ELectronics EXpress

LETTER

IEICE Electronics Express, Vol.11, No.24, 1–6

An adaptive neural network A/D converter based on CMOS/memristor hybrid design

Weiwei Wang, Zhiqiang You[a)], Peng Liu, and Jishun Kuang

College of Computer Science and Electronic Engineering, Hunan University, 410082, China

a) *you@hnu.edu.cn*

Abstract: A memristor is regarded as a promising device for modeling synapses in the realization of artificial neural systems for its nanoscale size, analog storage properties, low energy and non-volatility. In this letter, an adaptive T-Model neural network based on CMOS/memristor hybrid design is proposed to perform the analog-to-digital conversion without oscillations. The circuit is composed of CMOS neurons and memristor synapses. The A/D converter (ADC) is trained by the least mean square (LMS) algorithm. The conductance of the memristors can be adjusted to convert input voltages with different ranges, which makes the ADC flexible. Using memristors as synapses in neuromorphic circuits can potentially offer high density.

Keywords: analog-to-digital, neural network, memristor, hybrid circuit

Classification: Integrated circuits

References

[1] D. B. Strukov, G. S. Snider, D. R. Stewart and R. S. Williams: Nature **453** (2008) 80. DOI:10.1038/nature06932

[2] B. Govoreanu, G. S. Kar, Y. Chen, V. Paraschiv, S. Kubicek, A. Fantini, I. P. Radu, L. Goux, S. Clima and R. Degraeve: IEDM (2011) 31.6.1. DOI:10.1109/IEDM.2011.6131652

[3] T. Serrano-Gotarredona, T. Prodromakis and B. Linares-Barranco: IEEE Circuits Syst. Mag. **13** (2013) 74. DOI:10.1109/MCAS.2013.2256271

[4] T. Oya, T. Schmid, T. Asai, Y. Leblebici and Y. Amemiya: IEICE Electron. Express **2** (2005) 76. DOI:10.1587/elex.2.76

[5] A. Afifi, A. Ayatollahi and F. Raissi: IEICE Electron. Express **6** (2009) 148. DOI:10.1587/elex.6.148

[6] X. Zhu, X. Yang, C. Wu, J. Wu and X. Yi: IEICE Electron. Express **10** (2013) 20130404. DOI:10.1587/elex.10.20130404

[7] D. Tank and J. J. Hopfield: IEEE Trans. Circuits Syst. **33** (1986) 533. DOI:10.1109/TCS.1986.1085953

[8] O. Ishizuka, Z. Tang, T. Inoue, H. Matsumoto and S. Ohba: IJCNN (1991) 2288. DOI:10.1109/IJCNN.1991.170729

[9] L. Gao, F. Merrikh-Bayat, F. Alibart, X. Guo, B. D. Hoskins, K.-T. Cheng and D. B. Strukov: Proc. Nanoarch (2013) 19. DOI:10.1109/NanoArch.2013.6623031

[10] M. Soltiz, D. Kudithipudi, C. Merkel, G. S. Rose and R. E. Pino: IEEE Trans. Comput. **62** (2013) 1597. DOI:10.1109/TC.2013.75

DOI: 10.1587/elex.11.20141012
Received October 24, 2014
Accepted October 27, 2014
Publicized November 28, 2014
Copyedited December 25, 2014

[11] A. Afifi, A. Ayatollahi and F. Raissi: ECCTD (2009) 563. DOI:10.1109/ECCTD.2009.5275035

1 Introduction

Scientists from HP labs discovered a nanoscale device called the memristor in 2008 [1]. The memristor resistance changes with the electrical charge passing through it, which means that the memristor can store information based on the past history of its activation. The recent specification for metal-oxide memristor includes sub-10-nm size, excellent endurance, pico-Joule and sub-ns switching [2]. Memristors could provide the compact synapses required to a neural network circuit [3]. Hybrid CMOS-nano architectures provide high density for developing fault and defect tolerant neuromorphic networks [4]. Using CMOS/memristor hybrid architecture and a two-part spike, Afifi et al. demonstrated Spike-Timing-Dependent-Plasticity (STDP) learning implementation [5]. Zhu et al. proposed an approach for implementing Hamming network based on hybrid CMOS/memristor circuit design [6], in which pattern recognition and classification were demonstrated.

Neural network approach for an analog-to-digital converter (ADC) is significant for high resolution and flexibility compared to the conventional design. The ADC neural network in [7, 8] is composed of inter-connected resistors with large area, which would become an impediment to the implementation with very large scale integration (VLSI) technology. Moreover, once the input voltage range is changed, the ADC will not work. A 4-bit ADC based on Hopfield neural network is implemented with a CMOS/memristor hybrid circuit [9]. However, all neurons in the ADC must be reset periodically to 0 to avoid getting stuck in local minima, which increases the complexity of signal conversion.

In this letter, an adaptive 4-bit T-model neural network ADC based on CMOS/memristor hybrid design is proposed. The ADC is capable of finding the global optimum solution without local minima. It can be compact due to high density of analog weights implemented with memristive devices. As to the different input voltage ranges, the corresponding connection weights are different. The ADC can be trained with least mean square (LMS) algorithms, which makes the A/D converter flexible.

2 Memristor model

The memristor is composed of undoped region and doped region with oxygen vacancies. R_{on} and R_{off} are the lowest and highest resistances of the memristor respectively. We adopt a piecewise linear memristor model [10]. The model function is as follows:

$$M = \begin{cases} M - \frac{(\delta r \times \delta t \times V_m)}{(t_{pos} \times V_{th,pos})} & V_m \geq V_{th,pos} \\ M + \frac{(\delta r \times \delta t \times V_m)}{(t_{neg} \times V_{th,neg})} & V_m \leq V_{th,neg} \end{cases} \quad (1)$$

where δr is $R_{off} - R_{on}$, δt is the minimum time step interval, V_m is the voltage applied to the device, $V_{th,pos}$ is the positive voltage threshold of the device, $V_{th,neg}$ is

DOI: 10.1587/elex.11.20141012
Received October 24, 2014
Accepted October 27, 2014
Publicized November 28, 2014
Copyedited December 25, 2014

the negative voltage threshold of the device, t_{neg} is the time required to increase the memristance from R_{on} to R_{off}, and t_{pos} is the time required to decrease the memristance from R_{off} to R_{on}.

Synapses in brains behave similarly to memristive devices, prompting the method to take advantage of memristors in neuromorphic hardware [11]. Storing and updating the connection weights of synapse based on synaptic plasticity rules is one of the most troublesome operations in biologically-inspired neural networks. Memristors make it possible to update the connection weights efficiently. In this letter, we use the memristive conductance corresponding to the synapse weight.

3 Adaptive A/D converter

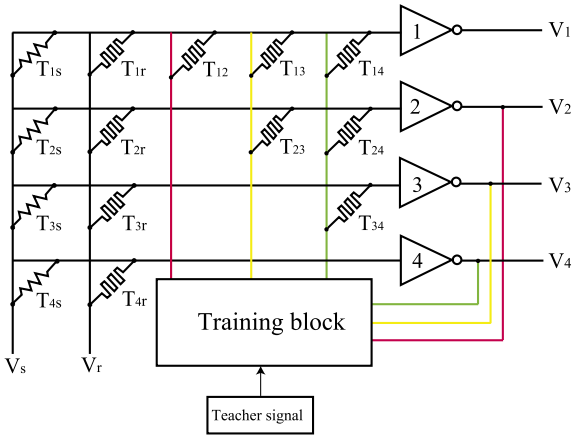

Fig. 1. Schematic of a 4-bit T-Model neural network ADC.

In this section, we propose a T-Model neural network ADC. The circuit schematic diagram is shown in Fig. 1. The ADC is based on CMOS/memristor hybrid design and consists of 4 amplifier neurons, 6 memristor connections, 4 bias memristors, 4 input resistors and a training block. Neurons are realized in CMOS layer and synapses are realized by memristor crossbar. Voltage V_s is the numerical value to be converted. Voltage V_r feeds different constant bias I_i into the corresponding neuron through memristor T_{ir}. A connection between neurons is determined by a memristor conductance T_{ij} which connects the output of neuron j to the input of neuron i. The outputs ($V_4V_3V_2V_1$) represent the binary value of the input voltage V_s. V_i ($i = 1, 2, 3, 4$) is read out as the 0 or 1 values of the amplifier output voltages.

A variety of learning algorithms can be used to train the T-Model neural network ADC. We adopt the LMS algorithm [8]. The algorithm minimizes the error function:

$$E = \frac{1}{2}\sum_{\{p\}}(y_p - t_p)^2 \quad p = 1, \cdots, N \tag{2}$$

where t_p is teacher signal, and y_p is actual output signal. Each neuron input includes three different set of signals: other neuron outputs y_j ($j > i$), the corresponding analog input and reference bias. After the process of neuron i, the output is as follows:

DOI: 10.1587/elex.11.20141012
Received October 24, 2014
Accepted October 27, 2014
Publicized November 28, 2014
Copyedited December 25, 2014

$$y_i = f\left(\sum_{j>i} T_{ij} y_j + T_{is} V_s + T_{ir} V_r\right) \tag{3}$$

The changes of weight can be formulated as:

$$\Delta T_{ij} = -\beta \frac{\partial E}{\partial T_{ij}} = -\beta \frac{\partial E}{\partial y_i} \cdot \frac{\partial y_i}{\partial T_{ij}} \tag{4}$$

where β is a learning rate. The connection weights T_{ij} are realized by memristors with conductance values G_{ij}. G_{ij} should be adjusted according to Eq. (4). The output y_i and teacher signal t_i are either 1 or 0. When t_j is 1, t_i is 1 and y_i is 0, G_{ij} will increase. When t_j is 1, t_i is 0 and y_i is 1, G_{ij} will decrease. In other cases, G_{ij} will remain unchanged. As a result, the change of memristors' conductance G_{ij} is as follows:

$$\Delta G = \begin{cases} \beta & when\ t_j = 1,\ t_i = 1,\ y_i = 0 \\ -\beta & when\ t_j = 1,\ t_i = 0,\ y_i = 1 \\ 0 & others \end{cases} \tag{5}$$

When training the memristor, we will apply a voltage V_{train} which is higher than the threshold voltage. The time step interval of V_{train} is:

$$\delta t = \begin{cases} \frac{t_{pos} \times V_{th,pos}}{\delta r \times \beta \times V_{train}} & when\ t_j = 1,\ t_i = 1,\ y_i = 0 \\ \frac{t_{neg} \times V_{th,neg}}{\delta r \times \beta \times V_{train}} & when\ t_j = 1,\ t_i = 0,\ y_i = 1 \\ 0 & others \end{cases} \tag{6}$$

The training algorithm for the ADC is shown in Algorithm 1.

Algorithm 1: Training Algorithm for the memristive ADC

1. $\beta \leftarrow$ Learning rate
2. $f()$ is modelled as a digital comparator
3. $T_{ij} \leftarrow$ small randomly generated value
4. $x_p \leftarrow$ randomly generated learning input signal
5. **for** each input x_p **do**
6. **while** error function $E > E_{threshold}$ **do**
7. $\Delta T_{ij} \leftarrow \beta(t_i - y_i)t_j$
8. **if** $\Delta T_{ij} = \beta$ **then**
9. apply V_{train} for δt under the memristor T_{ij}
10. **end if**
11. **if** $\Delta T_{ij} = -\beta$ **then**
12. apply $-V_{train}$ for δt under the memristor T_{ij}
13. **end if**
14. $E = \frac{1}{2}\sum_{\{p\}}(y_p - t_p)^2$
15. **end while**
16. **end for**

4 Experiments and analysis

The simulation is done in MATLAB simulink. The parameters of the memristor model are as follows: $R_{on} = 0.1\,\mathrm{M\Omega}$, $R_{off} = 20\,\mathrm{M\Omega}$, $V_{th,pos} = 1.25\,\mathrm{V}$, $V_{th,neg} = -1.20\,\mathrm{V}$, $t_{pos} = 5\,\mathrm{ms}$, $t_{neg} = 1\,\mathrm{ms}$.

DOI: 10.1587/elex.11.20141012
Received October 24, 2014
Accepted October 27, 2014
Publicized November 28, 2014
Copyedited December 25, 2014

Table I. The teacher signal when input voltage range is [0, 15] V

X	t4	t3	t2	t1	X	t4	t3	t2	t1
$0 \le x < 1$	0	0	0	0	$8 \le x < 9$	1	0	0	0
$1 \le x < 2$	0	0	0	1	$9 \le x < 10$	1	0	0	1
$2 \le x < 3$	0	0	1	0	$10 \le x < 11$	1	0	1	0
$3 \le x < 4$	0	0	1	1	$11 \le x < 12$	1	0	1	1
$4 \le x < 5$	0	1	0	0	$12 \le x < 13$	1	1	0	0
$5 \le x < 6$	0	1	0	1	$13 \le x < 14$	1	1	0	1
$6 \le x < 7$	0	1	1	0	$14 \le x < 15$	1	1	1	0
$7 \le x < 8$	0	1	1	1	$15 \le x < 16$	1	1	1	1

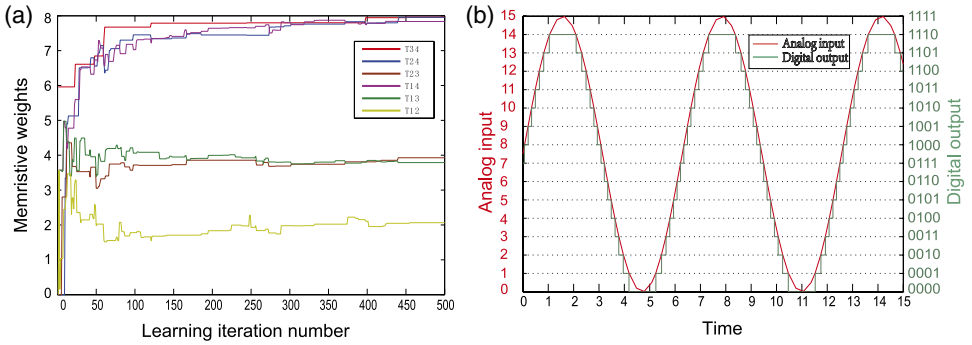

Fig. 2. The A/D converter training. (a) Relationship between memristive weights and the learning iteration number; (b) simulation results for [0, 15] V

We set the converted input voltage range [0, 15] V, Table I gives the ideal input-output characteristics for the ADC. The learning rate $\beta = 0.01$, the number of randomly generated learning input is 500 and $E_{threshold} = 1.0 \times 10^{-4}$. The memristive synapse initial weights are generated randomly. Then they are adjusted as shown in Fig. 2(a) and can reach a constant ultimately according to Algorithm 1. The input voltage and the corresponding digital code are shown in Fig. 2(b).

For the converted input voltage range [0, 3) V, we set the learning rate $\beta = 0.01$, the number of randomly generated learning input is 300, the error threshold $E_{threshold} = 1.0 \times 10^{-4}$. The memristive synapse weights are adjusted according to the LMS algorithm as shown in Fig. 3(a). Fig. 3(b) shows the input voltage and the corresponding digital code.

Table II compares the proposed ADC with those in [7], [8] and [9]. As to different input voltage ranges, the corresponding connection weights are different. The proposed ADC can be trained to convert input voltages with different ranges according to Algorithm 1. The size of a memristor is very small and with an area of less than $10 \times 10\,nm^2$. A hybrid circuit consisting of conventional CMOS circuits and memristive crossbar integrated on top of CMOS layer provides high density for developing networks. The Hopfield networks have local minima and the outputs do not correspond to the analog input [7]. The T-model neural network can be guaranteed of finding the global optimum solution. The neurons don't need to reset periodically to 0 to avoid getting stuck in local minima.

DOI: 10.1587/elex.11.20141012
Received October 24, 2014
Accepted October 27, 2014
Publicized November 28, 2014
Copyedited December 25, 2014

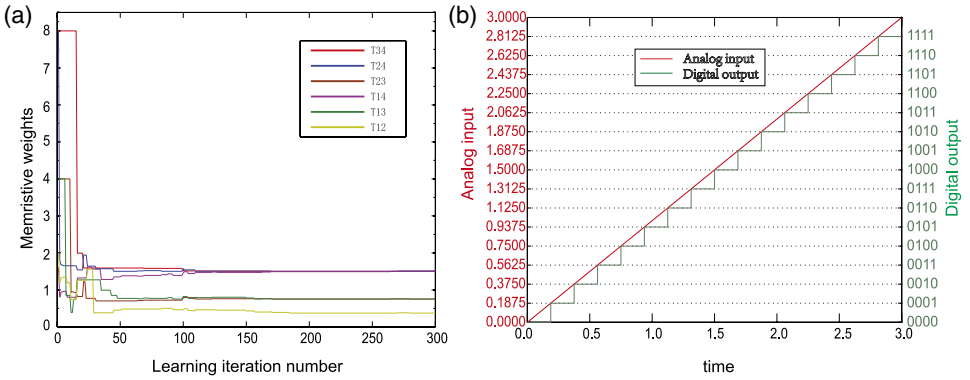

Fig. 3. The A/D converter training. (a) Relationship between memristive weights and the learning iteration number; (b) simulation results for $[0, 3)$ V

Table II. Comparison of the proposed ADC with previous works

	Ref [7]	Ref [8]	Ref [9]	This letter
Synapse	12 resistors	6 resistors	12 memristors	6 memristors
Global optimal	No	Yes	No	Yes
Flexibility	No	No	Yes	Yes

5 Conclusion

In this letter, we demonstrate a 4-bit ADC which is implemented with a T-model neural network. The ADC is based on CMOS/memristor hybrid design and consists of conventional CMOS circuits and memristive crossbar integrated on top of CMOS layer. It could be very compact potentially. The memristor synapse in the ADC can be adjusted along with the input voltage range, which makes the proposed ADC flexible. The simulation results in MATLAB show that the outputs correspond to the correct behavior of the ADC and have no problem of oscillations.

Acknowledgments

This work is supported by Program for New Century Excellent Talents in University (NCET-12-0165).

DOI: 10.1587/elex.11.20141012
Received October 24, 2014
Accepted October 27, 2014
Publicized November 28, 2014
Copyedited December 25, 2014