Proceedings of the Indian National Science Academy (2026) 92:1162–1184
https://doi.org/10.1007/s43538-025-00475-2

RESEARCH PAPER

Experimental realization of all logic elements and memory latch in SC-CNN Chua's circuit

Ashokkumar P.[1] · Sathish Aravindh M.[1,2] · Venkatesan A.[1] · Lakshmanan M.[2]

Received: 14 October 2024 / Accepted: 26 May 2025 / Published online: 25 June 2025
© Indian National Science Academy 2025

Abstract
The Chua's circuit is examined using a State Controlled-Cellular Neural Network (SC-CNN) framework with two logical square wave input signals. We illustrate, in particular, that this nonlinear circuit can generate all the basic logic operations, including OR/NOR, AND/NAND, and XOR/XNOR gates, by making use of the hopping of attractors which this circuit produces in different phase space regimes. Further, it is shown that besides two-inputs, the circuit emulates multi-input logic elements. Moreover, all these logic elements are effectively functioning for a tolerable limit of noise intensity. These observations are experimentally realized. Thus our investigation sheds new light in the field of digital technology where the existing static logic gates may be replaced or complemented by this kind of dynamical nonlinear circuits.

Keywords Chua's circuit · Logic gates · Memory device · Multi-input

Introduction

It is a well-established fact that the impending possibility of approaching the physical restrictions of Moore's law has prompted the development of alternative schemes to do further computations within the limited set of hardware (Mano 2003). Nonlinear dynamics based computing is one such alternate computing scheme to replace or supplement the current silicon chip based computer architecture (Hopfield 1982; Sinha and Ditto 1998, 1999; Prusha and Lindner 1999; Murali et al. 2009, 2009; Sinha et al. 2009; Bulsara et al. 2010; Guerra et al. 2010; Worschech et al. 2010; Zamora Munt and Masoller 2010; Zhang et al. 2010; Singh and Sinha 2011; Dari et al. 2011, 2011; Storni et al. 2012; Roychowdhury 2015; Kohar et al. 2017; Venkatesh et al. 2017a, b; Neves et al. 2017; Kia et al. 2017; Murali et al. 2018; Manaoj Aravind et al. 2018; Sathish Aravindh et al. 2018, 2020). An interesting topic of research in nonlinear dynamics is the use of the richness in nonlinear systems to generate flexible and adaptable logic units. Various methods have been proposed to realize reconfigurable logic gates by utilizing the underlying nonlinear phenomena. These schemes have also been implemented experimentally. Chaotic computing in which chaotic systems are used to demonstrate the implementation of flexible parallel logic gates has been suggested (Sinha et al. 2002). Piecewise-linear circuits are exploited to build all logic gates (Peng et al. 2008, 2010; Campos-Cantón et al. 2010). The analytical geometry of the equation is used to build basic logic elements and SR flip-flop (Cafagna and Grassi 2006; Campos-Cantón et al. 2012). It is emphasized in ref. (Canton et al. 2017) that reconfigurable logic gates and programmable wiring are produced by integrating a programmable matrix using nonlinear systems or circuits. In ref. Campos-Cantón et al. (2012) it is reported that a parameterized method can be employed to construct multivibrators through Chua's circuit. It has recently been demonstrated that if two low-amplitude square waves are

P. Ashokkumar and M. Sathish Aravindh authors have been contributed equally to this work.

✉ Venkatesan A.
av.phys@gmail.com

Ashokkumar P.
ak3phys@gmail.com

Sathish Aravindh M.
sathisharavindhm@gmail.com

Lakshmanan M.
lakshman.cnld@gmail.com

[1] PG and Research Department of Physics, Nehru Memorial College, Autonomous, Affiliated to Bharathidasan University, Puthanampatti, Tiruchirappalli 621 007, India

[2] Department of Nonlinear Dynamics, School of Physics, Bharathidasan University, Tiruchirappalli 620 024, India

used in a bistable system, the response of the system generates logic elements. The system's nonlinearity interacts with noise to provide a flexible and adaptable logical behavior. The phrase suggested for this phenomenon is Logical Stochastic Resonance (LSR) Bulsara et al. (2010). A large number of nonlinear systems, including a nanoscale device (Guerra et al. 2010), resonant tunnel diode (Worschech et al. 2010), vertical cavity surface emitting laser (Zamora Munt and Masoller 2010; Zhang et al. 2010), polarization bistable laser (Singh and Sinha 2011), certain chemical system (Bulsara et al. 2010), synthetic gene networks (Dari et al. 2011), and so on, have been found to exhibit the LSR behavior. By simply changing the value of the logic inputs, two of the present authors and Venkatesh have constructed dynamical logic gates in coupled nonlinear systems (Venkatesh et al. 2017a).

Instead of random noise, the possibility of noise-free LSR has also been examined by driving a bistable system with a periodic forcing (Gupta et al. 2011). Additionally, it has been reported that in noisy two state systems, periodic forcing improves the LSR (Kohar et al. 2014). All the basic logic gates and SR flip-flop can be realized in the Murali-Lakshmanan-Chua's circuit (MLC), by replacing the noise with a high frequency periodic forcing. This term is depicted as Logical Vibrational Resonance (LVR) Venkatesh and Venkatesan (2016); Venkatesh et al. (2017a, b). Both LSR and LVR are used to produce AND /NAND, OR/ NOR and SR-flip-flop in bistable systems. Then naturally the question arises whether by exploiting the LSR/LVR phenomena can one produce other logic gates including XOR/XNOR in bistable systems. Recently, the present authors addressed this issue (Ashokkumar et al. 2021). They pointed out that for LSR/LVR in bistable systems all the logic gates can be obtained depending upon whether the the system's response resided in the left well for input set (0, 0), and the right well for input set (1, 1). For other inputs (0, 1)/(1, 0), the output of the corresponding system may be the left or right well depending on the logic operations used. In bistable nonlinear systems, these assumptions are ideally suited to generate OR/NOR and AND/NAND gates. These assumptions, however, do not lead to the XOR/XNOR logic operations. Only when the inputs are at different logic levels, such as (0, 1) or (1, 0), does an XOR gate have a high logic output '1' and a low logic output '0', for the same logic inputs (0, 0)/(1, 1). The assumptions made in order to create logic gates in a bistable nonlinear system may result in error or information loss. However, the current authors solved these concerns by taking the MLC circuit into consideration (Ashokkumar et al. 2021) and demonstrating that this circuit is suitable for implementing all logic gates, including the XOR/XNOR gates.

As mentioned earlier that a bistable dynamical system produces the logic operation if two low-amplitude square waves are fed into this system in an optimal window of moderate noise. Instead of noise, it is also shown that the logic response is observed for a range of high frequency periodic forcing. In this connection, it is pertinent to ask: Can one observe large amplitude logical response of the system without any external forcing except for input signals and biasing? From a practical point of view, it is difficult to design the system with large number of external forces such as sinusoidal force, input signals biasing and noise. This question is addressed in the present work. We have shown that in a suitable nonlinear dynamical system, an appropriate internal system parameter can generate all logical elements even if two or three low-amplitude logic inputs are fed into the system. In particular, we have shown in the present work that when Chua's circuit is used to mimic logic gate operations, which mainly stem from the idea that even low-amplitude logic input signals can be detected by exploiting the resonance phenomenon inherent in the circuit, this is in fact possible. As a result, when we apply two low-amplitude square logic waves to the Chua's circuit, it is observed experimentally that the circuit produces large amplitude logic gate operations (See Secs.5-7 of the present paper, specifically Figs. 5, 6, 7, 10, 11, 12, 14, 15, 16) for the first time. Thus, our study shows that the Chua's circuit has the capability to produce all logic operations even for small amplitude logic inputs. It is well known that the Chua's circuit is a paradigm of chaotic systems with full controllability and that it can be used to observe a variety of chaotic behavior characteristics, including period doubling cascades, periodic windows, spiral attractors, and double scroll attractors.

Therefore, a State-Controlled Cellular Neural Network (SC-CNN) based Chua's circuit (Arena et al. 1995), is considered in the present work. Three CNN cells are used in its construction, along with biasing and noise. A Chua's diode, a linear resistor, two linear capacitors, and an inductor make up the typical Chua's circuit. The discrete inductor typically limits the circuit used to fabricate integrated circuits. The SC-CNN based Chua's circuit, on the other hand, is completely inductorless, which makes it easier to construct the associated hardware. Additionally, while expanding the study to coupled systems, the CNN is ideally suited. Therefore, in terms of hardware realization, the SC-CNN based Chua's circuit is more advantageous than the existing Chua's circuit. We demonstrate that the SC-CNN based Chua's circuit responds to two aperiodic logical signals by producing all of the logic gates that are accessible in digital electronics. We report that the XOR/XNOR gates, among other logic elements, are produced by the circuit. To be specific, we find that when the input streams are given to cell-2 of the SC-CNN Chua's circuit, the responses of cell-1 and cell-2 of the SC-CNN Chua's circuit produce NOR/NAND/XNOR gates depending upon the biasing value while the cell-3 of the circuit produces its complementary gates OR/AND/XOR.

Thus, two sets of the voltage variables of the Chua's circuit produce two sets of the same gates while the third voltage variable produces the complementary gates. Besides, we also show that when the logic inputs are given to cell-1, and cell-3, the corresponding response of the Chua's circuit produces all the gates. It is further demonstrated that the resulting logic gates function efficiently over an appropriate range of noise intensity. Also, the possibility of occurrence of both high active as well as low active SR flip-flop is reported. Multi-input gates are also realized in the circuit. Finally, we summarize the novelty of the present work as follows.

1. Obtaining all the logic gates OR/NOR, AND/NAND, and XOR/XNOR in a single nonlinear circuit system.
2. Constructing low active and high active SR flip-flop in the response nonlinear circuit.
3. The possibility of occurrence of multi input OR/NOR, AND/NAND gates in the same circuit.
4. Further, the gates generated by the circuit are immune to noise and the logic behaviour can also be obtained thorough LSR.

Also, in the present work, we have considered an analog version of the Chua's circuit. Switching from one segment to another segment takes the order of a few milliseconds (10^{-3}) (Kohar and Sinha 2012). Also, we wish to point out that we include only a low amount of input signal, $I_1 = I_2 = 500mV$. This is sufficient to build all the logic gates in our simple nonlinear Chua's circuit.

Thus the above simple nonlinear circuit has the ability to produce all the logic gates and memory latch and it is further tolerant to noise. Although CMOS logic still outperforms SC-CNN Chua gates in raw speed, SC-CNN offers a significant improvement over earlier analog chaotic designs. These logic gates consume considerably less power due to passive integration and low-energy chaotic transitions. Moreover, due to attractor based logic state separation, SC-CNN gates achieve better noise margins. An additional advantage is the dynamic state reprogramming capability, which traditional CMOS logic lacks. The concept of dynamic reconfigurability in nonlinear systems like the SC-CNN based on Chua's circuit is highly attractive for adaptive logic processing and reprogrammable hardware. Unlike fixed digital logic circuits, dynamically reconfigurable systems allow on-the-fly switching between different logic functions without physically altering the hardware. While the idea is promising, effective utilization demands a clear understanding of: (i) the mechanism enabling reconfigurability, (ii) the parameters under user control, (iii) the speed at which reconfiguration can occur, and (iv) the hardware and computational overhead involved. In SC-CNN-based Chua's circuits, dynamic reconfigurability refers to the ability to alter logic functionality (e.g., AND, OR, XOR, NAND) without changing the physical circuit layout. Instead, reconfiguration is achieved by adjusting internal system parameters or inputs during runtime (Groote et al. 2021). Further, the logic gates produced by the system are dynamic reconfigurable logic gates. Because of the dynamical phenomenon inherent in the circuit the convergence speed of logic gate, logic gate delay, etc. are better than the static logic gates available in the market.

The structure of this paper is as follows. In Sect. "SC-CNN based Chua circuit", we provide an overview of the SC-CNN cell structure of the Chua's circuit. In Sect. "Experimental realization of SC-CNN cell of Chua's circuit", the experimental implementation of Chua's circuit's CNN cell. In Sects. "Experimental results", "Realization of all gates when input signals applied to second cell ($\boldsymbol{x_2}$) of the Chua's circuit" and "Logic response in multi-input configuration", respectively, we also discuss the experimental findings, the implementation of all logic gates and multi-input logic gates. In Sect. "Effect of noise on logic gates", the impact of noise is examined. We conclude by summarizing our findings in Sect. "Conclusion".

SC-CNN based Chua circuit

Inductorless nonlinear Chua circuit systems are variants of Chua's circuit that eliminate the need for inductors, typically replacing them with switched-capacitor or other nonlinear circuit elements. The simple nonlinear Chua's circuit is useful in various chaos-based information engineering applications such as secure communications (Abel and Schwarz 2002; Ma et al. 2010), chaotic signal radar (Liu et al. 2007), chaos-based analog-to-information conversion (Xi et al. 2013), etc. The realization of Chua's circuit with the presence of a manually winding inductor with parasitic equivalent series resistor makes the Chua's circuit hardware bulky, unsuitable for IC design and less robust (Banerjee 2012). Numerous works have been reported on different realization schemes of the Chua's circuit (Bao et al. 2016; Li et al. 2018; Radwan et al. 2003; Kiliç 2004; Arena et al. 1995, 2005). All of these realizations are mainly focused on inductor-free realization of Chua's circuit, such as current feedback op amps (CFOAs) Kiliç (2004), general impedance converter (GIC) Kennedy (1992), complementary metal-oxide-semiconductor (CMOS) Radwan et al. (2003) and state controlled-cellular neural network (SC-CNN) Arena et al. (2005); Swathy and Thamilmaran (2013); Günay et al. (2007); Ashokkumar et al. (2021).

The Cellular Neural Network (CNN) was first developed by Chua and Yang in 1988 Chua and Yang (1988). The advantage of CNN is an n-dimensional array of resistors, capacitors, OP-AMPs, and other analog circuit components,

but without any inductors. This CNN is built from a huge number of interconnected dynamical systems.

CNN is a reasonably basic circuit that may be easily implemented experimentally using appropriate electronic circuit elements. These circuits are powerful tools for emulation of complex dynamics in nonlinear systems. Arena proposed a CNN paradigm generalization (Arena et al. 2005). In this case, the local output and voltage variables of the CNN cells are exchanged with one another. This generalization uses the analog components of CNN and it is known as the State Controlled - CNN (SC-CNN). Many chaotic circuits designed and implemented in terms of SC-CNN have been documented in the literature (Arena et al. 1995; Günay 2010; Günay et al. 2007; Swathy and Thamilmaran 2013; Ashokkumar et al. 2021). This simplicity leads to several technical advantages, particularly for logic circuit design and unconventional computing. Due to the small number of passive components and the absence of high-frequency switching, the energy required per operation is significantly reduced. Logic transitions occur passively as the system evolves toward a new attractor basin, making the process inherently energy-efficient. Digital control signals govern the trajectory of each SC-CNN cell, enabling dynamic reconfiguration of logic functions without hardware changes unlike traditional logic gates that require dedicated transistor networks for each gate type. SC-CNN logic is also inherently robust to small perturbations; a noisy environment is unlikely to cause incorrect logic state detection unless noise levels are sufficient to push the system across attractor basin boundaries, representing a significant improvement over voltage-threshold-based digital logic. Finally, the SC-CNN based Chua structure supports spatial scalability and parallel processing, as each cell operates autonomously but can be coupled with neighboring cells for more complex operations, making it highly suitable for high-density logic arrays and image processing hardware. The simplicity of the SC-CNN-based Chua's circuit lies not only in its hardware design but also in its methodological economy, minimal active control, direct utilization of natural dynamics, and attractor-based logic state separation. These features create a low power, noise-tolerant, and reconfigurable logic framework that compares favorably with both traditional CMOS and previous chaotic logic architectures.

The significant advantages of these CNN circuits are four fold, that is (1) they have no inductors, (2) their only circuitry is RC based, (3) they are parallelly connected and (4) they consume less power. Thus the SC-CNN circuits are realized with less number of hardware and are easily implemented in VLSI design. Motivated by these facts, we consider the SC-CNN framework of Chua's circuit in the present study. Two aperiodic logic signals are fed into the circuit, and as a consequence, all of the logic operations are observable in the circuit's response.

It is a well established fact that the normalized Chua's circuit equations are represented by the following system of coupled first order ordinary differential equations (ODEs) Lakshmanan and Rajasekar (2003):

$$\begin{aligned} \dot{x} &= \alpha(y - x - h(x)) \\ \dot{y} &= x - y + z \\ \dot{z} &= -\beta y \end{aligned} \tag{1}$$

with

$$h(x) = bx + 0.5(a-b).(|x+1| - |x-1|) \tag{2}$$

The SC-CNN variation of the Chua's circuit differs from the original one in that it uses just a small number of OP-AMPs in addition to RC-based circuits, as opposed to the standard Chua's circuit's linear inductor, Chua's diode, and linear RC components (see Fig. 1). The Chua's circuit's inductorless SC-CNN architecture makes it easier to implement as hardware rather than the conventional one (Fortuna et al. 2017). Thus, the SC-CNN Chua's circuit is considered in the present work and this circuit is used to study the impact of logic signals on it (see Fig. 2).

The dynamic model of the circuit (Fig. 2) of three fully connected generalized CNN cells are in accordance with the following coupled state equations,

$$\begin{aligned} \dot{x}_1 &= -x_1 + a_1 y_1 + a_{12} y_2 + a_{13} y_3 + \sum_{k=1}^{3} s_{1k} x_k + i_1, \\ \dot{x}_2 &= -x_2 + a_{21} y_1 + a_2 y_2 + a_{23} y_3 + \sum_{k=1}^{3} s_{2k} x_k + i_2, \\ \dot{x}_3 &= -x_3 + a_{31} y_1 + a_{32} y_2 + a_3 y_3 + \sum_{k=1}^{3} s_{3k} x_k + i_3, \end{aligned} \tag{3}$$

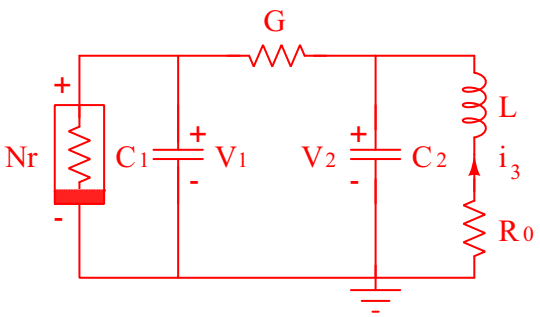


Fig. 1 Generalized experimental realization of Chua's circuit (Lakshmanan and Rajasekar 2003)

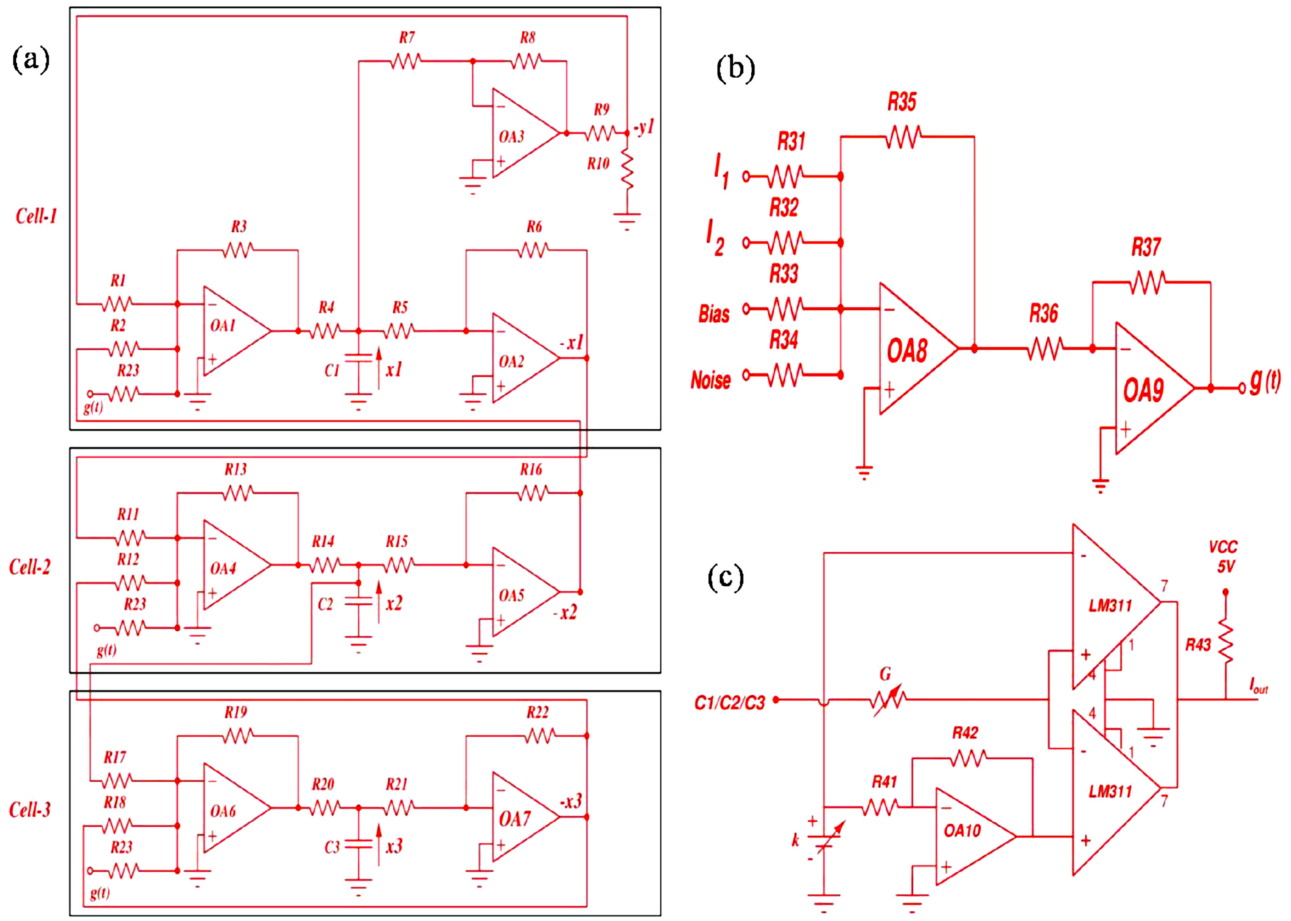


Fig. 2 **a** Experimental verification of Chua's SC-CNN-based circuit system using few OP-AMPs and resistors with three capacitors as suggested by Arena et al. Arena et al. (1995). **b** Combination of external logic inputs, constant bias and Gaussian white noise. **c** Detector/comparator circuit for implementing the traditional/static OR/AND/NOR/NAND/XOR/XNOR logic gates (Campos-Cantón et al. 2010)

where the voltage variables are x_1, x_2, and x_3 and the corresponding outputs are y_1, y_2, and y_3, respectively. With the choice:

$$a_{12} = a_{13} = a_2 = a_{23} = a_{32} = a_3 = a_{21} = a_{31} = 0;$$
$$s_{13} = s_{31} = s_{22} = 0;\ i_1 = i_2 = i_3 = g(t)$$

the above set of equations becomes

$$\begin{aligned} \dot{x}_1 &= -x_1 + a_1 y_1 + s_{11}x_1 + s_{12}x_2 + g(t), \\ \dot{x}_2 &= -x_2 + s_{21}x_1 + s_{23}x_3 + g(t), \\ \dot{x}_3 &= -x_3 + s_{32}x_2 + s_{33}x_3 + g(t). \end{aligned} \tag{4}$$

It is clear from the above equation that the unfolding of Chua's circuit equation is a special case of the above equation (4). In fact, assuming:

$$a_1 = \alpha(m_1 - m_0);\ s_{11} = 1 - \alpha \cdot m_1;\ s_{12} = \alpha;$$
$$s_{21} = s_{23} = 1;\ s_{32} = \beta;\ s_{33} = 1 - \gamma,$$

one can deduce the original set of Chua's circuit equations (1). In the unfolded Chua's circuit, the voltage variables x_1, x_2 and x_3 are equal to x, y and z of Eq.(1) and α, β, γ, m_0 and m_1 are the circuit parameter values. Also $g(t) = I(t) + E + D\eta(t)$, where I, E, and $\eta(t)$ are the logical input values, bias value, and Gaussian white noise with intensity 'D', respectively. Eq.(4) has been investigated thoroughly in numerical and experimental studies. It is reported that Eq.(4) exhibits quasiperiodic route to chaos, intermittency, period-doubling route to chaos, etc. Lakshmanan and Murali (1996). In this paper, we consider the underlying system as shown in Fig. 2 and study the effect of two-logic input signals and then more number of signals too.

Experimental realization of SC-CNN cell of Chua's circuit

Now, using the analog electrical circuit representation of the nonlinear system provided by Eq.(4), we evaluate and confirm its robustness to experiment. Fig. 2 displays the whole circuit realization for the three SC-CNN cells that provide the dynamics of the SC-CNN Chua's circuit as proposed by Arena et al. (1995) and how it has been modified for the current situation. It has three cells, each of which corresponds to the evolution of a different dynamical variable in Eq.(4). The three voltage variables x_1, x_2, and x_3 of Eq.(4) are connected, respectively, to the voltages v_1, v_2, and v_3 across the three capacitors C_1, C_2, and C_3. Simply taking into consideration the saturation of OP-AMPs results in the nonlinearity. The following specific cell components are used to produce circuit realization of Chua's SC-CNN-based circuit (Arena et al. 1995):

Cell 1: R1 = 13.2 KΩ, R2 = 5.7K Ω, R3 = 20 KΩ, R4 = 390 KΩ, R5 = 100K Ω, R6 = 100 KΩ, R7 = 74.8 KΩ, R8 = 970 KΩ, R9 = 27 KΩ, R10 = 2.22 KΩ, C1 = 51 nF;

Cell 2: R11 = R12 = R13 = R15 = R16 = 100 KΩ; R14 = 1 KΩ, C2 = 51 nF;

Cell 3: R17 = 7.8 KΩ, R18 = R19 = R21 = R22 = R23 = 100 KΩ; R20 = 1 KΩ, C3 =51 nF;

R31 − R37 = 10 KΩ, R41 = R42 = 10 KΩ, R43 = 1 KΩ, G = 1 KΩ (variable resistor) and 2 comparators LM311.

In all the above three cells, we choose and active element *IC*741 type voltage OP-AMPs with $\pm 12V$ supply voltages. Using *Agilent* function generator *(33220A)*, *Agilent digital storage oscilloscope* (*DSO*7014*B*) and *Tektronix mixed domain oscilloscope* (*MDO*3024), we obtained all our experimental results (Table 1).

After obtaining the experimental results, they are fed into the comparator (LM311) to compare with the logic output signal so as to yield the traditional/static logic outputs. This comparator behaves as follows: if the voltage at the non-inverting input is more positive than the voltage at the inverting input, the output corresponds to logic gate ON (1). If the voltage at the non-inverting input is less positive than the voltage at the inverting input, the output corresponds to the state OFF (0). This comparator realizes the static logic behavior with the help of high and low-amplitude system output signals (Campos-Cantón et al. 2010).

Table 1 Truth table of logic gates

Logic gates	(0, 0)	(0, 1)/(1, 0)	(1, 1)
OR	0	1	1
AND	0	0	1
XOR	0	1	0
NOR	1	0	0
NAND	1	1	0
XNOR	1	0	1

Experimental results

Now, we examine the impact of the logic inputs on the outputs produced by Chua's SC-CNN based circuit. How the corresponding outputs evolve, and what are the nature of different logic behaviors, are discussed in this section.

Three level logic input

Next, the system response (4) to a certain deterministic logic input signal I is then examined. We specifically use a low/moderate amplitude signal $I = I_1 + I_2$ with two square waves of strengths I_1 and I_2, encoding two logic inputs, to drive the system (4). The inputs have values either 0 or 1, resulting in four different sets of logic inputs (I_1, I_2) : $(0, 0), (0, 1), (1, 0)$, and (1, 1) .

We fix $I_1 = \pm 500mV$, $I_2 = \pm 500mV$, and $E = 0.3V$ in the experimental setup, which correspond to the dimensionless units of the circuit parameters we previously mentioned. By measuring the voltages v_1, v_2, and v_3 across the capacitors C_1, C_2, and C_3, respectively, it is possible to determine how the circuit dynamics changes as a result of the input streams. The output voltages v_1, v_2 and v_3 have an analog value.

Traditionally, the logical response is realized when the response of the system switches from one potential well to another potential well depending on the input logical streams. Switching from one well to another well causes the system to take some time to settle into a steady state. That is why the input signals are held constant for a long time (usually 10 to 10^3 seconds) Kohar and Sinha (2012). However in the present work, we have considered an analog version of the Chua's circuit. Switching from one segment to another segment of the piecewise linear components and settling into a particular segment takes the order of a few milliseconds (10^{-3}s). Therefore, the circuit that we considered in the present study is more appropriate for designing dynamical, reliable, and reconfigurable logic elements (Murali et al. 2018).

The logic input signals I_1 and I_2 are $+500mV$ when the logic input is '1' and $-500mV$ for '0'. For example, both the inputs are $I_1 = I_2 = -500mV$ for logical input '0' and they take the values $I_1 = I_2 = +500mV$ when it is '1'. The input signals I_1, I_2 and the corresponding resultant output of the three level square wave I are depicted in Fig.3(a), (b) and (c), respectively. Here, $I = I_1 + I_2$, takes the amplitude value $-1V(-500mV - 500mV)$ of the wave for the input set

Table 2 Definition of all the logic gates

Logic gates	Segment $v_1 < -ve$ (0, 0)	Segment $-ve < v_1 < +ve$ (0, 1)/(1, 0)	Segment $v_1 > +ve$ (1, 1)
OR	OFF	ON	ON
NOR	ON	OFF	OFF
AND	OFF	OFF	ON
NAND	ON	ON	OFF
XOR	OFF	ON	OFF
XNOR	ON	OFF	ON

(0, 0) , $0V(-500mV + 500mV)$ or $(+500mV - 500mV)$ for the input sets (0, 1) or (1, 0) and $+1V(+500mV + 500mV)$ for the input set (1, 1) .

The fundamental logical operations in digital circuits are OR/NOR and AND/NAND. These gates must be realized with at least two inputs in order to be converted into a single output. To get the high level logic output, for instance, one of the two inputs in an OR logic operation must at least be in a high level logic state. Similarly for the AND logic operation, it is essential that both of the logic inputs to be in the high level to obtain a high level logic output [see Tables 1 and 2].

The XOR logic differs significantly from the previously discussed AND/NAND and OR/NOR logic operations. Taking the XOR logic gate as an example, we note here that it accepts high/maximum level logic output only if the inputs are in separate logic levels, and low level logic output if the inputs are in the same logic level. Any bistable nonlinear system fails to realize this type of gate. Since we are concerned with the Chua's circuit, we have to make note of the fact that it has a three segment piecewise linear and continuous characteristic function. As a result, the present system can generate XOR and XNOR gates in addition to the fundamental logic gates OR, AND, NOR, and NAND. To construct XOR, we set the output to logical '1' if the voltage variable v_1 is between $-2.8V \leq v_1 \leq +2.8V$, and the output to logic '0' if the voltage variable v_1 is either totally negative or fully positive outside the aforementioned range.

It is a known fact that any given nonlinear system is characterized by attractors which are exhibited in the associated phase space. These attractors are generally hopping in different regimes of the phase space, depending on the input. Now let us discuss how the hopping of these attractors in different quadrants of the phase space produces logic elements. For example, the logic AND operation, is realized if and only if the voltage variable v_1 of the system resides in the positive region ($v_1 > +2.8V$) of the phase space for the logic input (1, 1) state and in the negative region ($v_1 < -2.8V$) for all the other logic input states (0, 0),(0, 1)/(1, 0). The detector/comparator circuit used to convert the nonlinear logic output to the traditional/static logic output. The experimental realization of such a strategy, extended to the realization of all the logic gates is given in Table 2. The implementation details are provided in the following sections.

Realization of all gates when input signals applied to second cell (x_2) of the Chua's circuit

When we apply the inputs to each of the different cells of the SC-CNN Chua's circuit, we obtain different responses in each cell. The details of the different responses/gates in each of the cells of the circuit are summarized in Table 3. To be specific, we feed the inputs, for example, to the cell-2 of the circuit and investigate the corresponding responses v_1, v_2 and v_3 of all the cells of the circuit. The circuit (Fig. 2(b)) consists of the logic inputs I_1 and I_2 and bias E, which are applied to the cell-2 of the circuit (Fig. 2(a)). Also, Fig. 2(c) represents the detector/comparator circuit, which is used to convert the nonlinear logic output to the traditional/static logic output that is connected to the output of cell-1($v_1(t)$)/

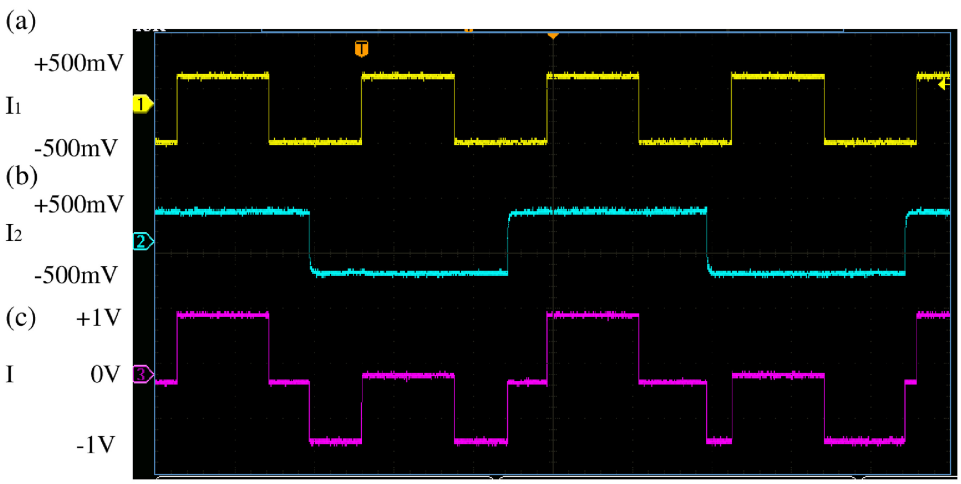


Fig. 3 Panels: **a** and **b** correspond to the two different waveforms of logic inputs I_1 and I_2, respectively. Panel (**c**) shows the corresponding output I, which is the combination of the two logic input signals, that is $I = I_1 + I_2$. Here, $I = I_1 + I_2$, takes the amplitude value $-1V(-500mV - 500mV)$ of the wave for the input set (0, 0) , $0V(-500mV + 500mV)$ or $(+500mV - 500mV)$ for the input sets (0, 1) or (1, 0) and $+1V(+500mV + 500mV)$ for the input set (1, 1)

Table 3 Different logic behaviors on application of logic inputs to different cells

Input $I_1 + I_2$ are given in	Output in the	With +ve bias	With -ve bias	With no bias	Parameter R	Result
Cell-1 only	Cell-1	NOR	NAND	XNOR		
	Cell-2	–	–	–	$0.260K\Omega < R_4 < 0.390K\Omega$	The response of the first cell exhibits NOR/NAND/XNOR dependending upon the bias while the third cell exhibits the complement of the first cell, namely OR/AND/XOR
	Cell-3	OR	AND	XOR		
	Cell-1	NOR	NAND	XNOR		
Cell-2 only	Cell-2	NOR	NAND	XNOR	$0.690K\Omega < R_{14} < 1.0K\Omega$	The response of the first cell & second cell corresponds to NOR/NAND/XNOR dependending upon the bias while the third cell exhibits the complement of the first cell, namely OR/AND/XOR
	Cell-3	OR	AND	XOR		
	Cell-1	NOR	NAND	XNOR		
Cell-3 only	Cell-2	NOR	NAND	XNOR	$0.575K\Omega < R_{20} < 0.985K\Omega$	NOR/NAND/XNOR gates are simultaneously obtained in all the three cells dependending on the biasing values
	Cell-3	NOR	NAND	XNOR		

cell-2($v_2(t)$)/cell-3($v_3(t)$) of the circuit (Fig. 2(a)) Campos-Cantón et al. (2010). Here, tuning the logic gate controller (G), we observed traditional/static logic responses with appropriate values. Initially, we obtained the logic outputs from the SC-CNN-based Chua circuit's terminals C_1, C_2 and C_3 respectively. After we obtained the output signal, it is fed into the detector circuit through the potentiometer (variable resistor) G. Here the potentimeter acts as a logic gate controller (sensitiveness controller) (G) . The voltage $k = 1\ V$ is a transition voltage for low and high-state logic outputs. One can realize the desired logic gates by gradually tuning the value of the logic gate controller G. The response of

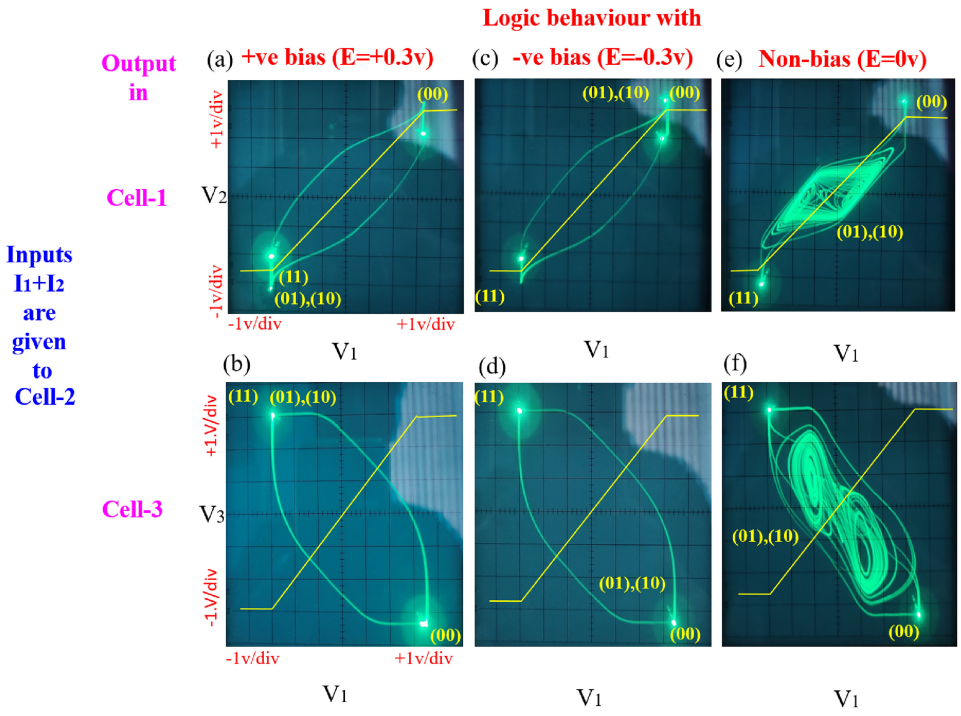


Fig. 4 Different phase space trajectories of the cells are displayed in panels (**a**)–(**f**). Panels (**a**) and (**b**) show the effect of positive bias, $E = +0.3V$; Panels (**c**) and (**d**) display the effect of negative bias, $E = -0.3V$; and Panels (**e**) and (**f**) display the effect without bias, $E = 0.0V$

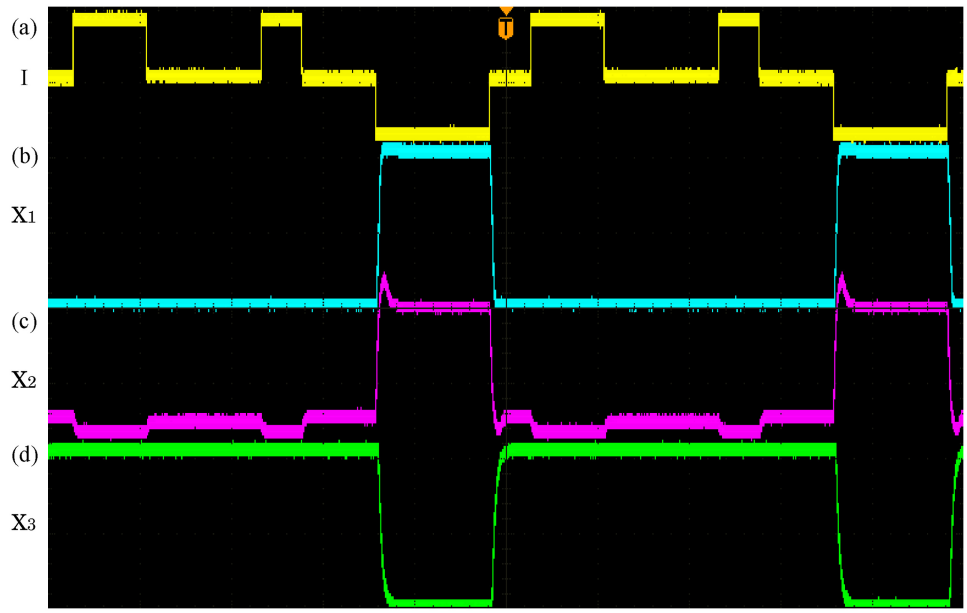


Fig. 5 Experimental observation of the NOR/OR logic gates with bias $E = +0.3V$ in Chua's circuit. A summation of two logic input signals, $I = I_1 + I_2$, is displayed in panel (**a**). The related dynamical reactions of the NOR logic behavior of the systems $v_1(t)$ and $v_2(t)$ are shown in panels (**b**) and (**c**), respectively, while panel (**d**) shows the dynamical responses of the OR logic behavior of the system $v_3(t)$

the circuit is measured by the voltages v_1, v_2 and v_3 across the capacitors C_1, C_2 and C_3, corresponding to the cell-1, cell-2 and cell-3 of the circuit, respectively. In particular, the trajectory plots in the $(v_1 - v_2)$ and $(v_1 - v_3)$ planes and the wave forms v_1, v_2 and v_3 are shown in Figs. 4, 5, 7, and 9, respectively, and we will investigate the nature of the responses of the circuit under different input streams.

Since the Chua's circuit consists of a three segment piece-wise linear element (resistor), the response of the circuit hops between the three segments in the $(v_1 - v_2)$ and $(v_1 - v_3)$ planes. That is, the response of the circuit is leaping between the different regions of the phase trajectory planes depending upon the strength of the input streams [see Fig. 4].

Realization of all logic gates

Now, we fix $I_1 = \pm 500mV$, $I_2 = \pm 500mV$, and the bias $E = +0.3V$. The logic input signals and the bias values, in the form $g(t) = I_1 + I_2 + E$, are fed into the cell-2 and we investigate the response of the circuit. In order to study the system's response, the logical operations of the required type may be retrieved by analyzing the trajectories of the circuit's voltage variables. Consider the voltage variables v_1, v_2, and v_3 of the circuit to design or extract an appropriate gate. For example, in the present SC-CNN based Chua's circuit, the circuit oscillates among the three segments, namely (i) the voltage variable v_1 resides essentially in the negative voltage range $(v_1 < -2.8V)$, (ii) the voltage variable v_1 oscillates between $-2.8V(14div) < v_1 < +2.8V(14div)$ and (iii) the voltage variable v_1 of the circuit exists for $v_1 > +2.8V$(+ve range).

(i) Realization of NOR/OR and NAND/AND logic gates

To obtain the NOR logic operation, it is assumed that (i) the voltage variable resides in the negative region when the input stream (I_1, I_2) is at (−500 mV, −500 mV) /(0, 0), and (ii) the voltage variable resides in the positive regions, when the input signals (I_1, I_2) are at (−500 mV, +500 mV)/(0, 1) or (+500 mV, −500 mV)/(1, 0) and (+500 mV, +500 mV)/(1, 1) states with the bias voltage E = +0.3V. Now, we analyze the trajectory of the circuit in the (v_1, v_2) plane. The three-level diagram for the logic input $I = I_1 + I_2$ is shown in Fig. 5(a). This is experimentally shown in Figs. 4(a) and 5(b-c) that the response of the system v_1 hops in the negative region for input signals I_1 and I_2 that correspond to either the (1, 1) or (0, 1)/(1, 0) states, and in the positive region of the $(v_1 - v_2)$ plane for the (0, 0) input state.

This verifies that the output of the circuit in cell 1 behave like a logical NOR operation. Inspection of Fig. 4(a) also shows that the two voltage variable $v_1(t)$ and $v_2(t)$ produce the logic NOR operation individually. It is also interesting to observe from Fig. 4(b) that when $v_1(t) > 0$, the $v_3(t)$ variable of the circuit is bounded in the regime $v_3(t) < 0$. In other words, the output of the v_3 variable is inverted from of v_1 variable and vice versa. Thus the two variables $v_1(t)$ and $v_3(t)$ build complementary logic gates that carry out NOR and OR operations at the same time in this circuit. These results are clearly seen in the time trajectory and wave form of the voltages $v_1(t)$, $v_2(t)$, and $v_3(t)$ in Figs. 5(b-d).

In Fig. 5(d), it is very clear that the waveform of the v_3 variable abides the inverted output of $v_1(t)/v_2(t)$ variables,

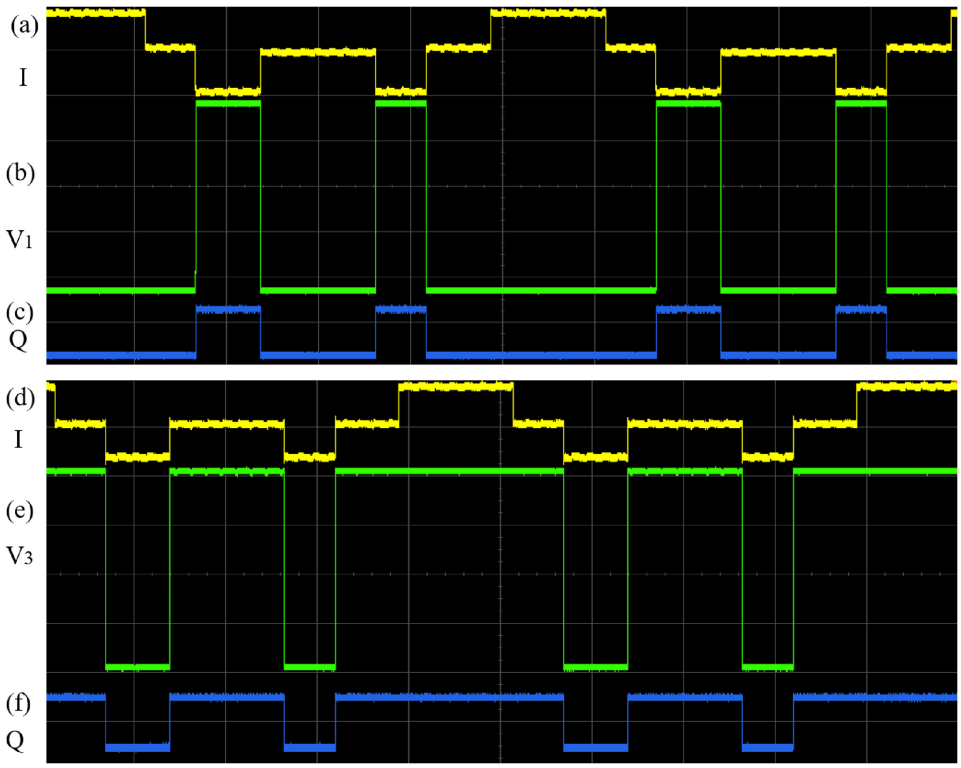


Fig. 6 Experimental observation of the NOR/OR logic gates with bias $E = +0.3V$ in Chua's circuit. A summation of two logic input signals, $I = I_1 + I_2$, is displayed in panel (**a**) and (**d**). The related dynamical reactions of the NOR and OR logic behaviors of the variables $v_1(t)$ and $v_3(t)$ are shown in panels (**b**) and (**e**), respectively, while panel (**c**) and (**f**) show the traditional/static related dynamical responses Q of the NOR/OR logic behavior

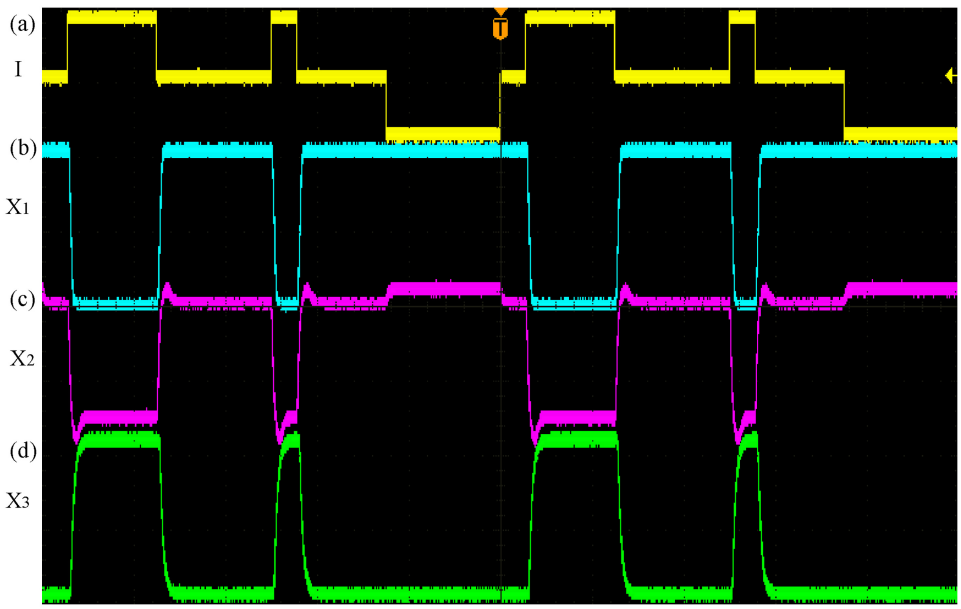


Fig. 7 Experimental observation of the NAND/AND logic gates with bias $E = -0.3V$ in Chua's circuit. A summation of two logic input signals, $I = I_1 + I_2$, is displayed in panel (**a**). The related dynamical reactions of the NAND logic behavior of the systems $v_1(t)$ and $v_2(t)$ are shown in panels (**b**) and (**c**), respectively, while panel (**d**) shows the dynamical responses of the AND logic behavior of the system $v_3(t)$

thereby produces the logic OR operation for the bias value $E = +0.3V$. It is also noted that when the bias changes from $E = +0.3V$ to $E = -0.3V$ [Fig. 4(c-d)], the output of the circuit variables $v_1(t)/v_2(t)$ of the system morphs from NOR logic behaviors to NAND logic (see Fig. 7). This circuit produces logic NOR/NAND through two of its voltage variables $v_1(t)$ and $v_2(t)$ while the other voltage variable, $v_3(t)$, realizes their complimentary logic operations OR/AND. For obtaining the traditional/static logic outputs from the existing nonlinear circuit logic outputs, one may proceed as follows. To implement the traditional/static NOR/OR and NAND/AND logic gates, we additionally added a detector circuit [see Fig. 2(c)]. Then, Figs. 6(c)/(f) and 8(c)/(f) represent the traditional/static logical outputs 'Q' for NOR/ OR and NAND/ AND logic gates from tuning the logic gate controller (G), G ∈ 0.440 KΩ - 0.830 KΩ (see Fig. 2(c)).

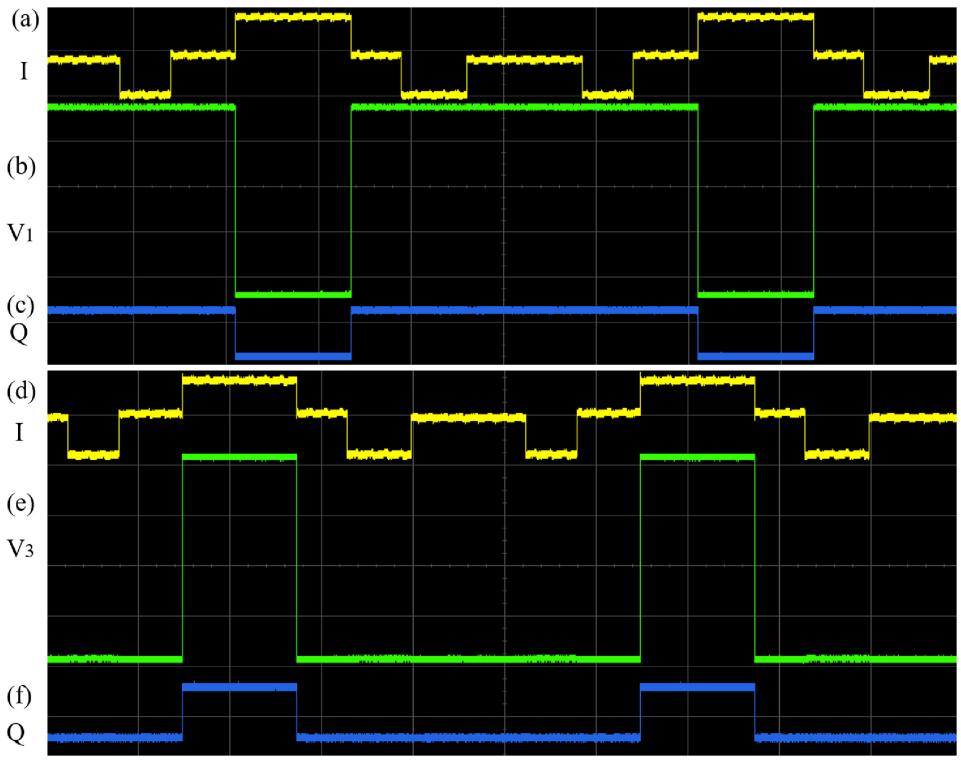


Fig. 8 Experimental observation of the NAND/AND logic gates with bias $E = -0.3V$ in Chua's circuit. A summation of two logic input signals, $I = I_1 + I_2$, is displayed in panel (**a**) and (**d**). The related dynamical reactions of the NAND and AND logic behavior of the variables $v_1(t)$ and $v_3(t)$ are shown in panels (**b**) and (**e**), respectively, while panel (**c**) and (**f**) show the traditional/static related dynamical responses Q of the NAND/AND logic behavior

Figures 6(a) & (d) and 8(a) & (d) are the three level logic inputs, and Figs. 6(b) & (e) and 8(b) & (e) are the system logic outputs (Campos-Cantón et al. 2010; Ashokkumar et al. 2021).

(ii) Realization of XOR/XNOR logic elements

In comparison to the gates AND, OR, NAND, and NOR that were previously discussed, the logic gate XOR is very different. If the input levels are different, either 0 and 1 or 1 and 0, the XOR gate admits a logic output of '1' or an ON state level. In contrast, if the inputs are at the same logic level, either (0, 0) or (1, 1), the output will be a '0' or OFF state level. The opposite condition holds for the XNOR gate. Thus, to implement XNOR, all we need is to set the output to be logic '0' if the voltage variable v_1 is in the range $-2.8V/(10div) < v_1 < +2.8V/(10div)$ while the

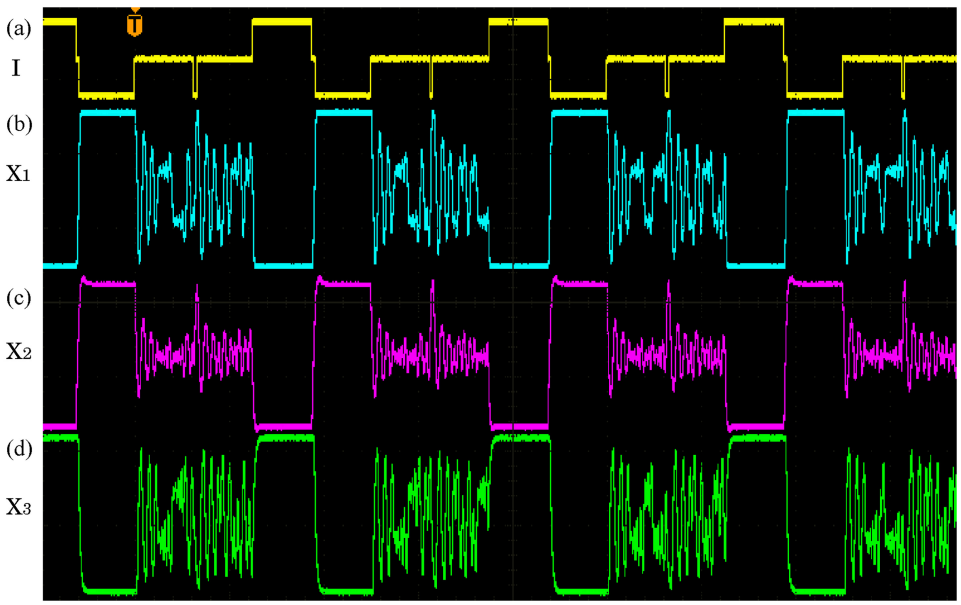


Fig. 9 Experimental observation of the XNOR/XOR logic gates with absence of bias $E = 0.0V$ in Chua's circuit. A summation of two logic input signals, $I = I_1 + I_2$, is displayed in panel (**a**). The related dynamical reactions of the XNOR logic behavior of the systems $v_1(t)$ and $v_2(t)$ are shown in panels (**b**) and (**c**), respectively, while panel (**d**) shows the dynamical responses of the XOR logic behavior of the system $v_3(t)$

output is considered to be logic '1' if the voltage variable v_1 dwells anywhere else.

Experimental observation of the NAND/AND logic gates with bias $E = -0.3V$ in Chua's circuit. A summation of two logic input signals, $I = I_1 + I_2$, is displayed in panel (a) & (d). The related dynamical reactions of the NAND and AND logic behavior of the systems $v_1(t)$ and $v_3(t)$ are shown in panels (b) and (e), respectively, while panel (c) & (d) shows the traditional/static related dynamical responses Q of the NAND/AND logic behavior.

This is realized if we construct the circuit by fixing $I_1 = I_2 = \pm 500mV$, $E = 0.0V$. It is clearly evident in Fig. 9 that the trajectory of the $(v_1 - v_2)$ plot for (0, 0) state of input, the output of the circuit voltage variable v_1 resides in the $v_1 < -2.8V$ region and for input (0, 1)/(1, 0), the output of the circuit voltage variable v_1 wanders within a trapping region between $-2.8V < v_1 < +2.8V$, and for (1, 1) state it is in the $v_1 > +2.8V$ region. As a result, the circuit permits XNOR logic operation for the aforementioned set of parameter values. This point is amply confirmed in the $(v_1 - v_2)$ plane in Figs. 9(b-c) and 4(e). The output of the two voltage variables, v_1 and v_2, is shown to be of the inverted form of the circuit variable v_3, on the other hand. As a result, the complementary logic gate XOR is produced by the voltage variable v_3 (Figs. 9(d) and 4(f)). Using the detector circuit (see Fig.2(c)), we implemented

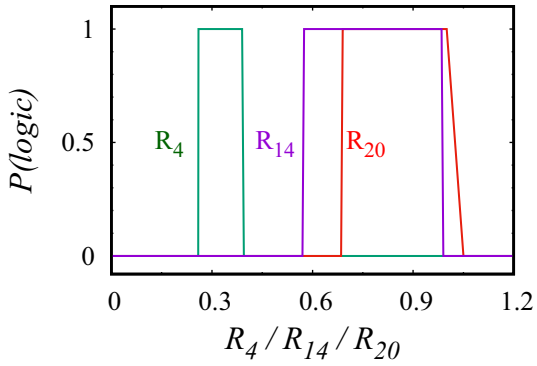


Fig. 11 The probability distribution of attaining a logical response for various R_4 and R_{14} and R_{20} values from cells 1 and 2 and 3 is shown in green, purple, and red, respectively

the traditional/static XOR and XNOR logic gates by tuning the logic gate controller (G), G ∈ 0.365 KΩ - 0.655 KΩ (see Fig.2c). Figure10(c) and (f) correspond to the traditional/static logical outputs 'Q'. Figure 10(a) and (d) are the three level logic inputs, and Fig. 10(b) and (e) are the system logic outputs (Campos-Cantón et al. 2010; Ashokkumar et al. 2021).

In particular, we ascertain the robustness of the logic response with respect to different resistance (R_4, R_{14}, R_{20}) values. For this propose, we calculate the probability P(logic) of getting logic gates for different resistance levels. Essentially P(logic) denotes the ratio of the total number of successful runs to the total number of runs. If the system

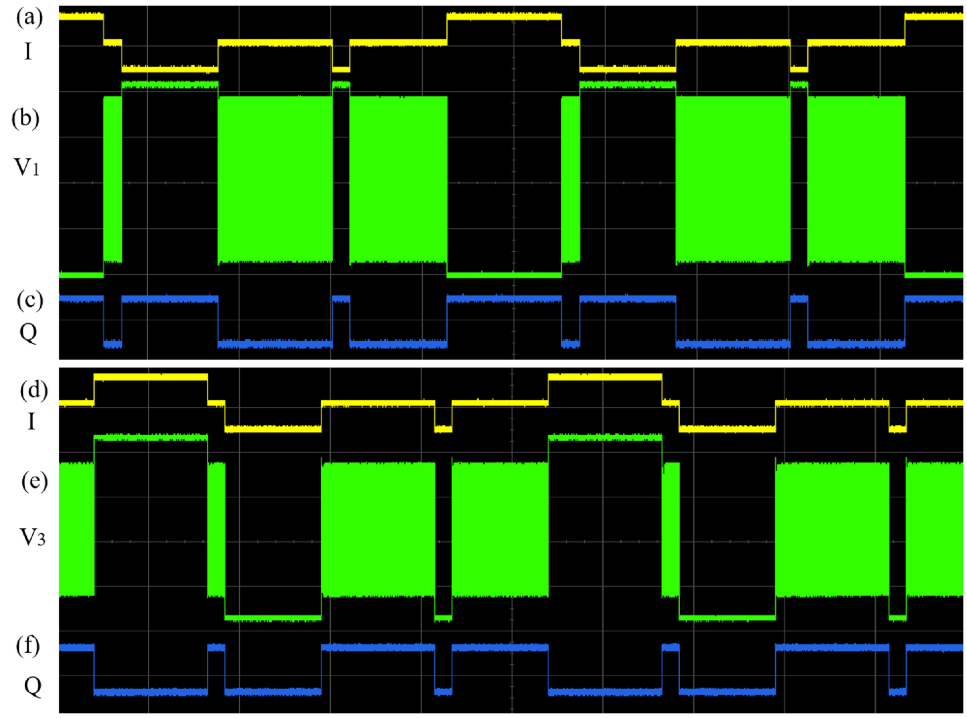


Fig. 10 Experimental observation of the XNOR/XOR logic gates in the absence of bias $E = 0.0V$ in Chua's circuit. A summation of two logic input signals, $I = I_1 + I_2$, is displayed in panel (**a**) and (**d**). The related dynamical reactions of the XNOR and XOR logic behavior of the variables $v_1(t)$ and $v_3(t)$ are shown in panels (**b**) and (**e**), respectively, while panel (**c**) and (**f**) show the traditional/static related dynamical responses Q of the XNOR/XOR logic behavior

exhibits the desired logic output in response to all the logic inputs, P(logic) is assigned a value '1', otherwise it is treated as '0'. Our numerical simulation, P(logic) for the circuit (4), is obtained by sampling 1000 runs of the given input set and this process is repeated for 500 such sets. It is clearly indicated in Fig. 11 for R_4, R_{14} and R_{20} for cells-1, 2 and 3. The system exhibits logic behavior for different cell-1(green), cell-2(red) and cell-3 (purple) in different color, respectively. Thus, we have clearly demonstrated the robustness of the logical response of the system. The switching time in SC-CNN-based Chua circuits is fundamentally determined by the RC time constant of the circuit elements. Larger RC values increase the switching time, causing slower transitions, while smaller values decrease switching time for faster transitions. This relationship is crucial for tuning the dynamic behavior and controlling the periodic or chaotic responses in Chua and SC-CNN circuits .

Logic response in multi-input configuration

Integrating Chua's circuit into a SC-CNN architecture extends its flexibility by enabling the realization of multiple logic gates within a single system. However, achieving such multifunctionality requires careful mechanisms to control system behavior. The initial condition sensitivity of Chua's circuit plays a crucial role; different initial states can lead to different trajectories that correspond to different logic operations. Input biasing of the CNN cells can effectively preselect a logic operation: feeding a specific

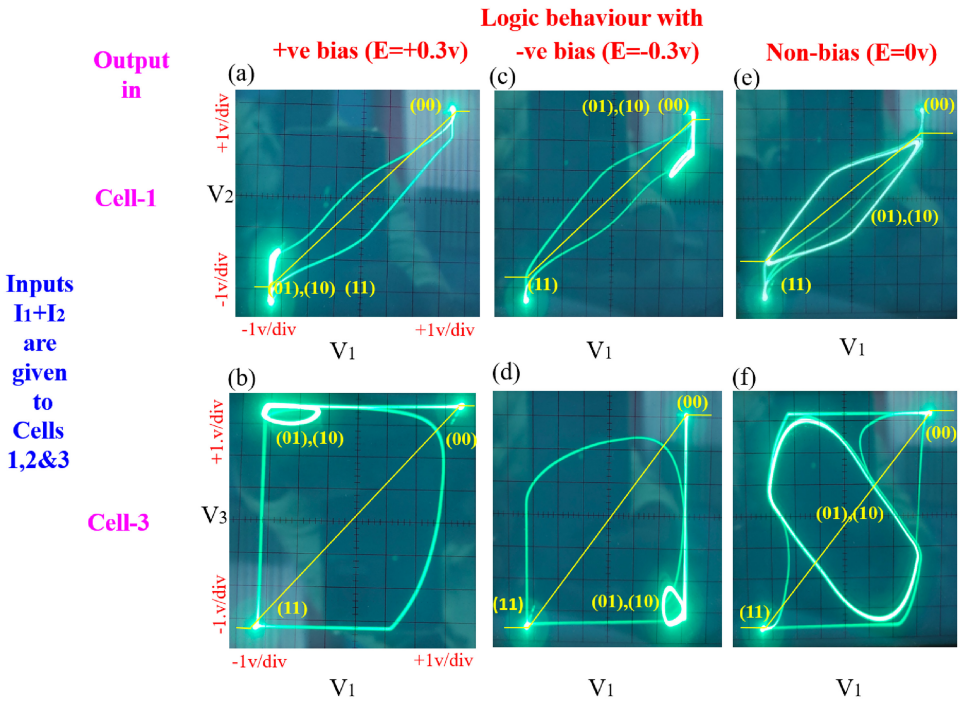


Fig. 12 Different phase space trajectories of the cells are displayed in panels (**a**)–(**f**). Panels (**a**) and (**b**) show the effect of positive bias, $E = +0.3V$; Panels (**c**) and (**d**) display the effect of negative bias, $E = -0.3V$; and Panels (**e**) and (**f**) display the effect without bias, $E = 0.0V$

Table 4 Using multi-inputs to obtain different logic behaviors

Input is given as	Output in the	With +ve bias	With -ve bias	With no bias	Parameter R	Result
$I_1 + I_2$	Cell-1	NOR	NAND	XNOR	$0.260K\Omega < R_4 < 0.390K\Omega$	NOR/NAND/XNOR gates are simultaneously obtained in cell-1 dependending on the biasing values
$I_1 + I_2$	Cell-2	NOR	NAND	XNOR	$0.690K\Omega < R_{14} < 1.0K\Omega$	NOR/NAND/XNOR gates are simultaneously obtained in cell-2 dependending on the biasing values
$I_1 + I_2$	Cell-3	XOR	XNOR	XNOR	$0.595K\Omega < R_{20} < 0.985K\Omega$	XOR/XNOR gates are simultaneously obtained in cell-3 dependending on the biasing values

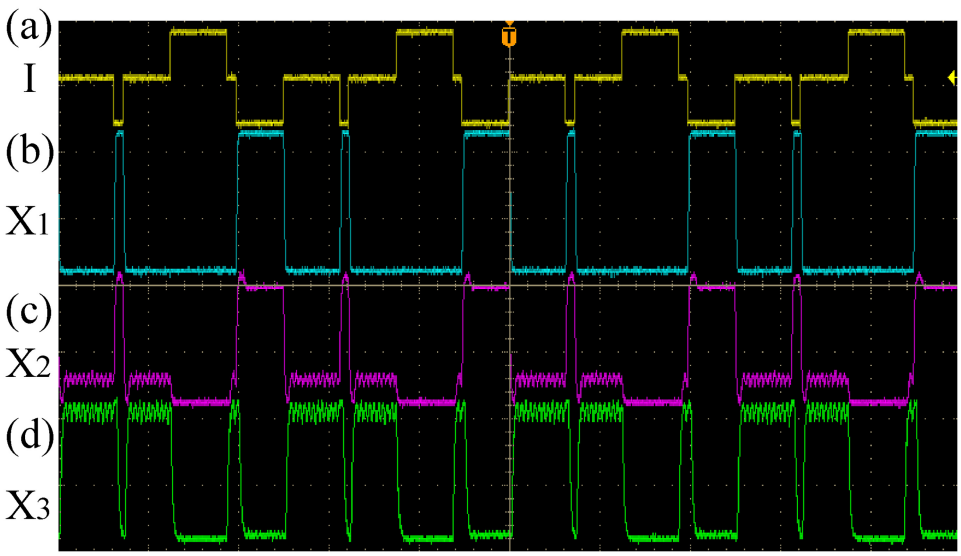


Fig. 13 Experimental observation of the NOR/XOR logic gates in experimental electronic circuit. A summation of two logic input signals, $I = I_1 + I_2$, is displayed in panel (**a**). The related dynamical reactions of the NOR logic behavior of the systems $v_1(t)$ and $v_2(t)$ are shown in panels (**b**) and (**c**), respectively, while panel (**d**) shows the dynamical responses of the XOR logic behavior of the system $v_3(t)$ with constant bias $E = +0.3V$, respectively

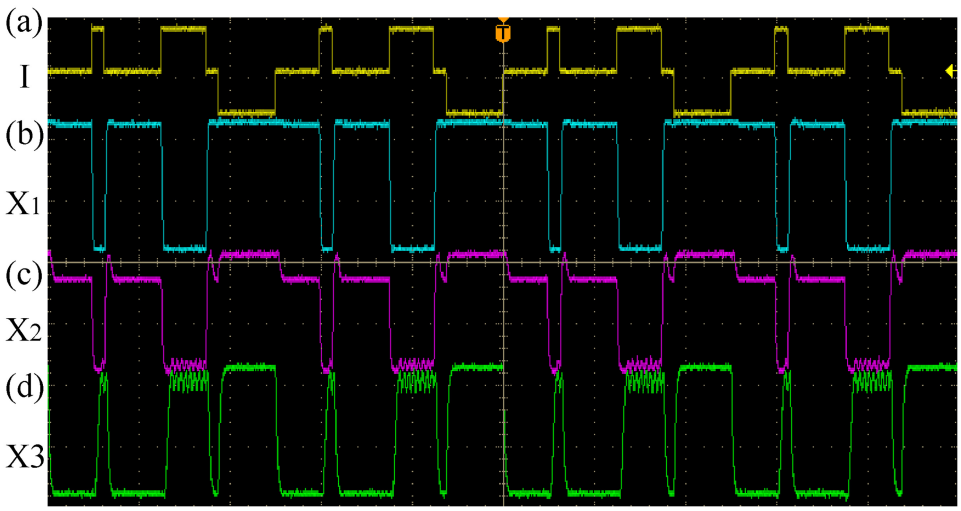


Fig. 14 Experimental observation of the NAND/XNOR logic gates in Chua's circuit. A summation of two logic input signals, $I = I_1 + I_2$, is displayed in panel (**a**). The related dynamical reactions of the NAND logic behavior of the systems $v_1(t)$ and $v_2(t)$ are shown in panels (**b**) and (**c**), respectively, while panel (**d**) shows the dynamical responses of the XNOR logic behavior of the system $v_3(t)$ with constant bias $E = -0.3V$, respectively

set of initial conditions or input voltage amplitudes can configure the system to perform as an AND gate, while a different bias can yield OR gate behavior.

Logic response when two-inputs are given to all the cells simultaneously

Fig. 12 shows the experimental results of the phase space of all the three regions. If we include the logic inputs and constant bias to all cells of the circuits, then $g(t) = I_1 + I_2 + E$ are fed simultaneously into the each cell-1, cell-2 and cell-3 and we investigate the response of the circuit (see Table 4). In Figs. 13(a), 14(a), and 15(a), we show the three level diagrams for the combinations of two logic inputs $I = I_1 + I_2$. When the bias is positive, the first and second time series $v_1(t)$ and $v_2(t)$ of the system dynamics show that the (1, 1) and (0, 1)/(1, 0) states are placed in the negative region and the (0, 0) state is placed in the positive region and they obey the NOR logic behavior [see Figs. 12(a) and 13(b),(c)]. The output of the third time series $v_3(t)$ represents the dynamics of the (0, 0) and (1, 1) states placed in the negative region and the (0, 1)/(1, 0) state placed in the positive region and it obeys the XOR logic behavior [see Figs. 12(b) and 13(d)]. Now we change the bias value from $E = +0.3V$ to $E = -0.3V$ and observe that the first and second time series of the system dynamics corresponding to the (0, 0) state is placed in the positive region and that of the (0, 1)/(1, 0) and (1, 1) states are placed in the negative region and they obey the NAND logic behavior [see Figs. 12(c) and 14(b), 14(c)]. The third time series output $v_3(t)$ represents the dynamics of the (0, 0) and (1, 1) states placed in the positive region and (0, 1)/(1, 0) state placed in the negative region and it obeys the XNOR logic behavior [see Figs. 12(d) and 14(d)]. Finally, when the constant bias is zero the system dynamics

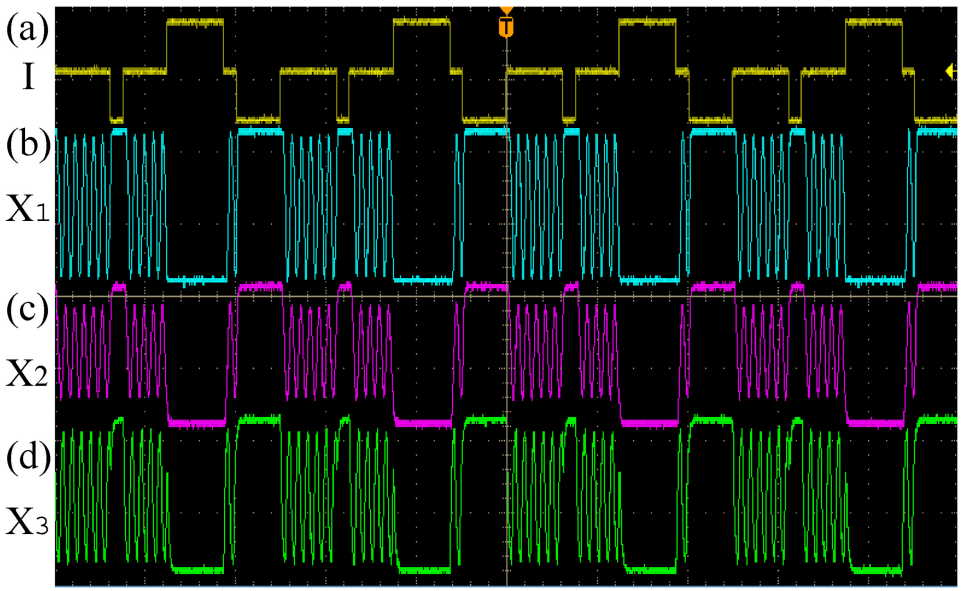


Fig. 15 Experimental observation of the XNOR logic gate in Chua's circuit. A summation of two logic input signals, $I = I_1 + I_2$, is displayed in panel (**a**). The related dynamical reactions of the XNOR logic behavior of the systems $v_1(t)$, $v_2(t)$ and $v_3(t)$ are shown in panels (**b**), (**c**) and (**d**) in the absence of bias, $E = 0.0$

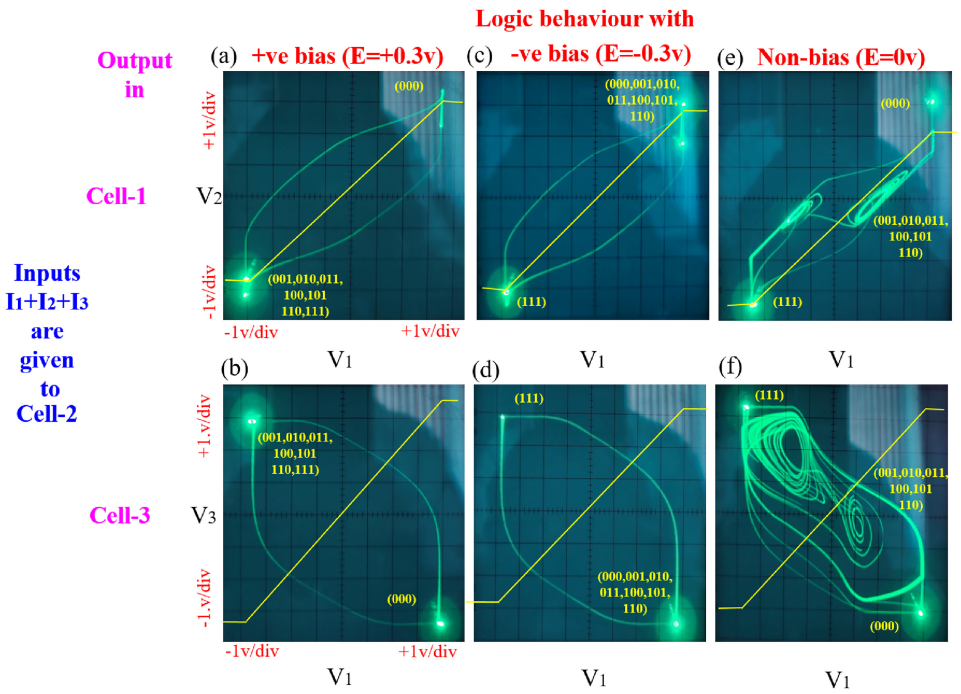


Fig. 16 Different phase space trajectories of the cells are displayed in panels (**a**)–(**f**). Panels (**a**) and (**b**) show the effect of positive bias, $E = +0.3V$; Panels (**c**) and (**d**) display the effect of negative bias, $E = -0.3V$; and Panels (**e**) and (**f**) display the effect without bias, $E = 0.0V$

of the first, second and third time series of (0, 0) state is placed in the positive region and (1, 1) state is placed in the negative region but (0, 1)/(1, 0) state oscillates in between positive and negative regions and it obeys the XNOR logic behavior [see Figs. 12(e), 15(b) and (c)]. The third time series output $v_3(t)$ represents the dynamics of the (0, 0) state placed in the negative region and the (1, 1) state is placed in the positive region while the (0, 1)/(1, 0) state is in between the positive and negative regions and they obey the XNOR logic behavior [see Figs. 12(f) and 15(d)].

Logic responses in a three-input configuration fed into any one of the three cells

Next, we explore if there is a way to expand the present nonlinear circuit system's usage to include three or even more inputs. We have observed that even when feeding three inputs, the circuit produces logic responses without making any change in the sytem parameters. When we apply the inputs to each one of the three different cells of the SC-CNN Chua's circuit, we obtain different responses

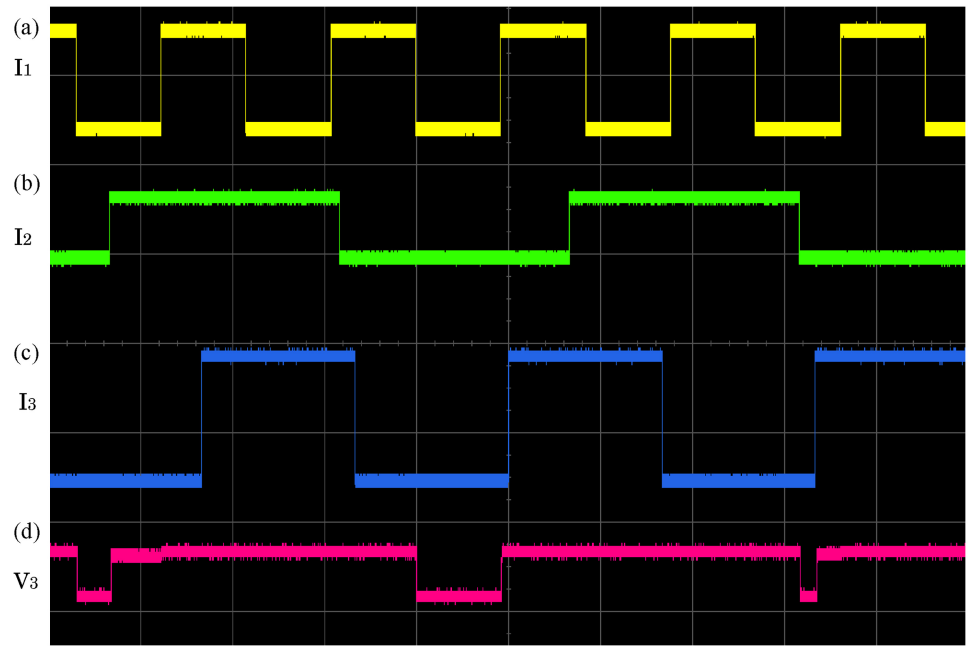


Fig. 17 Experimental observation of the OR logic gate in Chua's circuit. Three separate input signals, I_1, I_2, and I_3, are shown in Panels (**a**) through (**c**). The variable $v_3(t)$'s associated dynamic response of OR logic behavior is shown in panel (**d**)

Table 5 Truth table for three-input OR gate realization

Logic input $I_1\ I_2\ I_3$	Logic output
0 0 0	0
0 0 1	1
0 1 0	1
0 1 1	1
1 0 0	1
1 0 1	1
1 1 0	1
1 1 1	1

in every one of the three cells (see Table 5). As a specific example, in Fig. 16, we feed three inputs, I_1, I_2, and I_3, to the cell-2 of the circuit and investigate the corresponding responses v_1, v_2 and v_3 of all the cells of the circuit. For example, for obtaining the OR gate, we fix the experimental parameters as follows : $I_1 = \pm 600mV$, $I_2 = \pm 600mV$, $I_3 = \pm 600mV$, and $E = +0.4V$. The first three panels of Figs. 17(a-c) show the three different inputs I_1, I_2, and I_3, then $g(t) = I_1 + I_2 + I_3 + E$ given into the cell-2 while the fourth panel corresponds to logical OR dynamics, Figs. 16(b) and 17(d).

To obtain the OR logic operation, the corresponding any on of the logic inputs $(I_1/I_2/I_3)$ is high the output of the system is high otherwise the system output is low. Now, we analyze the trajectory of the circuit in the (v_1, v_3) plane. This confirms that the output of the circuit in the cell performs as the logical OR operation, as shown in Fig. 17(d). Furthermore, it is noted that when the bias is changed from the value $E = +0.4V$ to $E = -0.4V$, the output of each one of the system's circuit variables $v_3(t)$ changes from OR logic behavior to AND logic behavior. In the absence of the bias E, we realize the XOR gate (as in the two input signals case discussed previously). When the output voltage variable $v_3 < 0$, it is set to be logic '0', and it is considered to be logic '1' when $v_3 > 0$ (see Table 6). Similarly, we can also implement the traditional/static logic response Q in three-input and multi-input configurations with assistance of the detector circuit (Ref. Sec. 5).

Set-reset memory latch

In addition to developing logic gates, several studies have concentrated on using nonlinear systems to build flip-flop. The inputs for an SR flip-flop should be distinct since they result in several output states, as shown by the associated Truth Table 7. As a result, the inputs are encoded differently. In this case, the input I_1 is $'1'$ for logic input '1' and '0' for logic input '0'.

The second input, I_2, is complimentary to I_1, however. In other words, when the logic input I_1 is $'1'$, the input I_2 takes the value $'0'$, whereas it takes the value $'1'$ for the logic input $'0'$. Experimentally, this may be accomplished by applying the NOT operator on I_2. The logic values $0, -1, 1$, and 0 are represented for this example by the input streams $(I_1, I_2) : (0, 0), (0, 1), (1, 0)$, and $(1,\ 1)$, respectively. The $(1,\ 1)$ set in this case is a constrained one. For instance, both inputs for the logical input '0' are $I_1 = I_2 = -500mV$, whereas the values are $+500mV$ for the logical input '1'. [See Fig.18(a), (b), (e) and (f)]. This results in a three-level

Table 6 Logic behaviors using three different inputs

Input $I_1 + I_2 + I_3$ are given in	Output in the	With +ve bias	With -ve bias	With no bias	Parameter R	Result
	Cell-1	NOR	NAND	XNOR		
Cell-1	Cell-2	–	–	–	$0.260K\Omega < R_4 < 0.390K\Omega$	The response of first cell exhibits NOR/NAND/XNOR depending upon the bias while the third cell exhibits the complement of the first cell, namely OR/AND/XOR
	Cell-3	OR	AND	XOR		
	Cell-1	NOR	NAND	XNOR		
Cell-2	Cell-2	NOR	NAND	XNOR	$0.690K\Omega < R_{14} < 1.0K\Omega$	The response of the first cell & second cell exhibits NOR/NAND/XNOR dependending upon the bias while the third cell exhibits the complement of the first cell, namely OR/AND/XOR
	Cell-3	OR	AND	XOR		
	Cell-1	NOR	NAND	XNOR		
Cell-3	Cell-2	NOR	NAND	XNOR	$0.595K\Omega < R_{20} < 0.985K\Omega$	NOR/NAND/XNOR gates are simultaneously obtained in all the three cells, dependending on the biasing values
	Cell-3	NOR	NAND	XNOR		

Table 7 Set-Reset (SR) flip-flop truth table

(I_1) Set	(I_2) Reset	Output	State
0	0	Q	Last state
0	1	0	Reset
1	0	1	Set
1	1	?	Not allowed

wave shape for the input signal I given the cell-2, $I = I_1 - I_2$: $-1V$ for (0, 1), 0 for (0, 0)/(1, 1), and $1V$ for (1, 0) input sets [Fig.18(c) and (g)].

The SR flip-flop can be realized in the circuit when we employ the input streams as specified above. As usual, we assume the logical output is to be '1' when $v_1 > 0$ and 0 for $v_1 < 0$. The SR flip-flop operations produced by the circuit is as shown in Fig. 18. When both the inputs I_1 and I_2 are in the low states, it is observed that the response of the circuit $v_1(t)$ remains unchanged. Then, when the input I_1 is in the low level logic state and the input I_2 is in the high level logic state it is noticed that the value $v_1(t) < 0$ and hence it is assumed that the logic output is a low logic level. As a result, the latch is said to be Reset. On other hand, when I_1 is high while I_2 is in low logic level, the response of the circuit oscillates in the region $v_1(t) > 0$ and thus it is assumed that the logic output is to be the high level logic. Also the voltage variable v_2 exhibits the same logic output as that of the voltage variable v_1. Thus the circuit behaves as a latch with Set condition [see Fig. 18(d)].

It is obvious in Fig. 18(h) that the output of the dynamical variable $v_3(t)$ is the inverted output of the variable $v_1(t)$ and the response of the circuit behaves as an active high RS flip-flop. Active low and active high RS flip-flop are produced in digital electronics using two cross-coupled NAND and NOR gates, respectively. However, in the present study, these two kinds of flip-flop have been obtained via two dynamical variables, namely $v_1(t)$ and $v_3(t)$. As a result, the considered circuit is not only able to produce all the fundamental logic gates but also produces RS flip-flop of various categories, namely the active low/high RS flip-flop. Similarly, we can also obtain the traditional/static RS flip-flop by using an appropriate detector circuit (see Fig. 2) Campos-Cantón et al. (2010); Ashokkumar et al. (2021).

Effect of noise on logic gates

Now, we investigate the effect of noise on the logic behavior of the circuit system, Eq.(4). To realize the effect of noise, we include the optimal window noise $\eta(t)$ as an additive Gaussian noise with noise strength D. The output of the

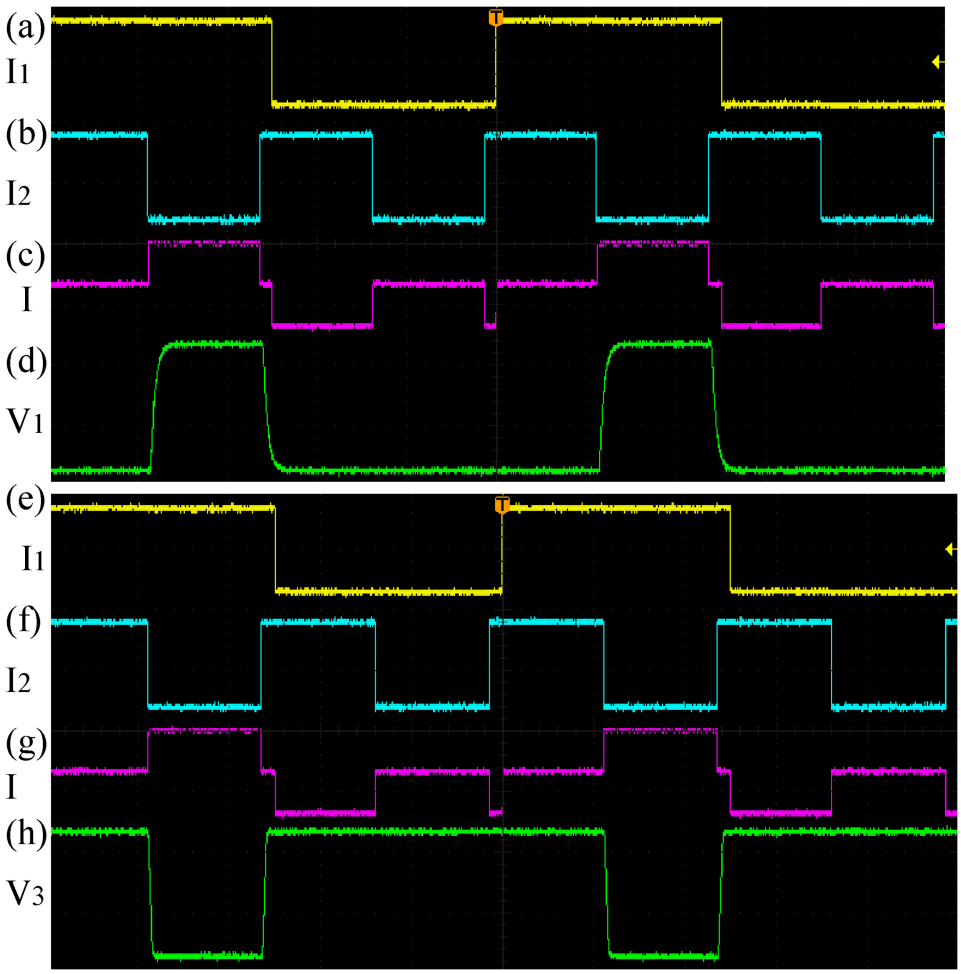


Fig. 18 Experimental observation of Set-Reset memory latch in Chua's circuit. Panels (**a**), (**b**) and (**e**), (**f**) show different input signals I_1, I_2. Panels (**c**) and (**g**) show the three level input I ($I = I_1 - I_2$). Panels (**d**) and (**h**) represent the corresponding dynamical responses of Set-Reset behavior of the system $v_1(t)$ and $v_3(t)$

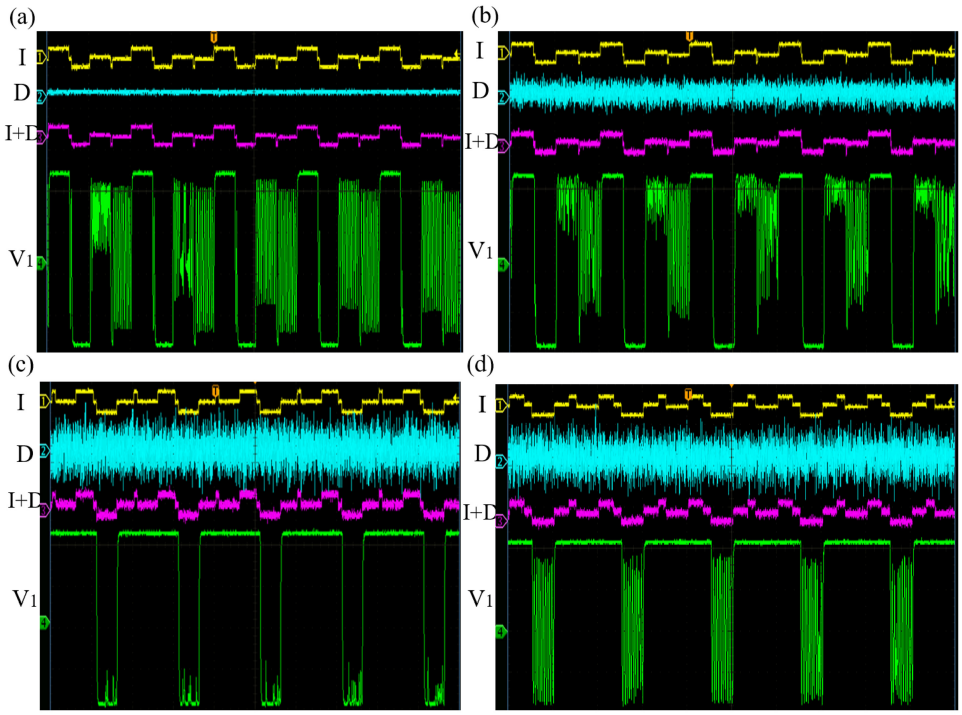


Fig. 19 Experimental observation of the OR logic gate in Chua's circuit. Panels (**a**)–(**d**) show the three level input set signals $I = I_1 + I_2$ and their corresponding output $v_1(t)$. In Panels (**a**)–(**d**) the noise strength varies from $D = 0V, D = 1.6V, D = 2.4V$ & $D = 2.9V$, respectively

system is determined by its voltage variable. If we include the two different logic inputs and Gaussian white noise, in the form $g(t) = I_1 + I_2 + D\eta(t)$ are fed into the cell-2 and we investigate the response of the circuit.

Specifically, it is assumed that the voltage variable I (see Fig. 19) is combination of two logic input signals ($I = I_1 + I_2$) (Ref.Sec.4.1), for appropriate noise strength D which will assist the system output $v_1(t)$ in moving from one region to another region for ambient noise strength. Now, we analyze the trajectory of the circuit in the (v_1, v_2) plane. When the noise strength lies in the region $D \in [0, 2.9]$, for low value of noise for the input (0, 1)/(1, 0) case the output oscillates between two regions. On further increase in the noise strength D, the output gradually moves to one region. For the optimal window of noise strength, the system exhibits the desired logic behavior (OR gate) in the corresponding output $v_1(t)$. On further increase in the noise strength, the system output moves randomly in between two regions/completely settled in one region, so that the system does not exhibit any logic behavior (see Fig. 19). Here noise is taken to have a correlation time much smaller than any other time scales in the system. The strength of noise correlated to the input signal is a fraction, as shown in Fig. 19. The effect of noise on the input signal is also shown in Fig. 19. In Fig. 19(a) we have shown the combination of two inputs, $I = I_1 + I_2$, and the corresponding output in the absence of noise, so that $D = 0V$. In Fig. 19(b) in the presence of noise with strength $D = 1.6V$ the system output is initiated to exhibits the logic behavior. On increase in the noise strength to $D = 2.4V$, the system completely shows the OR logic behavior [see Fig. 19(c)]. On further increase in the noise strength to $D = 2.9V$, the system completely losing the OR logic behavior [see Fig. 19(d)].

Now, we investigate the effect of noise on the logic behavior of this circuit system (4). For this propose, we calculate the probability of getting logic gates for different noise strength levels with a strength D. Essentially, the probability denotes the total number of successful runs/total number of runs. If the system exhibits the desired logic output in response to all the logic inputs, probability is assigned a value of '1', otherwise, it is treated as '0'. Probability for the circuit (4) is clearly indicated in Fig. 20. Initially, the system output oscillates randomly after adding the Gaussian white noise, and it slowly approaches the logic behavior and on further increase in the noise strength to $1.8V < D < 2.7V$, the system exhibits logic behavior. Beyond $D > 2.8V$, the system loses its logic behavior. Thus, it is clearly demonstrated that in the optimal window of noise, the system exhibits logic behavior. The absence of tolerance analysis can lead to unpredictable behaviors, incorrect logic outputs, and system instability. This report investigates: (i) the sensitivity of SC-CNN-based Chua's circuit to component variations; (ii) possible deviations in logic gate behavior due to tolerances; and (iii) mitigation techniques and design strategies for tolerance resilience. The reliability and correct operation of SC-CNN-based Chua's circuit logic gates are significantly influenced by real-world component tolerances. Given the nonlinear and chaotic nature of the underlying system, even small parameter variations can lead to large deviations in behavior.

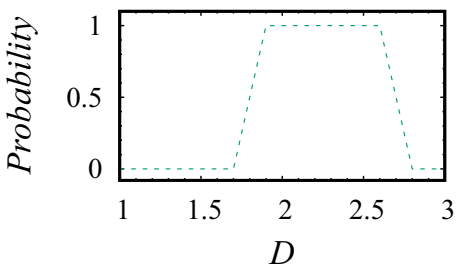


Fig. 20 The probability distribution of obtaining logical behavior in logical stochastic resonance for different values of D for the OR logic gate

Here, we use the Agilent 33220A function generator to produce noise-like signals. Also, we generate synthetic noise signals using its arbitrary waveform generation (AWG) capabilities. Arbitrary waveform generators (AWGs) provide enhanced flexibility, precision, and customization compared to traditional noise generators, enabling tailored signal synthesis for complex applications like chaos computing, optical systems, and multi-channel logic characterization. AWGs enable dynamic waveform synthesis beyond standard noise profiles (e.g., Gaussian or white noise). For instance: Sinc and Cardiac waveforms can simulate transient noise behaviors. By integrating these AWG-generated signals with logic inputs, we confirmed that the system outputs exhibit valid logical behavior (see Fig. 21). Thus, it is clearly demonstrated that the system exhibits logical behavior under different noise signals.

Also, the Gaussian white noise serves as an invaluable input signal for analyzing nonlinear circuits due to its broadband spectral characteristics and stochastic properties. The use of white noise facilitates nonparametric system identification techniques, including the rigorous validation of theoretical models by comparing simulated and empirical responses under stochastic excitation. The random nature of white noise is also effective in uncovering subtle nonlinear phenomena such as bifurcations, memory effects, and sensitivity to initial conditions which are hallmarks of chaotic or near-chaotic systems. Moreover, since practical electronic circuits often operate in noisy environments, testing with Gaussian white noise provides a realistic framework for evaluating noise robustness and ensuring reliable performance under real-world conditions. Hence, Gaussian white

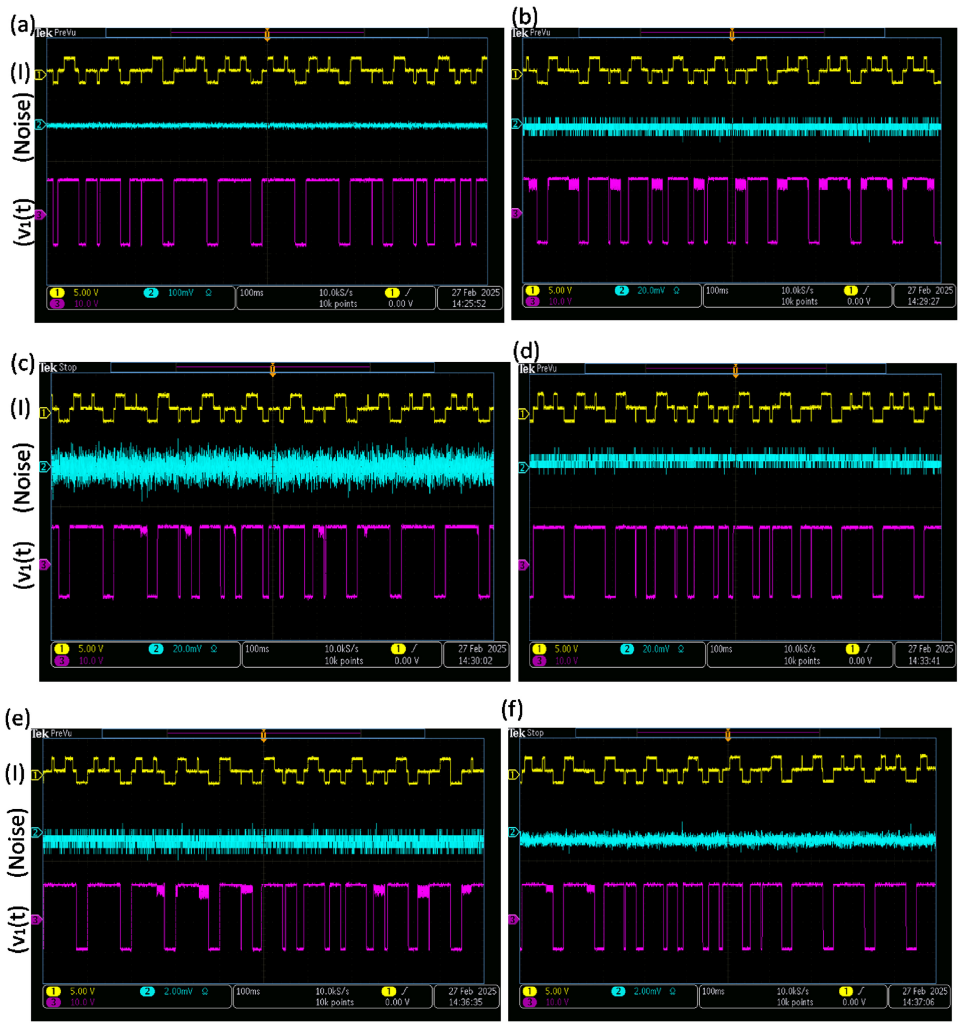


Fig. 21 Experimental observation of the OR logic gate in Chua's circuit. Panels (**a**)–(**f**) show the three level input set signals $I = I_1 + I_2$, different noise signals and their corresponding output $v_1(t)$. Specifically panels (**a**)–(**f**) correspond to different noise signals, namely Gaussian white noise, white noise with positive burst, white noise with negative burst, arbitrary waveform, arbitrary waveform with sinc and arbitrary waveform with cardiac waveform, respectively

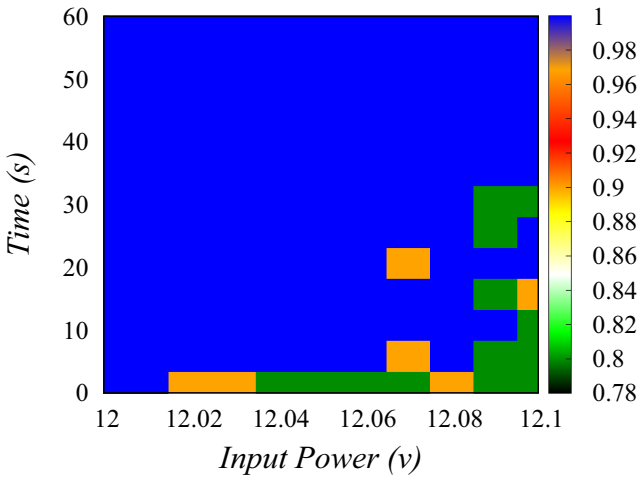


Fig. 22 Two-phase diagram for Input power (v) vs. Time (s) taken during circuit activation

noise offers a versatile and rigorous means for characterizing the complex behavior of nonlinear electronic systems. To exploit LSR, synthetic noise may need to be intentionally injected, or system gains and thresholds must be tuned appropriately. Traditionally, robust logic operation demands high signal-to-noise ratios (SNR), but with LSR, moderate noise levels actually reduce the required SNR, allowing reliable operation with lower input amplitudes. Design strategies to improve robustness include adjusting control parameters to enlarge target basins (particularly for logical '1' states in energy-penalized gates like AND), biasing the system?s initial conditions toward the center of the desired basin to enhance noise tolerance, and strengthening basin separation to maintain reliable operation even under moderate noise levels. In noisy environments, however, noise can blur or shift basin boundaries, thereby reducing reliability, especially for states initiated near basin edges, making optimized basin design and dynamic noise management strategies crucial for achieving robust real-world performance.

Table 8 Benchmark Table

Metric	SC-CNN Chua logic	CMOS logic gates	Standard Chua logic
Propagation delay	~ 2.1 ns	~ 0.5 ns	~ 3.5 ns
Power consumption	~ 25 μW	~ 180 μW	~ 4.5 μW
Noise margin	$\pm 120mV$	$\pm 80mV$	$\pm 100mV$
Area footprint	1.8 × (relative to CMOS)	1 ×	2.5 ×
Refonfigurability	High	Low	Medium

Finally, we have also studied the reliability of the logic behavior through variation of power inputs. Studying the basin of attraction is crucial in nonlinear dynamics because it helps us to understand the stability and long-term behavior of dynamical systems. In our study, we are mainly focusing on chaos computing. We first analyze the chaotic nature of the system, then introduce logical inputs and observe the resulting logical behavior within the chaotic system. Chaos computing demonstrates significant advantages including secure communication, fast switching capabilities, and the ability to transmit logic signals over long distances. Based on the literature, we have configured the system to maintain a chaotic state while embedding logical operations. We have included a two-phase diagram illustrating how power varies over time during circuit activation (see Fig. 22). The results confirm that both the selected initial conditions and system parameters successfully demonstrate logic operations under varying inputs.

Conclusion

In our previous studies we have demonstrated logical behavior with multiple external forces (e.g., sinusoidal inputs, biasing, noise) is complex. This work resolves this by leveraging intrinsic nonlinear dynamics (Ashokkumar et al. 2021). In this paper, we address a critical question: Can logical behavior emerge without external forcing besides the input signals and biasing? Here, we have studied the dynamics of the SC-CNN framework of Chua's circuit subjected to two logic input signals. In particular, we have applied the two logic inputs to each of the cells of the Chua's circuit. It is observed that the Chua's circuit has the ability to produce all the logic elements, namely AND/NAND, OR/NOR, XOR/XNOR under appropriate choice of the circuit parameters. Specifically, in the present paper, we have experimentally demonstrated the existence of all the logic gates in the circuit when streams of logic inputs are fed into the cell-2 of the circuit. It is found that the circuit produces two similar logic gates through its two voltage variables and complementary gates via its remaining third voltage variable. A simple circuit like the well-known SC-CNN Chua's circuit is able to produce two sets of logic gates of similar type and one of a complementary nature.

Further, we have extended our study on obtaining logic gates by feeding higher number of inputs in the circuit. It is observed that the system exhibits logic behaviors even feeding multi-inputs. Specifically, we have illustrated experimentally the possibility of obtaining logic gates when three-inputs are given. The function of physical logic gates with larger fan-in is slower than that of smaller fan-in gates. Since parallelism is inherited in the circuit, mere variation of a system parameter value paves the way for implementing all the logic operations. For example when two-inputs are fed into all the cells, it is observed that all logic operations are related in the circuit depending on the resistance values R_4, R_{14}, R_{20} corresponding to the cells 1, 2 and 3, respectively. In the static case, the logic operation is obtained one by one. But in the case of the dynamical system, the logic operations are obtained parallelly. That is why we have emphasized that the implementation of logic operations in dynamical logic gates is faster than static one. We have also studied the effect of noise in the corresponding circuit. It is noted that in an optimal range of noise the system clearly demonstrates the logic nature. Thus, all these logic behaviors observed in the circuit are tolerant to noise.

Our experimental findings explicitly show that the SC-CNN model of Chua's circuit functions as logic gates as well as memory latches, even for logic signals of low amplitude. It is clearly demonstrated in our experimental studies that the SC-CNN model of Chua's circuit acts as logic gates as well as memory latch even for small amplitude of logic signals. As a consequence, the circuit consumes low power. Further, we have explored the same circuit to obtain logic behaviors even for higher order inputs. Typically, two NAND gates or two NOR gates are cross-coupled to create the low active and high active RS flip-flop, respectively. However, the current work demonstrates that the same Chua's circuit may also produce low active RS flip-flop in one of its voltage variables and high active RS flip-flop in another voltage variable in addition to logic gates. Thus, the SC-CNN Chua's circuit is able to utilize its underlying dynamical features by building the circuit into two-input or multi-inputs logic elements and also RS flip-flop on the basis of requirement one needs. Also, the system exhibits logical behavior under different noise signals. Further, we have also implemented the traditional/static logic outputs using a comparator (LM311) from existing nonlinear logic outputs.

Table 9 Comparative Literature Review

Work/Reference	Circuit type	Speed	Power	Logic depth	Notes
Itoh and Chua (2005)	Analog Chua Logic	Low	Medium	2	No state control, less efficient
Manganaro et al. (2012)	CNN with chaos	Medium	Medium	3	Reconfigurable but high power consumption
Our results	SC-CNN Chua	High	Low	≥4	Best reconfigurability and energy efficiency
Traditional CMOS Logic	CMOS	High	High	5+	Standard benchmark, but limited by physical design constraints

A comparison of nonlinear dynamic and static logic gates shows that they differ fundamentally in design philosophy, performance characteristics, and application suitability. Static CMOS excel in stability and noise immunity, while dynamic logic gates offer higher speed and power efficiency but necessitates precise clock management. Also, dynamic logic gates are preferred for high-speed, low-voltage applications (e.g., CPUs). On the other hand, static logic gates dominate low-power, variation-tolerant designs (e.g., embedded systems) [see Tables 8,9].

Thus, we have proved that as the underlying attractors of the circuit hop in between different quadrants of its phase space they cause to produce parallel logic behaviors in the SC-CNN Chua's circuit and also induce memory latch. Furthermore, significant elements of this circuit, such as latency time and cell feature size, must be studied and compared to existing technology to determine the circuit's technical viability for design and development of dynamic architecture. We plan to explore these aspects in the near future. Also such studies will be helpful to implement them to develop suitable nonlinear dynamics based ICs. We do hope the present study sheds light into further studies for alternate computing.

Acknowledgements The authors would like to thank Professor K. Murali, Anna University, Chennai, India, for his suggestions. P.A. and A.V. express their gratitude to the DST-SERB for funding a research project with Grant No.EMR/2017/002813. Sincere appreciation is extended by M.S. to the Council of Scientific & Industrial Research, India for funding his fellowship via SRF Scheme No.08/711(0001)2K19-EMR-I. A.V. additionally thanks the DST-FIST for funding research projects via Grant No.SR/FST/ College-2018-372(C). M.L. thanks the DST-SERB National Science Chair program for funding under Grant No. NSC/2020/000029.

Data availability The data that support the findings of this study are available from the corresponding author upon reasonable request.

Declarations

Conflict of interest No potential Conflict of interest was reported by the author(s).

References

Abel, A., Schwarz, W.: Chaos communications-principles, schemes, and system analysis. Proc. IEEE **90**(5), 691–710 (2002)

Arena, P., Baglio, S., Fortuna, L., Manganaro, G.: Chua's circuit can be generated by CNN cells. IEEE Trans. Circuits Syst. I Regul. Pap. 42(2), 123– 125 (1995)

Arena, P., Baglio, S., Fortuna, L., Manganaro, G.: Simplified scheme for realisation of chua oscillator by using SC-CNN cells. Electron. Lett. **31**(21), 1794–1795 (1995)

Arena, P., Bucolo, M., Fazzino, S., Fortuna, L., Frasca, M.: The CNN paradigm: Shapes and complexity. Int. J. Bifur. Chaos **15**(07), 2063–2090 (2005)

Ashokkumar, P., Sathish Aravindh, M., Venkatesan, A., Lakshmanan, M.: Realization of all logic gates and memory latch in the SC-CNN cell of the simple nonlinear MLC circuit. Chaos **31**(6), 063119 (2021)

Banerjee, T.: Single amplifier biquad based inductor-free chua's circuit. Nonlinear Dyn. **68**, 565–573 (2012)

Bao, B., Wang, N., Chen, M., Xu, Q., Wang, J.: Inductor-free simplified chua's circuit only using two-op-amp-based realization. Nonlinear Dyn. **84**, 511–525 (2016)

Bulsara, A.R., Dari, A., Ditto, W.L., Murali, K., Sinha, S.: Logical stochastic resonance. J. Chem. Phys. **375**(2–3), 424–434 (2010)

Cafagna, D., Grassi, G.: Chaos-based sr flip-flop via chua's circuit. Int. J. Bifur. Chaos **16**(05), 1521–1526 (2006)

Campos-Cantón, I., Pecina-Sánchez, J., Campos-Cantón, E., Rosu, H.C.: A simple circuit with dynamic logic architecture of basic logic gates. Int. J. Bifurc. Chaos **20**(08), 2547–2551 (2010)

Campos-Cantón, I., Campos-Cantón, E., Rosu, H.C., Castellanos-Velasco, E.: SET-RESET flip-flop circuit with a simple output logic. Circuits Systems Signal Process. **31**(2), 753–760 (2012)

Campos-Cantón, E., Femat, R., Barajas-Ramírez, J.G., Campos-Cantón, I.: A multivibrator circuit based on chaos generation. Int. J. Bifurc. Chaos **22**(01), 1250011 (2012)

Canton, E.C., Martienz, M.G., Duron, R.R.R.: Method and circuit for integrating a programmable matrix in the field of reconfigurable logic gates employing a non-lineal system and an efficient programmable rewiring. Google Patents (2017)

Chua, L.O., Yang, L.: Cellular neural networks: Theory. IEEE Trans. Circuits Syst. **35**(10), 1257–1272 (1988)

Dari, A., Kia, B., Bulsara, A.R., Ditto, W.: Creating morphable logic gates using logical stochastic resonance in an engineered gene network. Europhys. Lett. **93**(1), 18001 (2011)

Dari, A., Kia, B., Wang, X., Bulsara, A.R., Ditto, W.: Noise-aided computation within a synthetic gene network through morphable and robust logic gates. Phys. Rev. E **83**(4), 041909 (2011)

Fortuna, L., Buscarino, A., Frasca, M., Famoso, C.: Control of Imperfect Nonlinear Electromechanical Large Scale Systems: from Dynamics to Hardware Implementation, vol. 91. World Scientific, Singapore (2017)

Groote, J.F., Morel, R., Schmaltz, J., Watkins, A.: Logic Gates, Circuits, Processors. Compilers and Computers. Springer, Switzerland (2021)

Guerra, D.N., Bulsara, A.R., Ditto, W.L., Sinha, S., Murali, K., Mohanty, P.: A noise-assisted reprogrammable nanomechanical logic gate. Nano Lett. **10**(4), 1168–1171 (2010)

Günay, E., Altun, K., Ünal, C.: Implementation of CSK communicating system with switched SC-CNN based chaos generator. In: 2017 10th International Conference on Electrical and Electronics Engineering (ELECO), pp. 1364– 1367 (2007). IEEE

Günay, E.: MLC circuit in the frame of CNN. Int. J. Bifurc. Chaos **20**(10), 3267–3274 (2010)

Gupta, A., Sohane, A., Kohar, V., Murali, K., Sinha, S.: Noise-free logical stochastic resonance. Phys. Rev. E **84**(5), 055201 (2011)

Hopfield, J.J.: Neural networks and physical systems with emergent collective computational abilities. Proc. Natl. Acad. Sci. **79**(8), 2554–2558 (1982)

Itoh, M., & Chua, L. O.: Nonlinear Oscillators with Hysteretic Chua's Diodes. *International Journal of Bifurcation and Chaos,* **15**(05), 1709–1735 (2005)

Kennedy, M.P.: Robust op amp realization of chua's circuit. Frequenz **46**(3–4), 66–80 (1992)

Kia, B., Lindner, J.F., Ditto, W.L.: Nonlinear dynamics as an engine of computation. Phil. Trans. R. Soc. A **375**(2088), 20160222 (2017)

Kiliç, R.: Experimental study of CFOA-based inductorless chua's circuit. Int. J. Bifurc. Chaos **14**(04), 1369–1374 (2004)

Kohar, V., Sinha, S.: Noise-assisted morphing of memory and logic function. Phys. Lett. A **376**(8–9), 957–962 (2012)

Kohar, V., Murali, K., Sinha, S.: Enhanced logical stochastic resonance under periodic forcing. Comm. Nonlinear Sci. Numer. Simulat. **19**(8), 2866–2873 (2014)

Kohar, V., Kia, B., Lindner, J.F., Ditto, W.L.: Implementing boolean functions in hybrid digital-analog systems. Phys. Rev. Applied **7**, 044006 (2017)

Lakshmanan, M., Murali, K.: Chaos in Nonlinear Oscillators: Controlling and Synchronization, vol. 13. World scientific, Singapore (1996)

Lakshmanan, M., Rajasekar, S.: Nonlinear Dynamics: Integrability. Chaos and Patterns. Springer-Verlag, Berlin Heidelberg (2003)

Li, S., Fahimi, B.: Inductor-free chua's circuit employing linear voltage-controlled resistor. In: 2018 IEEE 13th Dallas Circuits and Systems Conference (DCAS), pp. 1– 4 (2018). IEEE

Liu, Z., Zhu, X., Hu, W., Jiang, F.: Principles of chaotic signal radar. Int. J. Bifurc. Chaos **17**(05), 1735–1739 (2007)

Ma, J., Li, A.-B., Pu, Z.-S., Yang, L.-J., Wang, Y.-Z.: A time-varying hyperchaotic system and its realization in circuit. Nonlinear Dyn. **62**, 535–541 (2010)

Manaoj Aravind, V., Murali, K., Sinha, S.: Coupling induced logical stochastic resonance. Phys. Lett. A **382**(24), 1581–1585 (2018)

Mano, M.M.: Computer System Architecture. Prentice-Hall of India, India (2003)

Manganaro, G., Arena, P., & Fortuna, L.: Cellular neural networks: chaos, complexity and VLSI processing (Vol. 1). Springer Science & Business Media (2012)

Murali, K., Rajamohamed, I., Sinha, S., Ditto, W.L., Bulsara, A.R.: Realization of reliable and flexible logic gates using noisy nonlinear circuits. Appl. Phys. Lett. **95**(19), 194102 (2009)

Murali, K., Sinha, S., Ditto, W.L., Bulsara, A.R.: Reliable logic circuit elements that exploit nonlinearity in the presence of a noise floor. Phys. Rev. Lett. **102**(10), 104101 (2009)

Murali, K., Sinha, S., Kohar, V., Kia, B., Ditto, W.L.: Chaotic attractor hopping yields logic operations. PLoS ONE **13**(12), 0209037 (2018)

Neves, F.S., Voit, M., Timme, M.: Noise-constrained switching times for heteroclinic computing. Chaos **27**(3), 033107 (2017)

Peng, H., Yang, Y., Li, L., Luo, H.: Harnessing piecewise-linear systems to construct dynamic logic architecture. Chaos **18**(3), 033101 (2008)

Peng, H., Liu, F., Li, L., Yang, Y., Wang, X.: Dynamic logic architecture based on piecewise-linear systems. Phys. Lett. A **374**(13–14), 1450–1456 (2010)

Prusha, B.S., Lindner, J.F.: Nonlinearity and computation: Implementing logic as a nonlinear dynamical system. Phys. Lett. A **263**(1), 105–111 (1999)

Radwan, A.G., Soliman, A.M., El-Sedeek, A.-L.: An inductorless CMOS realization of chua's circuit. Chaos, Solitons Fractals **18**(1), 149–158 (2003)

Roychowdhury, J.: Boolean computation using self-sustaining nonlinear oscillators. Proc. IEEE **103**(11), 1958–1969 (2015)

Sathish Aravindh, M., Venkatesan, A., Lakshmanan, M.: Strange non-chaotic attractors for computation. Phys. Rev. E **97**(5), 052212 (2018)

Sathish Aravindh, M., Gopal, R., Venkatesan, A., Lakshmanan, M.: Realisation of parallel logic elements and memory latch in a quasiperiodically-driven simple nonlinear circuit. Pramana **94**(1), 1–14 (2020)

Singh, K.P., Sinha, S.: Enhancement of logical responses by noise in a bistable optical system. Phys. Rev. E **83**(4), 046219 (2011)

Sinha, S., Ditto, W.L.: Dynamics based computation. Phys. Rev. Lett. **81**(10), 2156 (1998)

Sinha, S., Ditto, W.L.: Computing with distributed chaos. Phys. Rev. E **60**(1), 363 (1999)

Sinha, S., Munakata, T., Ditto, W.L.: Flexible parallel implementation of logic gates using chaotic elements. Phys. Rev. E **65**(3), 036216 (2002)

Sinha, S., Cruz, J., Buhse, T., Parmananda, P.: Exploiting the effect of noise on a chemical system to obtain logic gates. EPL (Europhysics Letters) **86**(6), 60003 (2009)

Storni, R., Ando, H., Aihara, K., Murali, K., Sinha, S.: Manipulating potential wells in logical stochastic resonance to obtain XOR logic. Phys. Lett. A **376**(8–9), 930–937 (2012)

Swathy, P., Thamilmaran, K.: An experimental study on SC-CNN based canonical chua's circuit. Nonlinear Dyn. **71**(3), 505–514 (2013)

Venkatesh, P., Venkatesan, A.: Vibrational resonance and implementation of dynamic logic gate in a piecewise-linear Murali-Lakshmanan-Chua circuit. Commun. Nonlinear Sci. Numer. Simul. **39**, 271–282 (2016)

Venkatesh, P.R., Venkatesan, A., Lakshmanan, M.: Implementation of dynamic dual input multiple output logic gate via resonance in globally coupled Duffing oscillators. Chaos **27**(8), 083106 (2017)

Venkatesh, P.R., Venkatesan, A., Lakshmanan, M.: Design and implementation of dynamic logic gates and RS flip-flop using quasiperiodically driven Murali-Lakshmanan-Chua circuit. Chaos **27**(3), 033105 (2017)

Worschech, L., Hartmann, F., Kim, T., Höfling, S., Kamp, M., Forchel, A., Ahopelto, J., Neri, I., Dari, A., Gammaitoni, L.: Universal and reconfigurable logic gates in a compact three-terminal resonant tunneling diode. Appl. Phys. Lett. **96**(4), 042112 (2010)

Xi, F., Chen, S., Liu, Z.: Chaotic analog-to-information conversion: principle and reconstructability with parameter identifiability. Int. J. Bifurc. Chaos **23**(12), 1350198 (2013)

Zamora Munt, J., Masoller, C.: Numerical implementation of a VCSEL-based stochastic logic gate via polarization bistability. Opt. Express **18**(16), 16418–16429 (2010)

Zhang, L., Song, A., He, J.: Effect of colored noise on logical stochastic resonance in bistable dynamics. Phys. Rev. E **82**(5), 051106 (2010)

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.