MATHEMATICAL AND COMPUTER MODELING

UDC 621.377.621.12 (075.8)

STEPAN BILAN,
MYKOLA BILAN,
SERGII BILAN

NOVEL PSEUDO-RANDOM SEQUENCE OF NUMBERS GENERATOR BASED CELLULAR AUTOMATA

This paper considers a novel pseudo-random bit sequence generator, which is implemented on a cellular automaton. It presents the hardware implementation of the generator and it the software simulation. With the help of the software model is testing of the random number generator was conducted. Tests showed a positive result, which confirms the high statistical properties of the generated random sequence.

Keywords: generator of pseudo-random sequence of numbers, cellular automata.

Introduction. Today there are many random number generators [1-19]. Realization of generators has a different nature. A large number of the developed random number generators (RNG) are used due to their wide application in different fields of human activity. Of particular need for RNG are areas such as: cryptography, protection and diagnostics data transmission systems, game theory, simulation, and many other fields. All tools and processes that use the RNG, can be divided into the use of the resulting random number sequence, both in static and in dynamic. Using a static random number sequence assumes pre-generating random numbers, and the formation of a database. Then, the generated sequence is used effectively. There is also a need for the generation and use of random numbers in real time is used effectively.

However, the existing queries are not always satisfied by existing generators. This is due to difficulties in achieving optimal values of the basic parameters of the RNG. These are the main parameters:

1. The length of the repetition period of a sequence of random numbers.
2. Low statistical properties of the generated sequence.
3. Slow performance.
4. The degree of independence of successive values of numbers.
5. Range of numbers to select.

These characteristics may be acceptable for some tasks and unacceptable for others. Of great importance in the construction of RNG has its implementation (software or hardware). Often there are cases when the proposed generator has good statistical properties, but its implementation sharply reduces performance and may require additional resource costs that does not meet the goal. Based on the described characteristics, it becomes clear that the need for the creation and development of new algorithms for generating pseudo-random sequences that combine high performance and good statistical properties of the output sequence formed, is still relevant.

Statement of the Problem. The objective of this work is to create the pseudo-random sequences generator based on a cellular automaton that has a high statistical characteristics and high speed. Through the use of cellular automata (CA), we have the task of constructing a random number generator without feedback and with long period of formation of the sequence.

Review of existing methods and tools for generating pseudo-random numbers. To date developed a huge number of RNG [1-19]. All of them are divided into physical and deterministic. Physical generators are based on various physical phenomena and processes that occur at random to an observer. These RNG are indeed random. Deterministic generators form a predictable

sequence of random numbers, which depends on the computational structure of the generator and of the initial settings. Random sequences generated by the RNG is called deterministic pseudo-random and these generators are called pseudo-random number generator (PRNG).

Physical generators implement the transform of selected parameters any analog signal to a digital value (number). In fact, such generators act as a means for converting analog signals parameters which worked out by various physical phenomena and processes.

Deterministic generator is a device that implements a user-defined sequence of operations. Typically, such a device can be realized by a circuit design or mathematical models. From the point of view of the used model for implement the deterministic PRNG, they are divided into:

- mathematics;
- hardware;
- simple;
- combined;
- linear;
- nonlinear;
- Fibonacci generators;
- Mersenne whirlwind;
- on the basis of shift registers;
- based on cellular automata, etc.

The wide interest wipes in toward the creation PRNG working in real time. They are especially important for the implementation of systems, of streaming encryption, game theory and in various learning simulators. By such a PRNG were carried out a good overview of the [1]. Lot of attention paid to the implemented generators by the shift registers with linear and nonlinear feedbacks and their combinations [11-14]. There are also a lot of other generators that require further research and development.

PRNG based CA. To construct a PRNG can also be used CA. However, generators that are based on the CA, do not give the desired results [15-19]. For example, the PRNG based on one-dimensional CA, which has developed by Stephen Wolfram [18, 19], generates quite a random sequence of numbers. However, the use of such a generator to encrypt did not give the necessary protection and the cipher can be opened when known the plaintext [20, 21].

Also known PRNGs, which are implemented on the hybrid CA (HCA) [22-26]. In such HCA uses different rules for the functioning of individual cells. This leads to the evolution of the various embodiments of the CA and forms different pseudorandom sequences. Combining rules for different cell CA gives a pseudo-random sequence of numbers. However, their invariability of functioning and at small total number of cells leads to the formation of repetitions values. Moreover, if it is known that as the of generator AC is used, the rule for each cell can be calculated. Especially if such cells are little used to implement the rules. Furthermore, based on CA, PRNG better in hardware implemented, as a software implementation is time-consuming.

There are PRNG based on CA which is implemented using an additional generator, which is implemented on a linear feedback shift register [15, 16]. They also have increased the neighborhood of each cell, which impairs its statistical properties. For such a CA would be logical to carry out reading from all cells. And in case with the various operations within the cell function will have different values.

With all of this increases the number of connections for each cell of CA, which reduces the reliability of the operation. Moreover, increasing the number of cells in the analysis involves decreasing of generator speed. And the use of an additional generator for mixing which implemented on the shift register with feedback certainly improves its properties. However, this increases the amount of feedback, which also reduces the speed. The inhomogeneity of the proposed generator is used, which requires an initial setup it toughly. This increases the number of initial settings. The problem is also the fact that the use of several CA and a shift register twice complicates the implementation of circuit. While not shown on any change of states by the law is carried out of both CA. Actually carried out modulo-2 addition of the three of bits. Two bits are

defined by the functions of an array of cells that were part of the respective cells of CA, and the third bit is a bit at the output of generator, that implemented on a linear feedback shift register. In fact, using three separate generators, which have output bits that are the arguments of the resulting function.

All the pseudo-random number generators based on the CA depend on the behavior of the CA itself and the characteristics of their organization.

Features of the organization of CA with internal local control. In all the known CA that are used to implement the PRNG are chosen neighborhood cells for each control cells. All cells change their state at each time point. Their values are depended on the local function whose arguments are the values of the outputs of the cells belonging to the neighborhood.

Initially, this approach makes the PRNG on CA vulnerability. The sequence may be predictable, if it is known that the main element is the CA. Also, if in the hands of the analyst is the sequence that generated by such PRNG. It is enough to determine the length of the one-dimensional CA and build the appropriate truth table on which to calculate the local feature.

Thus, if all cells of CA are changed state at each instant of time in accordance with a selected local function, it is possible to determine this local function. Therefore the problem arises of building CA, wherein at each moment not change their state all the cells, and only those that are excited by. These CA are used for various tasks [27-31]. However, to realize PRNG their application by authors unknown.

Such CA may be described by the following model.

$$S_{i,j}(t+1)=\begin{cases} f\left[S_{i,j}(t), A_{S_{ij}}(t)\right], & if \quad \exists q_l(t)\in Q_{S_{ij}}, \bigvee_{l=1}^{D} q_l(t)=1, q_l(t)=\{0,1\} \\ S_{i,j}(t), & if \quad \forall q_l(t)\in Q_{S_{ij}}, \bigvee_{l=1}^{D} q_l(t)=0 \end{cases}, \qquad (1)$$

where $S_{i,j}(t)$ - state of a cell at a time t, which has coordinates (i,j); $A_{S_{ij}}(t)=\{a_{i,j}(t)\}$ - a lot of signals that make up the cell neighborhoods for the cell with coordinates (i,j); D – the number of neighborhood cells for cells with coordinates (i,j); $f[\ ]$ - function performed by the cell.

Equation (1) indicates the behavior of each cell at time t. The formula (1) shows that in the medium of CA distributed the excitation signal. Each excitation cell changes its state.

However, the model (1) is valid for the CA in which the cells are excited and the excited state is stay for them before the end of the operation. With every cycle time such CA increases the number of cells that simultaneously change their state according to the local function.

To implement the CA in which only one cell changes its state each time of cycle is used more outlets in each cell. They are determined by the vector G.

$$G=\langle g_1, g_2, ..., g_d \rangle,$$

where d – number of neighborhood cells which can receive excitation signal.

The signal at the corresponding excitation output of cell is determined by the following model

$$g_{S_{ij}}^{d}(t+1)=\begin{cases} 1, & if \quad \exists\{a_{ij}(t)\}, f\left[S_{i,j}(t), A_{S_{ij}}(t)\right]=d \\ 0, & in \quad other \quad case \end{cases}, \qquad (2)$$

where $g_{S_{ij}}^{d}(t)$ - signal on the d-th excitation output of cell with coordinates (i,j).

Thus, each cell has a single information output and d excitation outputs. At the same time $d \le D$. By using CA, which is described by formulas (1) and (2) can achieve a change in the state of only one cell at each time point. Such CA can effectively implement pseudo-random bit sequence generator (PRBSG).

Generators of pseudorandom bit sequence based on the locally controlled CA

The paper proposes PRBSG that uses CA as a basic element. The structure of this PRBSG is shown in Fig. 1 [31-33].

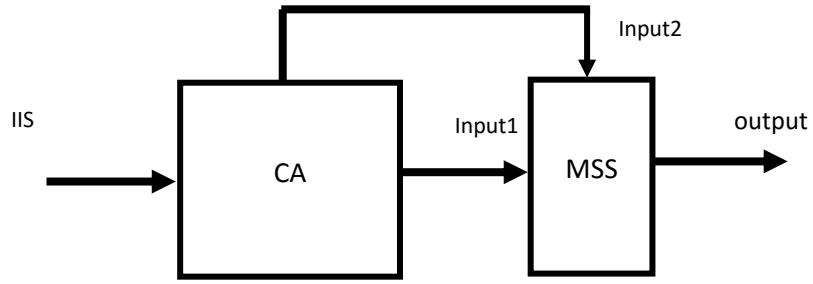

Figure 1 – The general structure of the PRBSG on the CA

The generator includes a CA and matrix switching system (MSS). The output of the MSS is generator output (output), and inputs of CA are inputs initial settings of generator (IIS).

For the initial installation of the generator the following actions are executed.

1. The structure of the CA and its geometric coverage are chosen.
2. The structure of the neighborhood cells are choosing.
3. Coverage map of CA are chosen.
4. Specifies the original coordinates of excited cells.
5. Specifies local function for each cell.
6. Sequences of cells are chosen to form a bit an additional sequence.

PRBSG on CA operated by the following algorithm.

1. The states card in CA are recorded.
2. In the first cell of CA are selected to start the spread of the excitation signal.
3. XOR operation is performed on the values of neighboring cells with the value of the own states and the value corresponding to the specified bit of the sequence.
4. The resulting bit values at the output of the generator are forming.
5. The excitation signal transmitting to one of the neighboring cells of a given local neighborhood functions.

The first initial settings item characterizes the geometrical shape of the cells (rectangular, triangular, hexagonal, etc.). From the chosen geometric shape of cell the coating on the CA is depends. Also from this form depends the neighborhood structure, which is chosen according to the second paragraph of the initial settings.

Coverage map of CA specifies cells that must install in one state, as well as cells that have zero state. Such a map can be chosen at random.

Also initial excited cell can be chosen randomly.

For the initial settings include a choice of local functions for all CA cells. For each cell two functions are chosen:

- Function which allows you to set the cell to the state, depending on the signal of state neighborhood cell (usually is the operation XOR);

- Function which indicates cells of neighborhood, which is transmitted excitation signal.

Last initial setting specifies how a additional sequence of bits is forming. The simplest option is enumeration bits which are located in the cells of lines CA. Formation occurs by reading bits cells of CA is left - right and top – down.

From these initial settings depends structure formed by a pseudo-random sequence.

Running the generator and the following operation is performed on the basis of initial settings. Reading generated bits at each time cycle is performed from the information output of a cell that is excited at this time point. With the help of MSS generated bits is output of generator. An example of operation PSBPG in three time cycle shown in Fig. 2 [32-34].

Comments in Fig. 2 allow us to determine the values of the intermediate bit, and bit of the result at each time cycle. From Fig. 2 shows how the transmission of excitation signal from cell to cell and like excited state is changing of each cell. The proposed generator has not feedback and performance is determined by execution of simple logic function and the time it takes to transmission the excitation signal from cell to cell.

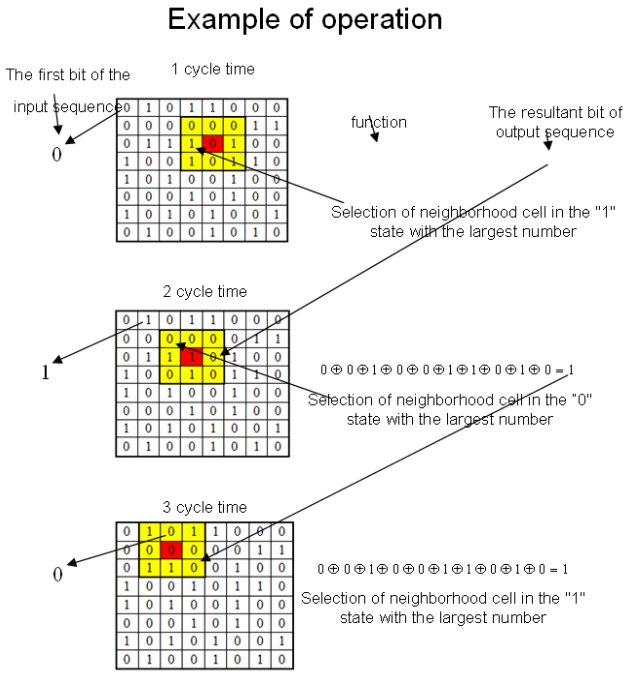

Figure 2 – An example of the functioning of the generator

Software implementation PRBSG on CA. For presentation and comfortable evaluation of the functioning of the generator developed its programming model. The program interface is shown in Fig. 3.

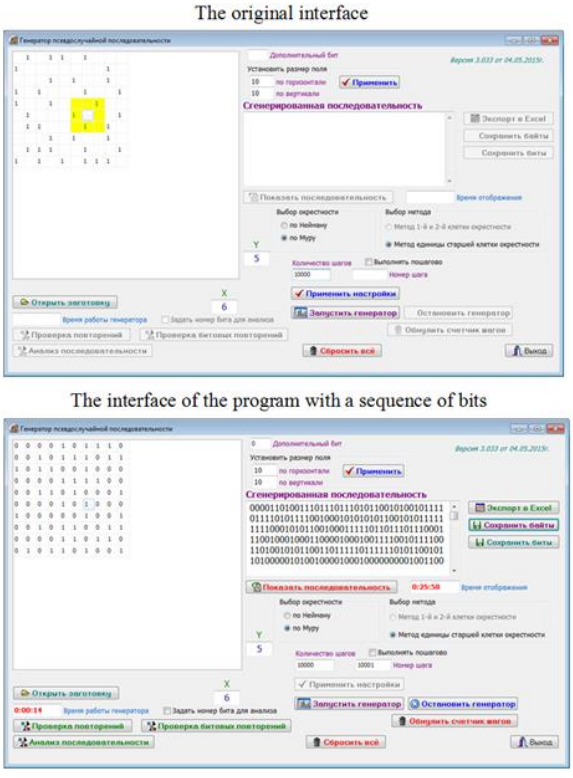

Figure 3 – The main interfaces of the program

For a better understanding of the work and research of the generator interface contains a field CA. This field is displayed in the dynamics of the work of the CA. Also, in another field shows the random bit sequence. The program allows you to save a random sequence of bits in the file. In addition, the program allows you to divide the resulting bit sequence into bytes and convert them to decimal numbers. Examples of the formed sequences and the numbers of bits shown in Fig. 4,5.

```
00001101001110111011101011001010010111011110101111001000
10101010110010101111111110001010110010001111101101110110
00111001000100011000010001001111001011110011010010101100 1
10111110111111010110010110100000101001000010001000000000 1
00110001101100011111010000011100011000101101000011111011 0
10001011011001011011100011110110100000000110001101110001 1
11000001010110001010100101100010111111010010010100111101 1
10101100100010000000110011000011100010101000111101111101 1
00010001001111100110100000101000101010110001010101110011 0
10000000110100011100111011101101000100110101101110111000 1
11110100101100100101100001100111110001110010111010001101 0
11100110010011110110001100110011111100110101001111100111 1
00101011011111110111001011001111010011111101110110101000 1
01010011110000111011010101011010000110010000100000100011 1
11100111100001000000111111101111010011111110011000001111 0
00010001000100011011000010011011000100101110000100110110 0
11010010000011010001111111111011011010001101111110001 1
01011000001011000001100111010 00
```

Figure 4 – Example random bit sequence

```
13 59 186 202 94 245 228 85 101 127 197 100 125 187 142
68 97 19 203 205 43 55 223 172 180 20 132 64 19 27 31 65
198 45 15 180 91 45 199 180 3 27 143 5 98 165 139 244 148
247 89 16 25 135 21 30 251 17 62 104 40 171 21 115 64 104
231 118 137 173 220 125 44 150 25 241 203 163 92 201 236
102 126 106 124 242 183 247 44 244 253 218 138 158 29 170
208 200 65 31 158 16 63 189 63 152 60 34 35 97 54 37 194
108 210 13 31 251 109 27 241 172 22 12 232
```

Figure 5 – An example of a random sequence of numbers and random bit sequence

The program allows you to create random bit sequence for testing.

Estimation of quality of the generated random sequences. Today proposed lot of tests to verify the pseudo-random sequences are described in detail in various literary sources [2, 3, 35]. All of them are divided into graphics and statistics tests. There is software which is arranged on sites [36-39]. These programs make it possible to assess the correct sequence of numbers placed on sites. These programs is: ENT, DIEHARD, RABENZIX, NIST etc. Each program implements a set of tests, which are detailed in the current literature. Some programs are updated, and the number of tests realized in them increases. It is believed that the larger successfully conducted test, the closer to the random sequence.

For our generator was used tests described the program ENT [36]. This program implements the following tests:

1. Calculates the entropy. The test is described in [40]. According to this test it is determined by the size of the resulting file compression. The sequence is considered to be random if the file compression does not reduce its size.
2. Chi-square Test. This test is to determine the rate of interest, which indicates the frequency of exceeding the calculated value. This gives an estimate of the percentage of random sequences [41].
3. Arithmetic Mean. Simple arithmetic test that determines the value obtained by dividing the sum of the byte length of the file. For random value should be close to 0,5.
4. Monte Carlo Value for Pi. The test determines the percentage of hits in the values of a circle inscribed in a square. Calculate the number of Pi. If this value approaches the value 3,143580574, then the sequence is determined by a random.
5. Serial Correlation Coefficient. The test determines the dependence of each byte from the previous. For random sequences, this quantity tends to 0 [41].

To test it was formed several sequences with lengths 1000, 100000, 500000 and 1000000 bit. All tests for all sequences have been successful, and pointed out that the sequence is random.

Examples of the generator and run tests are presented in Fig. 5.

Figure 5 – Examples of the programs in the generation and testing of random sequences

To assess the quality of the proposed generator partially packet of tests was used NIST800-22 [35]. Were used and the following tests were performed.

1. The Frequency (Monobit) Test.
2. Frequency Test within a Blok.
3. The Runs Test.
4. Test for the Longest-Run-of-Ones in a Blok.
5. The Binary Matrix Rank Test.
6. Discrete Fourier Transform (Spectral) Test.
7. The Non-overlapping Template Matching Test.
8. Overlapping Template Matching Test.
9. Maurer's "Universal Statistical" Test.
10. Linear Complexity Test.

All of the tests were implemented using individual programs that have shown successful results.

Conclusion. This paper presents a new pseudo-random number sequences generator, based on CA. The generator is different from the known generators, which are based on CA. It has better statistical characteristics that prove that the tests and has a high speed. The proposed generator is no feedback, which increases the reliability of functioning. Using an additional sequence which is formed by cells of the CA improves the statistical characteristics of the pseudo-random number sequence.

Further research. The authors have developed several modifications of generators based on such CA. We plan to increase the number of well-known tests for the qualitative analysis of the proposed generators.

REFERENCES

1. Shnayer, B. (2003), *Applied cryptography. Protocols , algorithms and source code in C* [*Prikladnaya kryptografiya. Protokoly, algorytmy I ishodnye teksty na yazike C*], Delo, Moskow, 24 p.
2. Chugunkov, I. V. (2012), *Methods and tools of evaulation the quality of pseudo-random sequence generator, aimed at solving problems of information security* [*Metody I sredstva ocenky kachestva generatorov psevdosluchaynih posledovatelynostey, orientirovannih na reshenye zadach zasyty informacyy : uchebnoye posobye*], NYAU, Moskow, 236 p.
3. Ivanov, M.A., Chugunkov, I.V. (2003), *The theory , application and evaluation of the quality of pseudo-random sequence generator* [*Teoriya, prymenenye i ocenka kachestva generatorov psevdosluchaynih posledovatelynostey*]. CUDYC-OBRAZ, Moskow, 240 p.
4. Von Neumann, J. (1951), *Various techniques used in connection with random digits*, Applied Mathematics Series, Vol. 12, pp. 36-38.
5. L'Ecuyer, P. (1994), *Uniform random number generation*, Annals of Operations Research, Vol. 53, pp. 77-120.
6. Haryn, U. S., Bernyk, V. I., Matveev, G. V. (1999), *Mathematical foundations of cryptology : tutorial* [*Matematycheskye osnovy kryptologyy : ucheb. posobye*]. BGU, Minsk, 319 p.
7. Lehmer, D. (1951), *Mathematical methods in large-scale computing units*, Large-Scale Digital Calculating Machinery : symp. proc, Harvard, pp. 141-146.
8. Thomson, W. (1958), *A modified congruence method of generating pseudo-random numbers*, Computer Journal, Vol. 1, pp. 83-86.
9. Hammer, P. (1951), *The mid-square method of generating digits*, Monte Carlo Method : symp. proc (Los Angeles, 1949), Washington, Vol. 12, pp. 33.
10. Marsaglia, G. (2003), *Random number generators*, Journal of Modern Applied Statistical Methods, Vol. 2, pp. 2-13.
11. Eichenauer, J., Lehn, J., Topuzoglu, A. (1988), *A nonlinear congruential pseudorandom number generator with power of two modulus*, Mathematics of Computation, Vol. 51, pp. 757-759.
12. Lewis, T., Payne, W. (1973), *Generalized feedback shift register pseudorandom number algorithms*, Journal of ACM, Vol. 21, pp. 456-468.
13. Matsumoto, M., Nishimura, T. (1998), *Mersenne twister : a 623-dimensionally equidistributed uniform pseudo-random number generator*, ACM Transactions on Modeling and Computer Simulation, Vol. 8, pp. 3-30.
14. Hell, M., Johansson, T., Meier, W. (2005), *Grain – A stream cipher for constrained environments*, The eSTREAM Project, available at : http://www.ecrypt.eu.org/stream/p3ciphers/grain/Grain_p3.pdf (accessed 23 March 2015).
15. Suhinin, B.M. (2010), *High-speed generators of pseudorandom sequences based on cellular automata* [*Visokoskorostnye generatory psevdosluchaynih posledovatelynostey na osnove kletochnyh avtomatov*], Prykladnaya dyskretnaya matematika, No. 2, pp. 34-41.
16. Suhinin, B.M. (2010), *Development of generators of pseudorandom binary sequences based on cellular automata* [*Razrabotka generatorov psevdosluchaynyh dvoichnyh posledovatelynostey na osnove kletochnyh avtomatov*], Nauka i obrazovanye, No. 9, pp. 1-21.
17. Wolfram, S. (1983), *Cellular automata*, Los Alamos Science, Vol. 9, pp. 2-21.
18. Wolfram, S. (1986), *Cryptography with cellular automata*, Lecture Notes in Computer Science, Vol. 218, pp. 429-432.
19. Wolfram, S. (1986), *Random sequence generation by cellular automata*, Advances in Applied Mathematics, Vol. 7, pp. 123-169.
20. Meier, W., Staffelbach, O. (1991), *Analysis of Pseudo Random Sequences by Cellular Automata*, Advances in Cryptology EUROCRYPT'91 Proceedings, Springer – Verlag, pp. 186-199.
21. Bartee, T.C., Schneider,D. L. (1963), *Computation Finite Fields*, Information and Control, Vol. 6, No. 2. pp. 79-88.

22. Fraile Ruboi, C., Hernandez Encinas, L., Hoya White, S., Martin del Rey and Rodrigues Sancher, A. (2004), *The use of Linear Hybrid Cellular Automata as Pseudorandom bit Generators in Cryptography*, Neural Parallel & Scientific Comp, No. 12 (2), pp. 175-192.

23. Martin, B., Sole, P. (2008), *Pseudo-random Sequences Generated by Cellular Automata*, International Conference on Ralations, Orders and Graphs: Interaction with Computer Scince, May 2008, Mandia, Tunisia, Nouha editions, pp. 401-410.

24. Wolfram, S. (1986), *Theory and applications of cellular automata*, World Scientific Publishing Co. Ltd., pp. 485-557.

25. Cattell, K., Muzio, J. (1996), *Synthesis of one-dimensional linear hybrid cellular automata*, IEEE Trans. On Computer-aided design of integrated circuits and systems, No. 15(3), pp. 325-335.

26. Cho, S. J., Choi, U. S., Kim, H. D., Hwang, Y. H., Kim, J. G., Heo, S. H. (2007), *New syntheesis of one-dimensional 90/150 liner hybrid group CA*, IEEE Transactions on comput-aided design of integrated circuits and systems, No. 25(9), pp. 1720-1724.

27. Belan, S. N. (2011), *Specialized cellular structures for image contour analysis*, Cybernetics and Systems Analys, September, Vol. 47, Iss. 5, pp 695-704.

28. Belan, S., Belan, N. (2012), *Use of Cellular Automata to Create an Artificial System of Image Classification and Recognition*, ACRI2012, LNCS 7495, Springer-Verlag Berlin Heidelberg, pp. 483-493.

29. Belan, S., Belan, N. (2013), *Temporal-Impulse Description of Complex Image Based on Cellular Automata*, PaCT2013, LNCS. Vol. 7979, Springer-Verlag Berlin Heidelberg, pp. 291-295.

30. Bilan, S. (2014), *Models and hardware implementation of methods of Pre-processing Images based on the Cellular Automata*, Advances in Image and Video Processing, Vol. 2, No. 5, pp. 76-90.

31. Bilan, S., Bilan, M., Bilan, S. (2015), *Application of Methods of Organization of Cellular Automata to Implement Devices of Forming Pseudorandom Sequences*, Information Technology & Computer Science Abstracts 11th Annual International Conference on Information Technology & Computer Science, 18-21 May 2015, Athens, Greece Edited by Gregory T. Papanikos, pp. 17-18.

32. Bilan, S. M., Bilan, M. M., Bilan, A. M., Bilan, S. S. (2014), *Generator pseudorandom bit sequence based on cellular automata* [*Generator psevdovypadkovih bitovyh poslidovnostey na osnovi klitynnyh avtomativ*], Patent of Ukraine na korysnu model, Bul. № 18 vid 25.09.14.

33. Bilan, S. M. (2015), *Information security in telecommunication systems : a manual for students of higher educational institutions* [*Zahist informacii v telekomunikaciynyh systemah: navch. posyb. dlya stud. vyshih navch. zakl. zalyzn. Transp*], DETUT, Kiev, 132 p.

34. Bilan, S. M., Bilan, M. M. (2014), *A computer program to create sequences of pseudorandom numbers and pseudorandom bit sequence based on cellular automata «pseudorandom sequence generator»* [*Komputerna programma dlya formuvannya psevdovipadkovih poslidovnostey chisel ta psevdovipadkovih bitovih poslidovnostey na osnovy klitinnyh avtomativ «Generator psevdosluchaynoy posledovatelynosty»*], Svidoctvo pro reyestraciu avtorsykogo prava na tvir № 54179 vid 20.03.2014.

35. National Institute of Standards and Technology (2001), NIST SP 800-38A. *Recommendation for block cipher modes of operation*, 66 p.

36. Walker, J. (2008), *A Pseudorandom Number Sequence Test Program*, available at : http://www.fourmilab.ch/random/ (accessed 03 February 2015).

37. National institute of standarts and technology, *Computer security division. Computer security resource center*, available at : http://csrc.nist.gov/groups/ST/toolkit/rng/documentation_software.html (accessed 20 February 2015).

38. Marsaglia, G., *The Marsaglia Random Number CDROM including the Diehard Battery of Tests of Randomness*, Department of statistics and supercomputer computations and research institute, available at : http://www.stat.fsu.edu/pub/diehard/ (accessed 15 March 2015).

39. Gerard van der Galiën, J. (2006), *RABENZIX Randomness Test Suite. 5.4*, available at : http://members.tele2.nl/galien8/rabenzix/rabenzix.html (accessed 15 March 2015).

40. Hamming, R. W. (1980), *Coding and Information Theory*, Englewood Cliffs NJ : Prentice-Hall, 60 p.
41. Knuth, D. E. (1969), *The Art of Computer Programming, Vol. 2, Seminumerical Algorithms*, Reading MA : Addison-Wesley, 784 p.

The article was received 29.03.2015.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Шнайер Б. Прикладная криптография. Протоколы, алгоритмы и исходные тексты на языке С: 2-е издание / Б. Шнайер. – М. : Дело, 2003. – 524 с.
2. Чугунков И. В. Методы и средства оценки качества генераторов псевдослучайных последовательностей, ориентированных на решение задач защиты информации : учебное пособие / И. В. Чугунков. М.: НИЯУ МИФИ, 2012. – 236 с.
3. Иванов М. А. Теория, применение и оценка качества генераторов псевдослучайных последовательностей / М. А. Иванов, И. В. Чугунков. – М. : КУДИЦ-ОБРАЗ, 2003. – 240 с.
4. Von Neumann J. Various techniques used in connection with random digits / J. von Neumann // Applied Mathematics Series. – 1951. – Vol. 12. – P. 36-38.
5. L'Ecuyer P. Uniform random number generation / P. L'Ecuyer // Annals of Operations Research. – 1994. – Vol. 53. – P. 77-120.
6. Харин Ю. С. Математические основы криптологии : учеб. пособие / Харин Ю. С., Берник В. И., Матвеев Г. В. – Минск : БГУ, 1999. – 319 с.
7. Lehmer D. Mathematical methods in large-scale computing units / D. Lehmer // Large-Scale Digital Calculating Machinery : Symp. proc. – Harvard, 1951. – P. 141-146.
8. Thomson W. A modified congruence method of generating pseudo-random numbers / W. Thomson // Computer Journal. – 1958. – Vol. 1. – P. 83-86.
9. Hammer P. The mid-square method of generating digits / P. Hammer // Monte Carlo Method : symp. proc (Los Angeles, 1949). – 1951. – Vol. 12. – P. 33.
10. Marsaglia G. Random number generators / G. Marsaglia // Journal of Modern Applied Statistical Methods. – 2003. – Vol. 2. – P. 2-13.
11. Eichenauer J. A nonlinear congruential pseudorandom number generator with power of two modulus / J. Eichenauer, J. Lehn, A. Topuzoglu // Mathematics of Computation. – 1988. – Vol. 51. – P. 757-759.
12. Lewis T. Generalized feedback shift register pseudorandom number algorithms / T. Lewis, W. Payne // Journal of ACM. – 1973. – Vol. 21. – P. 456-468.
13. Matsumoto M. Mersenne twister : A 623-dimensionally equidistributed uniform pseudo-random number generator / M. Matsumoto, T. Nishimura // ACM Transactions on Modeling and Computer Simulation. – 1998. – Vol. 8. – P. 3-30.
14. Hell M. Grain – A stream cipher for constrained environments [Electronic resource] / M. Hell, T. Johansson, W. Meier // The eSTREAM Project. 2005. 14 p. – Access mode : http://www.ecrypt.eu.org/stream/p3ciphers/grain/Grain_p3.pdf. – The title of the screen.
15. Сухинин Б. М. Высокоскоростные генераторы псевдослучайных последовательностей на основе клеточных автоматов / Б.М. Сухинин // Прикладная дискретная математика. – 2010. – № 2. – С. 34 – 41.
16. Сухинин Б. М. Разработка генераторов псевдослучайных двоичных последовательностей на основе клеточных автоматов / Б. М. Сухинин // Наука и образование. – 2010. – № 9. – С. 1-21.
17. Wolfram S. Cellular automata / S. Wolfram // Los Alamos Science. – 1983. – Vol. 9. – P. 2-21.
18. Wolfram S. Cryptography with cellular automata / S. Wolfram // Lecture Notes in Computer Science. – 1986. – Vol. 218. – P. 429-432.
19. Wolfram S. Random sequence generation by cellular automata / S. Wolfram // Advances in Applied Mathematics. – 1986. – Vol. 7. – P. 123-169.

20. Meier W. Analysis of Pseudo Random Sequences by Cellular Automata / W. Meier, O. Staffelbach // Advances in Cryptology EUROCRYPT'91 Proceedings, Springer – Verlag. – 1991. – pp. 186-199.

21. Bartee T. C. Computation Finite Fields / T. C. Bartee, D. L. Schneider // Information and Control. – 1963. – Vol. 6, No. 2. – P. 79-88.

22. Fraile Ruboi C. The use of Linear Hybrid Cellular Automata as Pseudorandom bit Generators in Cryptography / C. Fraile Ruboi, L. Hernandez Encinas, S. Hoya White, A. Martin del Rey, R. Sancher // Neural Parallel & Scientific Comp. – 2004. – No. 12 (2). – P. 175-192.

23. Martin B. Pseudo-random Sequences Generated by Cellular Automata / B. Martin, P. Sole // International Conference on Ralations, Orders and Graphs: Interaction with Computer Scince, May 2008, Mandia, Tunisia, Nouha editions. – P. 401-410.

24. Wolfram S. Theory and applications of cellular automata / S. Wolfram // World Scientific Publishing Co. Ltd. – 1986. – P. 485-557.

25. Cattell K. Synthesis of one-dimensional linear hybrid cellular automata / K. Cattell, J. Muzio // IEEE Trans. On Computer-aided design of integrated circuits and systems. – 1996. – No. 15 (3). – P. 325-335.

26. Cho S. J. New syntheesis of one-dimensional 90/150 liner hybrid group CA / S. J. Cho, U. S. Choi, H. D. Kim, Y. H. Hwang, J. G. Kim, S. H. Heo // IEEE Transactions on comput-aided design of integrated circuits and systems. – 2007. – No. 25 (9). – P. 1720-1724.

27. Белан С.Н. Специализированные клеточные структуры для контурного анализа изображений / С. Н. Белан // Кибернетика и системный анализ. – 2011. – № 5. – С. 33-44.

28. Belan S. Use of Cellular Automata to Create an Artificial System of Image Classification and Recognition / S. Belan, N. Belan // ACRI2012, LNCS 7495, Springer-Verlag, Berlin, Heidelberg. – 2012. – P. 483-493.

29. Belan S. Temporal-Impulse Description of Complex Image Based on Cellular Automata / S. Belan, N. Belan // PaCT2013, LNCS. – 2013. – Vol. 7979. – P. 291-295.

30. Bilan S. Models and hardware implementation of methods of Pre-processing Images based on the Cellular Automata / Stepan Bilan // Advances in Image and Video Processing. – 2014. – Vol. 2, No. 5. – P. 76-90.

31. Bilan S. Application of Methods of Organization of Cellular Automata to Implement Devices of Forming Pseudorandom Sequences / S. Bilan, M. Bilan, S. Bilan // Information Technology & Computer Science Abstracts 11th Annual International Conference on Information Technology & Computer Science, 18-21 May 2015, Athens, Greece Edited by Gregory T. Papanikos – P. 17-18.

32. Білан С. М. Генератор псевдовипадкових бітових послідовностей на основі клітинних автоматів» / С. М. Білан, М. М. Білан, А. М. Білан, С. С. Білан // Патент України на корисну модель № 93427, Бюл. № 18 від 25.09.14 р.

33. Білан С. М. Захист інформації в телекомунікаційних системах : навч. посіб. для студ. вищ. навч. закладів залізничного транспорту / С. М. Білан. – К.: ДЕТУТ, 2015. – 132 с.

34. Білан С. М. Комп'ютерна програма для формування псевдовипадкових послідовностей чисел та псевдовипадкових бітових послідовностей на основі клітинних автоматів «Генератор псевдослучайной последовательности» / С. М. Білан, М. М. Білан // Свідоцтво про реєстрацію авторського права на твір № 54179 від 20.03.2014 р.

35. Recommendation for block cipher modes of operation : NIST SP 800-38A [Electronic resource]. – Access mode : http://csrc.nist.gov/publications/nistpubs/800-38a/sp800-38a.pdf. – The title of the screen.

36. Walker J. A Pseudorandom Number Sequence Test Program [Electronic resource] / J. Walker. – Access mode : http://www.fourmilab.ch/random/. – The title of the screen.

37. National institute of standarts and technology [Electronic resource] / Computer security division // Computer security resource center. – Access mode : http://csrc.nist.gov/groups/ST/toolkit/rng/documentation_software.html. – The title of the screen.

38. Marsaglia G. The Marsaglia Random Number CDROM including the Diehard Battery of Tests of Randomness [Electronic resource] / George Marsaglia // Department of statistics and supercomputer computations and research institute. – Access mode : http://www.stat.fsu.edu/pub/diehard/. – The title of the screen.

39. Gerard van der Galiën J. RABENZIX Randomness Test Suite. 5.4 [Electronic resource] / J. Gerard van der Galiën. – Access mode : http://members.tele2.nl/galien8/rabenzix/rabenzix.html. – The title of the screen.

40. Hamming R. W. Coding and Information Theory / R. W. Hamming. – Englewood Cliffs NJ : Prentice-Hall, 1980. – 60 p.

41. Knuth D. E. The Art of Computer Programming. Volume 2. Seminumerical Algorithms / D. E. Knuth // Reading MA: Addison-Wesley, 1969. – 784 p.

СТЕПАН БІЛАН,
НИКОЛАЙ БІЛАН,
СЕРГЕЙ БІЛАН

НОВЫЙ ГЕНЕРАТОР ПСЕВДОСЛУЧАЙНЫХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ ЧИСЕЛ НА ОСНОВЕ КЛЕТОЧНОГО АВТОМАТА

Рассматривается новый генератор псевдослучайных последовательностей бит, который реализован на клеточном автомате. Представлена аппаратная реализация генератора и выполнено его программное моделирование. С помощью программной модели проведено тестирование разработанного генератора псевдослучайных чисел. Использованные тесты показали положительный результат, который подтверждает высокие статистические свойства сформированной случайной последовательности.

Ключевые слова: генератор псевдослучайной последовательности чисел, клеточный автомат.

СТЕПАН БІЛАН,
МИКОЛА БІЛАН,
СЕРГІЙ БІЛАН

НОВИЙ ГЕНЕРАТОР ПСЕВДОВИПАДКОВИХ ПОСЛІДОВНОСТЕЙ ЧИСЕЛ НА ОСНОВІ КЛІТИННОГО АВТОМАТА

Розглядається новий генератор псевдовипадкових послідовностей біт, який реалізований на клітинному автоматі. Представлена апаратна реалізація генератора і виконано його програмне моделювання. За допомогою програмної моделі проведене тестування розробленого генератора псевдовипадкових чисел. Використані тести показали позитивний результат, який підтверджує високі статистичні властивості сформованої випадкової послідовності.

Ключевые слова: генератор псевдовипадкової послідовності чисел, клітинний автомат.

Stepan Bilan, candidate of technical sciences, professor of cybersecurity and application of information systems and technologies department, Institute of special communications and information security National technical university of Ukraine « Kyiv polytechnic institute», Kyiv, Ukraine.
E-mail: bstepan@ukr.net.

Mykola Bilan, IT Teacher of the municipal educational institution Mayakskaya Secondary School of Mayak town, Republic of Moldova.
E-mail: nickni@mail.ru.

Sergii Bilan, Technical Lead, Win-Interactive LLC, Vinnytsia, Ukraine.
E-mail: belan@svitonline.com.

Степан Миколайович Білан, кандидат технічних наук, доцент, професор кафедри кібербезпеки та застосування автоматизованих інформаційних систем та технологій, Інститут спеціального зв'язку та захисту інформації Національного технічного університету України «Київський політехнічний інститут», Київ, Україна.

Миколай Миколайович Білан, вчитель інформатики МЗОУ Маякська ЗОШ, село Маяк, Молдова.

Сергій Степанович Білан, технічний керівник, ТОВ ВІН-ІНТЕРАКТИВ, Вінниця, Україна.

Степан Николаевич Белан, кандидат технических наук, профессор кафедри кибербезопасности и использования автоматизированных информационных систем и технологий», Институт специальной связи и защиты информации НТУУ «КПИ», Киев, Украина.

Николай Николаевич Белан, учитель информатики МОУ Маякская ОСШ, поселок Маяк, Молдова.

Сергей Степанович Белан, технический руководитель, ООО ВИН-ИНТЕРАКТИВ, Винница, Украина.

УДК 681.326.7

ВАСИЛИЙ КУЛИКОВ,
ВИТАЛИЙ КРАВЧУК

МЕТОД МОДЕЛИРОВАНИЯ ЦИФРОВЫХ СХЕМ С НЕИСПРАВНОСТЯМИ

Рассматривается метод моделирования цифровых схем применительно к решению задачи поиска неисправности, не обнаруживаемой по реакции схемы на заданную входную последовательность сигналов (проверки полноты теста). Метод позволяет строить модели цифровых схем, обладающие лучшими показателями скорости моделирования по сравнению с известными методами. Высокая скорость моделирования достигается за счёт сведения процесса моделирования к операциям поразрядного логического умножения и сложения рабочих полей, в которых содержится вся необходимая информация о сигналах и неисправностях схемы.

Ключевые слова: метод моделирования, цифровые схемы, неисправность.

Постановка задачи. Рассматриваемый метод является интерпретативным, событийным, дедуктивным методом Δ-моделирования от входов к выходам в троичном алфавите для цифровых схем, представленных на вентильном уровне, с неявным учетом задержек элементов, с параллельным моделированием схемы во всех рассматриваемых состояниях. Метод является дальнейшим развитием идей дедуктивного и параллельного моделирования и обладает лучшими их свойствами. Описание метода сопровождается примером применения его для моделирования простой комбинационной схемы, известной как схема C17 ISCAS'85 (см. рис.1) [1-4].

Идея метода предложена В.А. Ермиловым.