International Advance Journal of Engineering Research (IAJER)
Volume 3, Issue 3 (March- 2020), PP 33-39
ISSN: 2360-819X
www.iajer.com

Research Paper **Open Access**

Image Identification Using CMOS-Memristor Hybrid Architecture In Neuromorphic System

Minh-Huan Vo, Nguyen Thi Bich Thuy, Pham Sanh
HCMC University of Technology and Education, Vietnam

ABSTRACT:- Image identification using memristor in neuromorphic system is a specific application to demonstrate the usefulness and popularity of memristor in the future by the outstanding features such as storage capacity, high integration density, low power energy. This CMOS-memristor hybrid architecture identifies 10 images, each image is a 5x6 pixel array, each pixel is a signal to the system. There are 30 signals into the system and these signals pass through 300 memristors, divided into 10 memristor arrays, each array will respectively have 30 memristors. The system has 10 signals going out from 10 memristor arrays then continues into the integrated block. The integrated block has two components which are neurons and switch controller. Image identification performs two modes in training and test. An on-chip training mode is implemented by a simpler circuit controller in this paper. The recognition rate is evaluated in consideration of additive noise.

Keywords - Neuromorphic, memristor, CMOS, crossbar array, image identification

I. INTRODUCTION

Moore's law will reach the physical limits soon because the transistor size is shrunk. Thus, many researchers are looking for a new direction to orientate IC technology in the future. One of the directions is to find new devices. This is the memristor in which, the neuromorphic system for visual pattern recognition can be realized in hardware. A new learning rule based on modified Spike-timing-dependent plasticity (STDP) is also presented and implemented with memristors [1]. In 2008, the physical realization of a memristor was firstly demonstrated by HP Lab through a TiO2 thin-film structure [2]. Memristor can remember the total electric charge or flux ever flowing through it [3-4]. Memristor can achieve a very high integration density of 100 Gb/cm2 [5]. Memristor has been made to develop a passive device that behaves like a synapse [1]. The memristor crossbar array operates as an auto-associative memory and is applied to brain-state-in-a-box (BSB) neural networks [7]. CMOS technology allows large-scale integration of integrate-and-fire (I&F) neurons on a single chip. However, CMOS implementation of synaptic circuits requires many transistors, a considerable amount of power consumption [6], and the Von Neumann structure bottleneck [8]. So memristor has been selected to solve these problems.

Moreover, an on-chip training mode is implemented by a simpler circuit controller in this paper. Memristances are changed by applying a pulse width to determine the synaptic weights [8-10]. The process of training and working in a neuromorphic system is conventionally implemented separately. A training mode is performed by software such as Matlab or C/C++ language [9-10]. The training procedure works in the same way as programming ROM memory. That means the training process is implemented in off-chip conventionally. A simple on-chip training solution of combining training and working modes is proposed in this research, although various on-chip training techniques have published recently [11-12]. Here, weighting values can be programmed based on innate properties of a configurable resistor and memristor memory by backpropagation algorithm [11-12]. Simultaneously, the testing process can be done to recognize the input.

II. MEMSITOR MODEL

The memristor is defined by a nonlinear algebraic relation between its charge q and flux φ [8]. When the current flows in the forward direction, the impedance of the memristor decreases and vice versa. When the current is inverted, the impedance of the memristor increases. When the voltage stops applying to the memristor, it will remember the impedance in the final state. Holding the impedance value until power is restored. Each memristor is described by a memory impedance function, describing the flux change rate based on the charge flowing through the device.

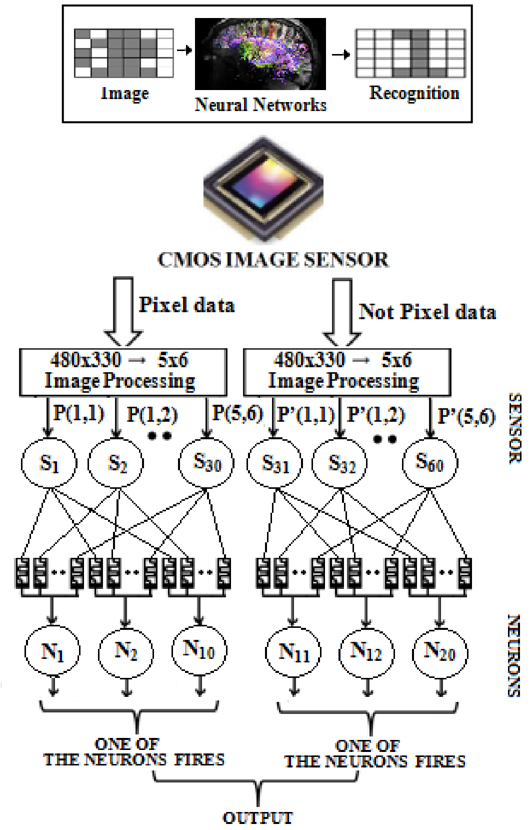

Figure. 1. Proposed scheme for character identification.

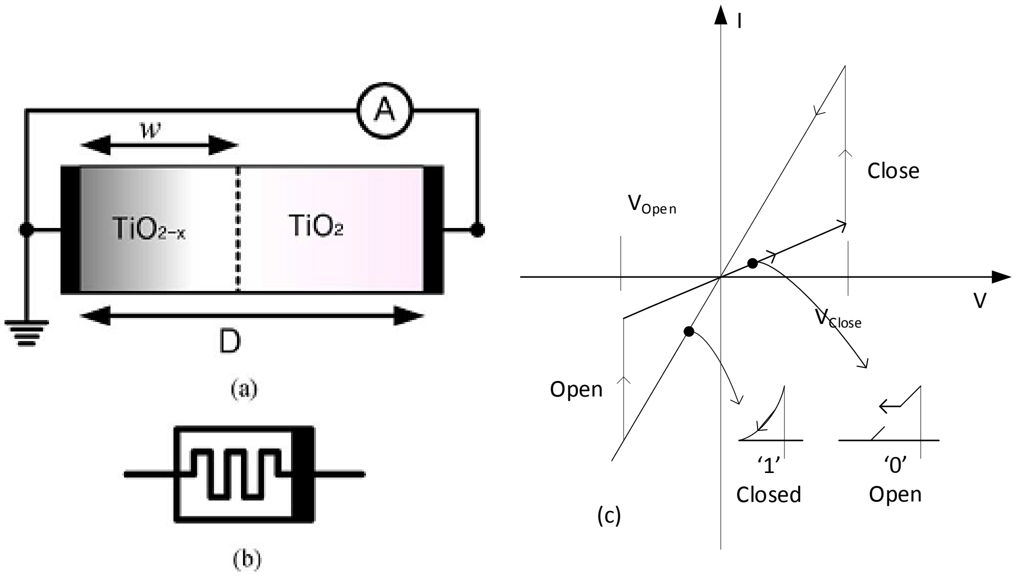

Figure. 2. (a) Structure of the TiO_2 memristor. (b) Symbol of the memristor. (c) I-V memristor characteristic [2]

According to Faraday's law of electromagnetic induction, the above formulas can be written in the following form:

$$M(q) = \frac{d\phi_m}{dq} \qquad (1)$$

Fig. 2 shows memristor layers that TiO_2 and TiO_{2-x} layers are sandwiched between two platinum electrodes. When a voltage/current is applied, its memristance (resistance of the memristor in Ohms) / memductance (conductance of the memristor) is altered respectively [2].

$$M(q(t))=\frac{\frac{d\phi}{dt}}{\frac{dq}{dt}}=\frac{v(t)}{I(t)} \qquad (2)$$

$$R(t)=\frac{v(t)}{I(t)} \qquad (3)$$

Assuming that M(q(t)) is a constant, it can derive an equation according to Ohm's law (3), but M(q (t)) is a change in time. It depends on the charge placed on the memristor that the power consumption can be written in the following.

$P(t) = V(t). I(t) = I^2(t). M(q(t))$ (4)

If the voltage is not applied to the memristor, V(t) = 0, I(t) = 0, and M (t) is constant. It is the impedance of the memristor. At the same time, the circuit does not consume energy.

In this study, an ideal memristor model is used. There are two states of high resistance and low resistance in this ideal memristor. If the input voltage is greater than V_{Close}, memristor is turned to low resistance state. Otherwise, if the input voltage is smaller than V_{Open}, the memristor is changed to high resistance state. The input voltage is in range of V_{Open} and V_{Close}, the memristor state keeps unchanged. The I-V memristor characteristic is shown in fig. 2c.

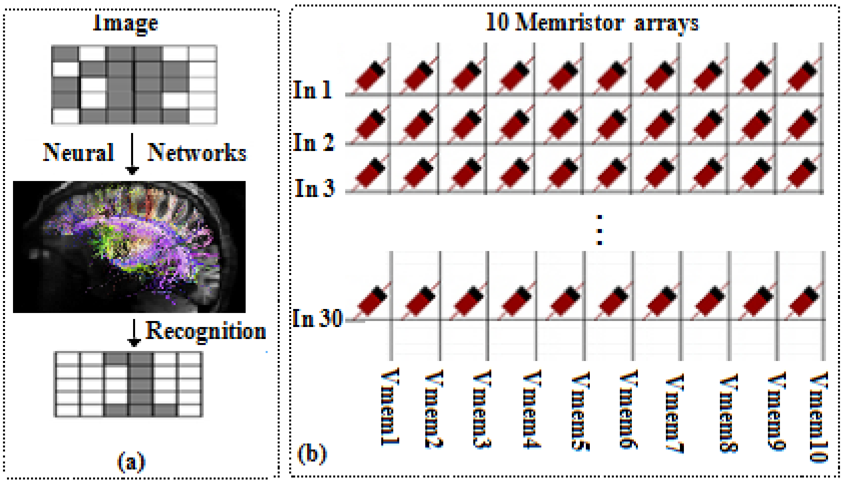

Figure 3. (a) Idea for image identification, (b) 10 memristor arrays from Vmem1 to Vmem10.

Shown in Fig. 3, each digital image coming into the regconization system has a size of 5x6 pixels. The voltage level is 3.3V corresponding to logic level '1', the voltage level of 0V corresponds to logic level '0'. So, each image including 30 pixels are 30 input signals into the memristor array block.

The memristor array (M) has 10 output signals from Vmem1, Vmem2, ..., to Vmem10. At t = 0, all memristors in the arrays have a high impedance of 21 KΩ. If the input is level-1 logic, the memristor moves from high impedance to low impedance of 100 Ω, if the signal is the level-0 logic, it will not change the impedance of the memristor.

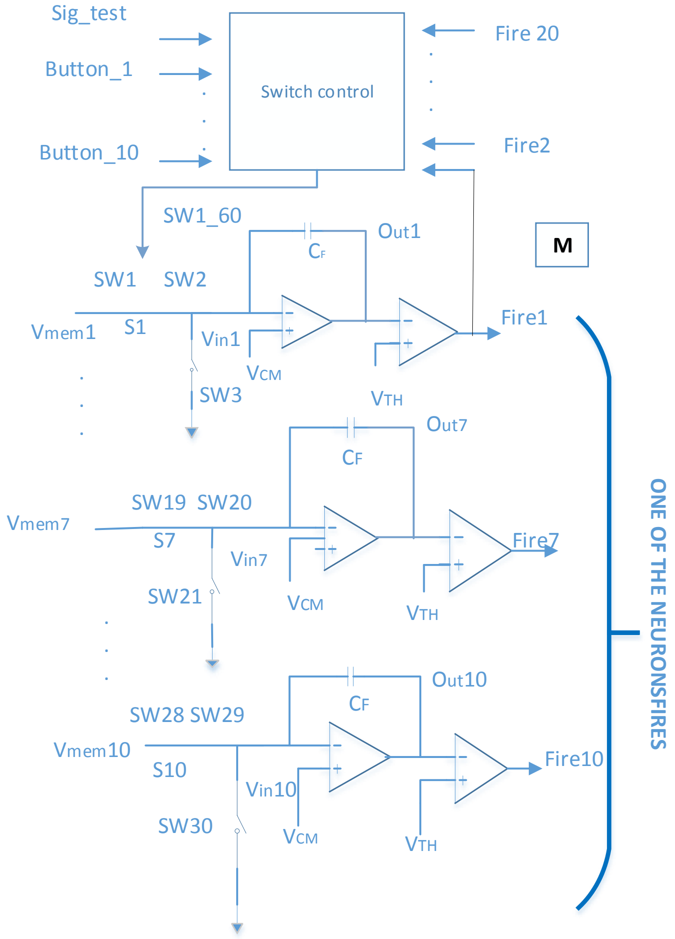

Figure 4. Integrated block in proposed neuromorphic system architecture consisting of an M memristor array, switch control block, integration circuit.

The proposed neuromorphic system architecture consists of a switch controller, integrated neuron circuit, memristor circuit as shown in Fig. 4. In this case, the switch control block will get 10 button signals to select which image is being trained. For example, if image 1 is selected for training process, the switch control block turns ON switches of SW1, SW2 and SW3, and turns OFF the other switches for M memristor array. Thus, pixel input signals of image 1 will program memristor memristance which changes to low state memristance in case of high logic pixel and keep high state memristance in case of low logic pixel. Thus, memristor can be programmed on-chip by input image pixels.

In test mode process, the switch control block turns ON switches of SW1, SW2, and turns OFF SW3. Thus, the Vmem1 signal connects to integration circuit to store charge on C_F capacitor. Similarly, switch controller allows the other Vmem signals to store charge on the C_F capacitors by disconnecting path to ground.

The sig_test signal makes decision whether test process or training process is performed. The sig_test gets logic level '0' in the traning process and gets logic level ‘1’ in the test process. The signals from fire1 to fire10 is output signals of M array. The signal firing result will get logic level ‘0’or ‘1’ at the output. The output of switch control block is logic level controlling SW1 to SW30.

The M array output includes of the 10 signals from Vmem1 to Vmem10 and the 10 output signals forming fire1 to fire10. However, the only one signal is fired the fastest and achieves logic level ‘1’the earliest. The other signals will get logic level ‘0’. The integration circuit consisting of C_F capacitor and Opamp plays a role to integrate charge for firing step.

III. SIMULATION RESULTS

The system operates in two modes of training process and test process. In training process, the original images consist of 10 number character images. The system imports these original images to the memristor crossbar array (M) as seen at Fig. 4.

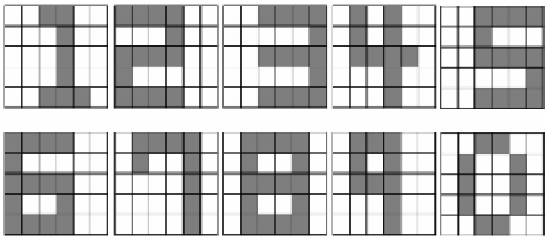

Figure 5. Training images for M array.

Next, we consider the effect of noise into recognition rate. Assuming that the test image is image-1. The important factor that makes image 1 reaches the first 3.3-volt voltage compared to the others is because the test image matches accurately to 8-pixel positions of original image. If one noise pixel with logic level '1' is added at any points in 5x6 pixel image, the noise image has 9 pixels with logic level '1' as shown in fig. 6

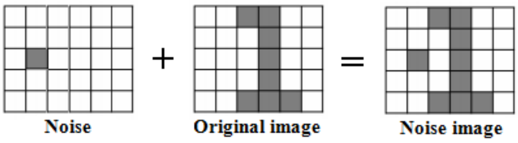

Fig. 6 Additive noise image

The pixel order is marked from left to right and from top to bottom. For example, a noise pixel is marked at the 14th pixel position as shown in fig. 6 and fig. 7, where identification image is digit 1. In fig. 7, 9 pixels are highlighted in red color, but pay attention to observe 8 pixels forming the digit 1. By masking the image 1 over image 2 and extracting the overlap area portion, image 2 has 6 black pixels overlapping to image 1 in level-1 pixels. Similarly, image 3, 5, 6, 8, 9 has 6 black level-1 overlap pixels. Images 4, 7, 0 have 5, 3, 4 level-1 overlap pixels to image 1, respectively.

At t = 0, image 1 will be fired with the highest voltage among the remaining images in case of no noise affected.

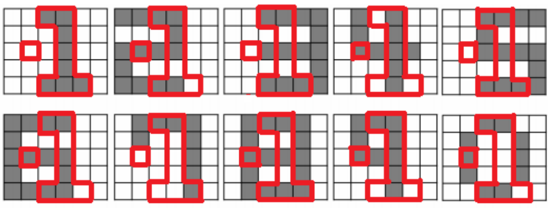

Fig. 7 Test process with one noise pixel for M array.

If only one noise pixel is added to original image 1, the scheme identifies successfully the 100% image-1 recognition rate. To explain more clearly, we look at the red border area in fig. 7. The images 2, 3, and 5 have 7 pixels with logic '1' in overlap portion including one noise pixel. Thus, these images cannot reach 3.3 Volt at same speed as image 1 which has 8 black pixels. For an example in image 2, it just reaches the 3.3 Volt at the same speed as image 1, if and only if at least 2 noise pixels are added to image 1 at the white pixel positions which are same positions as black pixels of image 2. There are 8 black pixels at outer overlap portion in image 2. With image size of 5×6 =30 pixels, the first noise pixel probability of 8 black pixel positions may be occurred to ratio of 8/30, so $P_{(1)} = 8/30$. Similarly, the second pixel noise probability will be $P_{(2)} = 7/30$. The probability occurs these 2 black noise pixels for 3 above images in this case as following: $P_{(2\ pixels\ )} = 3x\ (P_{(1)} x\ P_{(2)}) = 3x\frac{8}{30}\ x\frac{7}{30} = 18.7\ \%$. Here, there are 3 images being able to be affected by 2-pixel noise that will reach the same speed as image 1. As consequence, system identifies wrong image 1 in this 2-pixel noise case.

Similarly, there are 9 black pixels at outer overlap portion of image 1 and image 6. If image 6 has 2 black noise pixels added to outer overlap portion, image 1 will get wrong. The 2-pixel noise probability which image 6 affects to image 1 to get wrong image 1, is $P_{(2\ pixels)}=\frac{9}{30}\ x\frac{8}{30}$ =8%. Similarly, the 2-pixel noise probability which image 8 affects to image 1 to get wrong image 1, is $P_{(2\ pixels)}=\frac{10}{30}\ x\frac{9}{30}$ =10%. At the same way, the 2-pixel noise probability which image 9 affects to image 1 to get wrong image 1, is $P_{(2\ pixels)}=\frac{4}{30}\ x\frac{3}{30}$ =1.3%. In summary, the 2-pixel noise probability which all images 2, 3, 5, 6, 8, 9 affect to image 1 to get right image 1 $P_{(2\ pixels)}$=100% - (18.7%.+8%+10%+1.3%)=62%.

Image 4 at least adds 3 noise pixels to reach V_{DD} voltage at same speed as image 1. Thus, image 4 just affects to image 1 in case of considering 3 noise pixels. So, the 3-bit error probability of image 4 which affects to image 1 is $(P_{(1)} \times P_{(2)} \times P_{(3)}) = \frac{4}{30}\, x \frac{3}{30}\, x \frac{2}{30} = 8.9 \times 10^{-4}$. Among 10 images, there are 6 images including 2, 3, 5, 6, 8, and 9 that only 2 out of 3 noise pixels may get incorrect recognition for 3-noise pixel case. $P_{(3\ pixels)} = (P_{(1)}\, x\, P_{(2)}\, x\, P_{(3)}) + P_{(2\ pixels)}$ =38.1%. As the result, the recognition probability for 3 noise pixels in image 1: $P_{(3pixels)}$= 100%- 38.1.%=61.9%. Similarly, image 0 adds at least 4 noise pixels that cause error for image 1.

In this analysis, image 1 is used as test image with free noise shown in Fig. 8.

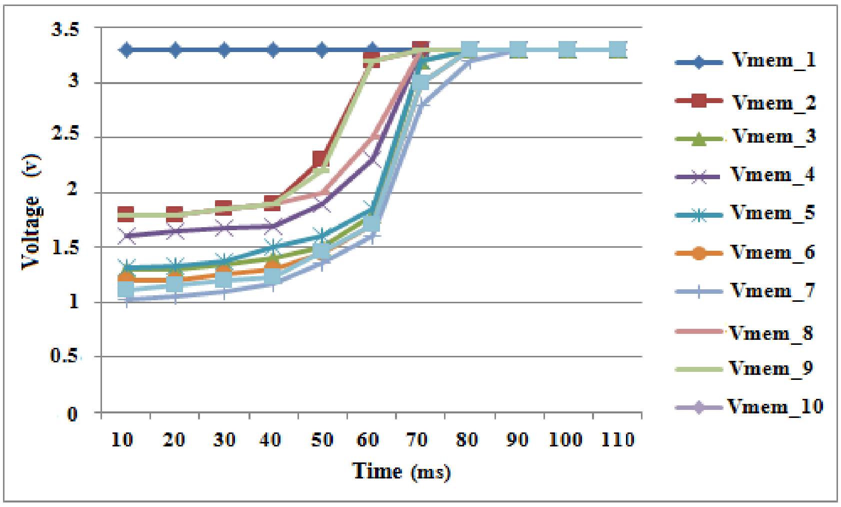

Figure 8. 10 Vmem signals with free noise pixel for memristor array.

Shown in fig. 8, the only Vmem1 signal reaches 3.3 volt the fastest during 60 ms, then other 9 Vmem signals reach 3.3-volt threshold later. As a result, neuron output that fires the earliest and fastest 1-level, identifies the desired image input.

The remaining images as explained in fig. 7. In case of image 0 and 7, these image pixels are a subset of image 8. Thus, only 1 noise pixel affects to these images 0 and 7, the recognition result will get wrong.

In case of two additive noise pixels, just adding two level-1 pixels to image 1, then image 1 and image 2 will be identical. This is because image 1 has eight 1-level pixels, image 2 has fourteen 1-level pixels. However, image 2 has six 1-level pixel positions overlapped by image 1 as shown in fig. 7. Thus, the image 1 only needs to be added two high level pixels at same black pixel positions of image 2, it causes duplication. The successful recognition probability of image 1 is probability of two additive noise pixels occurring at 8 black pixel positions of image 2 as shown fig. 7 and explained in above session.

Table 1 The comparison in identification capacity with two techniques in case of 2 additive noise pixels and free noise.

Recognition rate (%)	# 1	# 2	# 3	# 4	# 5	# 6	# 7	# 8	# 9	# 0
0 noise pixel	100	100	100	100	100	100	100	100	100	100
2 noise pixels	62	99.7	96.7	77.8	96.7	99.7	0	100	87.2	0

As seen in Table 1, the proposed technique which is constructed from Memristor array, can eliminate the noise pixels. The memristor array of the proposed technique identifies images of 2, 3, 5, 6, 8, giving the best results.

IV. CONCLUSION

The memristor array architecture are proposed to recognize the written characters. The controller circuit is designed to training the memristor on-chip simultaneously. The proposed technique improves the circuit in the direction of reducing the size of the control circuit while simultaneously extending capacity the additive noise. Image identification is accuracy up to 100% in case of free noise.

REFERENCES

[1]. M. Chu *et al*., Neuromorphic Hardware System for Visual Pattern Recognition With Memristor Array and CMOS Neuron, *IEEE Transactions on Industrial Electronics*, *62(4),* 2015, 2410-2419.

[2]. D. B. Strukov, G. S. Snider, D. R. Stewart, and R. S. Williams, The missing memristor found, *Nature, 453*, 2008, 80–83.

[3]. M. Di Ventra, Y. V. Pershin, and L. O. Chua, Circuit elements with memory: Memristors, memcapacitors, and meminductors, *Proc. IEEE*, *97(10)*, 2009, 1717–1724.

[4]. L. Chua, Resistance switching memories are memristors, *Appl. Phys. A, Mater. Sci. Process., 102(4),* 2011, 765–783.

[5]. Y. Ho, G. M. Huang, and P. Li, Nonvolatile memristor memory: Device characteristics and design implications, *Proc. IEEE/ACM ICCAD*, 2009, 485–490.

[6]. J. Seo et al, A 45nm CMOS Neuromorphic Chip with a Scalable Architecture for Learning in Networks of Spiking Neurons, *Proc. IEEE CICC*, 2011, 1-4.

[7]. M. Hu, H. Li, Y. Chen, Q. Wu, G. S. Rose and R. W. Linderman, Memristor Crossbar-Based Neuromorphic Computing System: A Case Study, *IEEE Transactions on Neural Networks and Learning Systems*, 25(10), 2014, 1864-1878.

[8]. C. R. Wu, W. Wen, T. Y. Ho and Y. Chen, Thermal optimization for memristor-based hybrid neuromorphic computing systems, *2016 21st Asia and South Pacific Design Automation Conference (ASP-DAC)*, 2016, 274-279.

[9]. S. N. Truong, K. Van Pham, W. Yang and K. S. Min, Sequential Memristor Crossbar for Neuromorphic Pattern Recognition, *IEEE Transactions on Nanotechnology*, 15(6), 2016, 922-930.

[10]. Truong SN, Ham SJ, Min KS, Neuromorphic crossbar circuit with nanoscale filamentary-switching binary memristors for speech recognition, *Nanoscale Res. Lett.*, 2014.

[11]. C. Zamarreño-Ramos, L. A. Camuñas-Mesa, J. A. Pérez-Carrasco, T. Masquelier, T. Serrano-Gotarredona, and B. Linares-Barranco, On spike-timing-dependent-plasticity, memristive devices, and building a self-learning visual cortex, *Frontiers Neurosci*., 5, 2011.

[12]. Yang Zhang , Xiaoping Wang , Eby G. Friedman, Memristor-Based Circuit Design for Multilayer Neural Networks, *IEEE Transactions on Circuits and Systems I, 65(2)*, 2018, 677 – 686.