







## Article

# Development of Additive Fibonacci Generators with Improved Characteristics for Cybersecurity Needs

Volodymyr Maksymovych <sup>1</sup>, Mariia Shabatura <sup>1</sup>, Oleh Harasymchuk <sup>2</sup>, Mikolaj Karpinski <sup>3,\*</sup>,  
Daniel Jancarczyk <sup>3</sup> and Pawel Sawicki <sup>4</sup>

- <sup>1</sup> Department of Information Technology Security, Institute of Computer Technologies, Automation and Metrology, Lviv Polytechnic National University, 79013 Lviv, Ukraine; volodymyr.m.maksymovych@lpnu.ua (V.M.); mariia.m.mandrona@lpnu.ua (M.S.)
- <sup>2</sup> Department of Information Protection, Institute of Computer Technologies, Automation and Metrology, Lviv Polytechnic National University, 79013 Lviv, Ukraine; oled.i.harasymchuk@lpnu.ua
- <sup>3</sup> Department of Computer Science and Automatics, Faculty of Mechanical Engineering and Computer Science, University of Bielsko-Biala, 43-309 Bielsko-Biala, Poland; djancarczyk@ath.bielsko.pl
- <sup>4</sup> SunsetPicnic UG, 10437 Berlin, Germany; mail@psawicki.de
- \* Correspondence: mkarpinski@ath.bielsko.pl

**Abstract:** Pseudorandom sequence generation is used in many industries, including cryptographic information security devices, measurement technology, and communication systems. The purpose of the present work is to research additive Fibonacci generators (AFG) and modified AFG (MAFG) with modules  $p$  prime numbers, designed primarily for their hardware implementation. The known AFG and MAFG, as with any cryptographic generators of pseudorandom sequences, are used in arguments with tremendous values. At the same time, there are specific difficulties in defining of their statistical characteristics. In this regard, the following research methodologies were used in work: for each variant of AFG and MAFG, two models were created—abstract, which is not directly related to the circuit solution, and hardware, which corresponds to the proposed structure; for relatively small values of arguments, the identity of models was proved; the research of statistical characteristics, with large values of arguments, was carried out using an abstract model and static tests NIST. Proven identity of hardware and abstract models suggest that the principles laid down in the organization of AFG and MAFG structures with modules of prime numbers ensure their effective hardware implementation in compliance with all requirements for their statistical characteristics and the possibility of application in cryptographic information security devices.

**Keywords:** pseudorandom sequences; additive Fibonacci generator; statistical characteristics; cybersecurity; information security



**Citation:** Maksymovych, V.; Shabatura, M.; Harasymchuk, O.; Karpinski, M.; Jancarczyk, D.; Sawicki, P. Development of Additive Fibonacci Generators with Improved Characteristics for Cybersecurity Needs. *Appl. Sci.* **2022**, *12*, 1519. <https://doi.org/10.3390/app12031519>

Academic Editors: Gianluca Lax and Agostino Forestiero

Received: 28 December 2021

Accepted: 27 January 2022

Published: 30 January 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Additive Fibonacci generators (AFG) are one of the types of pseudorandom sequence generators that are widely used in many technical means, particularly in cryptographic means of information protection. In their traditional design, they do not provide adequate cryptosecurity, but can be used as part of cryptographic devices [1–9]. Recently, we proposed a modified AFG (MAFG), in which the introduction to their structure and additional logic circuit, allowed us to include, in the process of arithmetic addition, the result of a logical function from the binary values of the resulting register, which significantly improved the statistical characteristics of the pseudorandom output sequence [10–15].

At present, almost all classic AFGs and new MAFGs, designed for hardware implementation, operate according to recurrent equations with modules whose values are equal to the power of two. It simplifies their hardware implementation but narrows their functionality and worsens the statistical characteristics of the output pseudorandom sequences.

In [16], we proposed AFGs that can work with an arbitrary value of a module, including a module whose value is a prime number. However, these devices do not have an additional logic circuit [10–15], which does not allow satisfactory statistical characteristics to be obtained without the involvement of additional devices.

In this article, we reveal the approach to constructing Fibonacci additive generators with modules of prime numbers. This construction method expands the capabilities of the hardware implementation of such generators and improves their output statistical characteristics, which allows them to be used effectively in cryptographic applications. A research methodology is proposed based on using abstract and hardware models of generators. Their identity is proved, which allows investigation of the statistical characteristics of such generators with the large values of arguments, which is especially important for cryptographic generators. The research results indicate that the proposed models and structures of generators can be effectively used to solve cryptographic problems of information security.

The aim of the work is to create and research the characteristics of AFGs and MAFGs with modules whose values are prime numbers. To achieve this goal, new generator structures are proposed, in which the introduction of additional structural elements allows us to ensure the operation of generators with arbitrary values of the recurrent equation modulus. This is the scientific novelty of the obtained results, which significantly improves the statistical characteristics of generators, expands their functionality, and expands the scope of their use in cryptographic means of information protection, particularly in streaming ciphers.

## 2. Related Works

A large number of works are devoted to the construction of AGF. In particular, analyses of the implementation of Fibonacci hardware generators on FPGA are given in [17]. There are also similar studies of Fibonacci generator implementations on FPGA in [18], and in [19], true random number generators, based on Fibonacci–Galois ring oscillators for FPGA, are considered, and the possibility of using these generators in cryptographic applications is shown. The results of research that used a combination of a hybrid of two existing generators—a linear congruential method and a delayed Fibonacci technique—are presented in [20]. The analysis of the efficiency of using a Fibonacci generator for cryptographic problems is also considered in [21,22]. Moreover, in [23], Fibonacci generators are used for the key generation algorithm with the necessary randomness and low algorithmic complexity. The work in [24] is devoted to the question of the correct choice of Fibonacci generator parameters.

AFGs operate according to the following generalized recurrent equation:

$$x_i = (x_{i-a} + x_{i-b} + \dots + x_{i-q}) \bmod (m), \quad (1)$$

where  $a > b > \dots > q > 0$ .

Usually, AFGs are used in which the module  $m = 2^n$ , where  $n$  is the number of generator structural elements binary bits, that simplifies their hardware implementation. Under certain conditions, the repetition period of such AFGs is not less than value  $2^n - 1$  [25].

It is known [26] that, if the module  $m = p$  is a prime number, then, according to the theory of finite fields, we can find such multipliers as  $a_1, a_2, \dots, a_k$ , so that the sequence can be defined by the following equation:

$$x_i = (a_i x_{i-1} + \dots + a_{k-1} x_{i-k+1} + a_k) \bmod p, \quad (2)$$

which will have the maximum possible period equal to  $p^k - 1$ . In this case, the following theorem holds. If the constants  $a_1, a_2, \dots, a_k$  are such that the polynomial  $x^k - a_1 x^{k-1} - \dots - a_k$  is primitive over the field  $GF(p)$ , and at least one of the elements  $x_0, x_1, \dots, x_k$  is not zero, then the generator period is equal to  $p^k - 1$ , at any initial values of the structural elements of the generator.

It is also known [26,27] that the search for primitive polynomials for prime number modules is a difficult task.

In [10–15] we proposed modified MAFGs, in which the module is determined by the equation  $m = 2^n$ , but they include an additional logic circuit (LC), the function of which is logical addition of the module 2 of the bits values of one of the generator registers, and then the result is added to the main operation of the arithmetic addition. This allows a significant increase in the repetition periods of the output sequences and an improvement of their statistical characteristics.

However, for today, there are no reasonable developments in which the structures of AFGs and MAFGs are proposed with an arbitrary value of the module of the recurrent equation.

### 3. Structure Scheme and Work Principle of AFG and MAFG with Arbitrary Value of the Module of the Recurrent Equation

Figure 1 shows the structure scheme of AFG and MAFG, which can operate with any value of the recurrent equation module. The AFG consists of registers RG1–RG6, adders AD1 and AD2, multiplexer MUX, and logical element OR. The logic circuit LC is additionally introduced to the MAFG structure.

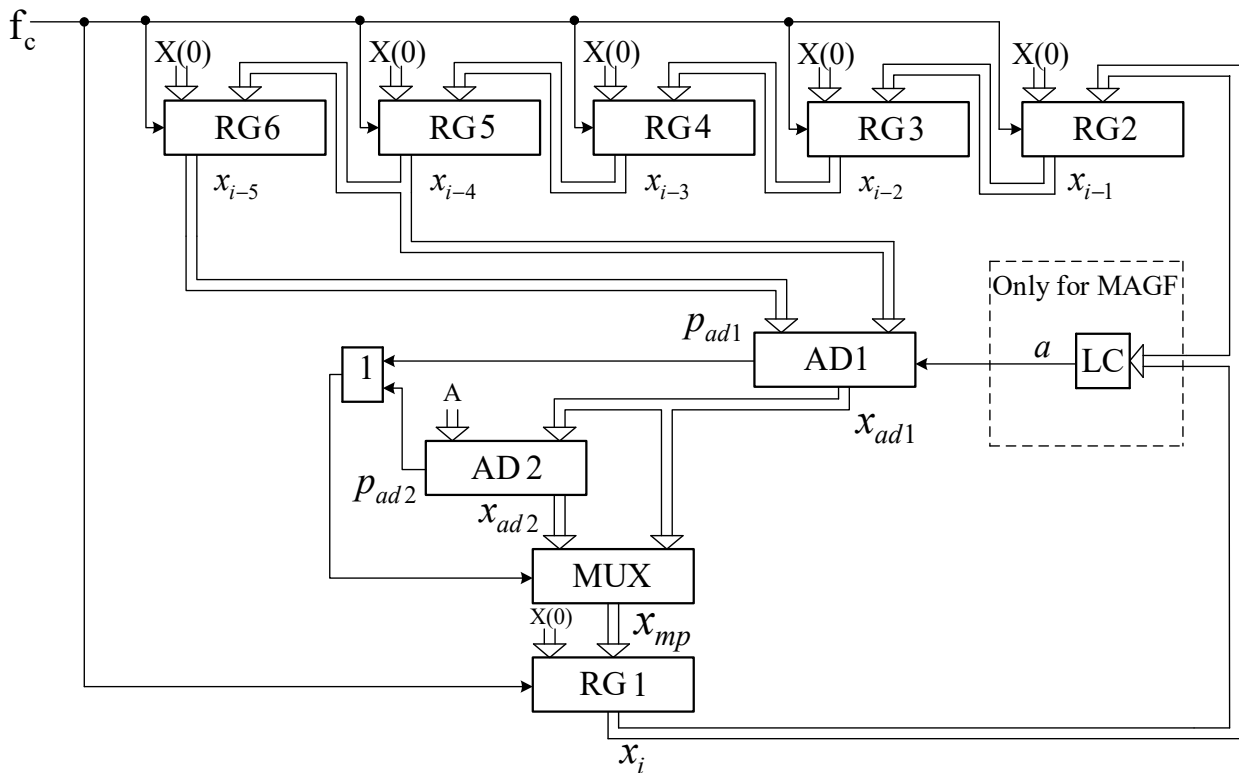


Figure 1. Structure scheme of AFG and MAFG.

The schemes are given for generators operating according to the following equations:

$$x_i = (x_{i-5} + x_{i-4}) \bmod p, \text{ (for AFG)} \tag{3}$$

$$x_i = (x_{i-5} + x_{i-4} + a) \bmod p, \text{ (for MAFG)} \tag{4}$$

where:  $x_i, x_{i-4}, x_{i-5}$ —numbers at the outputs of registers RG1, RG5, and RG6, respectively.

In Equation (4):

$$a = b_0 \oplus b_1 \oplus \dots \oplus b_s, \tag{5}$$

where:  $b_i (i = 0, 1, \dots, s; s \leq n)$ —values of the number  $x_i$  binary bits;  $n$ —the total number of binary bits.

With each clock pulse, new values of numbers are formed in the registers RG1–RG6, in particular in the register RG1—the number determined by the output signal of the multiplexer MUX.

At the output of the logic circuit LC, the signal  $a$  is formed in accordance with logic Equation (5). Adding the LC output signal  $a$ , in the process of arithmetic addition, implemented by the adder AD1, can significantly improve the statistical characteristics of the output pseudorandom signals of the generator.

In the absence of carry signals at the outputs of the adders AD1 and AD2, to the information inputs of the memory register RG1, through the multiplexer MUX, arrives a number from the information outputs of the adder AD1; moreover, if at least one of them is present, the number of information outputs are those of the adder AD2.

Compared with the known AFG and MAFG [10–15,28], the introduction of the second adder AD2, multiplexer MUX, and the establishment of new connections between these and other structural elements, allows changing the numbers in the registers RG1–RG6 in the range of values  $0 \div (p - 1)$ . Thus, AFG and MAFG operate with arbitrary module values according to Expressions (3) and (4), which confirmed our research, as mentioned in the following sections.

#### 4. Methods of AFG and MAFG Statistical Characteristics Research

AFG and MAFG, as with any cryptographic generators of pseudorandom sequences, are used in arguments whose values are enormous; therefore, there are some difficulties in determining their statistical characteristics.

In this regard, the following research methodology was used. Two models were created for each AFG and MAFG variant: firstly, the abstract, which is not directly related to the circuit design solution, and hardware, which corresponds to the proposed structure. For relatively small values of arguments, the identity of the models is proved. The study of statistical characteristics, with large values of arguments, is carried out using an abstract model.

The following algorithms represent different AFG and MAFG models. The hardware models are represented by equations that correspond to the structures' processes, shown in Figure 1. Abstract models are represented by equations that correspond to the processes that must occur in the additive Fibonacci generator when it operates with a module whose value can be arbitrary. Proving the identity of the results obtained with these models proves the correctness of the structures shown in Figure 1, in terms of achieving the desired result.

##### 4.1. Research of AFG Models

In AFG models, the logic circuit LC is not involved in the generator structure scheme (Figure 1).

The AFG hardware model operates in accordance with the following algorithm:

$$A = 2^n - p, \quad x_{i-5} = x_{i-4}, \quad x_{i-4} = x_{i-3}, \quad x_{i-3} = x_{i-2}, \quad x_{i-2} = x_{i-1}, \quad x_{i-1} = x_i, \quad x_i = x_{mp},$$

$$x_{ad1} = (x_{i-5} + x_{i-4}) \bmod 2^n, \quad \text{if } (x_{i-5} + x_{i-4}) < 2^n \quad \text{then } P_{ad1} = 0 \quad \text{else } P_{ad1} = 1,$$

$$x_{ad2} = (x_{ad1} + A) \bmod 2^n, \quad \text{if } (x_{ad1} + A) < 2^n \quad \text{then } P_{ad2} = 0 \quad \text{else } P_{ad2} = 1,$$

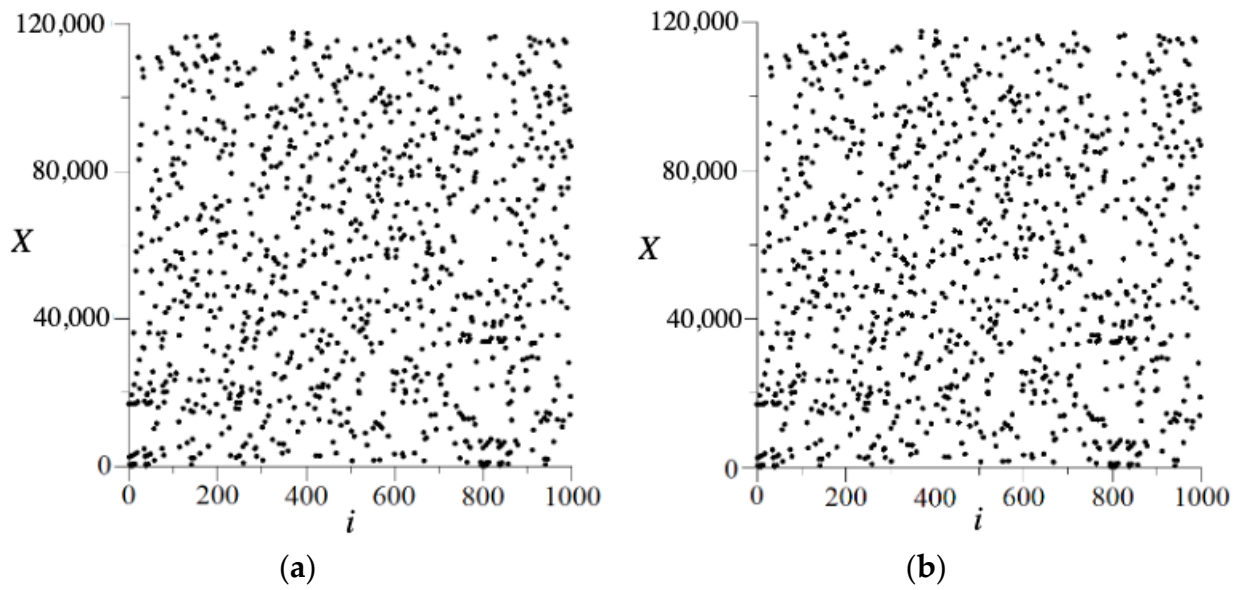
$$\text{if } (P_{ad1} = 0), \text{ and } (P_{ad2} = 0), \quad \text{then } x_{mp} = x_{ad1}, \text{ or } x_{mp} = x_{ad2},$$

where:  $x_i, x_{i-1}, x_{i-2}, x_{i-3}, x_{i-4}, x_{i-5}$ —numbers in registers RG1–RG6, respectively;  $x_{ad1}$  and  $x_{ad2}$ —numbers at the sum outputs of adders AD1 i AD2;  $P_{ad1}$  and  $P_{ad2}$ —numbers at the carry outputs of adders AD1 and AD2;  $x_{mp}$ —the number at the output of the multiplexer MUX;  $n$ —the number of the generator's structural elements binary bits (Figure 1).

The abstract AFG model is described by the following equations:

$$x_{i-5} = x_{i-4}, \quad x_{i-4} = x_{i-3}, \quad x_{i-3} = x_{i-2}, \quad x_{i-2} = x_{i-1}, \quad x_{i-1} = x_i, \quad x_i = x_{ad1}, \quad x_{ad1} = (x_{i-5} + x_{i-4}) \bmod p.$$

Figure 2 shows the dependences of the current values of pseudorandom numbers  $X$ , generated by AFG on the iteration step number,  $i$ , for the hardware and abstract model with the same initial value,  $X(0)$ .



**Figure 2.** Current values of pseudorandom numbers  $X$  (for AFG): (a) hardware model:  $p = 7$ ,  $n = 3$ ,  $A = 2^n - p = 1$ ,  $X(0) = 1$ ; (b) abstract model:  $p = 7$ ,  $X(0) = 1$ .

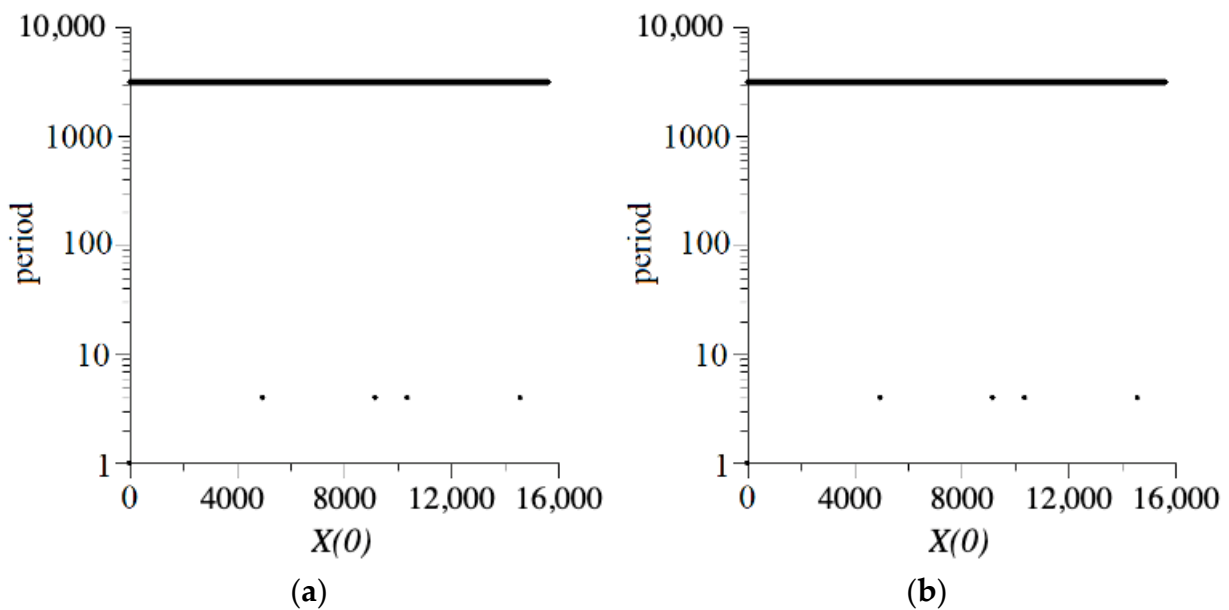
Numbers  $X$  and  $X(0)$  are defined by the following expressions:

$$X = p^5x_i + p^4x_{i-1} + p^3x_{i-2} + p^2x_{i-3} + px_{i-4} + x_{i-5}, \tag{6}$$

$$X(0) = p^5x_i(0) + p^4x_{i-1}(0) + p^3x_{i-2}(0) + p^2x_{i-3}(0) + px_{i-4}(0) + x_{i-5}(0), \tag{7}$$

where:  $x_i(0)$ ,  $x_{i-1}(0)$ ,  $x_{i-2}(0)$ ,  $x_{i-3}(0)$ ,  $x_{i-4}(0)$ ,  $x_{i-5}(0)$ —initial values of numbers  $x_i$ ,  $x_{i-1}$ ,  $x_{i-2}$ ,  $x_{i-3}$ ,  $x_{i-4}$ ,  $x_{i-5}$ , respectively.

Figure 3 on a logarithmic scale shows the dependence of the repetition periods of the AFG pseudorandom sequence numbers from the initial values,  $X(0)$ .



**Figure 3.** Dependencies of repetition periods from  $X(0)$  (for AFG): (a) hardware model:  $p = 5$ ,  $n = 3$ ,  $A = 2^n - p = 3$ ,  $X(0) = 0 \div p^6 - 1$ ; (b) abstract model:  $p = 5$ ,  $X(0) = 0 \div p^6 - 1$ .

The results (Figures 2 and 3) indicate complete identity of hardware and abstract models for forming a pseudorandom numbers sequence. Similar results were obtained for other  $p$  values, in particular for  $p$  values that are primes.

4.2. Research of MAFG Models

MAFG models: Figure 1 shows generator structure scheme with using logic circuit LC. The hardware model of the MAFG, operating according to the following algorithm:

$$A = 2^n - p, \quad x_{i-5} = x_{i-4}, \quad x_{i-4} = x_{i-3}, \quad x_{i-3} = x_{i-2}, \quad x_{i-2} = x_{i-1}, \quad x_{i-1} = x_i, \quad x_i = x_{mp},$$

$$a = b_0 \oplus b_1 \oplus \dots \oplus b_s$$

$$x_{ad1} = (x_{i-5} + x_{i-4} + a) \bmod 2^n, \quad \text{if } (x_{i-5} + x_{i-4} + a) < 2^n \quad \text{then } P_{ad1} = 0 \quad \text{else } P_{ad1} = 1,$$

$$x_{ad2} = (x_{ad1} + A) \bmod 2^n, \quad \text{if } (x_{ad1} + A) < 2^n \quad \text{then } P_{ad2} = 0 \quad \text{else } P_{ad2} = 1,$$

$$\text{if } (P_{ad1} = 0) \quad \text{and} \quad (P_{ad2} = 0) \quad \text{then } x_{mp} = x_{ad1} \quad \text{else } x_{mp} = x_{ad2},$$

where:  $b_i$ —values of the number  $x_i$  binary bits.

Abstract model of the MAFG operating according to the following equation:

$$x_{i-5} = x_{i-4}, \quad x_{i-4} = x_{i-3}, \quad x_{i-3} = x_{i-2}, \quad x_{i-2} = x_{i-1}, \quad x_{i-1} = x_i, \quad x_i = x_{ad1},$$

$$a = b_0 \oplus b_1 \oplus \dots \oplus b_s$$

$$x_{ad1} = (x_{i-5} + x_{i-4} + a) \bmod p.$$

Figure 4 shows the dependences of the current values of pseudorandom numbers,  $X$ , that were generated by the MAFG on the iteration step number,  $i$ , for the hardware and abstract model with the same initial value,  $X(0)$ .

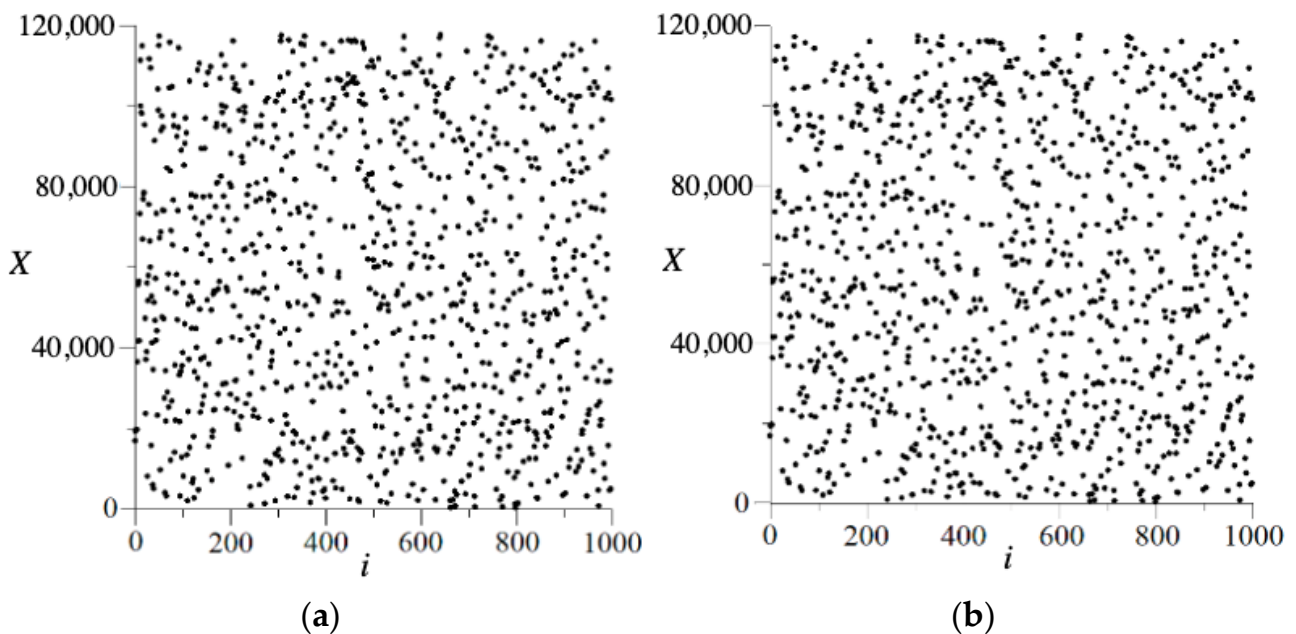
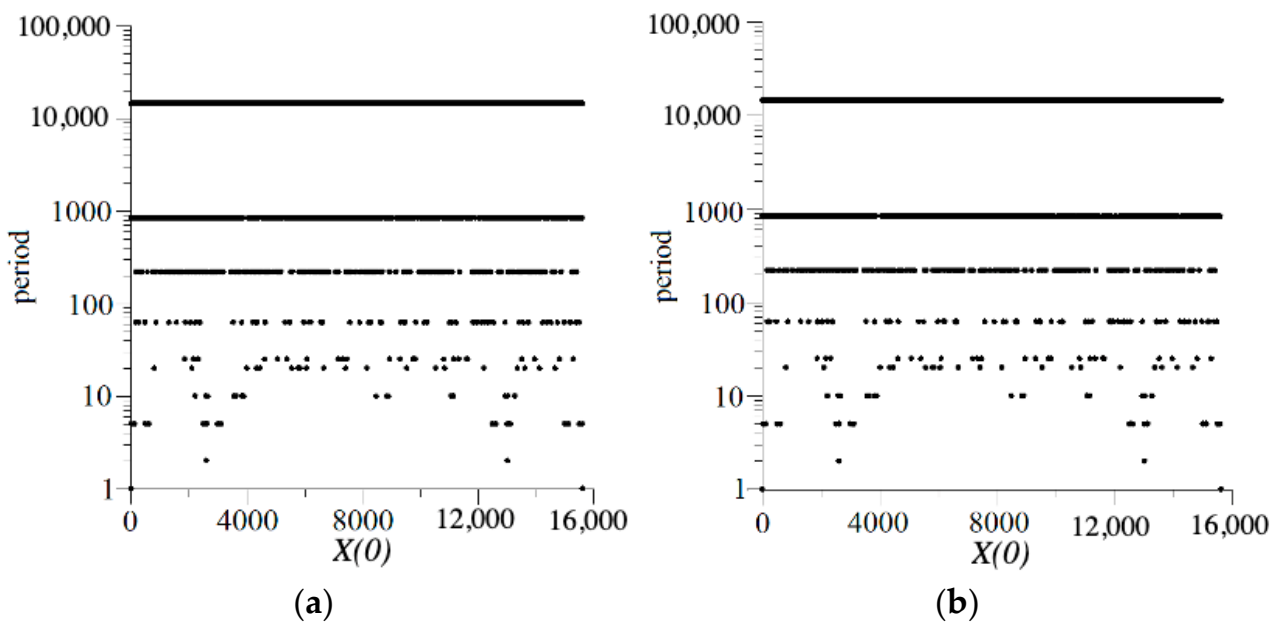


Figure 4. Current values of pseudorandom numbers  $X$  (for MAFG): (a) hardware model:  $p = 7, n = 3, A = 2^n - p = 1, a = b_0 \oplus b_1 \oplus b_2, X(0) = 1$ ; (b) abstract model:  $p = 7, a = b_0 \oplus b_1 \oplus b_2, X(0) = 1$ .

Figure 5 shows, on a logarithmic scale, the dependences of repetition periods of MAFG pseudorandom sequence on the initial values,  $X(0)$ .



**Figure 5.** Dependencies of repetition periods from  $X(0)$ : (a) hardware model:  $p = 5, n = 3, A = 2^n - p = 3, a = b_0 \oplus b_1 \oplus b_2, X(0) = 0 \div p^6 - 1$ ; (b) abstract model:  $p = 5, a = b_0 \oplus b_1 \oplus b_2, X(0) = 0 \div p^6 - 1$ .

The results (Figures 4 and 5) indicate complete identity hardware and abstract models for forming the pseudorandom numbers sequence. Similar results were obtained for other  $p$  values, in particular for  $p$  values that are primes.

**5. Results**

*5.1. Research of Repetition Periods of AFG and MAFG Pseudorandom Sequences*

The following research was conducted using an abstract model considering proven identity hardware and abstract generators models. It is necessary to speed up the simulation process.

Table 1 presents the received results of AFG and MAFG repetition periods,  $Tp$ , for a few small module  $p$  values that determined on the whole set of possible values of the initial number,  $X(0) = 0 \div p^6 - 1$ .

**Table 1.** Repetition periods of AFG and MAFG output sequences for  $p$  value on the whole set of possible values,  $X(0) = 0 \div p^6 - 1$ .

Some $p$ Values	Max and Min Repetition Period Values	
	AFG (without Logic Circuit LC)	MAFG (with Logic Circuit LC)
2	63	10 2
3	728	315 5
5	3124 4	14,409 5
7	2400 24	105,833 11,360

In this case, for MAFG, the output signal value  $a$  of the logic circuit LC (Figure 1) was determined, according to Equation (5), as the sum for the module 2 for all bits of number  $x_i$  in the register Pr1.

Table 1 shows the maximum and minimum values of the period  $Tp$ . It should be noted that when  $p = 2$  and  $p = 3$  on the whole set,  $X(0) = 0 \div p^6 - 1$  fixed only one value  $Tp = p^6 - 1$ . It coincides with the known theoretical results presented in Ref. [25].

Where for larger values of module  $p$ , determination of repetition period,  $Tp$ , on the whole set of values,  $X(0) = 0 \div p^6 - 1$ , requires a lot of machine time, all the following research was conducted for a fixed value,  $X(0) = 1$ . Table 2 shows the repetition period,  $Tp$ , for some  $p$  values and fixed values,  $X(0) = 1$ .

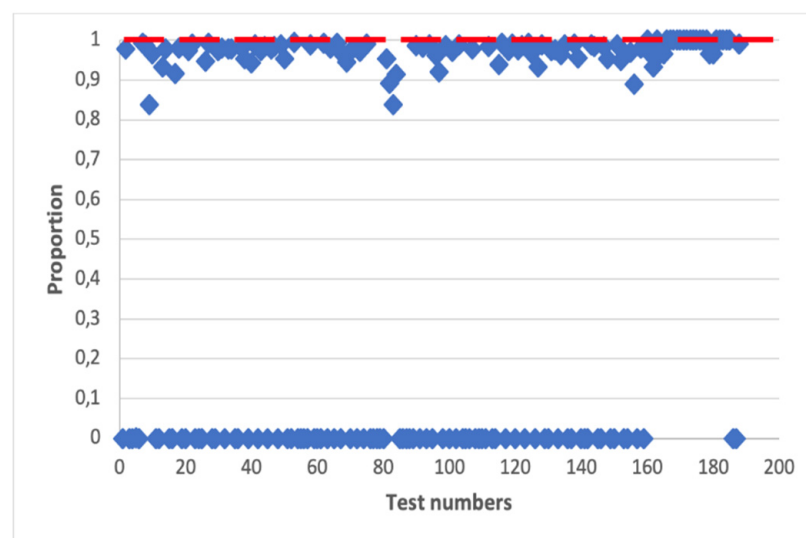
**Table 2.** Repetition periods of AFG and MAFG output sequences for some  $p$  values when  $X(0) = 1$ .

Some $p$ Values	Repetition Periods $Tp$	
	AFG (without Logic Circuit LC)	MAFG (with Logic Circuit LC)
11	118,103	1,601,719
13	371,291	2,636,108
17	88,415	9,810,767
19	2,476,097	26,974,957
37	845,657	382,733,921
41	1679	432,850,590
43	1,116,087	5,459,242,931
73	1,401,242,835	8,949,513,501
137	4,387,429,945	$>10^{10}$

Based on the research results of the output sequences, repetition periods of the AFG and proposed MAFG, in which the modules of the recurrent equations are prime numbers, such a conclusion can be made. When  $p > 3$  the repetition periods MAFG is significantly greater than the AFG. When  $p = 2$  and  $p = 3$ , the repetition periods of AFG reach, theoretically, the maximum value,  $Tp = p^6 - 1$ , for all possible values,  $X(0)$ .

### 5.2. Research of Statistical Characteristics of AFG and MAFG Pseudorandom Sequences

Research the statistical characteristics of the output pseudorandom bit sequences of AFG and MAFG for some  $p$  values were carried out with the NIST test package [29–31]. Results shows in Figures 6–9. Figure 6 presents a statistical portrait of the AFG output sequence at  $p = 137$ .



**Figure 6.** Statistical portrait of the AFG output sequence at  $p = 137$ ,  $X(0) = 1$ .



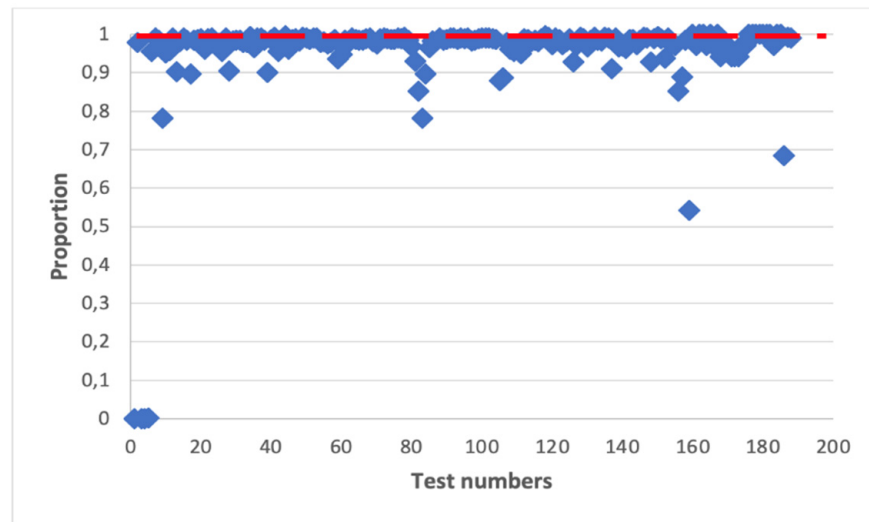


Figure 7. Statistical portrait of the MAFG output sequence at  $p = 137$ ,  $X(0) = 1$ ,  $a = b_0 \oplus b_1 \dots \oplus b_7$ .

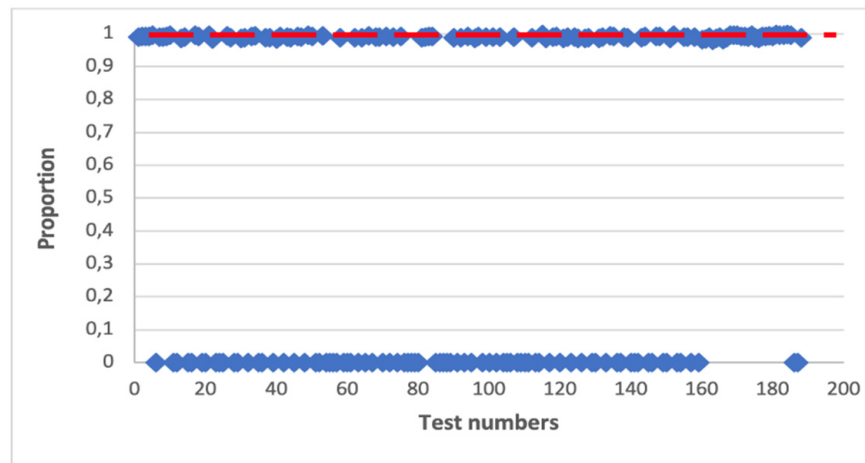


Figure 8. Statistical portrait of the AFG output sequence at  $p = 65,537$  and  $X(0)$ .

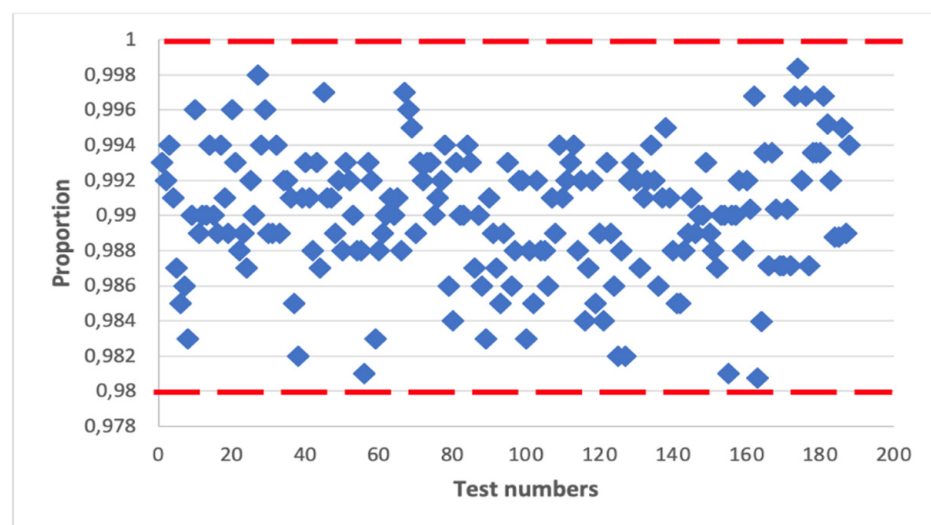


Figure 9. Statistical portrait of the MAFG output sequence at  $p = 65,537$ ,  $X(0) = 1$ ,  $a = b_0 \oplus b_1 \dots \oplus b_{16}$ .

As can be seen from Figure 6, the most tests valued at 0 and did not fall within the specified interval; meaning that the sequence does not meet the randomness requirements.

Figure 7 shows MAFG using the same initial data as AFG. The sequence also does not meet the randomness requirements, but there is a significant improvement over the AFG. In particular, most test values are above 0. So, the proposed modification demonstrates positive dynamics.

Figure 8 shows the statistical portrait of the AFG output sequence at  $p = 65,537$ . The tests failed and did not meet the randomness requirements.

Figure 9 presented the statistical portrait of the MAFG output sequence using the same parameters as AFG. As can be seen, all tests are within the allowable range. It means that such sequence has high statistical characteristics and meet the randomness requirements.

Analysis of statistical portraits (Figures 6–9) shows that, with the same parameters, the statistical characteristics of the output pseudorandom sequences of MAFG significantly predominated in the AFG. Thus, at  $p = 65,537$ ,  $X(0) = 1$ , and  $a = b_0 \oplus b_1 \dots \oplus b_{16}$  (Figure 9) MAFG statistical characteristics entirely pass all NIST tests.

The conducted research proves that the proposed Fibonacci additive generators can operate by recurrent equations, whose modules values can be arbitrary, including modules whose values are prime numbers. It distinguishes them from the known additive Fibonacci generators, whose value of the modules is equal to the power of two. That is, the class of proposed generators includes the known generators as a subclass. At the same time, the proposed generators have the best statistical characteristics and designs for hardware implementation primarily, in which will achieve their maximum speed when implementing the proposed structures in a modern element base, for example, in programmable logic integrated circuits (PLDs).

## 6. Conclusions

The present article proposes new structures of AFG and MAFG, in which adding additional structural elements allows the operation of the generator with arbitrary values of the modulus of the recurrent equation, in particular, with modules whose values are prime numbers.

In the present study, we proved the identity of hardware and abstract models, suggesting that the principles laid down in the organization of the AFG and MAFG structures with modules of prime numbers ensure their effective hardware implementation.

For the basic function  $x_i = (x_{i-5} + x_{i-4}) \bmod p$ , the MAFG selected for the research, which functions according to the equation  $x_i = (x_{i-5} + x_{i-4} + a) \bmod p$ , significantly predominated over AFG in the repetition period and statistical characteristics for all module values  $p > 3$ .

The AFG, at  $p = 2$  and  $p = 3$ , fixed the maximum possible repetition period,  $T_p = p^6 - 1$ , for all possible initial values of generator registers settings.

In further research, an important task is to find primitive polynomials over the field  $GF(p)$  for other values,  $p > 3$ , create AFG and MAFG structures for these values, and research their characteristics.

The obtained results can be used not only in the design of information security tools but also in other technology fields, such as in simulating random processes in measuring technologies.

**Author Contributions:** Conceptualization, V.M., M.S., and O.H.; methodology, M.S., and D.J.; validation, V.M., M.S., and P.S.; formal analysis, M.S., O.H., and P.S.; investigation, V.M., M.S., O.H., and D.J.; data curation, V.M., M.S., and O.H.; writing—original draft preparation, V.M., M.S., and O.H.; writing—review and editing, V.M., M.S., O.H., and D.J.; funding acquisition, M.K. All authors have read and agreed to the published version of the manuscript.

**Funding:** The research work reported in this paper was in part supported by the National Centre for Research and Development, Poland, under the project No. POIR.04.01.04-00-0048/20.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Mohamed, K.; Ali, F.H.H.M.; Ariffin, S.; Zakaria, N.H.; Pauzi, M.N.M. An Improved AES S-box Based on Fibonacci Numbers and Prime Factor. *Int. J. Netw. Secur.* **2018**, *20*, 1206–1214.
2. Belov, A.A.; Kalitkin, N.N.; Tintul, M.A. Visual Verification of Pseudo-Random Number Generators. *Keldysh Inst. Preprints* **2019**, *137*, 1–28. [\[CrossRef\]](#)
3. Stakhov, A.; Massingue, V.; Sluchenkova, A. *Introduction into Fibonacci Coding and Cryptography*; Publish Osnova: Kharkov, Ukraine, 1999.
4. Agarwal, P.; Agarwal, N.; Saxena, R. Data Encryption through Fibonacci Sequence and Unicode Characters. *MIT Int. J. Comput. Sci. Inf. Technol.* **2015**, *5*, 79–82.
5. Gosai, I. Fibonacci Sequence and Its Applications. *IJRAR* **2019**, *6*, 241–247.
6. Ahamad, M.V.; Siddiqui, U.M.; Masroor, M.; Fatima, U.A. Modified Playfair Encryption Using Fibonacci Numbers. *Int. J. Adv. Technol. Eng. Sci.* **2017**, *5*, 347–351.
7. Baldoni, S.; Battisti, F.; Carli, M.; Pascucci, F. On the Use of Fibonacci Sequences for Detecting Injection Attacks in Cyber Physical Systems. *IEEE Access* **2021**, *9*, 41787–41798. [\[CrossRef\]](#)
8. Agarwal, A.; Agarwal, S.; Singh, B.K. Algorithm for data encryption & decryption using Fibonacci primes. *J. Math. Control Sci. Appl.* **2020**, *6*, 63–71.
9. Yacoab, M.; Sha, M.; Ahmed, M.M. Secured Data Aggregation Using Fibonacci Numbers and Unicode Symbols for Wsn. *Int. J. Comput. Eng. Technol.* **2019**, *10*, 218–225. [\[CrossRef\]](#)
10. Mandrona, M.; Maksymovych, V. Investigation of the statistical characteristics of the modified Fibonacci generators. *J. Autom. Inf. Sci.* **2014**, *46*, 48–53. [\[CrossRef\]](#)
11. Maksymovych, V.; Harasymchuk, O.; Kostiv, Y.; Mandrona, M. Implementation of modified additive lagged Fibonacci generator. *Chall. Mod. Technol.* **2016**, *7*, 3–6.
12. Maksymovych, V.; Mandrona, M.; Garasimchuk, O.; Kostiv, Y. A study of the characteristics of the Fibonacci modified additive generator with a delay. *J. Autom. Inf. Sci.* **2016**, *48*, 76–82. [\[CrossRef\]](#)
13. Maksymovych, V.N.; Harasymchuk, O.I.; Mandrona, M.N. Designing Generators of Poisson Pulse Sequences Based on the Additive Fibonacci Generators. *J. Autom. Inf. Sci.* **2017**, *49*, 1–13. [\[CrossRef\]](#)
14. Mandrona, M.N.; Maksymovych, V.N. Comparative Analysis of Pseudorandom Bit Sequence Generators. *J. Autom. Inf. Sci.* **2017**, *49*, 78–86. [\[CrossRef\]](#)
15. Maksymovych, V.; Harasymchuk, O.; Oprisky, I. The Designing and Research of Generators of Poisson Pulse Sequences on Base of Fibonacci Modified Additive Generator. International Conference on Theory and Applications of Fuzzy Systems and Soft Computing. *Adv. Comput. Sci. Eng. Educ.* **2018**, *754*, 43–53. [\[CrossRef\]](#)
16. Maksymovych, V.; Harasymchuk, O.; Karpinski, M.; Shabatura, M.; Jancarczyk, D.; Kajstura, K. A New Approach to the Development of Additive Fibonacci Generators Based on Prime Numbers. *Electronics* **2021**, *10*, 2912. [\[CrossRef\]](#)
17. Deshmukh, P.; Sadawarte, Y. Pseudo-Random Number Generation by Fibonacci and Galois LFSR Implemented on FPGA. In Proceedings of the IJCA Proceedings on International Conference on Advancements in Engineering and Technology (ICAET 2015) ICQUEST 2015, Wardha, India, 1–3 October 2015.
18. Zulfikar, Z.; Away, Y.; Rafiq, S.N. FPGA-Based Design System for a Two-segment Fibonacci LFSR Random Number Generator. *Int. J. Electr. Comput. Eng.* **2017**, *7*, 1882–1891. [\[CrossRef\]](#)
19. Nannipieri, P.; Di Matteo, S.; Baldanzi, L.; Crocetti, L.; Belli, J.; Fanucci, L.; Saponara, S. True Random Number Generator Based on Fibonacci-Galois Ring Oscillators for FPGA. *Appl. Sci.* **2021**, *11*, 3330. [\[CrossRef\]](#)
20. Cybulski, R. Pseudo-random number generator based on linear congruence and delayed Fibonacci method: Pseudo-random number generator based on linear congruence and delayed Fibonacci method. *Tech. Sci.* **2021**, *24*, 331–349. [\[CrossRef\]](#)
21. Opoku-Mensah, E.; Abilimi, C.A.; Boateng, F.O. Comparative Analysis of Efficiency of Fibonacci Random Number Generator Algorithm and Gaussian Random Number Generator Algorithm in a Cryptographic System. *Comput. Eng. Intell. Syst.* **2013**, *4*, 50–57.
22. Mandrona, M.M.; Maksymovych, V.M.; Harasymchuk, O.I.; Kostiv, Y.M. Generator of pseudorandom bit sequence with increased cryptographic immunity. *Metall. Min. Ind.* **2014**, *6*, 24–28.
23. Amiruddin, A.; Ratna, A.A.P.; Sari, R.F. Construction and Analysis of Key Generation Algorithms Based on Modified Fibonacci and Scrambling Factors for Privacy Preservation. *Int. J. Netw. Secur.* **2019**, *21*, 250–258. [\[CrossRef\]](#)
24. Oduwole, H.K.; Shehu, S.; Adegoke, G.K.; Onubogu, J.L. Fibonacci Random Number Generator using Lehmer's Algorithm. *Math. Theory Modeling* **2013**, *3*, 56–62.
25. Schneier, B. *Applied Cryptography: Protocols, Algorithms, and Source Code in C*; John Wiley & Sons: Hoboken, NJ, USA, 2007; p. 675.
26. Slepovichev, I. *Pseudo-Random Number Generators*; SSU: Saratov, Russia, 2017; p. 118.
27. Beletsky, A.; Kovalchuk, A.; Novikov, K.; Poltoratsky, D. *Tables of Binary Irreducible Polynomials*; Monograph book; Agrar Media Group: Kyiv, Ukraine, 2021; p. 400.
28. Srinivas, A. Lagged Fibonacci Random Number Generators for Distributed Memory Parallel Computers. *J. Parallel Distrib. Comput.* **1997**, *45*, 1–12.

29. Faster Randomness Testing with the NIST Statistical Test Suite. Available online: [https://crocs.fi.muni.cz/\\_media/public/crocs/sys\\_space\\_2014.pdf](https://crocs.fi.muni.cz/_media/public/crocs/sys_space_2014.pdf) (accessed on 20 December 2021).
30. NIST SP 800-22 Version 1a. *A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications*; NIST: Gaithersburg, MD, USA, 2010; p. 131. Available online: <https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-22r1a.pdf> (accessed on 20 December 2021).
31. Gorbenko, I.D.; Gorbenko, Y.I. *Applied Cryptology: Theory. Practice. Application*; Fort Publishing House: Kharkiv, Ukraine, 2012; p. 880.