

УДК 681.32

**С.Ф. Тюрин<sup>1</sup>, Ю.А. Аляев<sup>2</sup>**

<sup>1</sup>Пермский национальный исследовательский политехнический университет,  
Пермь, Россия

<sup>2</sup>Российская академия народного хозяйства и государственной службы  
при Президенте Российской Федерации, Пермь, Россия

## **МЕТОДИКА ПРОВЕДЕНИЯ ЛАБОРАТОРНОГО ЗАНЯТИЯ «СИНТЕЗ АВТОМАТА-ФОРМИРОВАТЕЛЯ»**

В настоящее время одним из трендов российской науки и образования является так называемое импортозамещение электронной компонентной базы. На кафедре автоматики и телемеханики проводятся исследования в этом направлении, связанные с тематикой управления, контроля и испытаний авиадвигателей, аппаратуры связи и телекоммуникаций, комплексной защиты информации. Развивается творческая кооперация с академическими институтами РАН. Большое значение уделяется инженерной подготовке студентов и аспирантов по схемотехнической тематике. В этом плане в связи с ограничениями вновь вводимых ФГОС ВО возникает проблема дефицита времени на лабораторный практикум. Среди путей её решения могут быть более тесная «связка» дисциплин, изучаемых в одном семестре, с целью интенсификации учебного процесса, а также использование новых методических приемов, позволяющих проведение ускоренных экспериментов с помощью систем схемотехнического моделирования. Так, расчетная часть некоторых лабораторных занятий по дисциплине «Схемотехника» может быть выполнена в рамках дисциплины «Дискретная математика» в теме «Автоматы». Задания на синтез могут масштабироваться с учетом выделяемого времени, а сам эксперимент может проводиться на основе заготовок – шаблонов. Предлагаются методические приёмы ускоренного проведения лабораторного занятия по синтезу автомата-формирователя заданных последовательностей по дисциплине «Схемотехника» на основе схем-шаблонов, в которые надо внести изменения в соответствии с вариантом задания. Варианты заданий можно формировать в виде имён и фамилий студентов. Описывается методика, которая позволяет в рамках 2–4-часового занятия провести как расчёты, так и эксперимент.

**Ключевые слова:** синтез автомата-формирователя, схема-шаблон, лабораторное занятие.

**S.F. Tyurin<sup>1</sup>, Yu.A. Alyaev<sup>2</sup>**

<sup>1</sup>Perm National Research Polytechnic University, Perm, Russian Federation

<sup>2</sup>Russian presidential academy of national economy and public administration,  
Perm, Russian Federation

## **TECHNIQUE OF A FINITE-STATE MACHINE SEQUENCES SET GENERATOR LAB**

At present, one of the trends in Russian science and education is the so-called import substitution of electronic components. At the Department of Automation and Telemechanics, research is being carried out in this direction, related to the subjects of control and testing of aircraft engines, communication and telecommunications equipment, and complex information protection. There is creative cooperation with the academic institutes of the Russian Academy of Sciences. Great importance is given to the engineering preparation of students and post-graduate students in circuit engineering. In this regard, in connection with the limitations of the newly introduced standards of higher education, there arises the problem of a shortage of time for laboratory practice. One of the ways to solve it may be a closer "bundle" of disciplines studied in one semester in order to intensify the learning process, the use of new methodical techniques that allow carrying out accelerated experiments using circuit simulation systems. Thus, the estimated part of some laboratory exercises on discipline "Circuitry" can be performed within the discipline "Discrete Mathematics" in the topic "Finite-State Machine ". Tasks for synthesis can be scaled in view of the time allocated, and the experiment itself can be conducted on the basis of blanks - templates. The article offers methodological techniques of accelerated laboratory research on the synthesis of finite-state machine-sequences set generator for the discipline "Circuitry" on the basis of templates in which it is necessary to make changes in line with the job option. Possible tasks can be shaped in the form of names and surnames of students. The paper proposes a method that allows within 2–4-hour lesson hold both calculations and experiment.

**Keywords:** Finite-State Machine-Sequences Set Generator, Templates, Lab.

**Введение.** Кафедра автоматике и телемеханики участвует в исследованиях, проводимых Институтом проблем информатики Российской академии наук (Федерального исследовательского центра «Информатика и управление» Российской академии наук) в области так называемых самосинхронных схем [1, 2]. Большое внимание также уделяется тематике программируемой логики [3, 4]. Развивается тематика нейронечёткого управления [5, 6]. В этом плане, а также в плане задач импортозамещения элементной базы актуально совершенствование схемотехнического направления [7]. В учебных дисциплинах «Дискретная математика», «Математическая логика» [8], «Схемотехника» [9] часто рассматривается в качестве простого примера автомат-распознаватель одной последовательности (кодовый замок или, говоря современным языком, – распознаватель PIN-кода) [8, 9]. Такой пример позволяет осуществить своего рода преемственность обучения, когда расчёты осуществляются на основе математического аппарата

дискретной математики и математической логики, а проектирование схемы позволяет воплотить математику в «железо» [10]. Кроме этого подобный пример позволяет формировать большое число вариантов, что исключает «тупое» списывание – копирование студентами «отстающими» у студентов «передовиков». С другой стороны, это даёт возможность значительно сократить время на проведение эксперимента, так как используются как типовые расчёты, так и типовая схема-шаблон, в котором надо выполнить изменения по варианту. Наконец, такой подход упрощает преподавателю проверку знаний, умений и навыков студента. Причём небольшое увеличение размерности задачи может привести к изменению шаблона, так что студент, справившийся с заданием, может быть смело «номинирован» на высший балл. Следовательно, если детально рассмотрен синтез по последовательности, например 0132 и ей подобным, то последовательность 0137 требует уже выхода за пределы «шаблона». А уменьшение размерности задания вполне годится для «ускоренного» учебного процесса на заочном отделении. Так же и на зачёте (экзамене) может быть использована совсем простая последовательность – типа 013.

Так строятся лишь простейшие асинхронные автоматы с неповторяющимися символами соседнего кода.

При достаточном времени на лабораторные занятия (можно вынести эту тематику и на курсовое проектирование) возникает возможность исследования более сложных автоматов – с синхронными триггерами и тактовым генератором. Желательно «подвести» студентов к вопросу: «А ведь в PIN-коде цифры могут повторяться, как же строить такой автомат?». Это уже чисто синхронные автоматы. Кроме того, имеют большое практическое значение и другие классы автоматов: автоматы-формирователи, автоматы-преобразователи последовательностей. Автоматы-формирователи моделируют некоторые языки, рассматриваемые в курсе дискретной математики и математической логики. Рассмотрим особенности синтеза синхронного автомата-формирователя на простом примере.

**Абстрактный синтез автомата-формирователя заданных последовательностей {(маша), (саша)}.** Синтезируем автомат-формирователь простого языка {(маша), (саша)} [8]. Закодируем символы переменными  $z_2, z_1$ : «м» – 11, «с» – 00, «а» – 01, «ш» – 10. Построим граф переходов-выходов (рис. 1).

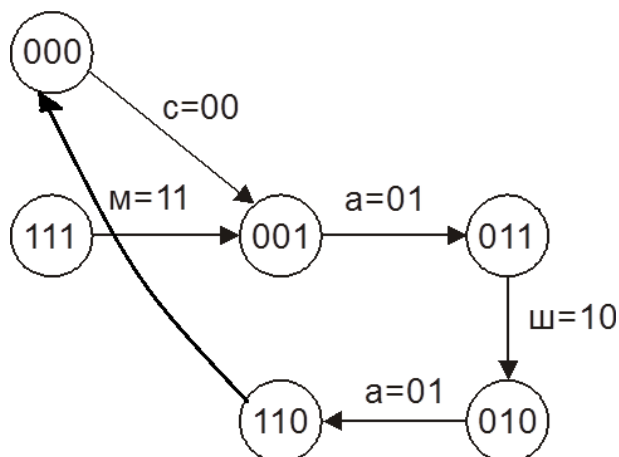


Рис. 1. Граф переходов автомата-формирователя «саша-маша»

Используем карту Карно на три переменные состояний  $Y_3, Y_2, Y_1$  для получения таблицы переходов-выходов [8] (рис. 2).

$Y_3(t)$	$Y_2 Y_1(t)$			
	00	01	11	10
0	000 001 00	001 011 01	011 010 10	010 110 01
1	100 ~	101 ~	111 001 11	110 000 00
				$\frac{Y_3 Y_2 Y_1(t+1)}{z_2 z_1}$

Рис. 2. Таблица переходов-выходов автомата-формирователя «саша-маша»

Получили таблицу переходов-выходов синхронного автомата, изменяющего своё состояние по синхросигналу, который никак не указан в таблице. В такой таблице вообще нет устойчивых состояний! Рассмотрим функцию  $Y_3(t+1)$  (рис. 3).

Y <sub>3</sub> (t)	Y <sub>2</sub> Y <sub>1</sub> (t)			
	00	01	11	10
0	000 <u>0</u>	001 <u>0</u>	011 <u>0</u>	010 <u>1</u>
1	100	101	111 <u>0</u>	110 <u>0</u>
				<u>Y<sub>3</sub>(t+1)</u>

Рис. 3. Определение рабочих наборов функции Y<sub>3</sub>(t+1)

Легко видеть, что согласно рис. 3 получаем: Y<sub>3</sub>(t+1) = 2[0,1,3,6,7]. Проанализируем функцию Y<sub>2</sub>(t+1) (рис. 4).

Y <sub>3</sub> (t)	Y <sub>2</sub> Y <sub>1</sub> (t)			
	00	01	11	10
0	000 <u>0</u>	001 <u>1</u>	011 <u>1</u>	010 <u>1</u>
1	100	101	111 <u>0</u>	110 <u>0</u>
				<u>Y<sub>2</sub>(t+1)</u>

Рис. 4. Определение рабочих наборов функции Y<sub>2</sub>(t+1)

Получаем: Y<sub>2</sub>(t+1) = 1,3,2[0,6,7]. Функция Y<sub>1</sub>(t+1) представлена на рис. 5.

Y <sub>3</sub> (t)	Y <sub>2</sub> Y <sub>1</sub> (t)			
	00	01	11	10
0	000 <u>1</u>	001 <u>1</u>	011 <u>0</u>	010 <u>0</u>
1	100	101	111 <u>1</u>	110 <u>0</u>
				<u>Y<sub>1</sub>(t+1)</u>

Рис. 5. Определение рабочих наборов функции Y<sub>1</sub>(t+1)

Таким образом,  $Y_1(t+1) = 0,1,7[3,2]$ .

Определим функции выходов. Функция  $z_2$  (рис. 6).

$Y_3(t)$	$Y_2 Y_1(t)$			
	00	01	11	10
0	000 $\bar{0}$	001 $\bar{0}$	011 $\bar{1}$	010 $\bar{0}$
1	100	101	111 $\bar{1}$	110 $\bar{0}$
				$\bar{z}_2$

Рис. 6. Определение рабочих наборов функции  $z_2$

Следовательно,  $z_2 = 3,7[0,1,2]$ . Наконец, функция  $z_1$  (рис. 7).

$Y_3(t)$	$Y_2 Y_1(t)$			
	00	01	11	10
0	000 $\bar{0}$	001 $\bar{1}$	011 $\bar{0}$	010 $\bar{1}$
1	100	101	111 $\bar{1}$	110 $\bar{0}$
				$\bar{z}_1$

Рис. 7. Определение рабочих наборов функции  $z_1$

Таким образом,  $z_1 = 1,2,7[0,3]$ . Все эти стандартные расчёты (конечно, желательно использовать короткие имена: «Вова» вместо «Владимир» и т.д.) могут выполняться при изучении дискретной математики и математической логики. При изучении схемотехники полученные логические функции «воплощаются» в реальную схему.

**Моделирование автомата-формирователя заданных последовательностей {(маша), (саша)}.** Создадим шаблон для моделирования автомата на «гибкой» логике в Electronics Workbench [10] на основе дешифратора 74154 (рис. 8).

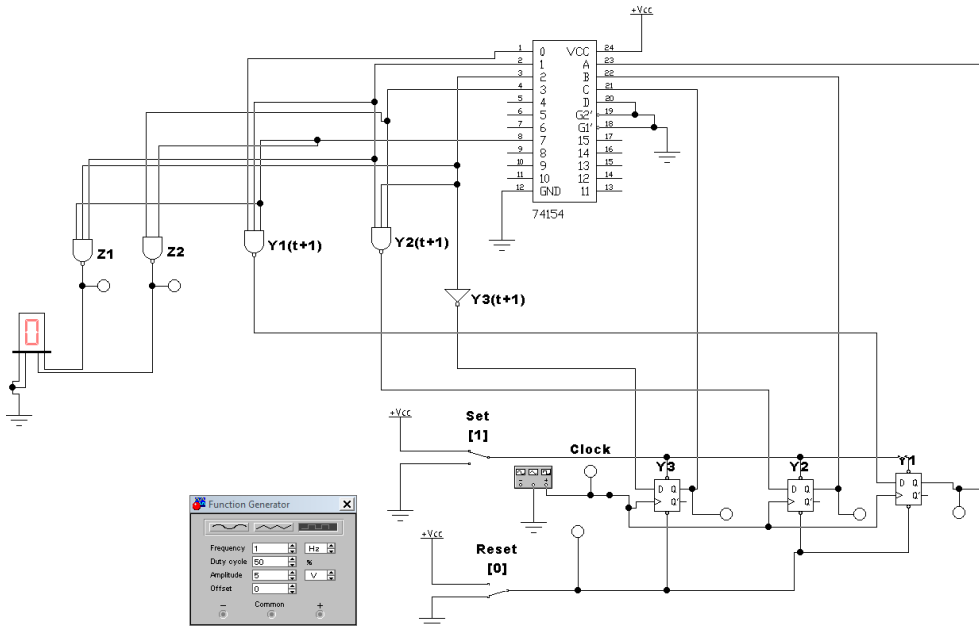


Рис. 8. Шаблон автомата-формирователя, настроенный на последовательности «саша-маша», состояние 000 на выходе «с» (00)

Функции переходов и выходов реализуются путём объединения по И-НЕ выходов дешифратора, соответствующих рабочим наборам (так как выходы дешифратора – инверсные). Для отображения цифровых эквивалентов букв используем семисегментный индикатор.

По сигналу Reset автомат «обнуляется», т.е. устанавливается в первое начальное состояние – формируется слово «саша». По сигналу Set автомат устанавливается во второе начальное состояние – «все единицы», формируется слово «маша».

Начинаем с нулевого состояния, на выходе буква «с» (см. рис. 8). Далее формируется буква «а» (рис. 9).

После этого автомат переходит в состояние 011, формируется буква «ш» (рис. 10).

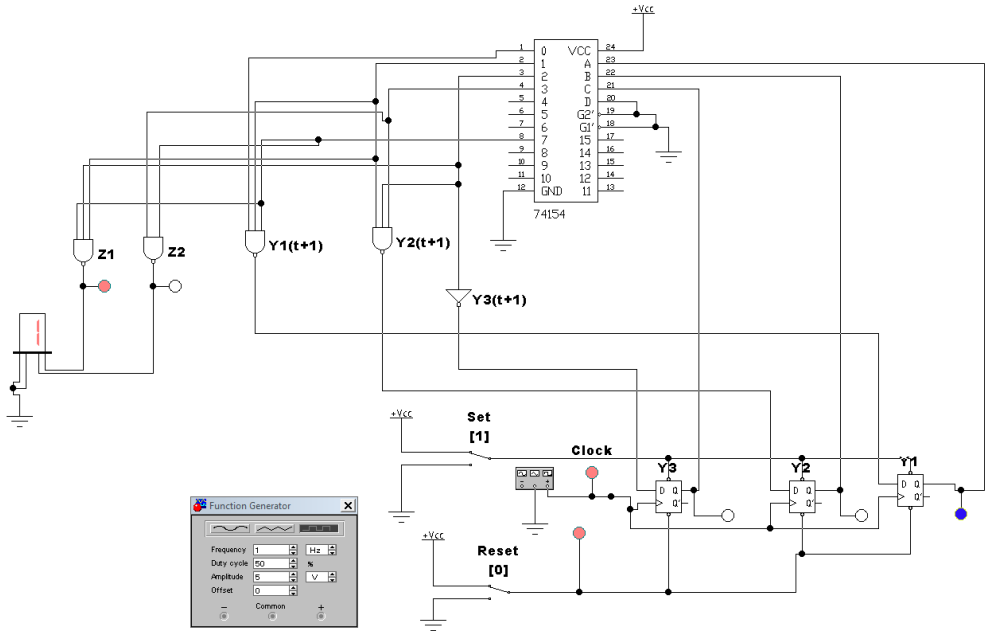


Рис. 9. Автомат-формирователь «саша-маша», состояние 001 на выходе «а» (1)

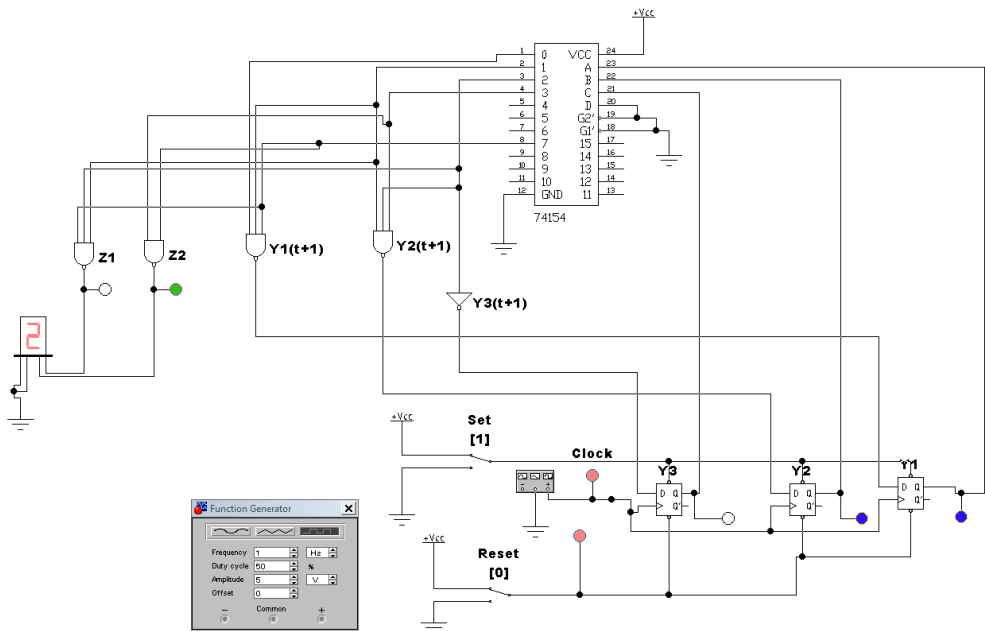


Рис. 10. Автомат-формирователь «саша-маша», состояние 011 на выходе «ш» (2)



Далее автомат в состоянии 010 опять формирует букву «а» (рис. 11).

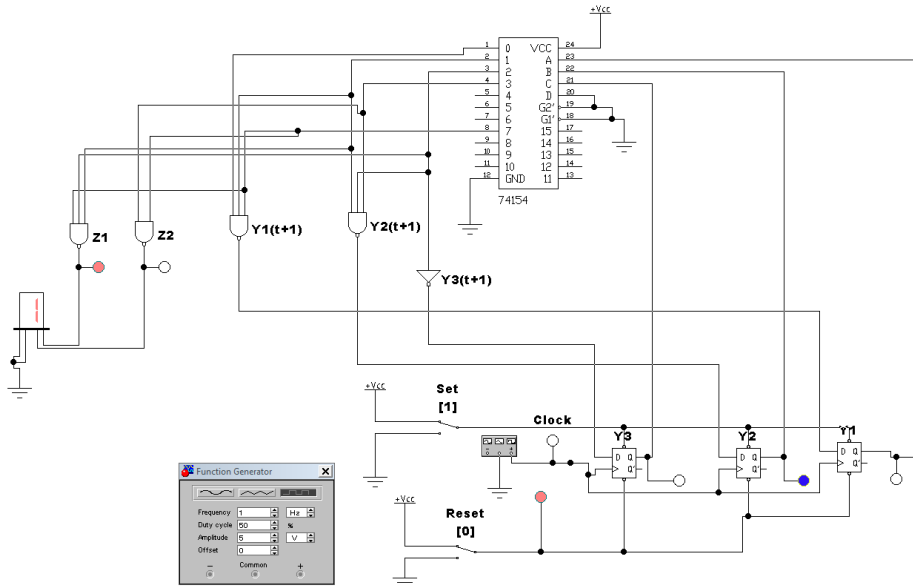


Рис. 11. Автомат-формирователь «саша-маша», состояние 010 на выходе «а» (1)

По следующему синхроимпульсу обеспечивается возврат в исходное состояние. Если установить автомат в состояние 111, то на выходе формируется буква «м» (рис. 12).

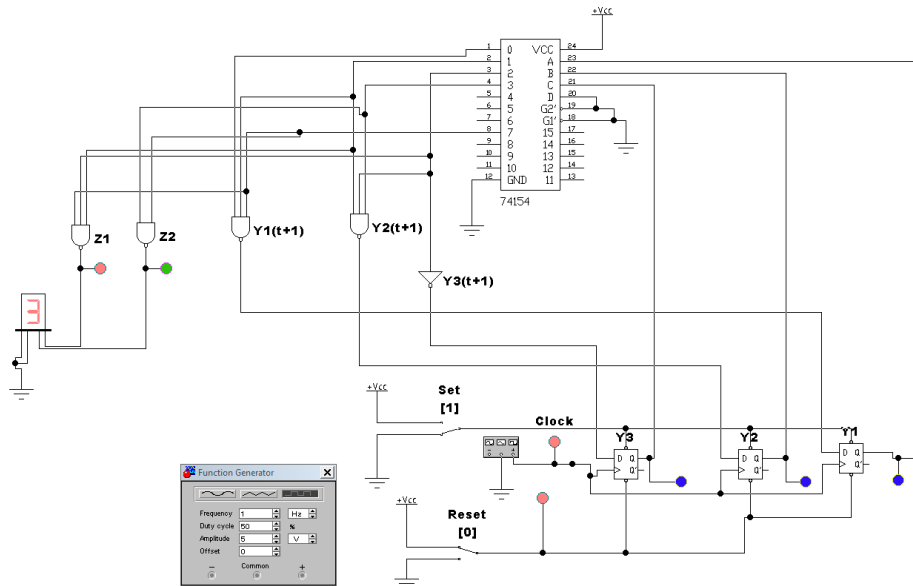


Рис. 12. Автомат-формирователь «саша-маша», на выходе «м» (3)

Далее будет формироваться подслово «аша», после чего автомат возвратится в состояние 000. Нетрудно ввести условие и для возврата в состояние 111. Для синтеза другого автомата, например «даша-саша», необходимо догадаться, что изменяется только кодировка: «м» (11) считается «д».

**Выводы.** Таким образом, синтез автомата-формирователя может быть осуществлён без минимизации логических функций в системе Electronics Workbench [10] на базе дешифратора 74154 в рамках двух-часовой лабораторной работы. Реализация синхронного автомата-формирователя позволяет задавать большое число вариантов, например, с использованием имён и фамилий студентов. В качестве исследовательского задания рекомендуется реализация в САПР Quartus II фирмы Altera (Intel) [11].

Предлагаемая методика обеспечивает масштабирование заданий и времени их выполнения (контроля), что в условиях современного дефицита времени может внести определённый вклад в дело подготовки специалистов в области электронной компонентной базы, особенно в свете задач так называемого импортозамещения.

В дальнейшем для студентов-членов СНО, магистров и аспирантов целесообразно выдать задания по самостоятельному изучению более сложных вопросов синтеза отказоустойчивых автоматов по направлениям, указанным в [12–14]. Другим актуальным направлением поддержания интереса обучаемых может быть так называемый Green IT Engineering [15–18]. Можно также одно из занятий посвятить рассмотрению диссертации Ph.D. [19], а также указать в качестве положительного примера исследования выпускника кафедры А.Н. Каменских [20], который участвовал в СНО, завершает обучение в аспирантуре и представил диссертацию к защите.

### **Библиографический список**

1. Самосинхронный вычислитель для высоконадежных применений / Ю.А. Степченков, Ю.Г. Дьяченко, Ю.В. Рождественский, Н.В. Морозов, Д.Ю. Степченков // Проблемы разработки перспективных микро- и наноэлектронных систем / под общ. ред. акад. А.Л. Стемпковского. – М.: Институт проблем проектирования в микроэлектронике РАН, 2010. – С. 418–423.

2. Библиотека самосинхронных элементов для технологии БМК / Ю.А. Степченков, Ю.Г. Дьяченко, Ф.И. Гринфельд, Н.В. Морозов, Л.П. Плеханов, А.Н. Денисов, О.П. Филимоненко, Ю.П. Фомин // Проблемы разработки перспективных микроэлектронных систем: сб. науч. тр. / под общ. ред. А.Л. Стемпковского. – М.: Институт проблем проектирования в микроэлектронике РАН, 2006. – С. 259–264.

3. Строгонов А., Цыбин С., Городков П. Новая серия ПЛИС 5578, разработанная в рамках импортозамещения зарубежной компонентной базы // Компоненты и технологии. – 2017. – Т. 2. – № 187. – С. 46–49.

4. Строгонов А., Цыбин С. Программируемая коммутация ПЛИС: взгляд изнутри [Электронный ресурс]. – URL: [http://www.kit-e.ru/articles/plis/2010\\_11\\_56.php](http://www.kit-e.ru/articles/plis/2010_11_56.php) (дата обращения: 16.02.2017).

5. Посягин А.И., Южаков А.А. Разработка двухслойной нейронной сети для самомаршрутизирующегося аналого-цифрового преобразователя на основе нейронной сети // Электротехника. – 2013. – № 11. – С. 10–13.

6. Хижняков Ю.Н., Южаков А.А. Нейрончеткий регулятор частоты газотурбинного двигателя // Приборы. – 2010. – № 5. – С. 17–21.

7. Угрюмов Е.П. Цифровая схемотехника: учеб. пособие. – 2-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2007. – 782 с.

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

9. Тюрин С.Ф., Громов О.А., Греков А.В. Реализация цифровых автоматов в системе Quartus фирмы Altera: лаборат. практикум. – Пермь: Изд-во Перм. нац. исслед. политехн. ун-та, 2011. – 133 с.

10. Сайт разработчика National Instruments [Электронный ресурс]. – URL: <http://www.ni.com/multisim/> (дата обращения: 22.03.2017).

11. Quartus II. Download Center [Электронный ресурс]. – URL: <https://www.altera.com/downloads/download-center.html> (дата обращения: 11.04.2017).

12. 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).

13. 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).

14. 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).

15. Kharchenko V., Kondratenko Y., Kasprzyk J. (Eds.) Green IT Engineering: Concepts, Models, Complex Systems Architectures, Studies in Systems, Decision and Control. – Vol. 74. – Berlin. Heidelberg: Springer International Publishing, 2017. DOI: 10.1007/978-3-319-44162-7

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

17. Хижняков Ю.Н. Нечеткое регулирование температуры теплоносителя // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2016. – № 20. – С. 5–12.

18. Южаков А.А., Щавлев В.Е. Моделирование нейросети с использованием возможностей объектно-ориентированных языков программирования // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2012. – № 6. – С. 248–256.

19. Mehta Nikil. An ultra-low-energy, variation-tolerant FPGA architecture using component-specific mapping: Dissertation (Ph.D.). – California Institute of Technology, 2013. – URL: <http://thesis.library.caltech.edu/7226/1/Nikil-Mehta-2013.pdf> (дата обращения: 09.12.2016).

20. Tyurin S., Kamenskih A. Green Logic: Models, Methods, Algorithms // Green IT Engineering: Concepts, Models, Complex Systems Architectures, Studies in Systems, Decision and Control / V. Kharchenko, Y. Kondratenko, J. Kasprzyk (Eds.). – Vol. 74. – Berlin, Heidelberg: Springer International Publishing, 2017. – P. 69–86. DOI: 10.1007/978-3-319-44162-7\_13

## References

1. Stepchenkov Iu.A., D'iachenko Iu.G., Rozhdestvenskii Iu.V., Morozov N.V., Stepchenkov D.Iu. Samosinkhronnyi vychislitel' dlia vysokonadezhnykh primenenii [Self-timed computer for high-reliability applications]. Problemy razrabotki perspektivnykh mikro- i nanoelektronnykh system. Ed. akademika A.L. Stempkovskogo. Moscow: Institut problem proektirovaniia v mikroelektronike Rossiiskoi akademii nauk, 2010, pp. 418-423.
2. Stepchenkov Iu.A., D'iachenko Iu.G., Grinfel'd F.I., Morozov N.V., Plekhanov L.P., Denisov A.N., Filimonenko O.P., Fomin Iu.P. Biblioteka samosinkhronnykh elementov dlia tekhnologii BMK [Library elements for self-timed FPGA technology]. *Problemy razrabotki perspektivnykh mikroelektronnykh sistem*. Ed. akademika A.L. Stempkovskogo. Moscow: Institut problem proektirovaniia v mikroelektronike Rossiiskoi akademii nauk, 2006, pp. 259-264.
3. Strogonov A., Tsybin S., Gorodkov P. Novaia seriia PLIS 5578, razrabotannaia v ramkakh importozameshcheniia zarubezhnoi komponentnoi bazy [New series FPGA 5578, developed in the framework of import substitution of the foreign component base]. *Komponenty i tekhnologii*, 2017, vol. 2, no. 187, pp. 46-49.
4. Strogonov A., Tsybin S. Programmiruemaia kommutatsiia PLIS: vzgliad iznutri [Programmable switching FPGA Insights], 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 16 February 2017).
5. Posiagin A.I., Iuzhakov A.A. Razrabotka dvukhsloinoi neuronnoi seti dlia samomارشrutiziruiushchegosia analogo-tsifrovogo preobrazovatel'ia na osnove neuronnoi seti [Development of a two-layer neural network for a self-routing analog-to-digital converter based on a neural network]. *Elektrotekhnika*, 2013, no. 11, pp. 10-13.
6. Khizhniakov Iu.N., Iuzhakov A.A. Neironechetkii regulator chastoty gazoturbinnogo dvigatel'ia [Neuro Fuzzy Frequency Control at the Zoturbine Engine]. *Pribory*, 2010, no. 5, pp. 17-21.
7. Ugriumov E.P. Tsifrovaia skhemotekhnika [Digital circuitry: a tutorial]. Saint Petersburg: BKhV-Peterburg, 2007. 782 p.
8. Tiurin S.F., Aliaev Iu.A. Diskretnaia matematika: prakticheskaia diskretnaia matematika i matematicheskaia logika [Discrete mathematics: practical discrete mathematics and mathematical logic]. Moscow: Finansy i statistika, 2010. 394 p.

9. Tiurin S.F., Gromov O.A., Grekov A.V. Realizatsiia tsifrovyykh avtomatov v sisteme Quartus firmy Altera: laboratornyi praktikum [The implementation of digital machines in the Quarter system of Altera: a laboratory workshop]. Permskii natsional'nyi issledovatel'skii politekhnicheskii universitet, 2011. 133 p.

10. Sait razrabotchika National Instruments [National Instruments Developer Site], available at: <http://www.ni.com/multisim/> (accessed 22 March 2017).

11. Quartus II. Download Center, available at: <https://www.altera.com/downloads/download-center.html> (accessed 11 April 2017).

12. 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).

13. 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).

14. 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).

15. Green IT Engineering: Concepts, Models, Complex Systems Architectures, Studies in Systems, Decision and Control. Eds. V. Kharchenko, Y. Kondratenko, J. Kacprzyk. Berlin, Heidelberg: Springer International Publishing, 2017, vol. 74. DOI: 10.1007/978-3-319-44162-7

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

17. Khizhniakov Iu.N. Nechetkoe regulirovanie temperatury teplonositel'ia [Fuzzy temperature control of the coolant]. *Vestnik Permskogo natsional'nogo issledovatel'skogo politekhnicheskogo universiteta. Elektrotehnika, informatsionnye tekhnologii, sistemy upravleniia*, 2016, no. 20, pp. 5-12.

18. Iuzhakov A.A., Shchavlev V.E. Modelirovanie neiroseti s ispol'zovaniem vozmozhnostei ob'ektno-orientirovannykh iazykov programmirovaniia [Simulation of a neural network using the capabilities of object-oriented programming languages]. *Vestnik Permskogo natsional'nogo issledovatel'skogo politekhnicheskogo universiteta. Elektrotehnika, informatsionnye tekhnologii, sistemy upravleniia*, 2012, no. 6, pp. 248-256.

19. Mehta Nikil. An ultra-low-energy, variation-tolerant FPGA architecture using component-specific mapping. Dissertation (Ph.D.), California Institute of Technology. 2013, available at: <http://thesis.library.caltech.edu/7226/1/Nikil-Mehta-2013.pdf> (accessed 09 December 2016).

20. Tyurin S., Kamenskih A. Green Logic: Models, Methods, Algorithms. *Green IT Engineering: Concepts, Models, Complex Systems Architectures, Studies in Systems, Decision and Control*. Eds. V. Kharchenko, Y. Kondratenko, J. Kacprzyk. Berlin, Heidelberg: Springer International Publishing, 2017, vol. 74, pp. 69-86. DOI: 10.1007/978-3-319-44162-7\_13

### Сведения об авторах

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

**Аляев Юрий Александрович** (Пермь, Россия) – кандидат технических наук, доцент, академик Академии информатизации образования РФ, профессор Российской академии естествознания, доцент кафедры математики и естественно-научных дисциплин Российской академии народного хозяйства и государственной службы при Президенте Российской Федерации (614094, г. Пермь, ул. Вильвенская, 19-84, e-mail: alyr1 @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).

**Alyayev Yuriy Aleksandrovich**, (Perm, Russian Federation) is a Ph.D. in Technical Sciences, Associate Professor, assistant professor of the pulpit mathematicians and naturally-scientific discipline Russian presidential academy of national economy and public administration (614094, Perm, 19-84, Vilvenskaja str., e-mail: alyr1 @yandex.ru).

Получено 28.04.2017