A novel circuit design for complementary resistive switch-based stateful logic operations*

Xiao-Ping Wang(王小平)[1,2,†], Lin Chen(陈林)[1,2], Yi Shen(沈轶)[1,2], and Bo-Wen Xu(徐博文)[1,2]

[1] School of Automation, Huazhong University of Science and Technology, Wuhan 430074, China
[2] Research Institute of Huazhong University of Science and Technology in Shenzhen, Shenzhen 518000, China

(Received 10 November 2015; revised manuscript received 13 January 2016; published online 5 April 2016)

Recently, it has been demonstrated that memristors can be utilized as logic operations and memory elements. In this paper, we present a novel circuit design for complementary resistive switch (CRS)-based stateful logic operations. The proposed circuit can automatically write the destructive CRS cells back to the original states. In addition, the circuit can be used in massive passive crossbar arrays which can reduce sneak path current greatly. Moreover, the steps for CRS logic operations using our proposed circuit are reduced compared with previous circuit designs. We validate the effectiveness of our scheme through Hspice simulations on the logic circuits.

Keywords: memristor, complementary resistive switch, crossbar arrays, logic circuits

PACS: 85.35.–p, 85.25.Hv, 87.85.Qr, 84.32.–y **DOI:** 10.1088/1674-1056/25/5/058502

1. Introduction

As it has become increasingly difficult to overcome various physical limits of the traditional CMOS technology,[1] alternative elements are desired for higher performance. An element called memristor (short for "memory resistor") is a promising candidate. A memristor was first theoretically postulated by Chua in 1971,[2] and later, Williams's team presented a resistance variable device as a memristor at HP Labs in 2008.[3] Different from other novel nonvolatile random access memories,[4,5] as an emerging nanoscale device, memristor has a lot of advantageous features, such as nonvolatility, high density, low power, and good scalability. Recently, researchers have also demonstrated its potential applications in programmable analog circuits,[6–8] chaotic oscillator circuits,[9–11] and neural networks.[12]

In addition, a memristor can be used in logic circuits,[13–15] logic arrays,[16–18] and other structures like CMOS-memristor hybrids.[19] Over the years, the memristor has been proposed as a replacement for the current CMOS-based memory architectures. For example, memristor-based crossbar arrays were used as feasible nonvolatile random access memories (RAMs) in Ref. [20]. However, passive crossbar arrays have the common problem about sneak path current due to interference from the neighboring cells when selecting a designated cell within the arrays.[21–23] Recently, the complementary resistive switch (CRS), which consists of two bipolar memristive elements anti-serially, was proposed to overcome the disadvantage of sneak path.[24] The element, including a high resistance state and a low resistance state, is used to encode either logic 0, 1, or ON state (see Fig. 1). If memristive element X is in the high resistance state (HRS), and memristive element Y is in the low resistance state (LRS), then CRS is in state 0. If memristive element X is in the LRS, and memristive element Y is in the HRS, then CRS is in state 1. If memristive elements X and Y are both in the LRS, then the CRS is in state ON. The transitions between these states depend on the applied biasing voltage across the CRS, controlled by V_{th1}, V_{th2}, V_{th3}, and V_{th4} threshold voltages.[24]

Since Linn *et al.* proposed the CRS architecture in Ref. [24], many scholars have paid much attention to the CRS. In Ref. [15], Yang *et al.* proposed a scheme to implement the CRS-based stateful logic operations using material implication (IMP). However, during the IMP operation, the state 1 stored in the CRS_P was reset to state 0, and for the NAND operation, the state 1 stored in the CRS_{P1} and CRS_{P2} was similarly reset to state 0. This scheme for the stateful logic operations partially destructs the original states stored in the input CRS cells. Moreover, the CRSs used in input and output cells are two different types of elements, then the configuration and parameter of input cell (CRS_P) and output cell (CRS_Q) are completely different. The output cell CRS_Q can be only used as an output cell, which cannot be used as an input cell to perform the next logic operation, so it is hardly applied to massive passive crossbar arrays. In order to solve these issues, we put forward a circuit which consists of two parts including two CMOS switches, a resistor, and two voltage comparators. The

*Project supported by the National Natural Science Foundation of China (Grant Nos. 61374150 and 11271146), the State Key Program of the National Natural Science Foundation of China (Grant No. 61134012), the Doctoral Fund of Ministry of Education of China (Grant No. 20130142130012), and the Science and Technology Program of Shenzhen City, China (Grant No. JCYJ20140509162710496).

†Corresponding author. E-mail: wangxiaoping@hust.edu.cn

© 2016 Chinese Physical Society and IOP Publishing Ltd

circuit can be highly compatible with the CRS, and can be integrated to the peripheral circuit. The circuit will not bring extra area consumption of storage cells in massive passive crossbar arrays either. The main contributions of this paper are as follows.

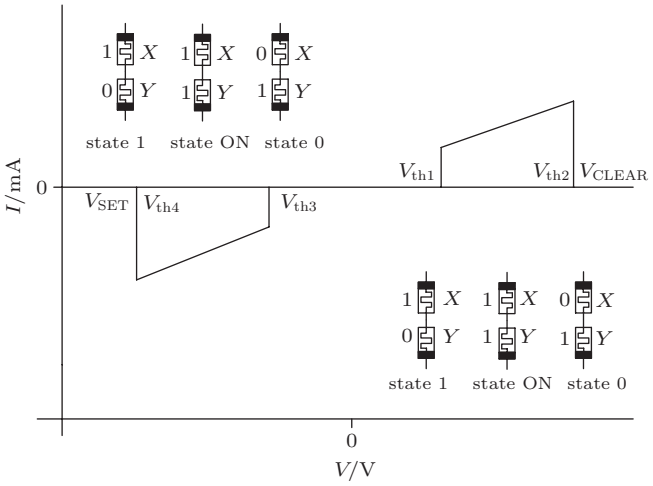


Fig. 1. CRS *I*–*V* characteristic and its states (state 0, state 1, state ON).

i) The proposed circuit can automatically write the destructive CRS cells back to the original states. In Ref. [15], the state 1 stored in CRS_P will be changed to state 0. However, the state of input cells will not be destroyed by using our proposed circuit. This virtue can ensure the integrity of the data, which is of high importance in some fields.

ii) The proposed circuit can be applied to massive passive crossbar arrays. The output cell CRS_Q in Ref. [15] can be only used as the output cell, which cannot be used as the input cell to perform the next logic operation. However, every storage cell in crossbar arrays can be either used as input cell or output cell using our proposed circuit.

iii) The steps of logic operations using our proposed circuit are fewer compared with Refs. [13]–[15]. Here, we only need two steps to implement the CRS IMP operation, and only three steps are required to execute CRS NAND operation regardless of the number of inputs. However, three steps are required to implement the CRS IMP operation, and four steps are required to perform CRS NAND operation in Ref. [15]. In Refs. [13] and [14], $N+1$ steps are required to execute N inputs of NAND operation.

The remainder of this paper is organized as follows. Section 2 presents our proposed circuit for CRS IMP, NAND operations and the CRS-based crossbar architecture with our proposed circuit. In Section 3, simulation results are provided to demonstrate the effectiveness and superior performance of the proposed circuit. Section 4 concludes this paper.

2. The proposed circuit and its applications to crossbar architecture

In this section, at first, we describe the CRS IMP operation using the proposed circuit. On this basis, we propose the CRS NAND operation. In order to apply CRS-based logic operations to crossbar arrays, we choose the same type of CRS as the CRS_P and CRS_Q. By implementing the two stateful logic operations, we show that our proposed circuit can perfectly address these problems existing in Ref. [15].

2.1. CRS IMP operation using our proposed circuit

To perform the CRS IMP operation using our proposed circuit, we consider the basic structure shown in Fig. 2.

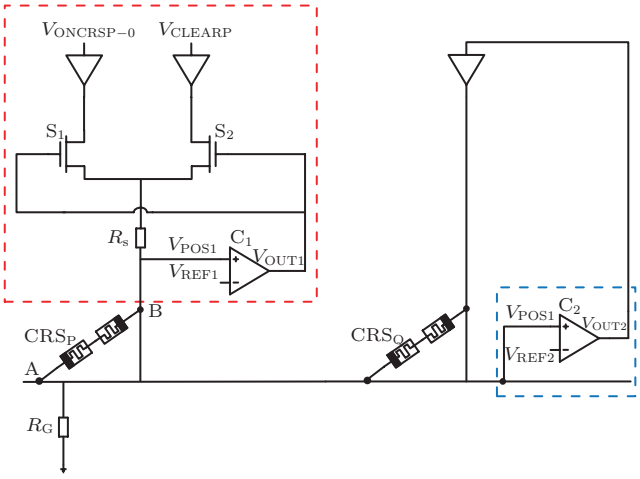


Fig. 2. The basic structure of the CRS IMP operation using the proposed circuit.

The circuits of red dashed line and blue dashed line present our proposed circuit. In the circuit of red dashed line, we adopt two CMOS switches S_1 and S_2, a voltage comparator C_1, and a resistor R_S of which resistance is fairly small compared with the resistance of CRS_P state ON (R_{CRSONP}). In the circuit of the blue dashed line, we only use a voltage comparator C_2 to implement the related functions. In order to perform the CRS IMP operation using our proposed circuit, voltage $V_{ONCRSP-0}$ and V_{CLEARP} are applied to switch S_1 and S_2. Voltage $V_{ONCRSP-0}$ is negative which can only switch CRS_P from state 0 to state ON, and voltage V_{CLEARP} is positive which resets CRS_P to state 0. The output voltage of C_1, namely V_{OUT1}, is used to select switch S_1 or S_2. The S_1 is a PMOS switch, and the S_2 is a NMOS switch. When voltage V_{OUT1} is negative, switch S_1 is turned on, S_2 is turned off, and voltage $V_{ONCRSP-0}$ is applied to CRS_P, resistor R_S and R_G. When voltage V_{OUT1} is positive, switch S_2 is turned on, S_1 is turned off, and voltage V_{CLEARP} is applied to the circuit. Now, we show how the circuit in Fig. 2 is used to perform the IMP operation. At the beginning, voltage V_{POS1} (positive terminal of C_1) is equal to

zero. We apply a positive pulse to voltage V_{REF1}, and thus voltage V_{OUT1} will be negative, which turns on switch S_1, imposes voltage $V_{\mathrm{ONCRSP-0}}$ to the circuit. Under voltage $V_{\mathrm{ONCRSP-0}}$, $\mathrm{CRS_P}$ either stays in state 1 or switches from state 0 to state ON. The magnitude of the current passing through R_{S}, $\mathrm{CRS_P}$, and R_{G} when $\mathrm{CRS_P}$ is in state ON is larger than the one when $\mathrm{CRS_P}$ is in state 0. In order to perform CRS IMP operation correctly, the relationships of various operation voltages are given in Eqs. (1)–(3). When $\mathrm{CRS_P}$ is in state ON, we define the potential of the node B (see Fig. 2) as V_{ONPOS1}, and we define the potential of the node A as V_{GON}. When $\mathrm{CRS_P}$ is in state 1, we define the potential of the node B as V_{1POS1}, and we define the potential of the node A as V_{G1}. Obviously, voltage V_{GON} and V_{G1} meet Eq. (4). Similarly, the voltage drop over R_{S} when $\mathrm{CRS_P}$ is in state ON is much larger than the one when $\mathrm{CRS_P}$ is in state 1. That is to say, voltage V_{1POS1} with $\mathrm{CRS_P}$ staying in state 1 is less than V_{ONPOS1} with $\mathrm{CRS_P}$ staying in state ON, so voltage V_{1POS1} and V_{ONPOS1} meet Eq. (5). Figure 3 presents the waveforms of the added voltage V_{REF1} and V_{REF2}. Here, the proposed circuit is designed by using the parameters listed in Section 3. The voltages including V_{ONPOS1}, V_{1POS1}, V_{G1}, and V_{GON} are measured in the circuit simulation, and the reference voltages, namely V_{REF1} and V_{REF2}, are the added voltages.

$$V_{\mathrm{Pth4}} < V_{\mathrm{ONCRSP-0}} < V_{\mathrm{Pth3}}, \tag{1}$$

$$V_{\mathrm{Pth2}} < V_{\mathrm{CLEARP}}, \tag{2}$$

$$V_{\mathrm{SETQ}} < V_{\mathrm{Qth4}}, \tag{3}$$

$$V_{\mathrm{GON}} < V_{\mathrm{G1}}, \tag{4}$$

$$V_{\mathrm{1POS1}} < V_{\mathrm{ONPOS1}}, \tag{5}$$

where V_{Pth2}, V_{Pth3}, and V_{Pth4} are the threshold voltages of $\mathrm{CRS_P}$, V_{Qth4} is the threshold voltage of $\mathrm{CRS_Q}$, V_{1POS1}, V_{ONPOS1}, V_{GON}, and V_{G1} are all negative voltage. As shown in Fig. 3, in Step 1, V_{REF1} is a positive voltage, and R_{REF2} is negative which is larger than V_{GON} and less than V_{G1}. In Step 2, voltage V_{REF1} is negative, which is larger than V_{1POS1} and less than V_{ONPOS1}; and voltage R_{REF2} is also negative which is less than V_{GON} and V_{G1}. Now, we analyze the four input cases and the operation process.

Case 1 $\mathrm{CRS_P}$ is in state 0, $\mathrm{CRS_Q}$ is also in state 0. As shown in Fig. 3, firstly, voltage V_{REF1} is larger than V_{POS1} which is about 0, and V_{OUT1} is negative which turns on S_1. Then $V_{\mathrm{ONCRSP-0}}$ is applied to the circuit (shown in Fig. 2). Under voltage $V_{\mathrm{ONCRSP-0}}$, $\mathrm{CRS_P}$ will be changed to state ON, then voltage V_{POS2} is equal to V_{GON} which is smaller than V_{REF2} (see Fig. 3). V_{OUT2} will be a negative voltage which is defined as V_{SETQ}, and voltage V_{SETQ} will set $\mathrm{CRS_Q}$ to state 1. Secondly, when $\mathrm{CRS_P}$ switches to state ON, V_{REF1} will be less than V_{ONPOS1}. In this way, switch S_2 is turned on, and voltage V_{CLEARP} will be applied to the circuit. Under voltage V_{CLEARP}, $\mathrm{CRS_P}$ will be changed from state ON to state 0. Note that voltage V_{CLEARP} is positive. As a result, V_{POS1} is also positive, switch S_2 will be always turned on. $V_{R_{\mathrm{G}}}$ will be a small positive voltage, at the moment, we will apply a negative voltage to V_{REF2} (see Fig. 3). So V_{OUT2} will be a positive voltage which we define V_{LOW}. Voltage V_{LOW} is less than V_{Qth1}, which does not change the state of $\mathrm{CRS_Q}$, here, V_{Qth1} is the threshold voltage of $\mathrm{CRS_Q}$, then $\mathrm{CRS_Q}$ will still stay in state 1. After that, $\mathrm{CRS_P}$ is in state 0, and $\mathrm{CRS_Q}$ is changed to state 1.

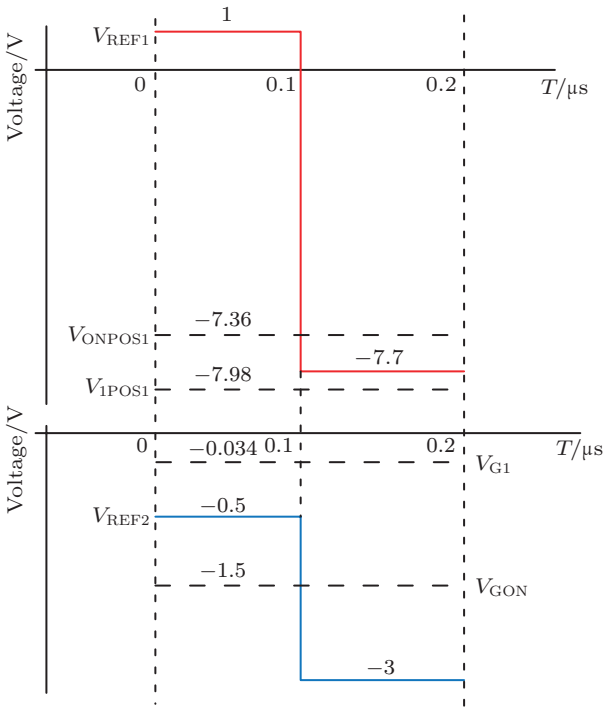


Fig. 3. The waveforms of the added voltage V_{REF1} and V_{REF2} in our proposed circuit.

Case 2 $\mathrm{CRS_P}$ is in state 0, $\mathrm{CRS_Q}$ is in state 1. Like Case 1, under $V_{\mathrm{ONCRSP-0}}$, $\mathrm{CRS_P}$ will be changed to state ON, then under voltage V_{CLEARP}, $\mathrm{CRS_P}$ will return to its original state (state 0). At the same time, $\mathrm{CRS_Q}$ will not be changed. After that, $\mathrm{CRS_P}$ is in state 0, $\mathrm{CRS_Q}$ is in state 1.

Case 3 $\mathrm{CRS_P}$ is in state 1, $\mathrm{CRS_Q}$ is in state 0. Firstly, under voltage $V_{\mathrm{ONCRSP-0}}$, $\mathrm{CRS_P}$ remains in state 1, then voltage V_{POS2} is equal to V_{G1} which is larger than V_{REF2}, V_{OUT2} will not be equal to voltage V_{LOW} which will not change the state of $\mathrm{CRS_Q}$. Secondly, when $\mathrm{CRS_P}$ is in state 1 under voltage $V_{\mathrm{ONCRSP-0}}$, V_{POS1} is equal to V_{1POS1} which is less than voltage V_{REF1}. Then V_{OUT1} is still negative, and switch S_2 will not be turned on. In this way, voltage V_{POS2} is still equal to V_{G1}, which is still larger than V_{REF2}. So, V_{OUT2} will still be equal to voltage V_{LOW} which will not change the state of $\mathrm{CRS_Q}$. After that, $\mathrm{CRS_P}$ is in state 1, $\mathrm{CRS_Q}$ is in state 0.

Case 4 $\mathrm{CRS_P}$ is in state 1, $\mathrm{CRS_Q}$ is also in state 1. Like Case 3, under $V_{\mathrm{ONCRSP-0}}$, $\mathrm{CRS_P}$ remains in state 1. At the

same time, CRS_Q will not be changed. After that, CRS_P is in state 1, CRS_Q is in state 1.

The CRS IMP operation can be implemented using the proposed circuit during two steps. Table 1 presents the truth table, and Table 2 presents the applied voltages and steps of CRS IMP operation. In Section 3, we will present the simulation on CRS IMP operation, and the simulation result validates our proposed scheme.

Table 1. The truth table of CRS IMP operation.

CRS_P	CRS_Q	Step 1		Step 2	
		CRS'_P	CRS'_Q	CRS''_P	CRS''_Q
0	0	ON	1	0	1
0	1	ON	1	0	1
1	0	1	0	1	0
1	1	1	1	1	1

Table 2. The operation voltages and steps of the CRS IMP operation.

	V_P	V_Q	Function
Step 1	$V_{ONCRSP-0}$	V_{SETQ} (if CRS_P is in state ON) V_{LOW} (if CRS_P is in state 1)	Operation
Step 2	V_{CLEARP} (if CRS_P is in state ON) $V_{ONCRSP-0}$ (if CRS_P is in state 1)	V_{LOW} (less than V_{Qth1})	Restoration

2.2. CRS NAND operation using our proposed circuit

According to Ref. [13], any logic operation can be implemented using NAND gates. Hence, the following section will illustrate the details of CRS NAND operation. The two inputs NAND operation needs three CRS cells, CRS_{P1}, CRS_{P2} (NAND inputs), and CRS_Q which will store the result after NAND operation. Figure 4 presents the basic structure for performing the CRS NAND operation.

Like the CRS IMP operation, the circuits of red dashed line and green dashed line are added to CRS_{P1} and CRS_{P2} respectively, and the circuit of blue dashed line is used to drive CRS_Q. The configuration and parameter in the circuits of red dashed line and green dashed line are completely the same. In order to describe the CRS NAND operation, we make some definitions. When CRS_{P1} is in state ON under voltage $V_{ONCRSP-0}$, we define the potential of the node B as V_{ONPOS1} (see Fig. 4). When CRS_{P1} is still in state 1 under voltage $V_{ONCRSP-0}$, we define the potential of the node B as V_{1POS1}. Similarly, we can define V_{ONPOS2} and V_{1POS2}. When CRS_{P1} and CRS_{P2} are both in state 1 under voltage $V_{ONCRSP-0}$, we define the potential of the node A as $V_{1-1POS3}$. When one of CRS_{P1} and CRS_{P2} is in state ON and the other is in state 1 under voltage $V_{ONCRSP-0}$, we define the potential of the node A as $V_{1-ONPOS3}$. When CRS_{P1} and CRS_{P2} are both in state ON under voltage $V_{ONCRSP-0}$, we define the potential of the node A as $V_{ON-ONPOS3}$. Apparently, equations (7)–(9) are all satisfied. Figure 5 presents the waveforms of the added voltage V_{REF1}, V_{REF2}, and V_{REF3}. Here, like the CRS IMP operation, the proposed circuit is also designed by using the parameters listed in Section 3. The voltages including V_{ONPOS1}, V_{1POS1}, V_{ONPOS2}, V_{1POS2}, $V_{1-1POS3}$, $V_{1-ONPOS3}$, and $V_{ON-ONPOS3}$ are measured in the circuit simulation, and the reference voltages, namely V_{REF1}, V_{REF2}, and V_{REF3} are the added voltages.

$$V_{Qth2} < V_{CLEARQ}, \tag{6}$$

$$V_{ON-ONPOS3} < V_{1-ONPOS3} < V_{1-1POS3}, \tag{7}$$

$$V_{1POS1} < V_{ONPOS1}, \tag{8}$$

$$V_{1POS2} < V_{ONPOS2}, \tag{9}$$

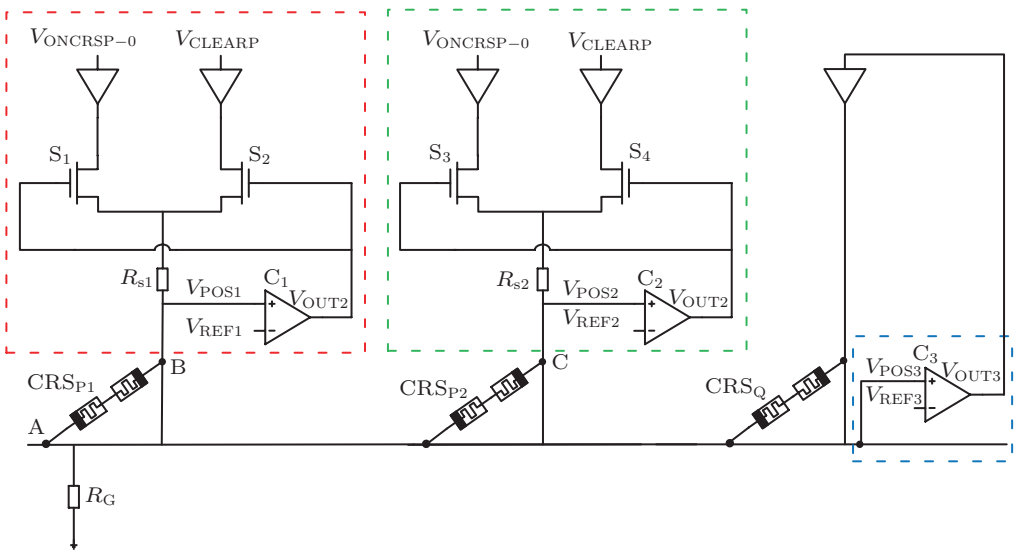


Fig. 4. The basic structure of CRS NAND operation using the proposed circuit.

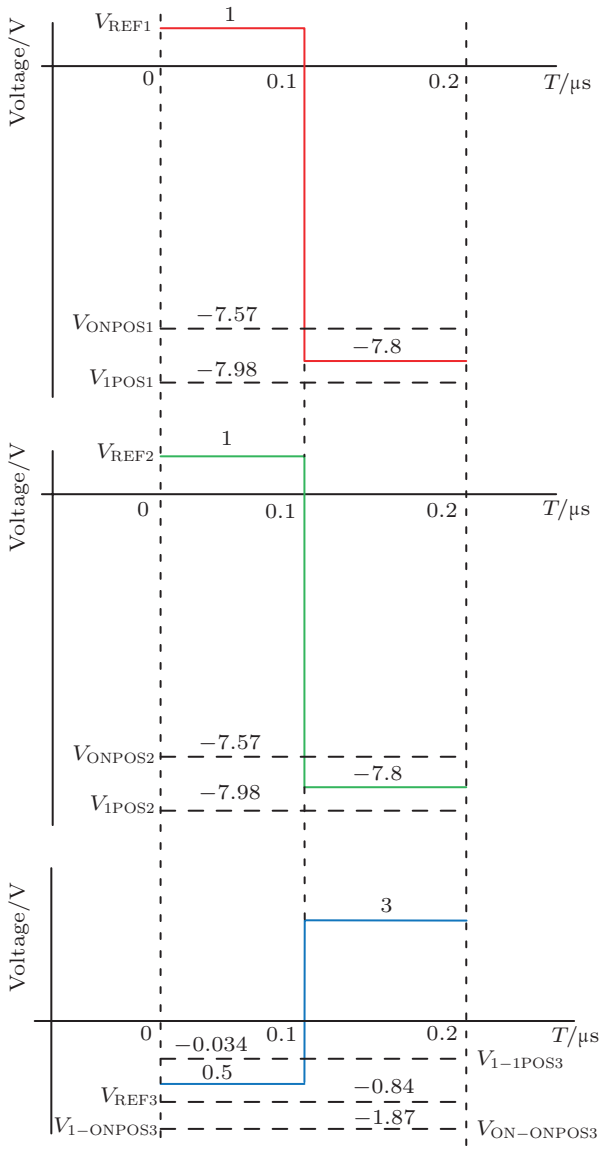


Fig. 5. The waveforms of the added voltage V_{REF1}, V_{REF2}, and V_{REF3} in our proposed circuit.

where V_{Qth2} is the threshold voltage of CRS_Q, voltage V_{ONPOS1}, V_{ONPOS2}, and $V_{1-1POS3}$ are all negative. To implement the CRS NAND operation, in Step 1, we apply voltage V_{SETQ} to CRS_Q, which sets CRS_Q to state 1 unconditionally. Then we remove voltage V_{SETQ}. Considering Fig. 5, in Step 2, for the fact that voltages V_{POS1} and V_{POS2} are about 0, we apply a positive voltage to V_{REF1} and V_{REF2}, then V_{OUT1} and V_{OUT2} are negative and will turn on switch S_1 and S_3, so voltage $V_{ONCRSP-0}$ will be added to the circuit. At the same time, a negative voltage is applied to V_{REF3} which is larger than $V_{1-ONPOS3}$, $V_{ON-ONPOS3}$, and less than $V_{1-1POS3}$. In Step 3, we apply a negative voltage to V_{REF1} which is larger than V_{1POS1} and less than V_{ONPOS1}, and it is the same as voltage V_{REF2}. In the meantime, a positive voltage is applied to V_{REF3} which is larger than V_{POS3}. Now, we analyze the four input cases and the operation process.

Case 1 CRS_{P1} is in state 0, and CRS_{P2} is also in state 0. Firstly, voltage V_{SETQ} sets CRS_Q state 1, then voltage $V_{ONCRSP-0}$ is added to the circuit, CRS_{P1} and CRS_{P2} will be changed to state ON under voltage $V_{ONCRSP-0}$. At the same time, the added voltage V_{REF3} is larger than $V_{ON-ONPOS3}$, then voltage V_{OUT3} is a negative voltage which we define as V_{QLOW}. The magnitude of V_{QLOW} is less than V_{Qth3}, which will not change the state of CRS_Q, here, V_{Qth3} is the threshold voltage of CRS_Q. Secondly, the added negative voltage V_{REF1} and V_{REF2} are less than V_{ONPOS1} and V_{ONPOS2}, respectively. So, voltage V_{OUT1} and V_{OUT2} are positive which will turn on switch S_2 and S_4, voltage V_{CLEARP} is added to the circuit. Under voltage V_{CLEARP}, CRS_{P1} and CRS_{P2} will be reset to state 0, and voltage V_{POS1} and V_{POS2} will be positive which are larger than V_{REF1} and V_{REF2} respectively, switch S_1 and S_3 will not be always turned on, therefore CRS_{P1} and CRS_{P2} will remain in state 0. However, biased by voltage V_{CLEARP}, V_{POS3} is a small positive voltage. So we apply an appropriate positive voltage to V_{REF3} which is larger than V_{POS3}, the voltage V_{OUT3} is equal to V_{QLOW} which will not change the state of CRS_Q. In this way, CRS_Q is still in state 1. After that, CRS_{P1} is in state 0, CRS_{P2} is in state 0, and CRS_Q is in state 1.

Case 2 CRS_{P1} is in state 0, CRS_{P2} is in state 1. Firstly, voltage V_{SETQ} sets CRS_Q state 1, then voltage $V_{ONCRSP-0}$ is added to the circuit. Under voltage $V_{ONCRSP-0}$, CRS_{P1} will switch to state ON, while CRS_{P2} keeps the state. Voltage V_{POS3} is $V_{1-ONPOS3}$, while the added voltage V_{REF3} is larger than $V_{1-ONPOS3}$, so voltage V_{OUT3} is equal to V_{QLOW} which will not change the state of CRS_Q. Secondly, the added negative voltage V_{REF1} is less than V_{ONPOS1}, then switch S_2 will be turned on, so CRS_{P1} will be reset to state 0 biased by voltage V_{CLEARP}. For the fact that the added negative voltage V_{REF2} is larger than V_{1POS2}, then switch S_4 will not be turned on, so the state of CRS_{P2} will not be changed. While biased by voltage $V_{ONCRSP-0}$ and V_{CLEARP}, V_{POS3} is a small positive voltage. Like Case 1, we apply the same voltage to V_{REF3} which is larger than V_{POS3}. In this way, CRS_Q is still in state 1. After that, CRS_{P1} is in state 0, CRS_{P2} is in state 1, and CRS_Q is in state 1.

Case 3 CRS_{P1} is in state 1, CRS_{P2} is in state 0. Case 3 is the same as Case 2. After that, CRS_{P1} is in state 1, CRS_{P2} is in state 0, and CRS_Q is in state 1.

Case 4 CRS_{P1} is in state 1, CRS_{P2} is also in state 1. Firstly, voltage V_{SETQ} sets CRS_Q to state 1, then voltage $V_{ONCRSP-0}$ is added to the circuit. Under voltage $V_{ONCRSP-0}$, CRS_{P1} and CRS_{P2} will keep the state. In the meantime, voltage V_{POS3} is equal to $V_{1-1POS3}$, and voltage V_{REF3} is less than $V_{1-1POS3}$ (shown in Fig. 5), which makes comparator C_3 output a positive voltage. Here, we define the positive voltage V_{CLEARQ}, which

Table 3. The truth table of CRS NAND operation.

CRS_{P1}	CRS_{P2}	Step 1	Step 2			Step 3		
		CRS_Q	CRS'_{P1}	CRS'_{P2}	CRS'_Q	CRS''_{P1}	CRS''_{P2}	CRS''_Q
0	0	1	ON	ON	1	0	0	1
0	1	1	ON	1	1	0	1	1
1	0	1	1	ON	1	1	0	1
1	1	1	1	1	0	1	1	0

Table 4. The operation voltages and steps of the CRS NAND operation.

	V_{P1}	V_{P2}	V_Q	Function
Step 1	–	–	V_{SETQ}	Initialization
Step 2	$V_{ONCRSP-0}$	$V_{ONCRSP-0}$	V_{CLEARQ} (CRS_{P1} = state 1, CRS_{P2} = state 1) V_{QLOW} (other cases, larger than V_{Qth3})	Operation
Step 3	$V_{ONCRSP-0}$ (CRS_{P1} = state 1) V_{CLEARP} (CRS_{P1} = state ON)	$V_{ONCRSP-0}$ (CRS_{P2} = state 1) V_{CLEARP} (CRS_{P2} = state ON)	V_{QLOW} (larger than V_{Qth3})	Restoration

can reset CRS_Q to state 0. Secondly, the added negative voltage V_{REF1} and V_{REF2} are larger than V_{1POS1} and V_{1POS2}, respectively. So, voltage V_{OUT1} and V_{OUT2} are negative which will not turn on switch S_2 and S_4, and voltage V_{CLEARP} will not be added to the circuit. At the same time, the added positive voltage V_{REF3} is larger than $V_{1-1POS3}$. The voltage V_{OUT3} is equal to V_{QLOW} which will not change the state of CRS_Q. After that, CRS_{P1} is in state 1, CRS_{P2} is in state 1, and CRS_Q is in state 0.

Table 5. Comparison about steps of IMP and NAND operations.

Logic operation	Implementation steps		
	CRS using our proposed circuit	CRS[15]	Memristor[13,14]
IMP	2	3	1
2 inputs NAND	3	4	3
3 inputs NAND	3	4	4
4 inputs NAND	3	4	5
N inputs NAND	3	4	$N+1$

The CRS NAND operation can be implemented using the proposed circuit during three steps. Table 3 presents the truth table, and Table 4 presents the operation voltages and steps of CRS NAND operation. In Section 3, we will show the simulation on CRS NAND operation, and the simulation result validates our proposed scheme.

To perform N inputs CRS NAND operation, we need $N+1$ CRS cells to achieve the function, in which N CRS cells are used for N inputs cells, and the other is considered as the output cell. It needs only three steps. In Step 1, CRS_Q is initialized to state 1, then N inputs CRS NAND operation can be executed by two steps using our proposed circuit. For the CRS NAND operation in Ref. [10], it needs four steps. For memristor NAND operation in the previous literature, it needs N times IMP, so N inputs memristor NAND operation requires $N+1$ steps (N IMP steps plus one initialization). The comparison about steps of IMP and NAND operations is shown in Table 5.

2.3. CRS-based crossbar architecture with our proposed circuit

In the preliminary, we have already implemented the two foremost CRS logic operations. According to Ref. [13], any logic operation can be performed using the NAND gates. In addition, other stateful logic operations can be also implemented using our proposed circuit. We will apply the CRS NAND operation with our proposed circuit to crossbar architecture. Figure 6 shows the basic structure. In Fig. 6, the circuits of red, green, and blue dashed line present our proposed circuit. The row decoder circuit, column decoder, and load resistor circuit are used to choose the selected row and column. To perform the CRS NAND operation in crossbar arrays, we take 4×4 crossbar arrays for example. The three rows in front are selected, and the second column is selected. In this way, the two red circled CRS cells are used as the input cells, and the blue circled CRS cell is considered as output cell. According to the operation steps presented in Section 2.2, we can implement the CRS NAND operation in crossbar arrays. We can also write and read any CRS cell in crossbar arrays using the write and read circuits (see Fig. 6). Not only CRS NAND operation can be performed, but also other CRS operations, such as CRS IMP, CRS NOR operations, and so on, can be implemented in the crossbar arrays.

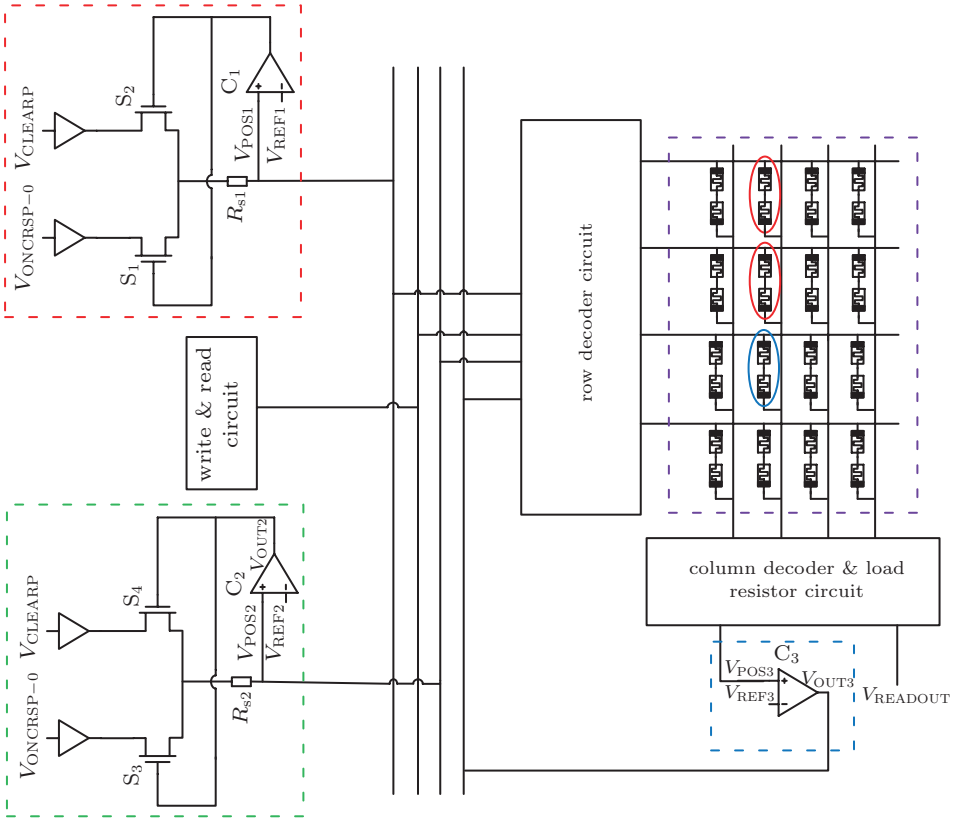


Fig. 6. The basic structure of CRS NAND operation in massive passive crossbar arrays using our proposed circuit.

3. Simulation results

The following presents the simulation results of CRS IMP and NAND operations using Hspice simulation tool. To perform the CRS stateful logic operations, the parameters of CRS cells should satisfy Eqs. (1)–(3), and (6). Here, for $\rm CRS_P$ and $\rm CRS_Q$, we choose $R_{\rm ON} = 100\ \Omega$ for state ON of memristive elements X and Y, and $R_{\rm OFF} = 10\ \rm k\Omega$ for state OFF of memristive elements X and Y. In this way, we can ensure the same configuration and parameter of $\rm CRS_P$ and $\rm CRS_Q$, which can be applied to massive crossbar arrays better. Moreover, we select $R_{\rm G} = 50\ \Omega$ and $R_{\rm S} = 20\ \Omega$. We use -12 V for $V_{\rm SETQ}$, 12 V for $V_{\rm CLEARP}$ and $V_{\rm CLEARQ}$. According to Eq. (1), $V_{\rm ONCRSP-0}$ is -8 V.

3.1. Simulation on CRS IMP operation

Figure 7 shows the CRS IMP operation simulation results. In Figs. 7(a) and 7(b), the states of $\rm CRS_P$ and $\rm CRS_Q$ are represented by two memristive elements which are called state PX, state QX (blue bold line) and state PY, state QY (green bold line), respectively. As shown in Fig. 7, there are three states for $\rm CRS_P$ and $\rm CRS_Q$. The first state is the original state. After Step 1, $\rm CRS_P$ and $\rm CRS_Q$ switch to the second state, and after Step 2, $\rm CRS_P$ and $\rm CRS_Q$ switch to the third state. Now, we take Case 1 for example. The original states of $\rm CRS_P$ and $\rm CRS_Q$ are state 0. After Step 1, $\rm CRS_P$ switches to state ON, while $\rm CRS_Q$ switches to state 1. After Step 2, $\rm CRS_P$ switches to state 0, while $\rm CRS_Q$ remains unchanged. After that, the state of $\rm CRS_Q$ is the result of IMP operation, and $\rm CRS_P$ remains in its original state, which is not destroyed. It is the same as the other three cases (see Fig. 7). The simulation results validate our proposed scheme.

3.2. Simulation on CRS NAND operation

Figure 8 shows the simulated results of the CRS NAND operation circuit (see Fig. 4). Like the CRS IMP operation, there are three states for $\rm CRS_{P1}$, $\rm CRS_{P2}$, and $\rm CRS_Q$ (see Fig. 8). The first states of $\rm CRS_{P1}$ and $\rm CRS_{P2}$ are the original states, and the first state of $\rm CRS_Q$ is the initialized state. During Step 1, $\rm CRS_Q$ is initialized to state 1. After Step 2, $\rm CRS_{P1}$, $\rm CRS_{P2}$, and $\rm CRS_Q$ switch to the second state, and after Step 3, $\rm CRS_{P1}$, $\rm CRS_{P2}$, and $\rm CRS_Q$ switch to the third state. Now, we take Case 1 for example. The original states of $\rm CRS_{P1}$ and $\rm CRS_{P2}$ are state 0. During Step 1, $\rm CRS_Q$ is initialized to state 1. After Step 2, $\rm CRS_{P1}$ and $\rm CRS_{P2}$ both switch to state ON, and $\rm CRS_Q$ remains in state 1. After Step 3, $\rm CRS_{P1}$ and $\rm CRS_{P2}$ both switch from state ON to state 0, and $\rm CRS_Q$ still remains in state 1. After that, the state of $\rm CRS_Q$ is the result of NAND operation, $\rm CRS_{P1}$ and $\rm CRS_{P2}$ remain in their original states, which are not destroyed. It is the same as the other three cases (see Fig. 8). The simulation results validate our proposed scheme.

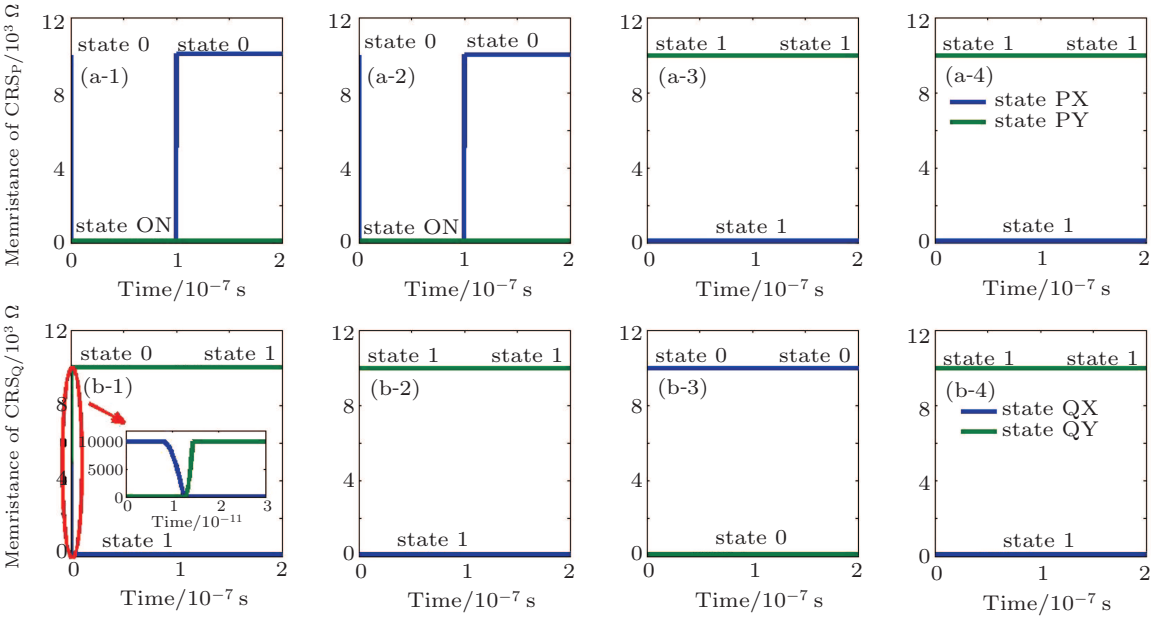


Fig. 7. The CRS IMP operation simulation results: (a) the state of CRS_P, and (b) the state of latched cell CRS_Q. (a-1), (b-1) Case 1; (a-2), (b-2) Case 2; (a-3), (b-3) Case 3; (a-4), (b-4) Case 4.

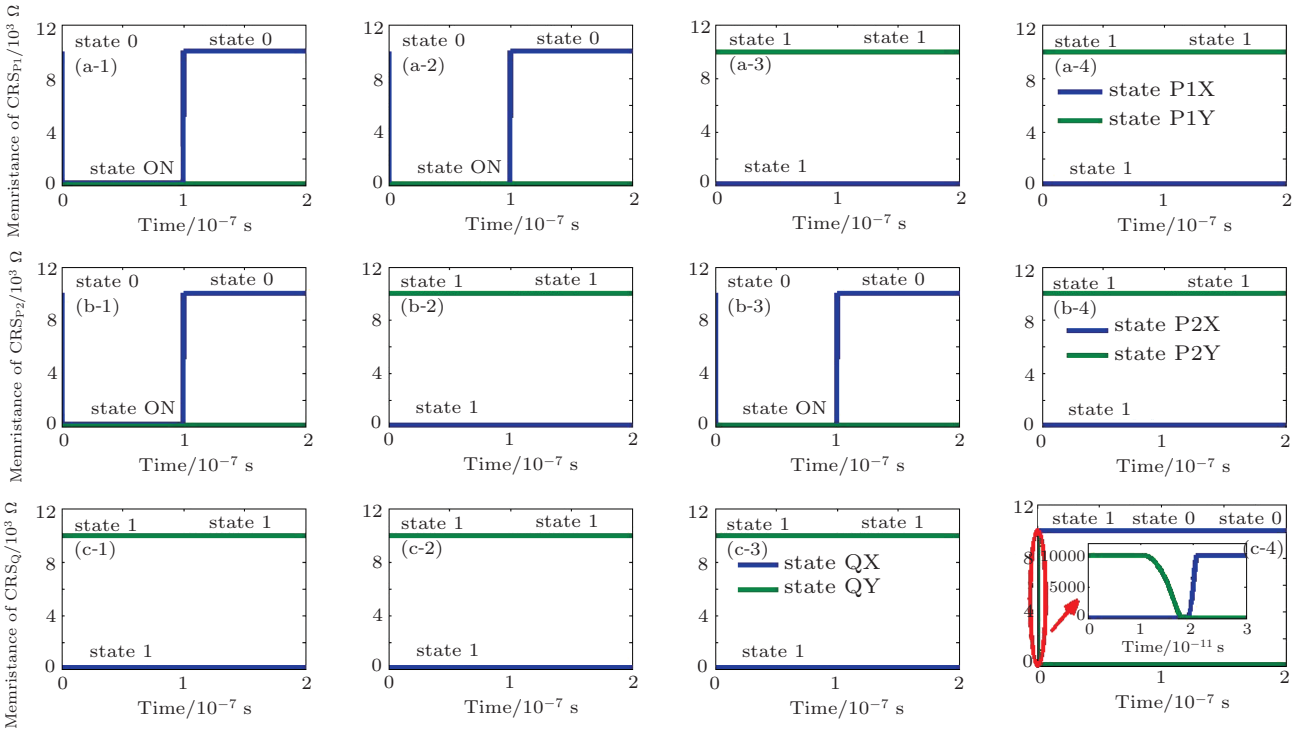


Fig. 8. The CRS NAND operation simulation results: (a) the state of CRS_{P1}, (b) the state of CRS_{P2}, and (c) the state of latched cell CRS_Q. (a-1), (b-1), (c-1) Case 1; (a-2), (b-2), (c-2) Case 2; (a-3), (b-3), (c-3) Case 3; (a-4), (b-4), (c-4) Case 4.

4. Conclusion

In this paper, we present a circuit design for CRS-based stateful logic operations. Here, we describe CRS IMP and NAND operations using our proposed circuit in detail. We only need two steps to implement the CRS IMP operation compared with the previous CRS IMP operation in Ref. [15]. Furthermore, only three steps are required for implementing N inputs NAND gate, whereas memristor based NAND operation needs $N+1$ steps. The most important feature of our proposed circuit is that the states stored in input cells will not be damaged. In the end, due to the fact that we use the same type of CRS as input and output cells, we can apply the circuit to massive passive crossbar arrays.

References

[1] Kuhn K J 2012 *IEEE Trans. Electron Device* **59** 1813

[2] Chua L O 1971 *IEEE Trans. Circuit Theory* **18** 507
[3] Strukov D B, Snider G S, Stewart G R and Williams R S 2008 *Nature* **453** 80
[4] Fan X, Chen H P, Wang Q, Wang Y Q, Lv S L, Liu Y, Song Z T, Feng G M and Liu B 2015 *Chin. Phys. Lett.* **6** 068301
[5] Tang S Y, Li R, Ou X, Xu H N, Xia Y D, Yin J and Liu Z G 2014 *Chin. Phys. Lett.* **7** 078503
[6] Shin S, Kim K and Kang S M 2011 *IEEE Trans. Nanotechnol.* **10** 266
[7] Pershin Y V and Ventra M D 2010 *IEEE Trans. Circuits Syst. I* **57** 1857
[8] Tian X B and Xu H 2013 *Chin. Phys. B* **22** 088501
[9] Li Z J and Zeng Y C 2013 *Chin. Phys. B* **22** 040502
[10] Yuan F, Wang G Y and Wang X Y 2015 *Chin. Phys. B* **24** 60506
[11] Buscarino A, Fortuna L, Frasca M and Gambuzza L V 2012 *An Interdisciplinary Journal of Nonlinear Science* **22** 023136
[12] Pershin Y V and Ventra M D 2010 *Neural Netw.* **23** 881
[13] Borghetti J, Snider G S, Kuekes P J, Yang J J, Stewart D R and Williams R S 2010 *Nature* **464** 873
[14] Shin S, Kim K and Kang S M 2011 *IEEE Trans. Circuits Syst. II* **58** 442
[15] Yang Y, Mathew J, Pradhan D K, Ottavi M and Pontarelli S 2014 *Design, Automation and Test in Europe Conference and Exhibition*, pp. 1–4
[16] Zhu X, Yang X, Wu C, Xiao N, Wu J and Yi X 2013 *IEEE Trans. Circuits Syst. II* **60** 682
[17] Kvatinsky S, Satat G, Wald N, Friedman E G, Kolodny A and Weiser U C 2014 *IEEE Trans. Very Large Scale Integr. Syst. (VLSI)* **22** 2054
[18] Linn E, Rosezin R, Tappertzhofen S, Bottger U and Waser R 2012 *Nanotechnology* **23** 305205
[19] Xia Q, Robinett, Cumbie M W, Banerjee N, Cardinali T J, Yang J J, Wu W, Li X, Tong W M and Strukov D B 2009 *Nano Lett.* **9** 3640
[20] Ho Y, Huang G M and Li P 2010 *IEEE Trans. Circuits Syst. I* **58** 724
[21] Jung C M, Choi J M and Min K S 2012 *IEEE Trans. Nanotechnol.* **11** 611
[22] Zidan M A, Eltawil A M, Kurdahi F, Fahmy H and Salama K N 2014 *IEEE Trans. Nanotechnol.* **13** 274
[23] Zidan M A, Omran H, Sultan A, Fahmy H and Salama K N 2015 *IEEE Trans. Nanotechnol.* **14** 3
[24] Linn E, Rosezin R, Kugeler C and Waser R 2010 *Nat. Mater.* **9** 403