Noise-Aided Invertible Logic from Coupled Nonlinear Systems

K. Murali,[1] Manaoj Aravind,[2] and Sudeshna Sinha[3,*]

[1] *Department of Physics, Anna University, Chennai 600 025, India*

[2] *School of Science, Constructor University, Campus Ring 1, Bremen 28759, Germany*

[3] *Indian Institute of Science Education and Research Mohali, Sector 81, Manauli, Punjab PO 140 306, India*

(Received 16 April 2023; revised 16 May 2023; accepted 25 August 2023; published 19 September 2023)

Invertible logic is a powerful new unconventional computing paradigm, providing bidirectional operations between inputs and outputs. It has found applications in important critical problems, such as integer factorization and machine learning. Here we propose a network of interconnected nonlinear systems that serve as our probabilistic bits ("p-bits") to implement invertible logic in the presence of a noise floor. In the forward (or directed) mode, the inputs are fixed in our network, yielding outputs in accordance with AND, OR, NAND, and NOR logic functions. In the reverse (inverted) mode the output is clamped in the network, and the input nodes fluctuate among all possible logical input values consistent with the different logic functions. So the system acts as a unique invertible logic circuit by exploiting the probabilistic transitions between the dynamical states of the coupled noisy nonlinear systems. Interestingly, both the directed and the inverted mode are most robust and reliable in an optimal band of moderate noise, reminiscent of stochastic resonance. The concept is verified in proof-of-principle electronic circuit experiments, demonstrating the robustness of the architecture and the potential of this idea to be realized in a wide range of physical situations.

DOI: 10.1103/PhysRevApplied.20.034041

I. INTRODUCTION

Invertible logic circuits potentially allow for a range of powerful features. They allow entire combinational circuits to be used completely in reverse. For instance, a multiplier could also be used as a factorizer. Also partial combinations of inputs and outputs could be presented, with the remaining terminals available as outputs. For instance, an adder fed with the sum and a single operand could output subtraction, while an invertible multiplier circuit could also perform division. There is also the potential of hardware cost reduction through multiple reuse of invertible logic circuits for different purposes. More importantly, invertible logic circuits are of immense interest in the fields such as cryptography and computer graphics, where reversible computing has demonstrated serious promise [1–8].

In recent years there has therefore been an intense focus on the use of probabilistic bits ("p-bits") which represent stochastic units that can be interconnected to implement Boolean logic functions [1], as this new computing paradigm has the ability to carry out bidirectional operations in invertible mode between inputs and outputs, unlike standard binary logic with unidirectional operations. These p-bits are intermediate between the logic bits used in conventional digital electronics and the q-bits used in quantum computing. A p-bit is a robust classical entity fluctuating in time between 0 and 1, and has the ability to interact with other p-bits in the same system [1]. These p-bits have been realized with nanomagnets [1], spintronic devices [1], and CMOS circuits [5–7].

Now, in another recent active research direction, it has been shown that noisy nonlinear systems possessing multiple wells, subjected to two square waves as inputs, produce consistent logical responses in an optimal window of noise [9–16]. The probability of getting the correct logic response increases to unity with increasing noise intensity, and then decreases for noise strengths exceeding the optimal noise strength, reminiscent of stochastic resonance [17–20]. So this novel effect has been named *logical stochastic resonance* (LSR), and it has been experimentally tested and used in a variety of fields. Specifically, LSR has been realized in electronic circuits [21], quantum dots [22,23], nanomechanical oscillators [24,25], optical systems [26], chemical [27] and biological systems [28–32], as well as in systems with multiple chaotic attractors [33], strange nonchaotic systems [34,35] and coupled nonlinear systems [36–38]. The generality of the phenomena under different kinds of noise has also been demonstrated [39–41].

However, so far, the concept of computational architectures based on noise-aided dynamics has only been utilized

*sudeshna@iisermohali.ac.in

to realize unidirectional logic gate structures, and the ability to implement invertible logic operations has not been demonstrated to date in the context of LSR. The focus of this work is to explore this promising open question, and to determine if phenomena analogous to stochastic resonance can be exploited to obtain gates that can operate in both forward (direct) and reverse (inverted) modes.

Here we consider three interconnected multistable elements, with nonlinearity tunable by a simple bias, serving as dynamical p-bits. We will demonstrate that our proposed network can successfully and reliably implement invertible logic, in the presence of noise. Specifically, we will show that, in an optimal window of noise, we obtain the following response. When the inputs are fixed, the system yields the correct AND, OR, NAND, or NOR logical outputs, thus operating as a direct logic gate. Additionally, when the output is fixed, the network fluctuates among all possible logical inputs that are consistent with the logical output, thus operating as an inverted logic gate. So, by exploiting the noise-aided hopping between dynamical states, our network of p-bits obtains the capability of effectively working as an invertible logic gate.

The plan of the paper is as follows. In Sec. II we first propose the general concept for implementation of p-bit architectures using a network of three nonlinear bistable systems in the presence of noise. We then explicitly demonstrate the success of our proposed network as an invertible logic gate in an optimal window of noise. Section III presents a proof-of-principle experimental demonstration of an invertible logic gate by using analog simulation circuits. Finally, in Sec. IV, we present a summary and discussion of the scope and outreach of our results.

II. NOISE-AIDED INVERTIBLE LOGIC GATES

The basic constituent of our gate is a p-bit, which is a nonlinear dynamical system given by

$$\dot{x} = F(x) + C + D\eta(t). \tag{1}$$

Here $F(x)$ is a nonlinear function obtained as the gradient of a bistable potential with a positive and a negative fixed point. The additive noise $\eta(t)$ is a zero-mean Gaussian white noise with unit variance, and D reflects the noise strength. The noise is considered to have correlation time smaller than any other time scale in the system, and such that it may be represented theoretically as delta correlated. The constant C controls the relative depths of the wells and can act as a simple and efficient knob regulating the "bias": C effectively modifies the shape of the nonlinear potential underlying the dynamics and determines the switching characteristics of the system in the presence of noise.

For weak noise intensity the dynamics is confined to either the positive well or the negative well depending upon the initial conditions. At an appropriate noise intensity switching between the wells is initiated [36,42]. When the bias is close to zero (i.e., $C \sim 0$), the wells are almost symmetric and the residence time of the state in either well is almost equal. However, as C changes the state is biased toward one of the wells. When C is positive, the system resides with higher probability in the positive well, as the potential is skewed to the positive well which is deeper. At sufficiently large positive C the system is in a positive state ($x(t) > 0$) with probability close to 1. Similarly, for negative C the dynamics is skewed to the negative well and for sufficiently negative C the system is in a negative state ($x(t) < 0$) with probability close to 1.

Since the two fixed points occupy markedly different regions of state space, it allows a clear mapping to binary values, and so the dynamical states of the system can be exploited to obtain logic operations. For instance, if $x(t)$ is positive, we can map it to the state 1, and if $x(t)$ is negative, we can map it to the state 0. Thus we can generate an output $m(t) = \mathrm{sgn}(x(t))$, which is a bipolar signal, with $+1$ representing the binary output 1 and -1 representing the binary output 0. Note that the bias term C controls the probability of switching between the binary outputs obtained from the state of the system. The output characteristics of the system with respect to the bias C are shown in Fig. 1, which depicts the dynamical transitions of the states as the bias C is varied. So C serves as a control "knob" that is very simple to manipulate, and can be exploited for implementing different logic operations. In this work we will use the patterns evident in Fig. 1 of the system, with respect to bias, to design both forward and reverse logic gates.

Figure 2 shows the estimated probability $P(1)$ of obtaining a positive state ($m = +1$) for varying bias C. Clearly this probability can be varied over the entire range from 0 to 1 using the bias term C in a simple and robust manner. This powerful property will serve as the basis of our proposal for inverted logic.

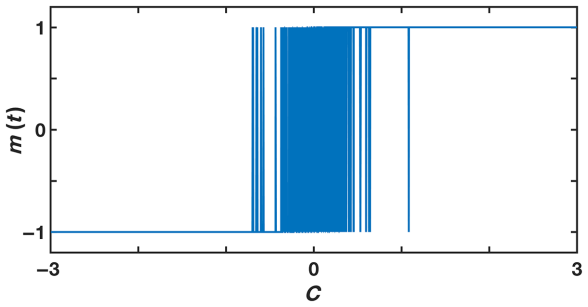

FIG. 1. Characteristics of the binary output $m(t) = \mathrm{sgn}(x(t))$ under varying bias C, of the p-bit given by Eq. (1), with $F(x) = x - x^3$, for noise strength $D = 1.0$. The binary state m of the system switches between $+1$ and -1 states in the presence of noise. The bias term C controls the probability of residence time in the two states by modifying the shape of the nonlinear potential underlying the dynamics.

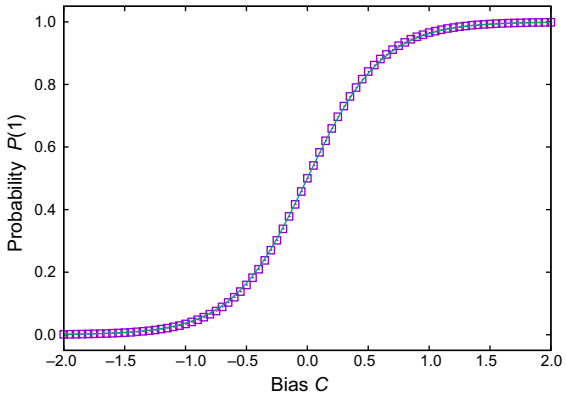

FIG. 2. Probability of obtaining a positive state ($m(t) = +1$) under varying bias C from the p-bit in Fig. 1, for $D = 1.0$. The symbols denote values obtained through numerical simulations of Eq. (1), and the curve denotes the best-fit sigmoid function: $P(1) \sim (1/(1 + e^{-\alpha C}))$, where α is a decreasing function of noise strength D. In order to estimate the probabilities, we have a considered a set of 10^5 initial states, uniformly sampled in a range $[-5, 5]$.

Now we use three such bistable systems (our p-bits) as nodes in a network in order to implement invertible logic in the presence of a sufficiently strong noise floor. As a testbed of our concept we consider the prototypical example of fundamental NOR, NAND, OR, and AND logic gates with two inputs and one output. This is realized through three connected p-bits represented as nodes A, B, and Y in the network configuration displayed schematically in Fig. 3, with A and B serving as input nodes and Y serving as the output node.

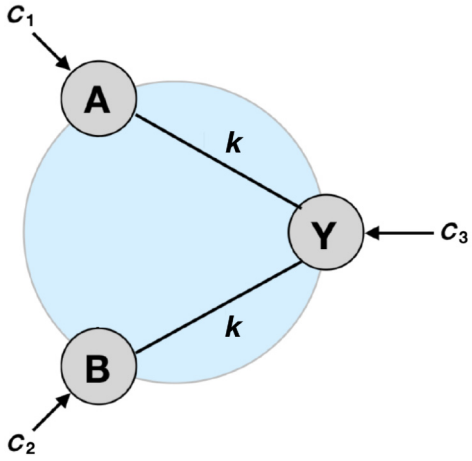

FIG. 3. Schematic of a network of nonlinear bistable elements implementing an invertible logic gate in the presence of noise. Here A, B, and Y are nodes whose dynamics are determined by Eq. (2), with A and B corresponding to the two inputs and Y corresponding to the output. C_1, C_2, and C_3 are external bias terms. The node interconnection weight is k.

TABLE I. Relationship between the two inputs (I_1,I_2), and the output of the fundamental OR, AND, NOR, and NAND logic operations, for the four distinct possible input sets $(0, 0)$, $(0, 1)$, $(1, 0)$, and $(1, 1)$ [43].

(I_1,I_2)	OR	AND	NOR	NAND
(0, 0)	0	0	1	1
(0, 1)/(1, 0)	1	0	0	1
(1, 1)	1	1	0	0

Denoting the state variables of nodes A, B and Y by x_1, x_2, and x_3, respectively, the dynamics of the network is given by the coupled system of equations:

$$\begin{aligned} \dot{x}_1 &= F(x_1) + C_1 + km_3 + D\eta_1, \\ \dot{x}_2 &= F(x_2) + C_2 + km_3 + D\eta_2, \\ \dot{x}_3 &= F(x_3) + C_3 + k(m_1 + m_2) + D\eta_3. \end{aligned} \tag{2}$$

Here $m_i = \mathrm{sgn}(x_i)$, $i = 1, 2, 3$, and η_1, η_2, and η_3 are additive zero-mean Gaussian white noise with unit variance and intensity D. C_1, C_2, and C_3 are external bias terms for nodes A, B and Y, and k is the strength of coupling between input nodes A and B, and the output node Y.

Now, to obtain logic operations, one needs to implement a robust input-to-output correspondence, as given by the truth table of the fundamental logic relations (cf. Table I). For the forward (directed) logic operations, the inputs need to be fixed. This is done by fixing the biases C_1 and C_2 of the input nodes A and B to values corresponding to the logic inputs I_1 and I_2, respectively. Without loss of generality, for the specific implementation presented here we consider $C_1 = -1$ to correspond to logic input $I_1 = 0$ and $C_1 = +1$ to correspond to logic input $I_1 = 1$. Similarly $C_2 = -1$ corresponds to $I_2 = 0$ and $C_2 = +1$ corresponds to $I_2 = 1$. The output is determined by the state of output node Y, with $x_3 < 0$ corresponding to logic output 0 and $x_3 > 0$ corresponding to logic output 1. Table II presents the necessary and sufficient conditions on the state x_3 of the output node Y, for the realization of different fundamental logic gates.

For the reverse (inverted) logic operations, the output needs to be fixed. This is done by clamping the bias C_3 of the output node Y to values reflecting the two different logic outputs. Without loss of generality, in the results presented here we have $C_3 = -2$ corresponding to logic output 0 and $C_3 = 2$ corresponding to logic output 1. The inputs corresponding to the given output are determined by the states of the two input nodes A and B that emerge when the logic output is fixed at 0 or 1 at the output node Y, with negative x_1 (x_2) corresponding to logic input $I_1(I_2)$ being 0 and positive x_1 (x_2) corresponding to logic input I_1 (I_2) being 1. So in order to successfully act like an invertible logic gate, the network of p-bits must display *both* behaviors consistently: in the direct operational mode, when the

TABLE II. Necessary and sufficient conditions, derived from the logic truth tables given in Table I, to be satisfied simultaneously by the output node of the network of p-bits given by Eq. (2), in order to implement the OR, AND, NOR, and NAND logic operations. The biases C_1 and C_2 of the input nodes A and B are fixed by the two logic inputs I_1 and I_2. In our representative example, we consider $C_1(C_2) = -1$ to correspond to logic input $I_1(I_2) = 0$ and $C_1(C_2) = 1$ to correspond to $I_1(I_2) = 1$ in Eq. (2).

Logic inputs (I_1,I_2)	Logic operation	Logic output	Necessary and sufficient condition for the state of output node Y
(0, 0)	OR	0	$x_3(t) < 0$
(0, 1)/(1, 0)	OR	1	$x_3(t) > 0$
(1, 1)	OR	1	$x_3(t) > 0$
(0, 0)	AND	0	$x_3(t) < 0$
(0, 1)/(1, 0)	AND	0	$x_3(t) < 0$
(1, 1)	AND	1	$x_3(t) > 0$
(0, 0)	NOR	1	$x_3(t) > 0$
(0, 1)/(1, 0)	NOR	0	$x_3(t) < 0$
(1, 1)	NOR	0	$x_3(t) < 0$
(0, 0)	NAND	1	$x_3(t) > 0$
(0 ,1)/(1, 0)	NAND	1	$x_3(t) > 0$
(1, 1)	NAND	0	$x_3(t) < 0$

inputs are fixed (i.e., C_1 and C_2 are fixed by the two logic inputs), the state of the output node should yield the correct AND, OR, NAND, or NOR logical output. Additionally, in the reverse mode when the output is fixed (i.e., C_3 is fixed by the logic output), the states of the input nodes A and B should fluctuate among all possible logical inputs that are consistent with the given logical output.

We first present results obtained via numerical simulations of the p-bit network in response to a stream of inputs, for the realization of a direct (forward) logic gate operation, in the presence of moderate noise. The time waveforms of the system variable $x_3(t)$ obtained from Eq. (2), for increasing noise strength, are depicted in Fig. 4. For the test runs, the logic inputs 0 and 1 are switched randomly to yield different combinations of input sets $(0, 0)$, $(0, 1)$, $(1, 0)$, and $(1, 1)$ in random order. Figure 4 shows the responses of the system to this varying input stream, for fixed bias C_3 in the output node Y, for different noise strengths D. For low noise level $D = 0.3$ [Fig. 4(b)], the system clearly cannot consistently produce the transitions necessary to yield the correct output with high probability, and occasionally gets trapped in a well and does not yield the desired transition in response to the input signal. As the noise level is increased, in an optimal band of noise level, the system switches in a consistent manner yielding a reliable NAND logic response. Figure 4(c) shows the representative case of $D = 1.0$ in this optimal band. Further increasing the noise level, for instance $D = 1.8$ shown in Fig. 4(d), leads to the increased occurrence of random switching, destroying the reliability of the response.

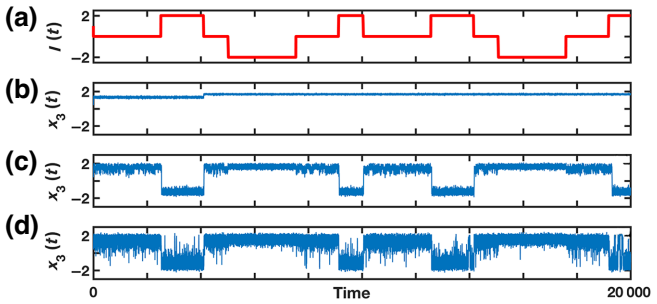

FIG. 4. Timing waveforms for the direct (forward) mode of NAND logic operation, obtained from numerical simulation of the network of p-bits given by Eq. (2), with $C_1(t)$ and $C_2(t)$ encoding the two logic inputs, and C_1 and C_2 taking the value -1.0 when the logic input is 0 and the value $+1.0$ when the logic input is 1. The output logic response is extracted from the state $x_3(t)$ of the output node Y. Here $C_3 = 1.0$ and $k = -1.0$. Panel (a) displays the signal $I(t) = C_1(t) + C_2(t)$. Panels (b)–(d) show the responses for increasing noise strengths (top to bottom): (b) $D = 0.3$, (c) $D = 1.0$, and (d) $D = 1.8$. For low noise level (b) the system cannot consistently produce the transitions necessary to yield the correct output. For moderate noise level (c) the systems switches in a consistent manner, yielding a reliable NAND logic response. For high noise level (d) increased random switching destroys the reliability of the response.

So this figure clearly illustrates that the state of the output node yields robust responses consistent with different logic operations, as detailed in Tables I–III, only for a moderate noise strength. This phenomenon of the input signals causing appropriate transitions between potential wells aided by the noise floor, is reminiscent of the LSR phenomenon [9–16].

Further, the time waveforms of the system variable $x_3(t)$ thus obtained for a fixed value of noise strength by Eq. (2), for realizing different direct logic operations, are depicted in Fig. 5. Figure 5 depicts the responses of the system to this varying input stream, for different biases C_3 in the output node Y and different connection strengths k, clearly demonstrating that the state of the output node yields clean and robust responses consistent with different

TABLE III. Illustrative values of external bias C_3 for the output node Y, for implementation of different directed (forward) logical operations (cf. Tables I and II). Bias terms of the input nodes A and B (i.e., C_1 and C_2, respectively), are fixed by the two logic inputs I_1 and I_2, with logic input 0 corresponding to a bias of -1 and logic input 1 corresponding to a bias of $+1$ in Eq. (2). Here noise strength $D = 1$.

C_3	k	Logic	Type of operation
1.0	1.0	OR	directed
−1.0	1.0	AND	directed
−1.0	−1.0	NOR	directed
1.0	−1.0	NAND	Directed

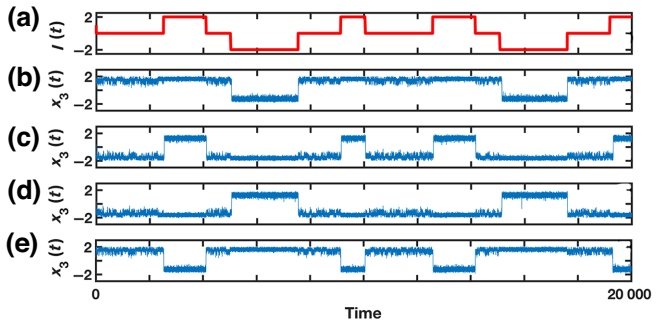

FIG. 5. Timing waveforms for the direct (forward) mode of operation, obtained from numerical simulation of the network of p-bits given by Eq. (2), with $I(t) = C_1(t) + C_2(t)$ encoding the sum of the two logic inputs shown in red in (a), for different logic responses obtained from the state $x_3(t)$ of the output node Y: (b) OR gate ($C_3 = 1.0, k = 1.0$), (c) AND gate ($C_3 = -1.0, k = 1.0$), (d), NOR gate ($C_3 = -1.0, k = -1.0$) and (e) NAND gate ($C_3 = 1.0, k = -1.0$). Here noise strength $D = 1.0$.

logic operations, as detailed in Tables I–II. For instance, for the direct mode of OR and AND logic [with $k = 1$ in Eq. (2)], when both logic inputs are 1, the biases of input nodes A and B are clamped to $C_1 = +1$ and $C_2 = +1$ (i.e., $C_1 + C_2 = +2$). This drives the state of output node Y always to be positive (i.e., $x_3(t) > 0$), corresponding to logic output 1. Denoting the combination of the two logic inputs and logic output, obtained from the two input nodes A and B and the output node Y, by [ABY], we obtain [ABY] = [111]. This satisfies the truth table of OR and AND logic. When both logic inputs are 0, both C_1 and C_2 are clamped to -1 (i.e., $C_1 + C_2 = -2$), leading to the state of the output node Y always being negative ($x_3(t) < 0$), corresponding to logic output 0. So [ABY] = [000] which again satisfies the truth table of OR and AND logic. Lastly, when the logic inputs are $(0, 1)$ or $(1, 0)$, either C_1 or C_2 is -1 and the other $+1$ (i.e., $C_1 + C_2 = 0$). Now the output node Y yields different responses for OR and AND logic operations, as determined by the value of bias C_3 given in Table III. For OR logic, x_3 is always positive, corresponding to logic output 1, and so [ABY] = [011]/[101], which satisfies the truth table of OR logic [cf. Fig. 5(b)]. When the bias C_3 corresponds to that of AND logic operations, this input set yields $x_3(t) < 0$, corresponding to logic output 0. So [ABY] = [010]/[100], which satisfies the truth table of AND logic [cf. Fig. 5(c)]. When the sign of the coupling k is switched, one obtains complementary logic responses. For instance, with $k = -1$ in Eq. (2), when both logic inputs are 1, the state of output node Y is now always negative, and when both logic inputs are 0, we find that the state of output node Y is always positive. So [ABY] is [001] and [110], which is consistent with NOR and NAND logic. For the input sets $(0, 1)/(1, 0)$ one obtains different responses depending on the sign of the bias C_3 at the output node Y. As given in Table III, when $C_3 = -1.0$, one obtains [100]/[010], consistent with NOR logic [cf. Fig. 5(d)], and when $C_3 = 1.0$ one obtains [011]/[101] consistent with NAND logic [cf. Fig. 5(e)].

For the inverted mode of operation we fix the bias of the output node, and the states of the two input nodes A and B must now reflect the different input sets that give rise to the appropriate output. For instance, for the inverted AND logic, we obtain the following behavior: if we encode logic output 0 by clamping the bias of the output node C_3 to -2, the states (x_1, x_2) of the input nodes A and B fluctuate between $(0, 0)$, $(0, 1)$, and $(1, 0)$, with $x_i > 0$ corresponding to 1 and $x_i < 0$ corresponding to 0. This is consistent with the truth table of AND logic gate. So the emergent combinations [ABY] that are most probable are [000], [010], and [100]. If we encode logic output 1 by clamping the bias of the output node C_3 to 2, the states (x_1, x_2) of the input nodes A and B are now always positive, corresponding to the logic input set $(1, 1)$. This is again consistent with the truth table of AND logic. So the most probable emergent combinations [ABY] are [000],[010],[100], and [111], yielding an inverted AND logic gate. Similarly, varying biases C_1, C_2 and the connection strength k (as detailed in Table IV) yields the other inverted logic operations. The success of the p-bit network in reverse mode, giving inverted OR, inverted AND, inverted NOR and inverted NAND, is also clearly evident from the time series $x_1(t)$ and $x_2(t)$ of the input nodes A and B, in response to different output states, shown in Fig. 6. Figures 7 and 8 present the numerically estimated probabilities of [ABY] for direct and inverted logic operations, providing a quantitative measure for the success of the different forward and reverse logic operations.

Additionally, we again find that noise aids the operation of the inverted operation, as it did the direct operation. Figure 9 shows the reverse mode for the illustrative example of NAND logic for three levels of noise. For low noise, as displayed in Fig. 9(a), just one of the input sets corresponding to the logic output emerges. On the other hand, for high noise displayed in Fig. 9(c), the system undergoes increasingly random hopping between states, resulting in a large number of instances of the states at the input nodes

TABLE IV. Illustrative values of external bias C_1 and C_2 for the input nodes A and B, for implementation of different inverted (reverse) logical operations (cf. Table I) with the network represented by Eq. (2). The bias of the output node is fixed by the logic output, with $C_3 = 2$ when logic output is 1 and $C_3 = -2$ when logic output is 0. Here noise strength $D = 1$.

C_1	C_2	k	Logic	Type of operation
-0.8	-0.8	1.0	OR	inverted
0.8	0.8	1.0	AND	inverted
-0.8	-0.8	-1.0	NOR	inverted
0.8	0.8	-1.0	NAND	Inverted

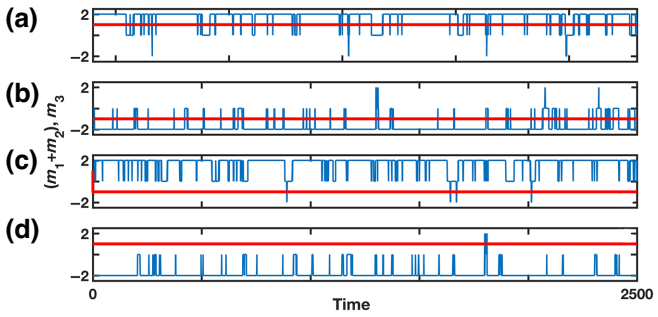

FIG. 6. Timing waveforms for the inverted (reverse) mode of operation, obtained from numerical simulation of the network of p-bits given by Eq. (2), with clamped output $m_3(t)$ (red) and $m_1(t) + m_2(t)$ (blue) obtained from the states $x_1(t)$ and $x_2(t)$ of the input nodes A and B, with $m_i(t) = \mathrm{sgn}(x_i(t))$ $(i = 1, 2)$, for different logic operations: (a) inverted OR gate ($C_1 = C_2 = -0.8$ and $k = 1.0$); (b) inverted AND gate ($C_1 = C_2 = 0.8$ and $k = 1.0$); (c) inverted NOR gate ($C_1 = C_2 = -0.8$ and $k = -1.0$); (d) inverted NAND gate ($C_1 = C_2 = 0.8$, and $k = -1.0$). Here noise strength $D = 1.0$.

A and B being inconsistent with NAND logic. However, for moderate noise the emergent states at input nodes A and B clearly yield all the input sets consistent with NAND logic, as reflected by $m_1 + m_2$ in Fig. 9(b).

Analysis of the inverted logic operation. Given two random events defined on the same probability space, the joint probability provides a statistical measure of the likelihood of these two events occurring together, thus yielding an understanding of all possible pairs of outputs. So in the context of stochastic switching in our network of p-bits, the joint probability is the probability of the state of one input node A being in the positive well at the same time that the other input node B is also in the positive well. Now the joint probability of these independent events is calculated as the probability of the event at input node A multiplied by the probability of the event at input node B, formally stated as $P(AB) = P(A) \times P(B)$. Now in the

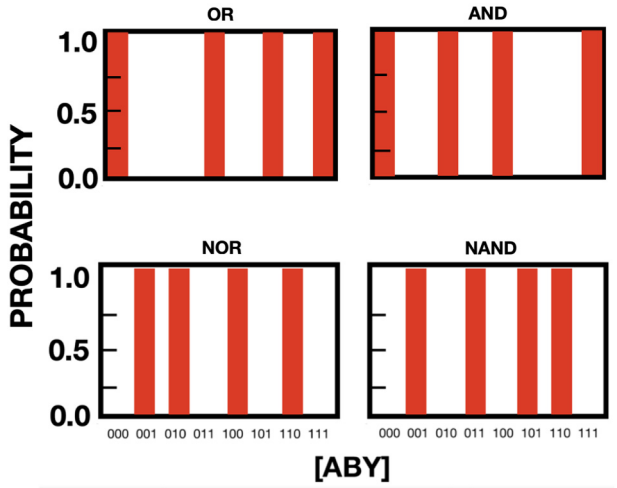

FIG. 7. Histograms showing the probability of occurrence of different combinations of the logic inputs and logic output, denoted by [ABY], where A and B denote the two logic inputs and Y denotes the logic output, for various direct logic operations. Here noise intensity $D = 1$, and the bias values at the nodes of the p-bit network, for the different logic functions, are as given in Table III.

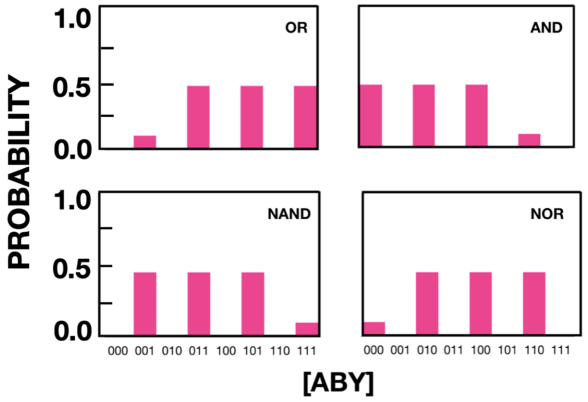

FIG. 8. Histograms showing the probability of occurrence of different combinations of the logic inputs and logic output, denoted by [ABY], where A and B denote the two logic inputs and Y denotes the logic output, for various inverted logic operations. Here noise intensity $D = 1$, and the bias values at the nodes of the p-bit network, for the different logic functions, are as given in Table IV.

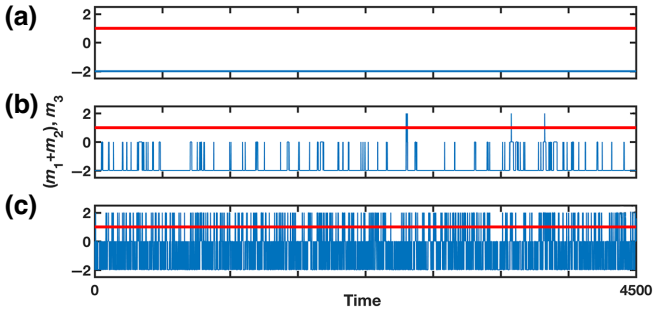

FIG. 9. Timing waveforms for the inverted (reverse) mode of operation, obtained from numerical simulation of the network of p-bits given by Eq. (2), with $m_3(t)$ clamped at logic output 1 (red) and $m_1(t) + m_2(t)$ (blue) obtained from the states $x_1(t)$ and $x_2(t)$ of the input nodes A and B, with $m_i(t) = \mathrm{sgn}(x_i(t))$ $(i = 1, 2)$, for inverted NAND gate ($C_1 = C_2 = 0.8$, and $k = -1.0$). Panels (a)–(c) show the responses for increasing noise strengths: (a) $D = 0.3$, (b) $D = 1.0$ and (c) $D = 1.8$. Clearly, only for moderate noise does one consistently obtain input states in accordance with NAND logic, as reflected by $m_1 + m_2$ in (b). In contrast, for low noise (a) only one of the input sets consistent with the logic output is obtained, and for high noise (c) one gets a large number of instances of input sets inconsistent with the logic output, thus degrading the reliability of the inverted logic gate operation.

inverse operational mode, the output node with bias fixed at the appropriate value C_3 yields m_3 that reflects the logic output consistently. So the individual probabilities of obtaining $m = 1$ corresponding to logic input 1 at an input node A and B are simply given by the probability distribution $P(1)$ displayed in Fig. 2 for the appropriate effective bias $km_3 + C_1$ and $km_3 + C_2$ at the nodes A and B. Similar arguments give the probability $P(0)$ of obtaining $m = -1$ corresponding to logic input 0 at an input node A and B as $1 - P(1)$. So the inverted operation is governed by the joint probabilities $P(00)$, $P(01) = P(10)$, and $P(11)$ obtained from the product of the individual probabilities $P(1)$ and $P(0)$ at the input nodes. We present the explicit analysis for some representative values from Tables III and IV.

For $C_1 = C_2 = -0.8$ and $k = 1$, we have the effective bias C equal to $1 - 0.8 = 0.2$ for $m_3 = 1$ corresponding to logic output 1, and $-1 - 0.8 = -1.8$ for $m_3 = -1$ corresponding to logic output 0. The probability of being in the positive well $P(1) \sim 2/3$ and $P(0) \sim 1/3$ for the former case (i.e., for logic output 1) and $P(1) \sim 0$ and $P(0) \sim 1$ for the latter (i.e., for logic output 0). This yields the joint probabilities for the input states corresponding to logic output 1 of $P(00) \sim 1/3 \times 1/3 = 1/9$, $P(11) \sim 2/3 \times 2/3 = 4/9$, and the probability of the state of either input node being 1 while the other is 0 is approximately $2 \times (2/3 \times 1/3) = 4/9$. For the input states corresponding to logic output 0, $P(1) \sim 0$ and so the joint probabilities are $P(00) \sim 1$, $P(11) \sim 0$, and the probability of either input node being 1 while the other is 0 is also approximately 0. So within reasonable probability, this response of the p-bit network is consistent with the truth table of OR logic, thus yielding an inverted OR logic operation, as it probabilistically gives the correct input sets that give rise to the different OR logic outputs. Similar analysis can easily be carried out to rationalize the success of the inverted logic operations of AND, NOR, and NAND logic.

Direct logic operations can also be rationalized through the probabilities $P(1)$ and $P(0)$. The effective bias of the output node for the forward mode is $k(m_1 + m_2) + C_3$. As an illustrative example, consider the OR operation for noise strength around $D \sim 1$, with $C_3 = 1$ and $k = 1$. For logic inputs $(0, 0)$, the appropriate C_1 and C_2 yield $m_1 + m_2 = -2$. So the effective bias is -1, where $P(1) \sim 0$ and $P(0) \sim 1$, and one obtains output 0 with probability close to 1. For logic input sets $(0, 1)$ or $(1, 0)$, $m_1 + m_2 = 0$, and so the effective bias is 1, which yields $P(1) \sim 1$ and $P(0) \sim 0$. So one obtains output 1 with probability close to 1. For logic inputs $(1, 1)$, $m_1 + m_2 = 2$, and so the effective bias is 3, which gives $P(1) \sim 1$ and $P(0) \sim 0$. So one obtains output 1 with probability close to 1. Thus one obtains a robust OR logic operation. Similar arguments can easily account for successful AND, NOR, and NAND operations for the values of C_3 and k listed in Table III. Note that these arguments are similar to those used for a single bistable system, where it was shown that the residence times in the two wells are such that the bit-averaged output signal becomes the desired one for sufficiently strong noise [42]. The additional aspect that is crucially important here is that we use the bias term and input signal as an effective "lever" to push the coupled system to the desired state by manipulating the symmetry of the wells of the constituent nodes of the network. So by exploiting this controlled "lever action" at the nodal dynamics we are able to correctly reflect the desired logic function.

III. PROOF-OF-PRINCIPLE EXPERIMENTAL VERIFICATION

In this section we verify the idea delineated above in electronic circuit analogues of the nonlinear system described by Eq. (2), and ascertain its robustness in actual circuit experiments. The schematic of the circuit realization is shown in Fig. 10. This circuit has two building blocks, namely, weight logic and p-bits. The weight logic circuits are constructed with op-amp summing amplifiers and an inverting amplifier. The p-bits are realized with a circuit capable of displaying stochastic resonance [44] along with a comparator and an inverting amplifier to produce the signals $m_i(t)$ and $-m_i(t)$, respectively. The interconnections are made as shown in Fig. 10.

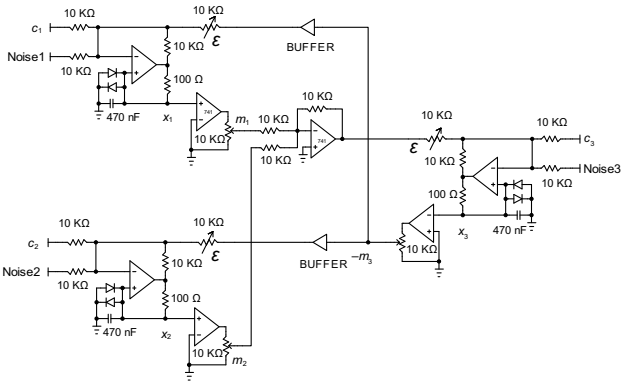

FIG. 10. Analog circuit diagram corresponding to Eq. (2) for implementing direct and reverse logic operations. Here the operational amplifiers are μA 741 or AD712, and the diodes are IN4148 or IN4007. The response state variables $x_1(t)$, $x_2(t)$, and $x_3(t)$ at the nodes A, B, and Y of the network, are obtained from the respective capacitor voltages. The power supply to the op-amps is fixed at +9 V–0 V–9 V. The weight logic portion consists of op-amp summing amplifiers and sign changers. The p-bit is realized with a circuit capable of displaying stochastic resonance [44]. The binary output signals $m_i(t)$ are generated from the op-amp comparators. The power supply to op-amps and the bias voltages C_i are drawn from Agilent or Keysight E3631A dc power supply. The input signals C_1, C_2, C_3 and the noise signal are drawn from an Agilent or Keysight 33522A function/arbitrary waveform generator.

Here we use a piecewise-linear circuit that can be constructed efficiently with very few circuit components [45], with $F(x)$ specifically given by the following piecewise-linear function:

$$\begin{aligned} F(x) &= x && \text{when } -0.5 \le x \le 0.5 \\ F(x) &= -(x+1) && \text{when } x < -0.5 \\ F(x) &= -(x-1) && \text{when } x > 0.5. \end{aligned} \tag{3}$$

In Fig. 10, C_1, C_2, and C_3 are constant- bias signals. The output voltages of the three capacitors correspond to the state variables $x_1(t), x_2(t)$, and $x_3(t)$ of Eq. (2), respectively. In the circuit, op-amps are realized with AD712 or μA741. The noise signal was drawn from an Agilent or Keysight 33522A function/arbitrary waveform generator. All oscilloscope trails were obtained using an Agilent or Keysight DSOX2012A. The power supply to op-amps and the bias voltages were drawn from an Agilent or Keysight E3631A dc power supply.

Representative experimental results for direct AND logic gate operations with the noise amplitude fixed at $D = 1.5$ V, and for the circuit parameters chosen in Fig. 10, are displayed in Fig. 11. A comparison with Fig. 5 clearly shows that the same phenomenon is observed in these circuit experiments; that is, with suitable bias at the nodes, in an optimal window of noise, we get the desired direct AND logic gate operation reliably.

Representative experimental results for inverted AND logic gate operations are displayed in Figs. 12 and 13. It is clearly evident from comparison with Fig. 6 that the same phenomenon is observed in these circuit experiments; that is, for suitable bias at the nodes, we get the desired inverted AND logic gate operation reliably in an optimal window of noise.

Though we have presented demonstrations of our idea with the specific bistable system given in Eq. (2), we can also obtain all these direct and invertible logic operations in a similar fashion, in the presence of a noise floor, using

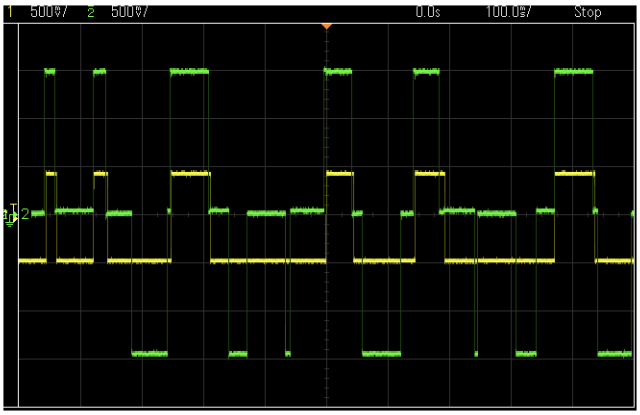

FIG. 11. Experimental observation of direct AND logic operation through the analog circuit simulation of Eq. (2). Timing waveforms: $m_3(t) = \mathrm{sgn}(x_3(t))$ (yellow) and $C_1 + C_2$ (green). Here C_1 and C_2 are two input signals, which take the value -750 mV when the logic input is 0 and the value $+750$ mV when the logic input is 1. The bias value $C_3 = -350$ mV and the noise amplitude $D = 1.5$ V. The noise signal is drawn from an Agilent or Keysight 33522A function/arbitrary waveform generator. The oscilloscope used is an Agilent or Keysight DSOX2021A. The power supply to op-amps and the bias voltages are drawn from an Agilent or Keysight E3631A dc power supply. The scale of the traces are 100 ms/Div (X axis) and 500 mV/Div (Y axis).

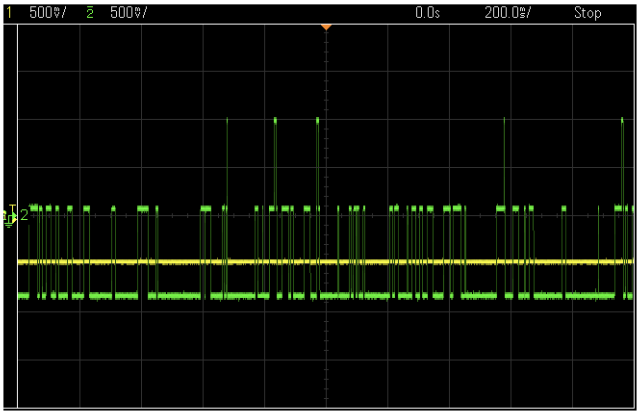

FIG. 12. Experimental results displaying the inverted operation mode for AND logic; oscilloscope trace of $m_1(t) + m_2(t)$ (green) obtained from the two input nodes, while the signal $m_3(t)$ (yellow) from the output node always stays at the rescaled value of -500 mV when C_3 is clamped to -700 mV (i.e., logic output 0). Note that the logic inputs given by m_1 and m_2 fluctuate between (00), (01), and (10), consistent with the truth table of AND logic gate with almost equal probability and the combination (11) occurs with very low probability.

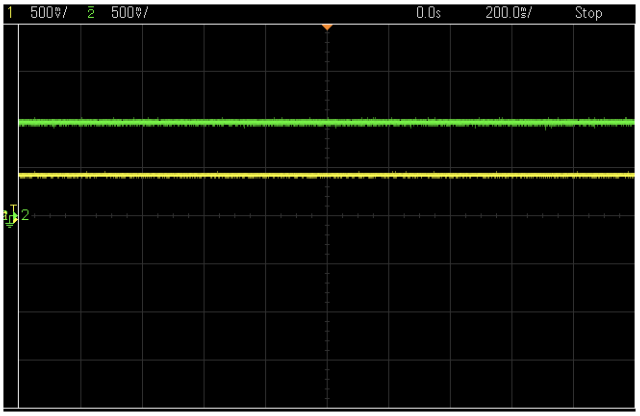

FIG. 13. Experimental results displaying the inverted operation mode for AND logic; Oscilloscope trace of $m_1(t) + m_2(t)$ (green) obtained from the two input nodes, while the signal $m_3(t)$ (yellow) from the output node always stays at the rescaled value of $+500$ mV when C_3 is clamped to $+700$ mV (i.e., logic output 1). Note that the logic inputs given by m_1 and m_2 now lock to the combination (11), consistent with the truth table of AND logic gate, with very high probability.

any bistable system, including a simple Schmitt trigger as the basic bistable unit.

IV. CONCLUSIONS

We have proposed a network of interconnected nonlinear systems to serve as probabilistic p-bits in the presence of a noise floor, to implement the powerful unconventional paradigm of invertible logic that has found applications in important critical problems such as integer factorization and machine learning. Such p-bits represent stochastic units that can be interconnected to implement Boolean logic functions with the capacity to carry out bidirectional operations and can operate in both direct and inverted modes, unlike standard binary logic which is only capable of unidirectional operations. They may be considered to be intermediate between logic bits used in conventional digital electronics and the q-bits used in quantum computing, and are robust classical entities fluctuating in time between 0 and 1, with the ability to interact with other p-bits.

Specifically, we have demonstrated the successful implementation of direct and invertible logic gates, with different logic functionalities OR, AND, NOR, and NAND, using noise-aided attractor hopping in our network of p-bits. Importantly, the probability of switching between attractors can be controlled in a clean and reliable manner by a tunable bias, and so the design of direct and inverted logic is based on finding the correct set of biases for the nodes of the network. The bistable nature of our p-bits allows us to map the states to binary inputs and output. We demonstrate that simply setting suitable bias values and network interconnection strengths allows us to flexibly reconfigure our network to yield both directed and inverted OR, AND, NOR, and NAND logic operations. Interestingly, furthermore, the operational reliability is optimal in a window of moderate noise, reminiscent of stochastic resonance. This concept was verified through numerical simulations, as well as electronic circuit experiments.

The ideas presented here can be realized in systems with multistable chaotic attractors as well, with the states tipped to the desired chaotic attractor by bias values that encode logic inputs for direct operations and logic output for the inverted mode. In further open research directions, more complex dynamics-based networks with high scalability can be used for implementing large-scale factorizations, training of neural networks, and other important computational tasks that are crucially based on backward operations.

In summary, this concept of noise-aided invertible logic from networks of multistable nonlinear systems has the potential to be realized in wide-ranging systems, and suggests a promising direction in exploiting noise-assisted attractor hopping in networks to design bidirectional computational devices.

ACKNOWLEDGMENTS

K.M. acknowledges support from the DST-FIST Programme (Grant No. SR/FST/PSI-200/2015(C)). S.S. acknowledges support from the J.C. Bose National Fellowship (Grant No. JBR/2020/000004).

[1] Kerem Yunus Camsari, Rafatul Faria, Brian M. Sutton, and Supriyo Datta, Stochastic P-Bits for Invertible Logic, Phys. Rev. X **7**, 031014 (2017).

[2] Rafatul Faria, Kerem Yunus Camsari, and Supriyo Datta, Low-barrier nanomagnets as p-bits for spin logic, IEEE Magn. Lett. **8**, 1 (2017).

[3] Ahmed Zeeshan Pervaiz, Lakshmi Anirudh Ghantasala, Kerem Yunus Camsari, and Supriyo Datta, Hardware emulation of stochastic p-bits for invertible logic, Sci. Rep. **7**, 10994 (2017).

[4] Kerem Yunus Camsari, Sayeef Salahuddin, and Supriyo Datta, Implementing p-bits with embedded mtj, IEEE Electron Device Lett. **38**, 1767 (2017).

[5] Naoya Onizawa and Takahiro Hanyu, Cmos invertible logic: Bidirectional operation based on the probabilistic device model and stochastic computing, IEEE Nanotechnol. Mag. **16**, 33 (2021).

[6] Naoya Onizawa, Makoto Kato, Hitoshi Yamagata, Koji Yano, Seiichi Shin, Hiroyuki Fujita, and Takahiro Hanyu, Sparse random signals for fast convergence on invertible logic, IEEE Access **9**, 62890 (2021).

[7] Naoya Onizawa, Akira Tamakoshi, and Takahiro Hanyu, Hardware acceleration of large-scale cmos invertible logic based on sparse hamiltonian matrices, IEEE Open J. Circuits Syst. **2**, 782 (2021).

[8] Josef-Stefan Wenzler, Tyler Dunn, Tommaso Toffoli, and Pritiraj Mohanty, A nanomechanical fredkin gate, Nano Lett. **14**, 89 (2014).

[9] K. Murali, Sudeshna Sinha, William L. Ditto, and Adi R. Bulsara, Reliable Logic Circuit Elements that Exploit Nonlinearity in the Presence of a Noise Floor, Phys. Rev. Lett. **102**, 104101 (2009).

[10] Adi R. Bulsara, Anna Dari, William L. Ditto, K. Murali, and Sudeshna Sinha, Logical stochastic resonance, Chem. Phys. **375**, 424 (2010).

[11] Yong Xu, Xiaoqin Jin, Huiqing Zhang, and Tingting Yang, The availability of logical operation induced by dichotomous noise for a nonlinear bistable system, J. Stat. Phys. **152**, 753 (2013).

[12] Nan Wang and Aiguo Song, Enhanced logical stochastic resonance in synthetic genetic networks, IEEE Trans. Neural Netw. Learn. Syst. **27**, 2736 (2015).

[13] Rong Gui, Huiyu Zhang, Guanghui Cheng, and Yuangen Yao, Set–reset latch logic operation in a bistable system under suprathreshold and subthreshold signals, Chaos: An Interdiscip. J. Nonlinear Sci. **30**, 023119 (2020).

[14] Vivek Kohar and Sudeshna Sinha, Noise-assisted morphing of memory and logic function, Phys. Lett. A **376**, 957 (2012).

[15] Vivek Kohar, Krishnamurthy Murali, and Sudeshna Sinha, Enhanced logical stochastic resonance under periodic

forcing, Commun. Nonlinear Sci. Numer. Simul. **19**, 2866 (2014).
[16] K. Murali, W. L. Ditto, and Sudeshna Sinha, Reconfigurable Noise-Assisted Logic Gates Exploiting Nonlinear Transformation of Input Signals, Phys. Rev. Appl. **18**, 014061 (2022).
[17] Luca Gammaitoni, Peter Hänggi, Peter Jung, and Fabio Marchesoni, Stochastic resonance, Rev. Mod. Phys. **70**, 223 (1998).
[18] Shanmuganathan Rajasekar and Miguel A. F. Sanjuan, *Nonlinear Resonances* (Springer, Basel, 2016).
[19] V. Sorokin and I. Demidov, On representing noise by deterministic excitations for interpreting the stochastic resonance phenomenon, Philos. Trans. R. Soc. A **379**, 20200229 (2021).
[20] Chen Yang, Jianhua Yang, Dengji Zhou, Shuai Zhang, and Grzegorz Litak, Adaptive stochastic resonance in bistable system driven by noisy NLFM signal: Phenomenon and application, Philos. Trans. R. Soc. A **379**, 20200239 (2021).
[21] K. Murali, I. Rajamohamed, Sudeshna Sinha, William L. Ditto, and Adi R. Bulsara, Realization of reliable and flexible logic gates using noisy nonlinear circuits, Appl. Phys. Lett. **95**, 194102 (2009).
[22] J. Zamora-Munt and C. Masoller, Numerical implementation of a VCSEL-based stochastic logic gate via polarization bistability, Opt. Express **18**, 16418 (2010).
[23] P. Pfeffer, F. Hartmann, S. Höfling, M. Kamp, and L. Worschech, Logical Stochastic Resonance with a Coulomb-Coupled Quantum-Dot Rectifier, Phys. Rev. Appl. **4**, 014011 (2015).
[24] Diego N. Guerra, Adi R. Bulsara, William L. Ditto, Sudeshna Sinha, K. Murali, and P. Mohanty, A noise-assisted reprogrammable nanomechanical logic gate, Nano Lett. **10**, 1168 (2010).
[25] L. Worschech, F. Hartmann, T. Y. Kim, S. Höfling, M. Kamp, A. Forchel, Jouni Ahopelto, I. Neri, A. Dari, and L. Gammaitoni, Universal and reconfigurable logic gates in a compact three-terminal resonant tunneling diode, Appl. Phys. Lett. **96**, 042112 (2010).
[26] Kamal P. Singh and Sudeshna Sinha, Enhancement of "logical" responses by noise in a bistable optical system, Phys. Rev. E **83**, 046219 (2011).
[27] Sudeshna Sinha, J. M. Cruz, T. Buhse, and P. Parmananda, Exploiting the effect of noise on a chemical system to obtain logic gates, Europhys. Lett. **86**, 60003 (2009).
[28] Hiroyasu Ando, Sudeshna Sinha, Remo Storni, and Kazuyuki Aihara, Synthetic gene networks as potential flexible parallel logic gates, Europhys. Lett. **93**, 50001 (2011).
[29] Anna Dari, Behnam Kia, Adi R. Bulsara, and William Ditto, Creating morphable logic gates using logical stochastic resonance in an engineered gene network, Europhys. Lett. **93**, 18001 (2011).
[30] Edward H. Hellen, Syamal K. Dana, Jürgen Kurths, Elizabeth Kehler, and Sudeshna Sinha, Noise-aided logic in an electronic analog of synthetic genetic networks, PLoS ONE **8**, e76032 (2013).
[31] A. Sharma, V. Kohar, M. Shrimali, and S. Sinha, Realizing logic gates with time-delayed synthetic genetic networks, Nonlinear Dyn. **76**, 431 (2014).
[32] Juan Wu, Yong Xu, Haiyan Wang, and Jürgen Kurths, Information-based measures for logical stochastic resonance in a synthetic gene network under Lévy flight superdiffusion, Chaos: An Interdiscip. J. Nonlinear Sci. **27**, 063105 (2017).
[33] K. Murali, Sudeshna Sinha, Vivek Kohar, Behnam Kia, and William L. Ditto, Chaotic attractor hopping yields logic operations, PLoS ONE **13**, e0209037 (2018).
[34] M. Sathish Aravindh, A. Venkatesan, and M. Lakshmanan, Strange nonchaotic attractors for computation, Phys. Rev. E **97**, 052212 (2018).
[35] M. Sathish Aravindh, A. Venkatesan, and M. Lakshmanan, Route to logical strange nonchaotic attractors with single periodic force and noise, Chaos: An Interdiscip. J. Nonlinear Sci. **30**, 093137 (2020).
[36] V. Manaoj Aravind, K. Murali, and Sudeshna Sinha, in *Nonlinear Dynamics and Control: Proceedings of the First International Nonlinear Dynamics Conference (NODYCON 2019), Volume II* (Springer International Publishing, 2020), p. 325.
[37] Manaoj Aravind, K. Murali, and Sudeshna Sinha, Coupling induced logical stochastic resonance, Phys. Lett. A **382**, 1581 (2018).
[38] Manaoj Aravind, P. Parmananda, and Sudeshna Sinha, Emergent noise-aided logic through synchronization, Phys. Rev. E **104**, 064207 (2021).
[39] Lei Zhang, Aiguo Song, and Jun He, Effect of colored noise on logical stochastic resonance in bistable dynamics, Phys. Rev. E **82**, 051106 (2010).
[40] Huiqing Zhang, Yong Xu, Wei Xu, and Xiuchun Li, Logical stochastic resonance in triple-well potential systems driven by colored noise, Chaos: An Interdiscip. J. Nonlinear Sci. **22**, 043130 (2012).
[41] G. Cheng, W. Liu, R. Gui, and Y. Yao, Sine-Wiener bounded noise-induced logical stochastic resonance in a two-well potential system, Chaos, Solitons Fractals **131**, 109514 (2020).
[42] S. Barbay, G. Giacomelli, and F. Marin, Experimental Evidence of Binary Aperiodic Stochastic Resonance, Phys. Rev. Lett. **85**, 4652 (2000).
[43] M. Morris Mano, *Computer System Architecture* (Prentice Hall, Englewood, Cliffs, NJ, 1993).
[44] Muthusamy Lakshmanan and Kallummal Murali, *Chaos in Nonlinear Oscillators: Controlling and Synchronization*, Vol. 13 (World scientific, Singapore, 1996).
[45] Elena Tamaševičius, Arūnas Vytautas Tamaševičius, Gytis Mykolaitis, Skaidra Bumelienė, and v Lindberg, Analogue electrical circuit for simulation of the duffing-holmes equation, Nonlinear Anal.: Modell. Control **13**, 241 (2008).