A New Chaos-Based PRNG Hardware Architecture Using the HUB Fixed-Point Format

Samuel Souza da Silva, Matheus Cardoso, Lucas Nardo, Erivelton Nepomuceno, *Senior Member, IEEE*, Michael Hübner, *Senior Member, IEEE*, and Janier Arias-Garcia

***Abstract*— Chaotic systems have been applied in many applications involving instrumentation and measurements, such as in sensors and control systems, due to their pseudorandom proprieties. However, reproducing chaos in digital systems is challenging because of the dynamical degradation of chaotic digital systems and, consequently, their intrinsic periodic orbits. Many techniques have been used to guarantee a sufficiently large period, making them suitable for such applications. Nevertheless, few articles pay attention to the effects of using different numerical representations on chaos degradation and, at the same time, keeping key design parameters, such as few logical resources and power consumption. Thus, this article aims to provide a new hardware architecture for a chaos-based pseudorandom number generator (PRNG) using the Half-Unit-Biased (HUB) format for fixed-point numbers by bi-coupling the tent map in conjunction with the Bernoulli map, causing a significant impact on its logical resources and performance. Results show that the HUB format is more effective than the standard fixed-point numerical representation and that the proposed approach is chaotic, with pseudorandomness.**

***Index Terms*— Bernoulli map, chaotic systems, computer arithmetic, Half-Unit-Biased (HUB) format, pseudorandom number generator (PRNG), tent map.**

I. Introduction

CHAOTIC systems present attractive properties to scientific and industrial communities, such as high sensitive dependence on initial conditions and unpredictability [1]. For instance, Fortuna et al. [2] proved that the peculiarity of chaos enhanced the performance of the ultrasonic sensors' instruments that use unique sequences to identify their echo. In addition, Garcia-Bosque et al. [3] proposed a new chaos-based pseudorandom number generator (PRNG) that can be applied in Monte Carlo simulations while keeping low hardware resource consumption. Furthermore, in [4], an encryption scheme based on a chaotic algorithm has been successfully used in the Ethernet 1000Base-X standard.

Although there are significant applications involving chaotic systems in the literature, preserving chaotic properties in digital chaotic maps can be a nontrivial task when using finite precision [1]. Whenever using a word length of n bits, the iterations of the digital chaotic map can only take a maximum of 2^n distinct values. Thus, because the value of a given iteration determines the value of the next one, its iterations will necessarily enter cycles that degenerate chaotic properties [3], [5]. An intuitive method to mitigate the dynamical degradation of chaos is using higher bit resolutions, although it might result in the use of many hardware resources. Consequently, to solve this problem, researchers have been studying different methods to reduce dynamical degradation. As a common approach, one can cascade chaotic maps [5].

Many articles have been adopting the use of fixed-point representation in digital chaotic systems [3], [6], [7], [8]. Specifically, in [6], a chaos-based hash function using fixed-point arithmetic was implemented, showing that the proposed system has low computational complexity. However, additional logical circuits would be necessary to perform rounding to the nearest number using such a format, which could considerably increase the hardware resources.

The use of the floating-point representation is also presented in recent articles [7], [9], [10], [11]. For instance, Aboulseoud and Ismail [9] introduced a hardware implementation of the generalized fractional-order logistic map using the IEEE-754 floating-point format to achieve high accuracy and precision. This numerical representation significantly increases the dynamic range. However, it is noticeable that algebraic operations are more complex to be accomplished in floating-point arithmetic when compared with the fixed-point format.

In addition, new formats to represent numbers have been studied in the literature, such as the Half-Unit-Biased (HUB) format and the round-to-nearest (RN) representation. Herewith, in [12], a new adder hardware implementation for the RN

Manuscript received 6 October 2022; revised 27 November 2022; accepted 21 December 2022. Date of publication 10 January 2023; date of current version 25 January 2023. The work of Erivelton Nepomuceno was supported in part by the Brazilian Research Agencies: Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq)/Instituto Nacional de Ciência e Tecnologia em Energia Elétrica (INERGE) under Grant 465704/2014-0, in part by the CNPq under Grant 425509/2018-4 and Grant 311321/2020-8, and in part by the Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG) under Grant APQ-00870-17. The work of Lucas Nardo was supported in part by the Fundação de Amparo à Pesquisa do Estado de Minas Gerais and in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior–Brazil (CAPES) (Finance Code 001). The Associate Editor coordinating the review process was Dr. Ziqiang Cui. *(Corresponding author: Janier Arias-Garcia.)*

Samuel Souza da Silva, Lucas Nardo, and Janier Arias-Garcia are with the Graduate Program in Electrical Engineering and Department of Electronic Engineering, Universidade Federal de Minas Gerais (UFMG), Belo Horizonte, MG 31270-901, Brazil (e-mail: janier-arias@ufmg.br).

Matheus Cardoso is with the Department of Electronic Engineering, Universidade Federal de Minas Gerais (UFMG), Belo Horizonte, MG 31270-901, Brazil.

Erivelton Nepomuceno is with the Centre for Ocean Energy Research, Department of Electronic Engineering, Maynooth University, Maynooth, W23 F2H6 Ireland.

Michael Hübner is with the Department of Computer Engineering, Brandenburgische Technische Universität (BTU) Cottbus-Senftenberg, 492667 Cottbus, Germany.

Digital Object Identifier 10.1109/TIM.2023.3235457

representation is presented, claiming that the proposed adder establishes a native RN architecture suitable to overcome the double rounding error. Nevertheless, it can be seen that RN numbers need at least one more bit than HUB numbers to perform operations with the same precision [12].

It is known that it would be necessary to use the same number of explicit bits to represent a number with the same precision using either the HUB format or the well-known standard fixed-point format. In addition to using the same storage cost as standard representations [13], the HUB format drastically simplifies the rounding-to-the-nearest operation only by the cost of truncation [13]. Hence, it can be inferred that the HUB representation in chaotic maps not only can reduce their resources in hardware and improve their performance to accomplish rounding to the nearest but also it can directly influence the orbit of the chaotic map. Therefore, HUB representation is a striking alternative for designing efficient hardware for digital chaotic systems. In such a way, this article presents a hardware architecture bi-coupling the tent map in conjunction with the Bernoulli map, in which a fixed-point representation based on the HUB format was used. Thus, the contributions of this article are summarized as follows.

1) A proposal of a new hardware architecture for a chaos-based PRNG, which uses the HUB fixed-point format, bi-coupling the tent map with the Bernoulli map.
2) A comparison analysis that evaluates the influence of the HUB format in the proposed chaotic system in comparison to the use of the well-known standard fixed-point representation.

The proposed hardware architecture was successfully tested by statistical test suites for random sequences, such as the NIST SP 800-22 and the ENT test suites, presenting pseudo-random properties. Hence, the proposed system can be used as a component in applications involving instrumentation and measurements, such as the Monte Carlo simulation. Moreover, the results show that using HUB representation is more efficient than other numerical representations.

The rest of this article is divided as follows. Section II presents preliminary concepts to have a better understanding of this article. Section III contains details regarding the proposed hardware architecture. Section IV shows the results and a comparative analysis of the system using the standard fixed-point format and the HUB fixed-point format. Finally, Section V presents the conclusions concerning this article.

II. Preliminary Concepts

A. Tent Map

The tent map is a discrete system that presents chaotic behavior depending on the value of its control parameter. It is represented by the following:

$$x_{n+1} = \begin{cases} \mu x_n, & \text{if } x_n \in [0, 1/2) \\ \mu(1 - x_n), & \text{if } x_n \in [1/2, 1) \end{cases} \quad (1)$$

where $x_n \in [0, 1)$ and the control parameter $\mu \in [0, 2]$. Moreover, the chaotic behavior of this map is obtained for $1 < \mu \leq 2$ [14].

B. Bernoulli Map

The 1-D Bernoulli map is a discrete system defined by the following:

$$x_{n+1} = \begin{cases} 2x_n, & \text{if } x_n \in [0, 1/2) \\ 2x_n - 1, & \text{if } x_n \in [1/2, 1). \end{cases} \quad (2)$$

It is noticeable that $x_n \in [0, 1)$ and the map does not have a control parameter. The Bernoulli map has a positive Lyapunov exponent, equal to $\ln(2) = 0.693$; therefore, it presents chaotic behavior [15].

C. Half-Unit-Biased Format

HUB is a representation format where numbers necessarily have an implicit least significant bit constant and equal to one [16]. Herewith, considering the term "exactly represented number" (ERN) as the number value that is represented after performing the rounding in a given operation, the ERN of a bit vector under the HUB format corresponds to the ERN of the same bit vector using the conventional format added a value of half unit in the last place (ulp) [16]. For instance, when using three bits for the fractional part and one bit for the integer part of a HUB number in fixed-point, the ERN of the bit vector 0.110_2 is 0.8125_{10}.

Some advantages of using the HUB format are that it performs operations, such as the two's complement, in a simpler way. The two's complement can be performed by bitwise inversion. Then, as an example, the two's complement of the HUB number 0.110_2 is given by 1.001_2. Furthermore, the ability to round to the nearest number can be done only by the cost of truncation [16]. Thus, the HUB format can remarkably reduce the hardware resources needed to perform the round-to-the-nearest operation since conventional standard formats require the usage of additional circuits to perform it.

Fig. 1 illustrates, in (a), the process of truncating a number in a conventional fixed-point representation, using three bits for the fractional part. The non-ERNs are shown in the figure in black circles, whereas the ERNs are shown in light blue circles. In this figure, it is clear that rounding is necessary to be done to represent the non-ERN when using only three bits for the fractional part. Therefore, when using the standard representation in (a), the truncation operation is responsible for an effective rounding down. However, due to the implicit least significant bit when using HUB numbers, truncating a number in (b) leads to rounding to the nearest ERN. In addition, it is important to mention that in (b), the ERN uses three explicit bits for the fractional part, and their implicit bit is shown in red for convenience.

Furthermore, to work with the HUB fixed-point format, a general rule can be applied to compute arithmetic operations. First, the bit vector used is extended with an additional least significant bit set to one. Then, one can apply the conventional arithmetic operations considering the extended bit vector. Finally, after calculating the result of such an arithmetic operation, one can have the output HUB format simply by truncating it, obtaining the desired number of bits [16].

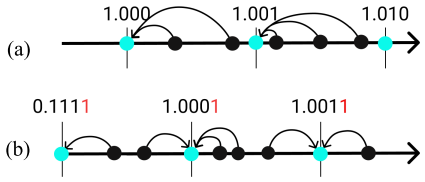

Fig. 1. Process of truncating a number using: (a) standard fixed-point representation for numbers using three bits for the fractional part and (b) HUB fixed-point format for numbers using three explicit bits for the fractional part. The non-ERNs are represented in black circles, and the ERNs after truncation are represented in blue circles.

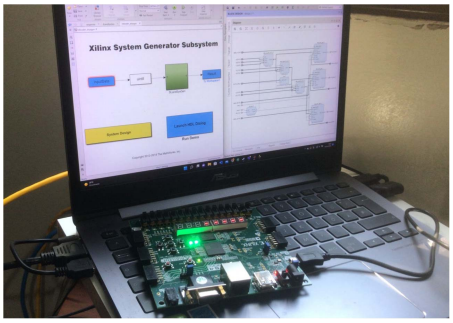

Fig. 2. Experimental setup using the Nexys 4 board with the low-cost FPGA Xilinx *Artix 7 XC7A100TCSG324*.

III. Hardware Architecture

For the fixed-point representation used in this article, we chose to represent the numbers using 0 bits for the integer part and 32 bits for the fractional part. Such a decision was made based on the fact that the domain of both the tent map and the Bernoulli map belongs to the interval $[0, 1)$. Moreover, VHDL was used for the description of the digital circuit along with the Xilinx Vivado 2019.1 synthesis tool. In addition, the hardware architecture of this article was mapped into the field-programmable gate array (FPGA) device *Artix 7 XC7A100TCSG324*. Furthermore, to reproduce the experimental tests done in this article, one can use the hardware setup shown in Fig. 2, which uses a Nexys 4 board along with a system generator tool and MATLAB on a personal computer.

To implement the tent map in hardware, it was necessary to pay attention to the singularities of the algebraic operations of its recurrence equation. It is known that to compute the subtraction of two real numbers x_1 and x_2, i.e., $x_1 - x_2$, when using the standard fixed-point format, one can simply add x_1 with the two's complement of x_2. Nonetheless, once the two's complement of a number in the HUB format is performed by bitwise inversion [17], the subtraction can be obtained by extending the implicit least significant bit of x_1 and x_2, adding x_1 with the two's complement of x_2, and then performing truncation to have the round to the nearest approximation.

The proposed tent map using the HUB format can be implemented as shown in Fig. 3(a). Since the subtraction only occurs when $x_n \geq 1/2$, as seen in (1), and because of $1/2 = 0.100, \ldots, 0_2$, the most significant bit, $x_n[31]$, is the only bit that matters to verify the aforesaid condition. Thus, if the most significant bit is set to "1", then the input of the tent map is greater than $1/2$. Otherwise, the number is less than $1/2$.

Seeing Fig. 3(a), the selector of the multiplexer (MUX) is set to "0" if x_n is less than half, making the output of the map be $x_{n+1} = \mu x_n$. Otherwise, the selector of the MUX would be set to "1" and the output of the chaotic map would be equal to $x_{n+1} = \mu(1 - x_n)$. Since all the 32 bits of the word length were being used for the fractional part, and as the control parameter of the tent map needs to belong to the interval $(1, 2]$ to present a chaotic behavior, the parameter μ was computed in such a way that it also had implicitly one bit set for the integer part, which makes sure that the control parameter μ is greater than one and that it belongs to the interval in which the tent map has a positive Lyapunov exponent. Furthermore, once it is not possible to represent exactly the integer number "1" in the HUB format, the symbol "1 + 1/2 ulp" indicates that the ERN of it consists of the value "1" added half ulp.

For the Bernoulli map, a similar approach was used (see Fig. 3(b)). Since it is not possible to represent exactly the number "2" in the HUB format, to approximate the multiplication by two, the following steps were taken: first, shift left the bits of the represented number by one bit, and then set the least explicit significant bit to "1". The MUX in Fig. 3(b) works the same way it was used for the tent map.

As will be demonstrated in Section IV, empirical tests indicate that the iterations of the hardware architectures of the tent map and Bernoulli map (Fig. 3(a) and (b)) have very limited periods for a variety of values of the parameter control, which makes these systems not suitable in certain applications, such as in image encryption. This limited period is due to the dynamical degradation when using finite precision.

Nevertheless, to mitigate the dynamical degradation process, this article applies a bi-coupling approach, which consists of a particular case shown in [5]. Thus, the output of the tent map influences the input of the Bernoulli map and vice versa. In the proposed system, the least significant 29 explicit bits of the parameter μ of the tent map are set to be equal to the least significant 29 explicit bits of the output of the Bernoulli map, whereas the three remaining explicit bits of the μ parameter was set to have a fixed value of "111_2," making the parameter μ assume values in the interval $[1.75, 2]$. This decision was made based on the fact that the Lyapunov exponent of the tent map is high for values of the control parameter close to "2," as well as it guarantees that the tent map presents a chaotic dynamic behavior in the system. Thus, the control parameter of the tent map of the architecture of this article is dictated by the output of the Bernoulli map. Fig. 3(c) shows the proposed architecture done in this article. Note that the control parameter of the tent map is determined based on the iterations of the Bernoulli map; therefore, the parameter μ is not an input of the hardware architecture.

A D flip-flop with an input set to the logical high was used in Fig. 3(c). The flip-flop makes the selector (S) of the MUX to be equal to the logical low in the first iteration, and, therefore, the output of the MUX will be equal to the initial condition (x_0) of the proposed chaotic system. Then, after the first iteration has been computed, the output of the flip-flop will assume the logical high, as it will receive a clock pulse, making the output of the MUX equal to the previous value generated by the hardware architecture.

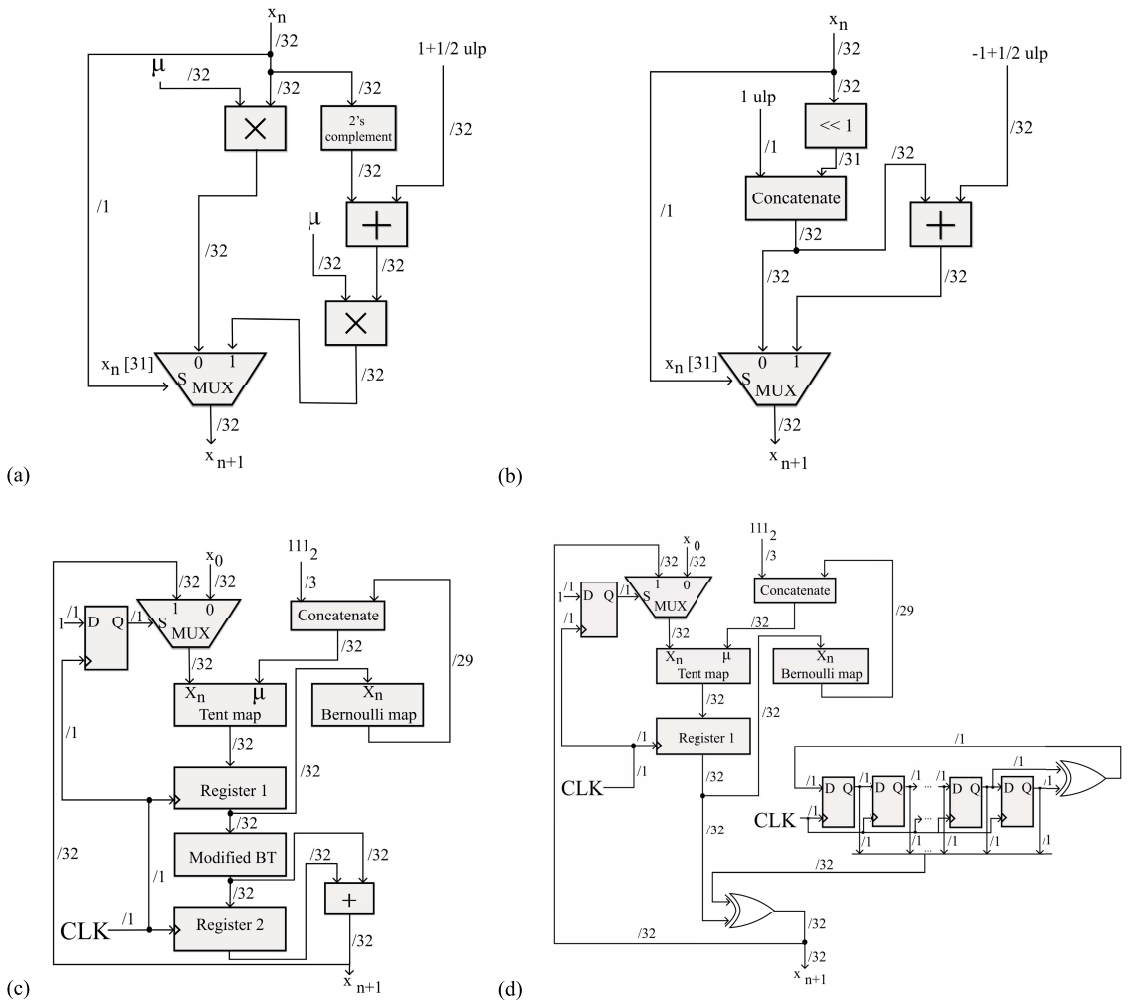

Fig. 3. Proposed hardware architectures using the HUB format for the following cases. (a) Tent map. (b) Bernoulli Map. (c) Bi-coupled map. (d) Bi-coupled map using an LFSR. For the addition and multiplication operations, the following steps were taken: first, explicitly extend the least significant bit of the operands; second, operate using conventional arithmetic; third, truncate the result to have 32 bits. The block "Concatenate" indicates the concatenation of bits, the symbol "$\ll 1$" corresponds to the shift left of one bit, and $x_n[31]$ corresponds to the most significant bit of the word length of 32 bits.

Furthermore, to make the proposed hardware architecture produce uniformly distributed sequences, an operation called "modified bit transformation (BT)" was used, which is slightly different from the BT suggested in [18]. The "modified BT" operation takes as input an array x, of 32 bits, as described in (3). Then, it inverts half of the input bits, as shown in (4). The other half, described in (5), is also used to compute f_h [see (6)], where the bitwise XOR operation is performed. The result of the "modified BT" operation, called "MBT{x}," shown in (7), is obtained by concatenating f_h and f_l. Finally, after computing it, an adder is introduced into the proposed chaotic system, adding two substates of the system, contributing to avoiding its dynamical degradation.

$$x = b_{31}b_{30}b_{29}, \ldots, b_3b_2b_1b_0 \tag{3}$$

$$f_l = b_0b_1b_2, \ldots, b_{15} \tag{4}$$

$$f_h' = b_{31}b_{30}b_{29}, \ldots, b_{16} \tag{5}$$

$$f_h = f_h' \oplus f_l \tag{6}$$

$$\text{MBT}\{x\} = f_h, f_l. \tag{7}$$

For comparison purposes, the proposed hardware architecture in Fig. 3(c) was also made using the standard fixed-point representation. However, it was not possible to represent this hardware architecture using the IEEE-754 single-precision floating-point standard since the "modified BT" operation can make the system produce values that are outside of the domain of both the tent map and the Bernoulli map for such arithmetic.

In addition, another approach to reduce the dynamical degradation was made in this article, as can be seen in Fig. 3(d). Hence, instead of using the "modified BT" operation along with an adder, this approach does a bitwise XOR operation between the output of the tent map and the output of a linear feedback shift register (LFSR), similar to what was done in [7]. Nevertheless, as it will be described in Section IV, this approach does not present better results compared with the architecture shown in Fig. 3(c).

The following nomenclature will be used to refer to the proposed hardware architectures in the rest of this article:

1) Bi-HUB: the proposed Bi-coupled map using the HUB fixed-point format;
2) Bi-Standard: the proposed Bi-coupled map using the standard fixed-point format;

TABLE I

PERIOD AND TRANSIENT OF THE CHAOTIC HARDWARE ARCHITECTURES USING THE HUB FORMAT. THE TERM "TRANSIENT" IS USED HERE TO REFER TO THE NUMBER OF ITERATIONS THAT OCCURS INITIALLY THAT DO NOT BELONG TO THE PERIOD OF THE PSEUDO-ORBIT ITSELF

Hardware Architecture	Period	Transient
Tent map (Fig. 3-A) - $\mu \approx 1.75$	20013	41279
Bernoulli map (Fig. 3-B)	1	30
Bi-HUB	Not found	Not found
Bi LFSR-HUB	63563	17155

3) Bi LFSR-HUB: the proposed Bi-coupled map using the HUB fixed-point format in conjunction with the LFSR.

IV. RESULTS

To demonstrate the effects of the perturbation method of this article, the proposed hardware architectures were all tested to evaluate the period of its pseudo-orbits. Once it is being used a finite precision (32 bits), it can be inferred that the proposed systems have necessarily periodic pseudo-orbits. However, if the period of a digital chaotic system is sufficiently large, it can be used in a variety of applications. Thus, Table I shows the period identified after testing each hardware architecture when using the HUB format. A million iteration was generated for testing each hardware architecture, considering the initial condition $x_0 \approx 0.71$. For the tent map, the value of the μ parameter used is $\mu \approx 1.75$. Nonetheless, for the final implementation, the value of the μ parameter changes on each iteration. Moreover, similar results were obtained when testing the same systems for a variety of other initial conditions and also when using the standard fixed-point format.

Based on Table I, it is clear that the process of dynamical degradation makes the Bernoulli map converge to one unique value, which is $0.4\overline{9}$ when using the HUB format. Furthermore, Table I shows the tent map has periodic pseudo-orbits with a period of 20013, for $x_0 \approx 0.71$ and $\mu \approx 1.75$. Nevertheless, this period is not high enough for some applications. For example, in image encryption schemes, when encrypting images of size 512×512, it would be necessary to iterate at least 262 144 times the chaotic system, which invalidates this system to be used in this kind of application. In addition, when using the perturbation method proposed in Fig. 3(c), it was not identified a period. Thus, in this case, it is possible to generate a sequence of 1 000 000 values, each value with 32 bits, without a defined period. Therefore, this is evidence that the proposed perturbation method in Fig. 3(c) mitigates chaos degradation, justifying its use. Moreover, the perturbation method using an LFSR is able to mitigate chaos degradation as its period increases in relation to when using only the tent map or the Bernoulli map. However, a period of 63563 is identified in this case, which limits its use in some specific applications.

To validate the proposed hardware architectures, the bi-coupled system was submitted to the NIST SP 800-22 test suite [19]. Accordingly, the NIST SP 800-22 test suite consists of a set of 15 statistical tests for random and PRNGs [19]. Each test has as output a number called "P-value," which belongs to the interval $[0, 1)$. If the output of the corresponding test has a P-value greater than a significance level α, typically equal to 0.01, then the sequence is considered random for the given statistical test. Thus, a system is considered to have pseudorandom properties if, for all tests, its P-values are greater than α. Herewith, Table II shows the results obtained for the NIST SP 800-22 test suite, considering 100 events of a bitstream length of 1 000 000 bits and a significance level $\alpha = 0.01$. Therefore, it is noticeable that in the Bi-HUB and Bi-Standard cases, the proposed systems have passed all the tests, as all the P-values obtained were greater than α. Moreover, for the Bi LFSR-HUB case, the system has passed most of the tests, but not all of them, since some tests present P-value $< \alpha$. Herewith, although the bi-coupled system with an LFSR did not pass all the tests, it is known that using an LFSR as a source of perturbation is a common practice [7], [20], [21], and, therefore, the results obtained for the Bi LFSR-HUB architecture are presented in this article only for compassion purposes.

In contemplation of validating the use of the proposed bi-coupled system as a PRNG, we also submitted the generated bitstream in the ENT test suite, a tool that has been extensively used to prove pseudorandom features of numerical sequences. By running six different statistical tests, this series of tests presents a satisfactory indicator of the quality of random generators. Table III shows the results obtained for each numerical representation used, which reaffirms what was shown by the NIST SP 800-22 test suite. These tests confirm that the bi-coupled system using HUB fixed-point or the standard fixed-point format presents outstanding pseudorandom features. Furthermore, the ENT result proves that the proposed PRNG can effectively be used in the Monte Carlo simulation as it got values close to π in the "Monte Carlo value for Pi" test.

Due to the nature of the perturbation method applied, the complexity of calculating the Lyapunov exponent analytically is very high. Thus, an approximation of the Lyapunov exponent was evaluated. The Lyapunov exponent was calculated based on the method described by Kantz [23]. Table III shows the results obtained for the proposed hardware architectures using different numerical representations. In each case, it used the software available in [22]. In all the cases, the results exhibit that the systems have a positive Lyapunov exponent, an important factor to indicate the presence of chaotic properties.

The state space of the system is shown in Fig. 4. For all the cases, the state space presents a complex shape due to the perturbation method applied. Moreover, Fig. 4 shows that the autocorrelation approximates a Dirac delta function in all the approaches, showing that the sequence generated does not have an apparent correlation between its values. Ultimately, the histogram was obtained, as shown in Fig. 4, and the distribution for the cases presented is close to a uniform distribution.

Furthermore, Fig. 5(a) exhibits the time series of the system evaluating the Bi-HUB and Bi-Standard approaches. Clearly, Fig. 5(a) shows that the numerical format used could influence directly the output of the system since the pseudo-orbits of the systems diverge after the first iteration. In addition, Fig. 5(b)

TABLE II

P-VALUE RESULTS AFTER RUNNING THE SP 800-22 TEST SUITE FOR THE PROPOSED HARDWARE ARCHITECTURES

	Bi-HUB		Bi-Standard		Bi LFSR-HUB	
Test	**P-value**	**Proportion**	**P-value**	**Proportion**	**P-value**	**Proportion**
Frequency	0.514124	100/100	0.437274	98/100	0.437274	97/100
Block Frequency ($m = 128$)	0.779188	99/100	0.534146	99/100	0.000199	100/100
Cusum-Forward	0.153763	100/100	0.779188	98/100	0.897763	97/100
Cusum-Reverse	0.595549	100/100	0.366918	98/100	0.834308	97/100
Runs	0.071177	98/100	0.021999	98/100	0.000000	83/100
Longest Runs of Ones	0.319084	100/100	0.574903	99/100	0.678686	97/100
Rank	0.534146	100/100	0.739918	100/100	0.779188	99/100
FFT	0.153763	100/100	0.383827	100/100	0.016717	98/100
Non-overlapping Templates	0.595549	99/100	0.678686	99/100	0.016717	98/100
Overlapping Templates ($m = 9$)	0.153763	98/100	0.191687	100/100	0.554420	99/100
Universal	0.657933	99/100	0.616305	98/100	0.000000	95/100
Approximate Entropy	0.699313	99/100	0.224821	98/100	0.045675	97/100
Random Excursions ($x = +1$)	0.897763	56/57	0.057146	64/65	0.153763	52/54
Random Excursions Variant ($x = -1$)	0.062821	56/57	0.723129	64/65	0.236810	53/54
Linear Complexity ($M = 500$)	0.045675	99/100	0.678686	99/100	0.334538	98/100
Serial ($m = 16$, $\nabla\Psi^2_m$)	0.574903	99/100	0.108791	99/100	0.816537	99/100

TABLE III

ENT TEST SUITE RESULTS OBTAINED FOR THE HARDWARE ARCHITECTURES OF THIS ARTICLE AND LYAPUNOV EXPONENT CALCULATED BASED ON [22]

Test	**Bi-HUB**	**Bi-Standard**	**Bi LFSR-HUB**
Entropy (bits per bit)	1.0	1.0	1.0
Chi-square (abs)	3.12	1.06	3.88
Chi-square (percentage)	7.72	30.31	4.88
Arithmetic Mean	0.5001	0.5000	0.5001
Monte Carlo Value for Pi	3.14103	3.14051146	3.14490
Serial Correlation	0.000139	0.000091	-0.001353
Lyapunov exponent	5.15388	5.15332	0.8514

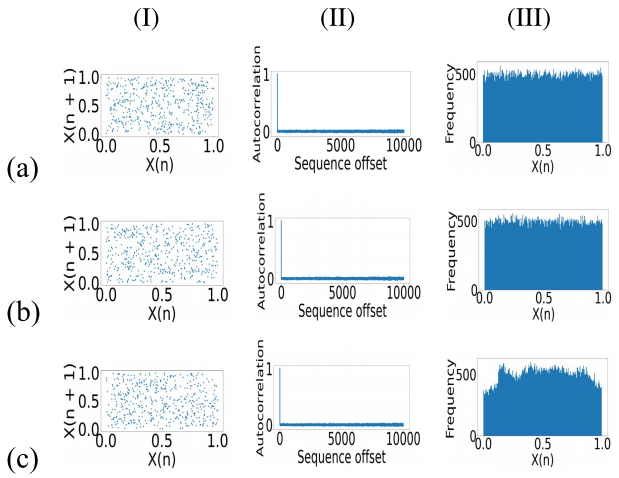

Fig. 4. Results for the proposed hardware architectures. Column (I) corresponds to the chaotic state space, column (II) shows the autocorrelation obtained, and column (III) exhibits the histograms of the systems. Cases (a), (b), and (c) correspond, respectively, to the following architectures: Bi-HUB, Bi-Standard, and Bi LFSR-HUB.

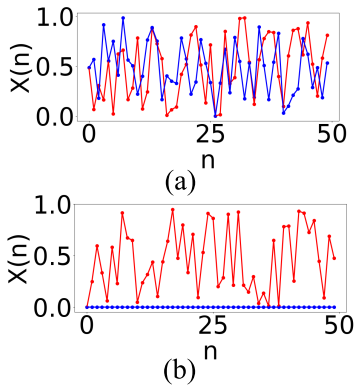

Fig. 5. Time series of the Bi-HUB architecture (in red) and the Bi-Standard architecture (in blue). In (a), both the systems had the same bit vector as the initial condition. In (b), an initial condition of a bit vector full of zeros was used.

reveals that using the HUB format has a positive effect of avoiding the chaos annulling condition when all 32 explicit bits of the initial condition are set to zero. Since this numerical representation has a half unit in the last place, such a condition makes the output of the chaotic maps different than zero, not degenerating the chaotic behavior. In contrast, for the same hardware using the standard fixed-point representation, if the input is set to be a bit vector with all the bits set to zero, it will completely degenerate chaos as the output of the system will always be zero. In addition, it is valid to mention that empirical tests were made to find more chaos annulling conditions for the architecture of Fig. 3(c), but no such values were found.

Table IV summarizes the hardware resources achieved for different representations. Note that the proposed system, using the HUB format (Bi-HUB), uses slightly fewer logical resources than the standard fixed-point representation (Bi-Standard). This is because, when adopting the HUB format in a given representation, some operations are simpler to be accomplished, such as the two's complement and rounding to the nearest. Moreover, the Bi-Standard architecture performs rounding by truncation, which corresponds to an effective rounding down. However, if we use the standard fixed-point representation performing rounding to the nearest, it would be necessary to use even more hardware resources, as additional logic circuits would be required.

Furthermore, Table IV shows comparison of the different hardware resource consumption and performance of other state-of-the-art chaotic system implementations, showing that the proposed hardware architectures are competitive with the state-of-the-art articles in terms of hardware resources and performance.

Summarizing, the hardware architecture that presents the best tradeoff between chaotic properties, pseudorandom

TABLE IV
COMPARISON ANALYSIS IN TERMS OF HARDWARE RESOURCES OF DIFFERENT APPROACHES

Chaotic system	Reference	LUT	FF	DSP	IO	Power (mW)	FMax (MHz)	Lyapunov exponent
Bi-HUB	This paper	242	65	8	66	114	61.94	5.15388
Bi-Standard	This paper	258	65	8	66	114	63.90	5.15332
Bi LFSR-HUB	This paper	226	65	8	66	113	64.81	0.8514
PRNG - Exponential map (32 bits)	[7]	915	1101	6	66	96	121.95	0.7576
PRNG - Exponential map (64 bits)	[7]	1466	1256	48	130	90	43.47	2.2733
PRNG - Henon Map	[8]	856	521	32	20	128	-	-
CSS-2	[24]	549	192	72	97	-	-	0.00154
PRNG - LCM Modified	[25]	5806	-	-	-	300	80.592	-
PWLCM	[26]	4215	578	0	-	10	106	1.16

features, and consumption of logical resources is the Bi-HUB architecture. The main advantages of using the HUB format identified in this article are as follows:

1) It prevents the system from falling into a chaos-annulling condition when having an input with a bit vector full of zeros;
2) It had a positive effect on lowering the use of hardware resources. This can make a considerable impact on platforms that have tight constraints, such as embedded systems, on which resources available are a key factor.

V. CONCLUSION

In this article, we described a new chaos-based PRNG using the HUB format, bi-coupling the tent map in conjunction with the Bernoulli map. We effectively implemented the hardware architecture in the low-cost FPGA *XC7A100TCSG324*. The results show that the proposed system has chaotic behavior and pseudorandom properties, with a positive Lyapunov exponent, a uniform histogram, an autocorrelation function similar to a Dirac delta function, and passing in statistical test suites for random sequences. These characteristics validate the use of the proposed chaotic system as a PRNG, being suitable for use in some applications involving instrumentation and measurements, such as the Monte Carlo simulation. Moreover, the adoption of the HUB format has a positive impact since it avoids the system from having chaos annulling conditions when using an input vector with all zeros as an initial condition, and it also makes the architecture use fewer hardware resources.

ACKNOWLEDGMENT

The authors would like to thank the Pró-Reitoria de Pesquisa (PRPq) of the Universidade Federal de Minas Gerais (UFMG).

REFERENCES

[1] R. M. Corless, "What good are numerical simulations of chaotic dynamical systems?" *Comput. Math. Appl.*, vol. 28, nos. 10–12, pp. 107–121, Nov. 1994, doi: 10.1016/0898-1221(94)00188-x.

[2] L. Fortuna, M. Frasca, and A. Rizzo, "Chaotic pulse position modulation to improve the efficiency of sonar sensors," *IEEE Trans. Instrum. Meas.*, vol. 52, no. 6, pp. 1809–1814, Dec. 2003, doi: 10.1109/TIM.2003.820452.

[3] M. Garcia-Bosque, A. Pérez-Resa, C. Sánchez-Azqueta, C. Aldea, and S. Celma, "Chaos-based bitwise dynamical pseudorandom number generator on FPGA," *IEEE Trans. Instrum. Meas.*, vol. 68, no. 1, pp. 291–293, Jan. 2019, doi: 10.1109/TIM.2018.2877859.

[4] A. Perez-Resa, M. Garcia-Bosque, C. Sanchez-Azqueta, and S. Celma, "Chaotic encryption applied to optical Ethernet in industrial control systems," *IEEE Trans. Instrum. Meas.*, vol. 68, no. 12, pp. 4876–4886, Dec. 2019, doi: 10.1109/TIM.2019.2896550.

[5] L. Liu, B. Liu, H. Hu, and S. Miao, "Reducing the dynamical degradation by bi-coupling digital chaotic maps," *Int. J. Bifurcation Chaos*, vol. 28, no. 5, May 2018, Art. no. 1850059, doi: 10.1142/s0218127418500591.

[6] J. S. Teh, K. Tan, and M. Alawida, "A chaos-based keyed hash function based on fixed point representation," *Cluster Comput.*, vol. 22, no. 2, pp. 649–660, Nov. 2018, doi: 10.1007/s10586-018-2870-z.

[7] M. B. R. Cardoso et al., "A new PRNG hardware architecture based on an exponential chaotic map," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2021, doi: 10.1109/ISCAS51556.2021.9401653.

[8] R. Hobincu and O. Datcu, "FPGA implementation of a chaos based PRNG targetting secret communication," in *Proc. Int. Symp. Electron. Telecommun. (ISETC)*, Nov. 2018, doi: 10.1109/isetc.2018.8583863.

[9] O. A. Aboulseoud and S. M. Ismail, "FPGA floating point fractional-order chaotic map image encryption," in *Proc. 31st Int. Conf. Microelectron. (ICM)*, Dec. 2019, doi: 10.1109/icm48031.2019.9021500.

[10] M. François, D. Defour, and P. Berthomé, "A pseudo-random bit generator based on three chaotic logistic maps and IEEE 754–2008 floating-point arithmetic," in *Lecture Notes in Computer Science*. Berlin, Germany: Springer, 2014, pp. 229–247, doi: 10.1007/978-3-319-06089-7_16.

[11] H. S. Hassan and S. M. Ismail, "CLA based floating-point adder suitable for chaotic generators on FPGA," in *Proc. 30th Int. Conf. Microelectron. (ICM)*, Dec. 2018, doi: 10.1109/icm.2018.8704074.

[12] T. Araujo, M. B. R. Cardoso, E. G. Nepomuceno, C. H. Llanos, and J. Arias-Garcia, "A new floating-point adder FPGA-based implementation using RN-coding of numbers," *Comput. Electr. Eng.*, vol. 90, Mar. 2021, Art. no. 106947, doi: 10.1016/j.compeleceng.2020.106947.

[13] J. Hormigo and J. Villalba, "New formats for computing with real-numbers under round-to-nearest," *IEEE Trans. Comput.*, vol. 65, no. 7, pp. 2158–2168, Jul. 2016, doi: 10.1109/TC.2015.2479623.

[14] T. Yoshida, H. Mori, and H. Shigematsu, "Analytic study of chaos of the tent map: Band structures, power spectra, and critical behaviors," *J. Stat. Phys.*, vol. 31, no. 2, pp. 279–308, May 1983, doi: 10.1007/bf01011583.

[15] D. J. Driebe, *Fully Chaotic Maps and Broken Time Symmetry* (Nonlinear Phenomena and Complex Systems), vol. 4. Amsterdam, The Netherlands: Springer, 1999.

[16] J. Hormigo and J. Villalba, "Optimizing DSP circuits by a new family of arithmetic operators," in *Proc. 48th Asilomar Conf. Signals, Syst. Comput.*, Nov. 2014, pp. 871–875, doi: 10.1109/acssc.2014.7094576.

[17] J. Hormigo and J. Villalba, "Measuring improvement when using HUB formats to implement floating-point systems under round-to-nearest," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 24, no. 6, pp. 2369–2377, Jun. 2016, doi: 10.1109/TVLSI.2015.2502318.

[18] C. Jiang and S. Wu, "A valid algorithm of converting chaos sequences to uniformity pseudo-random ones," in *Proc. Int. Symp. Inf. Eng. Electron. Commerce*, May 2009, pp. 295–298, doi: 10.1109/IEEC.2009.67.

[19] L. E. Bassham et al., "A statistical test suite for random and pseudo-random number generators for cryptographic applications," Nat. Inst. Standards Technol., Gaithersburg, MD, USA, Tech. Rep. 800-22, 2010, doi: 10.6028/nist.sp.800-22r1a.

[20] H. A. S. Ahmed, M. F. Zolkipli, S. M. Ismail, and Y. A. Alsariera, "Pseudo random bits' generator based on tent chaotic map and linear feedback shift register," *Adv. Sci. Lett.*, vol. 24, no. 10, pp. 7383–7387, Oct. 2018, doi: 10.1166/asl.2018.12946.

[21] M. Garcia-Bosque, C. Sanchez-Azqueta, G. Royo, and S. Celma, "Lightweight ciphers based on chaotic Map–LFSR architectures," in *Proc. 12th Conf. Ph.D. Res. Microelectron. Electron. (PRIME)*, Jun. 2016, pp. 1–4, doi: 10.1109/prime.2016.7519519.

[22] R. Hegger, H. Kantz, and T. Schreiber. *Nonlinear Time Series Routines*. Accessed: Apr. 21, 2022. [Online]. Available: https://www.pks.mpg.de/tisean/Tisean_3.0.1/index.html
[23] H. Kantz, "A robust method to estimate the maximal Lyapunov exponent of a time series," *Phys. Lett. A*, vol. 185, no. 1, pp. 77–87, Jan. 1994, doi: 10.1016/0375-9601(94)90991-1.
[24] G. Gugapriya, K. Rajagopal, A. Karthikeyan, and B. Lakshmi, "A family of conservative chaotic systems with cyclic symmetry," *Pramana*, vol. 92, no. 4, p. 48, Apr. 2019, doi: 10.1007/s12043-019-1719-1.
[25] M. Saber and M. M. Eid, "Low power pseudo-random number generator based on lemniscate chaotic map," *Int. J. Electr. Comput. Eng. (IJECE)*, vol. 11, no. 1, p. 863, Feb. 2021, doi: 10.11591/ijece.v11i1.pp863-871.
[26] V. R. Kopparthi, A. Kali, S. L. Sabat, K. K. Anumandla, R. Peesapati, and J. S. Armand Eyebe Fouda, "Hardware architecture of a digital piecewise linear chaotic map with perturbation for pseudorandom number generation," *AEU Int. J. Electron. Commun.*, vol. 147, Apr. 2022, Art. no. 154138, doi: 10.1016/j.aeue.2022.154138.

Samuel Souza da Silva was born in Belo Horizonte, Brazil. He received the B.Sc. degree in electrical engineering from the Universidade Federal de Minas Gerais (UFMG), Belo Horizonte, in 2021, where he is currently pursuing the master's degree in electrical engineering with the UFMG.

He is also a mid-level Software Product Analyst at Hexagon Mining, Belo Horizonte, Brazil.

Matheus Cardoso received the B.Sc. degree in control and automation engineering from the Universidade Federal de Minas Gerais (UFMG), Belo Horizonte, Brazil, in 2022.

His interests include software development, machine learning, data science, embedded systems, and chaos applications.

Mr. Cardoso spent a semester as an exchange student at the Southeast Missouri State University (SEMO), awarded a position on the Dean's List, in Spring 2020.

Lucas Nardo received the B.Eng. and M.Eng. degrees in electrical engineering from the Universidade Federal de São João del-Rei (UFSJ), São João del-Rei, Brazil, in 2017 and 2019, respectively. He is currently pursuing the Ph.D. degree in electrical engineering with the Universidade Federal de Minas Gerais (UFMG), Belo Horizonte, Brazil.

His research interests include dynamical systems, chaos, computer arithmetic, cryptography, and development of hardware architecture for embedded system applications. He is a Reviewer of IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II: EXPRESS BRIEFS, IEEE POTENTIALS, IEEE LATIN AMERICA TRANSACTIONS, *European Journal of Physics*, and *Inverse Problems*.

Erivelton Nepomuceno (Senior Member, IEEE) received the B.Eng. degree in electrical engineering from the Universidade Federal de São João del-Rei (UFSJ), São João del-Rei, Brazil, in 2001, and the Ph.D. degree in electrical engineering from the Universidade Federal de Minas Gerais (UFMG), Belo Horizonte, Brazil, in 2005.

From 2002 to 2021, he was an Associate Professor at the Universidade Federal de São João del-Rei, São João del-Rei. After that, he was appointed as an Assistant Professor at the Centre for Ocean Energy Research, Department of Electronic Engineering, Maynooth University, Maynooth, Ireland. He was a Visiting Research Fellow at the Instituto Tecnológico de Aeronáutica, São José dos Campos, Brazil, in 2005, the Imperial College London, London, U.K., in 2013 and 2014, the Saint Petersburg Electrotechnical University, Saint Petersburg, Russia, in 2019, and the City, University of London, London, in 2020 and 2021. His research interests include computer arithmetic, system identification, ocean energy, cryptography with chaos, and complex networks.

Dr. Nepomuceno was elected as the Secretary of the IEEE Technical Committee on Nonlinear Circuits and Systems. He has been elected as the Coordinator of the Technical Committee on System Identification and Data Science for the Brazilian Association of Automatic Control. He is the Deputy EiC of IEEE LATIN AMERICA TRANSACTIONS, and he is currently serving as an Associate Editor for IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II: EXPRESS BRIEFS, *Journal of Control, Automation and Electrical Systems*, and *Mathematical Problems in Engineering*.

Michael Hübner (Senior Member, IEEE) received the Dr.-Ing. and habilitation degrees from the Karlsruher Institut für Technologie (KIT), Karlsruhe, Germany, in 2007 and 2011, respectively.

He is a full Professor at the Brandenburgische Technische Universität (BTU), Cottbus, Germany, and the Chair of the Computer Engineering Group, BTU. His research interests include reconfigurable computing with applications in automotive systems.

Janier Arias-Garcia received the B.Eng. degree in physics engineering from the Universidad del Cauca (Unicauca), Popayán, Colombia, in 2007, and the M.Eng. and Ph.D. degrees in mechatronic systems from the Universidade de Brasília (UnB), Brasília, Brazil, in 2010 and 2014, respectively.

He is an Assistant Professor with the Department of Electronic Engineering (DELT), Universidade Federal de Minas Gerais (UFMG), Belo Horizonte, Brazil. He has developed architecture and custom hardware applications to implement numerical algorithms, involving the theory and practice of using reconfigurable hardware systems [field-programmable gate arrays (FPGAs)] and their synthesis tools to describe digital circuits, allowing for a controlled exchange of stability, numerical precision, and metrics, such as area, energy, and performance. He is currently working on digital hardware architecture for matrix calculations and classifiers, as well as cryptosystem architecture for embedded system applications.

Dr. Arias-Garcia is a member of the MACRO group.