Set-Reset Flip-Flop Circuit with a Simple Output Logic

I. Campos-Cantón*· E. Campos-Cantón†· H. C. Rosu‡· E. Castellanos-Velasco§

Abstract

The equation of the plane (EOP) in analytic geometry is used to build a logic *dynamic* architecture, i.e., a combination of set-reset flip flop (SR-FF) and basic logic gates. This is achieved by using two of the variables in the EOP as the input signals of the SR-FF and the remaining variable as the output signal. This theoretical proposal for mixing the SR-FF and the basic logic gates is confirmed experimentally by means of a simple electronic implementation.

KEY WORDS: set-reset flip-flop; equation of the plane; basic logic gates; bistable multivibrator

Circuits Syst Signal Process 31, 753-760 (2012)

*I. Campos-Cantón: Facultad de Ciencias, Universidad Autónoma de San Luis Potosí, Alvaro Obregón 64, 78000, San Luis Potosí, SLP, México, icampos@fc.uaslp.mx

†E. Campos-Cantón: IPICyT, Instituto Potosino de Investigacion Cientifica y Tecnologica, Camino a la Presa San José 2055, Col. Lomas 4a sección, CP. 78216, San Luis Potosí, S.L.P., Mexico, eric.campos@ipicyt.edu.mx

‡H.C. Rosu: IPICyT, Instituto Potosino de Investigacion Cientifica y Tecnologica, Camino a la Presa San José 2055, Col. Lomas 4a sección, CP. 78216, San Luis Potosí, S.L.P., Mexico, hcr@ipicyt.edu.mx

§E. Castellanos-Velasco: Facultad de Ciencias, Universidad Autónoma de San Luis Potosí, Alvaro Obregón 64, 78000, San Luis Potosí, SLP, México, ecastellv@fc.uaslp.mx

1 Introduction

Nonlinear dynamics is an important source of a rich variety of patterns which can be used in devices representing natural systems or even to perform computational tasks. Since about a decade, there is much interest in developing a type of dynamic logic architecture called chaos computing in which chaotic (non-linear) elements are employed to get the logical operations [14, 15]. There is already some progress concerning experimental setups whose functioning is based on this type of logic architecture [12, 6, 2]. The main goal in the applications is to achieve flexible structures that are able to change their response according to the control parameters and at the same time to reconfigure the settings of the device in order to obtain whatever logic result. In 1998, Sinha and Ditto [14] showed how a lattice of coupled chaotic maps can be used to simulate logic gates and perform specific arithmetic operations. This was the initial theoretical step indicating the possibility to build logic devices with dynamical architecture for computers and by 2005 Kuo [5] briefly reviewed the potential of chaotic elements to perform universal computing. More recently, Ditto and collaborators [4] reviewed the basic tenets of the chaos computing emerging paradigm and also discussed some proof of concept chips. In many theoretical works, the logistic maps are the chaotic elements used to simulate logic gates. Moreover, Sinha and Ditto [15] extended the scheme to encode numbers, perform specific arithmetic operations such as addition and multiplication, and other algorithms. Currently, there is stimulating research activity in exploiting the logic features of the nonlinear dynamical systems through their electronic and optoelectronic implementations, see [9, 7, 10, 8] and [13, 16], respectively. In addition, we point out that there are works in the literature on the reconfigurable properties of the neurons from the standpoint of Boolean logic gates, see e.g., [3].

The next step towards a full dynamical logic architecture is to build a device that has some elements of memory. The flip-flop electronic circuits are among the simplest ones since they are bistable oscillators that has two stable states and also one bit of memory. The fundamental latch of this type is the simple SR-FF. It is usually constructed from a pair of cross-coupled NAND or NOR logic gates. Such pairs of logic gates are called logic cells. The aim of this work is to present the theoretical and experimental results obtained by means of a simple SR-FF electronic circuit with dynamic logic architecture. The circuit design is based on the equation of the plane in analytic geometry. The electric diagram of the logic scheme is easy to implement in this type of

electronic circuit and the experimental results comply with the theoretical arguments given here. A few years ago, Cafagna and Grassi [1] proposed a SR flip flop based on a parallel architecture with a single chaotic Chua circuit. The difference between their work and the present one resides in the fact that their circuit has a full nonlinear dynamics whereas our circuit is essentially based on a linear dynamics. On the other hand, the designs of the two circuits are rather similar.

2 EOP and logic cells

In this section, we first show how to obtain a logic cell (LC) using the EOP. Next, we present the procedure to get an EOP-based SR-FF. Consider the EOP of the form

$$Ax_1 + Bx_2 + Cx_3 + D = 0 \ , \tag{1}$$

where A, B, and C are real constants and x_1 and x_2 are the input states of the logic cell with x_3 as its output. Then, the output function of the logic cell is given by

$$x_3 = N_1 x_1 + N_2 x_2 + M \ , \tag{2}$$

where $N_1 = -A/C$, $N_2 = -B/C$ and $M = -D/C$. Since LCs work with binary values, the states x_1 and x_2 are restricted to take values only from the set $\{0, 1\}$. Thus, the output function takes the values displayed in Table 1.

x_1	x_2	x_3
0	0	M
0	1	$N_2 + M$
1	0	$N_1 + M$
1	1	$N_1 + N_2 + M$

Table 1: The output x_3 given by the EOP equation (2) for the binary-valued inputs x_1 and x_2.

The LC has two different outputs if $N_1 \neq N_2$ for the inputs $(0, 1)$ and $(1, 0)$, but the same output is obtained if $N_1 = N_2$. Thus, we set $N_1 = N_2 = N$ which provides the following outputs: $U = M$ for the input $(0, 0)$, $V = N + M$ for the inputs $(0, 1)$ or $(1, 0)$, and $W = 2N + M$ for $(1, 1)$. A possible location of the outputs U, V, and W is illustrated in Figure 1, where

we also display an open interval $I_k = (-k, k)$ that generates a partition of the output phase space into I_k and its complement $I_k^c = (-\infty, -k] \cup [k, \infty)$.

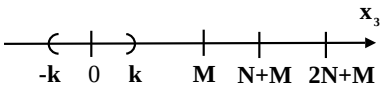

Fig. 1: The outputs x_3 of the EOP linear function and the open interval $)_k = (-k, k)$ used in the partition of the output phase space.

Now, it is possible to define a binary output under the following operation rule

$$y(x_3) = \begin{cases} 1, & \text{if } |x_3| < k \ , \\ 0, & \text{otherwise} \ . \end{cases} \tag{3}$$

Equations (2) and (3) define our LC. If the values of M and N are fixed at whatever real values, then there are several cases that produce different logic outputs. The interesting cases occur under the following assumption

$$0 < k < N < 2k \ . \tag{4}$$

Then, from the different sets of the output values of $\{U, V, W\}$ one can get the following logic gates

Case I: if $U \in I_k$ and $V, W \in I_k^c$. Then, the LC produces a NOR gate.

Case II: if $U, V \in I_k$ and $W \in I_k^c$. Then, the LC produces a NAND gate.

Case III: if $V \in I_k$ and $U, W \in I^c$. Then, the LC produces a XOR gate.

Case IV: if $V, W \in I_k$ and $U \in I_k^c$. Then, the LC produces an OR gate.

Case V: if $W \in I_k$ and $U, V \in I_k^c$. Then, the LC produces an AND gate.

The result of equations (2) and (3) is presented in Table 2.

We are ready now to develop a flip-flop procedure based on the EOP. Roughly speaking, a SR-FF device takes into account its output in order to elaborate a new output. Thus, in our case, we will consider a special type of feedback y for the EOP equation (2)

$$x_3 = N_1 x_1 + N_2 x_2 + M - F y \ , \tag{5}$$

	NOR	NAND	XOR	OR	AND
00	1	1	0	0	0
01	0	1	1	1	0
10	0	1	1	1	0
11	0	0	0	1	1

Table 2: The logic gates as produced by equations 2 and 3 under the condition 4.

x_1	x_2	x_3
0	0	$M - Fy$
0	1	$N_2 + M - Fy$
1	0	$N_1 + M - Fy$
1	1	$M - Fy$

Table 3: The RS-FF output x_3 according to equation (5).

where one should take $N_1 \neq N_2$ because the SR-FF inputs $(0, 1)$ and $(1, 0)$ add up to different results. Assuming that $0 < M$ and $N_1 = -N_2$, the resulting logical output x_3 is shown in Table 3. The feedback signal y acts like an offset. When it is in the lower state the offset is turned off, see Figure 2 (a), and if it is in the upper state the offset is turned on, see Fig. 2 (b). In order to get a SR-FF, the following requirements should be fulfilled $0 < N_2 + M < k < M$, see Fig. 2 (a) and $M - F < k$, see Fig. 2 (b).

Finally, making the change of variables $x_1 = S$, $x_2 = R$ and $y = Q$ in equations (3) and (5) one can get the SR-FF rules displayed in Table 4, where the subindex n corresponds to the present state, while the symbol X stands for Q_n or $\bar{Q}_n$ because the input (1,1) can be chosen at will.

It is worth noting that the SR flip flop structure proposed here has only

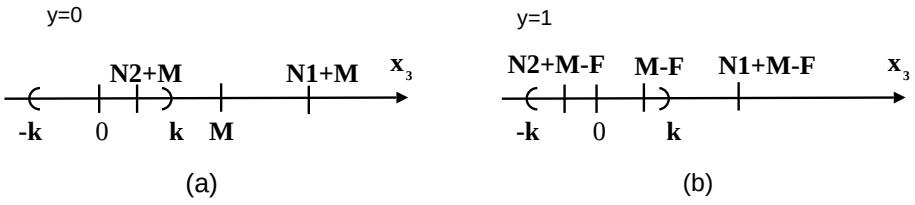

Fig. 2: (a) The output x_3 when the offset signal is turned off and (b) when it is turned on.

S	R	Q_{n+1}
0	0	Q_n
0	1	0
1	0	1
1	1	$X = Q_n$

Table 4: The SR-FF rules for the case $X = Q_n$. On the other hand, the case $X = \bar{Q}_n$ occurs when $N_1 \neq N_2$.

one feedback loop which is different of the standard flip flop based on two logic gates as used by Cafagna and Grassi [1]. This may be seen as an advantage since one can define the SR flip flop input (1,1) at will instead of being forbidden.

The approach presented here is confirmed experimentally in the electronic circuit presented in the next section.

3 Electronic Implementation of the SR-FF

Figure 3 shows the design of the schematic diagram of the SR-FF. This circuit consists of four comparators $LM311$ (labeled as A, B, C, D), four operational amplifiers $TL081$ (labeled as G, H, I, J), and thirteen resistors.

The circuit shown in Fig. 3 operates under the following logic states: The zero and one logic states have the potentials 0 V and 5 V, respectively. The design of the circuit permits that the potentials in the **a** and **b** nodes depend on the input values at x_1 and x_2. The outputs of the $LM311$ comparators A and B simulate a multiplication between N_1 and x_1 signals at the **a** node and between N_2 and x_2 signals at the **b** node, respectively. Thus, if x_1 is one logic, the potential at the **a** node is approximately N_1 because of the restriction $R_1 > 10R_0$, $\dfrac{R_1}{R_0 + R_1} N_1 \simeq N_1$. The other case is when x_1 is equal to zero implying a zero potential at the **a** node. In the same way, if x_2 is one logic, the potential at the **b** node is approximately N_2 under the restrictions $R_3 > 10R_2$, $\dfrac{R_3}{R_2 + R_3} N_2 \simeq N_2$. With the assumption that $N_1 = -N_2$, it is necessary to add an operational amplifier (labeled as G) with negative unitary gain, in which case the output is $c = -bR_4/R_3$. Then the potential at the x_3 node is obtained from the following equation

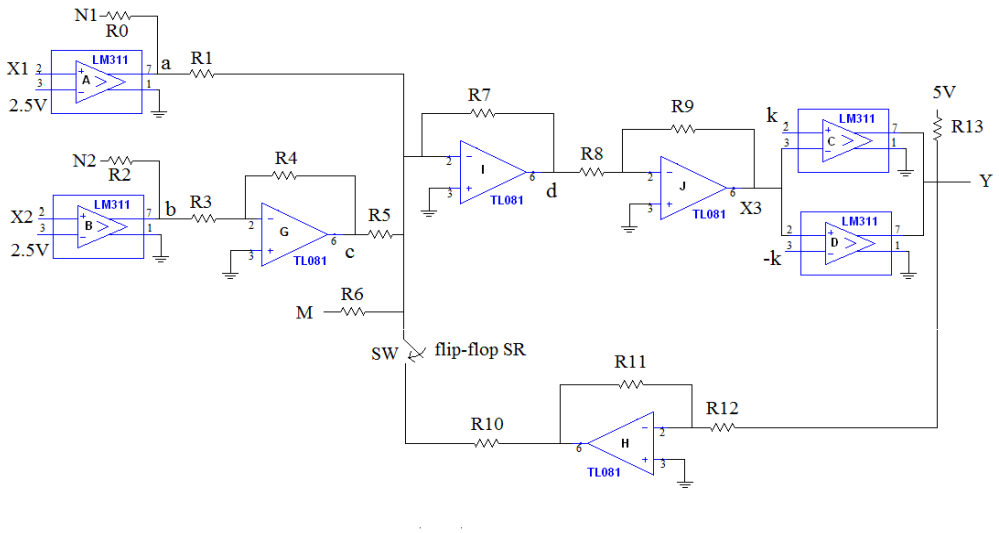

Fig. 3: Schematic diagram for the LC: a) if SW = open, we get a logic gate, b) if SW = close, we get the SR-FF.

$$x_3 = \begin{cases} \dfrac{R_9 R_7}{R_8}\left(\dfrac{a}{R_1} + \dfrac{c}{R_5} + \dfrac{M}{R_6}\right), & \text{if SW = open ,} \\ \dfrac{R_9 R_7}{R_8}\left(\dfrac{a}{R_1} + \dfrac{c}{R_5} + \dfrac{M}{R_6} - \dfrac{e}{R_{10}}\right), & \text{if SW = close ,} \end{cases} \tag{6}$$

where M is an offset value, $e = -yR_{11}/R_{12}$, and SW is an interruptor allowing the operation of the circuit as a flip flop or logic gate. The electronic design considers equal resistances for the set of resistors $R_1, R_3, R_4, R_5, R_6, R_7, R_8, R_9$ and R_{10}, such that equation (6) takes the form

$$x_3 = \begin{cases} N_1 x_1 - N_2 x_2 + M, & \text{if SW = open ,} \\ N_1 x_1 - N_2 x_2 + M - Fy, & \text{if SW = close ,} \end{cases} \tag{7}$$

where $F = R_{11}/R_{12}$. Finally, the comparators C and D define the open interval $I_k = (-k, k)$, where the y potential takes the one logic or zero logic values if the potential in the x_3 node belongs to the interval I_k or I_k^c, respectively. This operation corresponds to equation (3).

We implemented this electronic design on a printed circuit board (PCB) manufactured in our laboratory. A Tektronix $PS280$ unit provides ± 15 V to the comparators $LM311$ and operational amplifiers $TL081$. The resistances of the resistors employed experimentally $R_1, R_3, R_4, R_5, R_6, R_7, R_8, R_9$ and R_{10} have been all equal to $10\,k\Omega$, while in the case of the resistors R_0, R_2 and R_{13} we used $500\,\Omega$. The values of the F ratio and the potentials N_1, N_2, M and k are shown in Table 5. The maximum frequency for stable operation of the device is 1 MHz and the power consumption is 480 mW.

	AND	NAND	SR-FF
N_1	2	1.2	1
N_2	-4	-0.5	-3
M	6	0.5	1.5
F	1	1	$\frac{1}{5}$
k	1	1	1

Table 5: The numerical values of the quantities in the utmost left column for the logic gates AND and NAND and for the SR-FF.

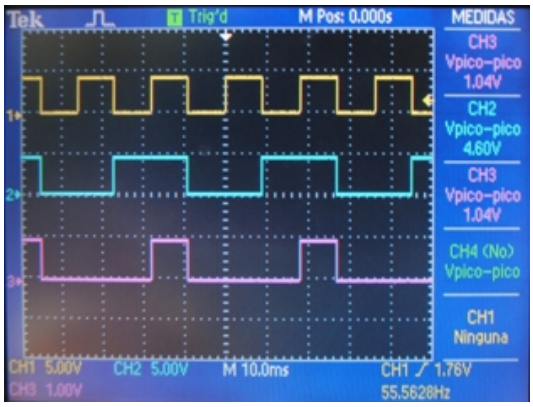

Fig. 4: The inputs and output of the basic logic gate AND for the numerical values given in Table 5.

Figure 4 shows the inputs and the output of the basic logic gate AND according to the values given in Table 5. The first (upper signal) and second (middle signal) channels correspond to the x_1 and x_2 inputs of (7) with SW open. The frequency of the x_1 signal is twice the frequency of x_2 signal in order to get the 00, 01, 10, 11 sequence. The third channel (down signal)

corresponds to the $y(x_3)$ output of equation (3). Figure 5 shows the inputs

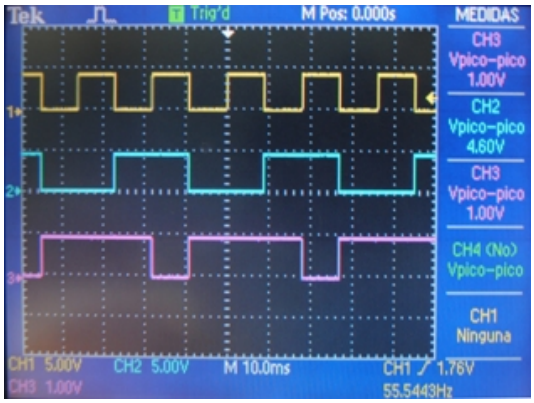

Fig. 5: The same as in the previous figure but for the basic logic gate NAND.

and the NAND output according to the values given in Table 5. The first (upper signal) and second (middle signal) channels correspond to the x_1 and x_2 inputs of (7) with SW open. The third channel (down signal) corresponds to the $y(x_3)$ output of equation (3). Finally, Fig. 6 shows the inputs and the

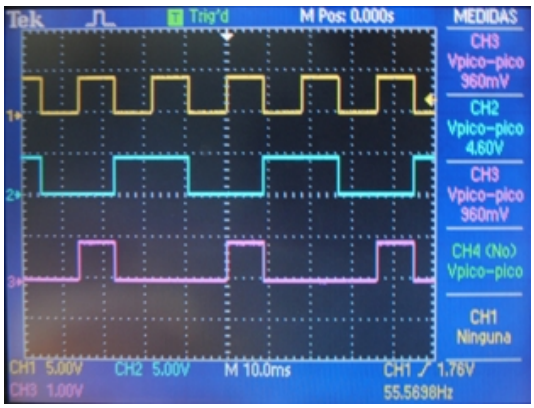

Fig. 6: The same as in the previous figure but for the SR flip flop. Notice that the input (1,1) is defined as the previous state Q_n.

output of the flip flop according to the values given in Table 5. The first (upper signal) and second (middle signal) channels correspond to the x_1 and x_2 inputs of (7) with SW close. The third channel (down signal) corresponds to $y(x_3)$ output of equation (3).

4 Conclusions

In this paper, we first presented a reconfigurable analog block able to simulate different logic gates and a SR-FF with the logic of the output signal based on the equation of the plane in analytic geometry. In addition, we validate this simple approach with some experimental results. The electronic circuit we propose has a dynamical architecture allowing the behavior of both basic logic gates and SR-FF. Such circuit realizations have many potential applications in reconfigurable computing. Finally, we notice that our design can be manufactured on just one chip because the final electronic circuit contains only semiconductors and passive components. This could lead to the reduction of the power supply.

References

[1] D. Cafagna, G. Grassi, Chaos-based SR-FF via Chua's circuit. Int. J. Bifurcation and Chaos **16**, 1521-1526 (2006)

[2] I. Campos-Cantón, E. Campos-Cantón, J.A. Pecina-Sanchez, H.C. Rosu, A simple circuit with dynamic logic architecture of basic logic gates. Int. J. Bifurcation and Chaos **20**, 2547-2551 (2010)

[3] F. Chen, W. Tang, G. Chen, Single-layer perceptron and dynamic neuron implementing linearly non-separable Boolean functions. Int. J. Circ. Theor. Appl. **37**, 433-451 (2009)

[4] W.L. Ditto, A. Miliotis, K. Murali. S. Sinha, M.L. Spano, Chaogates: Morphing logic gates that exploit dynamical patterns. Chaos **20**, 037107 (2010)

[5] D. Kuo, Chaos and its computing paradigm. IEEE Potentials **24**, 13-15 (2005)

[6] T. Munakata, S. Sinha, W. Ditto, Chaos computing: implementation of fundamental logical gates by chaotic elements. IEEE Trans. Circuits Syst., I: Fundam. Theory Appl. **49**, 1629-1633 (2002)

[7] K. Murali, A. Miliotis, W.L. Ditto, S. Sinha, Logic from nonlinear dynamical evolution. Phys. Lett. A **373**, 1346-1351 (2009)

[8] K. Murali, S. Sinha, W.L. Ditto, Implementation of NOR gate by a chaotic Chua's circuit. Int. J. Bifurcation and Chaos **13**, 2669-2672 (2003)

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

[10] K. Murali, S. Sinha, I.R. Mohamed, Chaos computing: experimental realization of NOR gate using a simple chaotic circuit. Phys. Lett. A **339**, 39-44 (2005)

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

[12] T. Roska, Circuits, computers, and beyond Boolean logic. Int. J. Circ. Theor. Appl. **35**, 485-496 (2007)

[13] A. Shinya, S. Mitsugi, T. Tanabe, M. Notomi, I. Yokohama, All-optical flip-flop circuit composed of coupled two-port resonant tunneling filter in two-dimensional photonic crystal slab. Opt. Express **14**, 1230-1235 (2006)

[14] S. Sinha, W. Ditto, Dynamics based computations. Phys. Rev. Lett. **81**, 2156-2159 (1998)

[15] S. Sinha, W. Ditto, Computing with distributed chaos. Phys. Rev. E **60**, 363-377 (1999)

[16] S.L. Yan, Chaotic synchronization of two inverse-injection semiconductor lasers for logic gates. Opt. Review **18**, 42-46 (2011)