The design and simulation of a titanium oxide memristor-based programmable analog filter in a simulation program with integrated circuit emphasis*

Tian Xiao-Bo(田晓波)† and Xu Hui(徐 晖)

Embedded System and Solid-State Engineering Technology Center, School of Electronic Science and Engineering, National University of Defense and Technology, Changsha 410073, *China*

(Received 31 October 2012; revised manuscript received 22 December 2012)

In many communication and signal routing applications, it is desirable to have a programmable analog filter. According to this practical demand, we consider the titanium oxide memristor, which is a kind of nano-scale electron device with low power dissipation and nonvolatile memory. Such characteristics could be suitable for designing the desired filter. However, both the non-analytical relation between the memristance and the charges that pass through it, and the changeable V–I characteristics in physical tests make it difficult to accurately set the memristance to the target value. In this paper, the conductive mechanism of the memristor is analyzed, a method of continuously programming the memristance is proposed and simulated in a simulation program with integrated circuit emphasis, and its feasibility and compatibility, both in simulations and physical realizations, are demonstrated. This method is then utilized in a first-order active filter as an example to show its applications in programmable filters. This work also provides a practical tool for utilizing memristors as resistance programmable devices.

Keywords: memristor, programmable filter, dopant drift, SPICE

PACS: 85.35.–p, 84.30.Vn, 87.85.Qr, 89.20.Ff **DOI:** 10.1088/1674-1056/22/8/088501

1. Introduction

Since the research and development of traditional electronic devices are gradually moving towards maturity, there is an urgent need for new techniques and materials for the higher performances of modern circuits, in terms of faster speed, lower power dissipation and cost, smaller geometric size, higher accuracy and reliability, etc. To realize these requirements, nano-electronics is the best option.[1] The titanium oxide memristor is a kind of typical nano-scale electron device with great proven potential in areas such as resistive random access memory (RRAM),[2] cellular neural networks (CNNs),[3] digital logical operation,[4] programmable analog applications,[5] and chaotic circuit analysis,[6,7] and will be widely utilized in the near future.

The memristor was first predicted as a basic circuit element by Chua in 1971.[8] It is defined by the relation between the charge quantity $q(t)$ and magnetic flux $\phi(t)$, and one of its theoretical characteristics is that it can remember the last state after the outer electrical bias has been cut off. Chua and Kang[9] later extended the memristor concept to a broader memristive system. In 2008 the first nano-scale titanium oxide memristor with a metal/oxide/metal structure was fabricated in a HP laboratory.[10] Since then, research around the characteristics, simulations, and applications of memristors have become hot issues in circuit research. Song *et al.*[11] studied the basic properties and applications of the memristor circuit. Zhang *et al.*[12] looked at the mathematical model and the properties of memristors with border constraint, whereas Bao *et al.*[13] investigated the initial state-dependent dynamical behaviors of the memristor-based chaotic circuit and transient chaos in a smooth memristor oscillator.[14]

In many communication and signal routing applications, such as material identification, it is desirable to have an analog filter. This should be composed of more than one single filter that only signals in a certain frequency range may pass through.[15] Moreover, the analog filter should have a small geometric size, dissipate a low amount of power, and be easily programmable. The nano-scale memristor has nonvolatile memory, and works at low current and low voltage, thus consuming a small amount of energy. Moreover, for memristive memory cells, the switching time can be in the sub-nanosecond range.[16] All the characteristics mentioned above make the memristor a potential candidate for realizing the desired programmable analog filter. However, there has been no detailed realization of such circuits in the existing research. Moreover, the nonlinear relations between the model variables and changeable conductive characteristics in real experimental tests block the simulations and applications of the memristor as a resistance-programmable device.

By using the memristive properties of vanadium dioxide, Driscoll *et al.*[17] experimentally demonstrated an adaptive filter by placing a memristor into an LC contour. They found that this circuit reacts to the application of select frequency signals

*Project supported by the National Natural Science Foundation of China (Grant Nos. 61171017 and F010505).

†Corresponding author. E-mail: txiaobo1985@gmail.com

© 2013 Chinese Physical Society and IOP Publishing Ltd

by sharpening the quality factor of its resonant response, and thus "learns" according to the input waveform. Strachan *et al.*[15] proposed a programmable analog filter that includes a crossbar array with a number of memristive junction elements. The research in Ref. [18] shows the passive filter characteristics in an RC circuit using a piecewise linear model. Fang *et al.*[19] proposed simulation program with integrated circuit emphasis (SPICE) modeling memristors with multilevel resistance states based on memristance versus a state map. However, there has been no detailed realization of memristor-based programmable filters that can be accurately controlled by setting the memristance to target values.

In this paper, compared with the physical experimental V–I results, the conductive mechanism of the memristor is analyzed and a voltage comparator-based memristance programming and reading circuit, which is feasible and compatible both in the model and in practical realizations, is designed. It is then tested in a first-order active filter as an example to demonstrate the application of the proposed circuit. The work also provides a method of continuously and accurately setting the memristance in practice.

2. The conductive mechanism of the titanium oxide memristor

Theoretically, the memristor can remember its history in terms of memristance $M(q)$. It can be described with magnetic flux $\varphi(t)$ and charge quantity $q(t)$ as follows:

for a charge-controlled memristor

$$\varphi = f(q), \tag{1}$$

for a flux-controlled memristor

$$q = f(\varphi). \tag{2}$$

If we take the charge-controlled memristor as an example, we could differentiate Eq. (1) and obtain

$$\frac{\mathrm{d}\varphi}{\mathrm{d}t} = \frac{\mathrm{d}f(q)}{\mathrm{d}q}\frac{\mathrm{d}q}{\mathrm{d}t}. \tag{3}$$

According to the definitions of $\varphi(t)$ and $q(t)$, we could transform Eq. (3) into

$$v(t) = M(q)i(t), \tag{4}$$

where $v(t)$ is the externally applied voltage. It is obvious that $M(q)$ is in units of Ohm and could be regarded as the transient resistance. In Eq. (4), the memristance depends on the charge quantity $q(t)$ that passes through the memristor. The charge quantity $q(t)$ is the time integral of current $i(t)$, and records the history of the device, therefore, the memristor can remember its state.

One typical titanium oxide memristor mainly consists of four parts: top and bottom platinum electrodes (TE and BE); one layer of titanium oxide with oxygen vacancies, which is regarded as the doped layer (DL) with dopants; and another layer of stoichiometric titanium oxide, meaning that the ratio of oxygen to titanium is perfectly stoichiometric at exactly two to one,[20] and is regarded as the undoped layer (UDL) as shown in Fig. 1.

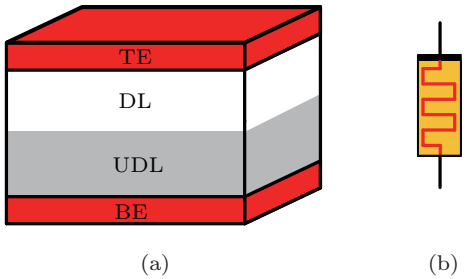


Fig. 1. (color online) The schematic structure and circuit symbol of a titanium oxide memristor, showing (a) the device structure and (b) the circuit symbol.

The microscopic nature of resistance switching and charge transport in memristive devices is still under debate, but one proposal is that the hysteresis requires some sort of atomic rearrangement that modulates the electronic current.[10] On the basis of this proposition and the physical experiments in Ref. [21], the HP laboratory explained the working principle of the memristor using the dopant drift mechanism.

2.1. Dopant drift-based mechanism

As shown in Fig. 1(a), the UDL consists of stoichiometric titanium oxide (TiO_2), and the ratio of oxygen to titanium is 2:1. In the UDL, each titanium atom corresponds to two oxygen atoms and a stable electrovalent bond forms between them. There are hardly any free charged ions moving in the inner part of the UDL, as shown in Fig. 2(a), thus this could be regarded as an insulator.

By contrast, the DL consists of titanium oxide that is missing a certain percentage of oxygen (TiO_{2-X}). The proportion of missing oxygen can be controlled in the process of oxidation reaction by adding different oxygen flows to the titanium target. The missing oxygen atoms form vacancies that are not stable and could easily transfer between the adjacent titanium atoms. The vacancies will move in the same direction when an outer voltage $v(t)$ is applied to the DL, as shown in Figs. 2(b)–2(d). The hollow circles represent the oxygen vacancies that could be regarded as +2 charge dopants. They can greatly improve the electrical conductivity of the titanium oxide, and thus TiO_{2-X} has a much lower resistance than TiO_2 with the same geometric structure. The up and down arrows represent the moving directions of the oxygen atoms and oxygen vacancies, respectively.

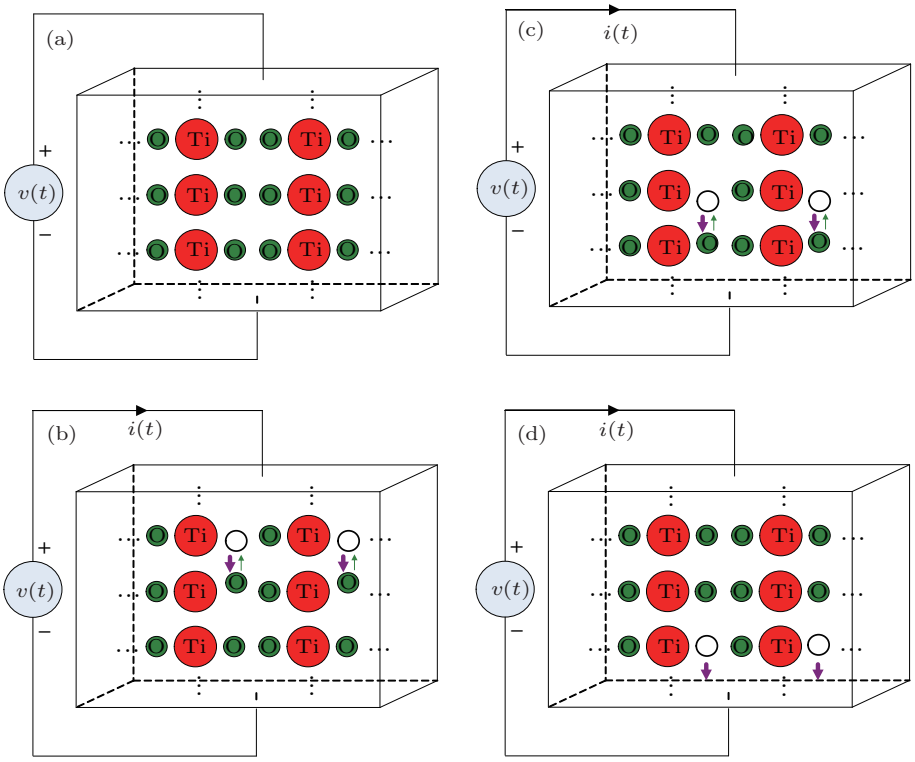


Fig. 2. (color online) The schematic structures of (a) the UDL and ((b)–(d)) the DL of the memristor and the responses to the outer voltage bias.

Since the oxygen vacancies catch positive charges that will move towards the negative polarity of the outer electrical field when $v(t)$ is applied to the memristor, part of the stoichiometric titanium oxide therefore begins to be mixed with oxygen vacancies, and thus the width of DL increases and the boundary between the two layers drifts downwards, as shown in Fig. 3(a). $w(t)$ and D are the width of the DL and the whole titanium oxide, respectively. On the contrary, figure 3(b) shows the reversed process. Theoretically, the boundary could move to the end of the device, which is demonstrated in Figs. 3(c) and 3(d).

The boundary stops moving when $v(t)$ has been cut off and remains stable until the next outer bias is connected to the memristor; thus it has non-volatile memory.

The resistance of a given material R depends on the length l, the cross-sectional area s, and the electrical resistivity ρ. It could be expressed as follows:

$$R = \rho \frac{l}{s}. \tag{5}$$

It is obvious that R is proportional to l once the material type and cross section are fixed. On the basis of the dopant drift mechanism, one memristor could be regarded as two variable resistors connected in series, and the equivalent circuit is shown in Fig. 4.

According to Ohm's law:

$$R_{\mathrm{DL}} = R_{\mathrm{ON}} \frac{w(t)}{D}, \tag{6}$$

$$R_{\mathrm{UDL}} = R_{\mathrm{OFF}} \left(1 - \frac{w(t)}{D}\right), \tag{7}$$

$$v(t) = (R_{\mathrm{DL}} + R_{\mathrm{UDL}}) i(t). \tag{8}$$

For the simplest case of Ohmic electronic conduction and linear ionic drift in a uniform field with average ion mobility μ_{v}, it is found that[10]

$$\frac{\mathrm{d}w(t)}{\mathrm{d}t} = \mu_{\mathrm{v}} \frac{R_{\mathrm{ON}}}{D} i(t). \tag{9}$$

Let the initial value of $w(t)$ be w_0, and the combination of Eqs. (8) and (9) yields an analytical relation between $M(q)$ and $q(t)$, as presented as follows:

$$M(q) = \left(\frac{w_0}{D} R_{\mathrm{ON}} + \left(1 - \frac{w_0}{D}\right) R_{\mathrm{OFF}}\right) - \frac{q(t)}{Q_0} (R_{\mathrm{OFF}} - R_{\mathrm{ON}}), \tag{10}$$

where $Q_0 = D^2/(\mu_{\mathrm{v}} R_{\mathrm{ON}})$ is a constant. Such a relation is consistent with the theoretical definition of the memristor that is discussed in Eq. (4).

In nanoscale devices, small voltages can yield enormous electric fields, which in turn can produce significant nonlinearities in ionic transport.[10] Linear dopant drift is a kind of ideal conductive mechanism that excludes the significant nonlinearity in ionic transport and only gives a fundamental explanation of the conductive behavior of the memristor. The real case is where the inner electrical field causes the boundary to drift nonlinearly. By adding window functions, such as,

$$F(w) = \frac{w(D-w)}{D^2}, \tag{11}$$

and

$$F(w) = 1 - \left(2\frac{w}{D} - 1\right)^{2p} \tag{12}$$

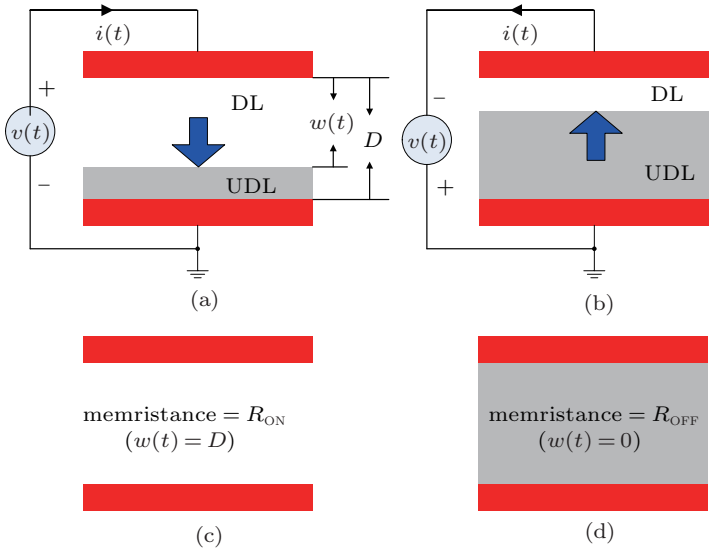


Fig. 3. (color online) The memristance changes as the boundary between the two layers drifts. Panel (a) shows that the positive outer bias drives the boundary downwards, panel (b) indicates that the negative outer bias drives the boundary upwards, panel (c) shows that the memristance equals $R_{\rm ON}$ when $w(t) = D$, and panel (d) displays that the memristance equals $R_{\rm OFF}$ when $w(t) = 0$.

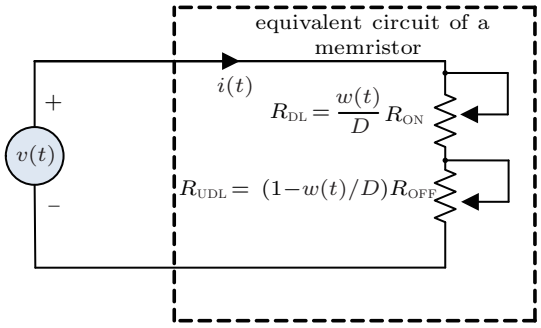


Fig. 4. (color online) The equivalent circuit of a memristor in the case of dopant drift.

to the velocity of the boundary drift, nonlinear dopant drift could be represented by

$$\frac{{\rm d}w(t)}{{\rm d}t} = F(w)\mu_{\rm v}\frac{R_{\rm ON}}{D}i(t). \tag{13}$$

In this case, the combination of Eqs. (8), (11), and (13) or Eqs. (8), (12), and (13) can no longer yield an analytical relation between $M(q)$ and $q(t)$. The simulation result is shown in Fig. 5(a). However, the experimental results could be much more complex. As shown in Fig. 5(b), the single memristor V–I curve indicates that asymmetrical and non-ideal conductive behaviors may exist. This can no longer be described accurately by an analytical relation, but approximately by a window function. In Fig. 5(c), a set of V–I curves are the test results of the same memristor under identical conditions, but they do not coincide with each other. A similar phenomenon appears in the results shown in Fig. 5(d), so it is not a casual case.

2.2. The tunnel barrier-based mechanism

According to the experimental results in Ref. [22], Yang *et al.*[23] proposed another explanation to describe the conductive behaviors of titanium oxide memristors. As shown in Fig. 6(a), a thin film of titanium dioxide (TiO_2) is sandwiched between two platinum electrodes (BE and TE). After an electroforming process,[24] which is a type of method that produces oxygen vacancies and improves the electrical conductivity, TiO_{2-X} are formed, which results in a highly conductive channel that shunts most of the oxide film except for a narrow tunnel barrier. The tunnel barrier width can be modulated by applying a voltage across the memristor.

In this case, the tunnel barrier plays a major role in determining the conductive behavior of the memristor; its width is time-varying and has been modelled according to the Simmons I–V expression for a rectangular barrier in Ref. [22].

Although some physical mechanisms, such as dopant drift and a tunnel barrier, have been proposed to explain the conductive characteristics of the memristor, there is still the question of which one or ones are the main factors. It is possible and reasonable that several different mechanisms may co-exist and work together, and different mechanisms could be dominant in different material systems.[23]

Theoretically, nonlinear dopant drift and the tunnel barrier mechanism are useful for explaining the conductive behaviors and designing memristor-based applications, but there have been no analytical relations between $M(q)$ and $q(t)$ in these cases. Moreover, it is hard to accurately set $M(q)$ to any target values by utilizing the theoretical characteristics in physical realizations since a memristor may become unstable in the repeating loops. Therefore, a method of accurately controlling the memristance is desirable both in the circuit simulations and the real applications.

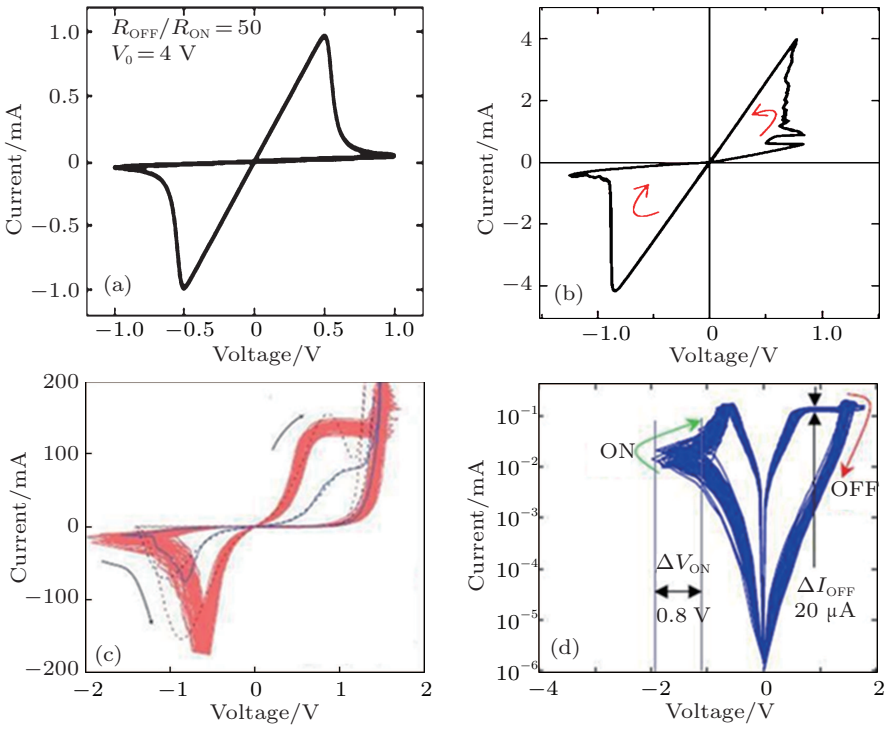


Fig. 5. (color online) The simulation results and experimental data of titanium oxide memristors with nonlinear dopant drift. Panel (a) shows the simulation results of the memristor with nonlinear dopant drift published in Ref. [10], and panels (b)–(d) display the experimental V–I curves of the memristors published in Refs. [21], [23], and [24], respectively.

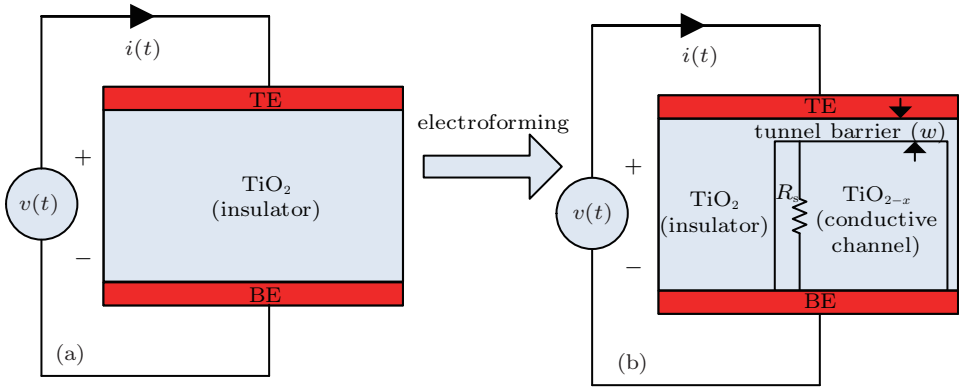


Fig. 6. (color online) Schematic diagram of the electroforming process for (a) the memristor before electroforming and (b) the memristor after electroforming.

3. Continuous memristance programming

A method to utilize the memristor as a multilevel memory was proposed in Ref. [25] and provides a useful way to change the memristance. The programming of the memristance is performed by applying additional current pulses to the memristor in the appropriate directions until the voltage of the memristor equals that of the selected node voltage in the resistance array. However, since the number of resistors in the resistance array is limited, the memristance cannot be changed continuously, only finite values can be selected. Moreover, once the structure of a resistance array is fixed, it is lacking in flexibility and hard to change. Compared with the above-mentioned method, a circuit for continuous memristance programming and reading is proposed and simulated in SPICE.

3.1. A basic circuit for memristance programming and reading

Though the conductive mechanism and non-analytical relation of memristors are complex, the memristance can be changed continuously. This is difficult to be directly tested and modified, so we adopt another method by measuring the node voltage in a simple circuit, which is much more feasible. Based on such a consideration, a basic circuit for programming and reading the memristance is proposed and shown in Fig. 7(a), where $V(t)$ is an input voltage connected at node 'a', which could be the programming signal or the reading one. $V_{\rm ref}(t)$ is the output voltage at node 'b', and node 'c' is connected to the ground.

According to Ohm's law, we have

$$V(t) = (M(q) + R_{\rm ref})i(t), \tag{14}$$

$$V_{\rm ref}(t) = R_{\rm ref} i(t). \tag{15}$$

The combination of Eqs. (14) and (15) leads to a one-to-one relation between $M(q)$ and $V_{\rm ref}(t)$

$$M(q) = R_{\rm ref}\left(\frac{V(t)}{V_{\rm ref}(t)} - 1\right). \tag{16}$$

According to Eqs. (9) and (13), the increase or decrease in the memristance depends on the polarity of the outer bias, so we utilize a positive voltage (PV) and a negative voltage (NV) as the programming signals, as shown in Fig. 7(b). Both of these have an amplitude of v_0. Programming is performed by applying PV or NV to the circuit. The reading voltage (RV) has a much smaller amplitude v_1, $v_1 \ll v_0$, since the weak signals hardly change the memristance. Besides, the RV consists of a positive and negative pulse with the same amplitude, and thus the influences on the memristance cancel each other out. Reading is performed by applying RV to the circuit.

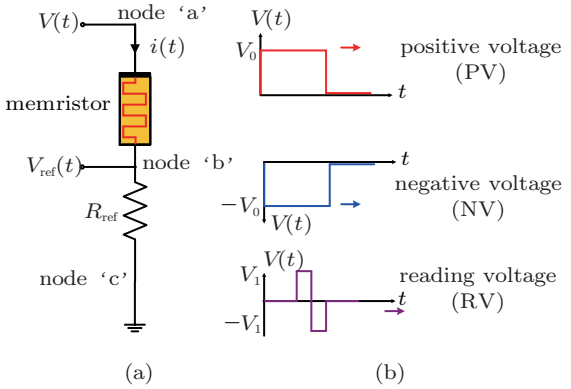


Fig. 7. (color online) The basic circuit for programming and reading the memristance of a memristor. Panel (a) shows the basic circuit, and panel (b) displays the programming and reading signals.

The proposed basic circuit is simulated in SPICE and the results are given in Fig. 8. The simulation time span $T = 1$ s and maximum time step $\Delta T = 1$ ms. The values for the parameters of the memristor in the simulations are as follows: $\mu_{\rm v} = 10^{-14}$ m$^2 \cdot$V$^{-1} \cdot$S^{-1}, $R_{\rm ON} = 100\ \Omega$, $R_{\rm OFF} = 5$ kΩ, and $D = 10$ nm, which are consistent with those already published in Ref. [10]. $R_{\rm ref} = 1$ kΩ. We adopt dopant drift-based theory as the conductive mechanism and Eq. (11) as the window function that defines the conductive behavior of the memristor. The version of our SPICE is PSPICE Plugin 10.5.0.a001.

In the case of dopant drift-based conductive processes, the simulation results shown in Fig. 8 prove the feasibility of the proposed basic circuit. The memristance could change continuously to any target values within the range of ($R_{\rm ON}$, $R_{\rm OFF}$), since the memristance has a one-to-one relation to $V_{\rm ref}(t)$ in the basic circuit. In the case of reading, a small voltage amplitude induces the memristance to change from 2550 Ω to 2546.5 Ω, which is a negligible difference, and it comes back to the initial value after reading has finished. Thus, RV is suitable for memristance detection.

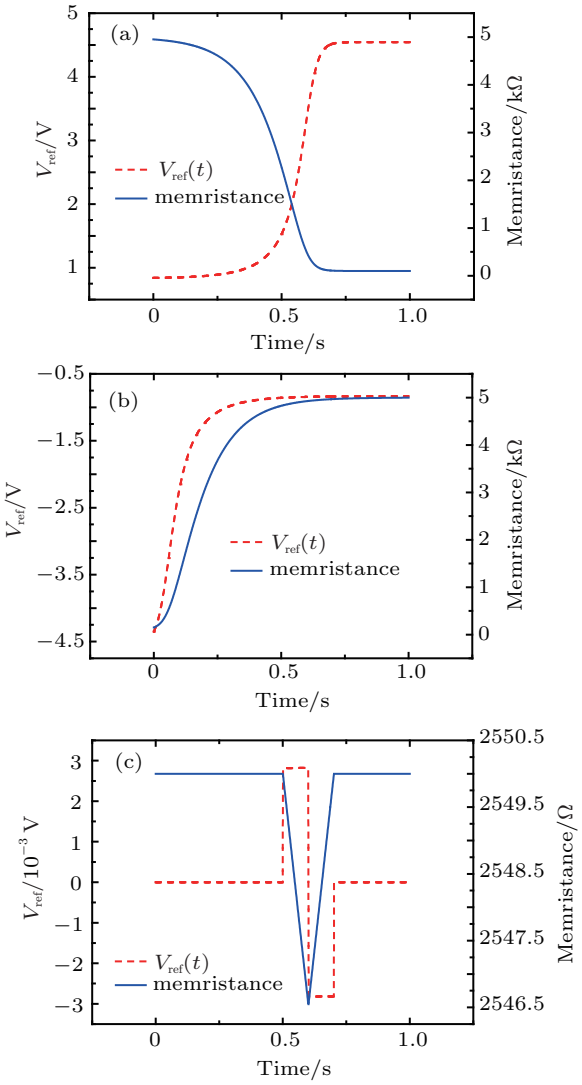


Fig. 8. (color online) The simulation results of the basic circuit for programming and reading the memristance. Panel (a) shows the positive memristance programming $w_0 = 0.1$ nm, $V(t) = 5.0$ V, panel (b) displays the negative memristance programming $w_0 = 9.9$ nm, $V(t) = -5.0$ V, and panel (c) exhibits the memristance reading $w_0 = 5.0$ nm. $V(t)$ is a voltage with a positive pulse followed by a negative one with the same amplitude and time span.

3.2. The continuous memristance programming circuit

Since the memristance has a one-to-one relation to the reference voltage $V_{\rm ref}$ in the basic circuit designed in Fig. 7, the memristance could be set to target value by controlling $V_{\rm ref}(t)$, which is relatively feasible and simple. Based on such a consideration, a voltage comparator-based memristance programming and reading circuit is proposed and shown in Fig. 9.

The reference voltage is an indicator of the memristance, so a feedback circuit is needed for the real-time detection of the reference voltage and to provide control signals to begin or stop the programming process. An operational amplifier (OPAMP) and a voltage-controlled analog switch $S_{\rm ctrl}$ are adopted as the feedback circuit part.

Let the absolute value of the output voltage of the open-loop OPAMP be $V_{\rm m}$, then the relation between input $V_{\rm in+}$, $V_{\rm in-}$,

and output $V_{\rm out}$ will be as follows:

$$V_{\rm out} = \begin{cases} V_{\rm m}, & (V_{\rm in+} > V_{\rm in-}), \\ -V_{\rm m}, & (V_{\rm in+} < V_{\rm in-}), \\ 0, & (V_{\rm in+} = V_{\rm in-}). \end{cases} \tag{17}$$

There are two inputs (nodes '3' and '4') and two outputs (nodes '1' and '2') in $S_{\rm ctrl}$. The voltage between nodes '3' and '4' is the control signal, and the circuit between nodes '1' and '2' is the switch part. The resistance between nodes '1' and '2' is denoted by $R_{\rm S}$. If the switch is on, $R_{\rm S}$ equals a rather low resistance $R_{\rm L}$, otherwise, $R_{\rm S}$ equals a much higher one, $R_{\rm H}$. If we suppose that the threshold voltage is $V_{\rm th}$, then we will have

$$R_{\rm S} = \begin{cases} R_{\rm L}, & (V(3,4) > V_{\rm th}), \\ R_{\rm H}, & (V(3,4) \le V_{\rm th}). \end{cases} \tag{18}$$

The working process of the designed circuit is as follows. S is a switch that can select node 'b' or 'c'. Once node 'b' is connected, the circuit is in the programming mode, the input voltage $V(t)$ provides PV or NV for the memristor, and $V_{\rm in-}$ gradually increases or decreases as $V(t)$ is applied to the memristor. $V_{\rm ref}(t)$ is pre-defined as the required value according to Eq. (17) so that $V_{\rm out}$ reaches zero when $V_{\rm in-}(t)$ equals $V_{\rm ref}(t)$. At the same time, $S_{\rm ctrl}$ is closed and the programming voltage is cut off, so the memristance will remain unchanged. Once node 'c' is connected, the circuit is in a reading mode, and $V(t)$ provides RV for the memristor. The memristance will remain unchanged and be read out as $V_{\rm in-}$, which guides the programming in turn.

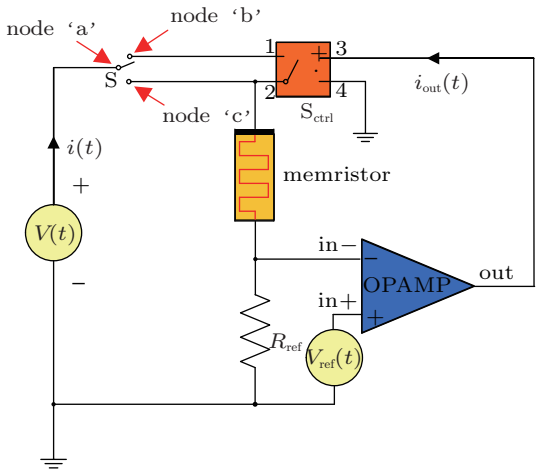


Fig. 9. (color online) A voltage comparator-based memristance programming and reading circuit.

The memristance programming processes are simulated in SPICE to prove the feasibility and accuracy of the proposed circuit. The target memristance values are set to be 0.5 kΩ and 3 kΩ in the positive and negative programming modes, respectively. The values for the parameters in the simulations are as follows: time span $T = 100$ ms, maximum time step $\Delta T = 10$ μs, $V_{\rm th} = 0$ V, $V_{\rm m} = 15$ V, $R_{\rm ref} = 1$ kΩ, $R_{\rm L} = 1$ Ω, and $R_{\rm H} = 1$ MΩ. The dopant drift-based mechanism is adopted and the parameter values of the memristor are the same as those used in the basic circuit simulations.

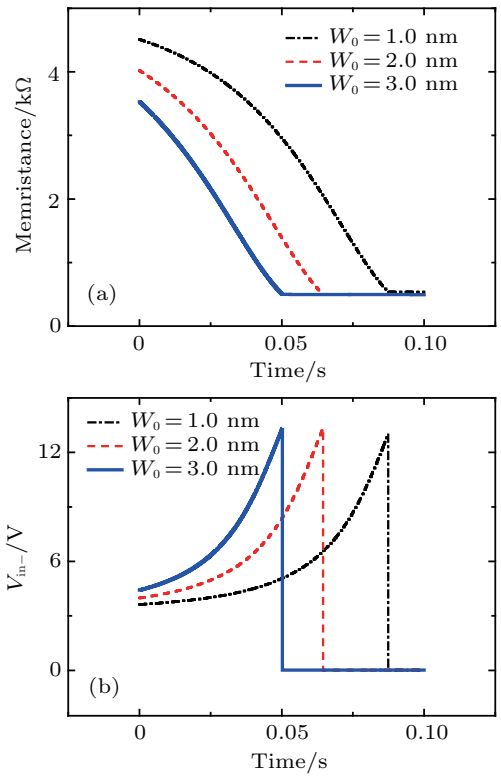


Fig. 10. (color online) The simulation results of memristance programming using PV, $V(t) = 20$ V, $V_{\rm ref} = 13.33$ V. Panel (a) shows the memristances varying with time, and panel (b) exhibits the voltages at node 'in-' hanging with time.

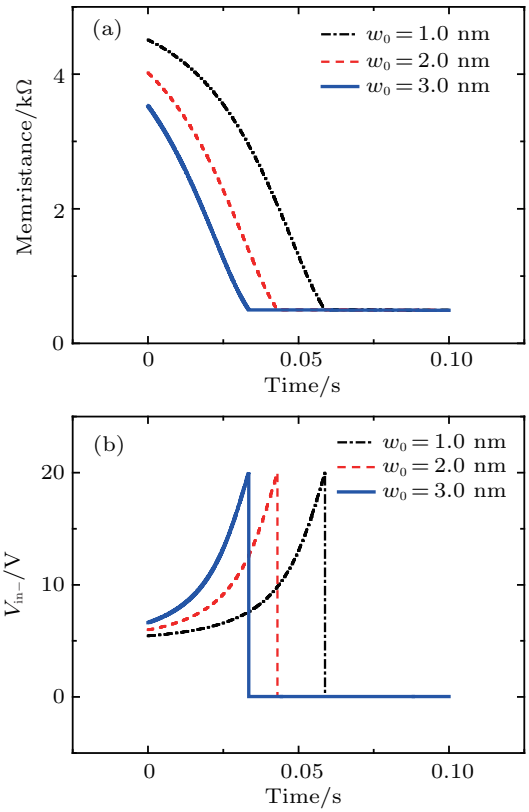


Fig. 11. (color online) The simulation results of memristance programming using PV, $V(t) = 30$ V, $V_{\rm ref} = 20$ V. Panel (a) shows the memristances varying with time, and panel (b) exhibits the voltages at node 'in-' changing with time.

We simulated four cases of memristance programming and give the results in Figs. 10–13. In each simulation, three different initial values of the DL width are adopted for comparison. The memristance is successfully set to be the pre-defined values, but the times consumed are not the same. Tables 1–4 show the variations in memristance (ΔM) between the initial and target value with the time consumed for programming in each simulation. In the case of the same input voltage, the analysis shows that it takes a shorter time for the memristance to change to the target value when the difference between the initial DL width and the target one is smaller, and *vice versa*. In another case of the same initial DL width, we conclude that it takes longer for a lower $V(t)$ to achieve programming and *vice versa*. All the parameters adopted in the simulations are consistent with those already published, and the results prove the feasibility and accuracy of the designed circuit.

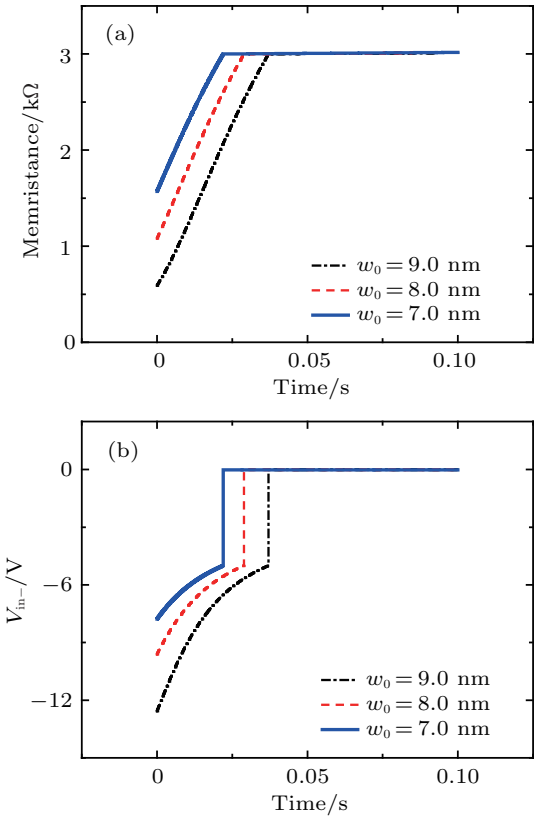


Fig. 12. (color online) The simulation results of memristance programming using NV, $V(t) = -20$ V, $V_{\rm ref} = -5$ V. Panel (a) shows the memristances varying with time, and panel (b) displays the voltages at node 'in-' changing with time.

In the case of practical applications of memristors, the non-ideal V–I characteristics of memristors are a group of similar curves that may not coincide with each other, which increases the degree of difficulty for memristance programming. Moreover, there have been no accurate models that simulate the complex conductive behaviors of physical memristors. However, the proposed method could be valid in such cases, since it avoids dealing directly with non-ideal or non-analytical relations between the memristance and outer bias either in real memristors or in models, but utilizes a relatively easy to control memristance programming circuit instead.

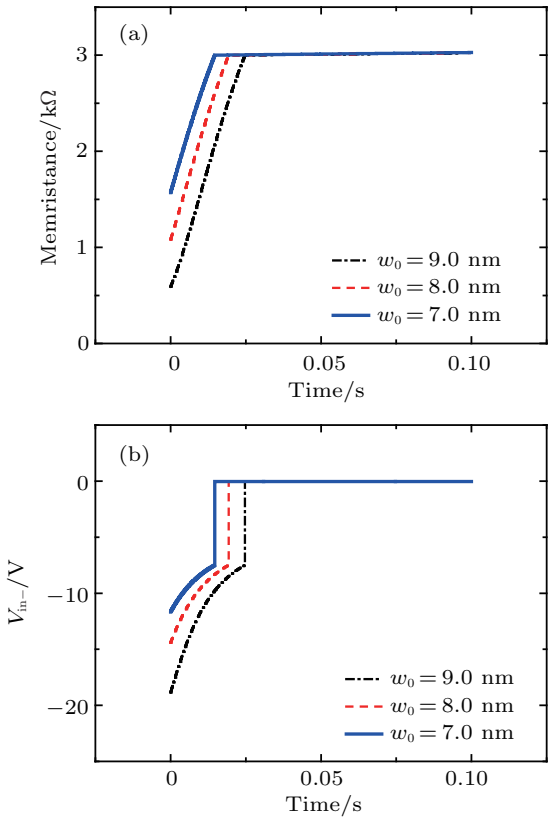


Fig. 13. (color online) The simulation results of memristance programming using NV, $V(t) = -30$ V, $V_{\rm ref} = -7.5$ V. Panel (a) shows the memristances varying with time, and panel (b) exhibits the voltages at node 'in-' varying with time.

Table 1. The time consumed in the simulations in Fig. 10.

	$w_0 = 1.0$ nm	$w_0 = 2.0$ nm	$w_0 = 3.0$ nm
$\Delta M/\Omega$	4010	4020	3530
Time/ms	88.2	64.5	50.2

Table 2. The time consumed in the simulations in Fig. 11.

	$w_0 = 1.0$ nm	$w_0 = 2.0$ nm	$w_0 = 3.0$ nm
$\Delta M/\Omega$	4010	4020	3530
Time/ms	58.8	43.0	33.5

Table 3. The time consumed in the simulations in Fig. 12.

	$w_0 = 9.0$ nm	$w_0 = 8.0$ nm	$w_0 = 7.0$ nm
$\Delta M/\Omega$	2410	1920	1430
Time/ms	37.1	28.9	22.0

Table 4. The time consumed in the simulations in Fig. 13.

	$w_0 = 9.0$ nm	$w_0 = 8.0$ nm	$w_0 = 7.0$ nm
$\Delta M/\Omega$	2410	1920	1430
Time/ms	24.7	19.3	14.7

4. Memristor-based programmable analog filters

Analog filters are basic building blocks of signal processing, and widely used in electronics. Traditional analog

filters have unchangeable parameters and characteristics once the circuit structure is fixed, which restricts many applications, such as material identifications that need a frequent change in the filter frequency. By utilizing the programming circuit designed in this paper, such a problem is solved. The memristance of a memristor could be changed to any desired values in the range between $R_{\rm ON}$ and $R_{\rm OFF}$. We use such a part as a controllable resistance in analog filters and change the parameters, such as cutoff frequency, by programming the memristance. An example of a memristor-based programmable analog filter is given in Fig. 14.

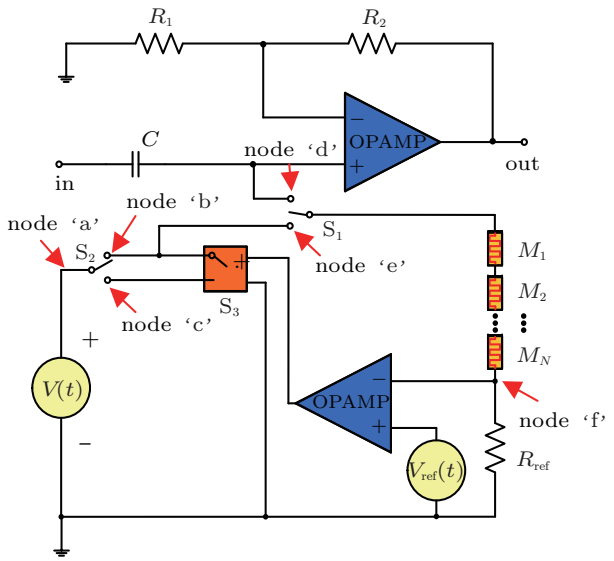


Fig. 14. (color online) An example of a memristor-based programmable high-pass filter.

The example filter shown in Fig. 14 consists of a typical first-order RC active filter, where the resistor is replaced by a memristance programming circuit. The filter is in the programming mode when switch S_1 is connected to node 'e' and S_2 is connected to node 'c'. In this case, several memristors form an array for the purpose of broadening the range of its memristance, and the number N of the memristors depends on the practical need. Theoretically, the memristance range of such an array could be extended to the range between $NR_{\rm ON}$ and $NR_{\rm OFF}$. Therefore, the cutoff frequency $f_{\rm C}$ of the filter shown in Fig. 14 will be in a range as follows:

$$\frac{1}{2\pi(R_{\rm ref}+NR_{\rm OFF})C} < f_{\rm C} < \frac{1}{2\pi(R_{\rm ref}+NR_{\rm ON})C}. \qquad (19)$$

The filter is in the reading mode when S_1 is connected to node 'e' and S_2 is connected to node 'b'. In this case, $V(t)$ provides the memristor array with reading signals, and the voltage at node 'f' indicates the resistance of the memristor array according to Eq. (16).

The circuit is in the filtering mode when S_1 is connected to node 'd'. In this case the series circuit, which consists of $R_{\rm ref}$ and the memristor array, is equivalent to a resistor, since the input signals of the filter do not change the memristance of the memristor array. Theoretically, we simulate the example filter in cases of $N = 1$ and $N = 10$ in SPICE and give the results in Figs. 15 and 16, respectively. The parameters of the simulations are as follows: $R_1 = 1$ kΩ, $R_2 = 1$ Ω, $R_{\rm ref} = 1$ kΩ, and $C = 1$ nF. The other parameters are the same as those used in Figs. 10–13.

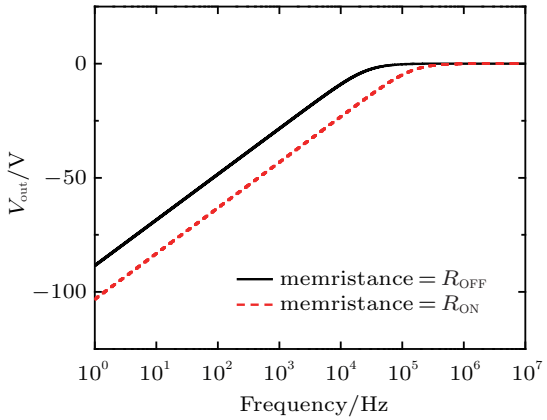


Fig. 15. (color online) Bode diagram of the example filter for the $N = 1$ case.

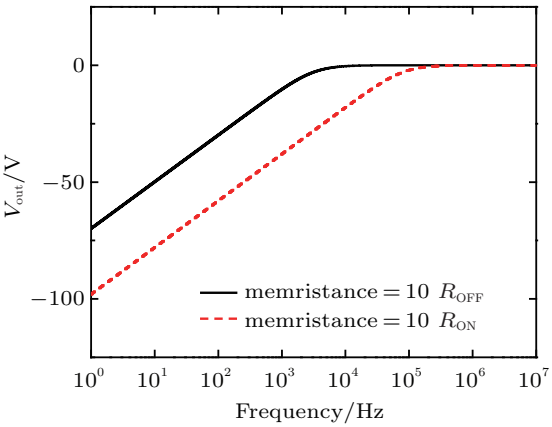


Fig. 16. (color online) Bode diagram of the example filter for the $N = 10$ case.

The simulation results demonstrate that by programming the memristance of the memristor array, the example filter circuit can accurately modify the amplitude–frequency characteristics without needing to change the structure of the circuit. It also has an adjustable pass-band range by setting the number of memristors in the filter. Moreover, the non-volatile memory of the memristors results in stable performance without the need for any reprogramming after the circuit restarts. Such a memristor-based circuit meets the requirements of the desired filter. The application of the memristance programming circuit is not limited to the example filter, it is applicable for any case where memristors are needed as resistance programmable devices.

5. Conclusions

The non-analytical relation between memristance and the charges that pass through a memristor in circuit simulations

and changeable V–I characteristics in physical tests of memristors block the applications where there is a requirement to accurately set the memristance to target values, such as in memristor-based programmable analog filters. To meet such a requirement, a memristance programmable circuit is proposed and simulated in SPICE to prove its feasibility. It is then tested in a first-order active filter as an example to demonstrate the principles of the designed circuit. Our work provides a practical method of designing programmable analog filters with memristors. It could also be utilized in applications where memristors are needed as multi-level resistance devices.

References

[1] Zhou J and Huang D 2012 *Chin. Phys. B* **21** 048401

[2] Duan S K, Hu X F, Wang L D, Li C D and Pinaki M 2012 *Sci. China: Inf. Sci.* **55** 1446

[3] Kim H, Sah M P, Yang C, Roska T and Chua L O 2011 *IEEE Trans. Circ. Sys.* **59** 148

[4] Raja T and Mourad S 2009 *International Conference on Communications, Circuits and Systems*, July 23–25, 2009, Milpitas, California, USA, p. 939

[5] Shin S, Kim K and Kang S M 2011 *IEEE Trans. Nanotechnol.* **10** 266

[6] Bao B C, Liu Z and Xu J P 2010 *Acta Phys. Sin.* **59** 3785 (in Chinese)

[7] Bao B C, Xu J P, Zhou G H, Ma Z H and Zou L 2011 *Chin. Phys. B* **20** 120502

[8] Chua L O 1971 *IEEE Trans. Circ. Theory* **18** 507

[9] Chua L O and Kang S M 1976 *Proceedings of the IEEE* **64** 209

[10] Strukov D B, Snider G S, Stewart D R and Williams R S 2008 *Nature* **453** 80

[11] Song D H, Lü M F, Ren X, Li M M and Zu Y X 2012 *Acta Phys. Sin.* **61** 118101 (in Chinese)

[12] Zhang X, Zhou Y Z, Bi Q, Yang X H and Zu Y X 2010 *Acta Phys. Sin.* **59** 6674 (in Chinese)

[13] Bao B C, Xu J P and Liu Z 2010 *Chin. Phys. Lett.* **27** 070504

[14] Bao B C, Liu Z and Xu J P 2010 *Chin. Phys. B* **19** 030510

[15] Strachan J P, Kuekes P J and Ribeiro G M 2012 US Patent 2012/0105143 A1 [2012-5-3]

[16] Torrezan A C, Strachan J P, Ribeiro G M and Williams R S 2011 *Nanotechnology* **22** 485203

[17] Driscoll T, Quinn J, Klein S, Kim H T, Kim B J, Pershin Y V, Ventra M D and Basov D N 2010 *Appl. Phys. Lett.* **97** 093502

[18] Wang W D, Yu Q, Xu C and Cui Y 2009 *International Conference on Communications, Circuits and Systems*, July 23–25, 2009, California, USA, p. 969

[19] Fang X D, Tang Y H and Wu J J 2012 *Chin. Phys. B* **21** 098901

[20] Williams R S 2008 *Spectrum IEEE* **45** 28

[21] Stewart D R, Ohlberg D A A, Beck P A, Chen Y and Williams R S 2004 *Nano Lett.* **4** 133

[22] Pickett M D, Strukov D B, Borghetti J L, Yang J J, Snider G S, Stewart D R and Williams R S 2009 *J. Appl. Phys.* **106** 074508

[23] Yang J J, Pickett M D, Li X M, Ohlberg D A A, Stewart D R and Williams R S 2008 *Nat. Nanotechnol.* **3** 429

[24] Yang J J, Miao F, Pickett M D, Ohlberg D A A, Stewart D R, Lau C N and Williams R S 2009 *Nanotechnology* **20** 215201

[25] Kim H, Sah M P, Yang C and Chua L O 2010 12*th International Workshop on Cellular Nanoscale Networks and Their Applications*, February 3–5, 2010, Berkeley, USA, p. 1