

УДК 681.32

Е.Ю. ДаниловаПермский государственный национальный исследовательский университет,
Пермь, Россия**ДИАГНОСТИРОВАНИЕ ЛОГИКИ ПЛИС
С НОВЫМИ ЭЛЕМЕНТАМИ**

Программируемые логические интегральные схемы широко применяются в цифровой аппаратуре, в том числе в аппаратуре критического применения, например, в медицине, авионике, в системах управления космическими аппаратами, атомными электростанциями, в военной и специальной аппаратуре. В настоящее время количество логических элементов в наиболее передовых образцах ПЛИС приближается к десяткам миллионов. Поэтому диагностика логики ПЛИС ответственного применения является весьма актуальной задачей. Эта задача усложняется тем, что количество переменных возрастает, и сейчас уже имеются отдельные элементы, реализующие функции шести, семи и даже восьми переменных. Следует ожидать дальнейшего увеличения размерности задачи, решаемой в одном логическом элементе. Существующие методы диагностирования логики ПЛИС базируются на граничном сканировании с помощью внешней аппаратуры по стандарту IEEE 1149.1, активно внедряются встроенные средства тестирования по стандарту IEEE P1500. Как правило, используется сравнение результатов вычислений в нескольких логических элементах, и, если результаты вычислений в одном из них отличаются от результатов, например, группы их трех элементов, то делается вывод о наличии неисправности в этом отдельном элементе. Все это занимает относительно много времени. Для некоторых областей применения такой подход неприемлем. В то же время методы и средства самоконтроля, а также ускоренного тестирования проработаны не в полной мере. Это же касается особенностей диагностирования новых логических элементов так называемой «зелёной логики». Хорошо зарекомендовавшие себя генетические алгоритмы требуют доработки с учетом этих новых факторов для совершенствования встроенного тестирования ПЛИС. В статье анализируются методы и средства диагностирования логики, программируемых логических интегральных схем для высоконадёжного применения. Показывается, что контролепригодность логических элементов можно обеспечить на базе предложенных новых схемных решений и модернизированных генетических алгоритмов. Ставится задача совершенствования научно-методического аппарата диагностирования логики ПЛИС, использующей новые логические элементы, на основе генетических алгоритмов.

Ключевые слова: Техническая диагностика, достоверность, логический элемент.

E.Yu. Danilova

Perm State National Research University, Perm, Russian Federation

DIAGNOSIS OF FPGA WITH NEW ELEMENTS

Programmable logic integrated circuits are widely used in digital equipment, including equipment for critical applications, for example, in medicine, avionics, in control systems for spacecraft, atomic power stations, in military and special equipment. Currently, the number of logical elements in the most advanced FPGA samples is approaching tens of millions. Therefore, the diagnosis of logic FPGA

responsible use is a very urgent task. This task is complicated by the fact that the number of variables increases and now there are individual elements that implement the functions of six, seven and even eight variables. We should expect a further increase in the dimension of the problem solved in a single logical element. Existing methods for diagnosing FPGA logic are based on boundary scanning using external hardware according to the IEEE 1149.1 standard, and embedded test tools are actively being implemented according to the IEEE P1500 standard. As a rule, a comparison of the results of calculations in several logical elements is used, and if the results of calculations in one of them differ from the results, for example, the group of three elements, then it is concluded that there is a fault in this separate element. All this takes a relatively long time. For some applications, this approach is unacceptable. At the same time, methods and means of self-control, as well as accelerated testing, are not fully developed. The same applies to the features of diagnosing new logical elements of the so-called "Green Logic". Well-proven genetic algorithms require refinement to reflect these new factors in order to improve the built-in testing of FPGAs. Article investigates methods and tools for diagnosing the FPGA logic for highly reliable applications. Authors shows that advanced logical elements can improves logic reliability via diagnosis. The task is to perfect the methods for diagnosing the logic of the FPGA using new logic elements, based on genetic algorithms.

Keywords: Technical diagnosis, accuracy of control, logical element.

Введение. Надежность [1] является одним из важнейших показателей качества объектов. Для её обеспечения широко используют контроль и диагностику [2]. Программируемые логические интегральные схемы (ПЛИС) [3] широко используются в цифровой аппаратуре, в том числе в высоконадёжной [4, 5]. В настоящее время количество логических элементов и адаптивных логических модулей в ПЛИС достигает нескольких миллионов, например, в таких микросхемах фирмы «Интел», как Agilex, Stratix [6, 7]. Стоимость некоторых подобных «самых продвинутых» ПЛИС достигает десятков тысяч долларов, хотя имеются и ПЛИС среднего и «эконом» класса (тысячи логических элементов) за несколько десятков долларов. Можно сказать, что стоимость ПЛИС в каком-то смысле пропорциональна количеству логических элементов.

По мере сокращения технологических норм, а они приближаются к единицам нанометра, с увеличением сложности и размерности логических элементов (в настоящее время реализуются любые логические функции шести переменных и некоторые функции семи и даже восьми переменных), когда общее количество транзисторов составляет уже несколько миллиардов на одном кристалле, возрастают масштаб и сложность диагностирования и контроля.

Стандарт IEEE P1500 предполагает разработку ПЛИС со встроенными средствами диагностирования [8, 9, 10].

В то же время актуальные задачи импортозамещения электронной компонентной базы требуют разработки микросхем, которые не уступают мировым аналогам и могут быть использованы в специальных областях. Таковы, например, ПЛИС серии 5578, выпускаемые Во-

ронезским ОА КТЦ-«Электроника», которые не являются прямыми аналогами зарубежных ПЛИС, а микросхемы 5578ХС14Т и 24Т имеют свою архитектуру и свою среду конфигурирования [11]. Поэтому задачи совершенствования логики отечественных ПЛИС, в том числе путем аппаратно-программных средств диагностирования, являются крайне актуальными.

1. Анализ существующих методов и средств диагностирования логики ПЛИС. Тенденция разработки объектов, в том числе интегральных микросхем со встроенными средствами диагностического обеспечения [12], зародилась ещё в середине 80-х гг. XX в., когда были разработаны только первые поколения ПЛИС (программируемые логические матрицы ПЛМ, программируемая матрица логики ПМЛ и другие универсальные логические модули уже использовались с 70-х гг.).

Техническое диагностирование (Technical diagnosis) позволяет определить техническое состояние объекта. При этом возможны следующие ответы: работоспособен/неработоспособен (это в узком смысле иногда называют контролем) либо указываются место и тип неисправности (это в узком смысле диагностика).

Различают рабочее (ранее называвшееся функциональным, английского эквивалента нет в ГОСТ, можно сказать образно – онлайн-диагностирование) и тестовое техническое диагностирование (Testing) [2]. При рабочем диагностировании объект применяется по назначению (находится в рабочем состоянии [1]), а дополнительные технические средства, работая параллельно, оценивают его состояние [2]. Такое диагностирование наиболее затратное, но оно оправданно для систем критического применения, которые не могут быть остановлены для тестирования в процессе применения. При тестовом диагностировании объект по назначению не применяется (например, в процессе технического обслуживания или после фиксации неисправности с помощью некоторых дополнительных средств рабочего контроля), отключается от штатных входов-выходов, и специальные тестовые средства оценивают его состояние [2]. Для диагностики используется модель отказов (неисправностей). При этом отказ – событие, вызывающее состояние неисправности [1].

Одним из важнейших показателей является достоверность диагностирования (контроля) как степень объективного соответствия

результатов диагностирования (контроля) действительному техническому состоянию объекта [2, 12].

Стандарт IEEE 1149.1, аппаратуру, вводимую которым, часто называют по рабочей группе, его разработавшей, JTAG (the Joint Test Action Group) [13], предполагает специальный последовательный интерфейс для ввода диагностических воздействий (тестов) и вывода тестовых реакций из микросхемы.

Общая структура тестового диагностирования логики ПЛИС может быть описана в терминах стандарта IEEE 1500 [8], который предполагает Infrastructure Intellectual Property (I-IP) и уже начал применяться для систем на кристалле (SoC). Предполагается также парадигма тестопригодного проектирования Design For Test – DFT. Если стандарт IEEE 1149.1 предполагает тестирование «извне», то стандарт IEEE 1500 вводит тестирование «внутри» на основе встроенных генераторов тестовых воздействий Automatic Test Pattern Generation ATPG [9, 10] (рис. 1).

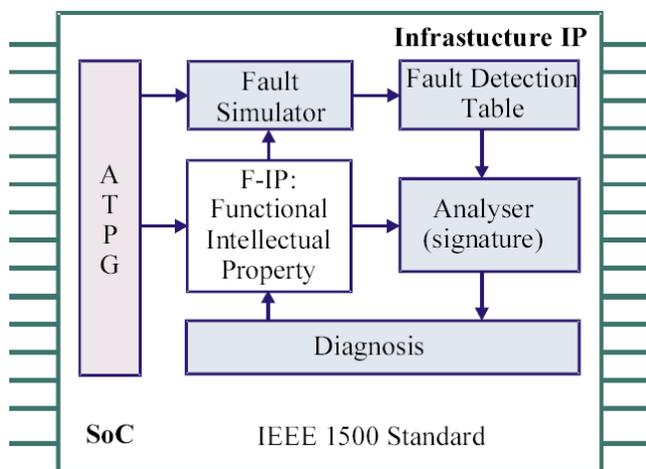


Рис. 1. Тестирование «внутри» по стандарту IEEE 1500

Таким образом, с помощью средства Fault Simulator моделируются (симулируются) неисправности, подаются тестовые наборы, получают ответы в виде таблицы Fault Detection Table. По таблице получают сигнатуры неисправностей. Если такие ответы будут получены без моделирования неисправностей, это будет свидетельствовать о соответствующем техническом состоянии инфраструктуры. Есть все основания ожидать в ближайшее время введения подобных средств и в ПЛИС.

Основой логики ПЛИС типа FPGA (Field-Programmable Gate Array, по-русски ППВМ, программируемая пользователем вентильная матрица) являются так называемые LUT (Look up Tables). LUT – это универсальный логический модуль, мультиплексор с настраиваемыми входами данных. Например, 1-LUT (на одну переменную) реализует любую функцию z одной переменной x :

$$z(x) = d_0 \cdot \bar{x} \vee d_1 \cdot x, \quad (1)$$

где d_0, d_1 – настройка функции $z(x)$, т.е. соответствующие значения из таблицы истинности функции $z(x)$.

На рис. 2 изображён такой 1-LUT.

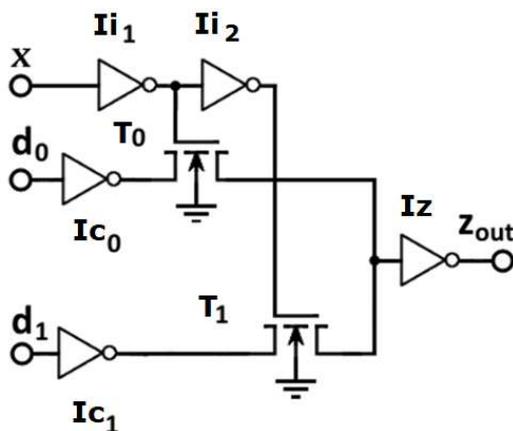


Рис. 2. 1-LUT, реализующий любую логическую функцию одной переменной по настройке (конфигурации) d_0, d_1

Обозначим передающие транзисторы (pass transistors) дерева T_0, T_1 . Инверторы $Ii_0, Ii_1; Ic_0, Ic_1; Iz$ на рис. 2 предназначены в основном для усиления сигналов, передаваемым по ветвям дерева и по матрице коммутаций (x – из других элементов, блоков или с внешних выводов; d_0, d_1 – из конфигурационной памяти SRAM – static random access memory, это по-русски оперативное запоминающее устройство ОЗУ), не указанной на рис. 2. Кроме того, инвертор Ii_0 обеспечивает собственно инверсию сигнала переменной x . Поэтому функцию (1) представим в развернутом виде:

$$z(x) = (d_0 \cdot Ic_0 \cdot \bar{x} \cdot Ii_0 \cdot T_0 \vee d_1 \cdot Ic_1 \cdot x \cdot Ii_0 \cdot Ii_1 \cdot T_1) \cdot Iz, \quad (2)$$

где $d_0; Ic_0; \bar{x}; Ii_0; T_0; d_1; Ic_1; x; Ii_1; Ii_1; T_1; Iz$ – переменные работоспособности 1-LUT, влияющие на результат вычислений.

Полный (тривиальный) тест заключается в переборе значений d_0, d_1, x и оценке значения $z(x)$. Но он позволяет зафиксировать только факт неисправности. Для определения места неисправности необходим перебор всех переменных в соответствии с моделью отказов (неисправностей) Fault Model [14]. Это могут быть константные неисправности на уровне логического элемента – константа 0, константа 1 (stuck-at faults at faults: stuck-at-0, stuck-at-1) $F_{0,1}$. Например, замыкание на шину питания или шину «Ноль вольт». Это могут быть неисправности на транзисторном уровне – постоянно открыт или постоянно закрыт (stuck on, stuck-closed; stuck-off, stuck-open) F_{T_0, T_1} . Это могут быть замыкания линий связи (мостящие неисправности – bridging faults), с учетом «проводных» логических операций И, ИЛИ + доминирование одних линий над другими (wired-AND, wired-OR, dominant) F_{brige} [14]. Кроме этих «логических» неисправностей есть ещё и параметрические, например, изменение временных задержек или других параметров элементов и схем. Безусловно, для такого простого элемента, как 1-LUT, нет смысла искать место неисправности. Как говорится, «отказал, так отказал!». Но при большем числе переменных можно в ряде случаев использовать хотя бы половину элемента [15]. С точки же зрения надёжности все переменные работоспособности соединены последовательно, и любая их неисправность приведёт к неисправности LUT. Можно ограничиться учетом только транзисторов (не учитывая надёжность линий связи), поэтому, принимая распределение, например Вейбулла [16], получим вероятность безотказной работы 1-LUT:

$$P(t)_{1-LUT} = e^{-12\lambda t^\alpha}; 1 \leq \alpha \leq 2, \quad (3)$$

12 – суммарное количество транзисторов, α – коэффициент распределения Вейбулла, λ – интенсивность отказов одного транзистора, t – время.

На самом деле, конечно, не каждая неисправность элементов на конкретном наборе приведет к неисправности $z(x)$ на выходе. Если необходимо учитывать внешние связи, надо включить в формулу (3) соответствующие интенсивности отказов, например, так:

$$P(t)_{1-LUT} = e^{-(12\lambda + \lambda_x + \lambda_{d_0} + \lambda_{d_1} + \lambda_z)t^\alpha}. \quad (4)$$

При отсутствии диагностики достоверность функционирования устройства равна вероятности безотказной работы. Обычно принимается допущение о том, что в объекте не более одного отказа, что позволяет снизить размерность задачи, но, однако, не всегда верно в случаях массивированного воздействия негативных факторов. Существующие методы диагностирования логики ПЛИС основаны, как правило, на внешнем тестировании в соответствии со стандартом IEEE 1149.1 (JTAG) [13], либо предлагаются варианты введения элементов стандарта IEEE P1500 [8]. В ряде случаев это требует загрузки нескольких конфигураций ПЛИС. Хороший обзор методов диагностирования логики дан в работе [17], в которой предложена новая встроенная аппаратура (Built-in Self-Test – BIST), обнаруживающая любые единичные неисправности в LUT каждого конфигурируемого логического блока CLB путём перекрестного сравнения результатов вычислений четырех LUT. При этом дополнительные затраты площади кристалла CLB оцениваются авторами в 19 %, а увеличение потребляемой мощности – на 25 %. При этом сами логические элементы LUT остаются неизменными. Используется также подход генетических алгоритмов в работах Ю.А. Скобцова [18, 19], Д.Е. Иванова. Такой подход, но с использованием новых элементов LUT, обеспечивающих как рабочее диагностирование [20], так и тестовое [21], исследован в диссертационной работе А.Ю. Городилова [22]. Однако в полной мере особенности и возможности новых элементов относительно их диагностики, включая [23–26], исследованы не были.

2. Новые логические элементы ПЛИС. Один из предложенных ранее логических элементов [23] использует ДНФ представление логической функции (реализуется система логических функций в ДНФ), который может конфигурироваться так же, как LUT. Показано, что использовать их целесообразно при большом количестве переменных n . Обозначим множество таких элементов M_{DNF} . Следующий новый элемент основан на реализации дешифратора [24], что позволяет реализовывать систему m логических функций в СДНФ. Обозначим множество таких элементов DC-LUT M_{DC-LUT} . Предложен также LUT, реализующий сразу две функции от одних и тех же аргументов [25], множество таких элементов $M_{duble-LUT}$. Анализ всех этих элементов представлен в [26]. С целью повышения контролепригодности

разработан LUT с самоконтролем [20], назовём такие элементы M_{st-LUT} , а также LUT с одновременной реверсной проверкой всех ветвей дерева транзисторов [21], назовём такие элементы $M_{rev-LUT}$. Введение самоконтроля [20] возможно и для остальных LUT-подобных элементов: $M_{st-DC-LUT}$; $M_{st-duble-LUT}$. Рассмотрим реверсное тестирование [21] (рис. 3).

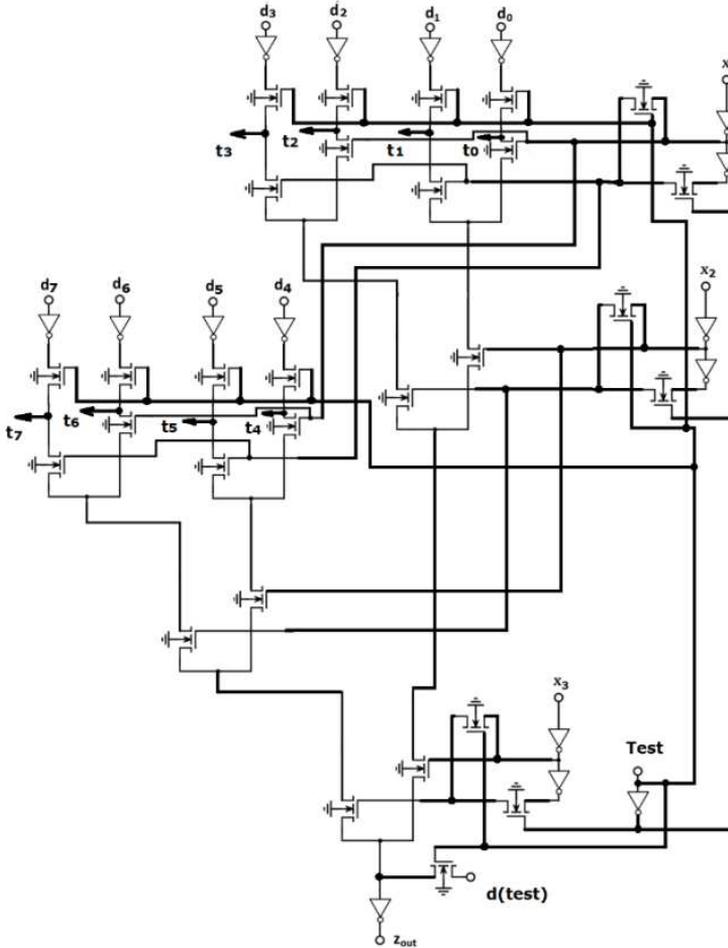
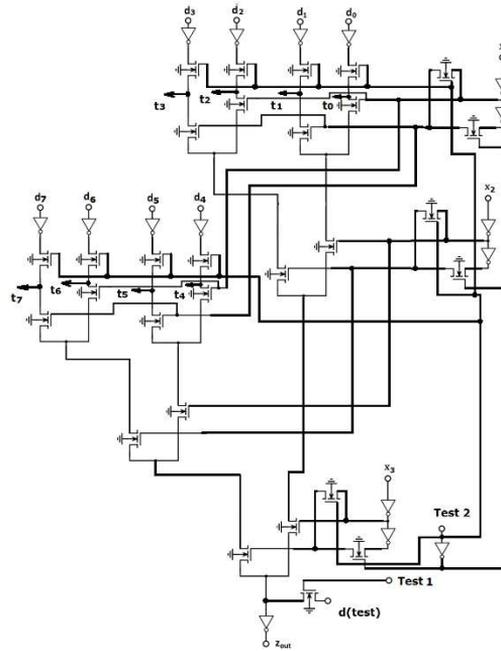
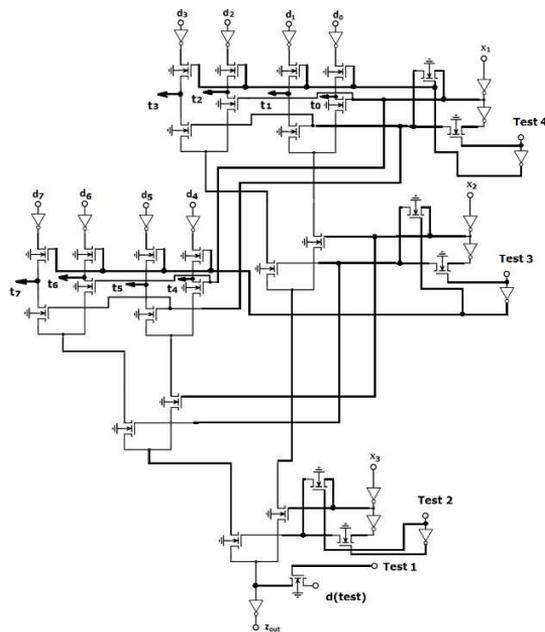


Рис. 3. Реверсное диагностирование (Test=1)

С целью усовершенствования реверсного тестирования предлагается введение средств выбора ветви (ранее все ветви активировались одновременно) (рис. 4, а) и с управлением отдельными адресными линиями (рис. 4, б). Обозначим такие модификации: $M_{rev1-LUT}$; $M_{rev2-LUT}$.



a



б

Рис. 4. Обратное диагностирование: *a* – ($\text{Test 1}=1, \text{Test 1}=1$) с возможностью выбора ветви дерева ($\text{Test 1}=1, \text{Test 1}=0$); *б* – с управлением отдельными адресными сигналами (Test 2,3,4)

3. Постановка задачи. Для диагностирования логики ПЛИС, включающих новые элементы, используем встроенные средства, обеспечивающие генетические алгоритмы GA определения технического состояния логики. В связи с этим возникает задача, с одной стороны, допустимого (DOP) снижения точности Φ_{GA} и/или быстродействия t_{GA} , увеличения объёма оборудования ΔL при максимизации достоверности диагностирования $D(\lambda + \Delta\lambda, t)$ (t – общее время работы оборудования) с учетом увеличения $\Delta\lambda$ интенсивности отказов (сбоев) за счет дополнительного вводимого оборудования.

Таким образом, дано множество логических элементов M :

$$M = M_{DNF} \cup M_{rev1-LUT} \cup M_{rev2-LUT} \cup M_{DC-LUT} \cup M_{duble-LUT} \cup M_{st-LUT} \cup \\ \cup M_{st-DC-LUT} \cup M_{st-duble-LUT} \cup M_{LUT}. \quad (5)$$

Требуется для встроенной аппаратуры разработать генетический алгоритм диагностирования M на основе заданной модели отказов

$$GA(M, F_{0,1}, F_{T0,T1}, F_{brige}), \quad (6)$$

так, чтобы $D_M(\lambda + \Delta\lambda, t) \rightarrow \max$; $\Phi_{GA} \leq \Phi_{GA-DOP}$; $t_{GA} \leq t_{GA-DOP}$; $\Delta L \leq \Delta L_{DOP}$.

Выводы. Таким образом, существующий научно-методический аппарат диагностирования логики ПЛИС основан на тестировании традиционных элементов без учета возможностей рабочего самотестирования и реверсного тестирования сразу всех ветвей дерева передающих транзисторов, что позволяет ускорить поиск неисправностей. Кроме того, недостаточно проработаны вопросы диагностики элементов на основе ДНФ представлений систем логической функций и элементов с реализацией систем функций на основе СДНФ. Поставлена задач совершенствования генетических алгоритмов встроенного тестирования логики, включающей новые логические элементы с самоконтролем и реверсным тестированием.

Библиографический список

1. ГОСТ 27.002–2015. Надежность в технике. Основные понятия. Термины и определения. (Введ. 2017–03–01). – М.: Стандартиформ, 2016. – 23 с.
2. ГОСТ 20911-89. Техническая диагностика. Термины и определения. – М.: Стандартиформ, 2009. – 11 с.

3. A new FPGA architecture and leading-edge FinFET process technology promise to meet next-generation system requirements [Электронный ресурс]. – URL: <https://www.intel.com/content/dam/www/programmable/us/en/pdfs/literature/wp/wp-01220-hyperflex-architecture-fpga-socs.pdf> (дата обращения: 04.05.2019).

4. A radiation hardened reconfigurable FPGA [Электронный ресурс] / Shankarnarayanan Ramaswamy, Leonard Rockett, Dinu Patel [et al.]. – URL: <https://pdfs.semanticscholar.org/57f8/ff540360eadceafc062797b7a01065f6f9cc.pdf> (дата обращения: 05.05.2019).

5. Carmichael C. Triple module redundancy design techniques for virtex FPGAs [Электронный ресурс]. – URL: https://www.xilinx.com/support/documentation/application_notes/xapp197.pdf (дата обращения: 03.05.2019).

6. Intel® Agilex™ I-Series SoC FPGA Product Table [Электронный ресурс]. – URL: <https://www.intel.ru/content/dam/www/programmable/us/en/pdfs/literature/pt/intel-agilex-i-series-product-table.pdf> (дата обращения: 03.05.2019).

7. Intel® Stratix®10 GX/SX Product Table [Электронный ресурс]. – URL: <https://www.intel.ru/content/dam/www/programmable/us/en/pdfs/literature/pt/stratix-10-product-table.pdf> (дата обращения: 04.05.2019).

8. Overview of the IEEE P1500 Standard / F. DaSilva, Y. Zorian, L. Whetsel, K. Arabi, R. Kapur [Электронный ресурс]. – URL: <http://mesl.ucsd.edu/gupta/cse291-fpga/Readings/P1500.pdf> (дата обращения: 04.05.2019).

9. Хаханов В.И. Инфраструктура диагностического обслуживания SoC [Электронный ресурс] // Вестник Томск. ун-та. – 2008. – №4(5). – URL: <http://sun.tsu.ru/mminfo/000063105/inf/05/image/05-074.pdf> (дата обращения: 05.05.2019).

10. Парфентий А.Н., Хаханов В.И., Литвинова Е.И. Модели инфраструктуры сервисного обслуживания цифровых систем на кристаллах // АСУ и приборы автоматики. – 2007. – Вып. 138. – С. 83–99.

11. Строгонов А.В., Городков П.С. Особенности проектирования устройств цифровой обработки сигналов в базисе ПЛИС серии 5578 // Известия вузов. Электроника. – 2017. – Т. 22. – № 3. – С. 256–265. DOI: 10.24151/1561-5405-2017-22-3-256-265

12. Надежность и эффективность в технике: справочник: в 10 т. / ред. совет: В.С. Авдеевский (председат.) [и др.]. – М.: Машиностроение,

1986–1990. – Т. 9: Техническая диагностика. – М.: Машиностроение, 1987. – 352 с.

13. IEEE Std 1149.1 (JTAG) – Texas Instruments [Электронный ресурс]. – URL: <http://www.ti.com/lit/an/ssya002c/ssya002c.pdf> (дата обращения: 04.05.2019).

14. Fault models [Электронный ресурс]. – URL: <http://www.eng.auburn.edu/~strouce/class/elec6970/BISTc2.pdf> (дата обращения: 04.05.2019).

15. Tyurin S.F. LUT's sliding backup // IEEE Transactions on Device and Materials Reliability. – Vol. 19. – Iss. 1. – P. 221–225. Published: MAR 2019. DOI: 10.1109/TDMR.2019.2898724

16. Weibull W. A statistical distribution function of wide applicability [Электронный ресурс]. – URL: <https://pdfs.semanticscholar.org/88c3/7770028e7ed61180a34d6a837a9a4db3b264.pdf> (дата обращения: 18.04.2019).

17. Jahanirad H., Karam H. BIST-based test-ing and diagnosis of LUTs in SRAM-based FPGAs // Emerging Science Journal. – 2017. – Vol. 1, № 4. – P. 217–219. DOI: 10.28991/ijse-01125

18. Сперанский Д.В., Скобцов Ю.А., Скобцов В.Ю. Моделирование, тестирование и диагностика цифровых устройств. – М.: ИНТУИТ, 2016. – 439 с.

19. Иванов Д.Е. Исследование масштабируемости параллельных генетических алгоритмов построения идентифицирующих последовательностей цифровых устройств // Новые информационные технологии в исследовании сложных структур: материалы X Рос. конф. с междунар. участ. / Томск. гос. ун-т; Горно-Алтайск. гос. ун-т; Ин-т оптики атмосферы им. В.Е. Зуева СО РАН. – Томск, 2014. – С. 4–5.

20. Программируемое логическое устройство: пат. Рос. Федерация №2547229 / С.Ф. Тюрин, А.Ю. Городилов, Р.В. Вихорев; опубл. 10.04.2015. Бюл. №10.

21. Программируемое логическое устройство: пат. Рос. Федерация № 2573758 / С.Ф. Тюрин, А.Ю. Городилов, Е.Ю. Данилова; опубл. 27.01.2016. Бюл. №3.

22. Городилов А.Ю. Методы и алгоритмы диагностирования и реконфигурации логики высоконадёжных ПЛИС: дис. ... канд. техн. наук: 05.13.05 / Южно-Рос. гос. политехн. ун-т (НПИ) им. М.И. Платова. – Новочеркасск, 2016. – 201 с.

23. Программируемое логическое устройство: пат. Рос. Федерация №2544750 / С.Ф. Тюрин; опубл. 20.03.2015. Бюл. №8.

24. Программируемое логическое устройство: пат. Рос. Федерация №2573732 / С.Ф. Тюрин, Р.В. Вихорев; опубл. 27.01.2016. Бюл. №3.

25. Программируемое логическое устройство: пат. Рос. Федерация №2637462 / С.Ф. Тюрин, А.С. Прохоров; опубл. 04.12.2017. Бюл. №34.

26. Tyurin S. Green logic: green LUT FPGA concepts, models and evaluations // Green IT Engineering: Concepts, Models, Complex Systems Architectures, Studies in Systems, Decision and Control, V. Kharchenko, Y. Kondratenko, J. Kasprzyk (Eds.). – XIV. 355. – P. 147 illus. – Berlin, Heidelberg: Springer International Publishing. – 2017. – P. 241–261. DOI: 10.1007/978-3-319-55595-9_12

References

1. GOST 27.002–2015. Nadezhnost' v tekhnike. Osnovnye poniatia. Terminy i opredeleniia. (Vved. 2017–03–01) [GOST 27.002–2015. Reliability in technology. Basic concepts. Terms and Definitions. (Introduced 2017–03–01)]. Moscow: Standartinform, 2016. 23 p.

2. GOST 20911-89. Tekhnicheskaja diagnostika. Terminy i opredeleniia [GOST 20911-89. Technical diagnostics. Terms and Definitions]. Moscow: Standartinform, 2009. 11 p.

3. A New FPGA architecture and leading-edge FinFET process technology promise to meet next-generation system requirements, available at: <https://www.intel.com/content/dam/www/programmable/us/en/pdfs/literature/wp/wp-01220-hyperflex-architecture-fpga-socs.pdf> (accessed 04 May 2019).

4. Ramaswamy Sh., Rockett L., Patel D. et al. A radiation hardened reconfigurable FPGA, available at: <https://pdfs.semanticscholar.org/57f8/ff540360eadceafc062797b7a01065f6f9cc.pdf> (accessed 05 May 2019).

5. Carmichael C. Triple module redundancy design techniques for virtex FPGAs, available at: https://www.xilinx.com/support/documentation/application_notes/xapp197.pdf (accessed 03 May 2019).

6. Intel® Agilex™ I-Series SoC FPGA Product Table, available at: <https://www.intel.ru/content/dam/www/programmable/us/en/pdfs/literature/pt/intel-agilex-i-series-product-table.pdf> (accessed 03 May 2019).

7. Intel® Stratix®10 GX/SX Product Table, available at: <https://www.intel.ru/content/dam/www/programmable/us/en/pdfs/literature/pt/stratix-10-product-table.pdf> (accessed 04 May 2019).

8. DaSilva F., Zorian Y., Whetsel L., Arabi K., Kapur R. Overview of the IEEE P1500 Standard, available at: <http://mesl.ucsd.edu/gupta/cse291-fpga/Readings/P1500.pdf> (accessed 04 May 2019).

9. Khakhanov V.I. Infrastruktura diagnosticheskogo obsluzhivaniia SoC [SoC Diagnostic Maintenance Infrastructure]. *Vestnik Tomskogo universiteta*, 2008, no. 4(5), available at: <http://sun.tsu.ru/mminfo/000063105/inf/05/image/05-074.pdf>. (accessed 05 May 2019).

10. Parfentii A.N., Khakhanov V.I., Litvinova E.I. Modeli infrastruktury servisnogo obsluzhivaniia tsifrovyykh sistem na kristallakh [Models of the service infrastructure of digital systems on crystals]. *Avtomatizirovannaiia sistema upravleniia i pribory avtomatiki*, 2007, iss. 138, pp. 83–99.

11. Strogonov A.V., Gorodkov P.S. Osobennosti proektirovaniia ustroystv tsifrovoi obrabotki signalov v bazise PLIS serii 5578 [Design Features of Digital Signal Processing Devices in 5578 Series PLD Basis]. *Izvestiia vuzov. Elektronika*, 2017, vol. 22, no. 3, pp. 256-265. DOI: 10.24151/1561-5405-2017-22-3-256-265

12. Nadezhnost' i effektivnost' v tekhnike: spravochnik [Reliability and efficiency in engineering: reference]. Pedaktsionnyi sovet: V.S. Avduevskii (predsedatel') et al. Moscow: Mashinostroenie, 1986-1990. vol. 9: Tekhnicheskaiia diagnostika. Moscow: Mashinostroenie, 1987. 352 p.

13. IEEE Std 1149.1 (JTAG) - Texas Instruments, available at: <http://www.ti.com/lit/an/ssya002c/ssya002c.pdf> (accessed 04 May 2019).

14. Fault models, available at: <http://www.eng.auburn.edu/~strouce/class/elec6970/BISTc2.pdf> (accessed 04 May 2019).

15. Tyurin S.F. LUT's Sliding Backup. *IEEE Transactions on Device and Materials Reliability*, vol. 19, iss. 1, pp. 221-225. Published: MAR 2019. DOI: 10.1109/TDMR.2019.2898724

16. Weibull W. A statistical distribution function of wide applicability, available at: <https://pdfs.semanticscholar.org/88c3/7770028e7ed61180a34d6a837a9a4db3b264.pdf> (accessed 18 April 2019).

17. Jahanirad H., Karam H. BIST-based test-ing and diagnosis of LUTs in SRAM-based FPGAs. *Emerging Science Journal*, 2017, vol. 1, no. 4, pp. 217-219. DOI: 10.28991/ijse-01125

18. Cперanskii D.V., Skobtsov Iu.A., Skobtsov V.Iu. Modelirovanie, testirovanie i diagnostika tsifrovyykh ustroystv [Digital Device Modeling, Testing, and Diagnostics]. Moscow: INTUIT, 2016. 439 p.

19. Ivanov D.E. Issledovanie masshtabiruemosti parallel'nykh geneticheskikh algoritmov postroeniia identifikatsionnykh posledovatel'nostei tsifrovyykh ustroystv [Study of scalability of parallel genetic algorithms of construction of identifying sequences of digital devices]. *Novyye informatsionnyye tekhnologii v issledovanii slozhnykh struktur. Materialy X Rossiiskoi konferentsii s mezhdunarodnym uchastiem. Tomskii gosudarstvennyi universitet; Gorno-Altayskii gosudarstvennyi universitet; Institut optiki atmosfery imeni V.E. Zueva SO RAN*. Tomsk, 2014, pp. 4-5.

20. Tiurin S.F., Gorodilov A.Iu., Vikhorev R.V. Programmiruemo logicheskoe ustroystvo [Programmable logic device]. Patent Rossiiskaia Federatsiia no. 2547229 (2015).

21. Tiurin S.F., Gorodilov A.Iu., Danilova E.Iu. Programmiruemo logicheskoe ustroystvo [Programmable logic device]. Patent Rossiiskaia Federatsiia no. 2573758 (2016).

22. Gorodilov A.Iu. Metody i algoritmy diagnostirovaniia i rekonfiguratsii logiki vysokonadezhnykh PLIS [Methods and algorithms of diagnostics and reconfiguration of logic of highly reliable PLIC]. Ph.D. thesis. Novocherkassk: Iuzhno-Rossiiskii gosudarstvennyi politekhnicheskii universitet (NPI) imeni M. I. Platova, 2016. 201 p.

23. Tiurin S.F. Programmiruemo logicheskoe ustroystvo [Programmable logic device]. Patent Rossiiskaia Federatsiia no. 2544750 (2015).

24. Tiurin S.F., Vikhorev R.V. Programmiruemo logicheskoe ustroystvo [Programmable logic device]. Patent Rossiiskaia Federatsiia no. 2573732 (2016).

25. Tiurin S.F., Prokhorov A.S. Programmiruemo logicheskoe ustroystvo [Programmable logic device]. Patent Rossiiskaia Federatsiia no. 2637462 (2017).

26. Tyurin S. Green Logic: Green LUT FPGA Concepts, Models and Evaluations. *Green IT Engineering: Concepts, Models, Complex Systems Architectures, Studies in Systems, Decision and Control*. Eds. V. Kharchenko, Y. Kondratenko, J. Kacprzyk. XIV. 355. pp. 147 illus. Berlin, Heidelberg: Springer International Publishing, 2017, pp. 241-261. DOI: 10.1007/978-3-319-55595-9_12

Сведения об авторе

Данилова Екатерина Юрьевна (Пермь, Россия) – старший преподаватель кафедры «Математическое обеспечение вычислительных систем» Пермского государственного национального исследовательского университета (614990, Пермь, ул. Генкеля, 7, корп. 2, e-mail: keteref@gmail.com).

About the author

Danilova Ekaterina Yurevna (Perm, Russian Federation) is a Senior Lecturer at the Department of Software Computing Systems Perm State University (614990, Perm, 7, Genkel str., Bldg. 2, e-mail: keteref@gmail.com).

Получено 17.07.2019