Physics Letters A 376 (2012) 957–962

Contents lists available at SciVerse ScienceDirect

Physics Letters A

www.elsevier.com/locate/pla

Noise-assisted morphing of memory and logic function

Vivek Kohar, Sudeshna Sinha *

Indian Institute of Science Education and Research (IISER) Mohali, Sector 81, SAS Nagar, Mohali 140 306, India

ARTICLE INFO

Article history:
Received 4 October 2011
Received in revised form 22 December 2011
Accepted 24 January 2012
Available online 28 January 2012
Communicated by A.R. Bishop

Keywords:
Stochastic resonance
Memory
Logic gates
Nonlinear circuits

ABSTRACT

We demonstrate how noise allows a bistable system to behave as a memory device, as well as a logic gate. Namely, in some optimal range of noise, the system can operate flexibly, both as a NAND/AND gate and a Set–Reset latch, by varying an asymmetrizing bias. Thus we show how this system implements memory, even for sub-threshold input signals, using noise constructively to store information. This can lead to the development of reconfigurable devices, that can switch efficiently between memory tasks and logic operations.

© 2012 Elsevier B.V. All rights reserved.

1. Introduction

While the number of transistors in an integrated circuit has approximately doubled every year in accordance with Moore's law, the rapid shrinking of computing platforms with smaller power supplies has brought with it problems of smaller noise margins and higher error rates. So wide ranging research efforts in recent years have focused on the issue of reliable operations in the presence of a noise floor [1].

In this context, recently it has been shown that a noisy nonlinear system, when driven by two square waves encoding two logical inputs, consistently goes to a state that mirrors a logical combination of the two inputs (such as AND/NAND and OR/NOR logic, cf. Table 1) in some optimal range of noise [2]. That is, the probability of getting the correct logical response increases to unity with increase in the intensity of noise and then decreases again when noise exceeds the optimal range. Further one can vary the threshold (or bias) and morph the output into different logical functions. This concept, named "Logical Stochastic Resonance" [2–9], helps one gain understanding of the counter-intuitive interplay between noise and nonlinearity [10]. Further, from the applied viewpoint, this idea can potentially lead to the design of flexible logic gates with enhanced performance in noisy environments [2–9].

Here we examine the possibility of utilizing such a noisy nonlinear system, not just as a logic gate, but also *directly as a memory device*, i.e., we explore if the system can behave as a latch in some optimal range of noise.

A latch is a system that has two stable states and can be used to store state information. The system can be made to change state by signals applied to one or more control inputs, as shown in the truth table (Table 2). Latches are a fundamental building block of a computing machine, and is omnipresent in computers and communication systems. So proposals of implementations of a latch, that are more efficient from the point of view of space or operational time, have far-reaching consequences.

Now latches can be built around a pair of cross-coupled inverting elements, such as vacuum tubes, bipolar transistors, field effect transistors, inverters, and inverting logic gates. Specifically, for instance, conventional latches can be constructed out of two cross coupled NAND or alternately NOR gates. In present computing systems including embedded systems and other small processing units, memory access and modification time, is a significant bottleneck limiting the speed of computation.

Here, unlike traditional latches built by concatenating two logic elements, we will use only *one* element to implement the latch truth table. Our proposal does not necessitate cross-coupling logic gates, nor does it involve many clock cycles. So the direct realization of the latch here has the potential to save both space and time costs.

Specifically we will demonstrate how we can produce the Set–Reset latch operation consistently in an optimal window of noise. Namely we will show, that when the noisy bistable system is presented low amplitude input signals, consisting of two aperiodic pulses encoding two logic inputs, the output will consistently mirror a latch output (as displayed in Table 2). We also show how one can use a bias to get different types of responses from the same system, thereby obtaining an element that is easily *reconfigurable* to yield, not only gates, but a memory device as well. That is, in

* Corresponding author. Tel./fax: +91 172 2240266.
E-mail address: sudeshna@iisermohali.ac.in (S. Sinha).

0375-9601/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.physleta.2012.01.039

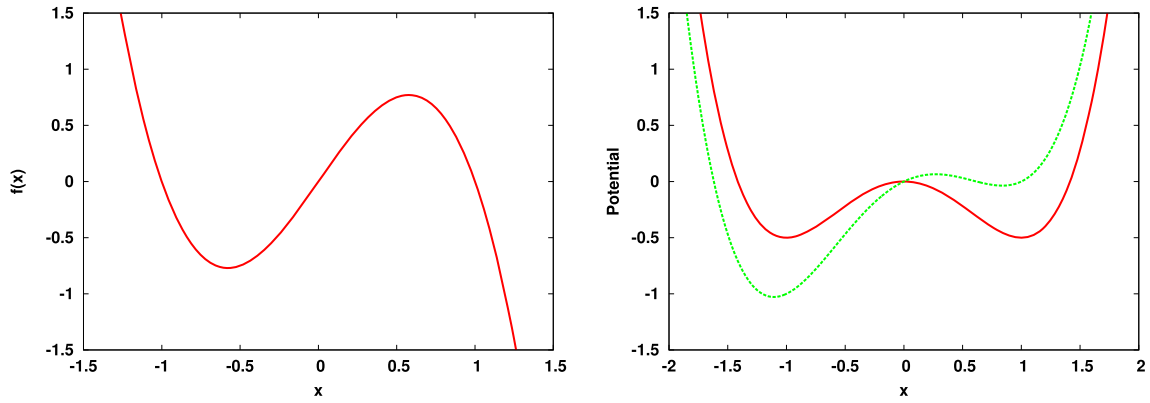

Fig. 1. For the system described by Eq. (2): (left) the function $F(x) = 2x - 2x^3$ with $b = 0$ and (right) the effective potential obtained by integrating the function $F(x)$ in Eq. (2), with bias $b = 0$ (red solid line) and $b = -0.5$ (dashed green line). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this Letter.)

an optimal range of noise, by varying the bias, the *same* system will yield logic functions like AND, OR, etc. as well as give latch operations directly.

2. General principle

Consider the general nonlinear system,

$$\dot{x} = F(x) + b + I_{in} + D\eta(t) \tag{1}$$

where $F(x)$ is a generic nonlinear function obtained via the negative gradient of a potential with two distinct stable wells. I_{in} is the input signal which encodes the logic inputs, b is bias to asymmetrize the two potential wells, and $\eta(t)$ is an additive zero-mean Gaussian noise with unit variance, with D being the amplitude (intensity) of the noise.

A logical input–output can be obtained by driving the system with two trains of aperiodic square pulses: $I_1 + I_2$, encoding the two logic inputs. For logic operations such as OR/NOR and AND/NAND, we consider the inputs to take value I when the logic input is 1, and value $-I$ when the logic input is 0, where input strength I is: $0 < I < 1$. Since, the binary logic inputs can be either 0 or 1, they produce 4 sets of binary input: $(0, 0)$, $(0, 1)$, $(1, 0)$, $(1, 1)$. These four input conditions give rise to three distinct values of I_{in}. Hence, the input signal I_{in} generated, is a 3-level aperiodic wave form.

The state of this system, can be interpreted as logic output 1 when $x > x^*$ and logic output 0 when $x < x^*$, where x^* is roughly given by the position of the barrier between the two wells. Such an interpretation allows one to consistently obtain logical responses, such as OR and AND, when noise intensity D is in an optimal band. Complementary gates, NOR and NAND, can also be obtained in a straight-forward manner, by the alternate output determination: $x < x^*$ corresponding to logic output 1, and 0 otherwise.

Now, to use this element as a Set–Reset latch, we need to modify the encoding of input values, so that we can distinguish between $(0, 1)$ and $(1, 0)$ states, as the latch truth table is *asymmetric* with respect to inputs unlike the usual logic gates. A simple way to accomplish this is to have the following asymmetric input encoding: the first input I_1 takes the value $-I$ when the logic input is 0 and I when the logic input is 1, while the second input I_2 takes the value I when the logic input is 0 and $-I$ when the logic input is 1, where $0 < I < 1$.

Equivalently, instead of the asymmetric input association described above, we can consider symmetric input associations (as in the logic operations), and apply a NOT operation to the second input I_2. This will also yield the same physical input signal I_{in}. Namely, corresponding to the 4 sets of binary inputs (I_1, I_2): $(0, 0)$, $(0, 1)$, $(1, 0)$, $(1, 1)$, the input signal I_{in} takes the values 0, -1, 1, and 0 respectively. Out of these four sets, the input corresponding to $(1, 1)$ is a restricted set and does not occur in the truth table. So we are left with three input sets, each one giving rise to a *distinct* value of I_{in}. Hence the input signal I_{in} generated, is again a 3-level aperiodic wave form.

Table 1
Relationship between the two logic inputs and the output of the fundamental OR, AND, NOR and NAND logic operations. The four distinct possible input sets $(0, 0)$, $(0, 1)$, $(1, 0)$ and $(1, 1)$ reduce to three conditions, as $(0, 1)$ and $(1, 0)$ are symmetric. Note that *any* logical circuit can be constructed by combining the NOR (or the NAND) gates [11].

Logic inputs	OR	AND	NOR	NAND
0, 0	0	0	1	1
0, 1	1	0	0	1
1, 0	1	0	0	1
1, 1	1	1	0	0

Table 2
Relationship between the two inputs and the output of Set–Reset latch.

Set (I_1)	Reset (I_2)	Latch
0	0	No change (maintain the previous state)
0	1	0
1	0	1
1	1	Restricted set

Logic response from output can be obtained, as in logic operations, by defining a threshold value x^*. If $x > x^*$, i.e., when the system is in the potential well x_+, then the logic output is taken to be 1, and 0, if $x < x^*$ and the system is in other well. Thus, the logic output toggles as the state of the system switches from one well to another.

3. Explicit example

We now explicitly demonstrate latch functionality, using a simple nonlinear system:

$$\dot{x} = 2x - 2x^3 + b + I_{in} + D\eta(t) \tag{2}$$

where D is amplitude of the Gaussian noise, b is asymmetrizing bias and the potential energy function is bistable (see Fig. 1). The input signal, $I_{in} = I_1 + I_2$, where I_1 and I_2 encode the two logic inputs, with the encoding associations for the logic operations and the Set–Reset latch being different. The bias b, for different operations, is set as displayed in Table 3.

Also note that the nonlinear function above, is efficiently realized by a linear resistor, linear capacitor, and a small number of

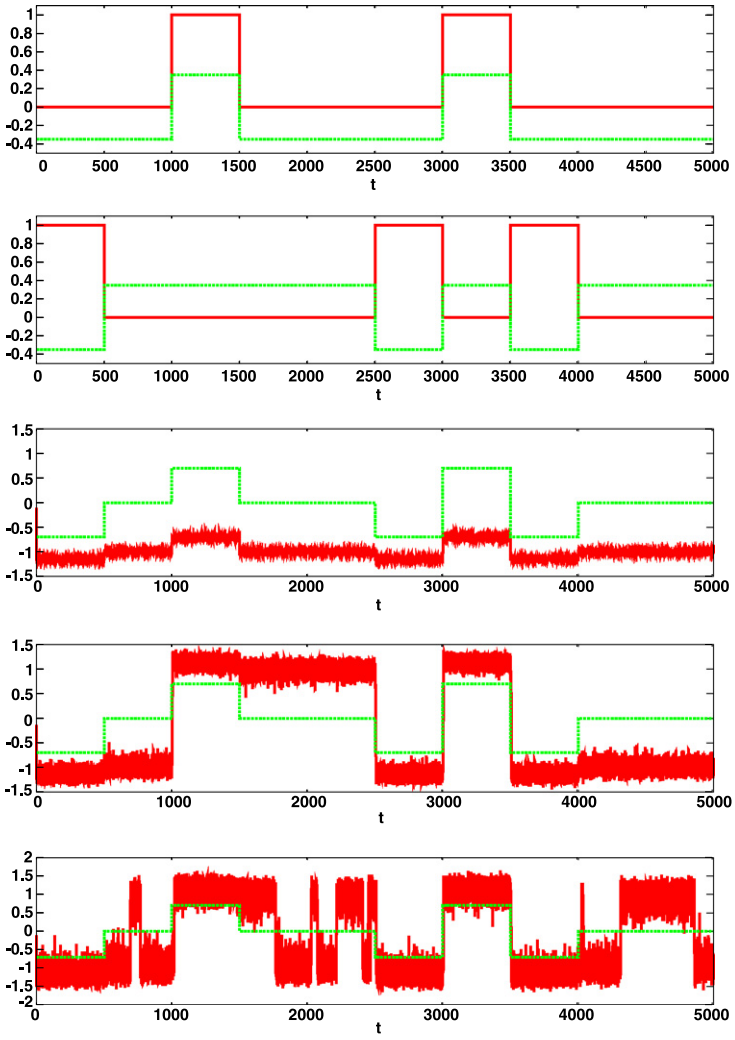

Fig. 2. Panels 1 and 2 show streams of the two inputs. The input strength is 0.35. The logical value of the input is shown as a solid red line and the actual value as a dashed green line. For I_1 we have -0.35 when logic input is 0 and 0.35 when logic input is 1. For I_2 we have 0.35 when logic input is 0 and -0.35 when logic input is 1. Panels 3–5 show the outputs $x(t)$ corresponding to $D = 0.05$, $D = 0.25$, $D = 0.5$.[2] Here bias $b = 0$, and the input signal $I_{in} = I_1 + I_2$ is indicated by the green line. Clearly, we get the desired output only when noise is within some optimal range (in this case panel 4 i.e. $D = 0.25$). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this Letter.)

CMOS transistors [3]. Further, it is capable of operating in very high frequency regimes, and such a system may be implemented with integrated circuits and nanoelectronic devices. Other forms of the nonlinear function $F(x)$ in Eq. (1) may be realized in optical [8], nanomechanical [4], chemical [5] and biological systems [9].

Specifically, input pulses are given to the system in such a way that *all* possible combination of transitions are presented to the system, in *random sequence* (see the top 3 panels of Fig. 2). The threshold for output determination, x^*, is 0 here. So to obtain the desired logic response, we interpret the state $x > 0$ as logic output 1 and $x < 0$ as the logic output 0. It is clearly evident from Fig. 2, that for moderate noise the system consistently yields the Set–Reset latch input–output association, while it fails to do so for very small noise and large noise.

Note that the system holds its output steady, while the input signal is held constant, over long times ($\sim 10^3$ in our simulations). Namely, the system manages to maintain its state, which is in a local equilibrium of the potential, for reasonably small noise intensity D and large barrier height ΔV. This is because the Kramer's rate for escaping from the potential well (i.e., the inverse of the average switching rate induced by noise alone), given by $\sim \exp(-\Delta V/D)$, is very small here. So the system does not switch wells under the influence of noise alone. Rather it needs both signal and noise to effect a change. However, it is evident (for instance from Fig. 2) that as noise increases, the probability of random noise-induced well hopping increases, leading to loss of robustness for noise levels beyond the optimal window.

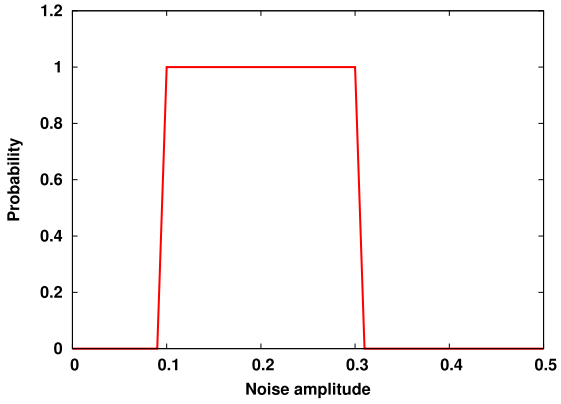

Fig. 3. Probability for obtaining the Set–Reset latch operation for different values of noise strength, with bias $b = 0$. Evidently we get the Set–Reset latch operation only within an optimal window of noise strength.

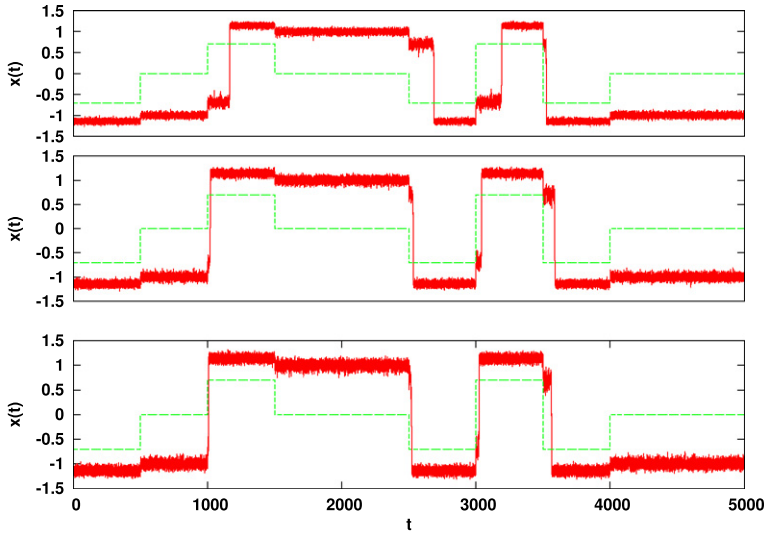

Fig. 4. Panels 1–3 display the output x corresponding to $D = 0.10$, $D = 0.12$ and $D = 0.15$ (from top to bottom). The stream of inputs $I_{in} = I_1 + I_2$, where input strength $I = 0.35$, is indicated by the green line in the figures. Clearly panel 3 has the shortest transients after input switches. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this Letter.)

We can quantify the consistency (or reliability) of obtaining a given logic output by calculating the probability of obtaining the desired logic output for different sets of inputs, namely the ratio of successful runs where the desired logic output is obtained (after transience), to the total number of runs. Since we want the logic response to be obtained for any random combination of inputs, any probability which is less than 1 is of no use to us. Thus, we further define $P(logic)$ to be 1 if it is equal to 1 (when the logic operation is completely obtained for all the given input sets), and 0 if it is anything less than 1.

The system was simulated by keeping the value of one input set constant over 1000 time steps, and we simulated the system over a sequence of 500 such sets. Unless otherwise stated, we allow a latency of 100 in the calculation of $P(logic)$. The values of inputs I_1 and I_2 were chosen to be randomly distributed so that $I_{in} = I_1 + I_2$ switched levels in an uncorrelated aperiodic manner. Thus the probability is one only if we get the desired response for all the 500 input sets.

It is evident from Fig. 3 that we obtain consistent memory operation in an optimal window of noise. So it is clear that the system behaves as a Set–Reset latch even for sub-threshold input signals, utilizing noise to store the information. This can lead to development of low power consuming memory devices.

We also observed the *reduction of latency* with increasing noise. This is evident in Fig. 4. Clearly, the system responds faster to inputs when noise intensity is higher. That is, the desired hopping between wells happens more rapidly under the influence of stronger noise. This is yet another feature where noise aids performance (see Fig. 5).

Further, it is evident from Fig. 6 that this system can also yield the logic response of a AND/NAND gate and a OR/NOR gate, by changing the bias b in Eq. (2). This suggests that within an optimal window of noise strength, we can morph the circuit to act either as a logic gate or as a memory device, by simply adjusting the value of bias, i.e., one can easily switch from Set–Reset latch operation to AND/NAND or OR/NOR logic operation (see Table 3). This is made possible by the change in the symmetry and depths of the potential wells due to changing bias b.

Note that the noise region of optimal operation depends on the form of the nonlinear function $F(x)$ in Eq. (1), as well as on the input signal strength, as displayed in Figs. 7–8. In order to obtain an overlapping window of optimal noise strength for the different operations, one can adjust the input strength appropriately or choose a nonlinear function that yields the desired operational window for the given noise floor.

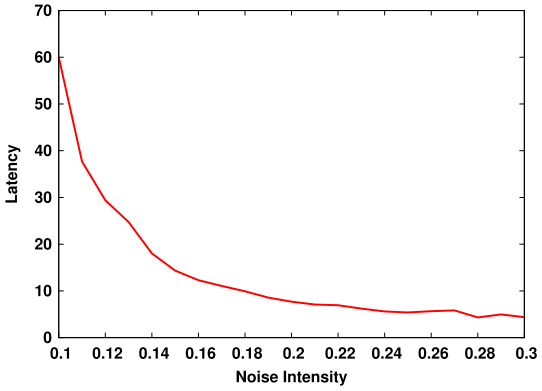

Fig. 5. Latency (averaged over a random stream of inputs) as a function of noise strength. Here latency has been defined as the time taken to reach the barrier from a well, when the input switches necessitating a change in the output.

Table 3
Representative values of the asymmetrizing bias b in Eq. (2) that yields the Set–Reset latch operation, OR/NOR logic operation and AND/NAND logic operation. Here $I = 0.5$.

Operation	Bias
Set–Reset latch	0
AND/NAND	−0.5
OR/NOR	0.5

4. Discussions

In this work, we have explicitly shown that a nonlinear system functioning in a noisy environment, can produce a completely consistent Set–Reset latch operation on two inputs, streaming in any random sequence. We observe that for very small or very large noise strengths the system does not yield a reliable output. However, in a reasonably wide band of moderate noise strength, the system produces the desired output very consistently. Furthermore, the response of the system can be easily switched from memory

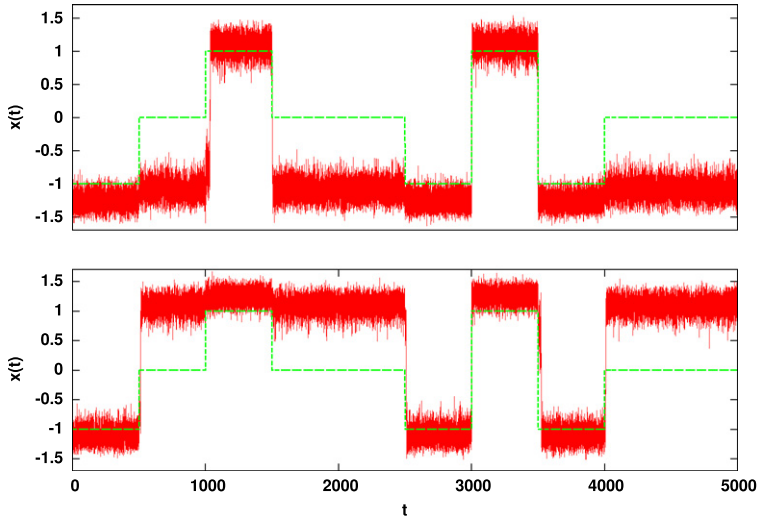

Fig. 6. Panels showing outputs $x(t)$ corresponding to bias $b = -0.5$ (top panel) and $b = 0.5$ (bottom panel), with the green line indicating the input signal $I_{in} = I_1 + I_2$ in these panels. Clearly, by adjusting the bias value we obtain logic output corresponding to AND gate (top) and OR gate (bottom). Here $I = 0.5$ and $D = 0.38$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this Letter.)

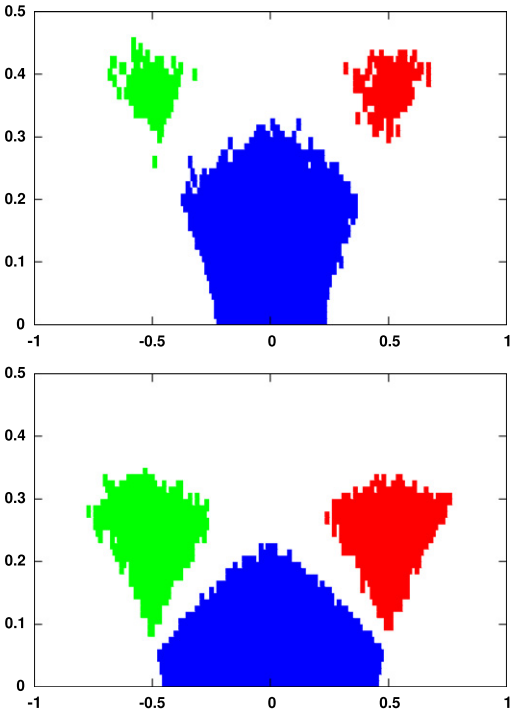

Fig. 7. Points marking where the Set–Reset latch operation (blue), OR/NOR logic (red) and AND/NAND logic (green) are obtained with probability 1, for nonlinear function $F(x) = 2x - 2x^3$ (top) and $F(x) = 2x - 4x^3$ (bottom), with input signal strength 0.5, for different values of bias (x axis) and noise strength (y axis). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this Letter.)

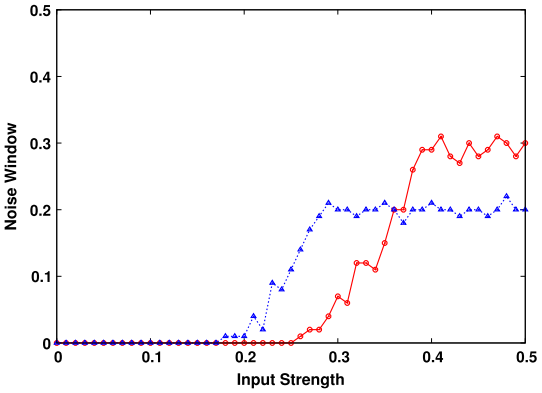

Fig. 8. Size of the noise window of optimal Set–Reset latch operation for nonlinear function $F(x) = 2x - 2x^3$ (red) and $F(x) = 2x - 4x^3$ (blue), for different values of input signal. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this Letter.)

to logic operations by varying the bias in the system. Lastly, it was also observed that noise reduced latency in the response of the system to switched inputs.

So it is evident that "LSR Elements" can reliably function as memory devices even for sub-threshold signals, thus consuming very low power. Further, the same elements can be used to produce outputs corresponding to, not only latches, but logic gates as well. Thus these "LSR Elements" can potentially act as building blocks of futuristic "Smart Computing Devices".

Potentially, such devices will not only operate robustly in noisy environments, but can also be capable of optimal utilization of their resources by configuring their "LSR Elements" into latches, or any of the logic gates, depending on the requirements of the task being performed. For example, if we are performing tasks requiring more computational power like running a code, then these computing devices will morph most of the LSR elements to logic gates, whereas in case of tasks requiring memory like plotting large values of data, LSR elements will be morphed into memory enabling efficient use of resources. Furthermore, it is conceivable that devices based on such elements can potentially help in reducing boot times thus achieving what is commonly called "instant boot". This can be accomplished by morphing large number of LSR elements into memory at the time of shut down and start up. This significant increase in memory will enable us to keep most of the data required for the applications readily accessible, paving way for faster boot times.

In conclusion, we have demonstrated how noise allows a bistable system to behave as a memory device, as well as a logic gate. This can lead to the development of reconfigurable devices, that can switch efficiently between memory tasks and logic operations.

References

[1] S.A. Ibanez, P.I. Fierens, R.P.J. Perazzo, G.A. Patterson, D.F. Grosz, Eur. Phys. J. B 76 (2010);
P.I. Fierens, S.A. Ibanez, R.P.J. Perazzo, G.A. Patterson, Phys. Lett. A 374 (2010).
[2] K. Murali, S. Sinha, W.L. Ditto, A.R. Bulsara, Phys. Rev. Lett. 102 (2009) 104101.
[3] K. Murali, I. Raja Mohamed, S. Sinha, W.L. Ditto, A.R. Bulsara, Appl. Phys. Lett. 95 (2009) 194102.
[4] D.N. Guerra, A.R. Bulsara, W.L. Ditto, S. Sinha, K. Murali, P. Mohanty, Nano Lett. 10 (2010) 1168.
[5] S. Sinha, J.M. Cruz, T. Buhse, P. Parmananda, Europhys. Lett. 86 (2009) 60003.
[6] L. Worschech, et al., Appl. Phys. Lett. 96 (2010) 042112.
[7] J. Zamora-Munt, C. Masoller, Optics Express 18 (2010) 16418.
[8] K.P. Singh, S. Sinha, Phys. Rev. E 83 (2011) 046219.
[9] H. Ando, S. Sinha, R. Storni, K. Aihara, Europhys. Lett. 93 (2011) 50001;
A. Dari, B. Kia, A.R. Bulsara, W.L. Ditto, Europhys. Lett. 93 (2011) 18001.
[10] See e.g. L. Gammaitoni, P. Hanggi, P. Jung, F. Marchesoni, Rev. Mod. Phys. 70 (1998) 223;
A.R. Bulsara, L. Gammaitoni, Physics Today 49 (1996) 39;
K. Wiesenfeld, F. Moss, Nature 373 (1995) 33.
[11] M.M. Mano, Computer System Architecture, Prentice Hall, 1993;
T.C. Bartee, Computer Architecture and Logic Design, McGraw–Hill, 1991.