International Journal of Engineering Science Invention
ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726
www.ijesi.org ||Volume 5 Issue 5|| May 2016 || PP.11-17

Synaptic memristor bridge circuit with pulse width based programable weights in digit character recognition

Ngoc-ThanhLe[1], Minh-Huan Vo[2]
[1,2]*Department of Electrical ElectronicEngineering, HCMC University of Technology and Education, Viet Nam.*

ABSTRACT :*Memristor bridge circuit linearly generates synaptic weights in the range [-1; 1]. A long width pulse programs the synaptic weights and a short width pulse computes multiplication results in ANN model. Each digit is sampled by 10 times, forming 5x4 matrix. Thecharacter recognition rate achieves 100% for digits from 1 through 5 . 5%, 10%, and 15% noisy patterns is added to consider the recognition rate.*
Keywords: *Memristor bridge, pattern recognition, neural networks, synaptic weight, memristor.*

I. INTRODUCTION

An artificial neural network (ANN) is a model of computation inspired by neo-cortex of human brain. It is likely to solve many problems in applications of prediction, recognition, control and optimization [1]. We can also describe it as a network of connected synaptic neurons that is capable of creating, modifying, and preserving information through sequential learning procedures. Therefore, building a brain-like machine is very important. Because of the lack of proper devices which are used to implement the synapses, researches have been limited success in this area. Among them, the cellular neural network (CNN) [2]-[4] is appreciated to be one of the successful approaches. However, even for the CNN methods, the use of devices to achieve efficient energy and density is essential for building an artificial brain.

The memristor (memory resistor) was invented by Chua in 1971 as the fourth basic element of electrical circuits [5]-[6]. This device has a special ability to change its resistance when a current or voltage is applied to the terminals. When the current or voltage is stopped, this device is still holding the state at that moment without loss in a long time. Memristor has been extensively studied with serve as a memory or logic gate to improve production technology with fast speed chip, power-saving, low cost, simple structure and denser integrated level. Especially, the memory resistors are used as synapses to mimic the functions of a real brain [7]. Memristor is much smaller in size compared to previous technology. In 2008, Stanley Williams and his group realized memristor in the form of a device in laboratory practice HP [8].

In this paper, we use a synaptic bridge circuit using four memristors and three transistors for character recognition circuit. We use two kinds of square pulse: a large width pulse is used to program the synaptic weights and a small width pulse is used to compute the weight and input. Each digit from 1 to 5, sampled by 10 times, forms a 5x4 matrix to show simulation results.

II. MEMRISTOR MODEL OF HP

In 2008, Stanley Williams and his group realized memristor in the form of a device in laboratory HP. To create memristor, they used a thin titanium oxide film (TiO2) [8]. The film is connected with 2 poles made of platinum (Pt). One pole is injected with oxygen holes. This oxygen holes are positively charged ions. Therefore there is a transition layer of TiO2, with a side of mixed and one side is not mixed.

Let us define the thickness of the doped region w, D is the thickness of two layers TiO2 and RON, ROFF are low and high impedance respectively. Accordingly, the memristance M (t) of TiO2 memristor model is given by the formula:

$$M(t) = R_{ON}\,x(t) + R_{OFF}\,(1 - x(t)) \qquad (1)$$

Where $x(t) = \frac{w(t)}{D}$ is the state variable of memristor.

The relationship between the voltage $v(t)$ and current $i(t)$ of the memristor is given by:

$$v(t) = M(t).\,i(t) \qquad (2)$$

State variables x(t) is determined:

$$\frac{dx(t)}{dt} = \mu_v \frac{R_{ON}}{D^2} i(t) \quad (3)$$

Where μ_v is the dopant mobility. Velocity of width change is linearly proportional to the current as in equation (3). Thus, we call this model linear model.

Various types of nonlinear memristor model have been published. One of them in the window model in which the state equation is multiplied by window function $F_p(w)$,namely:

$$\frac{dx(t)}{dt} = \mu_v \frac{R_{ON}}{D^2} i(t) F_p(w) \quad (4)$$

Where p is a parameter and $F_p(w)$ is defined by:

$$F_p(w) = 1 - \left(2\frac{w}{D} - 1\right)^{2p} \quad (5)$$

This is called the nonlinear drift model. It means that as the p decreases, the nonlinearity increases. In other words, as the p increases, the model tends to the linear one.

III. RESULTS AND DISSCUSION

A. Memristor based weight setting

Memristor bridge synapse is a Wheatstone bridge that circuit consists of four identical memristors with the polarities shown in figure 1 [7]. Input signal is applied from the left end of the circuit and the output is taken from two middle nodes as a differential form. When strong pulse Vin is applied as input, the memristance of each memristor will increase or decrease which depends on its polarity. For instance, when a positive pulse is applied as input, the memristances of M1 and M4 will decrease. The other side, the memristances of M2 and M3 will increase. It follows that the voltage VA at node A, with respect to ground, becomes smaller than the voltage VB at node B. Since, the node voltage VA is less than VB, the output voltage (Vout) across the bridge is negative weight.

The Vin is an input signal applied to the memristor bridge circuit as shown in Figure 1. The values in memristances are determined at time t. The output voltage will be divided according to memristance equations as follows:

$$V_{M1} = \frac{M1}{M1+M2} V_{in} \quad (6a)$$

$$V_{M2} = \frac{M2}{M1+M2} V_{in} = V_A \quad (6b)$$

$$V_{M3} = \frac{M3}{M3+M4} V_{in} \quad (6c)$$

$$V_{M4} = \frac{M3}{M3+M4} V_{in} = V_B \quad (6d)$$

Where M1, M2, M3 and M4 are the corresponding memristances of the memristors at a specific time t. Note that above equations for memristors are formulated by same principle for resistors.

The different voltage at output of A node and B node shows the output voltage as in equation 7, namely:

$$V_{out} = V_A - V_B = \left(\frac{M2}{M1+M2} - \frac{M4}{M3+M4}\right) V_{in} \quad (7)$$

From (7), the voltage Vout is given as,

$$V_{out} = \Psi \times V_{in} \quad (8)$$

Where $\Psi = \frac{M2}{M1+M2} - \frac{M4}{M3+M4}$ represents the synaptic weight.

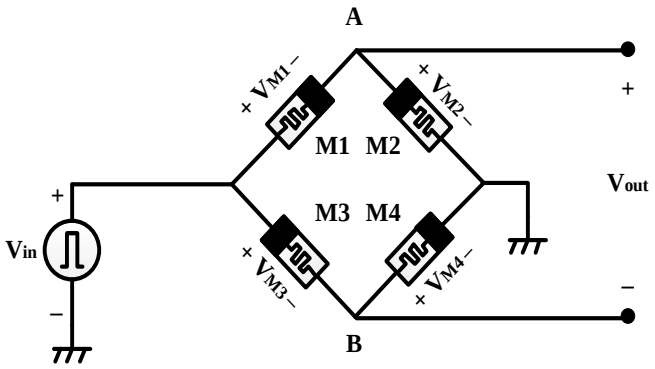

Fig 1: Synaptic memristor bridge circuit. The synaptic weight is programmable by varying the input pulse width Vin. The multiplication of the input signal and synaptic weght is also performed in this circuit.

We can note that the sign of synaptic weights depends on Ψ value. If Ψ is larger than 0, the synaptic weight is positive. It follows the below equation:

$$\frac{M2}{M1+M2} - \frac{M4}{M3+M4} > 0 \qquad (9)$$

It means that our memristor bridge circuit create a positive synaptic weight. From equation (9), we can conclude that the condition for positive, negative and zero synaptic weight is as follow,

$$\Psi = \begin{cases} \text{Positive} & \text{if} \quad \frac{M2}{M1} > \frac{M4}{M3} \\ \text{Negative} & \text{if} \quad \frac{M2}{M1} < \frac{M4}{M3} \\ 0 & \text{if} \quad \frac{M2}{M1} = \frac{M4}{M3} \end{cases} \qquad (10)$$

As shown in figure 1, the Vin is shared to use both the synaptic weight program signal and synaptic input. The two different kinds of signals are transferred in same Vin signal with different time slots. Note that the synaptic multiplication is used at short width pulses with negligible effect on memristance variation while synaptic weights are programmed for long width pulses.

B. Memristor based neuron circuit

The differential amplifier consists of three transistors as shown in figure 2 which converts the voltage to current. This converter provides interface to the neuron for summing input signals easily. The set of weighted input signals is summed by current mode circuit. As shown in figure 3, the 20 input signals from 5X4 pixel matrix are multiplied by programmed synaptic weights.

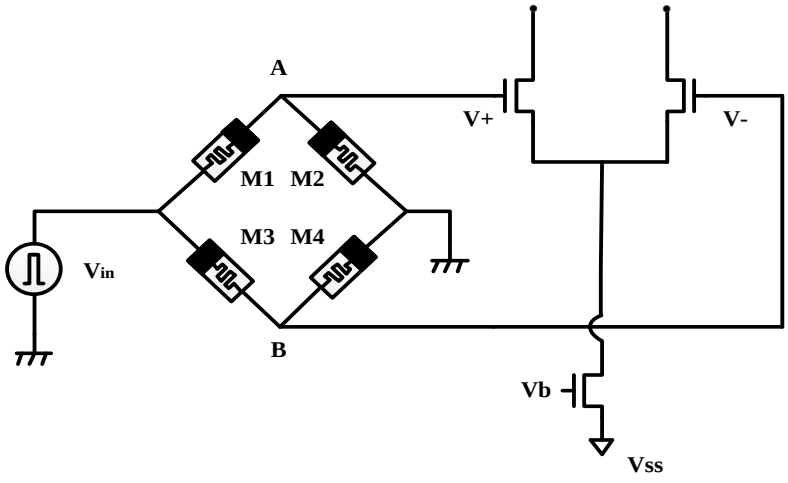

Fig 2: Synaptic memristor bridge circuit.

The circuit of the memristor synapse-based neuron using memristor bridges and differential amplifiers is shown in figure 3. Here, the synapse based neuron for digit 1 consists of 20 inputs from in1 to in20. The synaptic multiplications are conducted in the memristor bridges. Then multiplicationresults from Io_1 to Io_20 are summed at the output terminal in a neuron cell. Then, the current summation is converted back into a voltage by the load circuit RL. In this paper, we have five circuits corresponding to five digits.

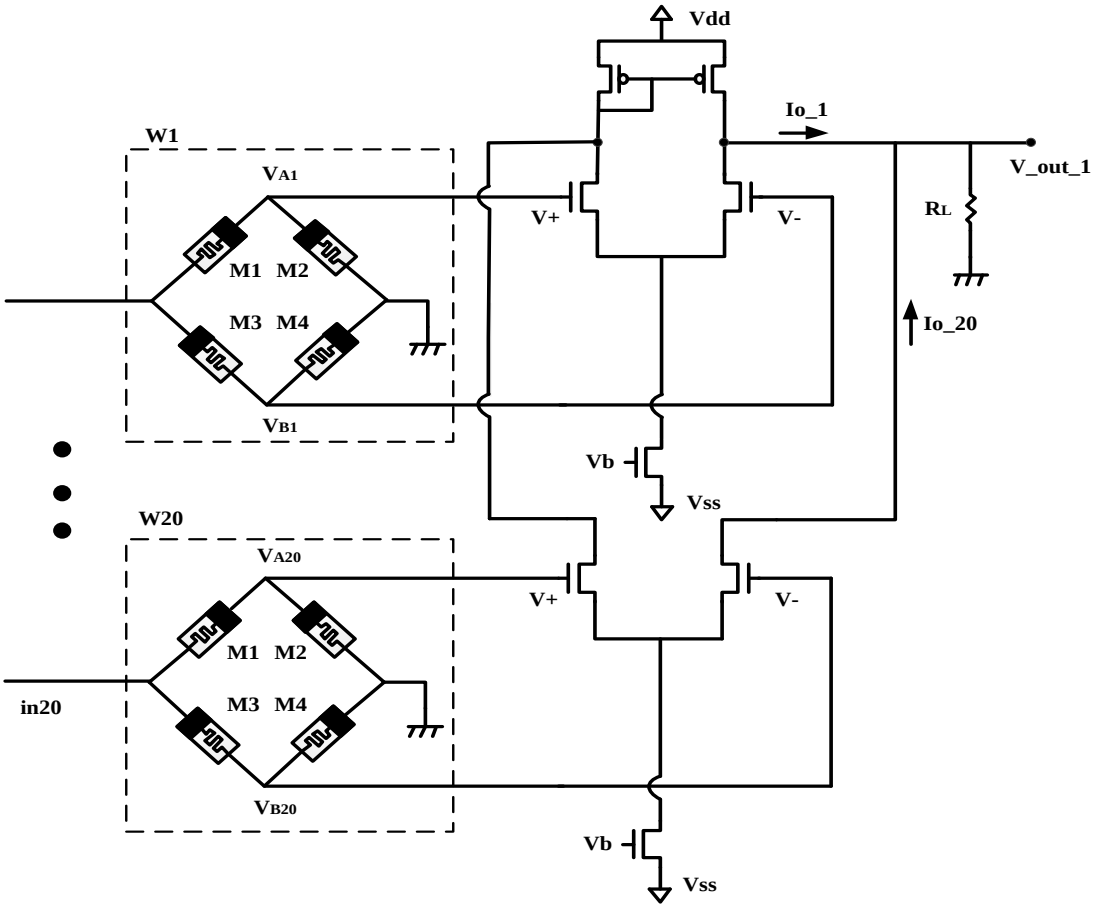

Fig 3: Memristor based neural circuit with digit 1.Weights W1 to W20 are programmed by inputs of in1 to in20, respectively.

IV. SIMULATION

The HP TiO2 memristor model is used for this simulation. The basic parameters for the simulations are based on the data given by HP, where RON=100Ω, ROFF=16 KΩ, qmax = 6e-5 C.

In the weight setting, a long width pulse is applied to change the state of memristor and a narrow pulse for synaptic multiplication is applied to avoid the drifting the memristor state. The weight setting of the architecture is verified through computer simulations and synaptic multiplication. Memristor synapse based neural circuit is performed in Cadence tool.

A. Synaptic weight programming in memristor bridge circuit

In order to set synaptic weight, a long width pulse of 1V amplitude is applied to change the memristance state of memristor. Figure 4(a) shows the increase or decrease in the memristances M1(t), M2(t), M3(t) and M4(t) as a function of time. These simulations of the memristor bridge circuit are obtained in figure 1 with initial memristances; M1(0) = M4(0) = 16 KΩ, M2(0) = M3(0) = 100 Ω. The characteristic of memristance changes shows linearly according to different times in figure 4 (a). Figure 4(b) is the corresponding weight computed with the memristance values in figure 4(a). As shown in the figures, changes in these computed memristances in figure 4(a) and the corresponding weight in figure 4(b) are very linear. It means that the weight setting can be programmed linearly in response of various times.

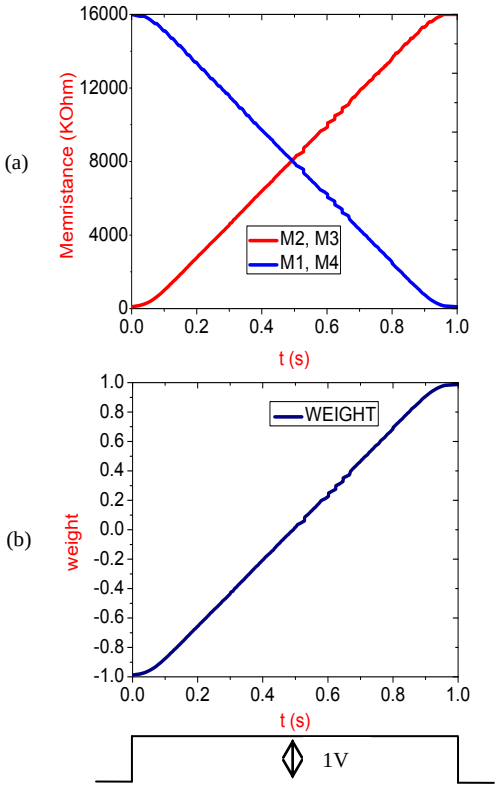

Fig 4: M1(t), M2(t), M3(t), M4(t) and weight □ (t) with ourmemristor bridge circuit at various width pulses. The linear memristor model is applied to this simulation. The initial memristances are M1(0) = M4(0) = 16 K□ , M2(0) = M3(0) = 100 □ . (a) M1(t), M2(t), M3(t) and M4(t) as a linear function of time. (b) Weight □ (t) covers both negative, zero, positive sign.

B. Character recognition application

In this paper, we identify each digit from 1 to 5, sampled by 10 times. The samples are taken into Matlab to train and create the weights corresponding to each pixel position on each digit. Then, we set these synaptic weights into memristor bridge to be multiplied with the input signal.

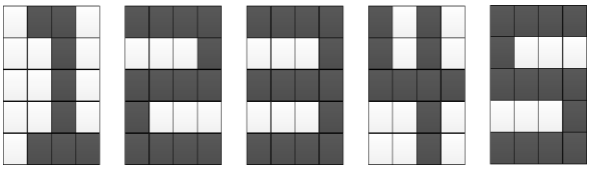

Fig 5: Training images for digits from 1 to 5

Figure 5 shows the prototype of the five digits. Each digit in addition to 9 other samples is put into Matlab for training. The more samples are, the higher the training is accuracy. Here we only train 10 samples with each digit. The number of samples puts into training to increase the precision of the actual application.

Each sample in Figure 5 is a 5x4 sized character matrix and each weight will have a memristor bridge synaptic circuit, so we will have 20 synaptic memristor bridge circuits.

Figure 6 shows the results of simulation output voltage of each digit from 1 to 5. The results show that we can identify the output digit based on the output voltage level. Among five output voltages, the digit can be recognized by setting a threshold voltage level.

Figure 7 describes the character in interference. We take character digit 1 as an example here. The percentage of added noise in Table I shows the recognition rate when there is interference. Recognition accuracy will reduce to 90%, 80%, 60% in case of 5%, 10%, and 15% noise, respectively.

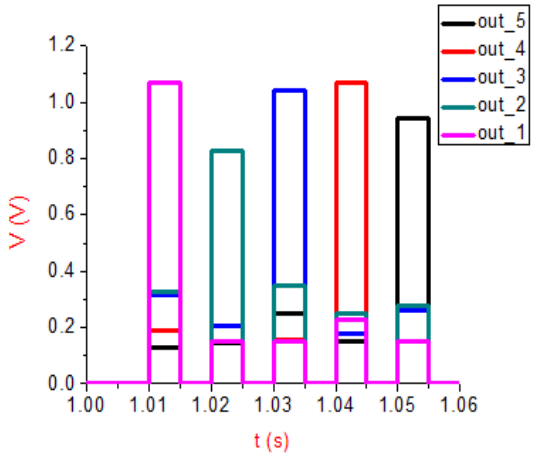

Fig 6: The outputs of 5 integrators.

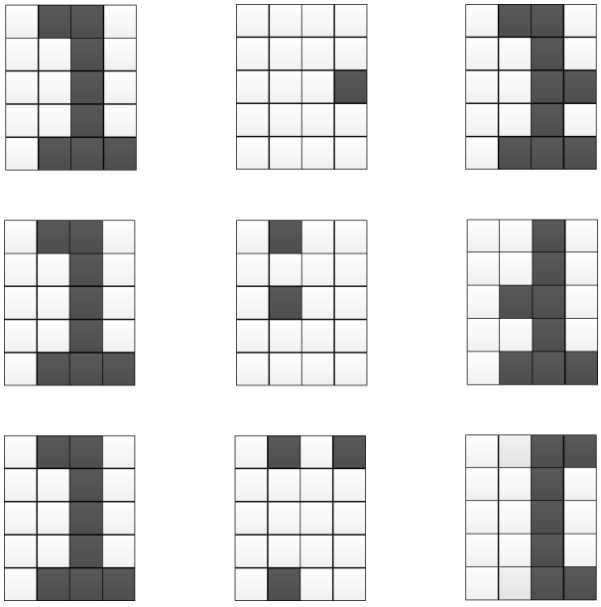

Fig 7: Noisy patterns used for recognition test at 5%, 10%, 15%.

TABLE I. Recognition rates for various noisy patterns

Noise	**Recognition rate**				
	No. 1	**No. 2**	**No. 3**	**No. 4**	**No. 5**
0%	100%	100%	100%	100%	100%
5%	90%	85%	90%	85%	85%
10%	80%	70%	80%	80%	75%
15%	60%	50%	55%	55%	50%

V. CONCLUSION

In this paper, we propose memristor bridge circuit with programmable synapses based pulse width which is able to set signed synaptic weights. Weights are programmed with long width pulses and synaptic multiplications are performed with short width pulses through a single input line. The recognition rates achieve 100% success for characters numbered 1 through 5. Each character has 10 samples and each sample is a 5x4 matrix. We tested 500 samples for each digit. In case of digit 1, recognition accuracy will reduce to 90%, 80%, 60% according to 5%, 10%, and 15% noise, respectively.

REFERENCES

[1]. The Sientific American Book of the Brain. New York: Scientific American, 1999.
[2]. L. O. Chua and L. Yang, "Cellular neural networks: Theory," IEEE Trans. Circuits Syst., vol. CAS-35, no.10, pp. 1257 – 1272, Oct. 1988.
[3]. L. O. Chua and L. Yang, "Cellular neural networks: Applications," IEEE Trans. Circuits Syst., vol. CAS-35, no.10, pp. 1273 – 1290, Oct. 1988.
[4]. J. M. Cruz and L. O. Chua, "A 16x16 cellular neural network universal chip: the first complete signle-chip dynamic computer array with distributed memory and with gray-scale input-output," Analog Integrated Circuits and Signal Processing, vol. 15, no. 3, pp. 227 – 237, 1998.
[5]. L. O. Chua, "Memristor – the missing circuit element," IEEE Trans. Circuit Theory, vol. 18, no. 5, pp. 507 – 519, 1971.
[6]. L. O. Chua and S. M. Kang, "Memristive devices and systems," Proc. of IEEE, vol. 64, no. 2, pp. 209 – 223, 1976.
[7]. Hyongsuk Kim, Maheshwar Pd.Sah, CH. Yang, T. Roska, and Leon Chua, "Memristor bridge synapses", proceedings of the IEEE, vol. 10, pp. 20.
[8]. Dmitri B. Strukov, Gregory S. Snider, Duncan R. Stewart & R. Stanley Williams, "The missing memristor found," Nature, vol. 453, May. 2008.