Realization of the fundamental NOR gate using a chaotic circuit

K. Murali*
Department of Physics, Anna University, Chennai 600 025, India

Sudeshna Sinha†
The Institute of Mathematical Sciences, Taramani, Chennai 600 113, India

William L. Ditto‡
Department of Biomedical Engineering, University of Florida, Gainesville, Florida 32611-6131, USA
and Control Dynamics Inc., 1662 101st Place SE, Bellevue, Washington 98004, USA

(Received 21 January 2003; published 11 July 2003)

We report the experimental verification of a simple threshold controller, which clips the chaos to periods of widely ranging orders, in a chaotic circuit. Then we use this to implement the fundamental NOR gate thus obtaining a proof of principle experiment demonstrating the universal computing capability of chaotic systems.

DOI: 10.1103/PhysRevE.68.016205 PACS number(s): 05.45.−a

Recently, there has been a new theoretical direction in harnessing the richness of chaos, namely, the exploitation of chaos to do flexible computations [1]. A system is capable of universal general purpose computing if it can emulate a NOR or NAND gate since any logic gate can be obtained by an adequate connection of NOR/NAND gate (i.e., any Boolean circuit can be built using a NOR (NAND) gate) [2].

In this paper, we will first discuss the experimental verification of a threshold controller that clips the chaos into different temporal patterns and serves as a basis for a scheme of computing with chaos [1]. Then, we will go on to give the circuit implementation of the fundamental NOR gate.

I. CLIPPING THE CHAOS TO DIFFERENT PERIODS BY THRESHOLDING

Consider a general N-dimensional dynamical system, described by the evolution equation $\dot{\mathbf{x}}=F(\mathbf{x})$, where $\mathbf{x}\equiv(x_1,x_2,\ldots x_N)$ are the state variables and variable x_i is chosen to be monitored and threshold controlled. Prescription for the threshold control in this system is as follows: control will be triggered whenever the value of the monitored variable exceeds a critical threshold x^* (i.e., when $x_i>x^*$) and the variable x_i is reset to x^* [3]. The dynamics continues till the next occurrence of x_i exceeding the threshold, when control resets its value to x^* again. So, this controller does not alter the original systems dynamics in any way, as there is no perturbation on the parameters. Further, no run-time knowledge of $F(\mathbf{x})$ is involved, and no computation is needed to obtain the necessary control.

The theoretical basis of the method lies in clipping desired time sequences (symbol sequences in maps) and enforcing a periodicity on the sequence through the thresholding action, which acts as a resetting of initial conditions. Chaos is advantageous here, as it possess a rich range of temporal patterns that can be clipped to different behaviors. This immense variety is not available from thresholding regular systems. For an instance, it can be shown analytically for one-dimensional (1D) maps that the threshold mechanism yields superstable orbits of all orders by simply varying the threshold level [3]. Further, in 1D maps, the system is trapped in the regular cycle the instant it exceeds threshold.

Now, we will experimentally demonstrate the success of this simple controller in a circuit realization of the well-known logistic map

$$x_{n+1}=4ax_n(1-x_n). \tag{1}$$

The circuit realization of the above discrete map is depicted in Fig. 1(a). In the circuit implementation, the notations x_{n-1}, x_n, and x_{n+1} denote voltages normalized by 10 V as the unit. Hereafter, unless otherwise stated, the variables x_{n-1}, x_n, x_{n+1}, and x^* are normalized by 10 V as the unit for experimental results. An analog multiplier integrated circuit IC (analog devices AD633) is used as a squarer and it produces the output voltage of $x_n^2/10$ V for the given x_n as the input. By utilizing the suitable inverting amplifier, inverting summing amplifier, and a sign-changer realized with opamps OA1, OA2, and OA3, respectively, the voltage proportional to $4x_n(1-x_n)$ will be available at the output of OA3. A variable resistor (VR1) is used to control the parameter a in Eq. 1. The value of a can be varied from 0 to 1. The output voltage of OA3 becomes a new input voltage to the multiplier AD633 after passing through a sample-and-hold circuit (SH1), a threshold controller circuit, and another sample-and-hold circuit (SH2). The sample-and-hold circuits are implemented with LF 398 or ADG412 IC's and these are triggered by timing pulses T1 and T2 as shown in Fig. 1(b). The timing pulses are being generated from the clock generator providing a delay of feedback. The delay is essential for obtaining the solution of Eq. 1. Here, a clock rate of either 5 or 10 kHz is used.

A precision clipping circuit [4] as depicted in Fig. 1(c) is employed for threshold control when the terminals are con-

*Electronic address: kmurali@annauniv.edu
†Electronic address: sudeshna@imac.ernet.in
‡Electronic address: william.ditto@bme.ufi.edu

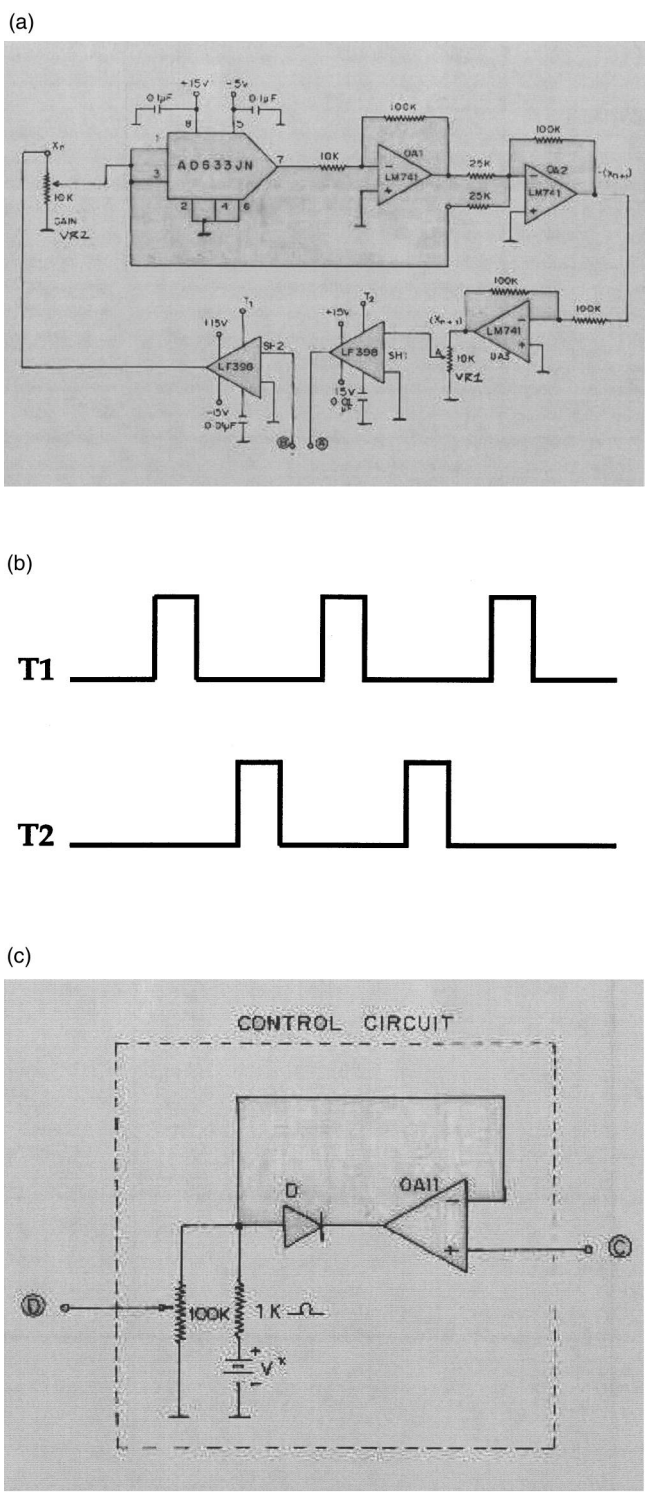

FIG. 1. Circuit for solving Eq. 1: (a) logistic map module, (b) timing pulses, and (c) threshold control module. See text for details.

nected as *A* to *C* and *D* to *B*. We have chosen component values for the control circuit to be the following: op-amp $=\mu A741$, diode=IN4148 or IN34A, series resistor=1 kΩ, and threshold control voltage=V^*, which sets the x^*. A delayed signal from SH1 is being monitored by the threshold control circuitry. The threshold controlled voltage is again

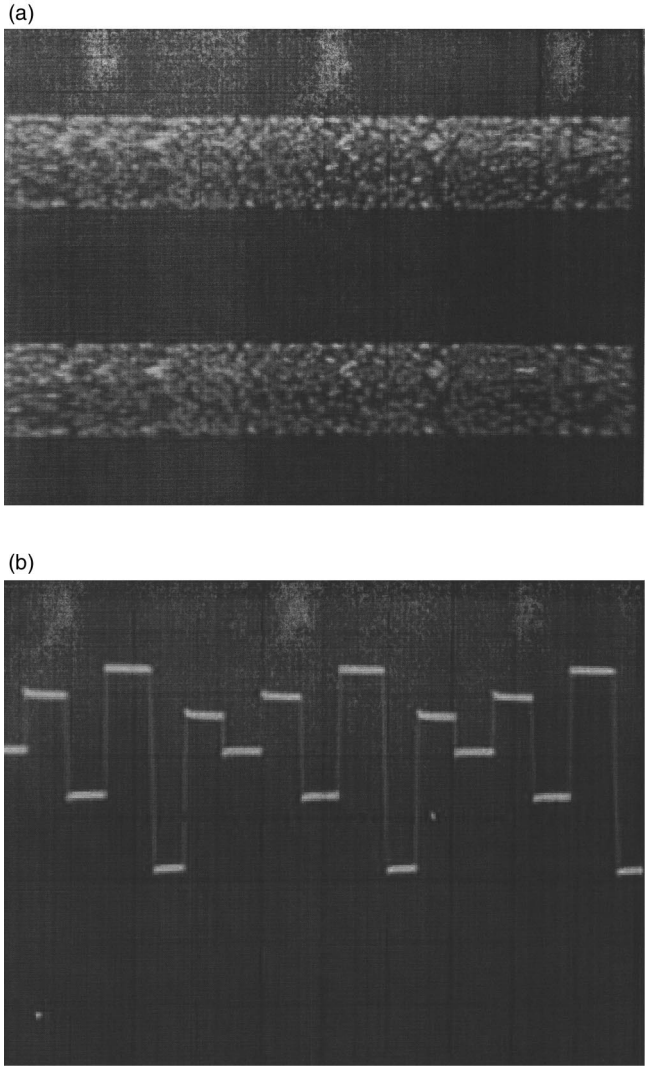

FIG. 2. (a) Trace of the (uncontrolled) chaotic attractor in the time domain. The ordinate is x_n produced by the circuit in Fig. 1 and the abscissa is the discrete time *n*, with time/div=1 msec. (b) Trace of x_n generated from the control circuit vs discrete time *n* for a controlled period 6 cycle with time/div=0.1 msec.

sampled by SH2 and fed as the input of AD633 through a suitable gain control resistor (VR2).

Figure 2(a) displays the uncontrolled chaotic attractor, and Figs. 2(b) and 3 show some representative results of the chaotic system under different threshold values x^*. It is clear that thresholding manages to yield cycles of varying periodicities. Also, note that simply setting the threshold beyond the bounds of the attractor gives back the original dynamics, and so the controller is easily switched on and off (see Table I).

A detailed comparison shows a *complete agreement between experimental observations and analytical results* [1]. For instance, the threshold which needs to be set in order to obtain a certain periodicity and the trajectory of the controlled orbit can be worked out *exactly* through symbolic dynamics techniques [1]. Further, the control transience is very short here (typically of the order of 10^{-3} times the

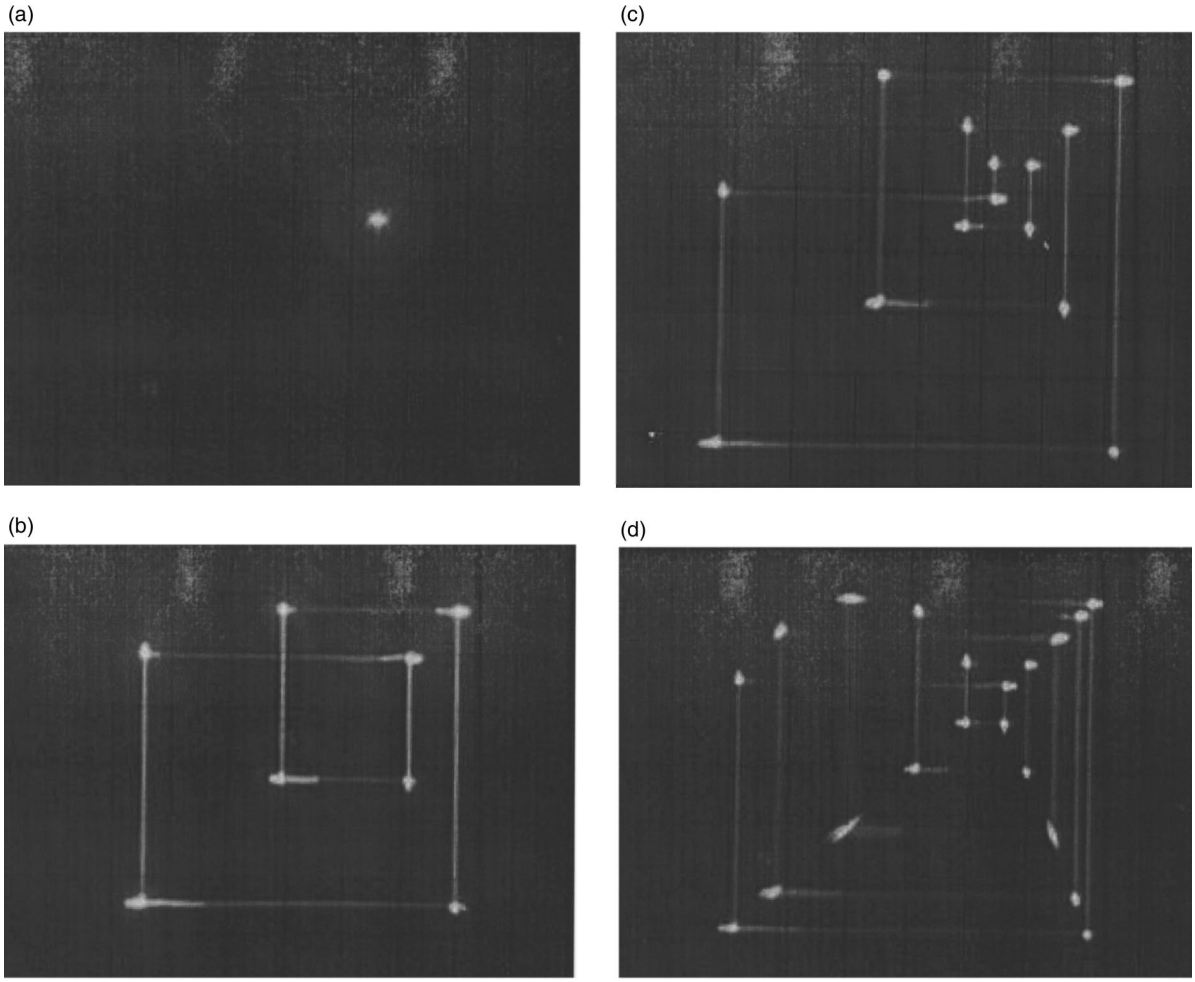

FIG. 3. Typical controlled attractors effected by setting different thresholds V^*. Ordinate and abscissa represent traces of x_{n+1} and x_n for (a) period 1 cycle with $V^*=5$ V; (b) period 4 cycle with $V^*=9.1$ V; (c) period 7 cycle with $V^*=9.355$ V; and (d) period 10 cycle with $V^*=9.5$ V.

controlled cycle) and the perturbations involved in threshold control are usually small.

This method is then especially useful in the situation where one wishes to *design* controllable components that can switch flexibly between different behaviors. Calibrating the system characteristics at the outset with respect to threshold gives one a look-up table to directly and simply extract widely varying temporal patterns. Thus, this scheme has a considerable potential for use in applications to chaos computing [1], communications [5], pattern formation, control systems, and signal processing. Also, the simplicity of the controller implies low complexity costs, which is important in technical applications seeking to exploit the richness of chaos in a direct and efficient way.

Now, in the following section, we will explicitly show one particular application: the realization of the fundamental NOR gate. This constitutes a proof of principle experiment demonstrating the universal computing capability of chaotic systems.

TABLE I. Threshold ranges vs periodicity of the controlled cycle for the chaotic logistic map circuit. Note that cycles of the same period, but different geometries, can be obtained in different threshold windows.

Range of threshold (in units of 10 V)	Nature of controlled orbit
$x^*<0.75$	Period 1 (fixed point)
$0.75<x^*<0.905$	Period 2 cycle
$x^*\sim 0.965$	Period 3 cycle
$0.905<x^*<0.925$	Period 4 cycle
$x^*\sim 0.979$	Period 5 cycle
$x^*\sim 0.93$	Period 6 cycle
$x^*\sim 0.9355$	Period 7 cycle
$x^*\sim 0.932$	Period 8 cycle
$x^*\sim 0.981$	Period 9 cycle
$x^*\sim 0.95$	Period 10 cycle

II. SCHEME FOR OBTAINING THE FUNDAMENTAL NOR GATE WITH A CHAOTIC SYSTEM

Consider the above chaotic logistic map circuit in the range of threshold values $x^*<0.75$, namely, in the fixed point regime, where the state of the system is always natu-

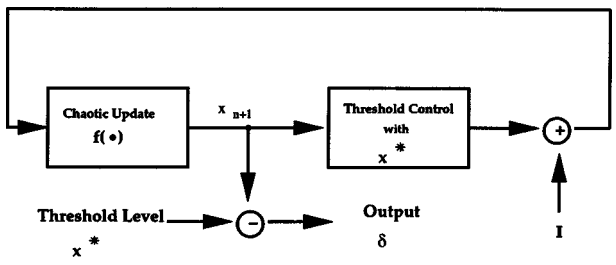

FIG. 4. Schematic circuit for implementing the NOR gate.

rally at x^* [Fig. 3(a)]. The basic NOR logic operation on a pair of inputs (I_1, I_2) involves the following steps.

(1) Inputs: $x \rightarrow x^* + X_1 + X_2$, where $X_{1,2} = 0$ when $I_{1,2} = 0$ and $X_{1,2} = \delta$ when $I_{1,2} = 1$. So, the state of the system is equal to x^* when input set is (0,0), $x^* + \delta$ when input set is (0,1) or (1,0), and $x^* + 2\delta$ when input set is (1,1).

(2) Chaotic update, i.e., $x \rightarrow f(x)$, where $f(x)$ is a chaotic function (specifically the logistic map here).

(3) Threshold mechanism to obtain output Z is

$$Z = 0 \quad \text{if} \quad f(x) \leqslant x^*,$$

$$Z = f(x) - x^* \quad \text{if} \quad f(x) > x,$$

where x^* is the threshold. This is interpreted as the logic output 0 if $Z = 0$ and logic output 1 if $Z = \delta$.

Note that, in our implementation, we demand that the *input and output have equivalent definitions* (i.e., 1 unit is the same quantity for input and output). So, constant δ assumes the same value throughout a network. This will allow the output of one gate element to easily couple to another gate element as input, so that gates can be ''wired'' directly into gate arrays implementing compounded logic operations.

Now, for a NOR gate implementation, the following must hold true:

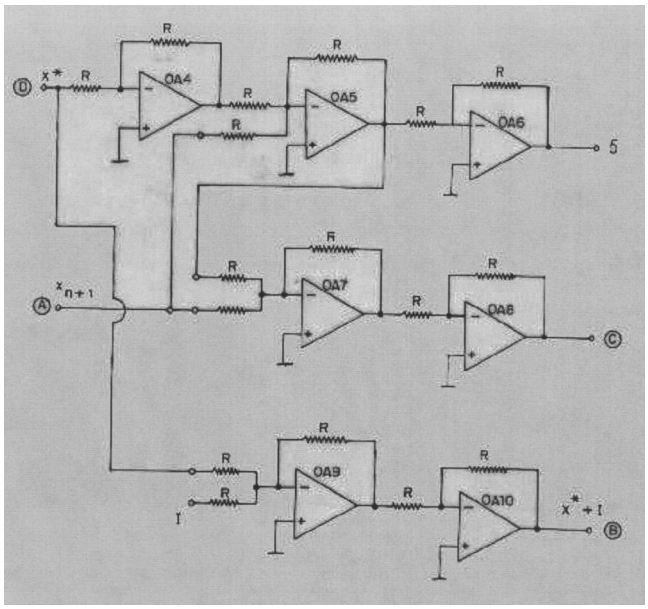

FIG. 5. Circuit module for implementing the NOR gate. See text for details.

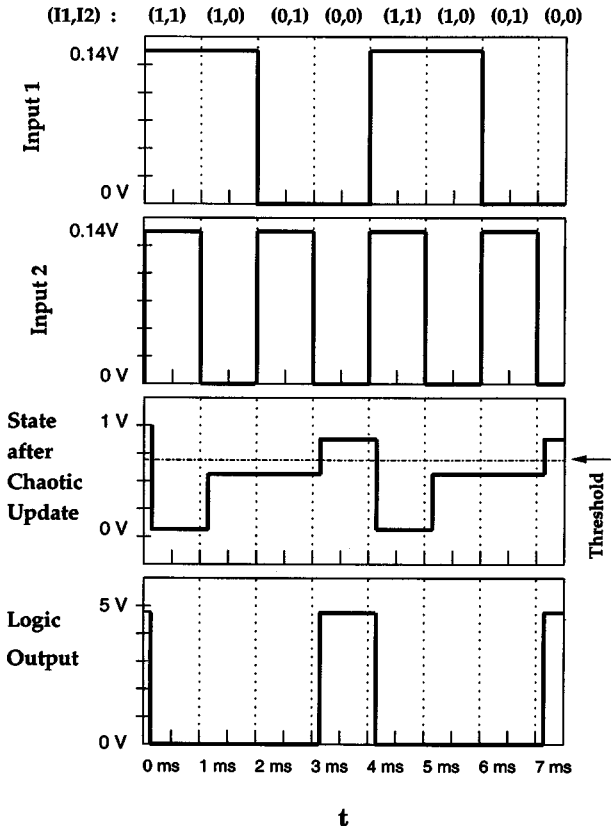

FIG. 6. Timing sequences from top to bottom: (1) first input I_1, (2) second input I_2, (3) state after chaotic update $f(x)$, and (4) output NOR (I_1, I_2) obtained by thresholding. Here, t refers to the timing of the logic signal waveforms. Note that the iteration frequency is much higher than logic signal frequency. The above waveforms are observed both in PSPICE circuit simulations and actual hardware realization within an accuracy of 5 mV.

(i) When $I_1 = 0$ and $I_2 = 0$, output is 1.
(ii) When $I_1 = 0$ and $I_2 = 1$, or $I_1 = 1$ and $I_2 = 0$, output is 0.
(iii) When $I_1 = 1$ and $I_2 = 1$, output is 0.

This is realized in the large range $x^* = 0.696$–0.75. In this regime, with $\delta = f(x^*) - x^* = 4x^*(1 - x^*) - x^*$, one has the following.

(i) $f(x^*) > x^*$; so when both $I_1 = I_2 = 0$, we have $Z \equiv f(x^*) - x^* = \delta$, thus yielding logic output 1.
(ii) $f(x^* + \delta) < x^*$; so when either I_1 or I_2 is 1, we have $Z \equiv 0$ thus yielding logic output 0.
(iii) $f(x^* + 2\delta) < x^*$; so when $I_1 = I_2 = 1$, we have $Z \equiv 0$ thus yielding logic output 0.

This clearly follows the NOR input-to-output association pattern, and this response pattern is robust as it can be obtained in a wide range of threshold.

The schematic diagram for the NOR gate implementation is depicted in Fig. 4. The actual circuit implementation of the NOR gate module is depicted in Fig. 5. Along with circuits of Figs. 1(a) and 1(c), the circuit of Fig. 5 is employed for the NOR gate realization. For this case, all marked terminals are

connected with respective terminals to form a closed circuitry. In Fig. 5, input I corresponds to input voltage levels and δ denotes the output voltage levels to implement the state of the NOR gate.

In our circuit, we specifically implement a threshold value of $x^*=0.7$. So, $\delta=4x^*(1-x^*)-x^*=0.14$; and (1) for input (0, 0), state after chaotic update is $f(0.7)=0.84>x^*$ and so output$=1$; (2) for inputs (0, 1)/(1, 0), state after chaotic update is $f(0.84)=0.5376<x^*$ and so output$=0$; (3) for input (1, 1), state after chaotic update is $f(0.98)=0.0784<x^*$ and so output$=0$ (see Fig. 6). So, the knowledge of the dynamics has allowed us to reverse engineer and obtain what must be done in order to select out the temporal patterns emulating the NOR gate [6].

In summary, we have experimentally demonstrated the basic principles for direct implementation of the fundamental NOR logic gate utilizing chaos. Thus, we provide a proof of principle experiment of the capability of chaotic systems for universal computing.

[1] S. Sinha and W. Ditto, Phys. Rev. Lett. **81**, 2156 (1998); S. Sinha and W. Ditto, Phys. Rev. E **60**, 363 (1999); S. Sinha *et al.*, *ibid.* **65**, 036214 (2002); **65**, 036216 (2002); T. Munakata *et al.*, IEEE Trans. Circuits Syst. **49**, 1629 (2002).

[2] M. M. Mano, *Computer System Architecture*, 3rd ed. (Prentice Hall, Englewood Cliffs, NJ, 1993); T. C. Bartee, *Computer Architecture and Logic Design* (McGraw-Hill, New York, 1991).

[3] S. Sinha and D. Biswas, Phys. Rev. Lett. **71**, 2010 (1993); L. Glass and W. Zheng, Int. J. Bifurcation Chaos Appl. Sci. Eng. **4**, 1061 (1994); S. Sinha, Phys. Rev. E **49**, 4832 (1994); Phys. Lett. A **199**, 365 (1995); Int. J. Mod. Phys. B **9**, 875 (1995); Phys. Rev. E **63**, 036212 (2001); S. Sinha and W. Ditto, *ibid.* **63**, 056209 (2001); S. Sinha, in *Nonlinear Systems*, edited by R. Sahadevan and M. Lakshmanan (Narosa, New Delhi, 2002), p. 309.

[4] R. J. Maddock and D. M. Calcutt, *Electronics: A Course for Engineers* (Addison Wesley Longman Ltd., London, 1997), p. 542.

[5] S. Hayes *et al.*, Phys. Rev. Lett. **70**, 3031 (1993).

[6] Our efforts are in contrast to the earlier efforts of A. Toth and K. Showalter, J. Chem. Phys. **103**, 2058 (1995), who obtain gates from chemical systems by delicately tuning many parameters (involving both the construction of the apparatus, as well as the geometric configuration and timing of the input and output waves). Fine adjustments of these lead to the desired phenomena. In contrast, here we have just one adjustable parameter.