Design and Simulation of an Infinite Impulse Response (IIR) Filter with Memristor

Ahmad Maleki[1], Vahid Rashtchi[2*], Jalil Mazloum[3]
1, 2- Department of Electrical and Computers Engineering, University of Zanjan, Zanjan, Iran.
Email: rashtchi@znu.ac.ir (Corresponding author)
3- Department of Electrical Engineering, Shahid Sattari Aeronautical University of Science and Technology, Tehran, Iran.
Email: j_mazloum@sbu.ac.ir

Received: November 2017 Revised: January 2018 Accepted: February 2018

ABSTRACT:
In this paper, a novel circuit for memristor based IIR filter implementation is presented. In this research for increasing the input voltage range in sampling the analog signal, complementary switches were used instead of single-transistor switches. In addition, to increase the filter accuracy, a delayed circuit with the ability to implement high-order filters is presented. In this work, coefficients of the IIR filter were implemented by memristor; using such component could provide in-system reconfigurability. The memristor could decline receiving negative values, where IIR filter coefficients have negative values. In this research a new method for generating negative numbers as filter coefficients is presented. During running an advanced search algorithm, the memristor values were set at six, seven, and eight bits of resolution; these values cause memristors have the lowest error rate in generating coefficients. All circuits were simulated by Cadence tools on TSMC 0.18 micrometer technology platform with 1.8 volt power supply. In simulation results, outputs of low/high-pass filters along with the error rate of coefficients calculated and compared to actual coefficients.

KEYWORDS: Delay Circuit, Generating Negative Coefficients, IIR Filter, Memristor, Sampling Circuit.

1. INTRODUCTION

After the digital processors were introduced into the market, the analog operations on the signal processing were gradually dimmed. Every year, the technology of transistor manufacturing becomes more advanced and its speed rises. This trend has made everyone digitize their digital signals as much as possible and process it in the digital world [1], [2]. In recent years, this trend has continued due to ever-increasing advances in the digital world. Once the transistor manufacturing technology arrived at the point where scaling was so difficult; this idea was created in the mind of scientists that some of the processes might be possible in analog topologies with a higher performance compared to digital processing circuits [3].

In this regard, a part of the heavy digital operation could be removed from the digital processors and be executed in parallel in the analog domain. This scheme attempts to build or enhance digital structures which not only their execution within digital processors consumes too much energy, but also they could be implemented using pure analog circuits. There are various methods to implement filters; practical and well-known topologies are Finite Impulse Response (FIR) and Infinite Impulse Response (IIR) [4,5]. Therefore, depending on the application, we can use both types of filters. Today, FIR and IIR filters are used in the processing of audio [6-8], image [9], [10] and video [11], [12] signals.

In this work, the research and design are based on the FIR filter designed with memristor in [13]. Until now, the IIR filter has not been implemented with the memristor. Hence, in this paper, we proposed a new approach to implement direct form IIR filters using analog components and memristors. In addition, in [14], the memristor-based FIR filter was implemented on the electronic board using discrete elements, practically. Thus, it does not have the performance of the CMOS circuit.

IIR filters have been widely used in different applications with both analog and digital implementations. In [15], [16], an analog IIR filter has been used in a Delta-Sigma modulator; the circuit is implemented with analog adders and multipliers. In [17], [19], a digital IIR filter has been implemented on FPGA. In the above-mentioned and majority of works, all IIR filters have been implemented based on multipliers and adders. But, in this paper, only memristors and operational amplifiers have been utilized to implement IIR filters coefficients. Thus, the

proposed circuit in this work has a better speed gain over other works due to its simple structure. Because, to the best of our knowledge, the proposed topology for IIR filters implementation is novel and no one has followed this method, parametric comparison between this work and the previous works is not easily available.

As mentioned, this paper focuses on the design and simulation of an IIR filter using memristor. There is necessarily a negative value in the IIR filter while memristors could not get negative values. Thus, a circuit is proposed for implementing negative coefficients. In addition, the input voltage range has been increased by selecting the appropriate sampling circuit.

The rest of this paper is organized as the following. In section 2, different structures of IIR filter are investigated and the general discussion of the memristor is expressed. The direct form second-order section (SOS) IIR filter with memristor are presented in section 3. Simulations and their results are all covered in section 4. Finally, in section 5, the result of this work will be explained.

2. IIR FILTER THEORY

The general differential equation of IIR filter is determined from equation (1).

$$\sum_{l=0}^{N} a_l y[n-l] = \sum_{k=0}^{N} b_k x[n-k] \quad (1)$$

If $a_0 = 1$ and $y[n]$ is the first output, then equation (1) is converted to equation (2).

$$y[n] = \sum_{l=1}^{N} a_l y[n-l] + \sum_{k=0}^{N} b_k x[n-k] \quad (2)$$

In equation (2), a_l and b_k are the feed-forward part coefficients and the feed-back part coefficients, respectively. $x[n]$ represents the input and $y[n]$ is the output of the filter. It is clear that for calculating each new output 2N + 1 multipliers are required [20]. For example, an IIR filter with 7 orders needs 15 multipliers, which is time-consuming for a digital filter with such an order. Therefore, for this volume of computing it is better to do it with an analog circuit along with the processors in a simple way.

2.1. Direct Form IIR Filter type I and type II

The direct form IIR filter type I is shown in Fig. 1 which consists of two feed-back and feed-forward parts.

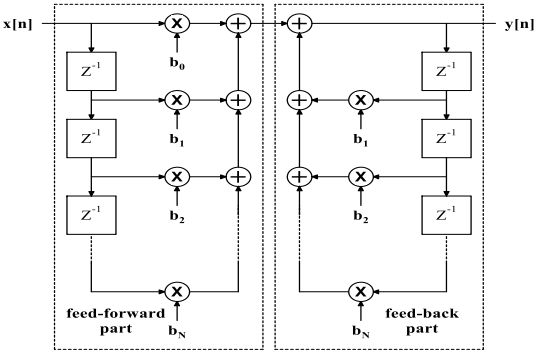

Fig. 1. Direct Form IIR Filter type I.

feed-back and feed-forward parts operate in series and independently.

According to equation (3), if the z-transform is taken from equation (2), the conversion function of the IIR filter is obtained.

$$H(z) = \frac{\sum_{k=0}^{M} b_k Z^{-k}}{1-\sum_{l=1}^{N} a_l Z^{-l}} = \frac{b_0+b_1 Z^{-1}+\cdots+b_M Z^{-M}}{1-a_1 Z^{-1}-\cdots-a_N Z^{-N}} \quad (3)$$

Nominator and denominator polynomial coefficients represent the coefficients of feed-forward and feed-back parts, respectively [20].

As shown in Fig. 2, the direct form of IIR filter type II is obtained by combining the feed-forward and feed-back parts.

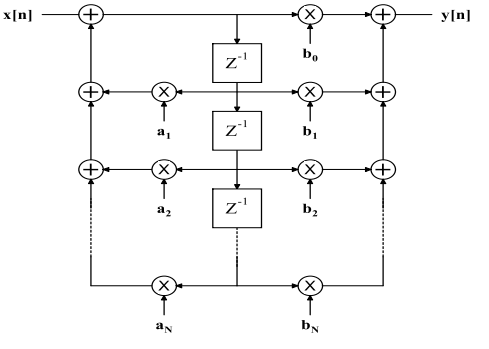

Fig. 2. Direct Form IIR Filter type II.

It can be seen that the number of delayed blocks in the direct form type II is half of the direct form type I.

To realize these circuits first, using a sampling circuit with a sampling period of T seconds, the input is sampled periodically. Using cascaded samplers, required delays are implemented. Also, using resistors and operational amplifiers, we can implement the coefficients according to Fig. 3.

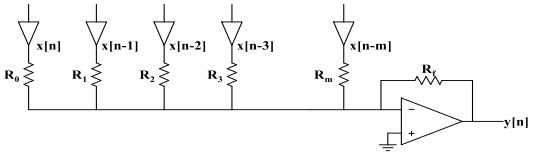

Fig. 3. Implementation of IIR filter coefficients with the resistor.

In Fig. 3, the coefficients will not change if they are implemented with resistors. For solving this problem, we can use the Memristors (as configurable resistors) to implement coefficients. By using memristors, the filter could be adapted to any application by setting memristor's conductance to different values, this also causes increasing the reconfigurability of filters. The memristor is explained below.

2.2. Memristor

Memristors were devised in 1960 by Widrow as a 3-terminal element to be used in adaptive circuits; its resistance was adjustable by one of its terminals which receives DC current [21]. However, the introduced memristor could not act as the fourth circuit theory element as the IV characteristic of the element was not fixed and it could only be determined within the circuit which it has been used in [22]. Accordingly, in 1968, Fano and colleagues claimed that except resistor, inductor, and capacitor, there is another missing circuit element [23]. In 1971, an article titled "Memristor- The Missing Circuit Element" proposing the fourth circuit element was presented by Chua. He believed that except three main circuit elements (Resistor, Capacitor, and inductor), another element called "Memristor" exists which there is a function between its current flow and its conductance [24]. In 2008, researchers in HP laboratories implemented memristors physically [25]. The circuit model for a memristor is shown in Fig. 2. The current should be greater than a certain value, therefore, when the current passes less than the specified value, the resistance of the memristor is almost constant. The black colored end specifies the polarity of the memristor. If the current flowing through the memristor is injected into it via this end, the resistance of the device is decreased and if the polarity of the current is reversed, the resistance would be increased [26]. Many different models have been presented for memristors [27-34]. In this work, we have utilized the HP model. Today, memristors have had several applications in programmable analog circuits [35], [36], digital circuits [37], [38], signal processing [39], [40], image processing [41], neural networks [42-44], and etc.

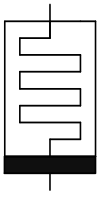

Fig. 4. Circuit Schematic of a memristor.

In this paper, the HP model is used for memristor. The Verilog-a code for memristor is shown in Table (1) [45].

Table 1. Verilog-a code for memristor.

```
`include "constants.vams"
`include "disciplines.vams"
module memristor_model2 (p, n) ;
inout p, n;
electrical p, n;
parameter real uv = 10f;
parameter real d = 10n;
parameter real ron = 1K;
parameter real roff = 1000k;
parameter real rinit = 5k;
real k, r0, r1, r2, R ,assert ,i;
analog begin
if(i == 0)
r0 = rinit;
k = 2 * uv * ron * (roff - ron) / pow(d,2);
if((R==ron)&&(V(p,n)>=0)||((R==roff)&&(V(p,n)<0))
)
begin
r0 = R;
assert = 1;
end
else if(assert!=0) begin
r0 = R;
assert = 0;
end
r1 = pow(r0,2) - k * idt( V(p,n), 0 ,assert);
r2 = min( pow(roff,2) , max(r1,pow(ron,2) ) );
R = sqrt(r2);
V(p,n) <+ R * I(p,n) ;
i = 1;
end
endmodule
```

2.3. Direct Form SOS IIR Filter

As mentioned in section 2.1, Fig. 1 and 2 consist of feed-forward and feed-back parts. The feed-forward coefficients are very small and close to each other, and cannot be implemented with the resolutions defined for memristor in practice (explained in section 3). To solve this problem, we can use the SOS structure of IIR filter [37], [38] shown in Fig. 6.

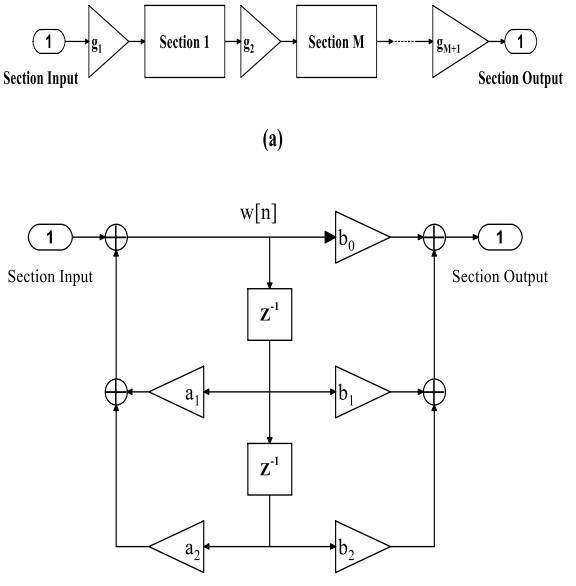

Fig. 6. (a) SOS structure of the IIR filter. (b) The block diagram of each section.

In Fig. 6, each section consists of three feed-forward coefficients and two feed-back coefficients. In this structure, one section and one scale value are considered for each of 2-orders. By converting the filter into parts with 2-orders, it is sufficient to enlarge the coefficients. As a result, the coefficients can be realized using memristor.

3. THE PROPOSED MEMRISTOR-BASED SOS STRUCTURE OF IIR FILTER

The proposed memristor-based SOS circuit of IIR filter is shown in Figure 7 for m-order.

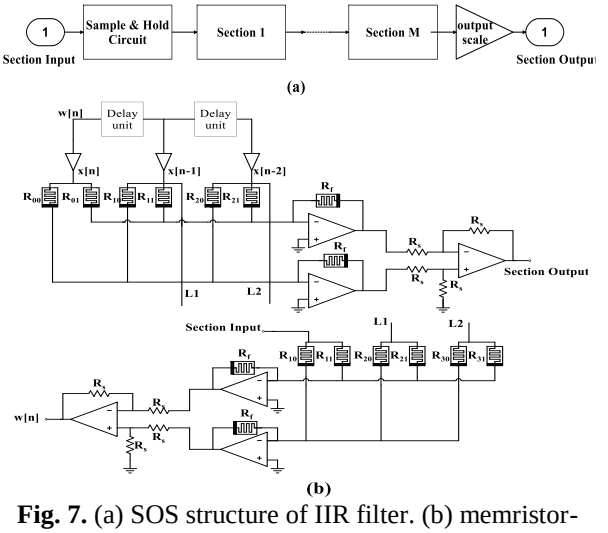

Fig. 7. (a) SOS structure of IIR filter. (b) memristor-based circuit for each section along with scale value.

In this structure, using a sampling circuit with a sampling period of T seconds, the input is sampled periodically. Using cascaded samplers, required delays are implemented. The required addition and multiplication operations could be implemented by memristor and operational amplifiers (Op-Amps). It should be noted that in figure (7-a), scale value blocks are considered inside each section.

3.1. The proposed circuit for generating negative coefficients

The proposed memristor-based FIR filter in [18] is only suitable for implementing specific filters which their coefficients are positive; because memristors could not get negative values. In IIR filter, there is necessarily a negative number. Therefore, in this article, this research proposed a new circuit topology (Fig. 8) for generating negative coefficients in IIR filters.

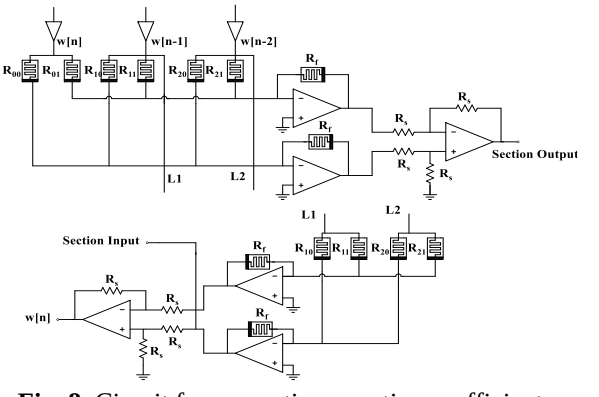

Fig. 8. Circuit for generating negative coefficients.

In Fig. 7, the output of each section is obtained using equation (4).

$$y[n] = x[n] + b_0 w[n] + (a_1 + b_1)w[n-1] + (a_2 + b_2)w[n-2] \quad (4)$$

Which a_j and b_i are the filter coefficients and describe from equations (5) and (6), respectively.

$$a_j = \frac{R_f}{R_{j+}} - \frac{R_f}{R_{j-}} \qquad j = 1,2 \quad (5)$$

$$b_i = \frac{R_f}{R_{i+}} - \frac{R_f}{R_{i-}} \qquad i = 0,1,2 \quad (6)$$

Where R_{i+} ،R_{i-} and R_{j+} ،R_{j-} are the memristance of memristors for generating positive and negative numbers, respectively. R_f can be used as a degree of freedom to assign the range of memristors.

For memristor, the maximum resolution is defined between 6 and 8 bits [19], [46], [47]. Thus, the range of memristors is divided into 64, 128, or 256 values.

Therefore, in setting filter coefficients, memristance values will be set to one of the acceptable values limited by resolution which this will lead to unavoidable errors in coefficients values. The memristor range is between 1KΩ - 1MΩ [48]. In this paper, a simple method (maximizing the memristance of memristor) and an advanced method (advanced search algorithm) are used to calculate the memristance of memristors, which is described below.

3.1.1. Maximizing the memristance of the memristor

In this method, according to equations (5) and (6), if the coefficient is negative, the resistance of memristor in positive part (R_{i+}) is set to the maximum value and vice versa. Also, R_f must be configured to ensure that all of the memristors are within the specified range. For example, for a negative coefficient, the memristance of memristor for positive and negative values are obtained from equation (7).

$$R_{i+} = 1\ M\Omega \qquad R_{i-} = \frac{R_f}{\frac{R_f}{1\ M\Omega} - b_i} \tag{7}$$

After calculating the memristor values, the coefficients are calculated again and the error is determined relative to their actual values. In this way, we can calculate the accuracy of the coefficients.

3.1.2. Advanced search algorithm

In this method, the various memristance (R_{i+} ،R_{i-} and R_f) is defined from equation (8).

$$R = R_{min} + \left(\frac{R_{max} - R_{min}}{2^n}\right) i\ , \quad i = 0,1, \dots, 2^n - 1 \tag{8}$$

In equation (8), R_{min} and R_{max}, the minimum and maximum values defined for memristor memristance and n are the memristor resolution in bits.

Using an advanced search algorithm, the value of R_{i+} ،R_{i-} and R_f are determined so that the error generated by the creation of coefficients is less than the real coefficients (equation 9).

$$E = \sum_{i=o}^{m} (b_i - \bar{b}_i)^2 \tag{9}$$

In equation (9), b_i are real coefficients and $\bar{b}_i$ are the calculated coefficients (the coefficients obtained from memristor by the advanced search algorithm).

It is worth to mention that the op-amp on the final stage is responsible for subtracting the results of the first stage op-amps; the value of all the final stage resistors have been selected to be equal deliberately here to give the unit gain to its inputs. Changing these resistors (R_s) could give more degrees of freedom in selecting memristors value, and also amplification or attenuation of the filtered signal.

3.2. Sampling circuit

Fig. 9a shows the sampling circuit used in [18].

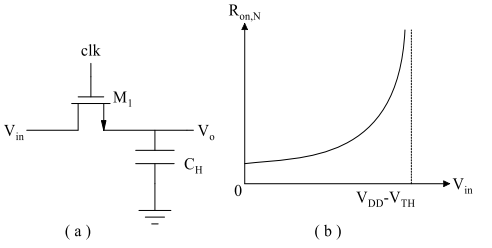

Fig. 9. (a) NMOS switch. (b) $R_{DS}(on)$ versus V_{in}.

The on-resistance increases with a large slope when the input is approaching $V_{DD} - V_{TH}$, That is, switch resistance is different for different input values. This has caused severe restrictions on the input voltage range, especially in recent technologies.

Using a PMOS instead of NMOS switch in the sampling circuit is another approach. This is illustrated in Fig. 10(a) it is on-resistance versus input voltage characteristic which is shown in Fig. 10(b).

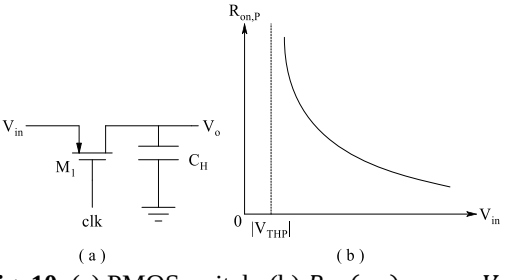

Fig. 10. (a) PMOS switch. (b) $R_{DS}(on)$ versus V_{in}.

According to the deficiency of NMOS and PMOS switches, the use of complementary switches is recommended for high input voltage range. This structure is shown in Fig. 11(a), called CMOS switch, and its characteristic is shown in Fig. 11(b) [49]. Obviously, for a significantly wide range of input voltage, the complementary switch shows a smooth and approximately linear on-resistance versus input voltage.

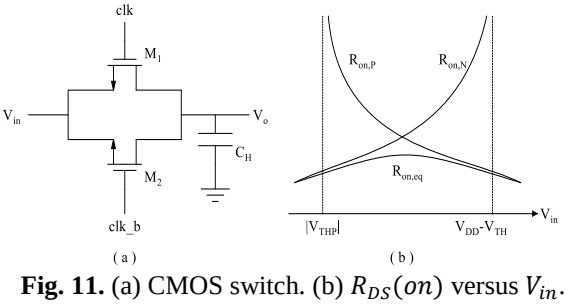

Fig. 11. (a) CMOS switch. (b) $R_{DS}(on)$ versus V_{in}.

In the sampler circuits, clock feed-through, noise KT/C and channel charge injection cause an error which occurs when switching is off. Dummy switches are used to eliminate charge injection and clock feed-through effects. The width in dummy switches is considered roughly double the width in CMOS switches. Also, the effect of clock feed-through is eliminated by gate-source and gate-drain capacitors of the M3 and M4 transistors [49]. The overall sampling circuit is presented in Fig. 12.

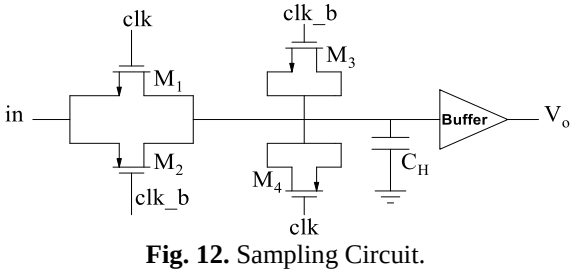

Fig. 12. Sampling Circuit.

3.3. Delay Circuit

The delay circuit is proposed in Fig. 13. By choosing this method, the applied voltage to the transistor gate will be a typical step voltage that can be easily produced. In addition, IIR filter with any order can be implemented by using of circuit Fig. 13. In this circuit, the first stage is called the Master, and similarly the second stage is called the Slave. According to Fig. 13, When clk = 0, the output of the sampler circuit is transmitted to V_x through Master. When $clk = V_{DD}$, the slave activated and V_x is transmitted to V_o. While V_x is mapped to the output, the master should be inactivated.

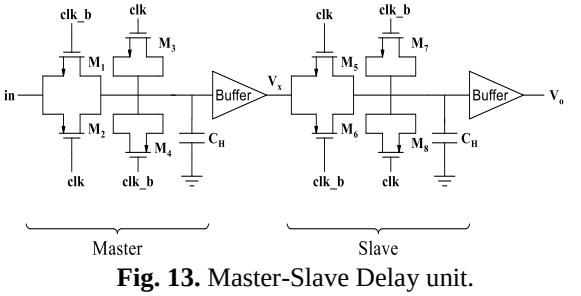

Fig. 13. Master-Slave Delay unit.

4. SIMULATION RESULTS

In this section, the simulation results of the IIR filter based on the memristor are presented. All simulations were performed by Cadence software in TSMC 0.18µm technology and power supply 1.8v. First, the accuracy of the sampling and delay circuits are shown. Then, low pass and high pass filters are tested and their results have been illustrated.

As mentioned, one of applications for IIR filter is audio signal processing. Hence, a component of the audio signal along with noise as input is applied and its output is investigated. The sampling frequency (f_s) is considered 400kHz which meets the Nyquist's sampling criterion [50].

The direct form SOS IIR filter type I and II are simulated for a resolution of 6, 7 and 8 bits. For simplicity, the results of direct form SOS type II for resolution of 7 bits are presented. As mentioned in Section 3.1, the HP model is used for memristor [25] and the memristor range is considered between 1KΩ-1MΩ [48]. For memristor, no library defined as a component in Cadence software. Thus, the memristor is added as a component in the Cadence software using the Verilog-a code given in Table (1).

4.1. Simulation sampler and delay

The sinusoidal signal defined by equation (10) is applied to the circuit as shown in Fig. 14.

$$x(t) = 0.2\sin(2\pi * 2 * 10^3\, t) \qquad (10)$$

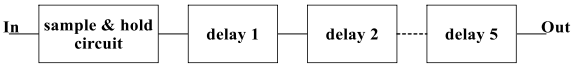

Fig. 14. The testing circuit for sampler and delay (for 5-order).

In Fig. 15, the output waveform of sampler and delay circuits (for 5 delay blocks) is shown for $C_H = 5\,pf$ and $f_s = 400\,KHz$. For each 20 µs, which is equivalent to the sampling frequency, from the input (blue) is sampled. Fig. 15 demonstrates the proper performance of sampling and delay circuits, clearly.

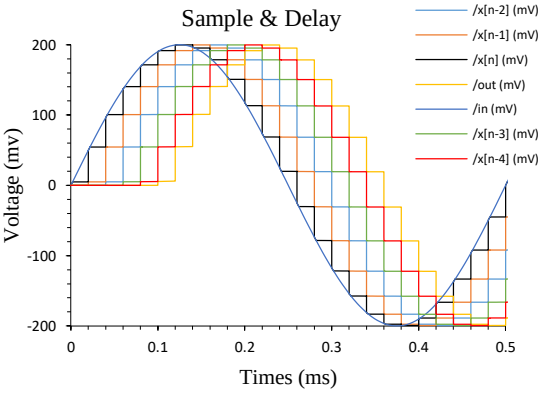

Fig. 15. The delayed versions of the input for five orders.

4.2. Implementation of a Low-Pass Filter

The characteristics of the lowpass IIR filter are shown in Table (2). In first, the filter was designed in FDATool environment of MATLAB and coefficients were extracted. As mentioned in Section 3.1, R_{i+}, $R_{i-}$and R_f should be selected so that the least error occurs in creating filter coefficients.

Table 2. Specifications of the low-pass filter.

Filter Type	f_s	f_c	Order
Low-pass	400 KHz	$20KHz$	8

According to Table 2, the sampling frequency (f_s) is 400KHz which meets the Nyquist's sampling criterion [50]. The cut-frequency (f_c) is set to 20KHz. For the test of the designed circuit, according to equation (11), an input is applied to the circuit shown in Fig. 7, which has two components: one voice signal with 2KHz and noise signal with 50KHz. As expected, the output is included only in the component with a low frequency and filters the high frequency component.

$$x(t) = 0.1\sin(2\pi * 2 * 10^3\, t) + 0.1\sin(2\pi * 50 * 10^3\, t) \quad (11)$$

For the desired filter, for 6, 7, and 8 bits of resolution, the coefficients have been calculated. To avoid mass of data, actual and calculated values for memristors and coefficients have been presented in Table 3 for only 7-bit resolution.

As shown in equation (12) the percentage error between actual and calculated coefficients.

$$Error = ABS(\frac{bi_{real} - bi_{calculated}}{bi_{real}}) \times 100 \quad (12)$$

For both methods, the percentage error of actual coefficients compared to calculated coefficients is shown in Figure 16 for section 1.

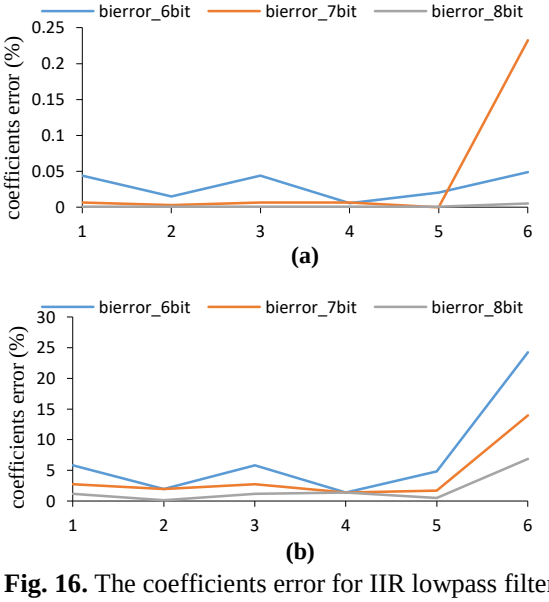

Fig. 16. The coefficients error for IIR lowpass filter in 6, 7 and 8 bits of resolution. a) Advanced search algorithm. b) Maximizing the memristance of memristor.

As shown in Fig. 16a, the error is very low in Advanced Search method (less than 0.25%), while in maximizing the memristance of memristor, the error is about 25%. Fig. 16, shows the superiority of advanced search method. Thus, we can implement a highly accurate filter using the advanced search method for any order.

In Fig. 17, the output of the lowpass filter is shown for the advanced search algorithm.

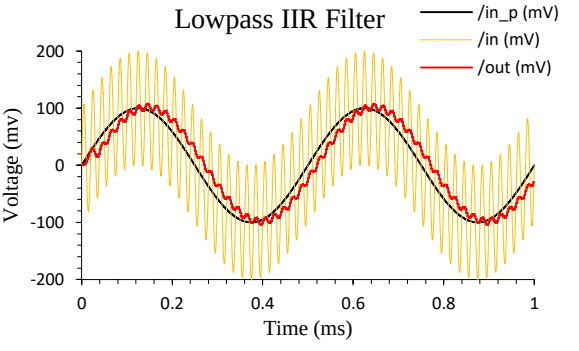

Fig. 17. Input /Output of lowpass Filter for 7 bit of resolution.

As shown in Fig. 17, the output is approximately equal to part of the input signal that should appear on the output. Therefore, it can be concluded that the circuit has a reasonable response for low-order filters.

Table 3. Values of the target and calculated coefficients and memristors of the low-pass filter for 7-bit resolution.

Coefficient Section1	real b_i	maximizing the memristor (R_f= 1000 KΩ)			advanced search (R_f= 984 KΩ)		
		Calculated b_i	R_{i+}(KΩ)	R_{i-}(KΩ)	Calculated b_i	R_{i+}(KΩ)	R_{i-}(KΩ)
b0	1	1.028397566	493	1000	0.999935384	438	789
b1	2	2.039513678	329	1000	1.999937272	227	422
b2	1	1.028397566	493	1000	0.999935384	438	789
a1	-1.63702328	-1.65957447	1000	376	-1.6369145	274	188
a2	0.837274182	0.851851852	540	1000	0.837274575	391	586
gain	0.050062725	0.058201058	945	1000	0.04994667	938	984
Coefficient Section2	real a_i	(R_f= 1000 KΩ)			(R_f= 602 KΩ)		
		Calculated b_i	R_{i+}(KΩ)	R_{i-}(KΩ)	Calculated b_i	R_{i+}(KΩ)	R_{i-}(KΩ)
b0	1	1.028397566	493	1000	0.999935904	344	805
b1	2	2.039513678	329	1000	2.000073013	196	563
b2	1	1.028397566	493	1000	0.999935904	344	805
a1	-1.42307894	-1.45700246	1000	407	-1.42306461	571	243
a2	0.597158836	0.6	625	1000	0.597024402	305	438
gain	0.043519973	0.049317943	953	1000	0.043527533	899	961
Coefficient Section3	real a_i	(R_f= 1000 KΩ)			(R_f= 579 KΩ)		
		Calculated b_i	R_{i+}(KΩ)	R_{i-}(KΩ)	Calculated b_i	R_{i+}(KΩ)	R_{i-}(KΩ)
b0	1	1.028397566	493	1000	0.999992033	337	805
b1	2	2.039513678	329	1000	1.999940705	188	540
b2	1	1.028397566	493	1000	0.999992033	337	805
a1	-1.29367682	-1.3241879	1000	430	-1.29359107	571	251
a2	0.45192744	0.453488372	688	1000	0.451929004	368	516
gain	0.039562655	0.040582726	961	1000	0.039527901	867	922
Coefficient Section4	real a_i	(R_f= 1000 KΩ)			(R_f= 508 KΩ)		
		Calculated b_i	R_{i+}(KΩ)	R_{i-}(KΩ)	Calculated b_i	R_{i+}(KΩ)	R_{i-}(KΩ)
b0	1	1.028397566	493	1000	0.999917557	282	633
b1	2	2.039513678	329	1000	1.999922575	188	727
b2	1	1.028397566	493	1000	0.999917557	282	633
a1	-1.23299903	-1.24215247	1000	446	-1.23300316	547	235
a2	0.383827157	0.390820584	719	1000	0.383816798	469	727
gain	0.037707033	0.040582726	961	1000	0.037721074	633	664

4.3. Implementation of a High-pass Filter

A high-pass filter was also designed with the specifications given in Table (4).

Table 4. Specifications of the high-pass filter.

Filter Type	f_s	f_c	Order
High-pass	$400\ KHz$	$20KHz$	8

To check the filter performance, according to equation (13), a sinusoidal input has been applied to Fig. 7.

$$x(t) = 0.1\sin(2\pi * 2 * 10^3\ t) + 0.02\sin(2\pi * 40 * 10^3\ t) \quad (13)$$

To better display the output, the amplitude of input components is selected differently. Is expected, the high frequency signal (40 KHz) appear at output and the low frequency signal (2 KHz) filtered. The memristance of memristors and the calculated coefficients are given in Table (5).

The output of the high-pass filter for advanced search algorithm is shown in Fig. 18.

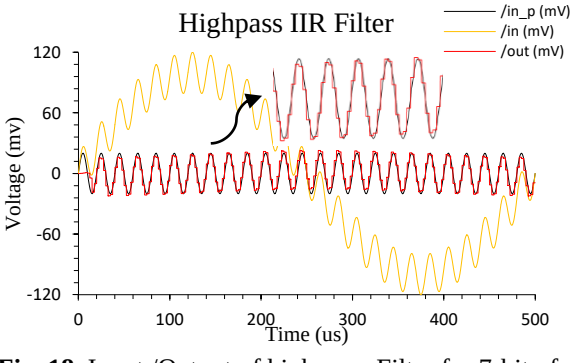

Fig. 18. Input /Output of high-pass Filter for 7-bit of resolution.

In Figure 18, the red trace is the filter output and the black trace represents the part of input signal that should appear on the output. According to the part which has been zoomed-in in Figure 18, it is visible that these two signals are almost equal.

Also, the percentage of error of the actual coefficients relative to the calculated coefficients is shown in Fig. 19.

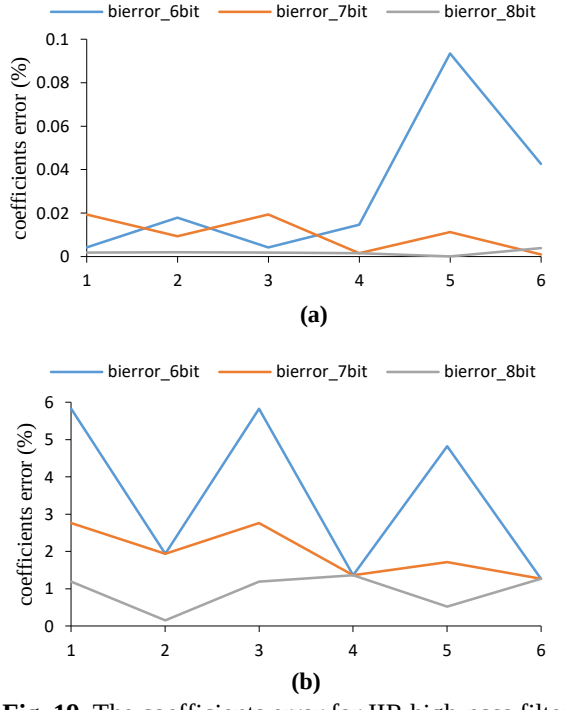

Fig. 19. The coefficients error for IIR high-pass filter in 6, 7 and 8 bits of resolution. a) Advanced search algorithm. b) Maximize memristance of memristor.

It can be observed that the maximum error for the maximizing the memristance of memristor method, like the low-pass filter, is extremely high (about 6%). In the worst case, the error of the calculated coefficients for advanced search method is less than 0.1%.

Table 5. Values of target and calculated coefficients and memristors of the high-pass filter for 7-bit resolution.

Coefficient Section1	real b_i	maximizing the memristor (R_f = 1000 KΩ)			advanced search (R_f = 914 KΩ)		
		Calculated b_i	R_{i+}(KΩ)	R_{i-}(KΩ)	Calculated b_i	R_{i+}(KΩ)	R_{i-}(KΩ)
b0	1	1.028397566	493	1000	0.999806544	430	813
b1	-2	-2.03951368	1000	329	-2.00018751	422	220
b2	1	1.028397566	493	1000	0.999806544	430	813
a1	-1.637023283	-1.65957447	1000	376	-1.63699897	383	227
a2	0.837274182	0.851851852	540	1000	0.837179785	204	251
gain	0.868574366	0.879699248	532	1000	0.868582807	313	446
Coefficient Section2	real a_i	(R_f = 1000 KΩ)			(R_f = 508 KΩ)		
		Calculated b_i	R_{i+}(KΩ)	R_{i-}(KΩ)	Calculated b_i	R_{i+}(KΩ)	R_{i-}(KΩ)
b0	1	1.028397566	493	1000	0.999917557	282	633
b1	-2	-2.03951368	1000	329	-1.99992257	727	188
b2	1	1.028397566	493	1000	0.999917557	282	633
a1	-1.423078943	-1.45700246	1000	407	-1.42311894	625	227
a2	0.597158836	0.6	625	1000	0.597046578	376	672
gain	0.755059445	0.776198934	563	1000	0.755021018	383	891
Coefficient Section2	real a_i	(R_f = 1000 KΩ)			(R_f = 579 KΩ)		
		Calculated b_i	R_{i+}(KΩ)	R_{i-}(KΩ)	Calculated b_i	R_{i+}(KΩ)	R_{i-}(KΩ)
b0	1	1.028397566	493	1000	0.999992033	337	805
b1	-2	-2.03951368	1000	329	-1.99994071	540	188
b2	1	1.028397566	493	1000	0.999992033	337	805
a1	-1.293676821	-1.3241879	1000	430.3	-1.29359107	571	251
a2	0.45192744	0.453488372	688	1000	0.451929004	368	516
gain	0.686401065	0.706484642	586	1000	0.686398494	383	703
Coefficient Section2	real a_i	(R_f = 1000 KΩ)			(R_f = 508 KΩ)		
		Calculated b_i	R_{i+}(KΩ)	R_{i-}(KΩ)	Calculated b_i	R_{i+}(KΩ)	R_{i-}(KΩ)
b0	1	1.028397566	493	1000	0.999917557	282	633
b1	-2	-2.03951368	1000	329	-1.99992257	727	188
b2	1	1.028397566	493	1000	0.999917557	282	633
a1	-1.232999025	-1.24282035	1000	445.9	-1.23300316	547	235
a2	0.011793858	0.390820584	719	1000	0.383816798	469	727
gain	0.007076315	0.661129568	602	1000	0.654265472	181	235

5. CONCLUSION

In this paper, a new approach for implementing the IIR filter using memristor is proposed. It was observed that the direct form SOS of IIR filter enables the implementation of the filter with any level. Also, with the help of the Advanced Search algorithm, it was shown that the error of actual coefficients is small compared to the calculated coefficients. At order 8, in the worst case, the lowpass filter error rate is less than 0.25% and the calculated error for the high pass filter is about 0.1%. In addition, the output waveform in IIR filters represents enough precision, when the filters are implemented with the memristor even at low-orders.

As a future scope of this work, the researchers can work on audio signals processing, audio data compression, speech recognition, image processing, and etc. using proposed IIR filters.

REFERENCES

[1] M. Rubinoff, **"Analogue vs. Digital Computers-A Comparison,"** *Proceedings of the IRE*, Vol. 41, No. 10, pp. 1254-1262, 1953.

[2] M. D. Hahm, E. G. Friedman, and E. L. Titlebaum, **"Analog vs. Digital: A Comparison of Circuit Implementations for Low-Power Matched Filters,"** *in Circuits and Systems*, 1996. ISCAS'96., Connecting the World., 1996 IEEE International Symposium on, 1996, Vol. 4, pp. 280-283: IEEE, 1996.

[3] C. R. Schlottmann and J. Hasler, **"High-Level Modeling of Analog Computational Elements For Signal Processing Applications,"** *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Vol. 22, No. 9, pp. 1945-1953, 2014.

[4] S. W. Smith, **"The Scientist and Engineer's Guide to Digital Signal Processing,"** 1997.

[5] S. M. Kay, **"Fundamentals of Statistical Signal Processing,"** *Prentice Hall PTR*, 1993.

[6] I. Boucherit and M. Guerti, **"Speech Analysis In Cochlear Implant Using Auditory Filter Bank Model,"** *in Modelling, Identification and Control (ICMIC), 2016 8th International Conference on, 2016*, pp. 274-278: IEEE.

[7] T. Qu, Q. Huang, Y. Huang, L. Li, and X. Wu, **"An Accurate Decorrelation Method For Parametric Stereo Coding,"** *in Audio, Language and Image Processing (ICALIP), 2016 International Conference on, 2016*, pp. 72-76: IEEE.

[8] R. Rehr and T. Gerkmann, **"An Analysis of Adaptive Recursive Smoothing with Applications to Noise PSD Estimation,"** *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, Vol. 25, No. 2, pp. 397-408, 2017.

[9] F. d. Lacerda, **"Conversor DSB-SSB a capacitores chaveados por Transformador de Hilbert em tecnologia CMOS de 180 nm,"** 2017.

[10] C.-C. Tseng and S.-L. Lee, **"Closed-Form Designs Of Digital Fractional Order Butterworth Filters Using Discrete Transforms,"** *Signal Processing*, Vol. 137, pp. 80-97, 2017.

[11] A. J. Lipton, H. Fujiyoshi, and R. S. Patil, **"Moving Target Classification and Tracking From Real-Time Video,"** *in Applications of Computer Vision*, 1998. WACV'98. Proceedings., Fourth IEEE Workshop on, pp. 8-14: IEEE, 1998.

[12] H.-Y. Wu, M. Rubinstein, E. Shih, J. Guttag, F. Durand, and W. Freeman, **"Eulerian Video Magnification for Revealing Subtle Changes In The World,"** 2012.

[13] S.-N. Mirebrahimi and F. Merrikh-Bayat, **"Programmable Discrete-Time Type I and Type II FIR Filter Design on The Memristor Crossbar Structure,"** *Analog Integrated Circuits and Signal Processing*, Vol. 79, No. 3, pp. 529-541, 2014.

[14] F. M. Bayat, F. Alibart, L. Gao, and D. B. Strukov, **"A Reconfigurable Fir Filter with Memristor-Based Weights,"** *arXiv preprint arXiv*:1608.05445, 2016.

[15] D. A. Johns and D. M. Lewis, **"Design And Analysis Of Delta-Sigma Based Iir Filters,"** *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, Vol. 40, No. 4, pp. 233-240, 1993.

[16] D. Johns and D. Lewis, **"IIR Filtering On Sigma-Delta Modulated Signals,"** *Electronics letters*, Vol. 27, No. 4, pp. 307-308, 1991.

[17] A. Madanayake, L. Bruton, and C. Comis, **"FPGA Architectures for Real-time 2D/3D FIR/IIR Plane Wave Filters,"** in *Circuits and Systems, 2004. ISCAS'04. Proceedings of the 2004 International Symposium on*, Vol. 3, pp. III-613: IEE,E, 2004.

[18] S. M. R. Islam, R. Sarker, S. Saha, and A. N. Uddin, **"Design of a Programmable Digital IIR Filter based on FPGA,"** in *Informatics, Electronics & Vision (ICIEV), 2012 International Conference on*, pp. 716-721: IEEE, 2012.

[19] Z. Gao, X. Zeng, J. Wang, and J. Liu, **"FPGA Implementation of Adaptive IIR Filters with Particle Swarm Optimization Algorithm,"** in *Communication Systems, 2008. ICCS 2008. 11th IEEE Singapore International Conference on*, pp. 1364-1367: IEEE, 2008.

[20] B. Gold, T. G. Stockham, A. V. Oppenheim, and C. M. Rader, **"Digital processing of signals,"** 1969.

[21] B. Widrow, **"Adaptive adaline Neuron Using Chemical memistors.".** 1960.

[22] H. Kim and S. P. Adhikari, **"Memistor is not memristor [express letters],"** *IEEE Circuits and Systems Magazine*, Vol. 12, No. 1, pp. 75-78, 2012.

[23] R. M. Fano, L. J. Chu, and R. B. Adler, **"Electromagnetic Fields, Energy, and Forces,"** 1968.

[24] L. Chua, **"Memristor-the Missing Circuit Element,"** *IEEE Transactions on circuit theory*, Vol. 18, No. 5, pp. 507-519, 1971.

[25] D. B. Strukov, G. S. Snider, D. R. Stewart, and R. S. Williams, **"The Missing Memristor Found,"** *nature*, Vol. 453, No. 7191, pp. 80-83, 2008.

[26] S. Kvatinsky, G. Satat, N. Wald, E. G. Friedman, A. Kolodny, and U. C. Weiser, **"Memristor-Based Material Implication (IMPLY) Logic: Design Principles and Methodologies,"** *IEEE Transactions*

on Very Large Scale Integration (VLSI) Systems, Vol. 22, No. 10, pp. 2054-2066, 2014.

[27] Z. Biolek, D. Biolek, and V. Biolkova, **"SPICE Model of Memristor with Nonlinear Dopant Drift,"** *Radioengineering*, Vol. 18, No. 2, 2009.

[28] T. Prodromakis, B. P. Peh, C. Papavassiliou, and C. Toumazou, **"A Versatile Memristor Model with Nonlinear Dopant Kinetics,"** *IEEE transactions on electron devices*, Vol. 58, No. 9, pp. 3099-3105, 2011.

[29] J. J. Yang, M. D. Pickett, X. Li, D. A. Ohlberg, D. R. Stewart, and R. S. Williams, **"Memristive Switching Mechanism for Metal/Oxide/Metal Nanodevices,"** *Nature nanotechnology*, Vol. 3, No. 7, pp. 429-433, 2008.

[30] E. Lehtonen and M. Laiho, **"CNN Using Memristors for Neighborhood Connections,"** *in Cellular Nanoscale Networks and Their Applications (CNNA)*, 2010 12th International Workshop on, 2010, pp. 1-4: IEEE.

[31] M. D. Pickett et al., **"Switching Dynamics In Titanium Dioxide Memristive Devices,"** *Journal of Applied Physics*, Vol. 106, No. 7, p. 074508, 2009.

[32] H. Abdalla and M. D. Pickett, **"SPICE modeling of memristors,"** *in Circuits and Systems (ISCAS)*, 2011 IEEE International Symposium on, 2011, pp. 1832-1835: IEEE.

[33] C. Yakopcic, T. M. Taha, G. Subramanyam, R. E. Pino, and S. Rogers, **"A Memristor Device Model,"** *IEEE electron device letters*, Vol. 32, No. 10, pp. 1436-1438, 2011.

[34] S. Kvatinsky, E. G. Friedman, A. Kolodny, and U. C. Weiser, **"TEAM: Threshold Adaptive Memristor Model,"** *IEEE Transactions on Circuits and Systems* I: Regular Papers, Vol. 60, No. 1, pp. 211-221, 2013.

[35] S. Shin, K. Kim, and S. M. Kang, **"Memristor Applications for Programmable Analog ICs,"** *IEEE Transactions on Nanotechnology*, Vol. 10, No. 2, pp. 266-274, 2011.

[36] Y. V. Pershin and M. D. Ventra, **"Practical Approach to Programmable Analog Circuits with Memristors,"** *IEEE Transactions on Circuits and Systems* I: Regular Papers, Vol. 57, No. 8, pp. 1857-1864, 2010.

[37] T. Raja and S. Mourad, **"Digital Logic Implementation in Memristor-Based Crossbars,"** *in 2009 International Conference on Communications, Circuits and Systems*, 2009, pp. 939-943.

[38] J. Cong and X. Bingjun, **"mrFPGA: A novel FPGA Architecture with Memristor-Based Reconfiguration,"** *in 2011 IEEE/ACM International Symposium on Nanoscale Architectures*, 2011, pp. 1-8.

[39] D. Biolek, V. Biolkova, and Z. Kolka, **"Memristive Systems for Analog Signal Processing,"** *in 2014 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2014, pp. 2588-2591.

[40] B. Mouttet, **"Proposal for Memristors in Signal Processing,"** *in Nano-Net: Third International ICST Conference*, NanoNet 2008, Boston, MA, USA, September 14-16, 2008, Revised Selected Papers, M. Cheng, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 11-13.

[41] X. Hu, S. Duan, L. Wang, and X. Liao, **"Memristive Crossbar Array with Applications In Image Processing,"** *Science China Information Sciences*, journal article, Vol. 55, No. 2, pp. 461-472, February 01 2012.

[42] D. Soudry, D. D. Castro, A. Gal, A. Kolodny, and S. Kvatinsky, **"Memristor-Based Multilayer Neural Networks with Online Gradient Descent Training,"** *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 26, No. 10, pp. 2408-2421, 2015.

[43] Y. V. Pershin and M. Di Ventra, **"Experimental Demonstration of Associative Memory With Memristive Neural Networks,"** *Neural Networks*, Vol. 23, No. 7, pp. 881-886, 2010/09/01/ 2010.

[44] X. Liu, Z. Zeng, and S. Wen, **"Implementation of Memristive Neural Network with Full-Function Pavlov Associative Memory,"** *IEEE Transactions on Circuits and Systems* I: Regular Papers, Vol. 63, No. 9, pp. 1454-1463, 2016.

[45] A. A. Emara, M. M. Aboudina, and H. A. Fahmy, **"Corrected and accurate Verilog-A for linear dopant drift model of memristors,"** in *Circuits and Systems (MWSCAS), 2014 IEEE 57th International Midwest Symposium on*, 2014, pp. 499-502: IEEE.

[46] C. E. Merkel, N. Nagpal, S. Mandalapu, and D. Kudithipudi, **"Reconfigurable N-level memristor memory design,"** in *Neural Networks (IJCNN), The 2011 International Joint Conference on*, 2011, pp. 3042-3048: IEEE.

[47] F. Alibart, E. Zamanidoost, and D. B. Strukov, **"Pattern Classification By Memristive Crossbar Circuits Using Ex Situ And In Situ Training,"** *Nature communications*, Vol. 4, p. 2 , 072, 2013.

[48] M. Prezioso, F. Merrikh-Bayat, B. Hoskins, G. Adam, K. K. Likharev, and D. B. Strukov, **"Training And Operation of An Integrated Neuromorphic Network Based on Metal-Oxide Memristors,"** *Nature*, Vol. 521, No. 7550, pp. 61-64, 2015.

[49] B. Razavi, *Design of Analog Cmos Integrated Circuits*, Second Edition ed. MA, Boston:McGraw-Hill, 2017.

[50] A. V. Oppenheim, R. W. Schafer, and J. R. Buck, *Discrete-time Signal Processing (2nd ed.)*. Prentice-Hall, Inc., 1999, p. 870.