

# Memristor SPICE Model and Crossbar Simulation Based on Devices with Nanosecond Switching Time

Chris Yakopcic, Tarek M. Taha, Guru Subramanyam, and Robinson E. Pino

**Abstract**—This paper presents a memristor SPICE model that is able to reproduce current-voltage relationships of previously published memristor devices. This SPICE model shows a stronger correlation to various published device data when compared to existing SPICE models. Furthermore, switching characteristics of published memristor devices with switching times in the nanosecond scale were modeled. Therefore, this model can be used to accurately simulate neural systems based on these high-speed memristors. This paper also demonstrates how this model can be used to accurately calculate switching energy of these high-speed devices, leading to more accurate power calculations in memristor based neural systems.

Memristor crossbar circuits provide a potential method for developing very high density neural classifiers. This model was able to simulate crossbar circuits containing up to 256 memristors. It is significantly less likely to cause convergence errors when operating in the nanosecond switching regime with a large number of devices when compared with existing SPICE models.

## I. INTRODUCTION

Since the initial discovery of the memristor [1][2], several device models have been proposed [3]-[6]. Furthermore, many SPICE subcircuits have been published [7]-[14] that provide the ability to simulate memristor based circuits. Some of the existing SPICE models [7],[8],[10],[11] correlate closely to the theoretical memristor definition first proposed by Dr. Leon Chua [15]. Although, these SPICE models do not match physical memristor characterization data very well. Other SPICE models have been proposed that show a very strong correlation to a specific memristor device [5][14]. These models appear to be valid for sinusoidal waveforms and cyclic voltage sweeps, but not repetitive voltage sweeps that induce multiple resistance states. Our experience shows these existing SPICE models have convergence issues with circuits containing more than 10 memristors switching at high frequencies. This severely limits the use of these models for high speed SPICE simulation of memristor-based neuromorphic circuits.

Memristors have the potential to act as synaptic memory devices [16] within analog neural circuits [2],[4],[12]. Memristor crossbars have been proposed as a potential method for developing neuromorphic circuits with a synaptic density comparable to that found in the human brain [17]. An accurate memristor SPICE model that can support these circuits with a large number of devices is essential for developing memristor-based neuromorphic systems.

A memristor SPICE subcircuit is presented in this paper that can be used to model many different memristor devices for a variety of different voltage inputs. The SPICE subcircuit is based on a previously published mathematical model [6]. This SPICE model is capable of closely matching physical characterization data for sinusoidal inputs and cyclic voltage sweeps. Additionally, the SPICE model can accurately simulate repetitive voltage sweeps displaying the multiple analog resistance states within a single device.

The capabilities of this model were extended in this paper, as it is shown how the model can be used to simulate devices based on time-domain switching characteristics (as opposed to just the I-V plot). This led to successful device characterizations based on matching of the off to on resistance ratios, the switching time, and the shape of the input voltage waveform. The devices modeled based on switching characteristics had a switching time in the nanosecond regime. This result allows for the model to be used in SPICE simulations of high speed neuromorphic circuits. These simulations were then used to present a method for determining the set and reset switching energies of the high-speed memristor devices. This is advantageous because accurate energy calculations can be performed when using this model in SPICE circuit simulations.

This paper also presents a study where memristor crossbar circuits are simulated to determine how large of a crossbar can be simulated before convergence errors start to occur. In this study wire resistance was considered between each of the memristor devices. Additionally, a large number of input and output transistors were considered to provide an accurate simulation of a realistic crossbar. This model is capable of simulating a crossbar array with up to 256 memristors before convergence errors began to occur. Another study was performed where the presented SPICE model is compared to other models in terms of accuracy and likelihood of convergence errors to further demonstrate the effectiveness of this SPICE model.

This SPICE model was developed with the intention of modeling bipolar memristor devices. At this point in time no attempt has been made to correlate to unipolar device data (such as PCRAM) using this model. For this to be done,

Manuscript received March 1, 2013. This work was supported by NSF CAREER Grant CCF-1053149.

C. Yakopcic is with the University of Dayton, Dayton, OH 45469, USA (email: cyakopcic1@udayton.edu)

T. M. Taha is with University of Dayton, Dayton, OH 45469 USA (phone: 937-229-3119; email: ttahal@udayton.edu).

G. Subramanyam is with University of Dayton, Dayton, OH 45469 USA (email: gsubramanyam1@udayton.edu).

R. E. Pino is with ICF International, Baltimore, MD 21202 USA (email: Robinson.Pino@icfi.com).

modifications would have to be made to equations used to develop this model.

The paper is organized as follows: Section II provides a description of the SPICE model, and Section III displays how the model is able to match the characteristics presented in several memristor device publications. Section IV describes how the model can be used to calculate memristor switching energy, Section V presents memristor crossbar simulations, and Section VI concludes the paper.

## II. MEMRISTOR SPICE MODEL

The schematic for the SPICE subcircuit of the presented memristor model can be seen in Fig. 1, and The SPICE code can be seen in Fig. 2. The current source  $G_m$  in Fig 1(a) holds the I-V relationship for the device. The terminals  $TE$  and  $BE$  represent the top and bottom electrodes of the memristor where the model would be connected in a circuit schematic.

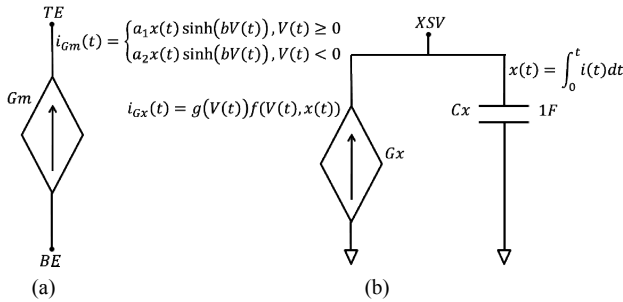


Fig. 1. Diagrams that describe the functionality of the SPICE model where the circuit in (a) determines the I-V relationship of the memristor model and (b) shows how the value of the state variable is determined.

Fig. 1(b) shows how the value of the state variable is determined using another current source and a capacitor. The current source  $G_x$  produces a current determined by the multiplication of the functions  $g(V(t))$  and  $f(x(t))$  (defined in [6]). The capacitor  $C_x$  is used to integrate the current generated by  $G_x$  to produce the value of the state variable. This technique is similar to those proposed in several memristor SPICE models [7],[9],[14]. The port  $XSV$  was created to provide a convenient method for plotting the state variable during a simulation. This is a helpful tool for debugging and analysis, but should not be used as a voltage output in a circuit design.

The code was written for use in LTSpice and the parameters provided were used to match the model to the characterization data provided in [18]. The code in Fig. 2 was used to generate the result seen in Fig. 3, and the alternate parameter sets for modeling different devices can be seen in the description of each corresponding figure.

## III. MEMRISTOR DEVICE SIMULATIONS

### A. Simulations Matching Published I-V Curves

Fig. 3 displays the first simulation result where the device in [18] was modeled with a sinusoidal input both at 100 Hz and 100 kHz. The hysteresis in the model diminished when the frequency was increased to 100 kHz just as it did in [18]. The simulated I-V characteristic was matched to the 100 Hz

data provided in [18] using selected data points (dots in Fig. 3(b)) with an average error of 84.8  $\mu A$  (6.64%). The percent error was determined by using the sum of the error divided by the sum of the device currents at each of the tested points.

```
* SPICE model for memristive devices
* Created by Chris Yakopcic
* Last Update: 12/21/2011
*
* Connections:
* TE - top electrode
* BE - bottom electrode
* XSV - External connection to plot state variable
* that is not used otherwise

.subckt mem_dev TE BE XSV

* Fitting parameters to model different devices
* a1, a2, b: Parameters for IV relationship
* Vp, Vn: Pos. and neg. voltage thresholds
* Ap, An: Multiplier for SV motion
intensity
* xp, xn: Points where SV motion is
reduced
* alphap, alphan: Rate at which SV motion decays
* xo: Initial value of SV
* eta: SV direction relative to voltage

.params a1=0.17 a2=0.17 b=0.05 Vp=0.16 Vn=0.15
+Ap=4000 An=4000 xp=0.3 xn=0.5 alphap=1 alphan=5
+xo=0.11 eta=1

* Multiplicative functions to ensure zero state
* variable motion at memristor boundaries
.func wp(V) = (xp-V)/(1-xp)+1
.func wn(V) = V/(1-xn)

* Function G(V(t)) - Describes the device
threshold
.func G(V) = IF(V <= Vp, IF(V >= -Vn, 0, -
An*(exp(-
+V)-exp(Vn))), Ap*(exp(V)-exp(Vp)))

* Function F(V(t),x(t)) - Describes the SV motion
.func F(V1,V2) = IF(eta*V1 >= 0, IF(V2 >= xp,
exp(-
+alphap*(V2-xp))*wp(V2), 1), IF(V2 <= (1-xn),
+exp(alphan*(V2+xn-1))*wn(V2), 1))

* IV Response - Hyperbolic sine due to MIM
structure
.func IVRel(V1,V2) = IF(V1 >= 0, a1*V2*sinh(b*V1),
+a2*V2*sinh(b*V1))

* Circuit to determine state variable
* dx/dt = F(V(t),x(t))*G(V(t))
Cx XSV 0 {1}
.ic V(XSV) = xo
Gx 0 XSV
+value={eta*F(V(TE,BE),V(XSV,0))*G(V(TE,BE))}
* Current source for memristor IV response
Gm TE BE value = {IVRel(V(TE,BE),V(XSV,0))}

.ends mem_dev
```

Fig. 2. Memristive device subcircuit for use in LTSpice.

Fig. 4 shows the simulation result when matching the model to the repetitively swept input data also provided in [18]. In this case, the model was able to match the characterization data with an average error of 32.7  $\mu A$

(6.66%). The characterization data provided for this simulation was only available for when positive voltage sweeps were applied. It was assumed that the parameters for the negative regime would closely match the parameters used in the sinusoidal simulation in Fig. 3. The one exception was that the negative conductivity parameter  $a_2$  was set to the values decided for  $a_1$  in the pulsed simulation.

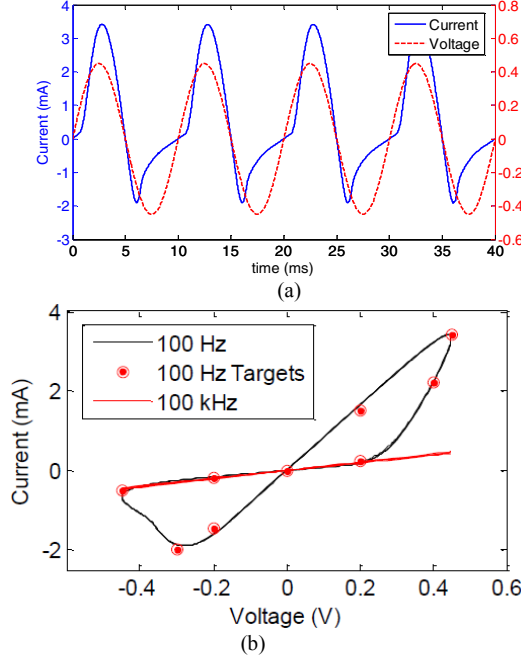


Fig. 3. Simulation results when modeling the device in [18] for a sinusoidal input (Dots in (b) show target data points). Plots show (a) the current and voltage waveforms and (b) the I-V curve. The plot in (b) also shows the I-V curve for a high frequency input where the device behaves as a linear resistor. In this simulation:  $V_p=0.16V$ ,  $V_n=0.15V$ ,  $A_p=4000$ ,  $A_n=4000$ ,  $x_p=0.3$ ,  $x_n=0.5$ ,  $\alpha_p=1$ ,  $\alpha_n=5$ ,  $a_1=0.17$ ,  $a_2=0.17$ ,  $b=0.05$ ,  $x_0=0.11$ .

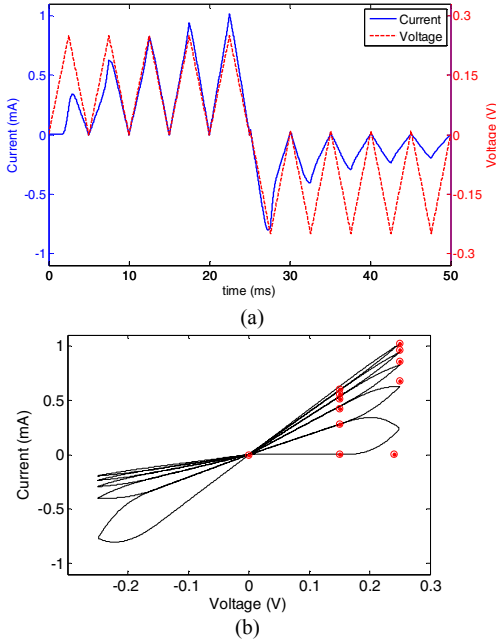


Fig. 4. Results matching the pulsed input data in [18]. Dots in (b) show the points from [18].  $V_p=0.16V$ ,  $V_n=0.15V$ ,  $A_p=4000$ ,  $A_n=4000$ ,  $x_p=0.3$ ,  $x_n=0.5$ ,  $\alpha_p=1$ ,  $\alpha_n=5$ ,  $a_1=0.097$ ,  $a_2=0.097$ ,  $b=0.05$ ,  $x_0=0.001$ .

When comparing the many parameters used to model the device characterized in [18], it can be seen that very few

adjustments were necessary when switching between the two modes of operation. Out of the total 12 parameters, 9 of them remained the same between the simulations in Figs. 3 and 4. The three parameters that were changed include the conductivity parameters  $a_1$  and  $a_2$ , and the initial position of the state variable  $x_0$ . Each of these parameters is highly related to the device thickness. These differences could have been caused by non-uniformities in the wafer if the characterizations in [18] were conducted on different devices.

The plot in Fig. 5 shows a similar result where the SPICE model was used to match the device characterization of a Tantalum Oxide ( $TaO_x$ ) memristor published by HP Labs [19]. This simulation is based on a cyclic voltage sweep and the average error in this case was determined to be  $81.1 \mu A$  (4.60%).

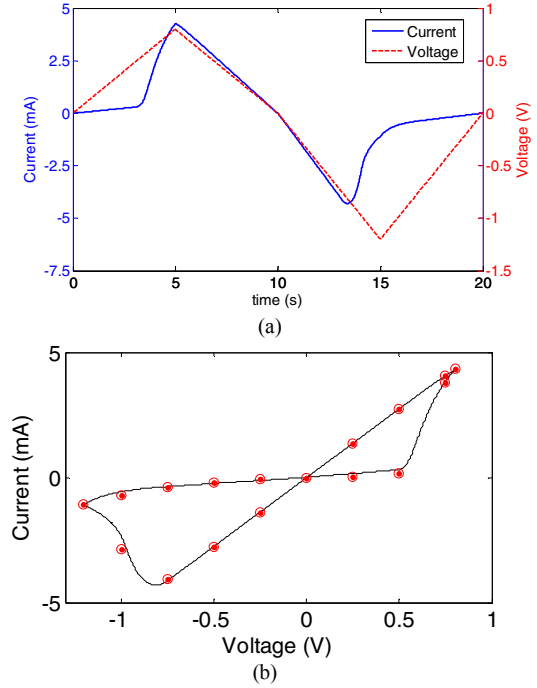


Fig. 5. Results when modeling the device in [19] for a cyclical DC sweep (dots in (b) show target data points).  $V_p=0.5V$ ,  $V_n=0.75V$ ,  $A_p=7.5$ ,  $A_n=2$ ,  $x_p=0.3$ ,  $x_n=0.5$ ,  $\alpha_p=1$ ,  $\alpha_n=5$ ,  $a_1=0.11$ ,  $a_2=0.11$ ,  $b=0.5$ ,  $x_0=0.11$ .

### B. Simulation to Match Published High-Speed Switching Characteristics

Figs. 6 and 7 display results when modeling high-speed switching observed in some memristor devices. The simulation result seen in Fig. 6 was also performed to match a  $TaO_x$  device characterization in [19]. This device had a different geometry than the one modeled in Fig. 5, and the characterization in [19] was performed at a much higher speed. Fig. 6 (a) displays a string of alternating write and erase pulses with a Gaussian shape, each followed by a lower magnitude read pulse. The read pulse is applied below the voltage threshold of the memristor device so data is not corrupted during a read operation. The Gaussian write pulses were chosen so that the write pulses in [19] could be matched as closely as possible where the full width at half maximum of the write and erase pulses was 1.5ns and 1.91ns respectively.

The published data [19] shows that the on state of the physical memristor that has been modeled in this section has a resistance of about  $100\ \Omega$ . Also, the off state of this memristor has a slight amount of variance, but is centered at about  $10^5\ \Omega$ . This correlates closely to the simulation data presented where the on and off state resistances of the model are  $100.08\ \Omega$  and  $93,360\ \Omega$  respectively.

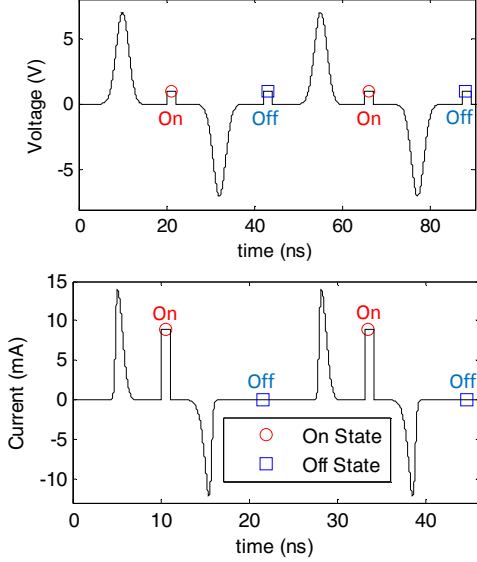


Fig. 6. Voltage and current waveforms simulated to match the 10ns switching memristor device described in [19]. The following fitting parameters were used in the model to obtain this result:  $V_p=1.1V$ ,  $V_n=1.1V$ ,  $A_p=1.9(10^9)$ ,  $A_n=1.9(10^9)$ ,  $x_p=0.675$ ,  $x_n=0.675$ ,  $a_p=0.01$ ,  $a_n=0.01$ ,  $a_I=0.2$ ,  $a_z=0.2$ ,  $b=0.05$ ,  $x_0=0.001$ .

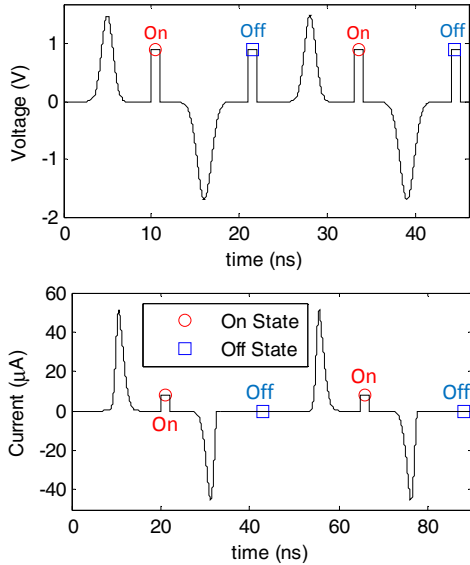


Fig. 7. Voltage and current waveforms simulated to match the 10ns switching memristor device described in [20]. The following fitting parameters were used in the model to obtain this result:  $V_p=1.088V$ ,  $V_n=1.088V$ ,  $A_p=816000$ ,  $A_n=816000$ ,  $x_p=0.985$ ,  $x_n=0.985$ ,  $a_p=0.1$ ,  $a_n=0.1$ ,  $a_I=1.6(10^{-4})$ ,  $a_z=1.6(10^{-4})$ ,  $b=0.05$ ,  $x_0=0.01$ .

The result in Fig. 7 is a similar simulation where a different published device [20] is modeled based on switching characteristics. When compared to the device simulated in Fig. 6, this device has a slightly longer switching time (10ns) and a larger off to on ratio ( $10^6$ ).

Additionally, this device operates at a much lower power with an on state resistance of about  $125k\Omega$  (determined by the  $8\mu A$  current from a  $1V$  read pulse). This correlates closely to the simulation data presented where the on and off state resistances of the model are  $124.95k\Omega$  and  $125.79 \times 10^9\Omega$  respectively.

### C. Comparison to Other Published Models

The model presented in this paper has a greater flexibility and accuracy in the shape of the I-V characteristics that it can produce when compared to other existing models. The plots in Fig. 8 show the results from SPICE equivalents of a number of other memristor models [3],[4],[7],[13] compared to the proposed model for the case of a repetitively sweeping input to induce multiple resistance states. Both the Joglekar [3] and Biolek [7] models have a problem in the shape of the I-V characteristic where the size of the hysteresis loops increases with device conductivity. The opposite effect is seen in physical characterizations [16]-[18]. These models are based on the early memristor equations first proposed by HP Labs [1]. Similarly to the model presented in this paper, the Laiho [4] and Chang [13] models are based on a hyperbolic sine relationship. This appears to provide a more realistic I-V characteristic. The model presented in this paper goes one step further, as it has the ability to be more accurately shaped to physical characterization data.

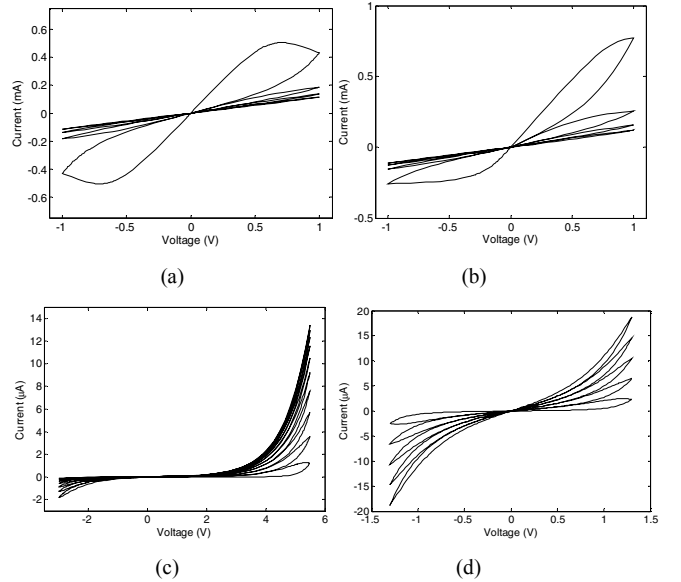


Fig. 8. Multiple state I-V characteristic for the (a) Joglekar [3], (b) Biolek [7], (c) Laiho [4], and (d) Chang [13] existing memristor models.

## IV. MEMRISTOR SWITCHING ENERGIES

This section shows how the results of the simulations in Figs. 9 and 10 can be used to calculate the switching energy of a memristor device. Both a set and reset pulse was applied to the model which has been matched to the devices in [19,20]. The applied voltage was then used to determine the total set and reset energies for the devices. The energy calculation in this section is based on the memristor model and the data present in the published device data [19,20].

This energy calculation does not consider the potential snapback effect present in some resistive switching devices [21,22].

The waveforms obtained from the simulation include the input voltage applied across the memristor, the memristor current, and the memristor state variable. The memristor state variable shows that the voltage applied is sufficient to fully switch the device as desired. The time varying power in the memristor device is determined by multiplying the current and voltage waveforms. Lastly, the power is integrated to determine the energy. It can be seen that there are two steady energy levels other than the initial value of zero. The on switching is equivalent to the energy value at Level 1, and the off switching energy is determined by subtracting the energy at Level 1 from Level 2. The on and off switching energies for the device simulated in Fig. 9 were determined to be 15.6pJ and 12.97pJ respectively. These relatively high values are mainly due to the low on state resistance of 100Ω. The device simulated in Fig. 10 was determined to have a much lower switching energy with values of 0.54pJ (on) and 0.46pJ (off).

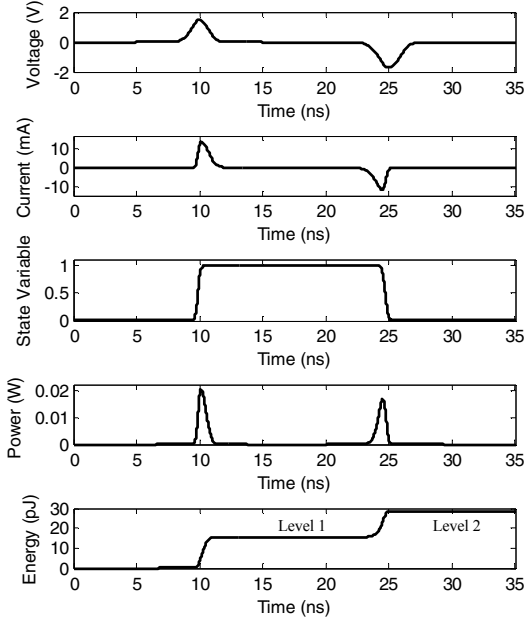


Fig. 9. Resulting waveforms and energy calculation for the device in [19].

## V. MEMRISTOR CROSSBAR SIMULATION

### A. Crossbar Circuit Setup

Simulations in this section were performed on crossbars with sizes of 4×4, 8×8, and 16×16, consisting of 16, 64, and 256 memristors respectively. The schematic for the 4×4 crossbar circuit can be seen in Fig. 11, and the larger circuits can be imagined as larger versions of this same structure. A large number of read/write cycles were performed to observe the performance of the crossbar circuits. This was done by generating a text file in MATLAB that contained a large string of read and write pulses. In addition to the memristor model being matched to published characterization data, wire resistances between devices as well as transistors surrounding the crossbar were simulated to improve the

accuracy of the result. Each wire resistance in the simulation was set at 500Ω. The higher resistance is due to the assumption that nanowires are used to connect the memristor elements in the crossbar.

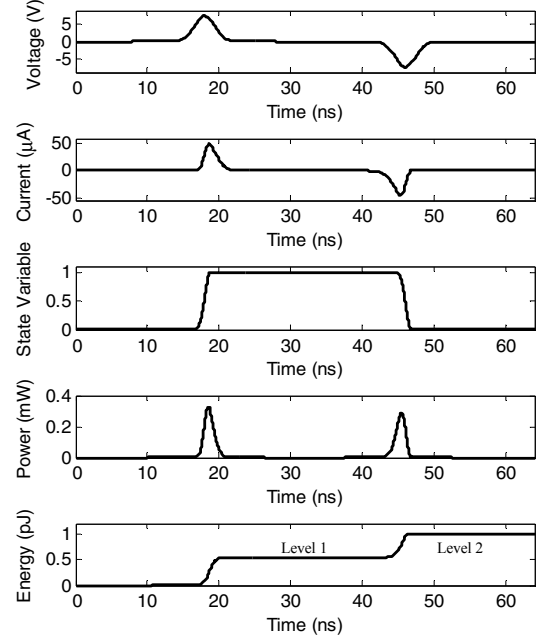


Fig. 10. Resulting waveforms and energy calculation for the device in [20].

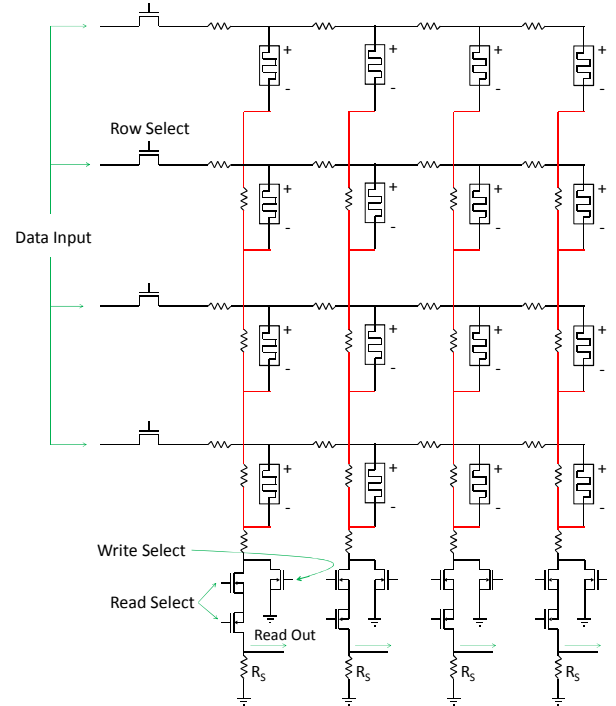


Fig. 11. Schematic for the 4 by 4 crossbar circuit simulated in LTSpice.

During a write operation, data signals are applied to one of the four row wires while grounding the others. A write operation in this design is considered to be a two-step process [21] that writes to an entire row of memristors (see Fig. 12). This two-step write technique is used to reduce the voltage across unselected memristors but is only beneficial when using bipolar memristive devices (as opposed to



unipolar). The first step is to apply a voltage of  $V_w/2$  to a selected row (grounding the other 3), and to apply a voltage of  $-V_w/2$  to all columns where a 1 should be written (where  $V_w$  is the write voltage). Furthermore, a voltage of  $V_w/2$  should be applied to all column wires where a 0 should be written. This consists of the first step in the write process. This will result in writing a 1 to only the memristors in the selected row that need to be set to 1. During the second step in the write process a voltage of  $-V_w/2$  is applied to the selected row, and all column wires are set as they were step 1. This will result in writing a 0 to the rest of the memristors in the row. A read operation is also performed in parallel. The selected row is set to the read voltage  $V_s$ , and the read select transistors in each column circuit are activated. The output voltage is considered to be the voltage across  $R_s$ .

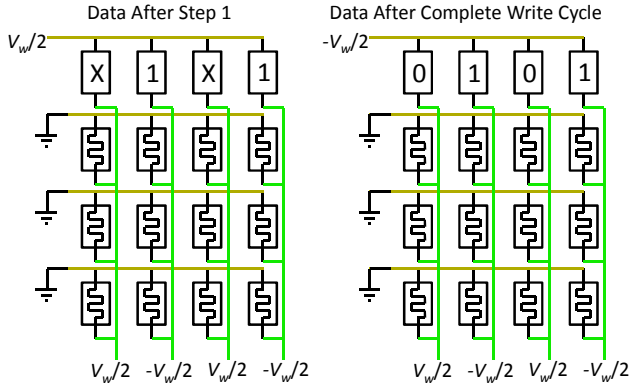


Fig. 12. Demonstration of the write operation in a crossbar simulation.

### B. Crossbar Simulation Result

The plots in Fig. 13 show a snapshot of a crossbar simulation. The waveforms displayed are the voltages applied to the row and column of a selected memristor, as well as the corresponding memristor state variable. A state change in the memristor will occur when a row write pulse and a column write pulse overlap with opposite polarity. During this small portion of the simulation, the memristor was switched on at about  $1.15\mu s$  due to a positive row pulse and a negative column pulse. Likewise the device was switched off at  $1.28\mu s$  due to a negative row pulse and a positive column pulse. It can also be seen that the write pulse pair at  $1.23\mu s$  did not change the memristor state value since the memristor was already in the desired state.

The other half-write voltage signals that do not align (such as the one at  $1.21\mu s$ ) are meant to impact other memristors in the crossbar and are combined with different write signals. While the positive column voltage pulse at  $1.21\mu s$  does not impact this memristor, it is paired with a different row voltage (not shown) to switch a different memristor in the crossbar. The read pulses can be seen in Fig. 13 as the thinner pulses between writes that are applied below the memristor write threshold.

It was apparent leakage paths within the crossbars degraded the signal as the crossbar size increased. A reasonable noise margin could be obtained for the  $4\times 4$  and  $8\times 8$  cases, but data errors occurred when reading the  $16\times 16$  crossbar. This issue and potential solutions are discussed further in [22,23].

### C. Convergence Based on Crossbar Size

The likelihood of convergence was studied in each of the three crossbar sizes ( $4\times 4$ ,  $8\times 8$ , and  $16\times 16$ ). This was done by applying a write operation to a single row and then reading each row in the crossbar in series to ensure the memristors were written correctly. So a single write/read cycle in these simulations would consist of 4 writes and 16 reads for the  $4\times 4$  crossbar, 8 writes and 16 reads for the  $8\times 8$  crossbar, and 16 writes and 256 reads for the  $16\times 16$  crossbar. The  $4\times 4$  crossbar performed very consistently without error for a number of simulations with at least 400 cycles (1600 writes and 6400 reads).  $8\times 8$  crossbar produced a similar result where a convergence error was rarely obtained within 400 read/write cycles (3200 writes and 25600 reads). In the  $16\times 16$  crossbar, convergence errors started to become more common, although many simulations were completed up to 200 read/write cycles (3200 writes and 51200 reads) were without error.

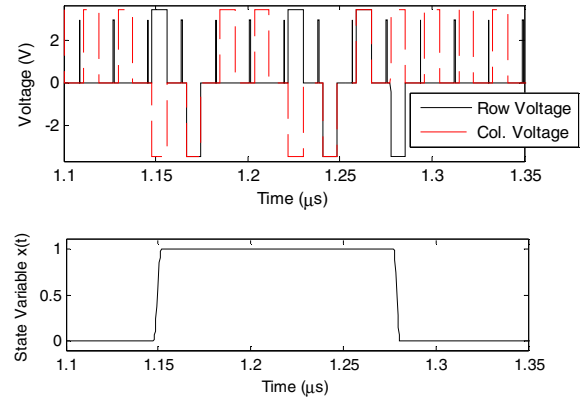


Fig. 13. A write operation for a single memristor from the crossbar simulation.

### D. Converge Comparison to Different SPICE Models

In preliminary work, it has been observed that many other memristor SPICE models produce convergence errors when more than about 10 memristors are simulated. The model in Fig. 2 in addition to a number of different SPICE models [3],[4],[7],[13] were used to simulate the circuit displayed in Fig. 11 to compare them in terms of convergence ability in a crossbar simulation (see Table 1). In cases where they previously did not exist, SPICE subcircuits were made for these models that could be used in LTSpice.

Of the models chosen to simulate, the Joglekar [3] and Biolek [7] models are based on the initial memristor equations first proposed by HP Labs. The difference between these models was in the windowing function used to add boundaries at the points of minimum and maximum resistance. When circuits using these models were simulated, simulation stopped at 517ns, which is a point where many of the memristors were being switched to the point of minimum resistance. The Biolek model suffered a convergence error, and the memristor boundary function failed on the Joglekar model leading to incorrect operation.

Both the Laiho model [4] and the model presented in this paper were able to complete the full crossbar simulation

which consisted of 400 read/write cycles for a total of 1600 memristor write operations and 6400 reads. However the Laiho model was not as accurate or versatile. The Biolek window function was added to the Laiho model to add bounds to the resistance values. Lastly, the Chang model was also unable to complete a full simulation of the crossbar circuit. Convergence errors occurred when the model was set to switch at a high speed.

While some of these models did not perform well in this specific simulation, they still excel in other applications. The most likely reason for the errors in some models was that they were not designed to switch at the nanoscale. In some cases, switching parameters had to be significantly increased, and this may have pushed certain models out of their realm of operation. However, the model presented in this paper appears to be the one most suited for modeling memristors that switch at the nanosecond scale.

Table 1. Time before convergence error of different SPICE models for a nanosecond switching 4×4 crossbar.

Measurement	Memristor Models Included in the Study				
	<i>Biolek</i> [7]	<i>Joglekar</i> [3]	<i>Laiho</i> [4]	<i>Chang</i> [13]	<i>Proposed Model</i>
Time Before Error (μS)	0.517	0.517	15 (Finished)	0.135	15 (Finished)
Accuracy	Low	Low	Medium	Medium	High
Generality	High	High	Medium	Medium	High

## VI. CONCLUSION

In this paper a memristor SPICE model has been presented that is capable of accurately reproducing the I-V curves of several devices. Furthermore, the switching characteristics of 2 different high-speed memristor devices were modeled. The model was also used to determine the switching energy for the high-speed memristor devices.

The model was then used in the simulation of different crossbar circuits where up to 256 memristors could be simulated at one time. This model can provide more accurate matching of I-V characteristics, and it appears to have less likelihood of causing convergence errors operating at high speeds when compared to other models. This model has the potential to improve the capability of simulating neuromorphic hardware given that it can be used to simulate high-speed devices in large crossbars while providing accurate data pertaining to energy consumption.

## REFERENCES

- [1] D. B. Strukov, G. S. Snider, D. R. Stewart, and R. S. Williams, "The missing Memristor found," *Nature*, 453, 80–83 (2008).
- [2] R. Williams, "How We Found The Missing Memristor," *IEEE Spectrum*, vol. 45, no. 12, pp. 28-35, 2008.
- [3] Y. N. Joglekar and S. J. Wolf, "The elusive memristor: properties of basic electrical circuits," *European Journal of Physics*, 30(661), (2009).
- [4] M. Laiho, E. Lehtonen, A. Russel, and P. Dudek, "Memristive synapses are becoming a reality," *The Neuromorphic Engineer*, (2010).
- [5] R. E. Pino, J. W. Bohl, N. McDonald, B. Wysocki, P. Rozwood, K. A. Campbell, A. Obblea, and A. Timilsina, "Compact method for modeling and simulation of memristor devices: Ion conductor chalcogenide-based memristor devices," *IEEE/ACM International Symposium on Nanoscale Architectures*, pp. 1 – 4, (2010).

- [6] C. Yakopcic, T. M. Taha, G. Subramanyam, R. E. Pino, S. Rogers, "A Memristor Device Model," *IEEE Electron Device Letters*, Oct. 2011.
- [7] Z. Biolek, D. Biolek, V. Biolková, "Spice Model of Memristor with Nonlinear Dopant Drift," *Radioengineering*, 18(2), 210-214 (2009).
- [8] D. Batas, H. Fiedler, "A Memristor SPICE Implementation and a New Approach for Magnetic Flux-Controlled Memristor Modeling," *IEEE Transactions on Nanotechnology*, 10(2), 2011.
- [9] A. Rak and G. Cserey, "Macromodeling of the memristor in spice," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 29(4), pp. 632 –636, Apr. 2010.
- [10] S. Benderli and T. Wey, "On SPICE macromodelling of TiO2 memristors," *Electronics Letters*, 45(7), pp. 377–379, 2009.
- [11] M. Mahvash and A. C. Parker, "A memristor SPICE model for designing memristor circuits," Aug. 2010, pp. 989–992.
- [12] E. Lehtonen and M. Laiho, "CNN using memristors for neighborhood connections," feb. 2010, pp. 1–4.
- [13] T. Chang, S. H. Jo, K. H. Kim, P. Sheridan, S. Gaba, W. Lu, "Synaptic behaviors and modeling of a metal oxide memristor device," *Applied Physics A*, 102 pp. 857-863, (2011).
- [14] H. Abdalla, and M. D. Pickett, "SPICE Modeling of Memristors," *ISCAS* (2011).
- [15] L. O. Chua, Leon O, "Memristor—The Missing Circuit Element," *IEEE Transactions on Circuit Theory*, 18(5), 507–519 (1971).
- [16] S. H. Jo, T. Chang, I. Ebong, B. B. Bhadviya, P. Mazumder, and W. Lu, "Nanoscale Memristor Device as Synapse in Neuromorphic Systems," *Nano Letters*, 10 (2010).
- [17] G. S. Snider, "Cortical Computing with Memristive Nanodevices," *SciDAC Review*, (2008).
- [18] A. S. Obblea, A. Timilsina, D. Moore, and K. A. Campbell, "Silver Chalcogenide Based Memristor Devices," *IJCNN*, (2010).
- [19] F. Miao, J. P. Strachan, J. J. Yang, M.-X. Zhang, I. Goldfarb, A. C. Torrezan, P. Eschbach, R. D. Kelley, G. Medeiros-Ribeiro, R. S. Williams, "Anatomy of a Nanoscale Conduction Channel Reveals the Mechanism of a High-Performance Memristor," *Advanced Materials*, vol. 23, no. 47, pp. 5633-5640, Nov. 2011.
- [20] W. Lu, K.-H. Kim, T. Chang, S. Gaba, "Two-terminal resistive switches (memristors) for memory and logic applications," in *Proc. 16th Asia and South Pacific Design Automation Conference*, 2011, pp. 217-223.
- [21] C.-N. Peng, C.-W. Wang, T.-C. Chan, W.-Y. Chang, Y.-C. Wang, H.-W. Tsai, W.-W. Wu, L.-J. Chen and Y.-L. Chueh, "Resistive switching of Au/ZnO/Au resistive memory: an in situ observation of conductive bridge formation," *Nanoscale Research Letters*, 7:559, 2012, pp. 1-6.
- [22] Chen TP, M. Tse S, Sun CQ, Fung S, Lo KF, "Snapback behaviour and its similarity to the switching behaviour in ultra-thin silicon dioxide films after hard breakdown," *J Phys D: Appl Phys* 2001, 34:95–98.
- [23] X. Dong, C. Xu, S. Member, Y. Xie, and N. P. Jouppi, "NVSIM: A Circuit-Level Performance, Energy, and Area Model for Emerging Nonvolatile Memory," *IEEE Trans. on Computer Aided Design of Integrated Circuits and Systems*, 31(7), July, 2012, pp. 994-1007.
- [24] T. M. Taha, R. Hasan, C. Yakopcic, and M. R. McLean, "Exploring the Design Space of Specialized Multicore Neural Processors," *IEEE International Joint Conference on Neural Networks (IJCNN)*, August 2013 (Accepted).
- [25] C. Yakopcic and T. M. Taha, "Energy Efficient Perceptron Pattern Recognition Using Segmented Memristor Crossbar Arrays," *IEEE International Joint Conference on Neural Networks (IJCNN)*, August 2013 (Accepted).