



УДК 681.5 + 621.3.049.771.14

DOI: 10.22184/NanoRus.2019.12.89.102.106

ОСОБЕННОСТИ ПРОЕКТИРОВАНИЯ СИСТЕМ СКАН-ТЕСТИРОВАНИЯ С КОМПРЕССИЕЙ

REGISTER REPLICATION IN EMBEDDED SCAN COMPRESSION SYSTEMS

ЛАДНУШКИН МАКСИМ СЕРГЕЕВИЧ

maxsl@cs.niisi.ras.ru

АРЯШЕВ СЕРГЕЙ ИВАНОВИЧ

ФГУ ФНЦ НИИСИ РАН

117218, Москва, Нахимовский просп., 36, к. 1

LADNUSHKIN MAXIM S.

maxsl@cs.niisi.ras.ru

ARYASHEV SERGEY I.

SRISA RAS

bld. 1, 36 Nakhimovskiy Ave., Moscow, 117218

Проведено исследование тестируемости логических сигналов цифровых СБИС и анализ эффективности метода дублирования функциональных триггеров по сокращению времени тестирования СБИС с аппаратной компрессией тестовых сигналов.

Ключевые слова: средства тестирования; скан-тестирование; дублирование триггеров; компрессия тестовых сигналов; моделирование.

The paper presents a new method of duplicating functional registers in order to reduce test time. It is shown that the method can improve testability by reducing fanouts on trigger's outputs.

Keywords: scan testing; rejecting; register duplicating; test compression; modeling.

ВВЕДЕНИЕ

Введение тестовых структур в СБИС позволяет увеличить тестовое покрытие и сократить время тестирования микросхемы. Встроенные средства тестирования создают дополнительный режим работы СБИС, в котором триггерная подсистема СБИС используется в качестве сдвигового регистра (скан-схемы) [1], что позволяет полностью контролировать, наблюдать состояние всех триггеров СБИС и использовать их для тестирования комбинационной части СБИС.

Рост объемов тестовых данных, вызванный увеличением количества триггеров и интегральной сложностью СБИС, является причиной встраивания аппаратных средств сжатия для сокращения времени тестирования [2], [3]. С ростом длин логических комбинационных путей и ростом количества сходящихся разветвлений снижается наблюдаемость и контролируемость узлов этих путей в режиме тестирования [4]. Для решения данной проблемы в определенные узлы СБИС встраиваются дополнительные тестовые схемы — тестовые точки контроля и обзора — дополнительные триггеры с управляющей логикой, позволяющие увеличить наблюдаемость и контролируемость отдельных узлов комбинационной подсистемы СБИС [5].

Современный метод установки тестовых точек, снижающий количество взаимных конфликтов неисправностей типа «залипание», позволяет сократить время тестирования в среднем в 2,2 раза для скан-схем с компрессией за счет увеличения количества неисправностей, тестируемых каждым тестовым вектором [6]. Однако установка тестовых точек уменьшает ресурс трассировки, увеличивает задержки распространения сигналов и увеличивает площадь тестовой логики, которая используется только в режиме тестирования и в рабочем режиме не функционирует. Современные методы создания тестовых точек используют существующие функциональные триггеры СБИС вместо дополнительных, что позволяет сократить аппаратурные затраты на тестовую логику, однако увеличения задержек

критических путей после установки тестовых точек избежать не удается [7], [8].

Метод создания копий (или дублирования) отдельных элементов известен как способ сокращения длин проводников при проектировании топологии СБИС. Дублирование отдельных логических элементов позволяет сократить задержки распространения сигналов [9]. В случае дублирования триггеров сам триггер и его копия в функциональном режиме находятся в одинаковом логическом состоянии в любой момент времени. Считается, что и в режиме скан-тестирования должно сохраняться равенство состояний этих триггеров, так как загрузка различных значений в дублированные скан-триггеры может привести к выходу микросхемы из строя в случае, если в схеме присутствуют логические элементы, допускающие сквозные токи [10]. Однако если таких элементов на кристалле немного или нет вовсе, то использование копий триггеров возможно.

ТЕОРЕТИЧЕСКИЙ АНАЛИЗ ТЕСТИРУЕМОСТИ ЛОГИЧЕСКИХ УЗЛОВ В РЕЖИМЕ СКАН-ТЕСТИРОВАНИЯ

Анализ тестируемости неисправностей логических узлов

Тестируемость логического узла (или вероятность обнаружения неисправности) комбинационной схемы может быть рассчитана как минимальное количество тестовых последовательностей, необходимых для обнаружения неисправности, деленное на максимальное количество уникальных входных воздействий [4]. Рассмотрим комбинационную схему с n входами $\vec{x} = (x_1, x_2, \dots, x_n)$ и m выходами $\vec{F} = (F_1, F_2, \dots, F_m)$. Пусть $g(\vec{x})$ — внутренний сигнал схемы, тогда тестируемость константных неисправностей типа «залипание-в-0» и «залипание-в-1» в сигнале g могут быть соответственно выражены [4], [11]

$$t(g/0) = S \left(g(\vec{x}) \sum_{j=1}^m \frac{\partial F_j}{\partial g} \right), \quad (1)$$

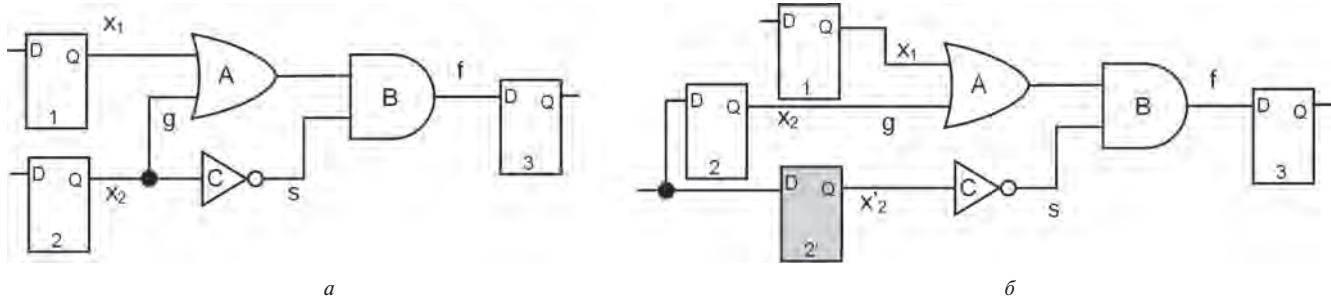


Рис. 1. Пример увеличения тестируемости в комбинационной схеме: а) исходная схема; б) модифицированная схема с дублированием триггера

$$t(g/0) = S\left(\bar{g}(\bar{x}) \sum_{j=1}^m \frac{\partial F_j}{\partial g}\right), \quad (2)$$

где $t(g/0)$ — тестируемость неисправности типа «залипание-в-0», а $t(g/1)$ — типа «залипание-в-1», $S(F)$ — синдром функции F :

$$S(F) = \frac{K}{2^n}, \quad (3)$$

где K — количество мinterмов функции F , а n — количество ее входов. Произведение функций в скобках — это пересечение множеств входных воздействий, которые задают необходимое значение в сигнале $g(\bar{x})$ и обеспечивают наблюдение этого значения на выходах \bar{F} . Если схема содержит сходящиеся разветвления, то это пересечение множеств может оказаться довольно небольшим, что в результате приводит к снижению величин $t(g/i)$, $i \in \{0; 1\}$.

Входные сигналы, имеющие разветвление на входах, могут быть разделены на отдельные независимые сигналы путем дублирования триггеров-источников, что позволяет сократить число сходящихся разветвлений в схеме, а значит, и увеличить тестируемость этой схемы.

Рассмотрим в качестве примера схему с двумя независимыми входами (см. рис. 1а), которые обозначены переменными x_1 и x_2 , и одним выходом — f . Данная комбинационная схема содержит сходящееся разветвление через узел x_2 . Тестируемость $t(g/0)$ сигнала $g = x_2$ согласно формуле (1) будет равна $t(g/0) = S(x_2 \bar{x}_1 \bar{x}'_2) = 0$.

Рассмотрим модифицированную схему с дублированием триггера 2 (см. рис. 1б). В данной схеме разветвление на выходе триггера 2 было разбито на два независимых логических сигнала, управляемых независимо триггерами 2 и 2'. Независимость управления достигается за счет загрузки различных значений в дублированные триггеры в процессе тестирования. Чтобы комбинационная схема была функционально эквивалентна исходной, входы D триггеров 2 и 2' объединяются. Оценка тестируемости неисправности $g/0$ для модифицированной схемы с дублированием триггера показала увеличение тестируемости $t(g/0)$, которая составила

$$t(g/0) = S(x_2 \bar{x}_1 \bar{x}'_2) = 1/8.$$

Анализ конфликтов неисправностей логических узлов

Взаимные конфликты неисправностей могут возникать в процессе генерации тестовых последовательностей, когда тестирование одного логического сигнала препятствует тестированию смежного сигнала, что приводит к созданию дополнительных тестовых векторов. Пусть s_0 — источник разветвления сигналов с ветвями s_1, \dots, s_n , тогда величины конфликтов выставления

0 и 1 в любой ветви s_k , $k \in [1; n]$ могут быть рассчитаны следующим образом [12]:

$$c_{s_k} = \min\{b_{s_k}; F_{s_k}\}, \quad (4)$$

$$C_{s_k} = \min\{B_{s_k}; f_{s_k}\}, \quad (5)$$

где c_{s_k} и C_{s_k} — количество конфликтов выставления логического «0» и «1» в узле s_k соответственно, b_{s_k} и B_{s_k} — необходимое количество логических состояний ветви s_k в «0» и «1» соответственно для обеспечения наблюдаемости неисправностей на всех остальных ветвях разветвления с источником s_0 , f_{s_k} и F_{s_k} — количество логических состояний ветви s_k в «0» и «1» соответственно, необходимое для наблюдения всех неисправностей, являющихся источником для сигнала s_0 , причем

$$F_{s_k} = F_{s_0} + \sum_{i=1}^n B_{s_i}, i \neq k, \quad (6)$$

$$f_{s_k} = f_{s_0} + \sum_{i=1}^n b_{s_i}, i \neq k. \quad (7)$$

Для разветвления сигнала s на выходе триггера значение $F_{s_0} = f_{s_0} = 0$, то есть $F_{s_k} = \sum_{i=1}^n B_{s_i}, i \neq k$, $f_{s_k} = \sum_{i=1}^n b_{s_i}, i \neq k$. Тогда если число ветвей разветвления сократить до $n = 1$, то $F_{s_k} = f_{s_k} = 0$. Отсюда $C_{s_k} = c_{s_k} = 0$ в данном разветвлении, то есть конфликтов неисправностей в нем не будет. Более того, для любого сигнала x в путях от s_0 до конечных приемников сигнала (триггеров или портов вывода) значения f_x и F_x всех разветвлений будут сокращены на $\sum_{i=1}^n b_{s_i}$ и $\sum_{i=1}^n B_{s_i}$ соответственно. Отсюда по формулам (4) и (5) получаем, что количество конфликтов c_x и C_x любого сигнала x может быть сокращено на $\sum_{i=1}^n B_{s_i}$ и $\sum_{i=1}^n b_{s_i}$ соответственно.

Дублирование триггера-источника разветвления позволяет сократить число ветвей разветвления до 1, значит, это приведет к снижению числа взаимных конфликтов во всех логически путях от этих триггеров.

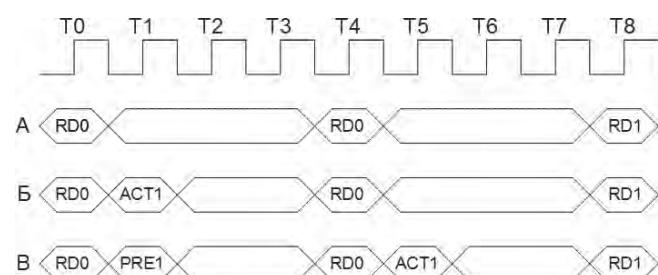


Рис. 2. Пример сокращения конфликтов неисправностей в комбинационной схеме: а) исходная схема; б) модифицированная схема с дублированием триггеров



На рис. 2а показан пример комбинационной схемы с взаимным конфликтом неисправностей в сигнале s . Отдельные части комбинационной схемы отмечены треугольниками, в которых количество неисправностей составляет M_i . По формулам (4)–(7) рассчитаем величины конфликтов в узлах s_1 и s_2 , получим:

$$c_{s_1} = \min\{b_{s_1}; F_{s_1}\} = \min\{0; \min\{M_2; M_1\}\} = 0,$$

$$C_{s_1} = \min\{B_{s_1}; f_{s_1}\} = \min\{M_3; M_4\},$$

$$c_{s_2} = \min\{b_{s_2}; F_{s_2}\} = \min\{M_4; \min\{M_1; M_2\} + M_3\},$$

$$C_{s_2} = \min\{B_{s_2}; f_{s_2}\} = \min\{0; 0\} = 0.$$

Согласно оценке возникает конфликт выставления «0» в узле s_2 и конфликт выставления «1» в узле s_1 . Для разрешения данного конфликта авторами [12] предлагается установить тестовую точку контроля ИЛИ-типа на сигнал s_2 , что приводит к снижению до 0 в данной схеме величин c_{s_2} и C_{s_1} .

Рассмотрим эквивалентную модифицированную схему с дублированием триггеров 1 и 2 (см. рис. 2б). Получается следующий набор выражений, характеризующий сигналы s_1 и s_2 в схеме:

$$c_{s_1} = \min\{0; 0\} = 0, \quad C_{s_1} = \min\{M_3; M_4\},$$

$$c_{s_2} = \min\{M_4; M_3\}, \quad C_{s_2} = \min\{0; 0\} = 0.$$

Сравнивая результаты оценок конфликтов неисправностей модифицированной и исходной схемы, можно заключить, что величина конфликта выставления «0» в узле s_2 теперь не зависит от переменных M_1 и M_2 , что приводит к снижению c_{s_2} на величину до $\min\{M_2; M_1\}$.

Так как предложенный метод дублирования триггера оказывает влияние на любой логический путь от этого триггера до конечных приемников сигналов за счет снижения величин взаимных конфликтов разветвления s на выходе этого триггера, то можно сделать вывод, что наибольший эффект от данного метода будет достигнут при дублировании триггеров

с наибольшим количеством комбинационных элементов в выходных путях этого триггера.

МЕТОДИКА ДУБЛИРОВАНИЯ ТРИГГЕРОВ В МАРШРУТЕ ПРОЕКТИРОВАНИЯ СБИС

Вариант дублирования всех триггеров, имеющих на выходе разветвление, приводит к чрезмерному росту площади. Таким образом, целесообразно выделить для дублирования только те триггеры, которые дают максимальный эффект увеличения тестируемости и сокращения количества конфликтов неисправностей. В данной работе анализируется алгоритм, который основан на поиске логических деревьев с наибольшим числом источников.

Алгоритм поиска триггеров-источников критических путей с порогом d_{max} по количеству элементов содержит следующую последовательность шагов:

- 1) задание количества триггеров для дублирования d_{max} ;
- 2) анализ схемы и получение множества сигналов L с низкой тестируемостью;
- 3) получение множества конечных источников S_i для каждого сигнала L_i ;
- 4) исключение из каждого множества S_i портов ввода-вывода и триггеров без разветвления на выходе;
- 5) сортировка множеств S_i по убыванию количества элементов;
- 6) последовательное добавление элементов из каждого множества S_i в множество D до тех пор, пока их количество не превысит d_{max} .

Результатом алгоритма является множество триггеров D , которые подлежат дублированию.

Дублирование триггеров осуществляется после того, как получена модель СБИС на уровне логических элементов (нет-лист). Сначала осуществляется поиск триггеров-источников критических путей с порогом d_{max} . Триггер, который имеет в выходном логическом пути элемент, допускающий сквозные токи, исключается из списка. Полученный список триггеров D дублируется, после чего осуществляется логическая оптимизация схемы, поскольку дублирование триггеров может привести к изменению задержек распространения сигналов. На основе оптимизированной схемы создается скан-схема с компрессией.

Таблица 1. Параметры модифицируемых СБИС

Проект	Количество триггеров, тыс.	Площадь, мм^2			Количество портов ввода/вывода
		Триггеров	Комбинационной логики	Общая	
cpu	96,2	1,04	2,98	10,59	840
eth	31,2	0,33	0,23	2,59	602
pcie_8x	98,3	1,05	0,90	3,18	3241
rio	65,0	0,71	0,91	1,96	1638
sata	14,0	0,15	0,21	0,47	2465
usb	25,8	0,28	0,32	4,11	617
2d	6,0	0,06	0,07	0,79	1857
proc	513,8	6,21	8,55	57,95	639
smpo	344,8	3,79	4,06	59,62	311
basis	187,6	2,20	2,77	27,28	226



Таблица 2. Параметры скан-схем СБИС

Проект	Длины скан-цепей скан-схем, триггеров		Количество введенных триггеров	Рост общей площади логики, %
	без дублирования	с дублированием		
cpru	586	598	1828	0,16
eth	612	622	505	0,21
pcie_8x	601	609	1411	1,18
rio	587	598	1276	0,97
sata	610	622	254	0,09
usb	602	618	684	0,14
2d	595	615	177	0,47
proc	346	351	7207	0,36
smpo	252	260	5018	0,09
basis	189	193	3725	0,14

Таблица 3. Результаты логического моделирования СБИС

Проект	Тестовое покрытие, %	Время тестирования скан-схем, с			Сокращение времени тестирования, %	
		без дублирования	с дублированием	с ТТ	с дублированием	с ТТ
cpru	93,6	0,284	0,264	0,26	7,0	8,5
eth	94,1	0,102	0,097	0,079	4,9	22,5
pcie_8x	95,3	0,322	0,309	0,2	4,0	37,9
rio	96,6	0,133	0,128	0,083	3,8	37,6
sata	86,2	0,050	0,037	0,023	26,0	54,0
usb	91,4	0,030	0,029	0,021	3,3	30,0
2d	79,6	0,025	0,016	0,012	36,0	52,0
proc	91,6	1,391	0,875	0,924	37,1	33,6
smpo	90,0	0,670	0,526	0,482	21,5	28,1
basis	91,5	0,196	0,195	0,175	0,5	10,7

Путем генерации и моделирования тестовых векторов для полученной схемы производится оценка времени тестирования и коэффициента тестового покрытия. Если необходимые параметры получены либо достигнут предел заполнения элементов на кристалле, то далее проектируется топология устройства. Если же необходимый коэффициент покрытия не достигнут либо есть необходимость сократить время тестирования и при этом пространство для дополнительных триггеров есть, можно изменить параметры скан-схемы [13] либо дополнительно дублировать триггеры в критических путях.

РЕЗУЛЬТАТЫ ЛОГИЧЕСКОГО МОДЕЛИРОВАНИЯ СБИС

Предложенная методика была использована при проектировании средств тестирования семи IP-блоков: ядра 64-разрядного микропроцессора (cpru), контроллера 2D-графики (2D), контроллера Ethernet 10/100/1000 Мбит/с (eth), контроллера PCI-E 2.0 8x (pcie_8x), контроллера последовательного RapidIO 4X с частотой передачи 3,125 Гбит/с (rio), контроллера SATA 3 Гбит/с (sata) и контроллера USB 2.0 (usb). Также согласно методике были спроектированы средства тестирования трех систем на кристалле (SoC): 64-разрядного микропроцессора с архитектурой «КОМДИВ»

и встроеннымми последовательными каналами RapidIO (proc), шестиканального коммутатора высокоскоростных последовательных каналов RapidIO 10 Гбит/с (smpo) и шестиканального коммутатора PCI Express 2.0 (basis). Вышеперечисленные СБИС отличаются количеством триггеров, объемом ОЗУ, количеством встроенных заказных блоков, количеством доменов синхросигналов, рабочими частотами, площадью комбинационной и триггерной логики. Все СБИС синтезировались на стандартных ячейках из библиотеки элементов TSMC с проектными нормами 65 нм, а также со встроенными интерфейсными приемопередатчиками, заказными блоками и блоками встроенной памяти ОЗУ. Параметры всех схем приведены в табл. 1.

Все вышеперечисленные СБИС были исследованы на предмет труднотестируемых узлов, в результате были получены множества узлов L с низкой тестируемостью для каждой СБИС.

Порог d_{max} по количеству дополнительно введенных элементов для нужд средств тестирования был задан равным 2% от числа всех триггеров СБИС, что является приемлемым значением роста тестовой логики [14]. После этого согласно вышеописанному алгоритму с заданным порогом d_{max} были получены множества триггеров D для каждой схемы, все элементы которых были дублированы.



Затем в каждой из полученных СБИС были созданы средства скан-тестирования (скан-схемы) с компрессией [15]. Описанный маршрут проектирования системы тестирования СБИС реализован в САПР Synopsys DFT Compiler. Параметры полученных схем, такие как количество внутренних скан-цепей и их длина, разрядность внешней шины тестовых данных, а также количество введенных триггеров в процессе дублирования триггеров, приведены в табл. 2. Рост площади логики СБИС, вызванный введением дополнительной логики, рассчитан после оптимизаций схем по площади и быстродействию. В среднем рост площади составил 0,38%.

Проведено логическое моделирование полученных скан-схем. Тестовые последовательности были созданы с помощью инструмента ATPG (Automatic Test Pattern Generation) — Synopsys Tetramax.

В итоге определены параметры скан-схем: тестовое покрытие и количество тестовых векторов. Исходя из частоты тестового синхросигнала 10 МГц, длины скан-цепей и количества векторов была рассчитана длительность тестирования каждой СБИС (см. табл. 3). Для тестирования скан-схем с дублированием функциональных триггеров потребовалось на 4,8–39,1% меньше тестовых последовательностей для достижения заданного тестового покрытия, чем исходным скан-схемам без дублирования, в среднем по всем исследуемым проектам на 16,1% меньше. С введением дополнительных триггеров выросли длины скан-цепей в схемах, что привело к увеличению длительности прохождения каждого тестового вектора. Тем не менее за счет сокращения количества векторов время тестирования сократилось на величины от 3,3% (IP-блок *gio*) до 37,1% (СБИС *proc*), в среднем на 14,4%.

Для сравнения эффективности метода дублирования функциональных триггеров с классическим методом установки тестовых точек (TT) [8] в каждую из исходных СБИС были установлены TT в целях увеличения тестового покрытия и снижения времени тестирования. Результаты моделирования всех схем с TT приведены также в табл. 3. Сокращение времени тестирования проектов после установки TT сократилось от 8,5% (IP-блок *sru*) до 54% (IP-блок *sata*), в среднем на 31,5%. Эффективность метода установки тестовых точек оказалась примерно в два раза выше эффективности метода дублирования триггеров, что является неплохим результатом, учитывая, что данные методы могут быть применены в рамках одного проекта независимо.

ЗАКЛЮЧЕНИЕ

Теоретический анализ testируемости логических узлов в режиме скан-тестирования показал, что предлагаемый метод создания копий отдельных функциональных триггеров позволяет увеличить testируемость схемы и сократить время тестирования за счет сокращения взаимных конфликтов неисправностей. Предложена методика дублирования триггеров в маршруте проектирования СБИС. Данный алгоритм дублирования триггеров был реализован в трех проектах СБИС и семи проектах IP-блоков при проектировании средств скан-тестирования с компрессией. Результаты показали снижение времени тестирования в среднем на 14,4% при аппаратурных затратах, не превышающих 1,2% общей площади СБИС. Сравнение предложенного метода с классическим методом установки тестовых точек показало примерно в два раза меньшую эффективность по снижению времени тестирования.

Авторы считают, что в данной работе новым является метод дублирования функциональных триггеров в целях снижения времени тестирования СБИС, а также сравнение предложенного метода с классическим методом установки тестовых точек. Результаты моделирования ряда проектов СБИС подтверждают эффективность метода.

Публикация выполнена в рамках государственного задания

ФГУ ФНЦ НИИСИ РАН по теме №0065-2019-0004

ЛИТЕРАТУРА

1. Abramovici M., Breuer M.A. and Friedman A.D. *Digital Systems Testing and Testable Design* // Computer Science Press, 1990. P. 364–366.
2. Touba N.A. *Survey of test vector compression techniques* // IEEE Design & Test of Computers. — July–August 2006. Vol. 23. №4. P. 294–303.
3. Kapur R. *Historical perspective on scan compression* // IEEE Design & Test of Computers. — March–April 2008. Vol. 25. №2. P. 114–120.
4. Savir J. *Good Controllability and Observability Do Not Guarantee Good Testability* // IEEE Transactions on Computers, 1983. V. 32. № 12. P. 1198–1200.
5. Pomeranz I., Reddy S. M. *Test-point insertion to enhance test compaction for scan designs* // Proc. ICDSN, 2000. P. 375–381.
6. Acer C., Feltham D., Liu Y., Moghaddam E., Mukherjee N., Patyra M., Rajski J., Reddy S. M., Tyszer J., Zawada J. *Embedded deterministic test points* // IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2017. P. 1–13.
7. Ren H., Kusko M., Kravets V., Yaari R. *Low cost test point insertion without using extra registers for high performance design* // Proc. ITC, 2009. P. 1–8.
8. Yang J., Touba N.A., Nadeau-Dostie B. *Test Point Insertion with Control Points Driven by Existing Functional Flip-Flops* // IEEE Transactions on Computers. 2012. V. 61. P. 1473–1483.
9. Srivastava A., Kastner R., Sarrafzadeh M. *Timing driven gate duplication: Complexity issues and algorithms* // Proc. ICCAD, 2000. P. 447–450.
10. Goessel M., Singh A., Sogomonyan E. *Scan-path with directly duplicated and inverted duplicated registers* // Proceedings of 20th IEEE VLSI Test Symposium. 2002. P. 47–52.
11. Savir J. *Syndrome-testable design of combinational circuits* // IEEE Transactions on Computers, 1980. V. 29. P. 442–451.
12. Liu Y., Moghaddam E., Mukherjee N., Reddy S. M., Rajski J., Tyszer J. *Minimal area test points for deterministic patterns* // Proc. ITC. — Nov. 2016. P. 1–7.
13. Ladnushkin M. S. *Snizhenie apparaturnykh zatrat i uvelichenie koeffitsienta kompressii sredstv testirovaniya konstantnykh neispravnostei KMOP tsifrovyykh SBIS* // VII Vserossiiskaya nauchno-tehnicheskaya konferentsiya «Problemy razrabotki perspektivnykh mikro- i nanoelektronnykh sistem — 2016». Sbornik trudov / pod obshch. red. akademika RAN A. L. Stempkovskogo. M.: IPPM RAN, 2016. Ch. 2. P. 68–75 (in Russian).
14. Acer C., Feltham D., Patyra M. et al. *On new test points for compact cell-aware tests* // IEEE Des. Test. — Dec. 2016. Vol. 33. №6. P. 7–14.
15. Wohl P., Waicukauski J.A., Ramnath S. *Fully X-Tolerant Combinational Scan Compression* // International Test Conference, 2007. P. 1–10.