IOP Conf. Series: Journal of Physics: Conf. Series **783** (2017) 012064 doi:10.1088/1742-6596/783/1/012064

# Ternary jitter-based true random number generator

**Rustam Latypov**[1] **and Evgeni Stolov**[2]

Department of Computational Mathematics and Information Technology
Kazan Federal University, Kazan, Russia 420008

E-mail: [1]Roustam.Latypov@kpfu.ru ;[2]ystolov@kpfu.ru

**Abstract.** In this paper a novel family of generators producing true uniform random numbers in ternary logic is presented. The generator consists of a number of identical ternary logic combinational units connected into a ring. All the units are provided to have a random delay time, and this time is supposed to be distributed in accordance with an exponential distribution. All delays are supposed to be independent events. The theory of the generator is based on Erlang equations. The generator can be used for test production in various systems. Features of multidimensional random vectors, produced by the generator, are discussed.

## 1. Introduction

Random number generators (RNG) are important in various areas of the science and technology, involving testing of circuits, stochastic modeling, generation of the cryptographic keys, random initialization of variables in cryptographic protocols, and communication security [1, 2]. A true random number generator (TRNG) is a type of RNG that generates unforeseeable data, in contrast to pseudo-random number generator which only provides statistically strong (but completely predictable) sequences. Recently, there were invented many types of TRNGs. Most of them are based on event known as "jittering" in electronic schemes. That is a kind of instability in signals produced by an electronic device [3]. The example of such a generator employing instability of clock generator controlling the work of a D-trigger is presented in [4]. Another idea exploits instability of outputs of a combinational circuit with its outputs being connected to inputs. [5]. The simplest example of a TRNG based on that invention is depicted in Figure 1. This idea was used by many authors in the binary case. Some references to original papers can be found in [5, 6, 7]. Under some assumptions on that process, one can develop a theory of jittering and create a circuit working as a generator of random values. Recently, it was proposed to use generators with jitter which are constructed by using reprogrammable digital circuits or constructions based on meta-stability [6, 8]. In both these approaches, the jitter is used for bit sequence generation [5, 6]. Even though there exist a number of offers for hardware-based TRNGs, the problem of discovering an efficient method for generation of true random numbers feasible with the use of solely logic gates, remains actual.

The main weaknesses of binary elements are the interconnection and pin-out problems. On the other hand, in the recent years, ternary circuits and their possible physical implementation have been studied increasingly. The detailed discussion can be found in [9, 10, 11]. Moreover, next generation nanoelectronic devices were developed (such as the resonant tunnelling devices and quantum-dot cellular automata) [10, 11] . Such schemes are often those of multiple states and could turn out a better device in multivalued circuits. The development of ternary logic

devices seems to be prospective in providing a higher speed of arithmetic operations; hence the development of such circuits becomes an actual problem. In our paper, we present a new family of TRNGs which can be implemented by using only ternary logic gates. The main purpose of the work is a theoretical study of the behavior of such devices.

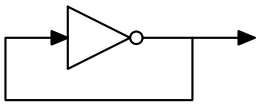

**Figure 1.** Inverter in jittering mode.

The remaining part of the paper has the following organization. In Section II the basic schema of the generator as well as its mathematical theory are given; in Section III statistical properties of the generator are studied; Section IV is dedicated to implementation of the generator; conclusions are given in Section V.

Throughout this paper, while dealing with matrices, the following notations are used:

(i) A[i,j] is an element of a matrix A, standing at i-th row and j-th column .

(ii) A[i,*] is the i-th row of the matrix A, A[ *,j] is the j-th column of this matrix .

(iii) $\theta$ is the zero matrix.

## 2. Preliminaries and basic schema

This paper proposes and analyzes a TRNG based on a balanced ternary logic system. Balanced ternary logic is a special case of ternary logic where the digits have values -1, 0, and 1. We will work with two-input single-output functions or gates. That means there are $3^2 = 9$ possible inputs, and $3^9 = 19383$ possible gates. Consider the scheme shown in Figure 2. Here $F$ stands for

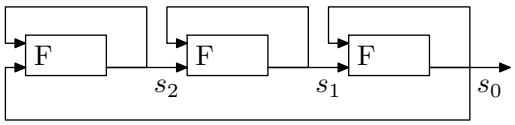


**Figure 2.** Version of the generator containing three ternary units.

a ternary combinational gate, $c = F(a, b), \quad a, b, c \in \{-1, 0, 1\}$. Generally, the schema consists of $N$ gates. Renumber all the units via numbers in the interval $[0, N-1]$. The output of the unit which uses the number $k$ is connected to one of the inputs of the same unit and to the input of the unit with the number $k-1 \bmod N$ (see Figure 2). After the event that at least one input signal changes, the output signal might change too, but it takes a delay time $DT$. In the paper, the following assumptions are accepted:

(i) $DT$ is a stochastic value having an exponential distribution for all units with the same parameter $T$. The latter means that $P(DT < d) = 1 - \exp(-Td)$.

(ii) At any time, only one unit can change its output; all those events are independent.

In what follows, we will assume that $T = 1$, it does not limit the generality of the arguments. All the restrictions, that are imposed on the system, allow us to implement Erlang theory [12], describing functionality of the generator. Let $s_{\mathrm{k}}(t)$ be the output signal of unit with number k at time $t$. Denote by vector $\boldsymbol{S}(t) = \langle s_0(t), \ldots, s_{\mathrm{N}-1}(t) \rangle$ the state of the TRNG at time $t$. Hence the system has $\mathrm{M} = 3^{\mathrm{N}}$ states, which also are indexed by numbers from 0 to M-1. For each $\mathrm{n} \in [0, \mathrm{M}-1]$ let $\mathrm{i}_1, \ldots, \mathrm{i}_{\mathrm{m}}$ be a list of all numbers of the states for which it is possible to transfer the TRNG to the state $\boldsymbol{S}_{\mathrm{n}}$ from states $\boldsymbol{S}_{\mathrm{i_k}}, \mathrm{k} = 1, \ldots, \mathrm{m}$ as a result of the change of output signal of one of the units. This list of the states depends on n. Create a system

**Table 1.** Functions suitable for generator building.

| Name | $F(0,0)$ | $F(1,1)$ | $F(-1,-1)$ |
|---|---|---|---|
| $F_1$ | 1 | -1 | 0 |
| $F_2$ | 1 | -1 | 1 |

of Erlang type differential equations describing dynamics of the TRNG [12]. Let $P_{\mathrm{n}}(t)$ be the probability that the TRNG is in state number n at time $t$. The probability $P_{\mathrm{n}}(t+\Delta t)$ is the sum of probabilities: $(1-\Delta t)^N P_{\mathrm{n}}(t)$ – no units changed its output, so the TRNG did not change its state; and $\Delta t \sum_{i_k} P_{\mathrm{i_k}}$ - exactly one unit changed its output and the TRNG transferred to state $\boldsymbol{S}_{\mathrm{n}}$. Here we used evident approximations: $\exp(-\Delta t) \sim 1-\Delta t$ and $1-\exp(-\Delta t) \sim \Delta t$ for small $\Delta t$. Making $\Delta t$ to tend to 0, we get for each $\mathrm{n} \in [0, M-1]$

$$\frac{\mathrm{d}P_{\mathrm{n}}}{\mathrm{d}t} = -NP_{\mathrm{n}}(t) + \sum_{\mathrm{k}=1}^{m} P_{\mathrm{i_k}}(t), \tag{1}$$

where $\mathrm{m}, \mathrm{i}_1, \ldots, \mathrm{i}_{\mathrm{m}}$ depend on n. This is an Erlang type equation, usually used in description of Queueing Systems.

*2.1. Choice of function F*

Equation (1) holds for any function F mentioned above. Now, we have to impose some restrictions on this function in order to guarantee some properties of the TRNG. First of all, the TRNG must have no stable states. It means that inequality $\forall t > 0\, P_{\mathrm{n}}(t) < 1$ holds for any n. Suppose that the following formula takes place

$$c = F(a,b), \quad \forall (a,b) \quad c \neq a, b. \tag{2}$$

From (2) it follows that any unit will change its output after a delay time, so the TRNG possesses no stable states. Find all functions possessing the property (2). First of all, if $a \neq b$, then $F(a,b)$ is defined uniquely, and therefore $F(a,b) = F(b,a)$. The latter means that we have to define just $F(-1,-1)$, $F(0,0)$, and $F(1,1)$. Let $\sigma$ be an arbitrary permutation of the values $-1, 0, 1$. $F(a,b)$ and $\sigma(F(\sigma(a), \sigma(b))$ can be thought of as the same function. Two functions meeting (2) are given in table 1.

**Proposition 1**

There are only two different ternary functions meeting (2).

The proof of the Proposition can be seen in appendices. Further, assume that $F = F_1$.

*2.2. Dynamics of States of Generator*

As far as the type of $F$ is defined, one can find all parameters in (1). Using matrix form, rewrite the system as follows:

$$\frac{\mathrm{d}\boldsymbol{P}}{\mathrm{d}t} = Matr \cdot \boldsymbol{P} \tag{3}$$

Here $\boldsymbol{P} = \langle P_0, P_1, \ldots, P_{\mathrm{M}-1} \rangle^T$. Since the size of $Matr$ increases rapidly while number of units grows in generator, we restrict ourselves to the cases $N = 1$, $N = 2$, or $N = 3$ in examples. First

of all, one has to index all states of the TRNG. Let $\boldsymbol{S} = \langle s_0, s_1, \ldots, s_{\mathrm{N}-1}\rangle$ be a state of TRNG, $s_{\mathrm{k}} \in \{-1, 0, 1\}$. Index of $(S)$ is a number

$$\mathrm{Ind}(\boldsymbol{S}) = \sum_{\mathrm{k}=0}^{N-1} 3^{\mathrm{k}}(s_{\mathrm{k}} + 1) \tag{4}$$

For example, for $N = 2$, $Matr$ is presented in (5)

$$Matr = \begin{pmatrix} -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -2 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & -2 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & -2 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & -2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & -2 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & -2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 \end{pmatrix} \tag{5}$$

Let us explain the sense of items in this matrix. Accordingly (4), the state $\langle 0, 1\rangle$ has index 7. In the row number 7 (starting from zero), $Matr$ has 1 in positions 1, 4,6. From (4) it follows that these indexes correspond to the following states: $\langle 0, -1\rangle$, $\langle 0, 0\rangle$, and $\langle -1, 1\rangle$. It means that the TRNG can transfer from those states to state 7. Indeed, suppose the TRNG is being in the state 6. The unit 0 has on its inputs -1 from unit 0 and 1 from unit 1. In accordance with the definition of $F_0$, it changes its output to 0; thus it transfers to the state 7. All the other cases can be analyzed the same way.

Formally, possessing the matrix $Matr$ and the differential equations (3), one can find the vector $\boldsymbol{P}(t)$ for any $t$. Our goal is to obtain properties of this vector without searching an exact solution of the system.

## 3. Asymptotic properties of $\boldsymbol{P}(t)$

Lines with the indexes 0, 4, and 8 in (5) have no 1's. It means that there are no transfers of the TRNG to the states with such indexes; in other words, these states are unattainable ones.

**Proposition 2**

The states of form $\langle x, x, \ldots, x\rangle$ with $x \in \{-1, 0, 1\}$ are the only unattainable states of the generator with $N > 1$..

The proof of the Proposition can be seen in appendices.

In general case, exclude from consideration the states of the form $\langle x, x, \ldots, x\rangle$. It means, one should exclude three items from the vector $\boldsymbol{P}$ and three rows and three columns from the matrix $Matr$ in (3). We keep the previous notation for new $\boldsymbol{P}$ and $Matr$. Since matrix $Matr$ in (3) is constant, the solution of the equation can be presented in form

$$\boldsymbol{P}(t) = \exp(Matr \cdot t) \cdot \boldsymbol{P}(0), \tag{6}$$

where $\boldsymbol{P}(0)$ is a stochastic vector defining initial state of the TRRG [14]. According to the definition,

$$Matr = Q - N \cdot I, \tag{7}$$

where $I$ is the identity matrix, while $Q$ has binary entries (1 or 0). Both the matrices have the size $M' \times M'$, $M' = M - 3$.

**Proposition 3**
Let $e_1, e_2, \ldots, e_{\mathrm{M}'}$ be all eigenvalues of $Q$ and

$$|e_1| \geq |e_2| \geq \cdots \geq |e_{\mathrm{M}'}| \tag{8}$$

Then $e_1 = N$.

The proof of the Proposition can be found in appendices.

It follows that $m_1 = 0$ is an eigenvalue for $Matr$, and for all other eigenvalues $m_2, \ldots, m_{\mathrm{M}'}$ of this matrix the inequality $\mathrm{Real}(m_j) \leq 0$, $j = 2, \ldots, M'$ holds. Suppose one has the strong inequality

$$\mathrm{Real}(m_j) < 0,\ j = 2, \ldots, M'. \tag{9}$$

It means that the matrix $E(t) = \exp(t \cdot Matr)$ has a single root 1, while all other roots of this matrix have absolute values less than 1. If $t \to \infty$, then $E(t)$ converges to a stable matrix. The same holds true for the vector $\boldsymbol{P}(t)$, and

$$\frac{\mathrm{d}\boldsymbol{P}(t)}{\mathrm{d}t} \to 0,\ t \to \infty$$

Following [12], one can find a stochastic vector $\bar{\boldsymbol{P}} = \boldsymbol{P}(\infty)$ by solving the system

$$Q \cdot \bar{\boldsymbol{P}} = N\bar{\boldsymbol{P}}. \tag{10}$$

*3.1. The case N=1*
There are three states of the TRNG: $\langle -1\rangle, \langle 0\rangle, \langle 1\rangle$ and

$$Matr = \begin{pmatrix} -1 & 0 & 1 \\ 1 & -1 & 0 \\ 0 & 1 & -1 \end{pmatrix}$$

The eigenvalues of $Matr$ are $0, -1.5, -1.5$, thus (9) is fulfilled, and $\bar{\boldsymbol{P}} = \langle 1/3, 1/3, 1/3\rangle$. It means that stable vector $\bar{\boldsymbol{P}}$ has the uniform distribution.

*3.2. The case N=2*
Here the eigenvalues of $Matr$ are $0, -1, -4, -3, -3, -1$, and the stable vector is $\langle 0.17, 0.17, 0.17, 0.17, 0.17, 0.17\rangle$; hence in this case we have the uniform distribution too.

*3.3. The case N=3*
In this case (9) is fulfilled, but stable vector $\bar{\mathbf{P}}$ has components

$$\begin{array}{cccccccccc} \langle & 0.037 & 0.037 & 0.037 & 0.037 & 0.074 & 0.037 & 0.037 & 0.037 & 0.037 \\ 0.037 & 0.037 & 0.037 & 0.037 & 0.037 & 0.074 & 0.037 & 0.037 & 0.037 & 0.074 & 0.037 \\ 0.037 & 0.037 & 0.037 & 0.037 & 0.037 & \rangle & & & & \end{array} \tag{11}$$

Hence we have a nonuniform final distribution of states. What is more, there exist the special states $\langle -1, 0, 1\rangle$, $\langle 1, -1, 0\rangle$, $\langle 0, 1-1\rangle$. In fact, it is the same state because each is a result of cyclic transform of the other states (see Figure 2). The corresponding lines in the table contain six 1's, while other lines contain three 1's.

*3.4. The general case*

Recall [13] that matrix $Q$ is a decomposable one if there exists a permutation matrix $Pr$ such that

$$Pr^T \cdot Q \cdot Pr = \begin{pmatrix} A & 0 \\ B & C \end{pmatrix}$$

where $A, C$ are square matrices. It can be proved that in general case matrix $Q$ in (7) is an indecomposable matrix. Because there is the restriction upon the size of the communication, we will omit the proof of this fact. It will be presented later in a paper dedicated to this problem. It is known ([13]) that for indecomposable matrices (9) holds. It means that the technique for calculation of distribution of states presented above is good for any number units in the TRNG. The only problem is to find eigenvectors of a matrix of large size. Examples show that stable distributions of $\bar{\boldsymbol{P}}$ are not uniform for $N > 2$. Since the corresponding stable vectors are of big size, we present the components of the vector in graphical form (see Figures 3). It means that

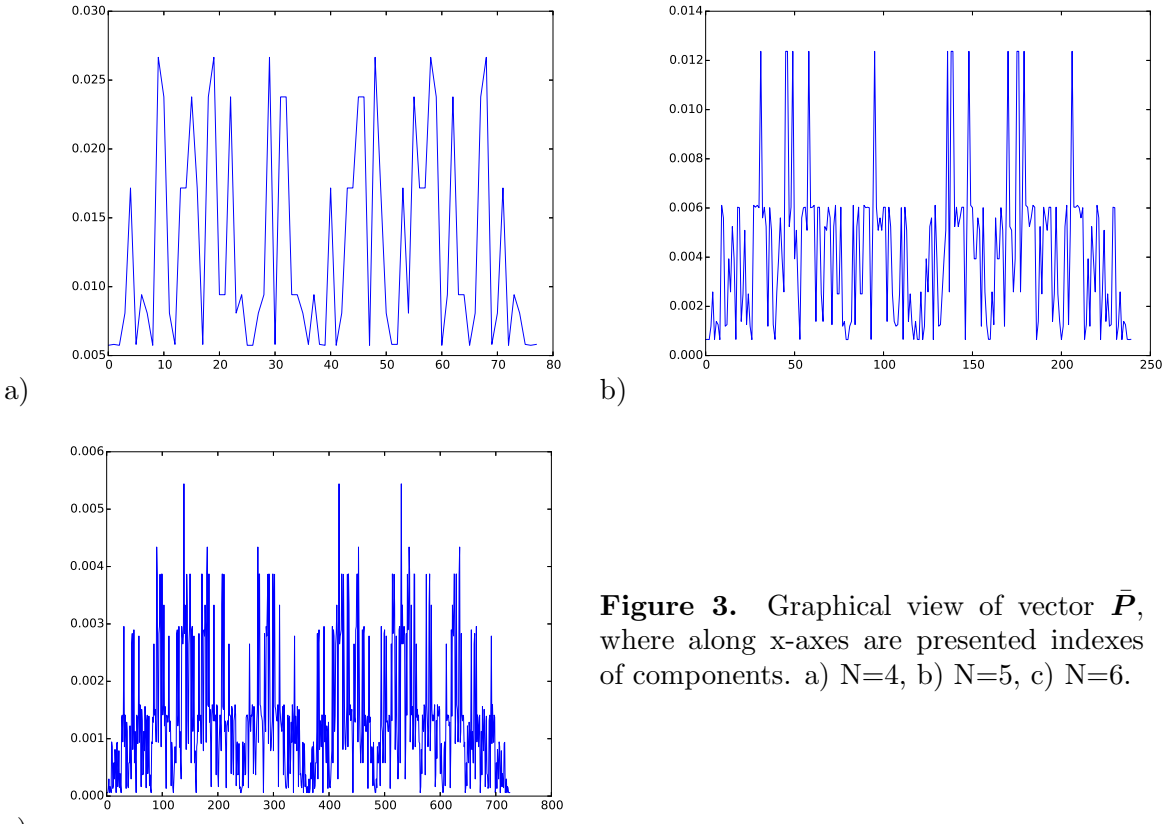


**Figure 3.** Graphical view of vector $\bar{\boldsymbol{P}}$, where along x-axes are presented indexes of components. a) N=4, b) N=5, c) N=6.

a kind of dependence among different outputs of the TRNG exists. On the other hand, because of the symmetry of the scheme, the output signal of any unit has the uniform distribution.

## 4. Implementation of ternary generator

The standard implementation of the generator is presented in Figure 4. Clock line denotes a series of impulses with the interval $D$. One must choose $D$ as long as the time could be enough for the TRNG to transfer to a stable state. In a real situation, one cannot get the stable distribution in a finite time; so only an approximation of the stable distribution can be gained.

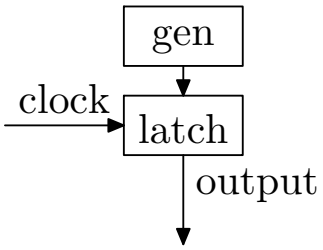


**Figure 4.** Implementation. Here gen is TRNG, whose outputs are controlled by clock.

Small instability of D does not influent on that distribution. To this end, we must evaluate the difference between $M' \times M'$ stable matrix $Stab$ of form

$$Stab = (\bar{\boldsymbol{P}}, \bar{\boldsymbol{P}}, \ldots, \bar{\boldsymbol{P}})$$

and matrix $Astab(D) = \exp(D \cdot Matr)$. Let $\boldsymbol{E} = \langle 1, 1, 1, \ldots, 1 \rangle$. Accordingly the definition, $\boldsymbol{E} \cdot Matr = \Theta = \langle 0, 0 \ldots, 0 \rangle$; hence $\boldsymbol{E} \cdot Matr^k = \Theta$ for any natural $k$, and $\boldsymbol{E} \cdot Astab(D) = \boldsymbol{E}$. In other words, the sum of entries in any column of $Stab(D)$ equals 1. Let

$$\delta = \max_k |\bar{\boldsymbol{P}} - Astab(D)[*, k]|, \tag{12}$$

where for vector $\boldsymbol{q} = \langle a_1, \ldots, a_n \rangle$ $|\boldsymbol{q}| = \max_k |a_k|$. Choose $\delta$ as a distance between two matrices. Some results of calculation are presented in Table 2.

**Table 2.** Distance between stable matrix and $Astab(D)$ depending on $D$.

| $N \backslash D$ | 2 | 4 | 8 | 16 |
|---|---|---|---|---|
| 1 | 3.1e-2 | 1.6e-3 | 3.9e-6 | 2.1e-7 |
| 2 | 4.6e-2 | 6.1e-3 | 1.1e-4 | 3.8e-8 |
| 3 | 3.6e-2 | 6.9e-3 | 2.4e-4 | 3.2e-7 |
| 4 | 2.1e-2 | 3.7e-3 | 1.4e-4 | 2.6e-7 |
| 5 | 9.8e-3 | 2.1e-3 | 9.2e-5 | 2.0e-7 |
| 6 | 6.0e-3 | 9.4e-4 | 3.6e-5 | 7.1e-8 |

As was mentioned above, a drawback of the suggested TRNG is absence of three special signals of form $\langle x, x, \ldots, x \rangle$, $x \in \{-1, 0, 1\}$ which are not released on outputs of the schema. One can suggest two methods to overcome that drawback. The first one is usage of a set of independent TRNGs (Figure 5). This scheme produces uniformly distributed signals.

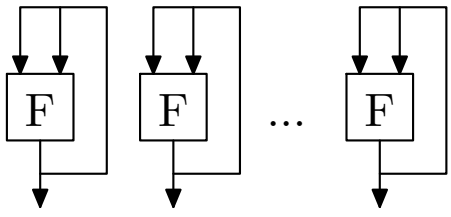


**Figure 5.** Generator on base of independent units.

In Figure 5 each TRNG contains a single unit, but one can use a TRNG having arbitrary number of units in such an implementation, although output of solely one unit will be used in

this case. There is another solution of the problem concerning the absence of the three special states at outputs of the TRNG. Suppose we have a scheme, presented in Figure 6. The output of unit 0 is omitted. Because of cyclic structure of the TRNG, any of the outputs can be omitted, but the distribution of the output signals will be the same. As a result, any ternary vector can arise at outputs of the TRNG, and the distribution of these vectors can be found on base of the previous calculations. For example, for $N = 3$, if only two outputs are used, one finds that the probabilities are (see (11))

(i) states $\langle x, x\rangle$ 0.074,
(ii) states $\langle 0, 1\rangle, \langle -1, 0\rangle, \langle 1, -1\rangle$ 0.148,
(iii) all other states — 0.111.

So the distribution of states is a bit closer to the uniform distribution than in the original example.

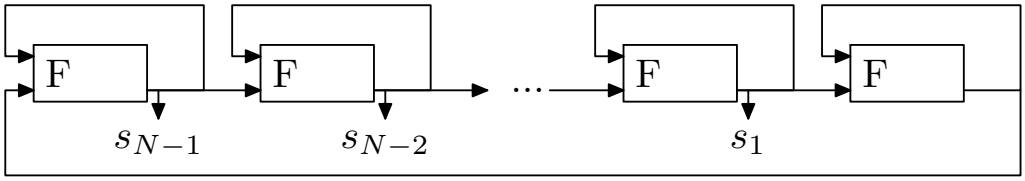


**Figure 6.** Generator with omitted output.

## 5. Conclusion

The supposition that the delay times in different units are independent events is intrinsic, while the restriction on the form of the distribution is much stronger. Because of the symmetry of the scheme, if the delay time for all units has the same distribution, then the output of each unit will be uniform regardless of the distribution. Nevertheless, the existence of the stable distribution and its shape, if exists, need additional research.

## Appendices

### Proof of Proposition 1

Let $F(a, b)$ meet (2) and $F(0, 0) = c_1, F(1, 1) = c_2, F(-1, -1) = c_3$. There are the following alternatives:

(i) $c_1, c_2, c_3$ is a permutation of $0, 1, -1$
(ii) exactly two elements in the set $c_1, c_2, c_3$ equal each other.

Consider the first of cases. Let $c_1 = 1$. Since all $c_i$ are different and $c_3 \neq -1$, we have $c_3 = 0$. So $F = F_1$. If $c_1 = -1$ then $c_2 = 0$ and $c_3 = 1$. Using the permutation $\sigma(0) = 0, \sigma(1) = -1, \sigma(-1) = 1$, one converts $F$ to $F_1$.

Now let us turn to the second of cases. Consider various possible distributions of values for $c_1, c_2, c_3$.

(i) $c_1 = 1, c_2 = 0, c_3 = 1$. Permutation $\sigma(0) = -1, \sigma(1) = 1, \sigma(-1) = 0$ converts $F$ to $F_2$.
(ii) $c_1 = 1, c_2 = 0, c_3 = 0$. Permutation $\sigma(0) = 1, \sigma(1) = -1, \sigma(-1) = 0$ converts $F$ to $F_2$.

If $c_1 = -1$, then the permutation $\sigma(0) = 0, \sigma(1) = -1, \sigma(-1) = 1$ reduces the deal to the previous situation.

**Proof of Proposition 2**

First of all, show that $\mathbf{S} = \langle x, x, \ldots, x\rangle$ is an unattainable state. Since only one output can change during transfer, any candidate state for transfer to $S$ must have form $\langle x, x, \ldots, x, y, \ldots, x\rangle$. Any unit in the circuit has at its inputs either $\angle x, x\rangle$, or $\angle x, y\rangle$, or $\angle y, x\rangle$. In all of the above cases, the unit can change its output to $x$ accordingly the definition of $F$. On the other hand, if $\mathbf{S} = \langle \ldots, x, y, \ldots\rangle$, where $x \neq y$, then the state $\langle \ldots, x, z, \ldots\rangle$, where $z \neq x$ and $z \neq y$, can transfer to $\mathbf{S}$.

**Proof of Proposition 3**

Accordingly the definition, the item $Q[i, j]$ in row with the number $i$ and the column with the number $j$ equals to 1 if and only if the generator can transfer from a state with number $j$ to the state with number $i$. Since the generator has $N$ units, each of them can change its output, and all the new states are different ones. It follows that for each $j$

$$\sum_i Q[i, j] = N \tag{13}$$

It means that $Q^T/N$ is a stochastic matrix, hence we have $e_1 = N$, and the inequality (8) holds.

**References**

[1] Asmussen S and Glynn P W 2007 *Stochastic Simulation: Algorithms and Analysis* ( New York: Springer Verlag) p 476
[2] Ferguson N, Schneie B, and Kohno T 2010 *Cryptography Engineering: Design Principles and Practical Applications* (Indianapolis: Wiley) p 384
[3] Horowitz P and Hill W 1980 *The art of electronics* (Cambridge :Cambridge University Press) p 1125
[4] Petrie C S and Connelly J A 1996 A noise-based IC random number generator for applications in cryptography *Proc. IEEE Int.Symp. Circuits and Systems(Atlanta)* vol 4 (New York: IEEE Press) pp 324-327
[5] Golic J D 2006 New Methods for Digital Generation and Postprocessing of Random Data *IEEE Trans.Comput.* **55** 1217-29
[6] Sunar B, Martin W J and Stinson D R 2007 A provably secure true random number generator with built-in tolerance to active attacks *IEEE Trans.Comput.* **56** 109-19
[7] Kuznetsov V, Pesoshin V and Stolov E 2008 Markov model of a digital stochastic generator *Automation and Remote Control* **69** 1504-09
[8] Wieczorek P and Golofit K 2014 Dual-Metastability Time-Competitive True Random Number Generator *IEEE Trans.Circuits and Systems* **61** 134-45
[9] Wu X W and Prosser F P 1990 CMOS ternary logic circuits *IEE Proc.Circuits, Devices and Systems* **137** 21-27
[10] Gaikwad V N and Deshmukh P R 2015 Design of CMOS ternary logic family based on single supply voltage *Proc.IEEE Int.Conf.on Pervasive Computing (Sydney)* vol 9 (New York: IEEE Press) pp 1-6
[11] Lisa N J and Babu H H 2015 Design of a Compact Ternary Parallel Adder/Subtractor Circuit in Quantum Computing *Proc.IEEE Int.Symp.on Circuits and Systems (Lisbon)* vol 28 (New York: IEEE Press) pp 2145-2148
[12] Kleinrock L 1975 *Queueing Systems: Volume I Theory* (New York: Wiley-Interscience) p 417
[13] Marcus M and Mink H 1964 *A survey of matrix theory and matrix inequalities* (Boston: Allys and Bacon) p 232
[14] Bellman R 1960 *Introduction to matrix analysis* ( New York: Macgrow-Hill) p 365