

УДК 681.32

DOI: 10.15593/2224-9397/2021.1.04

**А.В. Греков**

Пермский военный институт войск национальной гвардии, Пермь, Россия

## МАСШТАБИРОВАНИЕ НАДЁЖНОСТИ ПЛИС

Программируемые логические интегральные схемы (ПЛИС) в настоящее время находятся среди основных аппаратных базисов реализации цифровых устройств и систем. По мере их развития с 80-х гг. XX в., когда они содержали только сотни логических элементов, до наших дней, когда «топовые» ПЛИС содержат сотни тысяч и даже миллионы логических элементов, в них постепенно вводились средства масштабирования функциональных возможностей, быстродействия, энергопотребления. Примером могут быть технологии HyperFlex, Tri-Gate transistors, Voltage Reduction Technology (VRT), Dynamic voltage and frequency scaling (DVFS), Sleep modes, Power management. Алекс Яковлев предложил концепцию энергетически модулированного компьютера. С.Ф. Тюрин предложил концепцию масштабирования логических базисов. Внедряются стандарты встроенного диагностирования. Теперь ПЛИС выходят еще на более высокий уровень: они реализуют аппаратные ускорители для задач, традиционно решаемых программно. Несмотря на прогресс научно-математического аппарата масштабирования в области энергосбережения, производительности, косвенно влияющих на показатели надёжности, и научно-математического аппарата обеспечения надёжности, до сих пор не произошло их объединения для разработки технологии масштабирования пользователем показателей надёжности проекта на ПЛИС, т.е. масштабирования надёжности в полной мере пока достичь не удалось. **Цель исследования:** разработка концепции и теоретических основ масштабирования надёжности ПЛИС. **Методы:** анализ уровней обеспечения надёжности ПЛИС, разработка концепции масштабирования надёжности ПЛИС, синтез масштабируемой по надёжности архитектуры логики ПЛИС. **Результаты:** концепция масштабирования надёжности логики ПЛИС, метод анализа уровней обеспечения надёжности логики ПЛИС, метод синтеза масштабируемой избыточности логики ПЛИС. **Практическая значимость:** разработанные теоретические основы масштабирования надёжности логики ПЛИС могут обеспечить создание нового перспективного класса ПЛИС.

**Ключевые слова:** FPGA, масштабирование, надёжность.

**A.V. Grekov**

Perm Military Institute of National Guard Troops, Perm, Russian Federation

## SCALING OF THE RELIABILITY FPGA

Programmable logic integrated circuits (FPGA) are currently one of the main hardware bases for the implementation of digital devices and systems. As they developed from the 80s of the twentieth century, when they contained only hundreds of logical elements, to the present day, when the "top" FPGAs contain hundreds of thousands and even millions of logical elements, they gradually introduced means of scaling functionality, speed, power consumption. For example HyperFlex, Tri-Gate transistors, Voltage Reduction Technology (VRT), Dynamic voltage and frequency scaling (DVFS), Sleep modes, Power management. Concept of the energy modulated computing proposed by Alex Yakovlev.

S.F. Tyurin is proposed the concept of scaling logical bases. Embedded diagnostic standards are being introduced. Now FPGAs are reaching an even higher level: they implement hardware accelerators for tasks traditionally solved in software. However, reliability scaling has not yet been fully achieved. **Purpose** development of the concept and theoretical foundations for scaling the reliability of FPGAs. **Methods:** analysis of the levels of ensuring the reliability of FPGAs, development of the concept of scaling the reliability of FPGAs, synthesis of a scalable architecture of FPGA logic. **Results:** the concept of scaling the reliability of the FPGA logic, the method of analyzing the levels of ensuring the reliability of the FPGA logic, the method of synthesizing the scalable redundancy of the FPGA logic. **Practical significance:** The developed theoretical foundations for scaling the reliability of the FPGA logic can provide the creation of a new, promising FPGA class.

**Keywords:** FPGA, Scaling, Reliability.

## Введение

В настоящее время использование ПЛИС (FPGA) выходит на совершенно новый уровень, уровень реконфигурируемых вычислений, намеченный в известной книге А.В. Каляева как система с программируемой архитектурой в начале 80-х гг. XX века, когда скромные по возможностям ПЛИС только начинали использоваться [1]. Развитие подобного подхода уже в XXI в. исследуется в [2–5].

Фирма Xilinx разработала программный ускоритель для задач искусственного интеллекта (ИИ): AI Engine – Artificial Intelligence Accelerator [6, 7]. Это так называемые PCIe ускорительные платы Alveo (рис. 1).

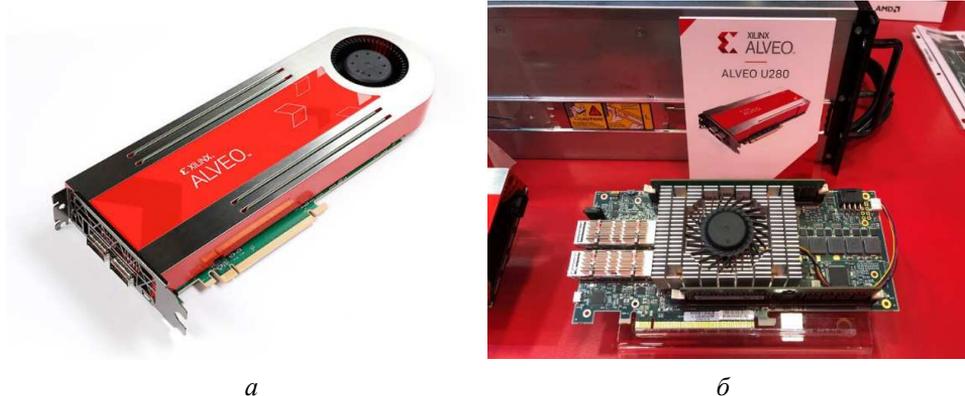


Рис. 1. PCIe ускорительные платы на основе FPGA от Xilinx:  
а – Alveo U200; б – Alveo U280

При этом FPGA рассматривается как функция на C-подобном языке, хост-компьютер вызывает эту функцию для ядра реализуемого проекта. Время реконфигурации и вычислений меньше времени реализации соответствующего алгоритма в процессоре общего назначения. При этом время отклика на сегодня составляет несколько десятков миллисекунд (Alveo U280). Производительность машинного обучения

возрастает в несколько десятков раз. Загрузка конфигурации при необходимости инициализации памяти происходит относительно медленно и измеряется секундами (используется только один канал с пропускной способностью 8 Гб в секунду), если же память не инициализируется, то процесс занимает доли секунды. Варьирование количества и «мощности» подобных ускорительных устройств (фактически сопроцессоров) обеспечивает масштабирование вычислительных возможностей и стоимости системы.

Широко используется, в том числе в FPGA, масштабирование энергопотребления: Voltage Reduction Technology (VRT), Dynamic voltage and frequency scaling (DVFS), Sleep modes, Power management [8, 9]. Концепция энергомодулированных вычислений (energy modulated computing) предложена Алексом Яковлевым из университета Нью Касла [10]. Он являлся членом известной группы В.И. Варшавского, исследовавшей аperiodические (самосинхронные) автоматы. Разработаны так называемые Tri-gate transistors, которые знаменуют собой новый этап взаимно-масштабирования быстродействия и энергопотребления [11]. Новые возможности масштабируемой логической функциональности ПЛИС включают так называемый HyperFlex [12] и новые типы логики [13–15].

Однако широкого распространения подобного подхода масштабирования надёжности [16], доступного для пользователей, пока не наблюдается. Хотя, учитывая расширение области применения ПЛИС, в ряде задач, например, связанных с работой в условиях радиации [17–20], это чрезвычайно необходимо. Тем более, что для этого уже созрели все предпосылки. Это относится прежде всего к встроенному диагностированию [21] ПЛИС и систем на кристаллах, зафиксированному в стандарте IEEE P1500 [22]. В некоторых ПЛИС уже используется встроенное структурное резервирование: мажоритарное резервирование (Triple Module Redundancy) [23]. На новый уровень надёжности выводят так называемые функционально-полные толерантные (ФПТ) элементы [24–25], частным случаем которых, как оказалось, являются элементы с транзисторным резервированием ТР [26–28]. Предложена концепция обеспечения энергонадёжности схем [29], скользящего резервирования путем использования остаточной функциональности LUT FPGA [30].

Таким образом, намеченные на рубеже XX и XXI веков новые подходы к обеспечению высокой надёжности ПЛИС, которые в то

время были еще не обеспечены технологически, теперь становятся практически реализуемыми кроме прочего в связи с резким уменьшением времени реконфигурации ПЛИС.

### **Формулировка научной проблемы**

Анализ предметной области позволяет выделить противоречие в практике: имеются примеры успешного использования масштабирования в области ПЛИС, однако масштабирование надёжности до сих пор в полной мере не применяется, хотя потребности в такой технологии очень существенны. Противоречием в науке является то, что, несмотря на прогресс научно-математического аппарата масштабирования в области энергосбережения, производительности, косвенно влияющих на показатели надёжности, и научно-математического аппарата обеспечения надёжности, до сих пор не произошло их объединения для разработки технологии масштабирования пользователем показателей надёжности проекта на ПЛИС. Предлагается расширить технологию HyperFlex путём разработки технологии HyperReliability (HR). Диапазон масштабирования в зависимости от стоящих задач: от 1) нерезервированной системы с адаптацией к отказам путем реконфигурирования (допускаются значительные перерывы на восстановление) и использования новых, еще не использованных элементов или использования остаточной функциональности, в том числе ФПТ [24, 25], через 2) многоканальную систему разного уровня избыточности с возможной кластеризацией и отдельными источниками питания (онлайн-задачи), в том числе скользящее резервирование с возможностью восстановления части отказавших элементов) к 3) резервированию на транзисторном уровне (ТР) также с возможной кластеризацией и отдельными источниками питания. Самый большой эффект в повышении вероятности безотказной работы позволяет достичь транзисторное резервирование (ТР), однако оно и самое затратное: для этого нужно проектировать совершенно новую ПЛИС. Кроме того, ТР не всегда возможно без декомпозиции логического элемента. Тем не менее, исследования показывают, что такой подход выигрывает по энергопотреблению, а в ряде случаев и по сложности. Сложность в том, что конфигурирование схемы из отдельных транзисторов на современном этапе весьма проблематично. Наиболее важной, хотя и по площади кристалла значительно уступающей памяти, составляющей частью ПЛИС являются логика, логические элементы, без которых невозмож-

на эффективная реализация проектов. Их надёжность в наибольшей степени определяет надёжность ПЛИС. В связи с этим возникает проблема разработки теоретических основ масштабирования надёжности логики ПЛИС.

### Модели надёжности логики ПЛИС

Используем экспоненциальные модели основных вариантов резервирования для распределения Вейбулла [31]. Дублирование одно-выходного канала с учетом вероятности безотказной (бесбойной) работы схемы сравнения (сложения по модулю два, исключаящее ИЛИ)  $e^{-\lambda_{\oplus} t^{\alpha}}$  оценивается выражением достоверности функционирования:

$$P_2^* = \left( e^{-2\lambda t^{\alpha}} + 2e^{-\lambda t^{\alpha}} (1 - e^{-\lambda t^{\alpha}}) \right) e^{-\lambda_{\oplus} t^{\alpha}}, \quad (1)$$

где  $\lambda$  – интенсивность отказов (сбоев) одного канала.

В случае дублирования отдельных  $i$ -х устройств из  $q$  устройств с  $m_i$  выходами и интенсивностью отказов (сбоев)  $\lambda_i$  («глубокое» дублирование) получим достоверность:

$$P_2^{**} = \prod_{i=1}^q \left( e^{-2\lambda_i t^{\alpha}} + 2e^{-\lambda_i t^{\alpha}} (1 - e^{-\lambda_i t^{\alpha}}) \right) \left( e^{-\lambda_i t^{\alpha}} \right)^{m_i}, \quad (2)$$

С учетом мажоритирования мажоритаров и интенсивности отказов (сбоев) мажоритаров  $\lambda$  получим для одного выхода канала резервирования по принципу  $\geq n$ ,  $n$  – число работоспособных каналов из общего количества  $2n-1$ ,  $n \geq 2$ :

$$P_{\geq n}^* = \left( e^{-\lambda t^{\alpha}(2n-1)} + \sum_{i=1}^{\lfloor \frac{2n-1}{2} \rfloor} \{ C_{2n-1}^i [e^{-\lambda t^{\alpha}(2n-1-i)} (1 - e^{-\lambda t^{\alpha}})^i] \} \right) \times \left( e^{-\lambda_M t^{\alpha}(2n-1)} + \sum_{i=1}^{\lfloor \frac{2n-1}{2} \rfloor} \{ C_{2n-1}^i [e^{-\lambda_M t^{\alpha}(2n-1-i)} (1 - e^{-\lambda_M t^{\alpha}})^i] \} \right), \quad (3)$$

где  $\lfloor \cdot \rfloor$  – округление в меньшую сторону,  $\lambda$  – интенсивность отказов (сбоев) одного канала  $C_{2n-1}^i$  – число сочетаний из  $2n-1$  по  $i$ .

Избыточность (в каналах) равна  $2n-1$ . Глубокое мажоритирование при выделении условных  $k$  слоёв в одновыходных каналах с интенсивностью  $\lambda$  описывается формулой:

$$P_{\geq n.зМ}^* = \left( e^{\frac{\lambda}{k} t^\alpha (2n-1)} + \sum_{i=1}^{\lfloor \frac{2n-1}{2} \rfloor} \{ C_{2n-1}^i [e^{-\frac{\lambda}{k} t^\alpha (2n-1-i)} (1 - e^{-\frac{\lambda}{k} t^\alpha})^i] \} \right)^k \times \left( e^{-\lambda_M t^\alpha (2n-1)} + \sum_{i=1}^{\lfloor \frac{2n-1}{2} \rfloor} \{ C_{2n-1}^i [e^{-\lambda_M t^\alpha (2n-1-i)} (1 - e^{-\lambda_M t^\alpha})^i] \} \right)^k. \quad (4)$$

В случае мажоритирования отдельных  $i$ -х устройств из  $q$  устройств с  $m_i$  выходами и интенсивностью отказов (сбоев)  $\lambda_i$  (глубокое мажоритирование) получим:

$$P_{\geq n.гМ}^{**} = \prod_{i=1}^q \left( e^{-\lambda_i t^\alpha (2q-1)} + \sum_{i=1}^{\lfloor \frac{2q-1}{2} \rfloor} \{ C_{2q-1}^i [e^{-\lambda_i t^\alpha (2q-1-i)} (1 - e^{-\lambda_i t^\alpha})^i] \} \right) \times \left( e^{-\lambda_M t^\alpha (2q-1)} + \sum_{i=1}^{\lfloor \frac{2q-1}{2} \rfloor} \{ C_{2q-1}^i [e^{-\lambda_M t^\alpha (2q-1-i)} (1 - e^{-\lambda_M t^\alpha})^i] \} \right)^{m_i}. \quad (5)$$

Транзисторное резервирование (TrR) описывается выражением:

$$P_{(r+1)^2} (t)_{TrR} = \left( \sum_{i=0}^r C_{(r+1)^2}^i \left\{ e^{-[(r+1)^2 - i] \lambda_1 t^\alpha} (1 - e^{-\lambda_1 t^\alpha})^i \right\} \right)^w, \quad (6)$$

где  $\lambda_1$  – интенсивность отказов (сбоев) одного транзистора,  $\alpha$  – коэффициент распределения Вейбулла,  $1 \leq \alpha \leq 2$ ,  $t$  – время работы,  $r$  – количество парлируемых отказов в группе резервированного транзистора  $(r+1)^2 - 1$  – избыточность,  $w$  – количество транзисторов в устройстве (проекте) до резервирования,  $(r+1)^2 \cdot w$  – всего транзисторов (после резервирования, если не было декомпозиции исходного устройства, увеличивающего это значение).

В случае декомпозиции в соответствии с ограничениями на число последовательно соединенных транзисторов [32, 33] необходимо увеличивать общее число транзисторов, введя коэффициент  $k > 1 : kw$ .

Метод скользящего резервирования  $sb$  с частичным восстановлением работоспособных элементов из нескольких отказавших, например, на основе функционально-полных толерантных элементов

ФПТ-элементов [24, 25] или «половинной» функциональности LUT [30] позволяет получить выражение:

$$P(t)_{sb} = \left\{ \sum_{i=q}^{q+g} C_{q+g}^i \cdot e^{-i \cdot \lambda \cdot t^\alpha} \cdot (1 - e^{-\lambda t^\alpha})^{q+g-i} \cdot e^{-\lambda_{sd} \cdot t^\alpha} + \sum_{j=1}^{\left\lfloor \frac{g}{\delta} \right\rfloor} C_{q+g}^{g+j} \cdot e^{-(q-j)\lambda t^\alpha} (1 - e^{-\lambda t^\alpha})^{g+j} e^{-\lambda_{sd} t^\alpha} \right\} \cdot P_{дв}(t)_{dr}, \quad (7)$$

где  $q$  – число основных элементов,  $g$  – число резервных элементов,  $\delta$  – максимальное число отказавших элементов для восстановления исходной функции,  $\lfloor \cdot \rfloor$  – ближайшее меньшее целое натуральное число (ceil),  $P_{дв}(t)_{dr}$  – вероятность безотказной работы средств диагностики и восстановления.

Поскольку, как уже было указано, конфигурирование отдельных транзисторов вне условий производства в настоящее время весьма проблематично, для масштабирования надёжности пользователем выбираем множество вариантов:

$$\{P_{2}^{**}, P_{\geq n}^*, P_{\geq n.гм}^{**}, P_{sb}\}. \quad (8)$$

Кроме того, возможно комбинирование вариантов (8) в одной ПЛИС, возможно с частичным использованием (6), а возможно и (8) в составе ПЛИС без использования реконфигурации:

$$\{P_{2}^{**}, P_{\geq n}^*, P_{\geq n.гм}^{**}, P_{sb}, P_{(r+1)^2 TrR}\}. \quad (9)$$

Масштабирование  $\Psi$  заключается конфигурировании части  $\delta$  логики ПЛИС, выделяемой под заданный проект, в соответствии с заданными показателями, по вариантам (9):

$$\Psi[\delta_1(P_{2}^{**}), \delta_2(P_{\geq n}^*), \delta_3(P_{\geq n.гм}^{**}), \delta_4(P_{sb}), \delta_5(P_{(r+1)^2 TrR})], \quad (10)$$

причём в общем случае  $\delta_5$  не равна нулю, но не может быть уменьшена или увеличена пользователем.

Следовательно, необходимо определить допустимый уровень масштабирования пользователем и уровень масштабирования на этапе проектирования ПЛИС, чтобы надёжность реализации проектов

$$P_n(\Psi) \rightarrow \max, \quad (11)$$

при аппаратных затратах, не превышающих заданных, учитывая (1)–(7) и то, что введение резерва снижает общее количество логических элементов. Кроме того, могут быть заданы ограничения по потребляемой мощности  $E$ , задержке  $T$ , площади кристалла  $S$ .

### Метод масштабирования надёжности логики ПЛИС

Для введения резервирования необходимо предусмотреть кластеризацию источников питания логики ПЛИС, поскольку использования одного нерезервированного источника может при его неисправности привести к отказу всей резервированной логики. Для масштабирования предлагается активно-пассивная отказосбоеустойчивость конфигурируемых логических блоков (КЛБ) [25]. Ранее предложенный метод обеспечения надёжности мелкозернистых ПЛИС основан на ФПТ-элементах [24, 25], сохраняющих функциональную полноту при заданной модели отказов (рис. 2).

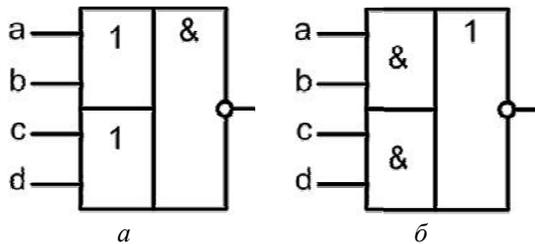


Рис. 2. Функционально-полные толерантные элементы для однократных отказов:

$a$  – ФПТ1 с функцией  $\overline{x_1 x_2} \vee \overline{x_3 x_4}$ ;  $b$  – ФПТ2 с функцией  $\overline{x_1} \overline{x_2} \vee \overline{x_3} \overline{x_4}$

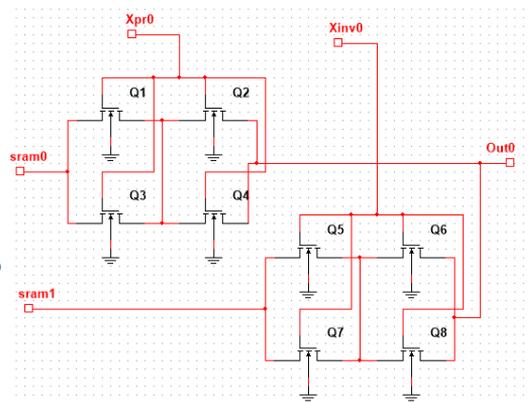
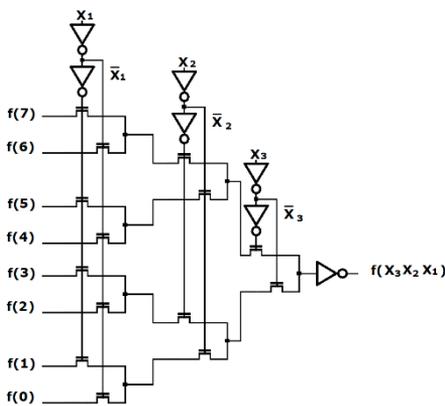
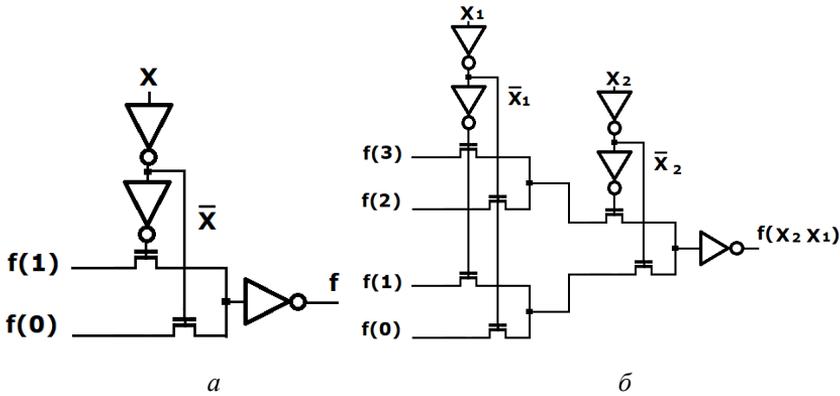
Однако оказалось, что наиболее целесообразно использовать не все остаточные базисы, а только «половинные», при интерпретации ФПТ как мультиплексора на одну переменную, причём реализация ФПТ-элемента на основе передающих транзисторов (pass transistors logic) практически соответствует элементу LUT-1 на одну переменную с инверсированием входов или выхода (рис. 3,  $a$ ). Здесь инверторы по входу переменной необходимы для восстановления уровня сигнала, прошедшего через матрицы коммутаций. Для этого же нужен и выходной инвертор. Инверторы по сигналам настройки  $f(0), f(1)$ , хранящимся в памяти, не указаны. Так, для ФПТ-функции  $\overline{x_1 x_2} \vee \overline{x_3 x_4}$  имеем соответствие с LUT:

$$x_2 = x; x_3 = \bar{x}, x_1 = \bar{f}(1); x_4 = \bar{f}(0). \quad (12)$$

Для функции  $\bar{x}_1 \bar{x}_2 \vee \bar{x}_3 \bar{x}_4$  имеем соответствие с LUT:

$$\bar{x}_2 = x; \bar{x}_3 = \bar{x}, \bar{x}_1 = f(1); \bar{x}_4 = f(0). \quad (13)$$

Интересно, что в такой интерпретации элемент на одну переменную  $f(0)\bar{x} \vee f(1)x$  с настройками конфигурации  $f(0), f(1)$  становится функционально полным, и путем использования только таких элементов можно реализовать любую функцию в СДНФ. А масштабирование логики заключается в каскадировании нескольких LUT-1, при этом получают LUT-2 (рис. 3, б), LUT-3 (рис. 3, в) и т.д. Резервирование транзисторов в LUT-1 (не указаны входные и выходной инверторы) показано на рис. 3, г.



б

г

Рис. 3. Масштабирование LUT:  $a$  – LUT-1;  $b$  – LUT-2;  $c$  – LUT-3;  $d$  – LUT-1 с расчетверенными передающими транзисторами (без инверторов по входам-выходам)

Транзисторы в инверторах, не указанных на рис. 3,  $z$ , также расчетверяются. Аналогично могут быть резервированы LUT-2, LUT-3 и др. При отказах одной половины LUT– $n$  возможно после определения работоспособной половины использовать его, как LUT –  $n-1$ .

Рассмотрим модификацию описания логики ПЛИС с целью масштабирования надёжности. Предлагается новая опция в САПР FPGA (например, Quartus фирмы Intel): задание резервирования для модификации программы на языке HDL (VHDL, Verilog). При этом может быть выбрано дублирование, мажоритирование, скользящее резервирование. Анализируется описание программы, затем формируются резервные каналы, схемы сравнения по модулю два или мажоритары. Для диагностики и восстановления логики при скользящем резервировании формируется специальный контроллер. Пример троирования (VHDL) схемы переноса показан на рис. 4.

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY Let12 IS
PORT (
    reset : IN STD_LOGIC := '0';
    clock : IN STD_LOGIC;
    A1 : IN STD_LOGIC := '0';
    B1 : IN STD_LOGIC := '0';
    C1 : IN STD_LOGIC := '0';
    A2 : IN STD_LOGIC := '0';
    B2 : IN STD_LOGIC := '0';
    C2 : IN STD_LOGIC := '0';
    A3 : IN STD_LOGIC := '0';
    B3 : IN STD_LOGIC := '0';
    C3 : IN STD_LOGIC := '0';
    P1 : OUT STD_LOGIC;
    P2 : OUT STD_LOGIC;
    P3 : OUT STD_LOGIC;
);
END Let12;

ARCHITECTURE BEHAVIOR OF Let12 IS
TYPE type_fstate IS (state1);
SIGNAL fstate : type_fstate;
SIGNAL reg_fstate : type_fstate;

BEGIN
    PROCESS (clock, reg_fstate)
    BEGIN
        IF (clock='1' AND clock'event) THEN
            fstate <= reg_fstate;
        END IF;
    END PROCESS;

    PROCESS (fstate, reset, A1, B1, C1, A2, B2, C2, A3, B3, C3)
    BEGIN
        IF (reset='1') THEN
            reg_fstate <= state1;
            P1 <= '0';
            P2 <= '0';
            P3 <= '0';
        ELSE
            P1 <= '0';
            P2 <= '0';
            P3 <= '0';
            CASE fstate IS
                WHEN state1 =>
                    IF (((((((((((((((((((NOT((A1 = '1')) AND NOT((B1 = '1')) AND NOT((C1 = '1')) OR ((NOT((A1 = '1')) AND NOT((B1 = '1')) AND (C1
                    -- inserting 'else' block to prevent latch inference
                    ELSE
                        reg_fstate <= state1;
                    END IF;
                END CASE;
            END PROCESS;
        
```

Рис. 4. VHDL-файл (фрагмент) с троированием функции P (создан по State Machine File)

Сам файл может быть получен также модификацией соответствующего графа и таблицы переходов (рис. 5).

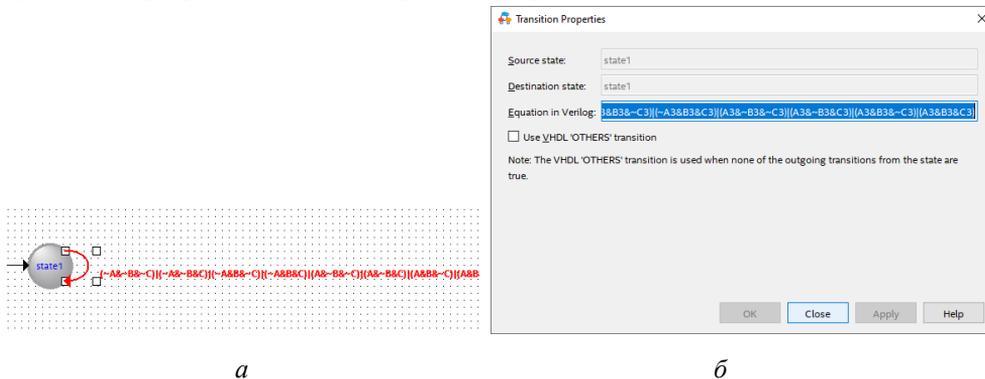


Рис. 5. Модификация State Machine, комбинационный автомат: *a* – граф из одной вершины с троированием функции; *б* – таблица переходов

В результате САПР Quartus строит, например, троированную схему (рис. 6).

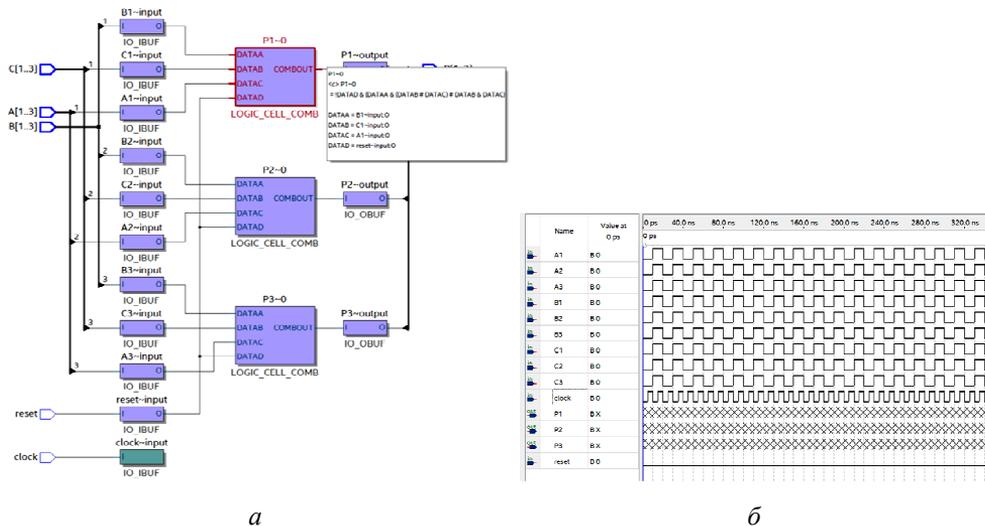


Рис. 6. Результат автоматического синтеза троированной схемы переноса (мажоритарной функции): *a* – схема Map Viewer с троированием LUT (Logic Cell); *б* – Waveform

Предлагается следующая методика масштабирования надёжности логики ПЛИС на основе пассивно-активной отказоустойчивости системы, состоящей из нескольких подсистем. Задаются заданный коэф-

коэффициент готовности  $k_2$ , допустимые аппаратные затраты  $W$ , допустимая временная задержка  $t$ .

Для оценки эффективности необходимо оценивать либо вероятность безотказной работы  $P$  (если проект не допускает перерывов в работе, отказы и сбои маскируются), либо коэффициент готовности  $k_2$  (перерывы в работе допускаются, используется дублирование либо дополнительные встроенные схемы диагностики, либо диагностика осуществляется хост-компьютером).

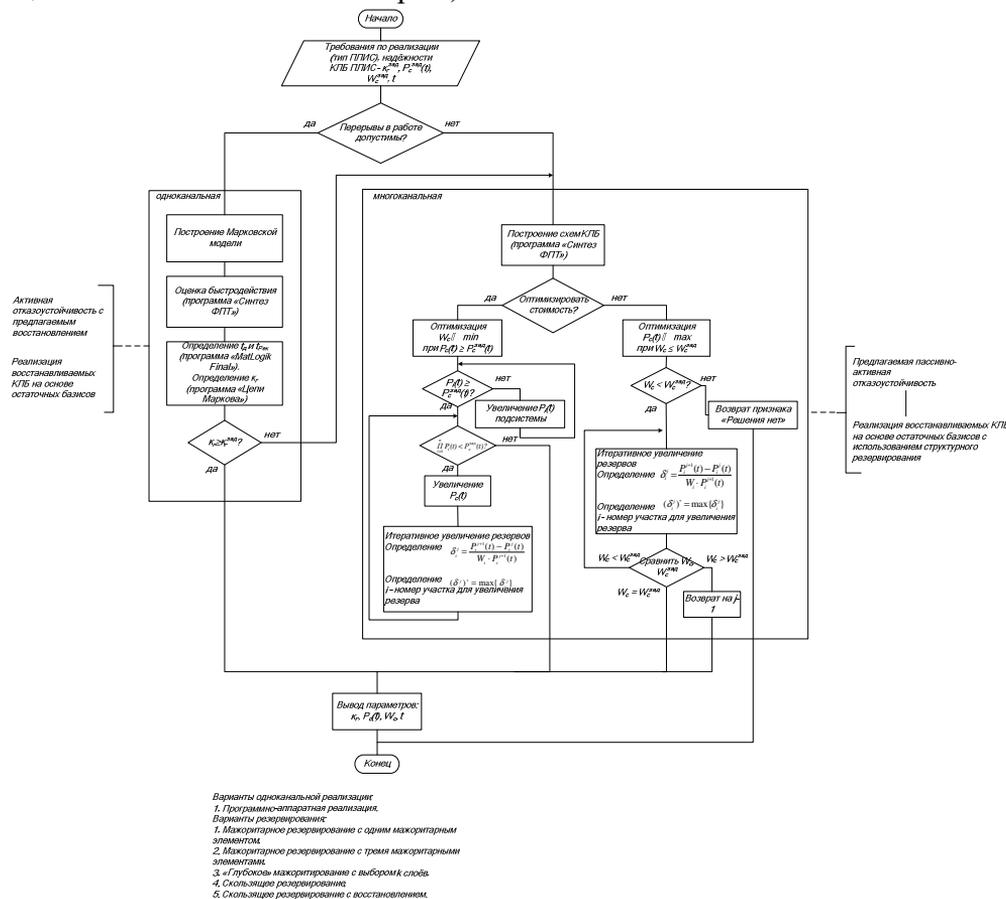


Рис. 7. Методика масштабирования конфигурируемых логических блоков (КЛБ) по заданным требованиям

В последнем случае измеряется время восстановления для настройки соответствующей марковской цепи, используемой для определения коэффициента готовности  $k_2$ .

Методика масштабирования конфигурируемых логических блоков (КЛБ) по заданным требованиям включает итеративное увеличение резервов и модифицированный градиентный метод оптимизации.

Предлагаемая методика масштабирования конфигурируемых логических блоков (КЛБ) по заданным требованиям изображена на рис. 7. Если же при заданных условиях задача не решается, то выводится соответствующее сообщение.

Оценку надёжности проекта в САПР целесообразно оформить опциями, подобными опциям оценки энергопотребления Power Play (Intel), XPower Estimator (Xilinx), а, возможно, и обобщенной опцией Power/Reliability/Delay с дополнительной оценкой быстродействия (временной задержки). Такая технология может рассматриваться как одно из возможных расширений известной технологии HyperFlex от фирмы Intel.

### **Заключение**

В статье представлено предлагаемое масштабирование надёжности логики ПЛИС, осуществляемое пользователем вне условий производства. Однако часть возможностей (например, расчетверение транзисторов некоторых элементов) может быть введена при производстве ПЛИС с целью формирования ПЛИС трех основных классов: «люкс», «бизнес» и «эконом». Предложен метод введения комбинированной структурной и транзисторной избыточности со скользящим резервированием и частичным восстановлением работоспособных элементов из нескольких отказавших за счет использования остаточной функциональности ФПТ-элементов и/или элементов LUT как частного случая ФПТ-элемента для двух переменных.

Для оценки эффективности масштабирования разработана методика, использующая модифицированный градиентный метод оптимизации. Коэффициент готовности определяется по графу соответствующей марковской цепи. Решение проблемы масштабирования надёжности ПЛИС позволяет создать новый класс ПЛИС, которые могут быть применены в областях, требующих сверхвысокой надёжности, в том числе в аппаратуре управления в атомной энергетике, медицине, авионике и космических аппаратах, а также и в военной технике.

### **Библиографический список**

1. Каляев А.В. Многопроцессорные системы с программируемой архитектурой. – М.: Радио и связь, 1984. – 240 с.

2. Филиппов А.К. Теоретические основы проектирования динамически реконфигурируемых систем обработки информации: учеб. пособие [Электронный ресурс]. – Владимир: Изд-во Владимир. гос. ун-та, 2010. – 118 с. – URL: <https://search.rsl.ru/ru/record/01004620116> (дата обращения: 31.01.2021).

3. Kulanov V., Perepelitsyn A., Zarizenko I. Method of development and deployment of reconfigurable FPGA-based projects in cloud infrastructure [Электронный ресурс] // IEEE 9th International Conference on Dependable Systems, Services and Technologies (DESSERT). – 2018. – URL: <https://ieeexplore.ieee.org/document/8409108> (дата обращения: 31.01.2021). DOI: 10.1109/DESSERT.2018.8409108

4. Perepelitsyn A., Zarizenko I., Kulanov V. FPGA as a Service Solutions Development Strategy [Электронный ресурс] // IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT). – 2020. – URL: <https://ieeexplore.ieee.org/document/9125017> (дата обращения: 31.01.2021). DOI: 10.1109/DESSERT50317.2020.9125017

5. Kulanov V., Perepelitsyn A. Scalable FPGA-based Projects via Static and Dynamic Parameterization Technique [Электронный ресурс] // 10th International Conference on Digital Technologies Location: Zilina, Slovakia (JUL 09-11, 2014); IEEE; ESRA; Visegrad Fund. – 2014. – P. 170–173. – URL: [https://apps.webofknowledge.com/full\\_record.do?product=WOS&search\\_mode=SourceByDais&qid=17&SID=F68b5e5iPMfMyGNfNi4&page=1&doc=2](https://apps.webofknowledge.com/full_record.do?product=WOS&search_mode=SourceByDais&qid=17&SID=F68b5e5iPMfMyGNfNi4&page=1&doc=2) (дата обращения: 31.01.2021).

6. Qualcomm Artificial Intelligence (AI) Engine – программный ускоритель ИИ, который будет задействовать различные компоненты SoC [Электронный ресурс]. – URL: <https://www.ixbt.com/news/2018/02/21/qualcomm-artificial-intelligence-ai-engine-soc.html> (дата обращения: 31.01.2021).

7. Ускорители для дата-центров Xilinx Alveo, в десятки раз быстрее CPU [Электронный ресурс]. – URL: [https://info.macrogroup.ru/xilinx\\_alveo?utm\\_source=yandex&utm\\_medium=cpc&utm\\_campaign=Xilinx\\_alveo\\_u200&utm\\_term=alveo%20u200&utm\\_content={creative}&yclid=238819592050085024](https://info.macrogroup.ru/xilinx_alveo?utm_source=yandex&utm_medium=cpc&utm_campaign=Xilinx_alveo_u200&utm_term=alveo%20u200&utm_content={creative}&yclid=238819592050085024) (дата обращения: 31.01.2021).

8. Тюрин С.Ф., Плотникова А.Ю. Концепция «зеленой» логики // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2013. – № 8. – С. 61–72.

9. Mehta Nikil. An ultra-low-energy, variation-tolerant FPGA architecture using component-specific mapping. Dissertation (Ph. D.) [Электронный ресурс] / California Institute of Technology. – URL: <http://thesis.library.caltech.edu/7226/1/Nikil-Mehta-2013.pdf> (дата обращения: 12.01.2021).

10. Yakovlev A. Energy-Modulated Computing [Электронный ресурс]. – URL: <http://async.org.uk/tech-reports/NCL-EECE-MSD-TR-2010-167.pdf> (дата обращения: 31.01.2021).

11. Тюрин С.Ф. Особенности архитектуры гиперфлекс // Вестник Воронеж. гос. ун-та. Сер: Системный анализ и информационные технологии. – 2018. – № 1. – С. 56–62.

12. Тюрин С.Ф., Чудинов М.А. FPGA LUT с двумя выходами декомпозиции по Шеннону // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2019. – № 29. – С. 136–147.

13. Tyurin S.F., Grekov A.V. Study of the multy input LUT complexity // Radio Electronics, Computer Science, Control. – 2018. – № 1. – P. 14–21. DOI: 10.15588/1607-3274-2018-1-2

14. 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. Kacprzyk (Eds.). – Berlin, Heidelberg: Springer International Publishing. – 2017. – XIV. – P. 241–261. DOI: 10.1007/978-3-319-55595-9\_12

15. Тюрин С.Ф., Вихорев Р.В. Адаптивный логический модуль ПЛИС с архитектурой FPGA // Вестник Рязан. гос. радиотехн. ун-та. – 2018. – № 63. – С. 69–76.

16. ГОСТ 27.002–2015. Надёжность в технике Основные понятия. Термины и определения. (Введ. 2017–03–01). – М.: Стандартинформ, 2016. – 23 с.

17. Kuzminova A.V, Kulikov N.A., Popov V.D. Investigation into Radiation Effects in a p-Channel MOS Transistor [Электронный ресурс] // Semiconductors. – August 2020. – Vol. 54, iss. 8. – P. 877–881. – URL: [https://apps.webofknowledge.com/full\\_record.do?product=WOS&search\\_mode=SourceByDais&qid=26&SID=F68b5e5iPMfMyGNfNi4&page=1&doc=1](https://apps.webofknowledge.com/full_record.do?product=WOS&search_mode=SourceByDais&qid=26&SID=F68b5e5iPMfMyGNfNi4&page=1&doc=1) (дата обращения: 31.01.2021). DOI: 10.1134/S1063782620080138

18. Petukhov K.A., Popov V.D. Effect of the active mode NMOS-transistor irradiated on formation of surface defects [Электронный ресурс] // 2nd International Telecommunication Conference on Advanced Micro- and

Nanoelectronic Systems and Technologies Location / Natl. Res. Nucl. Univ., Moscow Engr. Phys. Ins., Moscow, Russia (June 01–02, 2017); Natl Res Nucl Univ, Moscow Engr Phys Inst, Micro-& Nanoelectron Dept. – 2019. – Vol. 498. Article Number 012016. – URL: [https://apps.webofknowledge.com/full\\_record.do?product=WOS&search\\_mode=SourceByDais&qid=26&SID=F68b5e5iPMfMyGNfNi4&page=1&doc=2](https://apps.webofknowledge.com/full_record.do?product=WOS&search_mode=SourceByDais&qid=26&SID=F68b5e5iPMfMyGNfNi4&page=1&doc=2) (дата обращения: 31.01.2021). DOI: 10.1088/1757-899X/498/1/012016

19. Kulikov N.A., Popov V.D. Effect of the Electric Mode gamma and Irradiation on Surface-Defect Formation at the Si-SiO<sub>2</sub> Interface in a MOS Transistor [Электронный ресурс] // Semiconductors. – January 2019. – Vol. 53I, iss. 1. – P. 110–113. – URL: [https://apps.webofknowledge.com/full\\_record.do?product=WOS&search\\_mode=SourceByDais&qid=26&SID=F68b5e5iPMfMyGNfNi4&page=1&doc=3](https://apps.webofknowledge.com/full_record.do?product=WOS&search_mode=SourceByDais&qid=26&SID=F68b5e5iPMfMyGNfNi4&page=1&doc=3) (дата обращения: 31.01.2021). DOI: 10.1134/S1063782619010123

20. Kulikov N.A., Popov V.D., Chubunov P.A. Predicting the No-Failure Microcontroller Operation Probability in a Geostationary Orbit [Электронный ресурс] // Cosmic Research. – September 2018. – Vol. 56, iss. 5. – P. 400–404. – URL: [https://apps.webofknowledge.com/full\\_record.do?product=WOS&search\\_mode=SourceByDais&qid=26&SID=F68b5e5iPMfMyGNfNi4&page=1&doc=4](https://apps.webofknowledge.com/full_record.do?product=WOS&search_mode=SourceByDais&qid=26&SID=F68b5e5iPMfMyGNfNi4&page=1&doc=4) (дата обращения: 31.01.2021). DOI: 10.1134/S0010952518050040.

21. ГОСТ 20911-89. Техническая диагностика. Термины и определения. – М.: Стандартиформ, 2009. – 11 с.

22. Overview of the IEEE P1500 Standard [Электронный ресурс] / Francisco DaSilva, Yervant Zorian, Lee Whetsel, Karim Arabi, Rohit Kapur. – URL: <http://mesl.ucsd.edu/gupta/cse291-fpga/Readings/P1500.pdf> (дата обращения: 04.05.2019).

23. Carmichael C. Triple Module Redundancy Design Techniques for Virtex FPGAs [Электронный ресурс]. – URL: [https://www.xilinx.com/support/documentation/application\\_notes/xapp197.pdf](https://www.xilinx.com/support/documentation/application_notes/xapp197.pdf) (дата обращения: 12.01.2021).

24. Тюрин С.Ф. Проблема сохранения функциональной полноты булевых функций при «отказах» аргументов // Автоматика и телемеханика. – 1999. – № 9. – С. 176–186.

25. Tyurin S.F., Grekov A.V. Functionally Complete Tolerant Elements // International Journal of Applied Engineering Research. – 2015. – Vol. 10, № 14. – P. 34433–34442.

26. Tyurin S.F. Investigation of a Hybrid Redundancy in the Fault-Tolerant Systems // *Radio Electronics, Computer Science, Control*. – 2019. – № 2. – P. 23–33. DOI: 10.15588/1607-3274-2019-2-3

27. El-Maleh A.H., Al-Yamani A., Al-Hashimi B.M. Transistor-Level Defect Tolerant Digital System Design at the Nanoscale. Research Proposal Submitted to Internal Track Research Grant Programs [Электронный ресурс]. – URL: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.474.3844&rep=rep1&type=pdf> (дата обращения: 12.01.2021).

28. Tyurin S. A Quad CMOS gates checking method // *International Journal of Computing*. – 2019. – Vol. 18, iss. 3. – P. 258–264.

29. Kamenskih A.N., Tyurin S.F. The optimization of energy-efficiency and reliability using complex redundancy in computing systems // *Radio Electronics, Computer Science, Control*. – 2018. – № 3. – P. 135–142.

30. Tyurin S.F. LUT's Sliding Backup // *IEEE transactions on device and materials reliability*. – March 2019. – Vol. 19, iss. 1. – P. 221–225. DOI: 10.1109/TDMR.2019.2898724

31. Weibull W. A statistical distribution function of wide applicability [Электронный ресурс]. – URL: <https://pdfs.semanticscholar.org/88c3/7770028e7ed61180a34d6a837a9a4db3b264.pdf> (дата обращения: 12.01.2021).

32. Mead C.A., Conway L. Introduction to VLSI Systems [Электронный ресурс]. – URL: [https://www.researchgate.net/publication/234388249\\_Introduction\\_to\\_VLSI\\_systems](https://www.researchgate.net/publication/234388249_Introduction_to_VLSI_systems) (дата обращения: 12.01.2021).

33. Carver A. Mead, Lynn Conway. Introduction to VLSI Systems [Электронный ресурс]. – URL: <http://ai.eecs.umich.edu/people/conway/VLSI/VLSIText/PP-V2/V2.pdf> (дата обращения: 12.01.2021).

## References

1. Kaliaev A.V. *Mnogoprotsessornye sistemy s programmiruemoi arkhitekturoi* [Multiprocessor systems with programmable architecture]. Moscow: Radio i sviaz', 1984, 240 p.

2. Filippov A.K. *Teoreticheskie osnovy proektirovaniia dinamicheskii rekonfiguriruemyykh sistem obrabotki informatsii* [Theoretical foundations for the design of dynamically reconfigurable information processing systems]. Vladimir: Vladimirskii gosudarstvennyi universitet, 2010, 118 p., available at: <https://search.rsl.ru/ru/record/01004620116> (accessed 31 January 2021).

3. Kulanov V., Perepelitsyn A., Zarizenko I. Method of development and deployment of reconfigurable FPGA-based projects in cloud infrastructure. *IEEE 9th International Conference on Dependable Systems, Services*

and Technologies (DESSERT), 2018, available at: <https://ieeexplore.ieee.org/document/8409108> (accessed 31 January 2021). DOI: 10.1109/DESSERT.2018.8409108

4. Perepelitsyn A., Zarizenko I., Kulanov V. FPGA as a Service Solutions Development Strategy. *IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT)*, 2020, available at: <https://ieeexplore.ieee.org/document/9125017> (accessed 31 January 2021). DOI: 10.1109/DESSERT50317.2020.9125017

5. Kulanov V., Perepelitsyn A. Scalable FPGA-based Projects via Static and Dynamic Parameterization Technique. *10th International Conference on Digital Technologies Location: Zilina, Slovakia (JUL 09-11, 2014)*. IEEE; ESRA; Visegrad Fund, 2014, pp. 170-173, available at: [https://apps.webofknowledge.com/full\\_record.do?product=WOS&search\\_mode=SourceByDais&qid=17&SID=F68b5e5iPMfMyGNfNi4&page=1&doc=2](https://apps.webofknowledge.com/full_record.do?product=WOS&search_mode=SourceByDais&qid=17&SID=F68b5e5iPMfMyGNfNi4&page=1&doc=2) (accessed 31 January 2021).

6. Qualcomm Artificial Intelligence (AI) Engine - programmnyi uskoritel' II, kotoryi budet zadeistvovat' razlichnye komponenty SoC [Qualcomm Artificial Intelligence (AI) Engine is an AI software accelerator that will use various components], available at: <https://www.ixbt.com/news/2018/02/21/qualcomm-artificial-intelligence-ai-engine-soc.html> (accessed 31 January 2021).

7. Uskoriteli dlia data-tsentrov Xilinx Alveo, v desiati raz bystree CPU [Xilinx Alveo data center accelerators, dozens of times faster than CPU], available at: [https://info.macrogroup.ru/xilinx\\_alveo?utm\\_source=yandex&utm\\_medium=cpc&utm\\_campaign=Xilinx\\_alveo\\_u200&utm\\_term=alveo%20u200&utm\\_content={creative}&yclid=238819592050085024](https://info.macrogroup.ru/xilinx_alveo?utm_source=yandex&utm_medium=cpc&utm_campaign=Xilinx_alveo_u200&utm_term=alveo%20u200&utm_content={creative}&yclid=238819592050085024) (accessed 31 January 2021).

8. Tiurin S.F., Plotnikova A.Iu. Kontseptsiiia «zelenoi» logiki [“Green” logic concept]. *Vestnik Permskogo natsional'nogo issledovatel'skogo politekhnicheskogo universiteta. Elektrotehnika, informatsionnye tekhnologii, sistemy upravleniia*, 2013, no. 8, pp. 61-72.

9. Mehta, Nikil. An ultra-low-energy, variation-tolerant FPGA architecture using component-specific mapping. Dissertation (Ph. D.]. California Institute of Technology, available at: <http://thesis.library.caltech.edu/7226/1/Nikil-Mehta-2013.pdf> (accessed 12 January 2021).

10. Yakovlev A. Energy-Modulated Computing, available at: <http://async.org.uk/tech-reports/NCL-EECE-MSD-TR-2010-167.pdf> (accessed 31 January 2021).

11. Tiurin S.F. Osobennosti arkhitektury giperfleks [Features of the hyperflex architecture]. *Vestnik Voronezhskogo gosudarstvennogo universiteta. Sistemyi analiz i informatsionnye tekhnologii*, 2018, no. 1, pp. 56-62.

12. Tiurin S.F., Chudinov M.A. FPGA LUT s dvumia vykhodami dekompozitsii po Shennonu [FPGA LUT with two shannon decomposition outputs]. *Vestnik Permskogo natsional'nogo issledovatel'skogo politekhnicheskogo universiteta. Elektrotehnika, informatsionnye tekhnologii*, 2019, no. 29, pp. 136-147.

13. Tyurin S.F., Grekov A.V. Study of the multy input LUT complexity. *Radio Electronics, Computer Science, Control*, 2018, no. 1, pp. 14-21. DOI: 10.15588/1607-3274-2018-1-2

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

15. Tiurin S.F., Vikhorev R.V. Adaptivnyi logicheskii modul' PLIS s arkhitekturoi FPGA [FPGA adaptive logic module with FPGA architecture]. *Vestnik Riazanskogo gosudarstvennogo radiotekhnicheskogo universiteta*, 2018, no. 63, pp. 69-76.

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

17. Kuzminova A.V, Kulikov N.A., Popov V.D. Investigation into Radiation Effects in a p-Channel MOS Transistor. *Semiconductors*, August 2020, vol. 54, iss. 8, pp. 877-881. DOI: 10.1134/S1063782620080138, available at: [https://apps.webofknowledge.com/full\\_record.do?product=WOS&search\\_mode=SourceByDais&qid=26&SID=F68b5e5iPMfMyGNfNi4&page=1&doc=1](https://apps.webofknowledge.com/full_record.do?product=WOS&search_mode=SourceByDais&qid=26&SID=F68b5e5iPMfMyGNfNi4&page=1&doc=1) (accessed 31 January 2021).

18. Petukhov K.A., Popov V.D. Effect of the active mode NMOS-transistor irradiated on formation of surface defects. *2nd International Telecommunication Conference on Advanced Micro- and Nanoelectronic Systems and Technologies Location*. Natl. Res. Nucl. Univ., Moscow Engn. Phys. Ins., Moscow, Russia (June 01-02, 2017); Natl Res Nucl Univ, Moscow Engn Phys Inst, Micro-& Nanoelectron Dept. DOI: 10.1088/1757-899X/498/1/012016, 2019, vol. 498. Article Number 012016, available at:

[https://apps.webofknowledge.com/full\\_record.do?product=WOS&search\\_mode=SourceByDais&qid=26&SID=F68b5e5iPMfMyGNfNi4&page=1&doc=2](https://apps.webofknowledge.com/full_record.do?product=WOS&search_mode=SourceByDais&qid=26&SID=F68b5e5iPMfMyGNfNi4&page=1&doc=2) (accessed 31 January 2021).

19. Kulikov N.A., Popov V.D. Effect of the Electric Mode gamma and Irradiation on Surface-Defect Formation at the Si-SiO<sub>2</sub> Interface in a MOS Transistor. *Semiconductors*, January 2019, vol. 53I, iss. 1, pp. 110-113 DOI: 10.1134/S1063782619010123, available at: [https://apps.webofknowledge.com/full\\_record.do?product=WOS&search\\_mode=SourceByDais&qid=26&SID=F68b5e5iPMfMyGNfNi4&page=1&doc=3](https://apps.webofknowledge.com/full_record.do?product=WOS&search_mode=SourceByDais&qid=26&SID=F68b5e5iPMfMyGNfNi4&page=1&doc=3) (accessed 31 January 2021).

20. Kulikov N.A., Popov V.D., Chubunov P.A. Predicting the No-Failure Microcontroller Operation Probability in a Geostationary Orbit. *Cosmic Research*, September 2018, vol. 56, iss. 5, pp. 400-404. DOI: 10.1134/S0010952518050040, available at: [https://apps.webofknowledge.com/full\\_record.do?product=WOS&search\\_mode=SourceByDais&qid=26&SID=F68b5e5iPMfMyGNfNi4&page=1&doc=4](https://apps.webofknowledge.com/full_record.do?product=WOS&search_mode=SourceByDais&qid=26&SID=F68b5e5iPMfMyGNfNi4&page=1&doc=4) (accessed 31 January 2021).

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

22. Francisco DaSilva, Yervant Zorian, Lee Whetsel, Karim Arabi, Rohit Kapur. Overview of the IEEE P1500 Standard, available at: <http://mesl.ucsd.edu/gupta/cse291-fpga/Readings/P1500.pdf> (accessed 04 May 2019).

23. Carmichael C. Triple Module Redundancy Design Techniques for Virtex FPGAs, available at: [https://www.xilinx.com/support/documentation/application\\_notes/xapp197.pdf](https://www.xilinx.com/support/documentation/application_notes/xapp197.pdf) (accessed 12 January 2021).

24. Tyurin S.F. Problema sokhraneniia funktsional'noi polnoty bulevykh funktsii pri «otkazakh» argumentov [Retention of functional completeness of Boolean functions under "failures" of the arguments]. *Avtomatika i telemekhanika*, 1999, no. 9, pp. 176-186.

25. Tyurin S.F., Grekov A.V. Functionally Complete Tolerant Elements. *International Journal of Applied Engineering Research*, 2015, vol. 10, no. 14, pp. 34433-34442.

26. Tyurin S.F. Investigation of a Hybrid Redundancy in the Fault-Tolerant Systems. *Radio Electronics, Computer Science, Control*, 2019, no. 2, pp. 23-33. DOI: 10.15588/1607-3274-2019-2-3

27. El-Maleh A.H., Al-Yamani A., Al-Hashimi B.M. Transistor-Level Defect Tolerant Digital System Design at the Nanoscale. Research

Proposal Submitted to Internal Track Research Grant Programs, available at: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.474.3844&rep=rep1&type=pdf> (accessed 12 January 2021).

28. Tyurin S. A Quad CMOS gates checking method. *International Journal of Computing*, 2019, vol. 18, iss. 3, pp. 258-264.

29. Kamenskih A.N., Tyurin S.F. The optimization of energy-efficiency and reliability using complex redundancy in computing systems. *Radio Electronics, Computer Science, Control*, 2018, no. 3, pp. 135-142.

30. Tyurin S.F. LUT's Sliding Backup. *IEEE transactions on device and materials reliability*, March 2019, vol. 19, iss. 1, pp. 221-225. DOI: 10.1109/TDMR.2019.2898724

31. Weibull W. A statistical distribution function of wide applicability, available at: <https://pdfs.semanticscholar.org/88c3/7770028e7ed61180a34d6a837a9a4db3b264.pdf> (accessed 12 January 2021).

32. Mead C.A., Conway L. Introduction to VLSI Systems, available at: [https://www.researchgate.net/publication/234388249\\_Introduction\\_to\\_VLSI\\_systems](https://www.researchgate.net/publication/234388249_Introduction_to_VLSI_systems) (accessed 12 January 2021).

33. Carver A. Mead, Lynn Conway. Introduction to VLSI Systems, available at: <http://ai.eecs.umich.edu/people/conway/VLSI/VLSIText/PP-V2/V2.pdf> (accessed 12 January 2021).

### Сведения об авторе

**Греков Артем Владимирович** (Пермь, Россия) – кандидат технических наук, доцент кафедры «Программное обеспечение вычислительной техники и автоматизированных систем» Пермского военного института войск национальной гвардии Российской Федерации (614112, Пермь, ул. Гремячий Лог, 1, e-mail: grekartemvl@mail.ru).

### About the author

**Artem V. Grekov** (Perm, Russian Federation) – Ph. D. in Technical Sciences, Associate Professor of the Department of Software Computer Technology and Automated Systems Perm Military Institute of National Guard Troops (614112, Perm, str. Gremyachy Log, 1, e-mail: grekartemvl@mail.ru).

Получено 08.02.2021