A Hybrid Memristor-CMOS Multiplier Design Based on Memristive Universal Logic Gates

Mehri Teimoory, Amirali Amirsoleimani, Arash Ahmadi, Majid Ahmadi
Department of Electrical and Computer Engineering,
University of Windsor, Windsor, Canada
{amirsol, aahmadi, ahmadi}@uwindsor.ca

***Abstract*—Memristor is considered as one of the promising solutions to the fundamental limitations of the VLSI systems. Logic implementation with memristor device by considering its compatibility with CMOS fabric provides a new vision for digital logic circuits. This work presents a 2 by 2 multiplier cell design using a hybrid CMOS-memristor universal gate. The universal gate based implementation approach is the extension for memristor ratioed logic (MRL) with lower implementation cost. Simulation results confirm functionality of the proposed circuit. This circuit requires 16 memristors, 8 transistors and only one computational time step for multiplication. Compared with previous works, this approach presents considerably lower implementation cost.**

Keywords—Memristor, Logic Design, CMOS Hybrid, Universal Gate, Memristor Ratioed Logic.

I. INTRODUCTION

Due to the nano-scale integration, CMOS technology compatibility and non-volatility, memristor [1] is a key contender element for future computing systems [2]. Analog memories, neuromorphic and logic are some well-known applications of memristors. Logic design in combination with memory characteristics of the memristors creates a unique opportunity for the next generation computers in which memory and logic devices are blended in a fabric avoiding classic data band width limitation between processing units and the memory.

Among variety of memristor applications, memristor based logic design is a vital move. Several logic design methods using memristors have been presented [3-6]. Memristor Ratioed Logic (MRL) is a CMOS compatible logic [5]. Similar to CMOS logic, in this method logical states ('0' and '1') are defined by the voltage level of the output. Memristor Aided Logic (MAGIC) [3] and Material Implication Logic (IMPLY) [4] are two pure memristive logics proposed for crossbar structures. In these logic designs, unlike MRL logic, logic values are defined by memristance of the output memristor, which makes it challenging to combine these circuits with current VLSI technology in an integrated platform. Different memristor-based conventional arithmetic and computational building blocks have been presented previously, such as full adder [6][11], multiplier [7][15], ripple carry adder [8], counter [9], 4 by 2 compressor [10] and linear feedback shift register (LFSR) [12].

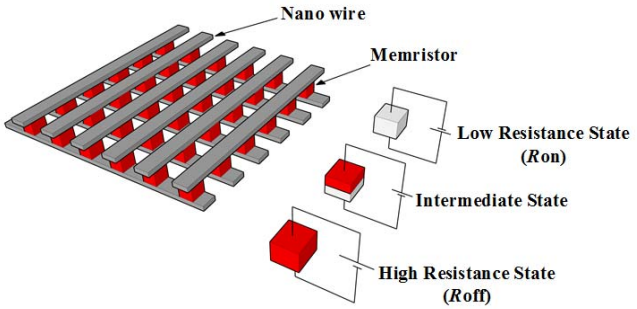


Fig. 1. Memristor crossbar structure and different memristor resistance state.

In this paper, based on hybrid CMOS-memristor universal gates, an efficient 2 by 2 multiplier cell design is proposed. The proposed implementation method is an extension for MRL logic. The utilized universal gate generates 3 different logics at the same time within one circuit by only 4 memristors and 2 CMOS. The designed array multiplier requires only one computational step for its operation by utilizing 16 memristors and 8 transistors. Compared with previous works, this approach presents considerably lower implementation cost.

The rest of paper is organized as follow: Section II provides an overview on the memristor and MRL logic circuit. Section III presents the proposed circuit and the simulation results, and paper is concluded in Section IV.

II. MEMRISTOR AND LOGIC DESIGN

A. Memristor

Mathematically, memristance is defined by a relationship between charge (q) and flux (φ) as

$$d\varphi = M \cdot dq \tag{1}$$

where M is the memristance of device. Based on this definition, memristor is a resistor with a variable resistance (memristance). Practically, memristor has two important characteristics. Firstly, its resistance changes based on the charge passing through the device or the applied voltage during the operation time. Secondly, it keeps the last resistance values, which make an ideal candidate for analog memories. The first reported successful fabrication [1, 2] was made of a titanium dioxide thin film with two platinum contacts. Memristor crossbar is widely known as a promising memristive structure and its high density makes it suitable platform for future

978-1-5090-6389-5/17/$31.00 ©2017 IEEE

memory and computing architecture. In this device, by applying a voltage bias, positively charged oxygen TiO_{2-x} vacancies drift towards one end of the device and it results decreasing the effective resistance of the memristor. The resistance of device remains unchanged when the voltage is removed. By applying a reverse bias voltage, vacancies can be pushed back to where they came from. Accordingly, memristance can be changed between a minimum value (R_{on}) and a maximum values (R_{off}). Interestingly, when the voltage is smaller than a threshold (V_{th}), the boundary between the doped and un-doped region remains stable which indicates the memory property of the memristor. The memristor crossbar structure and different memristor resistance states are displayed in Fig. 1.

B. Memristor Based Logic

Several logic design methods have been presented based on crossbar structure by which one can perform data processing without any need to transfer data from/to memory devices. Material implication or IMPLY logic [4] and MAGIC [3] are the most popular approaches for logic design inside memristor crossbars. The IMPLY logic is based on realization of logical operation of material implication using a sequence of in memory operations on memristors. The major drawback of this approach is its sequential nature which makes it very slow even for basic combinational logic circuits and requires a complex sequencer to produce right sequence of the applied voltages to the memristors. Compared with IMPLY, MAGIC has a simpler operation mechanism and does not need additional hardware to the crossbar. Further, MAGIC gate writes the output value to a separate memristor unlike IMPLY, which writes the output on one of the input memristors and erases that input's value and this destructive operation is not desired in some applications.

In general, it is challenging to integrate logic design methods, which are based on memristive crossbar fabrics, where logic values are represented by memristor state rather than voltage levels with current CMOS circuits. CMOS, on the other hand, is a successful and mature technology. Therefore, integrating memristors with it can create great opportunities and also a smooth transition from current technology to the next generation VLSI systems.

C. Memristor Ratioed Logic

MRL, was proposed by Eshraghian [13] and developed as logic in [5]. It is a CMOS compatible logic gate [3]. It has been provided by two memristors. Inputs and outputs are defined by level of voltage. The AND gate is shown in Fig. 2(a). Reversing polarity of the memristors gives the OR gate, see Fig. 2(b).

In the AND gate circuit, if $V_1 = 0$ and $V_2 = V_{high}$, and $R_{off} >> R_{on}$ then the output is calculated as:

$$V_{out,AND} = \frac{R_{off}}{R_{off} + R_{on}} V_{high} \cong V_{high} \quad (2)$$

which is logic '1' and when $V_1 = V_2 = 0$ the output is

$$V_{out,AND} = \frac{R_{on}}{R_{off} + R_{on}} V_{high} \cong 0 \quad (3)$$

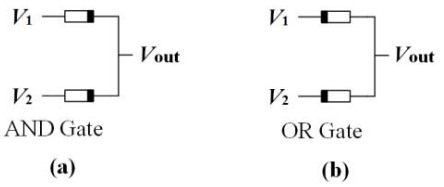


Fig. 2. The schematic of the MRL gates. (a) AND gate. (b) OR logic gate.

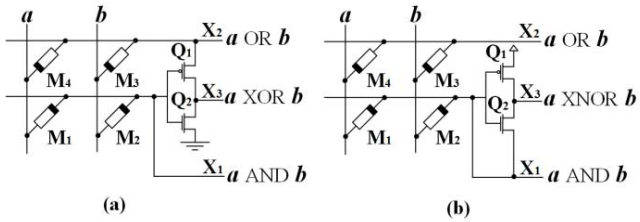


Fig. 3. Universal gates within a fabric (a) OR-XOR-AND gate (b) OR-XNOR-AND gate.

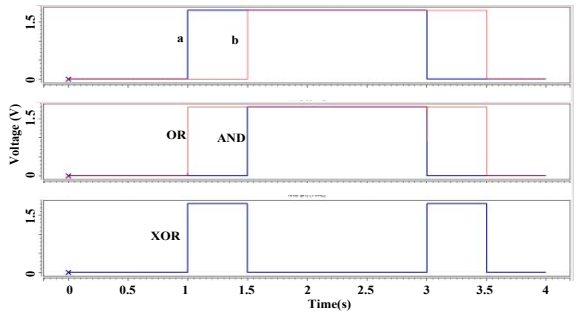
which is logic '0'. If the inputs have the same logic values, the output follows the inputs and has the same value too. Similar argument is applicable to the OR circuit structure.

III. Proposed Circuit

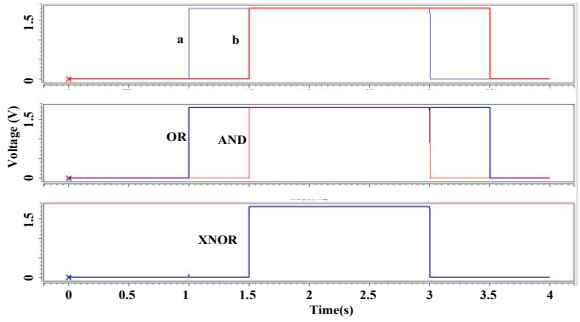
A. Memristor based Universal Gates

Utilizing the AND and OR gate structures in the MRL logic design of Fig. 2, an efficient memristor based universal gate can be made by combining two memristors in series and choosing appropriate polarities, as explained in [13]. The proposed universal gate is displayed in Fig. 3 (a). This gate has three outputs and provides AND, OR and XOR of the inputs at the same time. With this approach, the universal gate of OR-XOR-AND can be extended to the OR-XNOR-AND, as shown in Fig. 3 (b). Using these universal gates basic logic operations are available simultaneously depending on the selected output location in the circuit. These gates consist of four memristors and two MOSFET transistors. Here, MOSFETs work as a multiplexer where its inputs come from two memristors (M_1 and M_2), which create the AND gate structure, and two memristors (M_3 and M_4), which is the OR gate structure. With appropriate connection of the inputs we can obtain AND gate in node X_1, OR gate in node X_2, and XOR gate (as in Fig. 3 (a)) or XNOR gate (as in Fig. 3 (b)) in node X_3. Spice simulation results for these universal gates are presented in Fig. 4.

The proposed universal logic gates provide 3 different logic operations (OR-XOR-AND/OR-XNOR-AND), within one circuit with lower number of devices and less area in comparison with conventional CMOS logic. In Table 1, the number of devices, MOSFET and memristor, utilized in CMOS logic and proposed logic is presented for basic logic functions. In addition, the universal gate-based logic is more area efficient in comparison with other hybrid CMOS-memristor based logic designs [5][15]. In [15], the implementation of XOR and XNOR gates requires 4 CMOS



(a)



(b)

Fig. 4. Simulation result of universal gates (a) OR-XOR-AND gate (b) OR-XNOR-AND gate.

and 6 memristors while here in the proposed logic only 2 CMOS and 4 memristors have been utilized. The unique feature of universal gate-based logic is its capability to reproduce 3 different logic and the same time with few number of devices. In Table 2, the required number of elements to implement OR-XOR-AND and OR-XNOR-AND functions have been analysed for CMOS-based and different hybrid CMOS-memristor based logics. CMOS logic is compared with the proposed logic for implementation of OR-XOR-AND and OR-XNOR-AND in terms of utilized number of CMOS and memristors. The proposed logic is more area efficient in comparison with the state of art.

B. 2×2 Binary Multiplier

Multiplication is an essential arithmetic operation in digital and VLSI applications [14]. In DSP systems, multiplayers are used widely. In some DSP algorithms hardware implementation cost, power consumption and the execution time of the whole system are greatly dependent to the size, power and speed of the multipliers [14]. Therefore, high speed and low power multiplayers are essential circuits in modern digital processing systems.

These requirements need to be achieved both in algorithm level and circuit level design of the multipliers. The proposed hybrid CMOS memristor multiplayer circuit in this paper creates a new opportunity to achieve these essential requirements by utilizing unique characteristics of the memristor.

TABLE I. NUMBER OF DEVICES UTILIZED IN CMOS LOGIC AND PROPOSED.

Logic	Device	AND	NAND	OR	NOR	XOR	XNOR
CMOS Logic	Memristor	-	-	-	-	-	-
	MOSFET	6	4	6	4	12	12
Proposed	Memristor	2	2	2	2	4	4
	MOSFET	-	2	-	2	2	2

TABLE II. NUMBER OF DEVICES UTILIZED FOR HAVING AND-XOR-OR AND AND-XNOR-NOR IN DIFFERENT LOGIC METHODS .

Logic	Device	AND-XOR-OR	AND-XNOR-OR
CMOS Logic	Memristor	-	-
	MOSFET	24	24
MRL Logic [6]	Memristor	6	6
	MOSFET	2	4
MeMOS Logic [15]	Memristor	6	6
	MOSFET	4	4
Proposed	Memristor	4	4
	MOSFET	2	2

Theoretically, a multiplayer cell (a 2×2 binary multiplayer, as shown in Fig. 5(a)) can be made by two universal gates OR-XOR-AND (as maintained in previous section), however, there are considerations in the circuit signaling characteristics, which must be taken in to account.

XOR output of the circuit presented in Fig. 3 (a), might has a problem in pulling up the voltage when output is logic '1'. This is due to the fact that unlike a normal CMOS NOT gate, in the proposed gate source of the PMOS is not connected to the V_{DD} Directly. Instead, in this circuit it works like a multiplexer and forwards logic value of the M_3 and M_4 memristors to the output. These issue becomes even more serious when we need to connect several gates in series one after each other.

To avoid this problem, here we utilized AND-XNOR-OR universal gate with an NOT gate buffer in its output to correct the voltage level and compensate any current or voltage drop propagation in the circuit. This costs two more transistors in our universal gate but increases circuit reliability and noise margin. Proposed 2 by 2 multiplier cell is shown in Fig. 5. In this design two OR-XNOR-AND gates with NOT buffer in their outputs are utilized along with four MRL AND gates connected to the inputs. Circuit consists of 16 memristors and 8 transistors.

C. Simulation Setup

The proposed multiplier cell is evaluated using circuit level HSPICE simulations. In this evaluation HP memristor model is adopted with R_{on} = 10 KΩ, R_{off} = 1 MΩ, p = 1, L = 10 nm, W_{min} = 0.05, W_{max} = 0.95, p = 1, f_{on} = 40e-3, f_{off} = 40e-3. For MOS transistors BSIM3 (V3.1) is used with TSMC 0.18 um technology file. Simulation results are presented in Fig. 5 (c). As it can be seen, all voltages are in correct level and circuit logic operation is performed correctly. Logic values of the output for different inputs are in consistence with the original specifications for 2 by 2 multiplier. The proposed multiplier is also more efficient in terms of number of utilized devices in

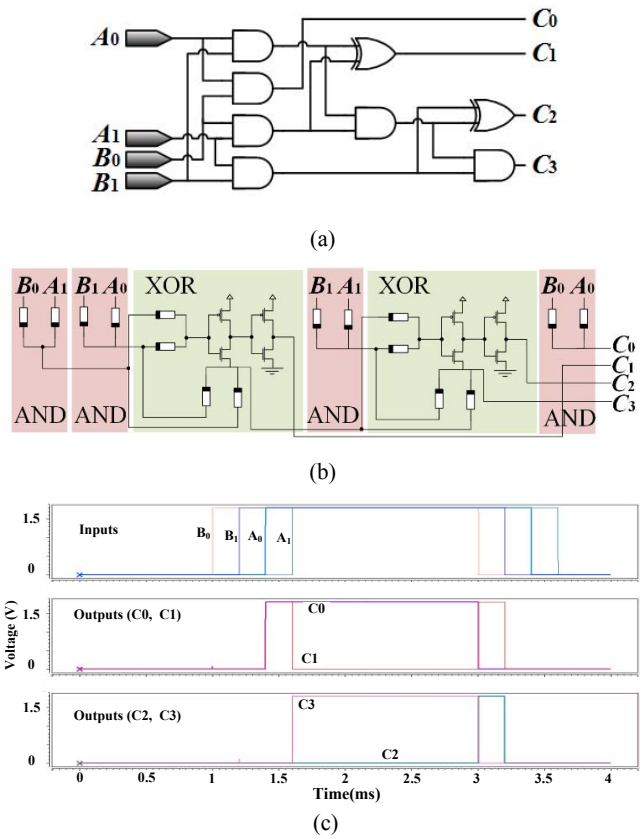


Fig. 5. Logic design of the 2 by2 multiplier. (a) Logic schematic (b) Circuit schematic (c) Simulation results.

TABLE III. NUMBER OF DEVICES UTILIZED FOR 2X2 MULTIPLIER.

Logic	Device	CMOS Logic	MeMOS Logic [15]	MAD Logic [7]	Proposed
2x2 Multiplier	Memristor	-	34	10	16
	MOSFET	62	32	28	8

comparison with different multipliers in state of art. In Table 3, the proposed multiplier design is compared with these designs.

IV. CONCLUSION

A 2 by 2 multiplier cell based on a universal CMOS memristor gate is presented. Presented design is implemented using only 16 memristors and 8 transistors, which is considerably low cost compared with conventional CMOS design. Also, it is more area efficient in comparison with hybrid CMOS-memristor based multiplier design in state of art. The functionality of the proposed multiplier is confirmed by simulations. Compared with stateful logic designs, such as IMLPY logic, proposed circuit has a considerably higher speed, does not require a sequencer, is CMOS compatible and needs only one supply voltage level. Simulation results confirm circuit's functionality. This design can be utilized in long length multipliers and ALUs.

REFERENCES

[1] 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.

[2] J. Borghetti, Z. Li, J. Strasnicky, X. Li, D. A. A. Ohlberg, W. Wu, D. R. Stewart, and R. S. Williams, "A hybrid nanomemristor/transistor logic circuit capable of self programming." Proceedings of the National Academy of Sciences, vol. 106, no. 6, pp. 1699-1703, 2009.

[3] S. Kvatinsky, D. Belousov, S. Liman, G. Satat, N. Wald, E.G. Friedman, A. Kolodny and U. C. Weiser, "MAGIC–memristor aided LoGIC." IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 61, no. 11 pp. 895-899, 2014.

[4] S. Kvatinsky, N. Wald, G. Satat, 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), vol. PP, pp. 1–13, 2013.

[5] S. Kvatinsky, N. Wald, G. Satat, E. G. Friedman, A. Kolodny, and U. C. Weiser, "MRL – Memristor Ratioed Logic," Proceedings of the International Cellular Nanoscale Networks and their Applications, pp. 1-6, August 2012.

[6] L. Guckert and E. E. Swartzlander, "MAD Gates—Memristor Logic Design Using Driver Circuitry," in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 64, no. 2, pp. 171-175, Feb. 2017.

[7] L. Guckert and E. E. Swartzlander, "Optimized Memristor-Based Multipliers," in IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 64, no. 2, pp. 373-385, Feb. 2017.

[8] P. L. Thangkhiew, R. Gharpinde, P. V. Chowdhary, K. Datta and I. Sengupta, "Area efficient implementation of ripple carry adder using memristor crossbar arrays," 2016 11th International Design & Test Symposium (IDT), pp. 142-147, 2016.

[9] A. Chakraborty, A. Dhara and H. Rahaman, "Design of memristor-based up-down counter using material implication logic," 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 269-274, 2016.

[10] A. Amirsoleimani, M. Ahmadi, M. Teimoory, and A. Ahmadi. "Memristor-based 4: 2 compressor cells design." In Circuits and Systems (ISCAS), 2016 IEEE International Symposium on, pp. 1242-1245. IEEE, 2016.

[11] M. Teimoory, A. Amirsoleimani, J. Shamsi, A. Ahmadi, S. Alirezaee, and M. Ahmadi. "Optimized implementation of memristor-based full adder by material implication logic." In Electronics, Circuits and Systems (ICECS), 2014 21st IEEE International Conference on, pp. 562-565. IEEE, 2014.

[12] M. Teimoory, A. Amirsoleimani, A. Ahmadi, S. Alirezaee, S. Salimpour, and M. Ahmadi. "Memristor-based linear feedback shift register based on material implication logic." In Circuit Theory and Design (ECCTD), 2015 European Conference on, pp. 1-4. IEEE, 2015.

[13] K. Cho, S.-J. Lee, K. Eshraghian, "Memristor-CMOS logic and digital computational components," Microelectronics Journal vol. 46, no. 3, pp. 214–220, 2015.

[14] Swartzlander Jr, Earl E. "Memristor-Based Addition and Multiplication." In Memristor Networks, pp. 473-486. Springer International Publishing, 2014.

[15] Tejinder Singh, "Hybrid Memristor-CMOS (MeMOS) based Logic Gates and Adder Circuits," CoRR, arXiv:1506.06735 [cs.ET], pp. 1-11, June 2015 [Pre-Print].