

УДК 681.32

**С.Ф. Тюрин<sup>1</sup>, А.С. Прохоров<sup>2</sup>**<sup>1</sup>Пермский национальный исследовательский политехнический университет,  
Пермь, Россия<sup>2</sup>АО «ЭР-Телеком Холдинг», Пермь, Россия

## ОТКАЗОУСТОЙЧИВАЯ ПРОГРАММИРУЕМАЯ ЛОГИЧЕСКАЯ МАТРИЦА

Булева алгебра логики, логические функции являются основой математического аппарата цифровой электроники и схемотехники. Вначале, с момента изобретения реле в середине XIX века, они реализовывались релейно-контактными схемами – последовательно-параллельными и мостиковыми, затем после создания транзисторов – на микроселектронном уровне. Были созданы микросхемы малой степени интеграции – «жесткая» логика, затем микросхемы средней степени интеграции, включающие «гибкую» логику. В 70-е годы XX века были выпущены первые программируемые логические матрицы, которые в дальнейшем были использованы в микросхемах большой степени интеграции. Такие структуры широко используются при создании, например, устройств управления и контроля. Программирование сначала было однократным, на заводе-изготовителе, затем появилась возможность программирования пользователем с помощью специального программатора. Потом были разработаны электрически перепрограммируемые логические матрицы. Они используются также в программируемых логических интегральных схемах (ПЛИС) типа CPLD (complex programmable logic devices), в отличие от FPGA (field-programmable gate array), где используется реализация в виде дерева передающих транзисторов, что, по существу, является мультиплексором-селектором или постоянной памятью разрядностью один бит с  $n$  адресными входами. Вызывает интерес исследование особенностей программирования логических матриц в CPLD с целью рассмотрения возможностей повышения их отказоустойчивости в специальных высоконадёжных применениях. В статье анализируются принципы программирования матрицы логики, выполняется моделирование реализации конъюнкции в системе схемотехнического моделирования NI Multisim 10 firm National Instruments Electronics Workbench Group, предлагается и оценивается отказоустойчивая реализация матрицы логики.

**Ключевые слова:** логическая матрица, моделирование, конъюнкция, ПЛИС, Multisim.

**S.F. Tyurin<sup>1</sup>, A.S. Prokhorov<sup>2</sup>**<sup>1</sup>Perm National Research Polytechnic University, Perm, Russian Federation<sup>2</sup>JSC « ER-Telecom Holding», Perm, Russian Federation

## FAULT TOLERANT PROGRAMMABLE LOGIC ARRAY

The Boolean algebra of logic, logic functions are the basis of a mathematical apparatus of digital electronics and circuit engineering. At the beginning, from the moment of the invention of the relay in the middle of the 19th a century, they were implemented by contact-relay diagrams - serial-to-parallel and bridge, then after creation of transistors – at the microelectronic level. Small-scale integrated microcircuits - "hard" logic, then chips of average integration scale including "flexible" logic were created. In the 1970s the first programmable logic arrays which were used further in larg-scale integrated microcir-

cuits were released. Such structures are widely used during creation, for example, of control and monitoring devices. Programming was first once, at the factory, and then there was the possibility of programming by the user with the help of a special programmer. Then reprogrammable logic arrays were developed electrically. They are used also in the CPLD (complex programmable logic devices) programmable logic integrated circuits (PLIC), unlike FPGA (field-programmable gate array) where implementation in the form of a tree of the transferring transistors is used that in essence is the multiplexer selector or a fixed memory digit capacity of one bits with  $n$  address inputs. It is of interest to study the features of programming logic matrices in CPLD in order to consider the possibilities of increasing their fault tolerance in special highly reliable applications. The principles of programming the logic matrix are analyzed in the article, the simulation of the implementation of the conjunction in the NI Multisim 10 system of the National Instruments Electronics Workbench Group is performed, and the fault-tolerant implementation of the logic matrix is proposed and evaluated.

**Keywords:** Logic Array, Simulation, Conjunction, FGPA, Multisim.

**Введение.** Логические операции [1–3] широко используются при реализации управляющих [4–6] и кодирующих, и других цифровых устройств [7–9]. В программируемых логических устройствах (ПЛИС) типа FPGA (field-programmable gate array) [10–13] логические функции реализуются путем настройки деревьев передающих транзисторов значениями таблиц истинности, поэтому они получили название LUT (Look upTable). Однако ПЛИС другого типа – CPLD (complex programmable logic devices) [14] основаны на логике программируемых матриц, которые были изобретены в начале 70-х годов XX века [15–17]. Первые матрицы были однократно программируемыми с так называемыми пережигаемыми (плавкими) перемычками, затем появились электрически перепрограммируемые матрицы [17–19]. Логические матрицы содержат подматрицы «И» и «ИЛИ» и могут быть классифицированы в соответствии с тем, какие подматрицы программируются, а какие фиксированы. Такая классификация программируемых матриц [19] представлена на рис. 1.

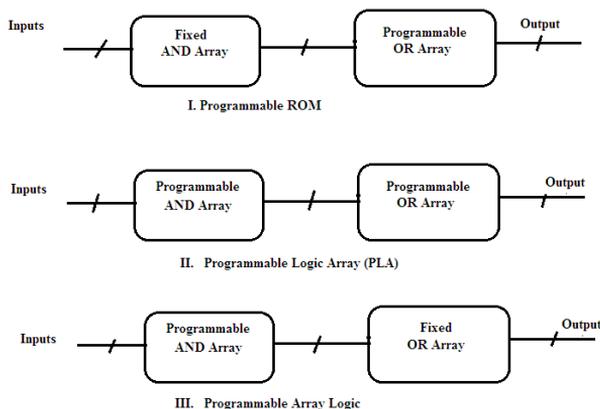
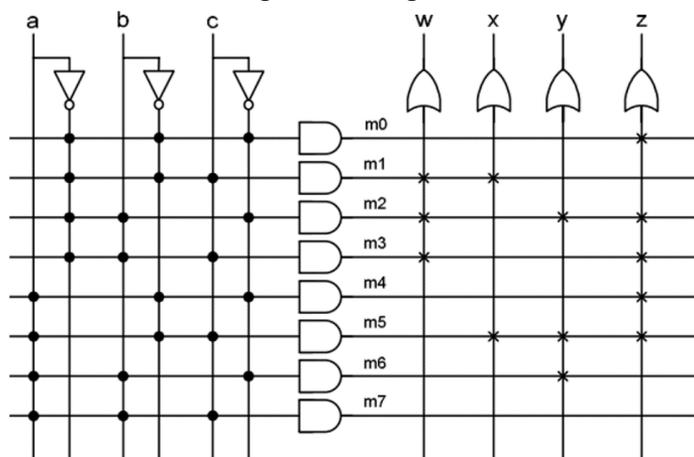
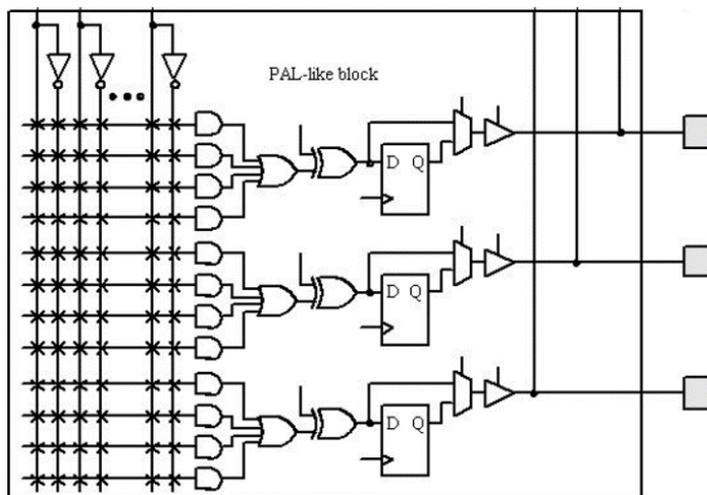


Рис. 1. Классификация программируемых матриц

Таким образом, в случае фиксированной матрицы «И» и программируемой матрицы «ИЛИ» получаем постоянную память (ПЗУ, ROM- read-only memory), для электрически перепрограммируемых устройств – это EEPROM (electrically erasable programmable read-only memory). Если программируются обе матрицы, получаем ПЛИМ – программируемую логическую матрицу (PLA – programmable logic array). В случае программирования только матрицы «И», получаем ПМЛ – программируемую матрицу логики (PAL – programmable array logic). Структура ПЛИМ и ПМЛ изображена на рис. 2.

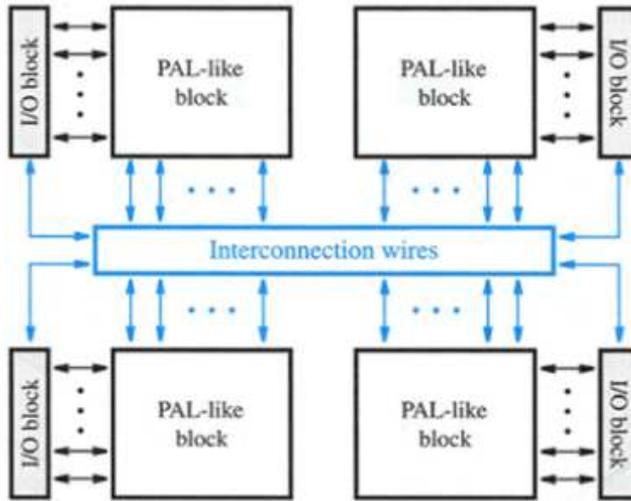


*a*



*б*

Рис. 2. Обобщённая структура логической матрицы: *a* – ПЛИМ (PLA);  
*б* – ПМЛ (PAL) в ПЛИС типа CPLD; *в* – структура логики CPLD



6

Рис. 2. Окончание

Однако приведённые структуры не раскрывают деталей программирования логики. Вызывает интерес рассмотрение особенностей программирования логических матриц с целью повышения их отказоустойчивости в специальных приложениях, например в радиационно-стойкой аппаратуре [11, 12].

**1. Принцип программирования логики.** Принцип программирования логики, например реализация некоторой конъюнкции, может заключаться в закорачивании либо в незакорачивании передающих МОП-транзисторов ( $A$ ,  $B$ ,  $C$ ,  $D$ ) по заданным переменным [17–19] (рис. 3, *a*).

Это может быть выполнено однократно – плавкой перемычкой либо другим транзистором, программируемым электрически. Однако возникают проблемы управления выходными транзисторами строк (см. рис. 3, *a*) при большом числе переменных [20], поэтому используются инверторы в качестве буферных элементов. Кроме того, резисторы реализуются с использованием тех же МОП-транзисторов переменных (рис. 3, *б*).

Программирование по  $i$ -й переменной  $j$ -й функции, зависящей от одной переменной, с учётом её возможного инверсного значения, может быть представлено следующим образом:

$$f_j = (x_i \vee s_i)(\bar{x}_i \vee \bar{s}_i). \quad (1)$$

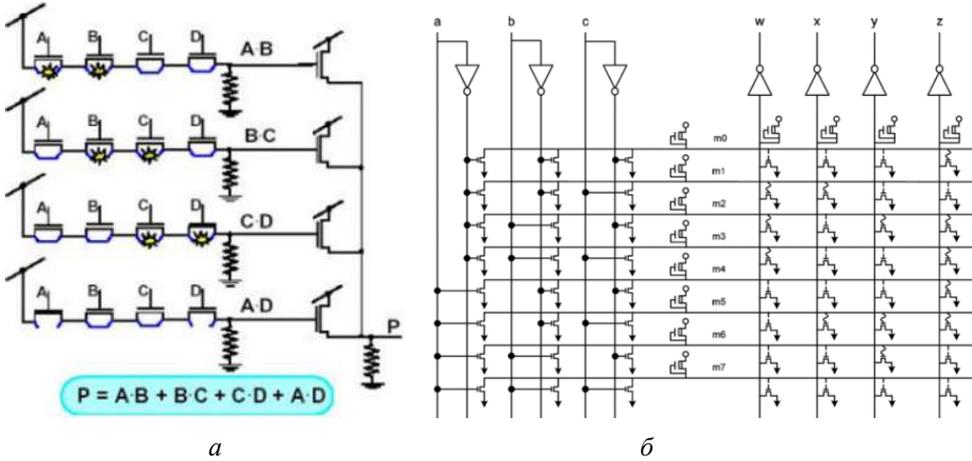


Рис. 3. Программирование логики: *a* – с использованием резисторов и с управлением затвором МОП-транзисторов, *б* – с реализацией резисторов на основе МОП-транзисторов и с буферными инверторами

Для функции, содержащей  $n$  переменных, получим:

$$f_j = \bigwedge_{i=1}^n (x_i \vee s_i)(\bar{x}_i \vee \bar{s}_i); j = 1, k \quad (2)$$

Причём для  $s$  выполняются следующие условия:  $s_i \bar{s}_i = 0; s_i \vee \bar{s}_i = 1$  для существенной переменной,  $s_i \bar{s}_i = 1$  для несущественной переменной.

### 3. Отказоустойчивый узел программирования переменной.

Для обеспечения отказоустойчивости программируемой логики используем результаты, описанные в [21–24]. Получим выражение, обеспечивающее парирование одного отказа транзистора переменной или транзистора настройки в ДНФ:

$$f_j = \bigwedge_{i=1}^n [(x_{i,1}x_{i,2} \vee x_{i,3}x_{i,4}) \vee (s_{i,1}s_{i,2} \vee s_{i,3}s_{i,4})] \quad (3)$$

$$[(\bar{x}_{i,1}\bar{x}_{i,2} \vee \bar{x}_{i,3}\bar{x}_{i,4}) \vee (\bar{s}_{i,1}\bar{s}_{i,2} \vee \bar{s}_{i,3}\bar{s}_{i,4})]; j = 1, k.$$

В КНФ выражение имеет вид:

$$f_j = \bigwedge_{i=1}^n [(x_{i,1} \vee x_{i,2})(x_{i,3} \vee x_{i,4}) \vee (s_{i,1} \vee s_{i,2})(s_{i,3} \vee s_{i,4})] \quad (4)$$

$$[(\bar{x}_{i,1} \vee \bar{x}_{i,2})(\bar{x}_{i,3} \vee \bar{x}_{i,4}) \vee (\bar{s}_{i,1} \vee \bar{s}_{i,2})(\bar{s}_{i,3} \vee \bar{s}_{i,4})]; j = 1, k.$$

Моделирование отказоустойчивого узла программирования одной переменной без учёта необходимости моделирования инвертора и резистора, реализованного на дополнительном транзисторе, изображено на рис. 4.

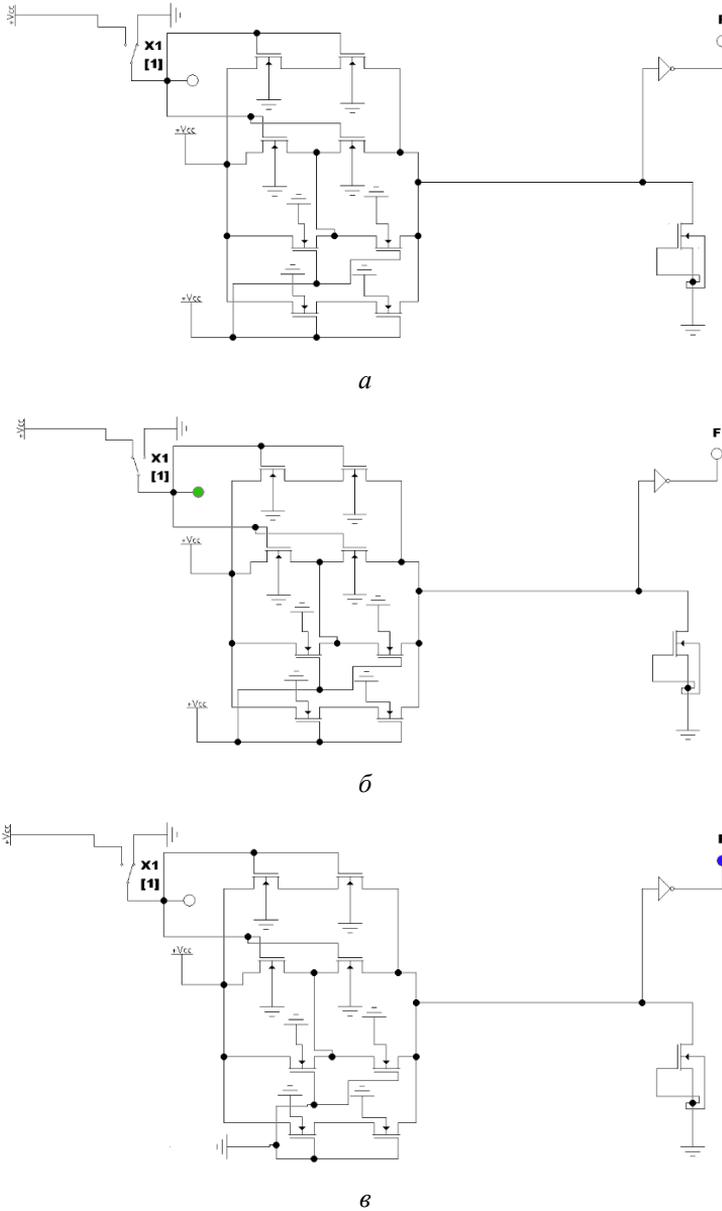


Рис. 4. Моделирование отказоустойчивого узла программирования одной переменной:  
*a* – переменная исключена, на входе 0; *б* – переменная исключена, на входе 1;  
*в* – переменная включена, на входе 0; *г* – переменная включена, на входе 1

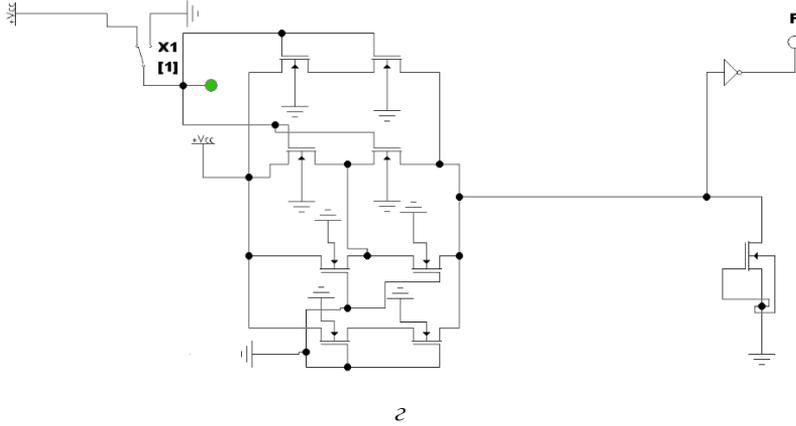


Рис. 4. Окончание

Таким образом, парировается отказ одного транзистора как в блоке переменной, так и в блоке настройки.

Для ДНФ парирование  $r$  отказов по некоторой  $i$ -й переменной описывается выражением:

$$f_j = \&\{ [\vee_{\chi=1}^{r+1} (\& x_{i,\chi})] \vee [\vee_{\chi=1}^{r+1} (\& s_{i,\chi})] \} \{ [\vee_{\chi=1}^{r+1} (\& x_{i,\chi})] \vee [\vee_{\chi=1}^{r+1} (\& s_{i,\chi})] \}; \quad j = 1, k. \quad (5)$$

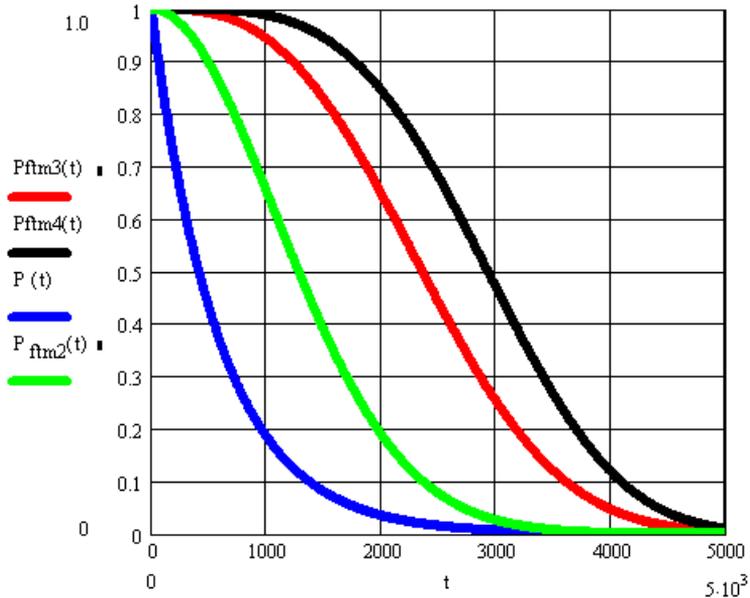
Вероятность безотказной работы для экспоненциальной модели имеет вид:

$$\sum_{i=0}^r C_{(r+1)^2}^i \{ e^{-[(r+1)^2-i]\lambda t^\alpha} \cdot (1 - e^{-\lambda t^\alpha})^i \}. \quad (6)$$

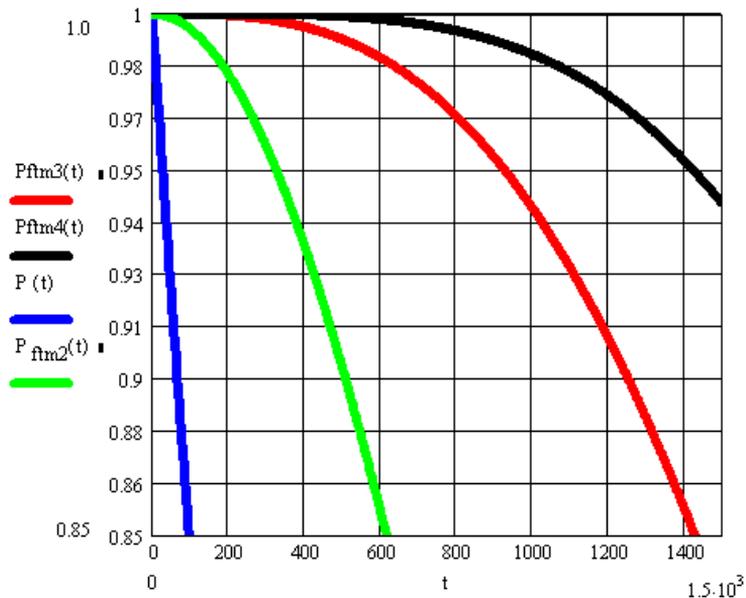
Так, в случае  $r = 3$  парировается отказ любых 3 транзисторов по каждой переменной:

$$e^{-(16)\lambda t} + 16 \cdot e^{-15\lambda t} (1 - e^{-\lambda t}) + 120 \cdot e^{-14\lambda t} (1 - e^{-\lambda t})^2 + 560 \cdot e^{-13\lambda t} (1 - e^{-\lambda t})^3. \quad (7)$$

Графики изменения вероятностей безотказной работы нерезервированного транзистора  $P(t)$ , схемы FCT, парирующей один отказ  $P(t)_{\text{fm}2}$ , схемы FCT, парирующей два отказа  $P(t)_{\text{fm}3}$  и схемы FCT, парирующей три отказа  $P(t)_{\text{fm}4}$  изображены на рис. 5.



a



б

Рис. 5. Графики изменения вероятностей безотказной работы нерезервированного транзистора, схемы FCT, парирующей один отказ  $P(t)_{ftm2}$ , схемы FCT, парирующей два отказа  $P(t)_{ftm3}$  и схемы FCT, парирующей три отказа  $P(t)_{ftm4}$  при интенсивности отказов 10 в минус пятой степени  $1/4$ ; а – в диапазоне вероятности от 1 до 0; б – в диапазоне вероятности от 1 до 0,4

**Выводы.** Таким образом, программирование конъюнкций в логической матрице может выполняться путём включения либо исключения данной переменной, либо её инверсии с помощью транзистора настройки. Для несущественной переменной исключаются и она сама, и её инверсия. При этом в случае истинности конъюнкции (равенства конъюнкции единице) с помощью нагрузочного резистора, которой реализуется на дополнительном передающем транзисторе, и инвертора на выходе матрицы «И» (на выходе инвертора) формируется логический ноль. В случае ложности конъюнкции (равенства конъюнкции нулю) на выходе инвертора формируется единица. Реализация матрицы «ИЛИ» может быть выполнена путём включения или исключения заданных выходов матрицы «И» с помощью транзисторов настройки матрицы «ИЛИ» для получения требуемой ДНФ. Для повышения вероятности безотказной работы логической матрицы возможно использовать транзисторное резервирование. Моделирование подтверждает работоспособность предложенного резервирования, а оценка вероятности безотказной работы доказывает высокую эффективность такой отказоустойчивой матрицы в высоконадёжных приложениях, например в аэрокосмической области, где требуется обеспечить работоспособность в условиях воздействия негативных внешних факторов.

### **Библиографический список**

1. Кузнецов О.П., Адельсон-Вельский Г.М. Дискретная математика для инженера. – М.: Энергоатомиздат, 1988. – 450 с.
2. Новиков Ф.А. Дискретная математика для программиста. – СПб.: Питер, 2001. – 502 с.
3. Горбатов В.А. Фундаментальные основы дискретной математики. Информационная математика: учеб. пособие для вузов. – М.: Наука, 2000. – 540 с.
4. Угрюмов Е.П. Цифровая схемотехника: учеб. пособие. – 2-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2007. – 782 с.
5. Самосинхронный вычислитель для высоконадёжных применений / Ю.А. Степченко [и др.] // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС): сб. тр. всерос. науч.-техн. конф. / Институт проблем проектирования в микроэлектронике РАН. – М., 2010. – № 1.

6. Библиотека самосинхронных элементов для технологии БМК / Ю.А. Степченков [и др.] // Проблемы разработки перспективных микроэлектронных систем – 2006: сб. науч. тр. / под общ. ред. А.Л. Степковскового / Институт проблем проектирования в микроэлектронике РАН. – М., 2006. – С. 259–264.

7. Тюрин С.Ф. Аляев Ю.А. Дискретная математика: практическая дискретная математика и математическая логика. – М.: Финансы и статистика, 2010. – 394 с.

8. Тюрин С.Ф., Аляев Ю.А. Дискретная математика: тест-драйв по дискретной математике и математической логике. – Пермь: Изд-во Перм. нац. исслед. политехн. ун-та, 2014. – 231 с.

9. Тюрин С.Ф. Дерево транзисторов для реализации систем логических функций // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2015. – № 2(14). – С. 37–45.

10. Строгонов А., Цыбин С. Программируемая коммутация ПЛИС: взгляд изнутри [Электронный ресурс]. – URL: [http://www.kite.ru/articles/plis/2010\\_11\\_56.php](http://www.kite.ru/articles/plis/2010_11_56.php) (дата обращения: 11.06.2017).

11. Carl Carmichael. 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: 07.12.2016).

12. Xilinx Reduces Risk and Increases Efficiency for IEC61508 and ISO26262 Certified Safety Applications. WP461 (v1.0) April 9, 2015 [Электронный ресурс]. – URL: [http://www.xilinx.com/support/documentation/white\\_papers/wp461-functional-safety.pdf](http://www.xilinx.com/support/documentation/white_papers/wp461-functional-safety.pdf) (дата обращения: 20.12.2016).

13. QPro Virtex-II 1.5V Platform FPGAs. DS122 (v3.0) April 7 2014 [Электронный ресурс]. – URL: [http://www.xilinx.com/support/documentation/data\\_sheets/ds122.pdf](http://www.xilinx.com/support/documentation/data_sheets/ds122.pdf) (дата обращения: 20.12.2016).

14. CPLD (Complex Programmable Logic Device) [Электронный ресурс]. – URL: <http://www.myshared.ru/slide/981511/> (дата обращения: 09.06.2017).

15. Виды программируемой логики [Электронный ресурс]. – URL: <http://www.pvsm.ru/programmirovanie/87810> (дата обращения: 10.06.2017).

16. Programmable Logic Devices [Электронный ресурс]. – URL: [http://ee.sharif.edu/~logic\\_circuits\\_t/readings/PLD.pdf](http://ee.sharif.edu/~logic_circuits_t/readings/PLD.pdf) (дата обращения: 04.06.2017).

17. Программируемая логика и её применение в микропроцессорных системах [Электронный ресурс]. – URL: <http://lektsii.org/7-10275.html> (дата обращения: 08.06.2017).

18. Arijit Banerjee, Sergiu Mosanhttp. Modelling and Design of A 45nm SLC 3D NAND Flash CPL [Электронный ресурс]. – URL: [//venividiwiki.ee.virginia.edu/mediawiki/images/c/cd/ECE7332\\_FinalPresentation.pdf](http://venividiwiki.ee.virginia.edu/mediawiki/images/c/cd/ECE7332_FinalPresentation.pdf) (дата обращения: 09.06.2017).

19. Stephen Brown, Jonathan Rose. Architecture of FPGAs and CPLDs: A Tutorial [Электронный ресурс]. – URL: <http://www.eecg.toronto.edu/~jayar/pubs/brown/survey.pdf> (дата обращения: 10.06.2017).

20. Ульман Дж.Д. Вычислительные аспекты СБИС / пер. с англ. А.В. Неймана; под ред. П.П. Пархоменко. – М.: Радио и связь, 1990. – 480 с.

21. Сайт разработчика National Instruments [Электронный ресурс]. – URL: <http://www.ni.com/multisim/> (дата обращения: 08.06.2017).

22. Тюрин С.Ф. Статическая оперативная память на основе отказоустойчивой ячейки базового матричного кристалла // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2016. – № 1(17). – С. 16–27.

23. Тюрин С.Ф. Автоматический синтез комбинационного автомата с помощью State Machine Editor системы QUARTUSII // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2015. – № 3(15). – С. 60–69.

24. Тюрин С.Ф. Радиационно-устойчивая ячейка SRAM // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2014. – № 4(12). – С. 14–30.

## **References**

1. Kuznetsov O.P., Adel'son-Vel'skii G.M. Diskretnaia matematika dlia inzhenera [Discrete mathematics for inzhenera]. Moscow: Energoatomizdat, 1988. 450 p.

2. Novikov F.A. Diskretnaia matematika dlia programmista [Discrete mathematics for programmer]. Saint Petersburg: Piter, 2001. 502 p.
3. Gorbatov V.A. Fundamental'nye osnovy diskretnoi matematiki. Informatsionnaia matematika [Fundamental bases discrete mathematics. Information mathematics]. Moscow: Nauka, 2000. 540 p.
4. Ugriumov E.P. Tsifrovaia skhemotekhnika [Digital circuitry]. 2nd ed. Saint Petersburg: BKhV-Peterburg, 2007. 782 p.
5. Stepchenkov Iu.A. [et al.]. Samosinkhronnyi vychislitel' dlia vysokonadezhnykh primenenii [Self-timed computer for high-reliability applications]. *Sbornik trudov vs Rossiiskoi nauchno-tekhnikeskoi konferentsii "Problemy razrabotki perspektivnykh mikro- i nanoelektronnykh sistem (MES)"*. Moscow: Institut problem proektirovaniia v mikroelektronike Rossiiskoi akademii nauk, 2010, no. 1.
6. Stepchenkov Iu.A. [et al.]. Biblioteka samosinkhronnykh elementov dlia tekhnologii BMK [Library elements for self-timed FPGA technology]. *Sbornik nauchnykh trudov "Problemy razrabotki perspektivnykh mikroelektronnykh sistem – 2006"*. Ed. A.L. Stempkovskogo. Moscow: Institut problem proektirovaniia v mikroelektronike Rossiiskoi akademii nauk, 2006, pp. 259-264.
7. Tiurin S.F. Aliaev Iu.A. Diskretnaia matematika: prakticheskaia diskretnaia matematika i matematicheskaia logika [Discrete mathematics and mathematical logic]. Moscow: Finansy i statistika, 2010. 394 p.
8. Tiurin S.F., Aliaev Iu.A. Diskretnaia matematika: test-draiv po diskretnoi matematike i matematicheskoi logike [Discrete mathematics: practical discrete mathematics and mathematical logic]. Permskii natsional'nyi issledovatel'skii politekhnicheskii universitet, 2014. 231 p.
9. Tiurin S.F. Derevo tranzistorov dlia realizatsii sistem logicheskikh funktsii [Tree of transistors for implementation of logic functions systems]. *Vestnik Permskogo natsional'nogo issledovatel'skogo politekhnicheskogo universiteta. Elektrotehnika, informatsionnye tekhnologii, sistemy upravleniia*, 2015, no. 2(14), pp. 37-45.
10. Strogonov A., Tsybin S. Programmiruemaia kommutatsiia PLIS: vzgliad iznutri [Programmable switching FPGA: a view from the inside], available at: [http://www.kit-e.ru/articles/plis/2010\\_11\\_56.php](http://www.kit-e.ru/articles/plis/2010_11_56.php) (accessed 11 June 2017).

11. Carl Carmichael. 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 07 December 2016).

12. Xilinx Reduces Risk and Increases Efficiency for IEC61508 and ISO26262 Certified Safety Applications. WP461 (v1.0) April 9, 2015, available at: [http://www.xilinx.com/support/documentation/white\\_papers/wp461-functional-safety.pdf](http://www.xilinx.com/support/documentation/white_papers/wp461-functional-safety.pdf) (accessed 20 December 2016).

13. QPro Virtex-II 1.5V Platform FPGAs. DS122 (v3.0) April 7 2014, available at: [http://www.xilinx.com/support/documentation/data\\_sheets/ds122.pdf](http://www.xilinx.com/support/documentation/data_sheets/ds122.pdf) (accessed 20 December 2016).

14. CPLD (Complex Programmable Logic Device), available at: <http://www.myshared.ru/slide/981511/> (accessed 09 June 2017).

15. Vidy programmiruemoi logiki [Types of programmable logic], available at: <http://www.pvsm.ru/programmirovanie/87810> (accessed 10 June 2017).

16. Programmable Logic Devices, available at: [http://ee.sharif.edu/~logic\\_circuits\\_t/readings/PLD.pdf](http://ee.sharif.edu/~logic_circuits_t/readings/PLD.pdf) (accessed 04 June 2017).

17. Programmiruemaia logika i ee primenenie v mikroprotssornykh sistemakh [Programmable logic and its application in microprocessor systems], available at: <http://lektsii.org/7-10275.html> (accessed 08 June 2017).

18. Arijit Banerjee, Sergiu Mosanhttp. Modelling and Design of A 45nm SLC 3D NAND Flash CPL, available at: [//venividiwiki.ee.virginia.edu/media-wiki/images/c/cd/ECE7332\\_FinalPresentation.pdf](http://venividiwiki.ee.virginia.edu/media-wiki/images/c/cd/ECE7332_FinalPresentation.pdf) (accessed 09 June 2017).

19. Stephen Brown, Jonathan Rose. Architecture of FPGAs and CPLDs: A Tutorial, available at: <http://www.eecg.toronto.edu/~jayar/pubs/brown/survey.pdf> (accessed 10 June 2017).

20. Ul'man Dzh.D. Vychislitel'nye aspekty SBIS [Computational Aspects of VLSI]. Ed. P.P. Parkhomenko. Moscow: Radio i sviaz', 1990. 480 p.

21. Sait razrabotchika National Instruments [National Instruments Developer Site], available at: <http://www.ni.com/multisim/> (accessed 08 June 2017).

22. Tiurin S.F. Sticheskaia operativnaia pamiat' na osnove otkazoustoichivoi iacheiki bazovogo matrichnogo kristalla [Static RAM based on a fault-tolerant cell of the base matrix crystal]. *Vestnik Permskogo natsional'nogo issledovatel'skogo politekhnicheskogo universiteta. Elektrotekhnika, informatsionnye tekhnologii, sistemy upravleniia*, 2016, no. 1(17), pp. 16-27.

23. Tiurin S.F. Avtomaticheskii sintez kombinatsionnogo avtomata s pomoshch'iu State Machine Editor sistemy QUARTUSII [Automatic synthesis of a combinational machine using the State Machine Editor of the QUARTUSII system]. *Vestnik Permskogo natsional'nogo issledovatel'skogo politekhnicheskogo universiteta. Elektrotehnika, informatsionnye tehnologii, sistemy upravleniia*, 2015, no. 3(15), pp. 60-69.

24. Tiurin S.F. Radiatsionno-ustoichivaia iacheika SRAM [Radiation-resistant SRAM cell]. *Vestnik Permskogo natsional'nogo issledovatel'skogo politekhnicheskogo universiteta. Elektrotehnika, informatsionnye tehnologii, sistemy upravleniia*, 2014, no. 4(12), pp. 14-30.

### Сведения об авторах

**Тюрин Сергей Феофентович** (Пермь, Россия) – заслуженный изобретатель Российской Федерации, доктор технических наук, профессор кафедры автоматике и телемеханики Пермского национального исследовательского политехнического университета (614990, Пермь, Комсомольский пр., 29, e-mail: tyurinsergfeo@yandex.ru).

**Прохоров Андрей Сергеевич** (Пермь, Россия) – программист АО «ЭР-Телеком Холдинг», аспирант кафедры автоматике и телемеханики Пермского национального исследовательского политехнического университета университета (614990, Пермь, ш. Космонавтов., 111, корпус 43, e-mail: прохор007@yandex.ru).

### About the authors

**Tyurin Sergey Feofentovich** (Perm, Russian Federation) is a Honored Inventor of the Russian Federation, Doctor of Technical Sciences, Professor at the Department of Automation and Telemechanics Perm National Research Polytechnic University (614990, Perm, 29, Komsomolsky pr., e-mail: tyurinsergfeo@yandex.ru).

**Prokhorov Andrey Sergeevich** (Perm, Russian Federation) is a Programmer of JSC “ER-Telecom Holding”, Postgraduate Student at the Department of Automation and Telemechanics Perm National Research Polytechnic University (614990, Perm, sh. Kosmonavtov, 111, korpus 43, e-mail:прохор007@yandex.ru).

Получено 31.07.2017