Circuits Syst Signal Process (2016) 35:3830–3846
DOI 10.1007/s00034-016-0248-0

A New Approach to Analysis and Design of Chaos-Based Random Number Generators Using Algorithmic Converter

Esmaeil Fatemi-Behbahani[1] · **Karim Ansari-Asl**[1] · **Ebrahim Farshidi**[1]

Received: 11 June 2015 / Revised: 6 January 2016 / Accepted: 7 January 2016 /
Published online: 2 February 2016

Abstract This paper presents a new approach to analysis and design of ADC-based random number generators. To this end, different full-bit and half-bit redundant stages of algorithmic converter are used to design chaotic maps. It is shown that, in the redundant and nonredundant structures, output probability density function of the converter stages and their related chaotic functions always converge to uniformity. It is demonstrated that residues become independent and uniformly distributed. This fact leads to the randomness and uniformity of distribution of the random number generator output bits. Moreover, it is shown that some common chaotic maps that are employed in chaotic random number generators can be implemented using nonredundant and half-bit redundant stages of algorithmic converter. In this way, the capability of ADC-based generators in designing chaotic maps and producing random number sequences is illustrated. The validity of the proposed chaos-based random number generator is confirmed using NIST statistical tests even in the presence of nonidealities in algorithmic converter. Since the ADCs are mixed-signal integrated circuits and can be used in high-speed applications, the ADC-based random number generator has high throughput and is easily embeddable in all analog and digital circuits.

Keywords Algorithmic ADC · Chaotic map · Half-bit redundancy · Probability density function (pdf) · Random number generator · Uniform distribution

✉ Ebrahim Farshidi
farshidi@scu.ac.ir

Esmaeil Fatemi-Behbahani
e-fatemi@phdstu.scu.ac.ir

Karim Ansari-Asl
karim.ansari@scu.ac.ir

[1] Department of Electrical Engineering, Faculty of Engineering,
Shahid Chamran University of Ahvaz, Ahvaz, Iran

1 Introduction

Multistage analog-to-digital converters (ADCs) such as pipelined, algorithmic (cyclic), and successive approximation register (SAR) are widely used in medium-to-high-resolution and bandwidth applications. In these architectures, the analog-to-digital conversion is performed by using one or several successive simple stages in a way that makes it possible to compromise between speed, power consumption, resolution, and circuit area. Among all kinds of ADCs, algorithmic converter, due to the cyclic nature of its conversion process, needs smaller area and lower power and is preferred for some applications [14,17].

One of the very important issues in analog-to-digital converters is the quantization noise theory. Based on this theory, if an ADC is modeled as a uniform quantizer, without considering its internal structure, the output quantization noise is almost white with uniform distribution [27,31]. Few studies have been carried out on the analysis of quantization noise specified for multistage converters. In [8,16], it has been shown that with the passage of signal from different stages of a full-bit and half-bit redundant pipelined converter, the residue probability density function (pdf) and the residue joint pdf at different times converge to uniformity. The obtained results reveal that in a redundant pipeline ADC, the last-stage residue distribution is uniform but does not cover the full converter dynamic range. The same feature exists for the algorithmic converters with full-bit and half-bit redundant structures. These properties allow us to use them as random number generators (RNGs) and produce independent and identically distributed random bit sequences.

Random number generators have numerous applications in symmetric- and public-key cryptography algorithms [1,7], communication systems [6] as well as calibration of algorithmic and pipelined ADCs by dither injection [13]. RNGs can be divided into two categories: pseudorandom number generators (PRNGs) and true random number generators (TRNGs) [10,23]. PRNGs have high throughput and because of their intrinsic nature can be easily embedded in any digital circuit or system. However, because of their deterministic and finite memory algorithms, they have periodic behavior and are far from the ideal features required in some important applications such as information security and cryptography [12,28]. Accordingly, TRNGs which are called physical generators are preferred to be used for high-end security applications [12]. They are usually implemented with a combination of three blocks: entropy source, harvesting mechanism, and postprocessing, and physical random processes are used as their entropy sources [21,32].

Low throughput [4] and high cost of embedding physical RNGs in digital circuits [1,7] have led the attention of recent researches to the sources that have both features of the physical RNGs and the simplicity of digital sources simultaneously. Therefore, chaos-based RNGs which use chaotic maps have found widespread applications [18,21,29]. Some of these chaotic maps are very similar to the building blocks of practical electronic circuits. For instance, the input–output characteristic of the half-bit redundant stage which has been exploited in designing pipelined ADC in [26] is fully equivalent to the chaotic map that has been used in [6,29]. For this reason, in [1,21], this 1.5-bit stage was employed to implement an ADC- based chaotic RNG. Although today this characteristic is not commonly used in multistage ADCs, it is shown that popular used full-bit and half-bit redundant stages are also efficient to design chaotic maps. Since the ADCs are mixed-signal integrated circuits and can be used in high speeds [11], the ADC-based random number generator has high throughput and is easily embeddable in all analog and digital circuits.

In this paper, the input–output characteristics of the various stages of the algorithmic ADC are compared with some of the common chaotic maps that are used in chaos-based RNGs. It is shown that they are fully similar to each other. One of these chaotic maps is Bernoulli map [3,12,30], which fully matches with the characteristic of the ideal 1-bit stage. The characteristic of the full k-bit stage is also fully identical to the N-way Bernoulli shift map with $N = 2^k$ [25], which is the more general form of the Bernoulli map. Furthermore, using the fundamental theorem for function of a random variable [20], a new approach to analyze the output of chaotic maps as well as the different cycle residues of algorithmic ADC is presented. To this end, the propagation of the output pdf of the chaotic maps in different cycles of the algorithmic ADC is studied. It is demonstrated that regarding the random nature and the uniform distribution of the nonredundant algorithmic ADC output bits, this converter can be used to implement N-way Bernoulli shift map and generate random sequences. For the half-bit redundant algorithmic ADC, the residue pdf converges to uniformity in the center half of the stage full-scale range and out of it converges to zero. Thus, after a sufficiently large number of cycles, each stage will be fully equivalent to the common chaotic map which has been used in [6,29]. The performance of the proposed ADC-based RNG is evaluated by using the US National Institute for Standards and Technology (NIST) randomness test suite [24]. Since analog-to-digital converters are sensitive to device parameters' variations [5,11,22], these nonidealities are also considered. Test results show that algorithmic-based RNG successfully passes all NIST 800-22 statistical tests in the presence of mismatches.

The rest of the paper is organized as follows. In Sect. 2, some of the common chaotic maps that are very similar to the characteristic of ADC building blocks are investigated. Section 3 introduces the algorithmic ADC architecture. In Sect. 4, the use of full-bit algorithmic ADC, and in Sect. 5, the application of half-bit redundant converter in random bit generation is explored. Section 6 evaluates the performance of the proposed RNG using the NIST statistical test suite, and a brief conclusion is drawn in Sect. 7.

2 Chaos-Based RNGs

Random sequence generation can be modeled by the toss of an ideal coin, which the probability of each side is 1/2. Such tosses are independent from each other, and seeing each toss does not affect the probability of the next tosses observations. That is, the system state cannot be predicted. This behavior can be described as the two-state Markov chain of Fig. 1, which is a special kind of Markov processes. Several chaotic maps were presented for this Markov chain. One of them is Bernoulli map which is expressed as follows [12,30]:

$$M : [-1, 1] \to [-1, 1], \quad M(x) = \mu x \bmod 2 - 1 \tag{1}$$

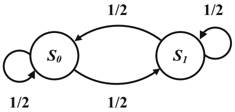


Fig. 1 Markov chain of the fair coin toss

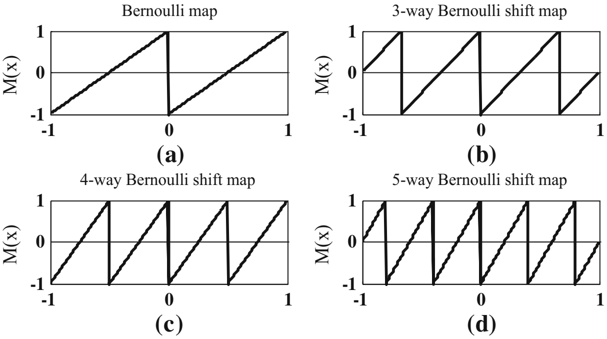


Fig. 2 N-way Bernoulli shift map for different values of N

where $0 < \mu \leq 2$ is the chaos control parameter and the maximum entropy of the source is obtained by $\mu = 2$. Equation (1) can be expressed as follows:

$$M : [-1, 1] \to [-1, 1], \quad M(x) = \begin{cases} \mu x + 1 & -1 < x < 0 \\ \mu x - 1 & 0 < x < 1. \end{cases} \tag{2}$$

Assuming $\mu = 2$, Bernoulli shift map fully matches with the 1-bit stage input–output characteristic of algorithmic converter. In Sect. 4, it will be shown that after sufficient iterations of this map, the distribution of the x-samples becomes uniform over $[-1, 1]$. In this way, the Bernoulli map and 1-bit stage of algorithmic ADC are exactly equivalent to Markov chain of Fig. 1 and knowing the previous sequences reveals no information about its future values. Also, since the probability of being in each of two states equals 1/2, the entropy of the source is one bit.

The more general form of (1) and (2) is N-way Bernoulli shift map [25], which can be expressed as

$$M : [-1, 1] \to [-1, 1], \quad M(x) = Nx \bmod 2 - 1. \tag{3}$$

The N-way Bernoulli shift map that is widely used in signal- processing tasks is shown in Fig. 2 for different values of N. Since this map can be modeled by two-state Markov chain of Fig. 1 [1,21], knowing the previous sequences reveals no information about its future values.

Another chaotic map that has been found to have a good performance in RNGs is according to the following expression [6,29]:

$$M : [-1, 1] \to [-1, 1], \quad M(x) = (2x + 1) \bmod 2 - 1. \tag{4}$$

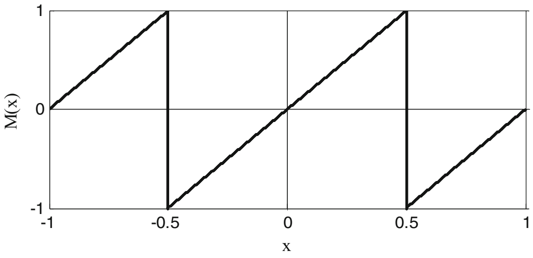


Fig. 3 The chaotic map (5)

This map that is shown in Fig. 3 can be expressed as:

$$M:[-1,1]\rightarrow[-1,1],\quad M(x)=\begin{cases}2x+2 & -1<x<-\dfrac{1}{2}\\ 2x & -\dfrac{1}{2}<x<\dfrac{1}{2}\\ 2x-2 & \dfrac{1}{2}<x<1.\end{cases} \tag{5}$$

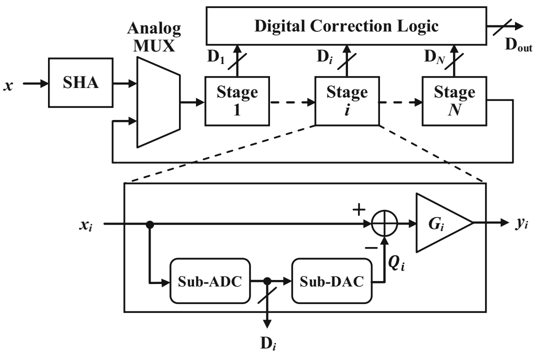
This chaotic map is identical to the characteristic of the 1.5-bit stage employed in [26] for pipelined ADC which uses the half-bit redundancy in order to achieve the related advantages. An important feature of this map compared with Bernoulli map is its flexibility against noise and nonideality impacts on electronic circuits in a way that it has all the necessary conditions for a piecewise affine Markov chaotic map [21]. In this map, the distribution of x-samples over $[-1, 1]$ converges to uniformity. It can be shown that from the statistical point of view and after sufficient iterations, the behavior of the map can be illustrated exactly as the Bernoulli process and two-state Markov chain of Fig. 1 [1].

3 Algorithmic ADC Architecture

The architecture of an algorithmic analog-to-digital converter is shown in Fig. 4. This ADC consists of an input sample-and-hold amplifier (SHA) and one or several number of successive simple stages. All signals are normalized to V_{ref}, so the converter dynamic range is $[-1, 1]$. The SHA converts the continuous-time input signal x_{in} into a sampled sequence $x(k) = x_{\text{in}}\ (kT_{\text{s}}\ )$, where T_{s} denotes the sampling period. Each algorithmic stage consists of a flash sub-ADC, a sub-DAC, a subtractor, and an interstage amplifier. In the ith stage, the flash sub-ADC generates an m_i-bit digital estimation D_i of the stage input x_i and the sub-DAC converts this digital word to an analog signal. Then, the difference between the sampled signal x_i and its quantized version (Q_i) is amplified with the interstage gain G_i to produce a residue signal y_i which is used as the input x_{i+1} to the next stage. The algorithmic converter extracts the required number of bits in several clock cycles and by returning back to a sequential approach.

In the algorithmic converter, input–output characteristic of each stage is according to the following expression:

Fig. 4 Algorithmic ADC architecture



$$x_i = \frac{1}{G_i} x_{i+1} + Q_i. \tag{6}$$

Assuming the input signal is passed sequentially through N cycles, it can be written as [13,16]

$$x = \sum_{i=1}^{N} \left(\prod_{j=1}^{i-1} \frac{1}{G_j} \right) Q_i + e, \tag{7}$$

where e is the quantization error and

$$e = \left(\prod_{i=1}^{N-1} \frac{1}{G_i} \right) x_N. \tag{8}$$

If there are no other error sources than the quantization noise, the converter signal-to-quantization-noise ratio (SQNR) is obtained by:

$$\text{SQNR} = 10 \log \left(\frac{\overline{x^2}}{\overline{\mathrm{e}^2}} \right) = 10 \log \left(\overline{x^2} \right) - 10 \log \left(\overline{\mathrm{e}^2} \right). \tag{9}$$

By applying a full-scale sinusoidal input, the converter effective number of bits (ENOBs) can be calculated by [19]:

$$\text{ENOB} = \frac{\text{SQNR} - 1.76}{6.02}. \tag{10}$$

In terms of the structure and function of the algorithmic ADC and according to (8), by increasing the number of cycles, the quantization error can be reduced to any desired level. Thus, the SQNR and ENOB of this converter can be theoretically increased to any arbitrary level. But, in an analog circuit, the initial conditions cannot be determined with infinite precision and the impact of noise is inevitable. By applying one initial condition to the circuit, and after too many cycles that the high significant bits are extracted, the low significant bits indicate the noise value, which affects the initial conditions, and thus, the output bits become quite random. So, it is sufficient to throw away the extracted bits of the first cycles to see an exactly unpredictable behavior in the circuit of Fig. 4.

One important property of the algorithmic ADC is converting analog to digital by the aid of a series redundant or nonredundant successive simple stages. This property makes it possible to map each stage input pdf to its output pdf and use this feature for easier analysis of the noise distribution. To this end, the fundamental theorem for function of a random variable [20] can be exploited and the propagation of the residual pdf in successive stages with full-bit and half-bit redundant architectures can be shown [8]. In the algorithmic ADC architecture (Fig. 4), each stage might be redundant or nonredundant. Firstly, the analysis of the residue pdf in ideal nonredundant stages is investigated. This is exactly the same as analysis of the Bernoulli shift chaotic map. It will be shown that after sufficient iterations the system will be fully equivalent to the Markov chain of Fig. 1. This fact indicates that the system can be used as a chaos-based RNG.

4 RNG Using Full-Bit Algorithmic ADC Structure

In 1-bit/stage algorithmic ADC, $G_i = 2$ and (6) can be expressed as

$$x_{i+1} = y_i = \begin{cases} 2\left(x_i + \frac{1}{2}\right) & -1 < x_i < 0 \ (D_i = 0) \\ 2\left(x_i - \frac{1}{2}\right) & 0 < x_i < 1 \quad (D_i = 1)\,. \end{cases} \tag{11}$$

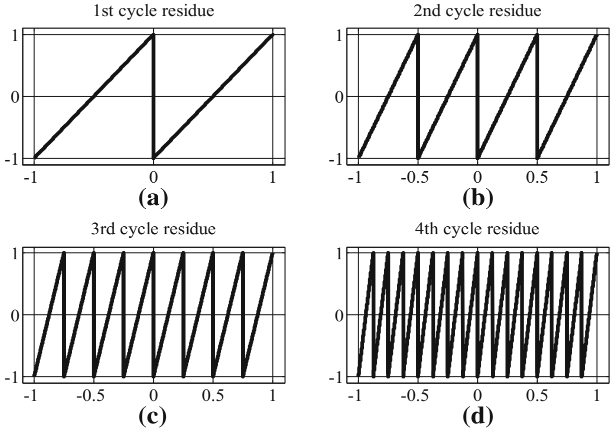
This characteristic is fully matched to the Bernoulli map with $\mu = 2$, which was described in (2) and depicted in Fig. 2a. Thus, the algorithmic ADC structure can be used to implement and iterate the Bernoulli shift map.

The residue transfer characteristic of the different cycles in a 1-bit/cycle converter is shown in Fig. 5. Comparing Figs. 2 and 5 shows that two cycles of the converter is same as the four-way Bernoulli shift map and, in general, N cycles of the algorithmic ADC are exactly identical to M-way Bernoulli shift map where $M = 2^N$.

In [8], using the fundamental theorem for function of a random variable, it has been shown that for the input pdf $f_x(x)$, the converter Nth cycle output pdf is obtained by

$$f_{y_N} = \begin{cases} \frac{1}{M} \sum_{m=0}^{M-1} f_x\left(\frac{y - (2m + 1 - M)}{M}\right) & -1 < y < +1 \\ 0 & \text{otherwise,} \end{cases} \tag{12}$$

where $M = 2^N$. It is clear that if the input pdf is uniform over $[-1, 1]$, then the first cycle output pdf and, as a result, the output distributions of all the subsequent cycles are uniform. Therefore, N-way Bernoulli shift map does not change the pdf of an input with uniform density.



(a) (b) (c) (d)

Fig. 5 The different cycle residues in a 1-bit/cycle algorithmic converter

Since $f_x(x)$ and $f_{y_N}(y)$ are nonzero only over $[-1, +1]$, they can be extended into periodic signals $\tilde{f}_x(x)$ and $\tilde{f}_{y_N}(y)$ with period 2 and Fourier series coefficients a_k and b_k, respectively. The Fourier series representation of these periodic signals can be written as

$$\tilde{f}_x(x) = \sum_{k=-\infty}^{+\infty} a_k \mathrm{e}^{j\pi kx} \Leftrightarrow a_k = \frac{1}{2}\int_{-1}^{+1} f_x(x)\mathrm{e}^{-j\pi kx}\mathrm{d}x \tag{13}$$

and

$$\tilde{f}_{y_N}(y) = \sum_{k=-\infty}^{+\infty} b_k \mathrm{e}^{j\pi ky} \Leftrightarrow b_k = \frac{1}{2}\int_{-1}^{+1} f_{y_N}(y)\mathrm{e}^{-j\pi ky}\mathrm{d}y. \tag{14}$$

By substituting (12) in (14), b_k can be calculated as

$$\begin{aligned} b_k &= \frac{1}{2M}\int_{-1}^{+1}\sum_{m=0}^{M-1} f_x\left(\frac{y-(2m+1-M)}{M}\right)\mathrm{e}^{-j\pi ky}\mathrm{d}y \\ &= \frac{1}{2}\sum_{m=0}^{M-1}\left(\int_{1-\frac{2(m+1)}{M}}^{1-\frac{2m}{M}} f_x(x)\mathrm{e}^{-j\pi Mkx}\mathrm{d}x\right)\mathrm{e}^{-j\pi k(2m+1-M)}. \end{aligned} \tag{15}$$

As regards $\mathrm{e}^{-j\pi k(2m+1-M)} = (-1)^k$

$$\begin{aligned} b_k &= \frac{(-1)^k}{2}\sum_{m=0}^{M-1}\int_{1-\frac{2(m+1)}{M}}^{1-\frac{2m}{M}} f_x(x)\mathrm{e}^{-j\pi Mkx}\mathrm{d}x \\ &= \frac{(-1)^k}{2}\int_{-1}^{+1} f_x(x)\mathrm{e}^{-j\pi Mkx}\mathrm{d}x. \end{aligned} \tag{16}$$

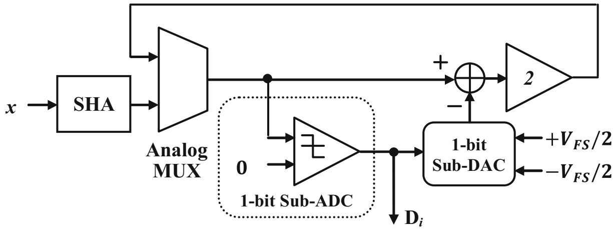


Fig. 6 RNG using 1-bit/cycle algorithmic ADC

Comparing (13) and (16) reveals that

$$b_k = (-1)^k a_{Mk}. \tag{17}$$

It can be observed that kth Fourier coefficient of $\tilde{f}_{y_N}(y)$ is equal to (Mk)th harmonic of $\tilde{f}_x(x)$. Hence, the last-stage residue pdf retains only the Fourier series coefficients which are integer multiples of M. So, after propagating through a sufficient number of stages, only the DC component

$$b_0 = a_0 = \frac{1}{2}\int_{-1}^{+1} f_x(x)\mathrm{d}x = \frac{1}{2} \tag{18}$$

is preserved and all harmonics are weeded out. It is clear that with increasing the total number of bits of the ADC (N), the last-stage residue pdf converges to the uniform distribution:

$$f_{y_N}(y) = \begin{cases} \frac{1}{2} & -1 < y < +1 \\ 0 & \text{otherwise.} \end{cases} \tag{19}$$

In this way, the probability of the Nth stage output bit is $\Pr(D_N = 0) = \Pr(D_N = 1) = 1/2$, which indicates the uniform distribution of the output streams. After this, each stage of the generator acts like the Markov chain of Fig. 1, which can be directly used to implement ideal RNG. In order to implement this RNG, it is enough to consider a 1-bit/cycle algorithmic ADC without its digital correction logic according to Fig. 6, and let it run for infinite number of bits. This is equivalent to iterate the Bernoulli chaotic map $x_{n+1} = M(x_n)$ that the output bit determines the RNG output and the circuit state.

To illustrate the results, a simulation was performed for the 1-bit/cycle algorithmic ADC of Fig. 6. Initial condition was set to $0.25\,V_{\mathrm{FS}}$ and noise to $0.001\,V_{\mathrm{FS}}$. The converter residual pdfs after different cycles are shown in Fig. 7. It can be seen that after many cycles, residue pdf converges to uniformity. So, after passing less than 18 cycles, the system will be exactly equivalent to the Markov chain of Fig. 1. Two runs of the system starting at the same initial condition with a 0.1-mV noise floor, which has led to different trajectories, are shown in Fig. 8. It can be observed that by throwing less than 18 first samples of the output sequence, the system will have a quite random

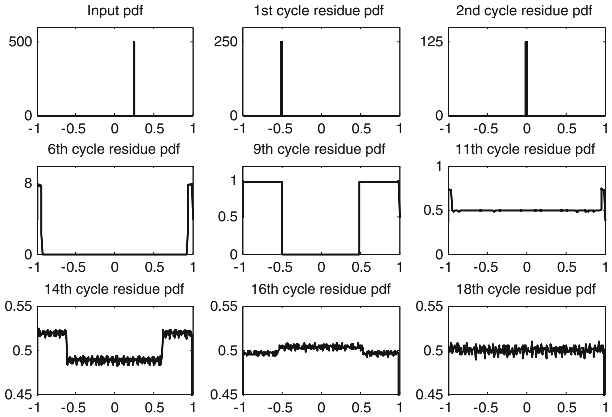


Fig. 7 The 1-bit/cycle converter residue pdfs after different cycles. Initial condition was set to $0.25V_{\mathrm{FS}}$ and noise floor to $0.001V_{\mathrm{FS}}$

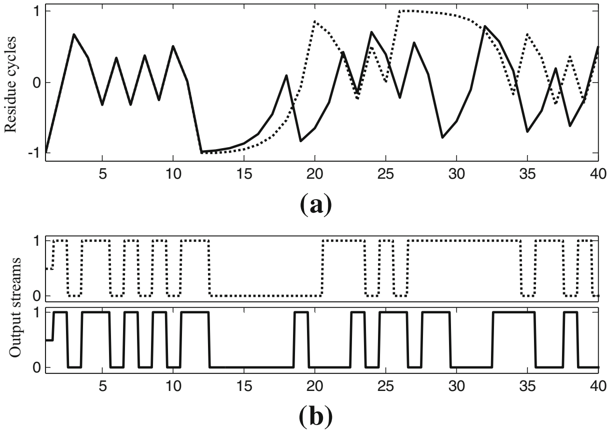


Fig. 8 Twice running the RNG of Fig. 6. **a** Residue cycles. **b** Output streams

and unpredictable output bitstream. Running this system for a thousand sequences of length 10,000 at different times, the entropy of the output sequences is shown in Fig. 9. As it is expected, this RNG has one bit of entropy.

The input–output characteristic of a k-bit stage algorithmic ADC is identical to $k \times$ 1-bit stages (Fig. 6). This characteristic is equivalent to the 2^k-way Bernoulli shift map and can be used to implement this map. In [8,16], it has been shown that the impact of each k-bit stage on input pdf is the same as to $k \times$ 1-bit stages. Consequently, in an algorithmic converter with a k-bit stage, every cycle is exactly identical to one run of 2^k-way Bernoulli shift map. Thus, by increasing the number of converter cycles, residue probability density function converges to uniform distribution of (19).

Small variations in Bernoulli map parameters can bring about stable equilibrium points in the system; thus, this map is not suitable for the electronic implementation [21]. In fact, the impact of noise and nonidealities of the electronic circuits on Bernoulli

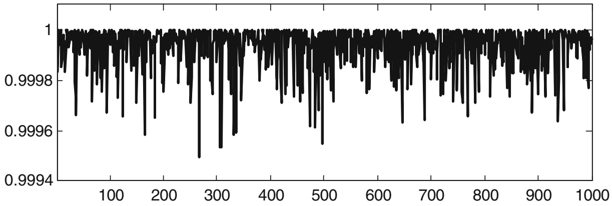


Fig. 9 The output sequence entropy of Fig. 6

map is similar to the nonidealities impacts on the algorithmic full-bit stage characteristic, which there is not enough safety against them. So, redundancy is employed in such stages.

5 RNG Using Half-Bit Redundant Algorithmic ADC

Half-bit redundancy is widely used in the structure of multistage converters. Using redundancy in multistage ADCs, in addition to increasing the speed and decreasing the circuit power [15], can make converters less sensitive against the elements nonidealities impacts and environmental mismatches [2,9].

For an ideal 1.5-bit/cycle algorithmic ADC, the stage input–output characteristic is according to the following expression:

$$x_{i+1} = y_i = \begin{cases} 2\left(x_i + \frac{1}{2}\right) & -1 < x_i < -\frac{1}{4} \quad \left(d_{i_1} d_{i_2} = 00\right) \\ 2x_i & -\frac{1}{4} < x_i < \frac{1}{4} \quad \left(d_{i_1} d_{i_2} = 01\right) \\ 2\left(x_i - \frac{1}{2}\right) & \frac{1}{4} < x_i < 1 \quad \left(d_{i_1} d_{i_2} = 10\right). \end{cases} \tag{20}$$

In [18], it has been shown that for the input pdf $f_x(x)$, the converter Nth cycle output pdf is obtained by:

$$f_{y_N} = \begin{cases} \frac{1}{2^N} f_x\left(\frac{y-(2^N-1)}{2^N}\right) & -1 < y < -\frac{1}{2} \\ \frac{1}{2^N} \sum_{m=-(2^N-1)}^{(2^N-1)} f_x\left(\frac{y-m}{2^N}\right) & -\frac{1}{2} < y < \frac{1}{2} \\ \frac{1}{2^N} f_x\left(\frac{y+(2^N-1)}{2^N}\right) & \frac{1}{2} < y < 1. \end{cases} \tag{21}$$

It is observed that with increasing the total number of bits (N), the output pdf is more concentrated in center half of the stage full-scale range. Since after a sufficient number of stages, $f_{y_N}(y)$ is fully concentrated over $[-1/2, 1/2]$, by repeating the middle half of the last-stage full-scale range it can be extended into a periodic signal $\hat{f}_{y_N}(y)$ with period 1 and Fourier series coefficients c_k

$$\hat{f}_{y_N}(y) = \sum_{k=-\infty}^{+\infty} c_k \mathrm{e}^{j2\pi ky} \Leftrightarrow c_k = \int_{-1/2}^{+1/2} f_{y_N}(y)\mathrm{e}^{-j2\pi ky}\mathrm{d}y. \tag{22}$$

By substituting (21) in (22), c_k can be calculated as

$$\begin{aligned} c_k &= \frac{1}{M}\int_{-1/2}^{+1/2} \sum_{m=-(M-1)}^{M-1} f_x\left(\frac{y-m}{M}\right)\mathrm{e}^{-j2\pi ky}\mathrm{d}y \\ &= \sum_{m=-(M-1)}^{M-1}\left(\int_{-\frac{m}{M}-\frac{1}{2M}}^{-\frac{m}{M}+\frac{1}{2M}} f_x(x)\mathrm{e}^{-j2\pi Mkx}\mathrm{d}x\right)\mathrm{e}^{-j2\pi mk} \end{aligned} \tag{23}$$

where $M = 2^N$. As regards $\mathrm{e}^{-j2\pi mk} = 1$

$$c_k = \int_{-1+\frac{1}{2M}}^{+1-\frac{1}{2M}} f_x(x)\mathrm{e}^{-j2\pi Mkx}\mathrm{d}x. \tag{24}$$

Comparing (13) and (24) reveals that with increasing the total number of bits of the converter (N), c_k converges to $2a_{2Mk}$. It means that the last-stage residue pdf retains only the Fourier series coefficients which are integer multiples of $2M$. So, after propagating through a sufficient number of stages, only the DC component

$$c_0 = 2a_0 = \int_{-1}^{+1} f_x(x)\mathrm{d}x = 1 \tag{25}$$

is preserved and all harmonics are weeded out. Therefore, after many number of cycles, residue pdf converges to the uniform distribution:

$$f_{y_N} = \begin{cases} 1 & -\frac{1}{2} < y < \frac{1}{2} \\ 0 & \text{otherwise} \end{cases}. \tag{26}$$

In order to study whether the characteristic of (20) can be used for RNG, the interval partition $X_0 = [-1, -1/2)$, $X_1 = [-1/2, -1/4)$, $X_2 = [-1/4, 0)$, $X_3 = [0, 1/4)$, $X_4 = [1/4, 1/2)$, $X_5 = [1/2, 1]$ of $[-1, 1]$ is considered and the state x_i is defined as $x \in X_i$. Since the distribution of the x-samples, after passing sufficient cycles becomes uniform and limited over $[-1/2, 1/2]$, the probability of the x_0 and x_5 is 0 and other states have the probability 1/4. So, after sufficient cycles, the system will be fully equivalent to the chaotic map shown in Fig. 3. In [21], it has been shown that the evolution of this process can be expressed by the Markov chain of Fig. 10a. Since this chain has memory and its different states are not independent, it is not suitable for direct implementation of the RNG. To eliminate this drawback, the state aggregations

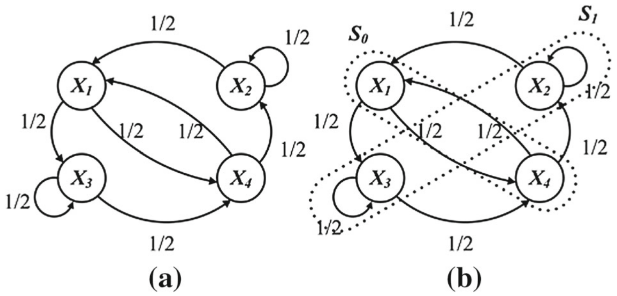


Fig. 10 **a** Markov chain of 1.5-bit stage, **b** related state aggregation

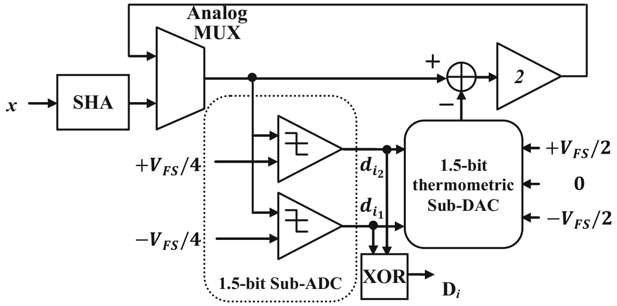


Fig. 11 RNG using 1.5-bit/cycle algorithmic ADC structure

of Fig. 10b can be exploited and an easier Markov chain with two macrostates S_0 and S_1 be obtained [7,21]. It is evident that the new Markov chain is equivalent to the two-state Markov chain of Fig. 1. By considering the thermometric digital coding for the 1.5-bit stage:

$$x_{i+1} = y_i = M(x_i) = \begin{cases} 2\left(x_i + \frac{1}{2}\right) & -1 < x_i < -\frac{1}{4} & \left(d_{i_1}d_{i_2} = 00\right) \\ 2x_i & -\frac{1}{4} < x_i < \frac{1}{4} & \left(d_{i_1}d_{i_2} = 01\right) \\ 2\left(x_i - \frac{1}{2}\right) & \frac{1}{4} < x_i < 1 & \left(d_{i_1}d_{i_2} = 11\right) \end{cases} \tag{27}$$

it is sufficient to calculate XOR of d_{i_1} and d_{i_2} to determine whether the system is in S_0 or S_1 [21].

To clarify the obtained results, a simulation was performed for the algorithmic converter of Fig. 11 including an ideal 1.5-bit stage. The converter residue pdfs after different cycles are shown in Fig. 12. It can be observed that after many cycles, the residue probability density function over [−1/2, 1/2] converges to uniformity. So, after passing less than 15 cycles, the system will be exactly equivalent to the Markov chain of Fig. 1. Two runs of the system starting at the same initial condition with a 0.1-mV noise floor, which has led to different trajectories, are shown in Fig. 13. It can

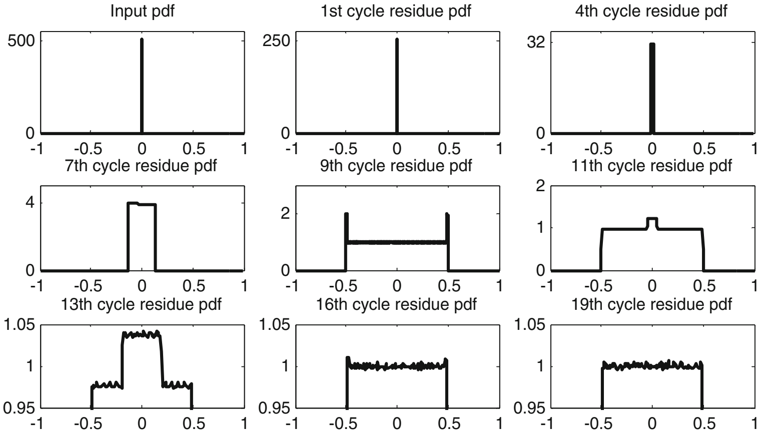


Fig. 12 The 1.5-bit/stage converter residue pdfs after different cycles. Initial condition was set to zero and noise floor to 0.001V_{FS}

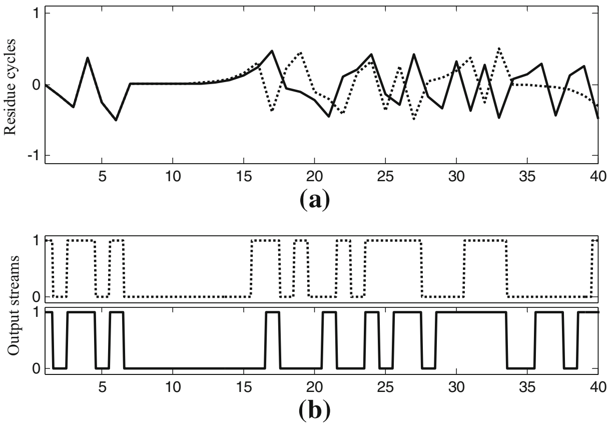


Fig. 13 Twice running RNG of Fig. 13. **a** Residue cycles. **b** Output streams

be seen that by throwing away less than 15 first samples of the output bitstream, a quite random and unpredictable sequence appears in the output. This system was run for a thousand sequences of length 10,000 at different times, which the output sequence entropies are shown in Fig. 14. As it is expected, this RNG has one bit of entropy.

6 Randomness Test Results

To evaluate the randomness of the proposed ADC-based RNG output bitstream using 1.5-bit stage of pipelined converter, the NIST test suit [24] is applied to the RNG output bit stream of Fig. 11. The statistical test suite v2.1.2, July 2014, which is the latest version available at the time of this study, is used for evaluation of the captured data. $\alpha = 0.01$ gives the set of p values as shown in Table 1. p value $\geq$0.01 means the

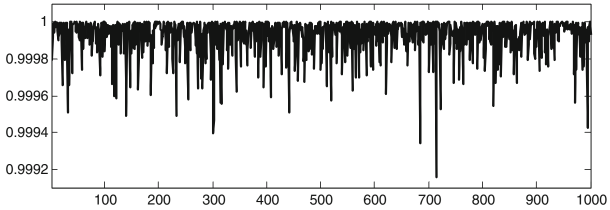


Fig. 14 The output sequence entropy of Fig. 13

Table 1 NIST test results for the ideal 1.5 bit/cycle

NIST SP800-22 tests	*p* value	Proportion	Result
Frequency	0.791422	0.9860	Pass
Block frequency	0.512530	0.9910	Pass
Cumulative sums	0.813214	0.9890	Pass
Runs	0.772305	0.9860	Pass
Longest run of ones	0.412538	0.9850	Pass
Rank	0.876172	0.9910	Pass
Spectral (DFT)	0.887691	0.9900	Pass
Nonperiodic template	0.780481	0.9920	Pass
Overlapping template	0.618628	0.9860	Pass
Universal statistical	0.659372	0.9880	Pass
Approx. entropy	0.553671	0.9840	Pass
Random excursions	0.491826	0.9920	Pass
Random exc. var.	0.574307	0.9860	Pass
Serial	0.459146	0.9930	Pass
Linear complexity	0.809135	0.9890	Pass

test is passed and p value ≥ 0.01 is interpreted as the test is failed [24]. The test results show that output random bitstream successfully passes all NIST 800-22 statistical tests without any postprocessing.

Since all ADCs are sensitive to device parameters variations, we take into account the inevitable circuit nonidealities such as comparator offset errors, capacitor mismatches and gain error that may go along with the ADC-based RNG of Fig. 11. Assuming these nonidealities, the RNG of Fig. 11 output bit stream was evaluated. A Monte Carlo simulation model was run for 1,000,000 length sequences. The NIST statistical test results are shown in Table 2. Test results show that algorithmic-based RNG successfully passes all NIST 800-22 statistical tests.

7 Conclusion

A new approach to analysis and design of chaos-based RNGs using ADC building blocks has been presented. Regarding the fact that multistage converters have theoretically infinite precision, the structure of algorithmic ADC was used to design random number generators. The input–output characteristics of the full-bit and half-bit redundant stages of algorithmic ADC were compared with different chaotic maps. It was

Table 2 NIST test results in the presence of nonidealities

NIST SP800-22 tests	p value	Proportion	Result
Frequency	0.706815	0.9850	Pass
Block frequency	0.513376	0.9840	Pass
Cumulative sums	0.735904	0.9870	Pass
Runs	0.659325	0.9900	Pass
Longest run of ones	0.490765	0.9850	Pass
Rank	0.561729	0.9910	Pass
Spectral (DFT)	0.783120	0.9830	Pass
Nonperiodic template	0.539106	0.9880	Pass
Overlapping template	0.458136	0.9900	Pass
Universal statistical	0.513914	0.9920	Pass
Approx. entropy	0.421976	0.9820	Pass
Random excursions	0.583054	0.9830	Pass
Random exc. var.	0.413716	0.9810	Pass
Serial	0.437819	0.9880	Pass
Linear complexity	0.516293	0.9930	Pass

found that 1-bit stage of this converter can be used to implement the Bernoulli map, and also the 2^k-way Bernoulli shift map can be implemented using $k \times$ 1-bit stages or one k-bit stage. It was revealed that in the half-bit redundant algorithmic ADC, after a sufficiently large number of cycles, residue pdf becomes concentrated in the center half of the stage full-scale range. In this way, the 1.5-bit stage characteristic will be fully equivalent to the common chaotic map that is employed to generate random number sequences. Therefore, this stage is suitable to implement chaotic RNG. With regard to the uniformly distributed and statistically independent residue signals of this converter at different times, it was found that the structure of multistage algorithmic converter is suitable to implement multi-bit chaos-based RNGs and is capable of being embedded to high-speed analog and digital circuits.

References

1. S. Callegari, R. Rovatti, G. Setti, Embeddable ADC-based true random number generator for cryptographic applications exploiting nonlinear signal processing and chaos. IEEE Trans. Signal Process. **53**(2), 793–805 (2005)
2. T. Cho, P.R. Gray, A 10 b, 20 Msamples, 35 mW pipeline AD converter. IEEE J. Solid State Circuits **30**(3), 166–172 (1995)
3. I. Cicek, A.E. Pusane, G. Dundar, A new dual entropy core true random number generator. Analog Integr. Circuits Signal Process. **81**(1), 61–70 (2014)
4. I. Cicek, A.E. Pusane, G. Dundar, A novel design method for discrete time chaos based true random number generators. Integra. VLSI J. **47**(1), 38–47 (2014)
5. P. Crippa, C. Turchetti, M. Conti, A statistical methodology for the design of high-performance CMOS current-steering digital-to-analog converters. EEE Trans. Comput. Aided Design Integr. Circuits Syst. **21**(4), 377–394 (2002)
6. M. Delgado-Restituto, A. Rodríguez-Vázquez, Mixed-signal map configurable integrated chaos generator for chaotic communications. IEEE Trans. Circuits Syst. I Fundam. Theory Appl. **48**(12), 1462–1474 (2001)

7. M. Drutarovsky, P. Galajda, Chaos-based true random number generator embedded in a reconfigurable hardware. J. Electr. Eng. **57**(4), 218–225 (2006)
8. E. Fatemi-Behbahani, E. Farshidi, K. Ansari-Asl, A new approach to analysis of residue probability density function in pipelined ADCs. Integr. VLSI J. **52**(1), 51–61 (2016)
9. J. Guerber, M. Gande, U.-K. Moon, The analysis and application of redundant multistage ADC resolution improvements through PDF residue shaping. IEEE Trans. Circuits Syst. I Reg. Pap. **59**(8), 1733–1742 (2012)
10. U. Güler, S. Ergün, A high speed, fully digital IC random number generator. AEU Int. J. Electron. Commun. **66**(2), 143–149 (2012)
11. P. Huang, S. Hsien, V. Lu, P. Wan, S.C. Lee, W. Liu, B.W. Chen, Y.P. Lee, W.T. Chen, T.Y. Yang, G.K. Ma, SHA-less pipelined ADC with in-situ background clock-skew calibration. IEEE J. Solid State Circuits **46**(8), 1893–1903 (2011)
12. O. Katz, D.A. Ramon, I.A. Wagner, A robust random number generator based on a differential current-mode chaos. IEEE Trans. Very Large Scale Integr. VLSI Syst. **16**(12), 1677–1686 (2008)
13. J.P. Keane, P.J. Hurst, S.H. Lewis, Digital background calibration for memory effects in pipelined analog-to-digital converters. IEEE Trans. Circuits Syst. I Reg. Pap. **53**(3), 511–525 (2006)
14. M.G. Kim, P.K. Hanumolu, U.-K. Moon, A 10 MS/s 11-bit 0.19 mm^2 algorithmic ADC with improved clocking scheme. IEEE J. Solid State Circuits **44**(9), 2348–2355 (2009)
15. C.C. Lee, M.P. Flynn, A SAR-assisted two-stage pipeline ADC. IEEE J. Solid State Circuits **46**(4), 859–869 (2011)
16. B. Levy, A propagation analysis of residual distributions in pipeline ADCs. IEEE Trans. Circuits Syst. I Reg. Pap. **58**(10), 2366–2376 (2011)
17. J. Li, G. Ahn, D. Chang, U.-K. Moon, A 0.9-V 12-mW 5-MSPS Algorithmic ADC With 77-dB SFDR. IEEE J. Solid State Circuits **40**(4), 960–969 (2005)
18. N. Li, W. Pan, S. Xiang, L. Yan, B. Luo, X. Zou, Influence of statistical distribution properties on ultrafast random-number generation using chaotic semiconductor lasers. Optik Int. J. Light Electron. Opt. **125**(14), 3555–3558 (2014)
19. F. Maloberti, *Data Converters* (Springer, Berlin, 2007)
20. A. Papoulis, S.U. pillai, *Probability Random Variables and Stochastic Processes*, 4th edn. (McGraw-Hill, New York, 2002)
21. F. Pareschi, G. Setti, R. Rovatti, Implementation and testing of high-speed CMOS true random number generators based on chaotic systems. IEEE Trans. Circuits Syst. I Reg. Pap. **57**(12), 3124–3137 (2010)
22. M.J.M. Pelgrom, A.C.J. Duinmaijer, A.P.G. Welbers, Matching properties of MOS transistors. IEEE J. Solid State Circuits **24**(5), 1433–1439 (1989)
23. S. Robson, B. Leung, G. Gong, Truly random number generator based on a ring oscillator utilizing last passage time. IEEE Trans. Circuits Syst. II Exp. Briefs **61**(12), 937–941 (2014)
24. A. Rukhin, J. Soto, J. Nechvatal, M. Smid, E. Barker, S. Leigh, M. Levenson, A statistical test suite for random and pseudorandom number generators for cryptographic applications, in *National Institute of Standards and Technology (NIST), Special Publication800-22, Revision 1a, (2010)*. http://csrc.nist.gov/groups/ST/toolkit/rng/documents/SP800-22rev1a.pdf. (online)
25. G. Setti, G. Mazzini, R. Rovatti, S. Callegari, Statistical modeling of discrete-time chaotic processes-basic finite-dimensional tools and applications. Proc. IEEE **90**(5), 662–690 (2002)
26. E.G. Soenen, R.L. Geiger, An architecture and an algorithm for fully digital correction of monolithic pipelined ADCs. IEEE Trans. Circuits Syst. II Analog Digit. Signal Process. **42**(3), 143–153 (1995)
27. A.B. Sripad, D.L. Snyder, A necessary and sufficient condition for quantization errors to be uniform and white. IEEE Trans. Acoust. Speech Signal Process. **ASSP–25**(5), 442–448 (1977)
28. T. Stojanovski, L. Kocarev, Chaos-based random number generator–part I: analysis. IEEE Trans. Circuits Syst. I Fundam. Theory Appl. **48**(3), 281–288 (2001)
29. T. Stojanovski, L. Kocarev, Construction of Markov partitions in PL1D maps. IEEE Trans. Circuits Syst. II Exp. Briefs **60**(10), 702–706 (2013)
30. T. Stojanovski, J. Pihl, L. Kocarev, Chaos-based random number generator–part II: practical realization. IEEE Trans. Circuits Syst. I Fundam. Theory Appl. **48**(3), 382–385 (2001)
31. B. Widrow, I. Kollar, *Quantization Noise-Roundoff Error in Digital Computation, Signal Processing, Control, and Communications* (Cambridge University Press, Cambridge, 2008)
32. P.Z. Wieczorek, An FPGA implementation of the resolve time-based true random number generator with quality control. IEEE Trans. Circuits Syst. I Reg. Pap. **61**(12), 3450–3459 (2014)