1-1-2022

Design, Analysis, and Application of Flipped Product Chaotic System

Md Sakib Hasan
University of Mississippi

Partha Sarathi Paul
University of Mississippi

Anurag Dhungel
University of Mississippi

Maisha Sadia
University of Mississippi

Md Razuan Hossain
University of Mississippi

Recommended Citation

Hasan, M. S., Paul, P. S., Dhungel, A., Sadia, M., & Hossain, M. R. (2022). Design, analysis, and application of flipped product chaotic system. IEEE Access, 10, 125181–125193. https://doi.org/10.1109/ACCESS.2022.3225661

IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 3 October 2022, accepted 20 November 2022, date of publication 30 November 2022,
date of current version 5 December 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3225661

RESEARCH ARTICLE

Design, Analysis, and Application of Flipped Product Chaotic System

MD SAKIB HASAN, (Member, IEEE),
PARTHA SARATHI PAUL, (Graduate Student Member, IEEE),
ANURAG DHUNGEL, (Graduate Student Member, IEEE),
MAISHA SADIA, (Graduate Student Member, IEEE), AND
MD RAZUAN HOSSAIN, (Graduate Student Member, IEEE)
Department of Electrical and Computer Engineering, University of Mississippi, Oxford, MS 38677, USA
Corresponding author: Md Sakib Hasan (mhasan5@olemiss.edu)

ABSTRACT In this paper, a novel method is proposed to build an improved 1-D discrete chaotic map called flipped product chaotic system (FPCS) by multiplying the output of one map with the output of a vertically flipped second map. Two variants, each with nine combinations, are shown with trade-off between computational cost and performance. The chaotic properties are explored using the bifurcation diagram, Lyapunov exponent, Kolmogorov entropy, and correlation coefficient. The proposed schemes offer a wider chaotic range and improved chaotic performance compared to the constituent maps and several prior works of similar nature. Wide chaotic window and improved chaotic complexity are two desired characteristics for several security applications as these two characteristics ensure enhanced design space with elevated entropic properties. We present a general Field-Programmable Gate Array (FPGA) design framework for the hardware implementation of the proposed flipped-product schemes and the results show good qualitative agreement with the numerical results from MATLAB simulation. Finally, we present a new Pseudo Random Number Generator (PRNG) using the two variants of the proposed chaotic map and validate their excellent randomness property using four standard statistical tests, namely NIST, FIPS, TestU01, and Diehard.

INDEX TERMS Nonlinear dynamical systems, chaos, field-programmable gate arrays (FPGA), bifurcation diagram, lyapunov exponent, discrete-time map, random number generation (RNG).

I. INTRODUCTION

Since Lorenz's seminal work demonstrating chaotic motion on a strange attractor in 1963 [1], chaos has attracted a lot of attention from diverse fields of enquiry. Nonlinear problems attract the interest of researches from a wide range of disciplines including, biology, physics, chemistry, ecology and engineering [2], as most systems of nature are inherently nonlinear. Nonlinear dynamics describe the change of the state variables of a system over time. When the steady-state trajectory of a nonlinear deterministic dynamic system shows aperiodicity and extreme sensitivity to slight perturbation of initial state, we refer to this phenomenon as 'chaos'. When the parameters of a nonlinear deterministic dynamic system are tuned to the chaotic window, the output of the system never repeats and two initial states, even if they are infinitesimally close, drastically diverge and result in two completely uncorrelated steady-state trajectories. This extreme sensitivity to the initial state is popularly know as 'the butterfly effect' i.e. a tiny change caused by a butterfly flapping its wings in one region leading to a drastic change of weather in another distant part of the globe. These two defining features, namely, deterministic aperiodicity and sensitive dependence on the initial state render chaotic systems suitable for numerous applications including, data and image encryption [3], [4], [5], pseudo-random number generation (PRNG) [6], [7], [8], dynamical system modeling [9], [10], reconfigurable logic [11], [12], side channel attack mitigation [13], [14], secure communication [15], [16], logic obfuscation [17], [18] and so on.

The associate editor coordinating the review of this manuscript and approving it for publication was Chao-Yang Chen.

Based on the nature of the time steps, chaotic systems can be divided into two categories: i) Continuous-time ii) Discrete-time chaotic systems. Logistic [19], sine, and tent maps are classic examples of one-dimensional (1-D) discrete-time maps defined by the evolution of one state variable according to one equation in discrete time steps whereas, Henon map [20] is an example of a famous two-dimensional discrete-time map. On the other hand, Lorenz system [1] is a famous example of a multi-dimensional continuous-time chaotic system as it is constituted of three coupled ordinary differential equations that define the evolution of three state variables as a function of time in continuous domain. While all classes have found their way in numerous security applications, 1-D discrete-time maps have gained popularity for their simplicity in implementation. In this work, we confine our discussion to 1-D discrete-time maps.

One weakness of 1-D discrete-time maps comes from fact that most common systems are not robust since they exhibit strong chaotic property only within a very narrow range of parameter values. Beyond this narrow window, the chaotic properties of these maps degrade and eventually disappear. That poses a problem in the security applications as the system may deviate from narrow chaotic window due to undesirable parameter fluctuations, causing a security compromise. Hence, robust chaos i.e. absence of periodic windows and coexisting attractors in some neighborhood in the parameter space of a dynamical system [21] is desirable which can mitigate such issues [22], [23]. Another shortcoming is that the entropy of the generated sequence even within this narrow range is not very high as measured by different entropy metrics such as Lyapunov exponent, Kolmogorov entropy etc. Due to poor entropic properties, many chaotic system behavior can be analyzed and predicted [24], [25] compromising their potential for security applications. Multiple schemes have been reported to improve the chaotic performance of discrete-time 1D maps. Deng et al. proposed a feedback control method to mitigate performance degradation of digital chaotic systems [26]. Li et al. introduced a reseeding-mixing method [27] to build high throughput PRNG using logistic map but it lacks parameter reconfigurability since the design was optimized for a single parameter value with high chaotic entropy. Another reported method is to widen the chaotic region of a map by modulating the chaotic parameter within a narrow high-performance range through a linear transformation of the output from a second map [28], [29]. This scheme, however, does not necessarily improve the Lyapunov exponent and is susceptible to performance degradation through perturbation in the linear transformation block. Cascading multiple maps under certain conditions improves chaotic entropy as shown in [29], [30] but this scheme does not necessarily improve the chaotic parameter range and can lead to unpredictable behavior for two maps with unequal parameter value [31]. An exponential chaotic map was introduced in [23] which exhibits robust chaos but the entropy is limited to the highest value achievable by its seed maps and it requires computationally expensive exponentiation and logarithmic operation making hardware implementation in resource constrained applications difficult.

In this work, we propose a general framework of 1-D chaotic maps called flipped product chaotic map (FPCM) where the output of one map is multiplied with the output of a vertically flipped second map to get the final output. With the help of bifurcation plot and chaotic entropy measures, it is demonstrated that FPCM offers a wider chaotic region with improved chaotic entropy than the constituent maps henceforth referred to as seed maps. We first propose the basic scheme called Basic Flipped-Product Chaotic System (BFPCS) and then show an improved version called Enhanced Flipped-Product Chaotic System (EFPCS) requiring more computational cost. A field-programmable gate array (FPGA) design is presented to demonstrate a possible hardware implementation of these schemes. Finally, we introduce a new pseudo-random number generator (PRNG) using the novel map and demonstrate its excellent properties using four standard statistical tests, namely NIST, FIPS, TestU01, and Diehard.

The remainder of the paper is organized as follows: the seed maps are introduced in section-II. The general scheme is presented in section-III followed by two variants in section-IV and section-V accompanied with requisite analysis of their chaotic properties using transfer curve, bifurcation diagram, Lyapunov exponent, Kolmogorov entropy, and correlation coefficient. section-VI compares the proposed work with similar prior works. An FPGA implementation using Verilog HDL (hardware description language) is presented in section-VII. Section-VIII introduces a novel PRNG scheme using the proposed map and validates its excellent properties using standard statistical tests. Finally, section-IX gives the concluding remarks with possible future direction of our research.

II. SEED MAPS

This section reviews three existing 1-D chaotic maps namely, logistic, tent, and sine maps as background. They will be used as seed maps to generate new chaotic maps in Section-IV and section-V. For ease of comparison, we are using the normalized versions of these seed maps such that their domain, range, and parameter values are within [0, 1].

Logistic map can be mathematically defined as,

$$x_{i+1} = \mathcal{L}(x_i) = 4rx_i(1 - x_i). \tag{1}$$

where r is the control parameter and $r \in [0, 1]$.

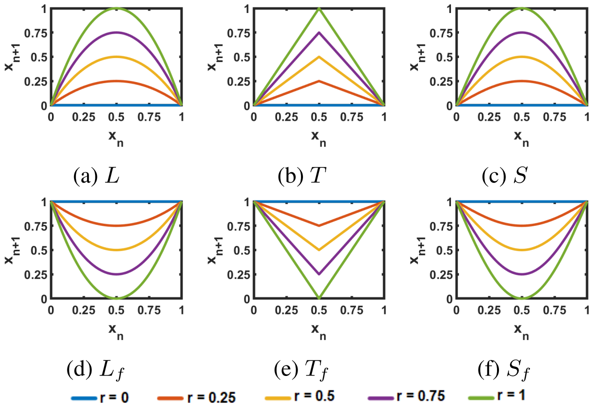
Tent map can be mathematically defined as,

$$x_{i+1} = \mathcal{T}(x_i) = \begin{cases} 2rx_i & when,\ x_i < 0.5 \\ 2r(1 - x_i) & when,\ x_i \geq 0.5 \end{cases} \tag{2}$$

where r is the control parameter and $r \in [0, 1]$.
Sine map can be mathematically defined as,

$$x_{i+1} = \mathcal{S}(x_i) = r\sin(\pi x_i) \tag{3}$$

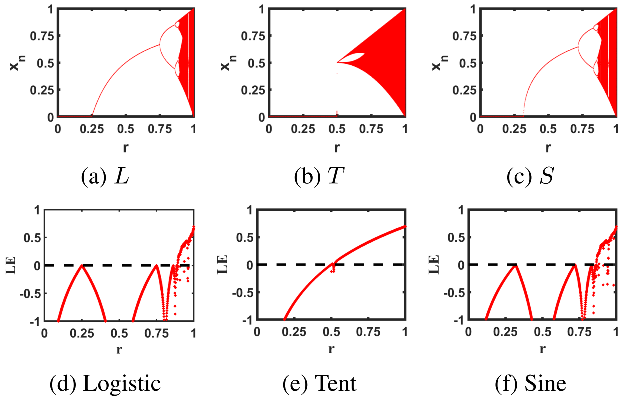
where r is the control parameter and $r \in [0, 1]$.



(a) L (b) T (c) S

(d) L_f (e) T_f (f) S_f

FIGURE 1. Transfer characteristics of different seed maps (a-c) and corresponding flipped maps (d-f).



(a) L (b) T (c) S

(d) Logistic (e) Tent (f) Sine

FIGURE 2. Bifurcation diagram and Lyapunov exponent of three seed maps.

The transfer curve shows the input-output characteristics of a map for different values of the control parameter. The transfer curves for the three seed maps are shown in Fig. 1(a-c) and the transfer curves for their corresponding flipped maps are shown in Fig. 1(d-f). The effect of a control parameter on a dynamical system can be visualized with a bifurcation diagram where for each parameter value, a long sequence of steady-state output values is plotted. The advantage of the bifurcation diagram is that it clearly shows the period doubling process which causes the system to transition from a fixed point to a periodic region and eventually to a chaotic region while the control parameter is varied. The chaotic property in the output is evaluated with a widely used metric called the Lyapunov exponent (LE). A positive LE demonstrates the existence of chaotic behavior [2]. Fig. 2 plots the bifurcation diagrams and LEs of the logistic, sine, and tent maps with the change of their control parameters. As can be observed, the logistic, sine, and tent maps have chaotic behaviors when $r \in [0.89, 1]$, $r \in [0.87, 1]$, and $r \in (0.5, 1)$, respectively.

III. PROPOSED SCHEME

As shown by Feigenbaum [32], any differential unimodal (V-shape or inverted V-shape) map can potentially generate a chaotic sequence. In this work, we propose a general scheme to combine two 1-D maps of opposite shape (V and inverted V-shape) with one common parameter into a single map called flipped product chaotic map (FPCM) with improved chaotic properties. As constituent seed maps, we consider three common 1-D inverted-V shape discrete maps, namely logistic, tent, and sine maps. Then we introduce the concept of flipped map which is obtained by vertically flipping these maps. If we have a seed map, $S(r, x)$ with highest value of S_{max}, its flipped version $S_f(r, x)$ can be written as, $S_f(r, x) = S_{max} - S(r, x)$. For the three considered seed maps, $S_{max} = 1$. Now, Fig. 3 shows the schematic of the proposed flipped product chaotic system (FPCS). The output of a seed map is multiplied with the output of a flipped map. The control parameter, r, remains the same for both the seed map and flipped map. There is a scaling factor A, which is chosen to ensure that the maximum output of the FPCM does not exceed 1. This FPCM output is then fed back as input for the next iteration to build the chaotic oscillator system henceforth called FPCS (flipped product chaotic system). The flipped product chaotic map can be mathematically defined as,

$$x_{i+1} = FPCM(r, x_i) = (1/A) * S(r, x_i) * S_f(r, x_i), \quad (4)$$

where r is the control parameter. It is assumed that the ranges of state variable and control parameters for all considered maps are in the interval [0,1].

In the next section, we introduce the basic flipped product chaotic system (BFPCS) where A is a global constant for a particular map irrespective of the value of the control parameter, r. Then in Section-V, we introduce the enhanced flipped product chaotic system (EFPCS) where A is a function of r which leads to further improvement of the chaotic properties.

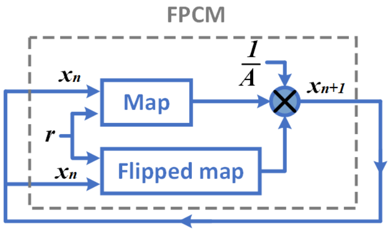


FIGURE 3. The schematic of the FPCS (flipped product chaotic system).

IV. BASIC FLIPPED-PRODUCT CHAOTIC SYSTEM (BFPCS)

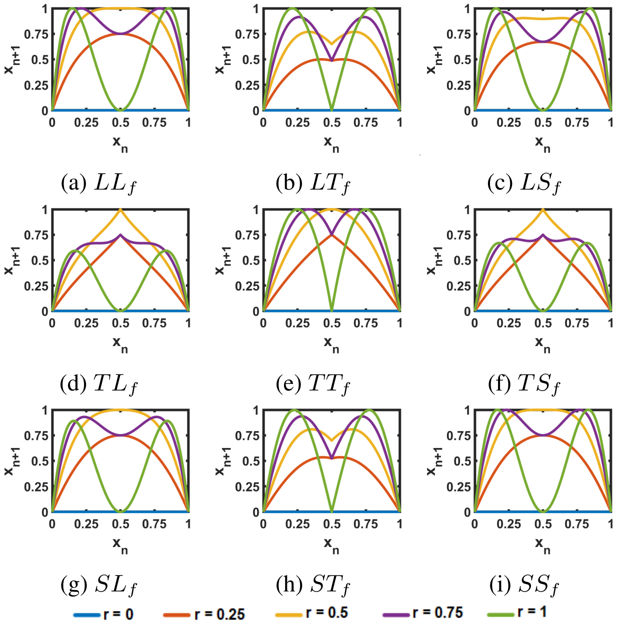
In basic flipped product chaotic system (BFPCS), A is considered a constant, independent of the control parameter, r. For any seed map $S(r, x)$, A is set as the maximum possible value of $S * S_f(r, x)$ for all $x, r \in [0, 1]$.

A. MATHEMATICAL EXPRESSION

Using three constituent seed maps (logistic, tent, and sine), there can be nine possible FPCMs. In Table-1, the mathematical expressions for these nine combinations along with the corresponding scaling factor A are shown.

TABLE 1. Mathematical expression for BFPCS.

Map name	Symbol	Mathematical expression	A
Logistic-flipped-Logistic	$LL_f(r,x)$	$x_{n+1} = (1/A) * L(r,x_n) * L_f(r,x_n)$	0.25
Logistic-flipped-Tent	$LT_f(r,x)$	$x_{n+1} = (1/A) * L(r,x_n) * T_f(r,x_n)$	0.385
Logistic-flipped-Sine	$LS_f(r,x)$	$x_{n+1} = (1/A) * L(r,x_n) * S_f(r,x_n)$	0.279
Tent-flipped-Logistic	$TL_f(r,x)$	$x_{n+1} = (1/A) * T(r,x_n) * L_f(r,x_n)$	0.25
Tent-flipped-Tent	$TT_f(r,x)$	$x_{n+1} = (1/A) * T(r,x_n) * T_f(r,x_n)$	0.25
Tent-flipped-Sine	$TS_f(r,x)$	$x_{n+1} = (1/A) * T(r,x_n) * S_f(r,x_n)$	0.25
Sine-flipped-Log	$\mathrm{SL}_f(r,x)$	$x_{n+1} = (1/A) * S(r,x_n) * L_f(r,x_n)$	0.25
Sine-flipped-Tent	$\mathrm{ST}_f(r,x)$	$x_{n+1} = (1/A) * S(r,x_n) * T_f(r,x_n)$	0.358
Sine-flipped-Sine	$\mathrm{SS}_f(r,x)$	$x_{n+1} = (1/A) * S(r,x_n) * S_f(r,x_n)$	0.25



(a) LL_f (b) LT_f (c) LS_f

(d) TL_f (e) TT_f (f) TS_f

(g) SL_f (h) ST_f (i) SS_f

FIGURE 4. Transfer characteristics of different BFPCS maps (a-i).

B. TRANSFER CHARACTERISTICS

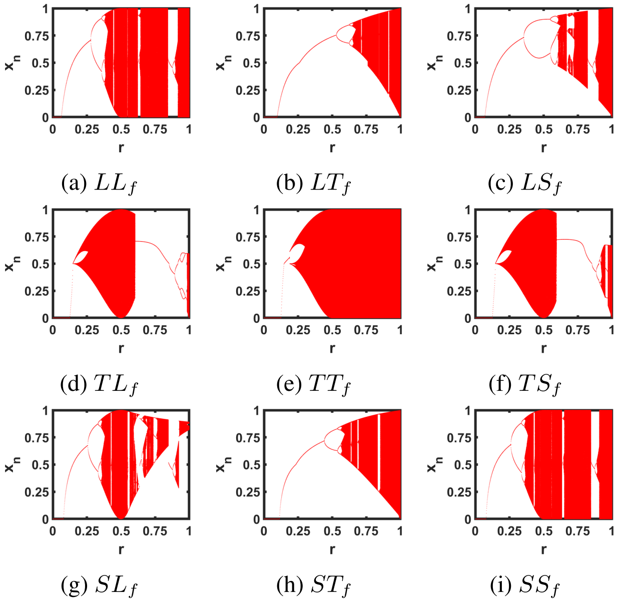
The transfer curves for the nine possible combinations of FPCM are shown in Fig. 4. As we can see from these figures, the output of these FPCMs cover the entire range with more oscillation due to their bimodal characteristics in contrast to the unimodal transfer curve (Fig. 1) of their constituent seed maps which lie at the core of their improved characteristics.

C. BIFURCATION DIAGRAM

Fig. 5 shows the bifurcation diagrams of nine combinations of BFPCS. It shows that the BFPCS has a wider chaotic range with higher signal swing compared to the seed maps when both constituent maps are of the same type, namely Logistic-flipped-Logistic (LL_f), Tent-flipped-Tent (TT_f), and Sine-flipped-Sine (SS_f). However, when the two maps are different, the improvement is less pronounced. Later, in Section-V, we will introduce an improvement scheme that gets rid of this problem.

D. LYAPUNOV EXPONENT (LE)

A characteristic of the chaotic system is the sensitive dependence on its initial condition. On average, two adjacent orbits, generating from slightly different initial conditions, will diverge exponentially fast under chaotic operating conditions. Lyapunov exponent is a widely used parameter to measure this exponential divergence capturing the system's sensitive dependence on the initial condition. For a discrete-time chaotic map $f(x)$, it is defined as [2],



(a) LL_f (b) LT_f (c) LS_f

(d) TL_f (e) TT_f (f) TS_f

(g) SL_f (h) ST_f (i) SS_f

FIGURE 5. Bifurcation diagrams of nine BFPCS.

$$LE = \lim_{n\to\infty} \frac{1}{n} \sum_{i=0}^{n-1} ln|f'(x_i)|. \tag{5}$$

Here, n is the total number of iterations and $f'(x_i)$ indicates the first derivative of the map function at its i^{th} iteration. If the operating region consists of stable fixed points or cycles, LE is negative whereas, for chaotic attractors, its value is positive [2]. Bigger positive LE values indicate faster divergence of output trajectories and consequently, better chaotic performance.

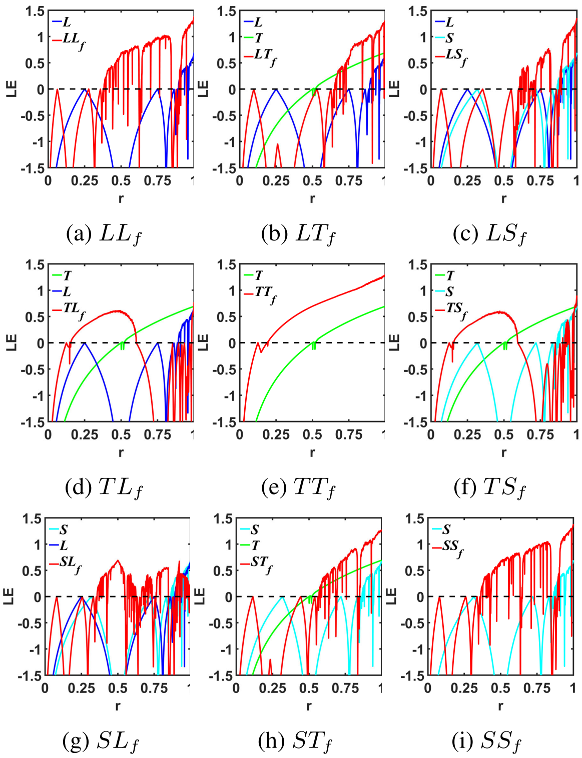
Fig. 6 shows the comparison of LE value between the nine BFPCMs and corresponding seed maps. It is found that in general, they have positive LE value within a much wider parameter window and the values are significantly higher compared to their constituent seed maps.

E. KOLMOGOROV ENTROPY (KE)

KE is another useful metric to quantitatively measure the complexity in a given sequence [33]. It captures how much extra information is required to predict the next output of a dynamical system given its previous outputs and is defined in 6 [33].

$$KE = -\lim_{\tau\to\infty}\lim_{\epsilon\to\infty}\lim_{d\to\infty}\frac{1}{n}\sum_{i_1,\ldots,i_d} p(i_1, i_2, \ldots, i_d) \times ln(p(i_1, i_2, \ldots, i_d)) \tag{6}$$

Positive KE implies chaotic unpredictability and a higher value implies increased unpredictability. We have generated

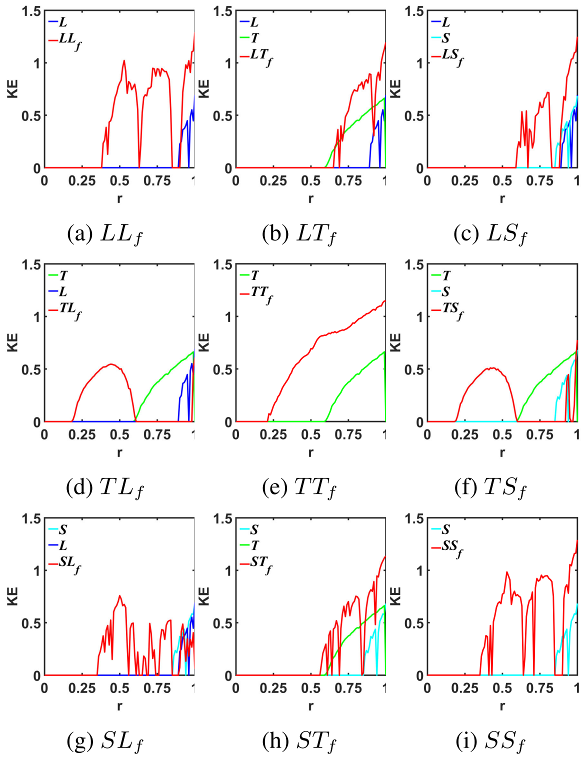


(a) LL_f (b) LT_f (c) LS_f

(d) TL_f (e) TT_f (f) TS_f

(g) SL_f (h) ST_f (i) SS_f

FIGURE 6. LE results for different seed maps and FPCS.



(a) LL_f (b) LT_f (c) LS_f

(d) TL_f (e) TT_f (f) TS_f

(g) SL_f (h) ST_f (i) SS_f

FIGURE 7. KE values of different seed maps and FPCS.

sequences with 13,000 iterations, truncated the first 1000 transient values, and computed the KE based on the remaining sequence. Fig. 7 shows the comparison of KE value between nine BFPCMs and corresponding seed maps. As this figure shows, BFPCM, in general, has a positive KE value within a much wider parameter window and the values are significantly higher compared to the traditional seed maps.

F. CORRELATION COEFFICIENT (CC)

The correlation between two sequences of data X and Y can be measured by Pearson's correlation coefficient $CC(X, Y)$ defined as [28],

$$CC(X, Y) = \frac{E[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X \sigma_Y} \tag{7}$$

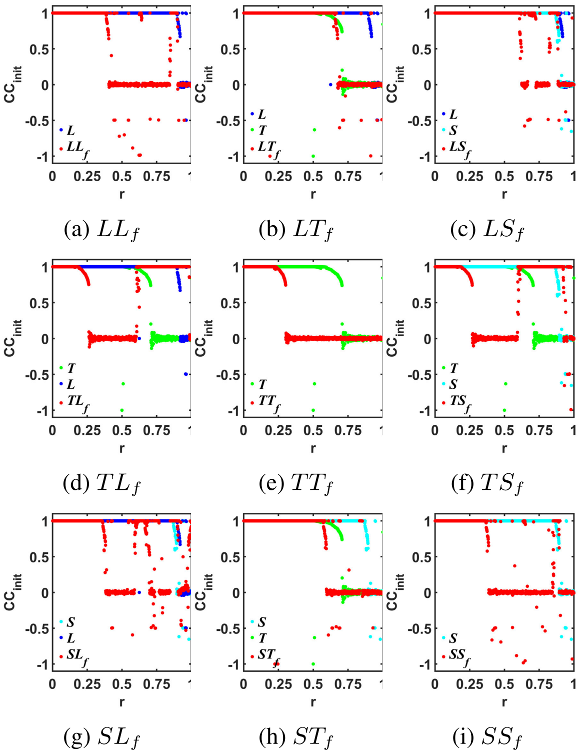
Here, μ and σ represent the mean value and standard deviation, respectively. The expectation operator is denoted by 'E[]'. The value of CC is between -1 and $+1$. If the correlation value is close to $+1$/-1, then two data sequences are highly correlated i.e. their relationship comes close to a linear dependence. On the other hand, a correlation value close to 0 indicates no discernible relationship between the data sequences. Here, we have used CC to measure the sensitive dependence of a chaotic map on initial value and parameters. Fig. 8 shows the value of CC between a pair of long sequences generated from a slightly different initial condition for different values of parameter r. Within the non-chaotic window, we expect this value to be close to 1 due to the convergence of both sequences. However, inside the chaotic window, a slight change in initial condition eventually leads to exponential divergence of both sequences and lead to an almost uncorrelated sequence i.e. $CC \approx 0$. Similarly, Fig. 9 shows the CC between pairs of sequences generated from slightly perturbed parameter values. Fig. 8 and Fig. 9 clearly show that the BFPCM maps have wider chaotic region compared to the seed maps with high sensitivity to perturbation in initial value and parameter and the chaotic window matches with the bifurcation diagrams shown in Fig. 5.

V. ENHANCED FLIPPED-PRODUCT CHAOTIC SYSTEM (EFPCS)

In this section, we propose an improved configuration called enhanced flipped product chaotic system (EFPCS) which gets rid of some of the limitations of the initial scheme, BFPCS. In contrast to BFPCS, the scaling factor, A is not a constant for a particular map. Rather it varies as a function of r. We came up with an analytical or semi-analytical expression for $A(r)$, which gives the maximum value of the product for that particular r. This ensures a very wide chaotic region with good entropic properties for all combinations.

A. MATHEMATICAL EXPRESSION

We illustrate this scheme for Logistic-flipped-Tent (LT_f). For LT_f, we define the product, $p(r, x)$ as $p(r, x) = L(r, x) * T_f(r, x)$. The value of x at which we get the maximum

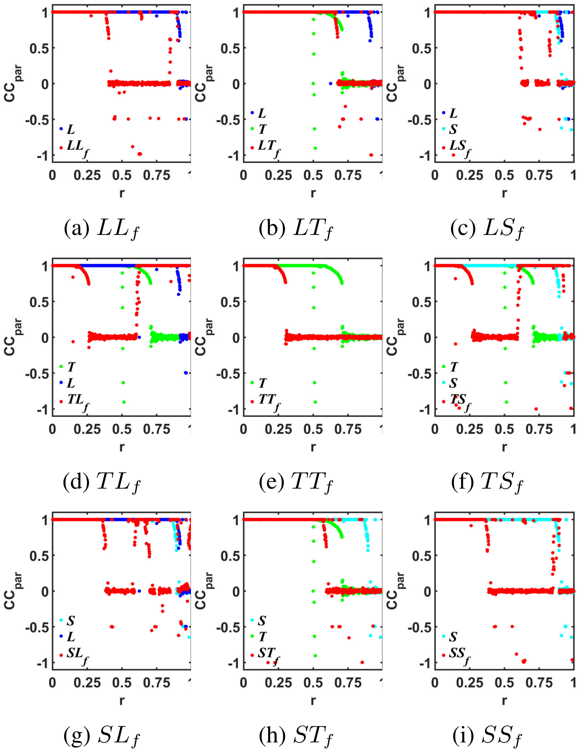


(a) LL_f (b) LT_f (c) LS_f

(d) TL_f (e) TT_f (f) TS_f

(g) SL_f (h) ST_f (i) SS_f

FIGURE 8. CC measurement for initial value sensitivity of BFPCS.

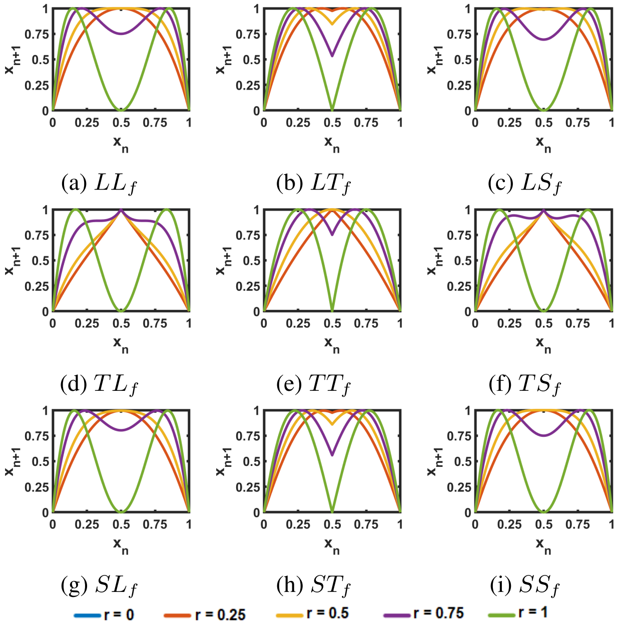


(a) LL_f (b) LT_f (c) LS_f

(d) TL_f (e) TT_f (f) TS_f

(g) SL_f (h) ST_f (i) SS_f

FIGURE 9. CC measurement for parameter sensitivity of BFPCS.



(a) LL_f (b) LT_f (c) LS_f

(d) TL_f (e) TT_f (f) TS_f

(g) SL_f (h) ST_f (i) SS_f

FIGURE 10. Transfer characteristics of different EFPCS maps (a-i).

value of this product for a particular r can be evaluated as $x_{max} = (2r + 1 - \sqrt{(2r + 1)^2 - 6r})/6r$. So, the scaling factor, A would be $A = p(r, x_{max})$. Similarly, for all other combinations, the crucial step is the calculation of the corresponding x_{max}. Table 2 gives the mathematical expressions for all nine combinations along with corresponding A and x_{max}. For some combinations such as LS_f, there is no close-form analytical solution and in those cases, we have approximated the solution with simple piece-wise linear or exponential functions along with a correction factor (cf) which is set to 1.005.

B. TRANSFER CHARACTERISTICS

The transfer curves for the nine possible combinations derived using Table 2 are shown in Fig. 10. As we can see from these figures, unlike BFPCMs, the output of these EFPCMs has a higher signal swing for all parameter values (4) which lead to its enhanced performance as demonstrated in the following subsections.

C. BIFURCATION DIAGRAM

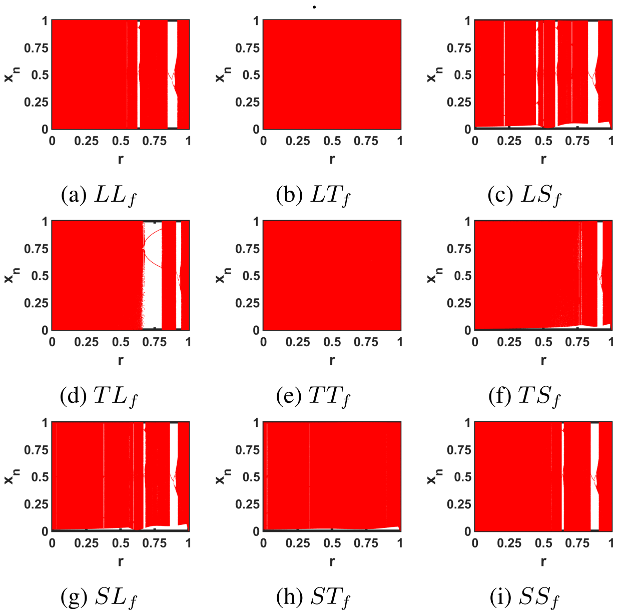
Fig. 11 shows the bifurcation diagrams of nine combinations of EFPCS. These maps have significantly wider (close to 100% in most cases) chaotic range with higher signal swing compared to seed maps(Fig. 2) and BFPCS (Fig. 5).

D. LYAPUNOV EXPONENT (LE)

Fig. 12 shows the Lyapunov exponents for all nine combinations using EFPCS. They have positive LE values across almost the entire parameter window and the values are significantly higher compared to their constituent seed maps. These figures also show marked improvement in LE compared to BFCPS (Fig. 6).

TABLE 2. Mathematical expression for EFPCS.

Symbol	Expression	A	x_{max}
$LL_f(r,x)$	$x_{n+1} = (1/A) * L(r,x_n) * L_f(r,x_n)$	$A = L(r,x_{max}) * L_f(r,x_{max})$	$x_{max} = \begin{cases} 0.5 & ; r < 0.5 \\ \frac{2r-\sqrt{2r(2r-1)}}{4r} & ; r \geq 0.5 \end{cases}$
$LT_f(r,x)$	$x_{n+1} = (1/A) * L(r,x_n) * T_f(r,x_n)$	$A = L(r,x_{max}) * T_f(r,x_{max})$	$x_{max} = (2r+1-\sqrt{(2r+1)^2-6r})/6r$
$LS_f(r,x)$	$x_{n+1} = (1/A) * L(r,x_n) * S_f(r,x_n)$	$A = cf * L(r,x_{max}) * S_f(r,x_{max})$	$x_{max} = \begin{cases} 0.5 & ; r < 0.0.447 \\ c_1 exp(-r/\tau) + c_2 & ; r \geq 0.447 \end{cases}$ $[c_1 = 4.09, c_2 = 0.14, \tau = 0.18]$
$TL_f(r,x)$	$x_{n+1} = (1/A) * T(r,x_n) * L_f(r,x_n)$	$A = T(r,x_{max}) * L_f(r,x_{max})$	$x_{max} = \begin{cases} 0.5 & ; r < 0.8 \\ \frac{2r-\sqrt{r(4r-3)}}{6r} & ; r \geq 0.8 \end{cases}$
$TT_f(r,x)$	$x_{n+1} = (1/A) * T(r,x_n) * T_f(r,x_n)$	$A = T(r,x_{max}) * T_f(r,x_{max})$	$x_{max} = \begin{cases} 0.5 & ; r < 0.5 \\ \frac{1}{4r} & ; r \geq 0.5 \end{cases}$
$TS_f(r,x)$	$x_{n+1} = (1/A) * T(r,x_n) * S_f(r,x_n)$	$A = cf * T(r,x_{max}) * S_f(r,x_{max})$	$x_{max} = \begin{cases} 0.5 & ; r < 0.774 \\ -0.371r + 0.548 & ; r \geq 0.774 \end{cases}$
$SL_f(r,x)$	$x_{n+1} = (1/A) * S(r,x_n) * L_f(r,x_n)$	$A = cf * S(r,x_{max}) * L_f(r,x_{max})$	$x_{max} = \begin{cases} 0.5 & ; r < 0.552 \\ c_1 exp(-r/\tau) + c_2 & ; r \geq 0.552 \end{cases}$ $[c_1 = 14.54, c_2 = 0.139, \tau = 0.149]$
$ST_f(r,x)$	$x_{n+1} = (1/A) * S(r,x_n) * T_f(r,x_n)$	$A = cf * S(r,x_{max}) * T_f(r,x_{max})$	$x_{max} = -0.274r + 0.5$
$SS_f(r,x)$	$x_{n+1} = (1/A) * S(r,x_n) * S_f(r,x_n)$	$A = L(r,x_{max}) * L_f(r,x_{max})$	$x_{max} = \begin{cases} 0.5 & ; r < 0.5 \\ \frac{asin(1/(2r))}{\pi} & ; r \geq 0.5 \end{cases}$

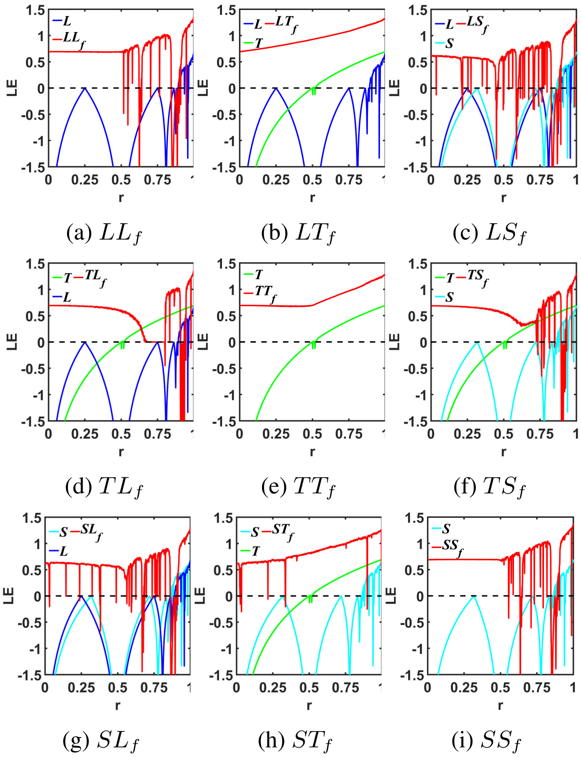


(a) LL_f (b) LT_f (c) LS_f

(d) TL_f (e) TT_f (f) TS_f

(g) SL_f (h) ST_f (i) SS_f

FIGURE 11. Bifurcation diagrams of nine EFPCS.



(a) LL_f (b) LT_f (c) LS_f

(d) TL_f (e) TT_f (f) TS_f

(g) SL_f (h) ST_f (i) SS_f

FIGURE 12. LE results for different seed maps and EFPCS.

E. KOLMOGOROV ENTROPY (KE)

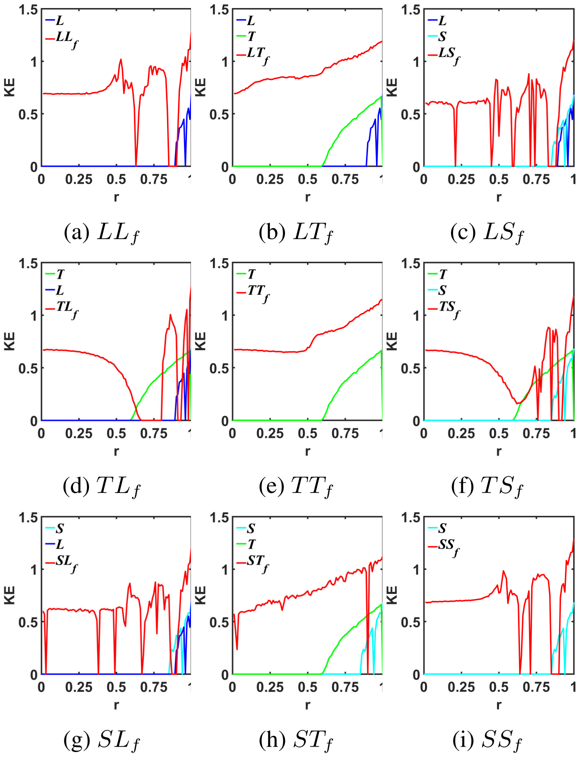
Fig. 13 shows the comparison of KE value between nine combinations of EFPCS and corresponding seed maps. As evident from this figure, EFPCS has a positive KE value across almost the entire parameter range with higher values compared to their constituent seed maps. The KE values also show significant improvement compared to BFPCS (Fig. 7).

F. CORRELATION COEFFICIENT (CC)

Fig. 14 and Fig. 15 show the CCs capturing the initial value and parameter sensitivity, respectively for all nine combinations using EFPCS. These figures are consistent with our findings from the bifurcation diagram, LE, and KE and clearly demonstrate chaotic operation (implied by CC value of 0) within a much wider window compared to seed maps and BFCPS (Fig. 8 and Fig. 9).

VI. PERFORMANCE COMPARISON

The first advantage of the proposed design is its much wider chaotic region i.e. increase in the quantity of chaotic design space. The second advantage is higher entropic properties across wider chaotic range i.e. improvement of quality of chaotic operation. Due to finite precision arising from discretized digital representation, only finite number of distinct parameters values are available within a certain range of real numbers. For a particular digital implementation, if a system has p parameters and each parameter can have N distinct values, then the entire parameter space (EPS) can be defined

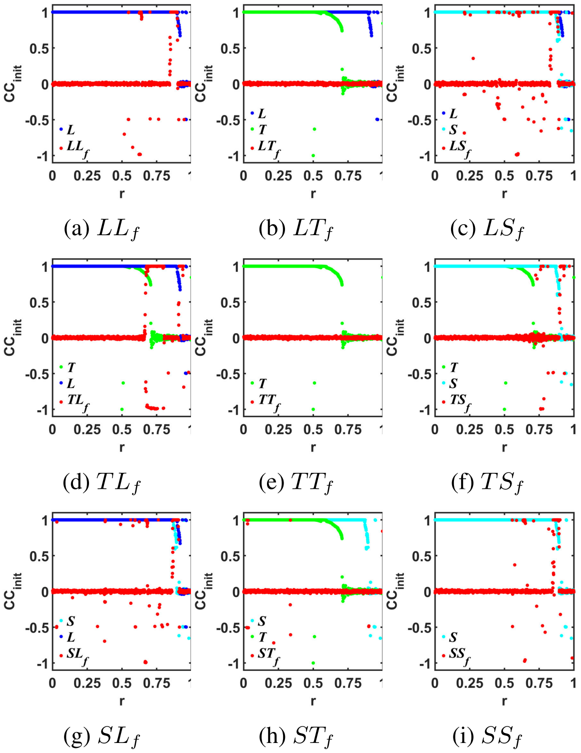


(a) LL_f (b) LT_f (c) LS_f

(d) TL_f (e) TT_f (f) TS_f

(g) SL_f (h) ST_f (i) SS_f

FIGURE 13. KE values of different seed maps and EFPCS.

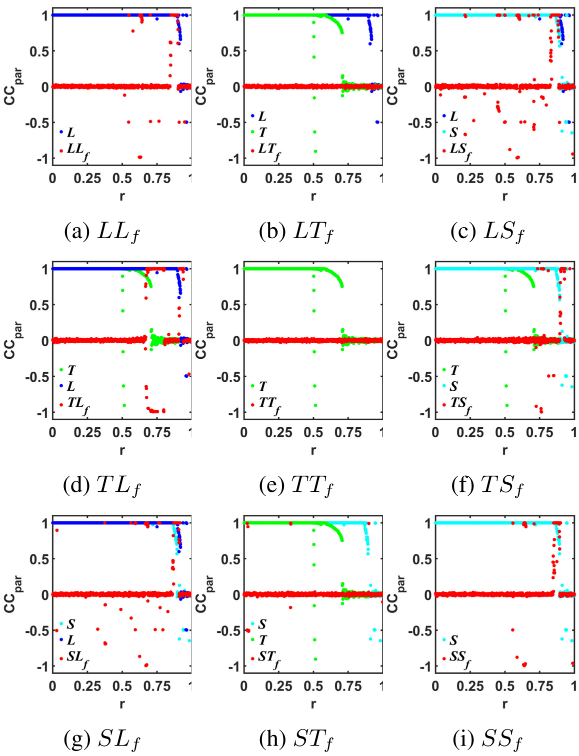


(a) LL_f (b) LT_f (c) LS_f

(d) TL_f (e) TT_f (f) TS_f

(g) SL_f (h) ST_f (i) SS_f

FIGURE 14. CC measurement for initial value sensitivity of EFPCS.



(a) LL_f (b) LT_f (c) LS_f

(d) TL_f (e) TT_f (f) TS_f

(g) SL_f (h) ST_f (i) SS_f

FIGURE 15. CC measurement for parameter sensitivity of EFPCS.

as, $EPS = N^p$. A subset of this space is chaotic which we call chaotic parameter space (CPS). Chaotic ratio (CR) is defined as the ratio of CPS to EPS [35].

$$CR(\%) = \frac{CPS}{EPS} \times 100 \tag{8}$$

For quality assessment, we are averaging LE, KE, and the absolute value of CC across the chaotic region to come up with a single global metric for each entropy measure. Higher average LE (ALE), and average KE (AKE) imply better entropic properties. Similarly, a lower average CC magnitude (ACC) closer to zero implies more initial state sensitivity i.e. better chaotic quality. We also report the maximum value of LE, and KE (MLE and MKE) and minimum absolute value of two types of CC (mCC). In addition, the dynamic swing range of the steady state output voltage inside the chaotic region should be as close to highest output range, R as possible. This is captured by a metric called average normalized dynamic range (ANDR) [35] which is defined as,

$$ANDR(\%) = (\frac{1}{CPS} \sum_{i \in CPS} \frac{V^i_{max} - V^i_{min}}{R}) \times 100. \tag{9}$$

Table 3 compares our proposed design, EFPCS (in bold) with the three basic seed maps as well as four previous works, namely ZBC [34], CCS [30], DPCCS [28] and ECM [23] using the above mentioned metrics and it shows significant improvement considering all aspects of chaotic operation.

VII. FPGA IMPLEMENTATION

A. FPGA DESIGN OF FPCS

As a representative example to show the simplicity of FPCS implementation, we have used Verilog to implement

TABLE 3. Comparison of chaotic performance.

Design	ALE	MLE	AKE	MKE	ACC_{init}	mCC_{init}	ACC_{par}	mCC_{par}	$CR(\%)$	$ANDR(\%)$
Logistic	0.388	0.694	0.397	0.693	0.244	1.1×10^{-4}	0.247	4.14×10^{-5}	9.89	76.25
Tent	0.393	0.693	0.334	0.67	0.391	1.47×10^{-5}	0.387	4.4×10^{-5}	49.05	42.29
Sine	0.408	0.689	0.417	0.681	0.195	4.55×10^{-5}	0.193	3.12×10^{-6}	11.89	75.09
ZBC(LT) [34]	0.684	0.71	0.658	0.693	0.0072	4.55×10^{-5}	0.0068	1.32×10^{-6}	100	99.89
CCS(LT) [30]	0.788	1.27	0.684	1.14	0.093	1.32×10^{-4}	0.08	1.84×10^{-4}	14.3	73.21
DPCCS(LT) [28]	0.45	0.683	0.464	0.964	0.272	2.05×10^{-5}	0.272	5.05×10^{-5}	100	59.74
ECM(LT) [23]	0.676	0.695	0.653	0.695	0.007	3.32×10^{-5}	0.0064	4.46×10^{-5}	100	99.99
EFPCS(LT_f)	**0.949**	**1.326**	**0.91**	**1.19**	**0.007**	$\mathbf{1.61 \times 10^{-6}}$	**0.0067**	$\mathbf{1.52 \times 10^{-6}}$	**100**	**99.99**

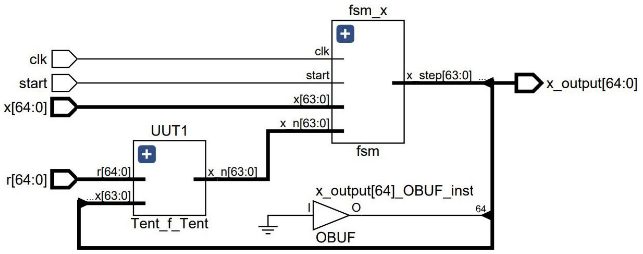


FIGURE 16. Schematic of FPGA implementation.

Tent-flipped-Tent (TT_f) in FPGA using both BFPCS and EFPCS. The state variable and control parameter for all considered maps are real numbers in the interval [0,1]. To represent them in hardware, real numbers between 0 and 1 are divided into $2^{64} + 1$ states and each value is represented by a 65-bit binary number. The 64 least significant bits (LSB) are used to represent [0,1) and the most significant bit (MSB) is used to include 1. The circuit has two 65-bit inputs r and x. The output is also a 65-bit number denoted by x_{output} as shown in Fig. 16.

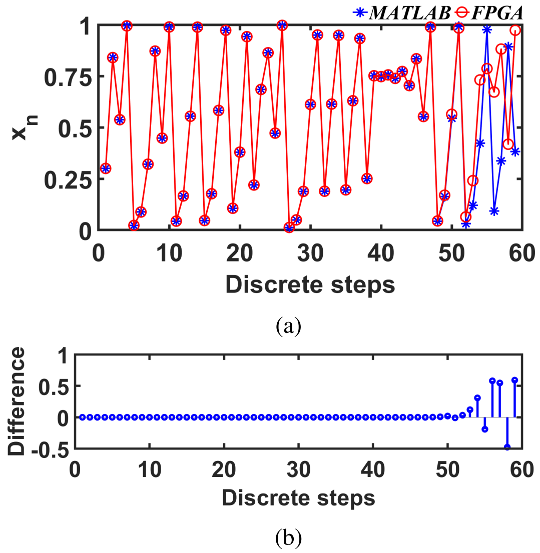
There are two modules that make up the digital circuit as shown in Fig. 16:

1. Tent_f_Tent: This module implements the mathematical operation needed to compute FPCM. Here, we show results for Tent-flipped-Tent (TT_f) map using both BFPCS or EFPCS scheme.

2. FSM: This module is used to store the value of the state variable x_i of the chaotic system as its state. In the first round, the initial state (x_0) is defined by the user defined input. After the device user presses the start button, the circuit updates the state in every clock cycle with the output of the module 'Tent_f_Tent'based on the previous state.

B. RESULTS

We have computed the first 60 iterations for TT_f map in both MATLAB and FPGA and compared these values in Fig. 17. The sequence produced by MATLAB and FPGA start diverging around 50^{th} iteration for $x_0 = 0.3$ and $r = 0.5$ as shown in figure 17. This is inevitable given the different number representation method between 64-bit IEEE-754 floating-point representation [36] in MATLAB and our fixed point FPGA implementation. However, in security applications, the more important consideration is having a chaotic sequence with good long-term entropic characteristics which can be evaluated by metrics such as LE, KE, and CC. We have calculated these metrics for different values of r for MATLAB and FPGA implementation of TT_f using both BFPCS and EFPCS. The comparison results between MATLAB and FPGA for BFPCS and EFPCS are shown in Fig. 18 and Fig. 19, respectively. As it clearly shows, FPGA results match very well with MATLAB results over the entire range demonstrating their functional equivalence.



(a)

(b)

FIGURE 17. Comparison of trajectory between FPGA vs MATLAB Simulation. Here, $x_0 = 0.3, r = 0.5$.

VIII. PRNG USING FPCS

High quality PRNGs play a critical role in information security and crytographic applications [37], [38]. The deterministic aperiodicity and sensitive dependence on the initial condition in a chaotic system give rise to apparently random sequence which have been leveraged to build PRNGs [27], [39], [40]. Due to the improved chaotic properties of FPCS, they are promising for building high quality PRNGs. Here, we introduce a new PRNG using the proposed BFPCS and EFPCS. As a representative example, we are showing the results for Tent-flipped-Tent (TT_f) map but the general scheme can be adapted to any FPCS.

The schematic of the proposed PRNG is shown in Fig. 20. We have two parallel chaotic oscillators, one using seed map (SM) and the other one using FPCM. At every iteration, we truncate the 64-bit output to extract the last 8 bits and XOR

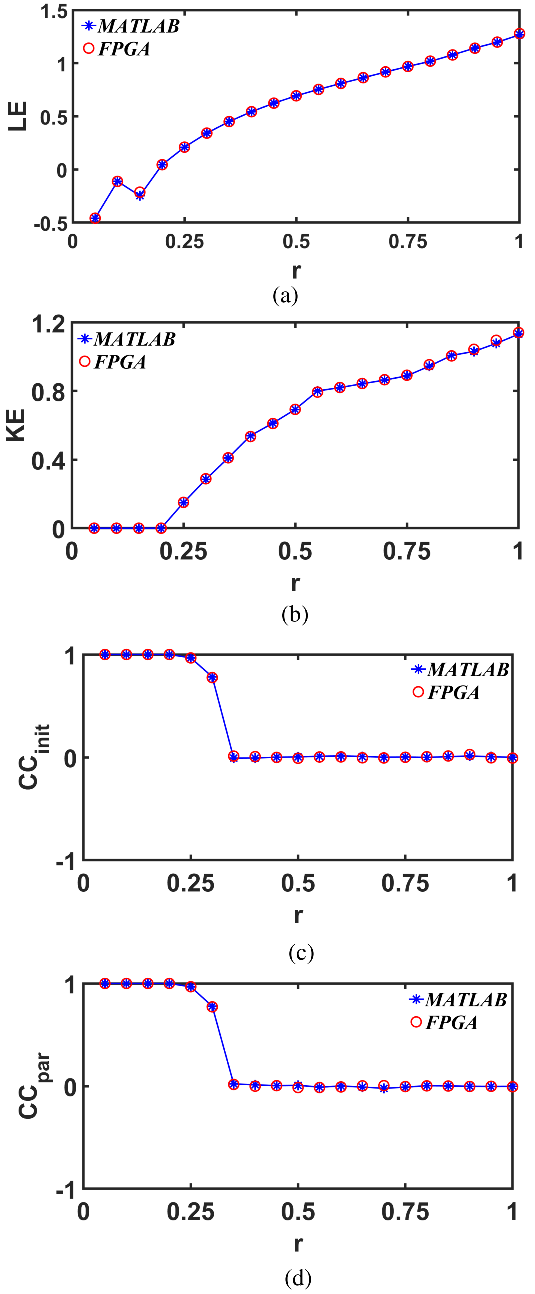


FIGURE 18. LE, KE, CC_{init}, CC_{par} comparison between FPGA and MATLAB Simulation for TT_f of BFPCS.

them to produce the final 8-bit output i.e. a throughput of 8 bits/iteration. The excellent performance of the PRNG has been verified using four statistical randomness tests, namely NIST, FIPS, Diehard and TestU01.

A. NIST SP 800-22 TEST SUITE

The test suite from the National Institute of Standards and Technology (NIST) offers 15 statistical sub-tests to measure the randomness in a sequence [41]. We ran the test with 100 bit-streams generated from 100 different initial condition with each bit-stream having a length of 1 million bits. The significance level was set to 0.01. Hence, a sequence with

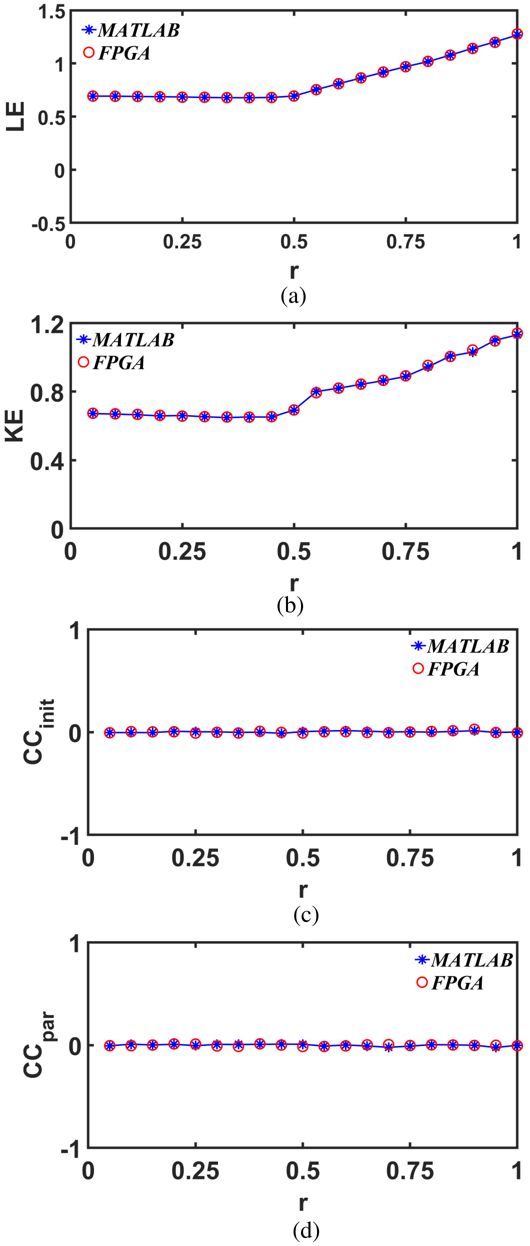


FIGURE 19. LE, KE, CC_{init}, CC_{par} comparison between FPGA and MATLAB Simulation for TT_f of EFPCS.

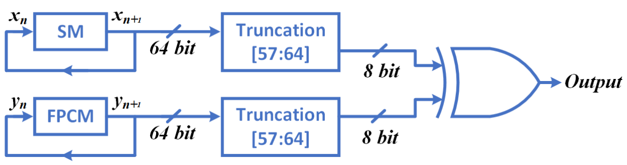


FIGURE 20. Schematic of the proposed PRNG.

100 million bits (containing 100 bit-streams) will pass a particular test if at least 96 out of the 100 bit-streams generate a p-values greater than 0.01. The test suite allocates each of the 100 generated p-values in 10 sub-intervals from 0 to 1 and evaluates the uniformity in the distribution with χ^2-test.

TABLE 4. NIST results (*shows an average of multiple tests).

NIST TEST	Pass rate		$P-value_T$	
	BFPCS	EFPCS	BFPCS	EFPCS
Frequency	0.99	1.00	0.85	0.76
Block frequency	1.00	0.97	0.35	0.38
Cumulative sums*	0.99	1.00	0.81	0.52
Runs	0.99	0.99	0.017	0.51
Longest runs of ones	0.99	0.97	0.98	0.72
Rank	0.97	0.97	0.016	0.031
FFT	1.00	0.98	0.70	0.88
Non-overlapping template*	0.99	0.99	0.51	0.45
Overlapping template	0.99	0.98	0.78	0.28
Universal	0.96	0.99	0.80	0.25
Approximate entropy	0.98	1.00	0.55	0.30
Random excursion*	1.00	0.99	0.18	0.50
Random excursion variant*	1.00	0.99	0.23	0.48
Serial*	1.00	1.00	0.14	0.42
Linear complexity	0.98	1.00	0.76	0.25

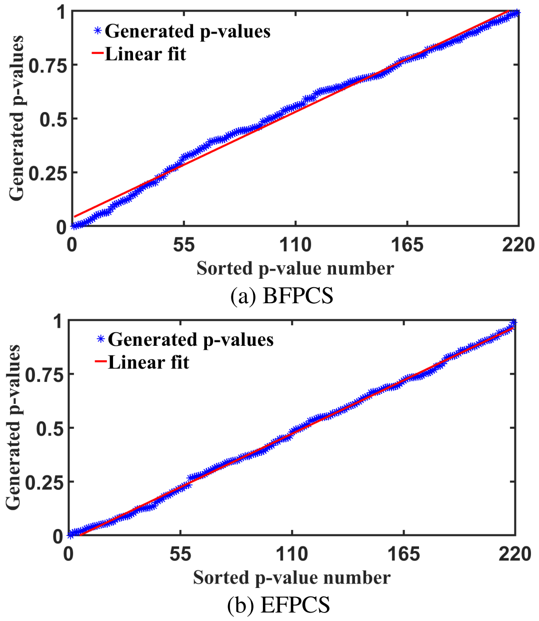
TABLE 5. FIPS test results.

PRNG	Total success	Monobit	Poker	Runs	Long run
BFPCS	**4998**	-	1	1	-
EFPCS	**4996**	1	-	-	3

The sequence under test can be considered uniform if the p-value generated from the χ^2-test (refers to $p-value_T$) is greater than or equal to 0.0001. NIST results, presented in Table 4, show that both BFPCS and EFPCS sequences using TT_f based PRNG pass all requirements of 15 sub-tests.

B. FIPS PUB 140-2

The Federal Information Processing Standards Publications (FIPS PUB) 140-2 test suite was developed by NIST [42]. FIPS tests the randomness of a binary sequence by dividing the sequence into 20,000-bit blocks. Hence, for a test sequence with 100 million bits, there will be 5000 blocks in total. The blocks are subjected to 4 sub-tests namely, Monobit, Poker, Runs, and Long run. The Monobit test counts the number of 1's in each 20,000-bit block. To pass the test, this number must be within the range of [9725, 10275]. The Poker test divides each 20,000-bit block into 5,000 successive 4-bit segments. The 4-bit segment can have 16 possible values. The occurrences of 16 values are counted and stored. This sub-test examines the uniformity of the 4-bit segment. Runs test counts and stores the maximum sequence of consecutive 1's or 0's in a 20,000-bit block. A run of 26 or more of either 1's or 0's is defined as a Long run. The total number of Long runs in a 20,000-bit block is counted as the total failure. TABLE 5 shows the FIPS test result for BFPCS and EFPCS using TT_f based PRNG. The second column (from the left) of TABLE 5 shows the total number of blocks passing the test out of the total 5000 blocks and the last four columns show the number of failed blocks under corresponding sub-tests. The results show close to 100% success implying great randomness.



(a) BFPCS

(b) EFPCS

FIGURE 21. Diehard statistical test result.

TABLE 6. TestU01 results.

PRNG	Bit Length	*Rabbit*	*Alphabit*	*BlockAlphabit*
BFPCS	2^{20}	38/38	17/17	102/102
	2^{24}	39/39	17/17	102/102
	2^{28}	40/40	17/17	102/102
EFPCS	2^{20}	38/38	17/17	101/102
	2^{24}	39/39	17/17	102/102
	2^{28}	40/40	17/17	102/102

C. DIEHARD STATISTICAL TEST SUITE

The Diehard statistical test suite was developed by Marsaglia [43]. It generates 219 p-values under 15 sub-tests. A sequence is considered to be random if the generated p-values range between [0,1). On the other hand, if there are six or more (out of 219) p-values of either 0 or 1 then the sequence fails. Our test sequences contain 100,000,032 bits (with a padding of 32 1's at the beginning). FIGURE 21 shows the plots of p-values, organized in ascending order. The linear fits in both plots show close conformity with the generated p-value trends, demonstrating excellent randomness for both BFPCS and EFPCS using TT_f based PRNG.

D. TestU01

TestU01 offers a collection of utilities for empirical statistical testing. This test suite comes as a software library generated in ANSI C language [44]. We ran three test batteries namely, *Rabbit*, *Alphabit*, and *BlockAlphabit*. The complete test was run on three test sequences containing 2^{20}, 2^{24}, and 2^{28} bits. Depending on this sequence size, the *Rabbit* test consists of 38 sub-tests whereas, *Alphabit* consists of 17 sub-tests and *BlockAlphabit* consists of 6 blocks of the same 17 sub-tests (102 tests in total). The sequence passes a sub-test if the generated p-value remains between 0.001 and 0.999. TABLE 6 presents the ratio between passes and the total number of sub-tests in each case and demonstrates excellent performance for sequences generated from both BFPCS and EFPCS using TT_f based PRNG.

IX. CONCLUSION

In this work, we have introduced FPCS, a new methodology for building high-quality 1-D chaotic map using existing maps, and outlined two schemes called BFPCS and EFPCS. Nine configurations based on three seed maps are shown for each scheme, and the resulting performance improvement of these new maps compared to their constituent maps has been demonstrated using bifurcation diagram, Lyapunov Exponent, Kolmogorov entropy, and correlation coefficient. EFPCS is computationally more expensive compared to BFPCS but yields better chaotic properties with wider chaotic window for all combinations. We compared our results against prior works which show marked improvement in several important metrics. We also presented hardware implementation of both schemes in FPGA to illustrate their simplicity of implementation and verified their entropic properties against software simulation. The improved entropy metrics seem promising for various security applications. We showed one application by building a new PRNG using proposed maps and validated their excellent randomness using four standard statistical tests, namely NIST, FIPS, Diehard, and TestU01. Since a common parameter was used for both constituent maps in this paper, future work may include extension of the proposed framework using two maps with two independent parameters.

REFERENCES

[1] E. N. Lorenz, "Deterministic nonperiodic flow," *J. Atmos. Sci.*, vol. 20, no. 2, pp. 130–141, Mar. 1963.

[2] S. H. Strogatz, *Nonlinear Dynamics and Chaos With Student Solutions Manual: With Applications to Physics, Biology, Chemistry, and Engineering*. Boca Raton, FL, USA: CRC Press, 2018.

[3] X. Wang, X. Wang, J. Zhao, and Z. Zhang, "Chaotic encryption algorithm based on alternant of stream cipher and block cipher," *Nonlinear Dyn.*, vol. 63, no. 4, pp. 587–597, Mar. 2011.

[4] K.-W. Wong, Q. Lin, and J. Chen, "Simultaneous arithmetic coding and encryption using chaotic maps," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 57, no. 2, pp. 146–150, Feb. 2010.

[5] X. Wang, L. Teng, and X. Qin, "A novel colour image encryption algorithm based on chaos," *Signal Process.*, vol. 92, no. 4, pp. 1101–1108, Apr. 2012.

[6] S.-L. Chen, T. Hwang, and W.-W. Lin, "Randomness enhancement using digitalized modified logistic map," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 57, no. 12, pp. 996–1000, Dec. 2010.

[7] Y. Wu, Z. Hua, and Y. Zhou, "*n*-dimensional discrete cat map generation using Laplace expansions," *IEEE Trans. Cybern.*, vol. 46, no. 11, pp. 2622–2633, Nov. 2016.

[8] P. S. Paul, M. Sadia, M. R. Hossain, B. Muldrey, and M. S. Hasan, "Design of a low-overhead random number generator using CMOS-based cascaded chaotic maps," in *Proc. Great Lakes Symp. VLSI*, Jun. 2021, pp. 109–114.

[9] J. F. Lindner, V. Kohar, B. Kia, M. Hippke, J. G. Learned, and W. L. Ditto, "Strange nonchaotic stars," *Phys. Rev. Lett.*, vol. 114, no. 5, Feb. 2015, Art. no. 054101.

[10] M. Ercsey-Ravasz and Z. Toroczkai, "Optimization hardness as transient chaos in an analog approach to constraint satisfaction," *Nature Phys.*, vol. 7, no. 12, p. 966, 2011.

[11] B. Kia, J. F. Lindner, and W. L. Ditto, "A simple nonlinear circuit contains an infinite number of functions," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 63, no. 10, pp. 944–948, Oct. 2016.

[12] V. Kohar, B. Kia, J. F. Lindner, and W. L. Ditto, "Implementing Boolean functions in hybrid digital-analog systems," *Phys. Rev. A, Gen. Phys.*, vol. 7, no. 4, Apr. 2017, Art. no. 044006.

[13] M. B. Majumder, M. S. Hasan, A. Shanta, M. Uddin, and G. Rose, "Design for eliminating operation specific power signatures from digital logic," in *Proc. Great Lakes Symp. VLSI*, May 2019, pp. 111–116.

[14] M. S. Hasan, A. S. Shanta, P. S. Paul, M. Sadia, M. B. Majumder, and G. S. Rose, "Design of an enhanced reconfigurable chaotic oscillator using G^{4}FET-NDR based discrete map," in *Proc. IEEE 14th Dallas Circuits Syst. Conf. (DCAS)*, Nov. 2020, pp. 1–5.

[15] M. Feki, "An adaptive chaos synchronization scheme applied to secure communication," *Chaos, Solitons Fractals*, vol. 18, no. 1, pp. 141–148, 2003.

[16] L. Wang, X. Mao, A. Wang, Y. Wang, Z. Gao, S. Li, and L. Yan, "Scheme of coherent optical chaos communication," *Opt. Lett.*, vol. 45, no. 17, pp. 4762–4765, 2020.

[17] G. S. Rose, "A chaos-based arithmetic logic unit and implications for obfuscation," in *Proc. IEEE Comput. Soc. Annu. Symp. VLSI*, Jul. 2014, pp. 54–58.

[18] A. S. Shanta, M. B. Majumder, M. S. Hasan, and G. S. Rose, "Physically unclonable and reconfigurable computing system (PURCS) for hardware security applications," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 40, no. 3, pp. 405–418, Mar. 2021.

[19] R. M. May, "Simple mathematical models with very complicated dynamics," *Nature*, vol. 261, pp. 459–467, Jun. 1976.

[20] M. Hénon, "A two-dimensional mapping with a strange attractor," in *The Theory of Chaotic Attractors*. New York, NY, USA: Springer, 1976, pp. 94–102.

[21] E. Zeraoulia, *Robust Chaos and Its Applications*, vol. 79. Singapore: World Scientific, 2012.

[22] I. Sushko, L. Gardini, and K. Matsuyama, "Robust chaos in a credit cycle model defined by a one-dimensional piecewise smooth map," *Chaos, Solitons Fractals*, vol. 91, pp. 299–309, Oct. 2016.

[23] Z. Hua and Y. Zhou, "Exponential chaotic model for generating robust chaos," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 51, no. 6, pp. 3713–3724, Jun. 2021.

[24] L. Lin, M. Shen, H. C. So, and C. Chang, "Convergence analysis for initial condition estimation in coupled map lattice systems," *IEEE Trans. Signal Process.*, vol. 60, no. 8, pp. 4426–4432, Aug. 2012.

[25] D. Li, M. Han, and J. Wang, "Chaotic time series prediction based on a novel robust echo state network," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 5, pp. 787–799, May 2012.

[26] Y. Deng, H. Hu, W. Xiong, N. N. Xiong, and L. Liu, "Analysis and design of digital chaotic systems with desirable performance via feedback control," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 45, no. 8, pp. 1187–1200, Aug. 2015.

[27] C.-Y. Li, Y.-H. Chen, T.-Y. Chang, L.-Y. Deng, and K. To, "Period extension and randomness enhancement using high-throughput reseeding-mixing PRNG," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 20, no. 2, pp. 385–389, Feb. 2012.

[28] Z. Hua and Y. Zhou, "Dynamic parameter-control chaotic system," *IEEE Trans. Cybern.*, vol. 46, no. 12, pp. 3330–3341, Dec. 2016.

[29] P. S. Paul, M. Sadia, and M. S. Hasan, "Design of a dynamic parameter-controlled chaotic-PRNG in a 65 nm CMOS process," in *Proc. IEEE 14th Dallas Circuits Syst. Conf. (DCAS)*, Nov. 2020, pp. 1–4.

[30] Y. Zhou, Z. Hua, C.-M. Pun, and C. L. P. Chen, "Cascade chaotic system with applications," *IEEE Trans. Cybern.*, vol. 45, no. 9, pp. 2001–2012, Sep. 2015.

[31] P. S. Paul, M. Sadia, M. R. Hossain, B. Muldrey, and M. S. Hasan, "Cascading CMOS-based chaotic maps for improved performance and its application in efficient RNG design," *IEEE Access*, vol. 10, pp. 33758–33770, 2022.

[32] M. J. Feigenbaum, "Universal behavior in nonlinear systems," *Phys. D, Nonlinear Phenomena*, vol. 7, nos. 1–3, pp. 16–39, May 1983.

[33] P. Grassberger and I. Procaccia, "Estimation of the Kolmogorov entropy from a chaotic signal," *Phys. Rev. A, Gen. Phys.*, vol. 28, no. 4, p. 2591, 1983.

[34] Y. Zhou, L. Bao, and C. P. Chen, "A new 1D chaotic system for image encryption," *Signal Process.*, vol. 97, no. 11, pp. 172–182, Apr. 2014.

[35] M. Sadia, P. S. Paul, M. R. Hossain, B. Muldrey, and M. S. Hasan, "Design and application of a novel 4-transistor chaotic map with robust performance," in *Proc. 28th IEEE Int. Conf. Electron., Circuits, Syst. (ICECS)*, Nov. 2021, pp. 1–5.

[36] D. Zuras, M. Cowlishaw, A. Aiken, M. Applegate, D. Bailey, S. Bass, D. Bhandarkar, M. Bhat, D. Bindel, S. Boldo, and S. Canon, *IEEE Standard for Floating-Point Arithmetic*, IEEE Standard 754–2019, 2008, pp. 1–70.

[37] I. V. Chugunkov, M. A. Ivanov, E. A. Gridneva, and N. Y. Shestakova, "Classification of pseudo-random number generators applied to information security," in *Proc. IEEE Conf. Russian Young Researchers Electr. Electron. Eng. (EIConRus)*, Feb. 2017, pp. 370–373.

[38] Y. Dodis, D. Pointcheval, S. Ruhault, D. Vergniaud, and D. Wichs, "Security analysis of pseudo-random number generators with input: /Dev/random is not robust," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur. (CCS)*, 2013, pp. 647–658.

[39] 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.

[40] R. B. Naik and U. Singh, "A review on applications of chaotic maps in pseudo-random number generators and encryption," *Ann. Data Sci.*, pp. 1–26, Jan. 2022.

[41] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, and E. Barker, "A statistical test suite for random and pseudorandom number generators for cryptographic applications," Booz-allen and hamilton inc mclean va, Tech. Rep., 2001.

[42] *Security Requirements for Cryptographic Modules*, Standard FIPS 140-2, Information Technology Laboratory, National Institute of Standards and Technology, Gaithersburg, MD, USA, 2001.

[43] G. Marsaglia. (2022). *The Marsaglia Random Number CDROM Including the Diehard Battery of Tests of Randomness*. [Online]. Available: https://web.archive.org/web/20160125103112/http://stat.fsu.edu/pub/diehard/

[44] P. L'Ecuyer and R. Simard, "TestU01: AC library for empirical testing of random number generators," *ACM Trans. Math. Softw.*, vol. 33, no. 4, pp. 1–40, 2007.

MD SAKIB HASAN (Member, IEEE) received the B.Sc. degree in electrical and electronic engineering from the Bangladesh University of Engineering and Technology, in 2009, and the Ph.D. degree in electrical engineering from the University of Tennessee, Knoxville, in 2017. He is currently working as an Assistant Professor at the Department of Electrical and Computer Engineering, University of Mississippi. His research interests include security solutions using nonlinear dynamics, neuromorphic computing, semiconductor device modeling, and VLSI design.

PARTHA SARATHI PAUL (Graduate Student Member, IEEE) received the B.Sc. degree in electrical and electronic engineering from the Bangladesh University of Engineering and Technology, in 2014, and the M.Sc. degree in electrical and computer engineering from Oregon State University, in 2017. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, University of Mississippi. His research interests include mixed-signal circuit design for chaos-based hardware security applications such as random number generator and reconfigurable logic.

ANURAG DHUNGEL (Graduate Student Member, IEEE) is currently pursuing the B.Sc. (Eng.) degree with the Department of Electrical and Computer Engineering, University of Mississippi, USA. His research interests include nonlinear dynamics, chaos based hardware security applications, digital design, and deep learning.

MAISHA SADIA (Graduate Student Member, IEEE) received the B.Sc. and M.Sc. degrees in electrical engineering from the University of Mississippi, in 2017 and 2019, respectively, where she is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering. Her research interests include vehicular ad-hoc networks and chaos-based hardware security applications.

MD RAZUAN HOSSAIN (Graduate Student Member, IEEE) received the B.Sc. (Eng.) degree from the Department of Electrical Electronic and Communication Engineering, Military Institute of Science and Technology, Bangladesh, in 2015, and the M.Sc. degree from the Department of Electrical and Computer Engineering, North Dakota State University, in 2019. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, University of Mississippi. His current research interests include neuromorphic computing and nonlinear dynamics.