршевые тесты для самотестирования ОЗУ*. Минск: Бестпринт; 2009.
4. Goor A.J. *Testing Semiconductor Memories: Theory and Practice*. Chichester, UK: John Wiley & Sons; 1991.
5. Yarmolik V.N., Nicolaidis M., Kebichi O. Aliasing-Free Signature Analysis for RAM BIST. *IEEE International Test Conference*. 1994:368-377.
6. Bushnell M.L., Agrawal V.D. *Essentials of Electronic Testing for Digital, Memory & Mixed-Signal VLSI Circuits*. N.Y., USA: Kluwer Academic Publishers; 2000.
7. Goor A.J., Al-Ars Z. Functional Memory Faults: A Formal Notation and a Taxonomy. *IEEE VLSI Test Symposium (VTS'00)*. 2000:281-289.
8. Hamdioui S., Wadsworth R., Reyes J.D., Goor A.J. Memory Fault Modeling Trends: A Case Study. *Journal of Electronic Testing*. 2004;20(3):245-255.
9. Иванюк А.А. Моделирование функциональных неисправностей цифровых устройств средствами языка VHDL. *Информатика*. 2007;1:30-39.
10. Mrozek I., Yarmolik V.N. Multiple Control Random Testing. *Fundamenta Informaticae*. 2019;144(1):23-43.

References

1. Nicolaidis M. Theory of transparent BIST for RAMs. *IEEE Transactions on Computers*. 1996;45(10):1141-1156.
2. Yarmolik V.N., Murashko I.A., Kummert A., Ivaniuk A.A. [Transparent Memory Testing]. Minsk: Bestprint; 2005. (in Russ.)
3. Yarmolik V.N., Zankovich A.P., Ivaniuk A.A. [RAM Self-Test March Tests]. Minsk: Bestprint; 2009. (in Russ.)
4. Goor A.J. *Testing Semiconductor Memories: Theory and Practice*. Chichester, UK: John Wiley & Sons; 1991.
5. Yarmolik V.N., Nicolaidis M., Kebichi O. Aliasing-Free Signature Analysis for RAM BIST. *IEEE International Test Conference*. 1994:368-377.
6. Bushnell M.L., Agrawal V.D. *Essentials of Electronic Testing for Digital, Memory & Mixed-Signal VLSI Circuits*. N.Y., USA: Kluwer Academic Publishers; 2000.
7. Goor A.J., Al-Ars Z. Functional Memory Faults: A Formal Notation and a Taxonomy. *IEEE VLSI Test Symposium (VTS'00)*. 2000:281-289.
8. Hamdioui S., Wadsworth R., Reyes J.D., Goor A.J. Memory Fault Modeling Trends: A Case Study. *Journal of Electronic Testing*. 2004;20(3):245-255.

9. Ivaniuk A.A. [Modeling functional faults of digital devices using VHDL]. *Informatika = Informatics*. 2007;1:30-39. (in Russ.).
10. Mrozek I., Yarmolik V.N. Multiple Control Random Testing. *Fundamenta Informaticae*. 2019;144(1):23-43.

Вклад авторов

Ярмолик В.Н. сформулировал идею неразрушающих тестов на базе двойных адресных последовательностей.

Леванцевич В.А. и Деменковец Д.В. принимали участие в обобщении и анализе результатов.

Мрозек И. провел экспериментальные исследования.

Authors' contribution

Yarmolik V.N. formulated the idea of non-destructive tests based on double address sequences. Levantsevich V.A. and Demenkovets D.V. took part in the synthesis and analysis of the results. Mrozek I. conducted the experimental research.

Сведения об авторах

Ярмолик В.Н., д.т.н, профессор, профессор кафедры программного обеспечения информационных технологий Белорусского государственного университета информатики и радиоэлектроники.

Мрозек И., доктор, адъюнкт, адъюнкт Белостокского технического университета.

Леванцевич В.А., магистр технических наук, старший преподаватель кафедры программного обеспечения информационных технологий Белорусского государственного университета информатики и радиоэлектроники.

Деменковец Д.В., магистр технических наук, старший преподаватель кафедры программного обеспечения информационных технологий Белорусского государственного университета информатики и радиоэлектроники.

Information about the authors

Yarmolik V.N., D.Sc., Professor, Professor at the Department of Information Technology Software of the Belarusian State University of Informatics and Radioelectronics.

Mrozek I., D.Sc., Adjunct, Adjunct of the Bialystok University of Technology.

Levantsevich V.A., M.Sc., Senior Lecture at the Department of Information Technology Software of the Belarusian State University of Informatics and Radioelectronics.

Demenkovets D.V., M.Sc., Senior Lecture at the Department of Information Technology Software of the Belarusian State University of Informatics and Radioelectronics.

Адрес для корреспонденции

220013, Республика Беларусь,
г. Минск, ул. П. Бровки, 6,
Белорусский государственный университет
информатики и радиоэлектроники;
тел. +375-29-769-96-77;
e-mail: yarmolik10ru@yahoo.com
Ярмолик Вячеслав Николаевич

Address for correspondence

220013, Republic of Belarus,
Minsk, P. Brovka str., 6,
Belarusian State University
of Informatics and Radioelectronics;
tel. +375-29-769-96-77;
e-mail: yarmolik10ru@yahoo.com
Yarmolik Vyacheslav Nikolaevich