# A Compact memristor-CMOS hybrid Look-up-table Design and Potential Application in FPGA

Yanwen Guo, Xiaoping Wang, and Zhigang Zeng

***Abstract*—Due to the conventional look-up-table (LUT) using the static random access memory (SRAM) cell, field programmable gate arrays (FPGAs) almost reach the limitation in term of the density, speed and configuration overhead. This paper proposes an improved memristor-based look-up-table (MLUT) circuit which is compatible with the mainstream LUT circuit in FPGA. Any arbitrary combined logic functions can be implemented in the MLUT through specific configurations. Then the MLUT shows superior advantages over the conventional LUT such as smaller area overhead and fewer data transmission. As a case study, a one-bit full adder is simulated to verify that the design is of practice in PSPICE. Moreover, the adder can be cascaded into multi-bit full adder demonstrating competitiveness against the conventional configurable logic block in FPGA technology. MLUT can be a candidate to replace the conventional SRAM-based LUT and further improves the performance of FPGAs.**

***Keywords*—memristor, nonvolatile reconfigurable memory, look-up-tables, logic computing.**

## I. Introduction

Memristor is postulated by professor Chua in 1971, as a missing circuit element [1]. Later the first successful fabrication is implemented by HP Labs in 2008 [2], which arouses extensive interest of industry and academia owing the advantages of nano-scale dimension, nonvolatility, fast access and high density in comparison to the complementary metal oxide semiconductor (CMOS) technology. The resistance of memristor depends on the history of the applied electronic signal (voltage or current), which can be tuned to any arbitrary resistive state in the permissible range with an appropriate applied voltage/current. The nonvolatile character of memristor makes it a competitive substitution for the storage element. Taking into account that the sneak path current is inevitable in pure memristor crossbar arrays [3], the memristor-CMOS hybrid architecture is an effective approach to overcome the defect of sneak paths [4], shows the compatibility with current mainstream CMOS technology and can be easily accessed [5], [6]. On the other hand, memristor shows good performance in logic computing [7], [8], [9], [10], [11].

The work is supported by the National Natural Science Foundation of China (Grant Nos. 61374150, 11271146 and 61503418), the State Key Program of the National Natural Science Foundation of China (Grant No. 61134012), and the Doctoral Fund of Ministry of Education of China (Grant No. 20130142130012).

The authors are with the School of Automation and the Key Laboratory of Image Processing and Intelligent Control of Education Ministry of China, Huazhong University of Science and Technology, Wuhan 430074, China (email: wangxiaoping@hust.edu.cn).



Field Programmable Gate Array is a powerful tool to implement complex computing and high speed digital signal processing, having been applied widely in the information technology field. As known to all, key elements of Configurable Logic Blocks (CLB) in FPGA are LUTs, D flip-flops and carry control logic [12]. Previous researches about LUT focus on that how the LUT size affects the performance of FPGA [13], [14]. In this paper, combining FPGA with the memristor technology, a novel look-up-table is designed, which does not need download configuration information from external storage memory (the configuration information can be stored in memristors). That is to say, both of the conventional read only memory (ROM) and the look-up-table are replaced by the novel memristor-based look-up-table, which takes up much smaller area overhead and shows more efficient data transmission. The proposed MLUT focuses more about the detailed circuit implementation and performance analysis, the circuit structure that utilizes a more compact decoder is simpler than that in [15]. The improvement utilizing memristor for the whole FPGA architecture is emphasized in [16], [17], [18], however, the analysis and discussion of MLUT circuit detailed implementation are not much. On the other hand, we simplify the basic cell structure (2T1M) in comparison with that in [19] (3T1M).

In this study, a MLUT-based simplified CLB of FPGA is proposed and simulated. Combining memristor and CMOS, a configurable logic block architecture with possible higher density and better efficiency is revealed. The rest of the paper is organized as follows: The conventional CLB of FPGA is depicted briefly, then reviews about memristor and the combination with CMOS technology are given in Section II. The circuit design of MLUT in FPGA using memristor-CMOS hybrid approach is presented in Section III. Comprehensive results and analysis of circuit simulation are discussed in Section IV. In the end, the paper is concluded in section V.

## II. Preliminaries

### A. Conventional Configurable Logic Block

CLBs are the main logic resources for implementing sequential as well as combinatorial logic circuits. The CLB details and varying capabilities are described in the Spartan-6 FPGA Configurable Logic Block User Guide [12]. Each CLB contains a pair of slices. Every slice contains four logic function generators (look-up-tables) and eight storage elements. Four of these storage elements in a slice that can be configured as either edge-triggered D-type flip-flops or level-sensitive latches for implementing sequential logic.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCAD.2017.2681079, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems



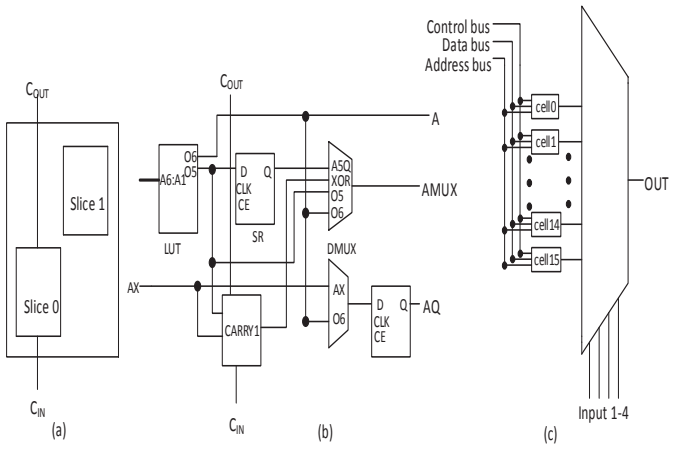


Fig. 1. (a) Arrangement of slices within a CLB. (b) Quarter of a slice. (c) Schematic of conventional four-input LUT.

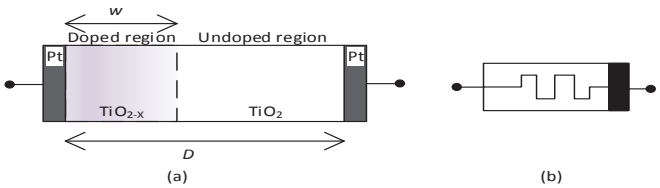


Fig. 2. (a) Memristor device structure. (b) Symbol for memristor.

Furthermore, each function generator can implement a 64-bit ROM. ROM contents are loaded at each device configuration. The function generators in Spartan-6 FPGA are implemented as six-input LUTs. There are six independent inputs and two independent outputs for each of the four function generators in a slice. The function generators can implement any arbitrarily defined six-input boolean function. In addition to the basic LUTs, a slice contains several multiplexers. These multiplexers are used to combine up to four function generators to provide any function of eight inputs in a slice, shown in Fig. 1. Note that we use four-input LUTs as the example in this paper.

### B. Memristor Character

The $TiO_2$-based thin film memristor fabricated successfully by HP Labs [2] is one of the most popular representatives in many types of memristor. The $TiO_2$ thin film of width $D$, sandwiched between two metal electrodes, is divided into two regions, as shown in Fig. 2. The one region is made up of pure $TiO_2$ with a highly resistive undoped region. The other one of width $w(t)$ contains $TiO_{2-x}$ with a highly conductive doped region. The total resistance of memristor (memristance) is determined by the parameter $w(t)$. Through the normalization, $x(t) = w(t)/D$, called the state variable of memristor, can better depict the inherent character of memristor. Note that $x(t)$ is limited to vary between 0 and 1, where $x(t) = 0$ ($x(t) = 1$) means that the current memristance is $R_{off}$ ($R_{on}$). $R_{off}$ is the maximum memristance and $R_{on}$ is the minimum memristance. Applying a positive voltage to the doped region expands the doped region, decreasing the memristance. Similarly, applying a negative voltage to the doped region shrinks the doped region, increasing the memristance [20]. Thus, memristance can be tuned to any arbitrary resistive state in the permissible range with an appropriate applied voltage/current. The modified memristor model adopted in the paper is reported in [2], [20], [21]. According to [6], ionic mobility $\mu_\nu$ is set to $10^{-7}$, to control the memristor programing time scale.

Considering the threshold character of memristor [3], when the absolute value of voltage is less than the threshold, the memristance will not change in the computing circuit. Supposing the absolute value of positive and negative threshold voltages both are 1V, we utilize the specific small voltage under threshold to work as computing mode. When the voltage is larger than the threshold, the memristance will change depending on the applied voltage.

## III. Memristor-based look-up-table design

In this section, the basic reconfiguration logic design is presented in [19]. The memristor-CMOS hybrid structure can provide a computationally complete combinational logic function. A digital binary decoder is necessarily used to select the right single output out of the specific input signals combinations in the structure. Correspondingly, a digital $n$-$2^n$ decoder is a necessity in $n$-input look-up-tables, which selects a single output according to the dedicated logic function. The detailed schematic circuit works in two modes: computing mode and configuring mode.

### A. computing mode circuit design

The computing architecture is made up of: NMOS and PMOS field effect transistors, resistors, memristors, a decoder and a comparator. The detailed working process is depicted in Fig. 3(a). The voltage VC, which is less than the threshold voltage, supplies a power for the circuit. The voltage VR, which is larger than the threshold, delivers the voltages required to switch memristors $M_i$ between $R_{off}$ and $R_{on}$. In addition, node K selects logic 0/1 to turn on/off switch transistors, then to compute/configure the reconfigurable logic gate respectively.

Taking Fig. 4 as an example, when the decoder inputs A1 and A2 both are equal to logic one, the output port Y4 of the decoder is logic one, while other ports Y1,Y2 and Y3 are logic zero. Under this circumstance, the gate electrode of NMOS transistor T11 is in the ON state, the gate electrodes of NMOS transistors T5,T7 and T9 remain in the OFF state. Assuming node K is connected to logic zero, the gate electrodes of PMOS transistors T1 and T3 will be in the ON state. While gate electrodes of NMOS transistors T2 and T4 will be in the OFF state. Hence, the VC will be connected in series with the resistor R1 and the drain terminal of NMOS transistor T11. Moreover, memristors will be connected to ground through the PMOS transistor T3. The circuit current flow is showed in Fig. 3(a) with dotted lines.

Supposing that the channel resistance for all transistors can be neglected, the voltage across the chosen memristor $M_i$(i=1, 2, 3, 4) is given by the following equation

$$V(M_i) = \begin{cases} V_C \frac{M_i}{R_1 + M_i}, & Y_i = 1 \\ 0, & Y_i = 0 \end{cases} \quad (1)$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCAD.2017.2681079, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems



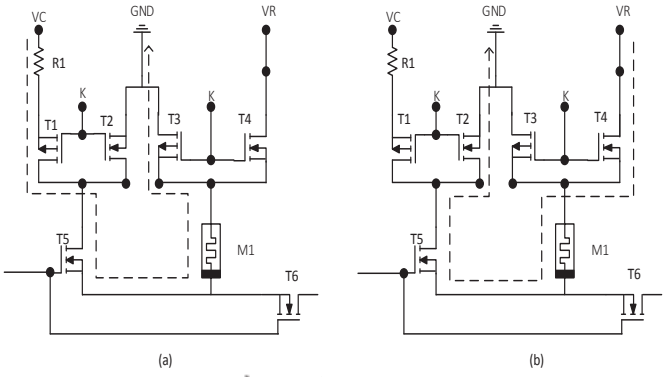


Fig. 3. (a) Schematic of the computing mode circuit current flow (K=0). (b) Schematic of the configuring mode circuit current flow (K=1).

$M_i$(i=1, 2, 3, 4) and $Y_i$ are the $i$-th memristor and $i$-th decoder output in Fig. 4, respectively. $V_C$ is the voltage source, whose value is set to 0.9V. The parameters of memristor model mentioned in the subsection II-B are $R_{off}$=100kΩ, $R_{on}$=1kΩ. To achieve the maximum read sensing margin, the resistance (R1) value is the geometric mean of $R_{on}$ and $R_{off}$, i.e., $R$1=10kΩ [22]. Then $V(M_i)$ can vary from $V(M_i = R_{on})$ = 0.082 V up to $V(M_i = R_{off})$ = 0.818 V. When Y4=1, the NMOS transistor T12 will be in the ON state. This will deliver the voltage potential $V(M_i)$ to the input of the voltage tuning block circuit (a comparator). The function of the voltage tuning block is to set the input voltage to the TTL voltage. That is to say, the output port will be 5V if the input of the block $V(M_i)$ is larger than 0.5V. While the output port will be 0V if $V(M_i)$ is lesser than 0.5V.

Note that there are one memristor and two transistors for each decoder output. The MLUT output directly depends on the state of the memristor connected with it. Thus the overall output of the logic circuit just relies on a single memristor device that is chosen through the decoder, while the states of all other memristors can be ignored. Besides, compared with [19], the decoupling transistor can be omitted because the voltage of each unselected decoder output is 0V, which does not impact the chosen cell. Here we use the NAND function as an example to validate that the structure can implement any two-bit combinational logic. The true table of the two-input computing logic NAND function configuration is showed in Table. I. Moreover, the XOR function is also designed in the MLUT since the specific logic function will be applied in the following adder design in the CLB of FPGA. The true table is also showed in Table. II. Here, all memristors are configured well beforehand using the configuring mode circuit.

TABLE I
TRUTH TABLE OF TWO-INPUT COMPUTING LOGIC: NAND

| Input Value | | Decoder output selection | Memristor state configuration | Output value |
|---|---|---|---|---|
| in1 | in2 | | | |
| 0 | 0 | Y1 | M1=$R_{off}$ | 1 |
| 0 | 1 | Y2 | M2=$R_{off}$ | 1 |
| 1 | 0 | Y3 | M3=$R_{off}$ | 1 |
| 1 | 1 | Y4 | M4=$R_{on}$ | 0 |

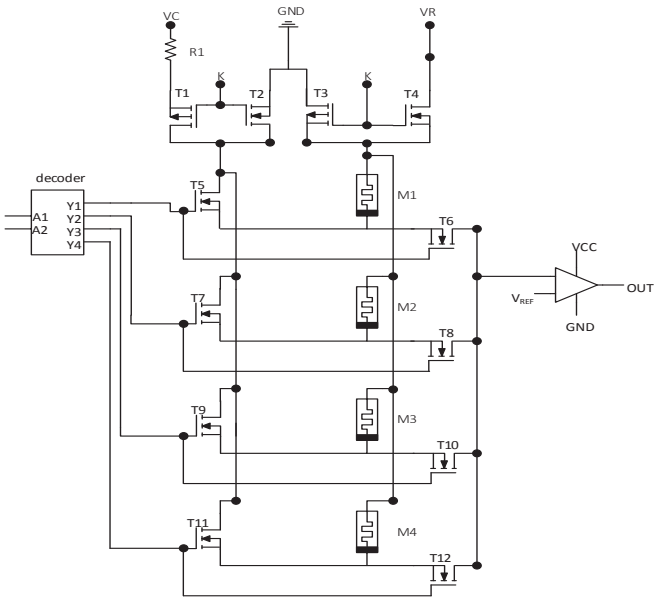


Fig. 4. Detailed schematic of memristor-based two-input look-up-tables.

TABLE II
TRUTH TABLE OF TWO-INPUT COMPUTING LOGIC: XOR

| Input Value | | Decoder output selection | Memristor state configuration | Output value |
|---|---|---|---|---|
| in1 | in2 | | | |
| 0 | 0 | Y1 | M1=$R_{on}$ | 0 |
| 0 | 1 | Y2 | M2=$R_{off}$ | 1 |
| 1 | 0 | Y3 | M3=$R_{off}$ | 1 |
| 1 | 1 | Y4 | M4=$R_{on}$ | 0 |

### *B. Configuring Mode Circuit Design*

The logic reconfiguration of all memristors can be achieved in the circuit schematic in Fig. 3(b). The node K is set to logic one. This makes the gate electrodes of NMOS transistors T4 and T2 be in the ON state, while the gate electrodes of PMOS transistors T1 and T3 will be in the OFF state. Thus, a direct path from node $V_R$ to ground is now established and node $V_C$ is isolated. Now that the memristor is connected directly to ground, node $V_R$ can be used to apply the above-threshold voltage to set the memristor device to any continuous analog state with proper continuous time. Only one memristor can be configured at a time and the memristor can be reconfigured corresponding to the output port selected by the decoder. Thus, any particular logic function can be implemented by configuring one memristor at a time through several steps. The step count is equal to the amount of the memristors used in the MLUT.

### *C. Discussion on MLUT Circuit*

When the cell is working on the computing mode, namely, K=-5v, VC=0.9v, Y1=5V, then PMOS transistors T1,T3 and NMOS transistor T5 are in the ON state. To avoid the boundary effect of memristor, we assume $R'_{off}$=90kΩ. $V(M_i)$ is calculated as 90/(10+90)*VC=0.81V according to Eq. (1). The simulation result is 801.8mV. The difference between the calculation and the simulation result is just 1.01%, which is

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCAD.2017.2681079, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems



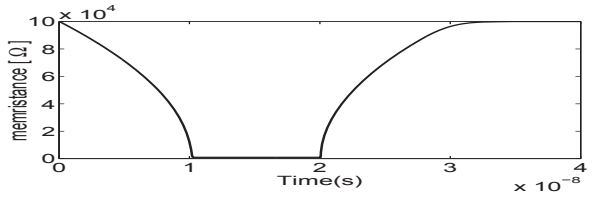


Fig. 5. Tuning memristance from low to high when VR=5V takes 10.3ns. While tuning memristance from high to low when VR=-5V takes 13.4ns.

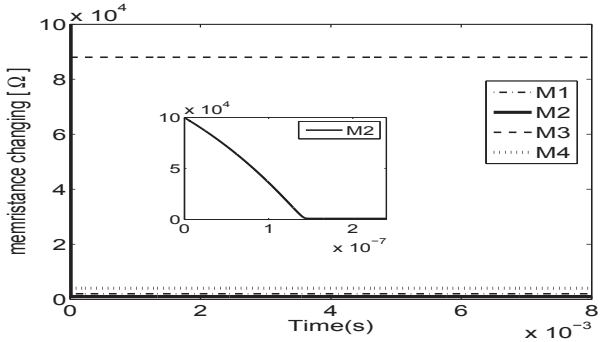


Fig. 6. The memristor M2 is changing from $R_{off}$ to $R_{on}$ when VR=5V (here memristance changing process of M2 is shown in the local magnifying figure). The other unchosen memristors varying little vertify the robustness against the sneak path current interference.

acceptable in digital logic. Similarly, we assume $R'_{on}$=2kΩ. $V(M_i)$ is calculated as 2/(10+2)*VC=0.15V. The simulation result is 148.2mV. Thus, the difference is about 1.2%.

When the cell is working on the configuring mode, namely, K=5v, VR=5v, then NMOS transistors T4, T5 and T2 are in the ON state, T2 and T4 are working on the saturation region and work well enough. However, T5 is working on the nonlinear region. So it does not work under the perfect circumstance, but the configuration is still correct. In Fig. 5, the memristance of the chosen memristor can be tuned from low resistance to high resistance when VR=-5V. Similarly, the memristance can be tuned from high resistance to low resistance when VR=5V. That is to say, the configuring mode circuit can work correctly.

Writing power for one memristor is 3.65pJ while the power is 4.06pJ in [5]. Reading power for one memristor is 162fJ while the power is 188fJ in [5]. The power consumption is similar because both are utilizing memristor-CMOS hybrid structures. While the operating power for one SRAM cell is 5fJ, much smaller compared with memristor.

The reconfigurable architecture is improved compared with [19]. Each basic cell can save one transistor. Considering there are millions of cells in an FPGA chip, the overall area saved by using the more compact cell will be substantial. Another advantage of this design is that there is almost no sneak path current, compared with the pure memristor crossbar array [4]. The memristances of unchosen memristors do change little, validating nearly no sneak path current in the structure, shown in Fig. 6.

## IV. Application of MLUT and Simulation

The proposed circuit design combines the computing logic and configuring method. Thus, a relatively entire circuit system is presented. Utilizing the MLUT circuit, a common full adder is designed and analyzed in detail. This design will offer a good reference for the engineers in memristor circuit implementation [23].

We use the modified SPICE netlist in [21] for the memristor and general CMOS chips OP-07, 74HC138, IRF250 and IFR9130 for logic cells to start the PSPICE simulation in CADENCE environment. The four-input LUT circuit can be encapsulated as a block in Fig. 7. The conventional CLB utilizes SRAM-based LUT, carry control logic and D flip-flop to implement the function of $n$-bit adder. With reference to section II-A, one bit full adder is designed utilizing MLUT, shown in the dashed line in Fig. 7. The logic operation of the adder is shown as the following equations

$$S_i = A_i \oplus B_i \oplus C_{i-1} \tag{2}$$

$$C_{OUT} = (A_i * B_i) + C_{i-1} * (A_i \oplus B_i) \tag{3}$$

The MLUT-0 used in Fig. 7 has been configured well to implement XOR function shown in Table. II. The simulation is shown in Fig. 8, which totally agrees with the theoretical analysis in Eq. (2) and Eq. (3). The inner working mechanism of LUT is mainly reflected by the voltage before and after the comparator, shown in Fig. 8. The XOR logic function is correctly implemented by the MLUT. Moreover, the one-bit full adder can be cascaded, showing the compatibility with the current FPGA technology. The MLUT-based $n$-bit full adder is a ripple carry adder in Fig. 7. The comparison between 8-bit adder in [24] and proposed 8-bit adder is shown in Table. III.

TABLE III
Comparison between 8-bit adder in [24] and proposed adder

| | Adder in [24] | Proposed adder |
|---|---|---|
| Device count | (9M)*8 | (4M+34T)*8 |
| Execution steps | 58 | 8 |
| CMOS Compatibility | no (need auxiliary circuits) | yes |
| Logic reconfiguration | yes (need redesign steps) | yes |

TABLE IV
Comparison of memristor count and transistor count among conventional LUT, MLUT in [15] and proposed MLUT

| $n$-inputs | Conventional LUT | MLUT in [15] | Proposed MLUT |
|---|---|---|---|
| 2 | 24T | 4M+24T | 4M+12T |
| 4 | 96T | 16M+62T | 16M+36T |
| 6 | 384T | 64M+146T | 64M+132T |

In Fig. 4, the schematic of two-input proposed MLUT is presented. The conceptional diagram of conventional four-input LUT is shown in Fig. 1(c). The MLUT in [15] adopts relatively complex and redundant decoders that leads larger area overhead while not considering the cascading. The difference among the three structures is shown in Table. IV. The conventional two-input LUT at least needs 4 storage elements(6T*4) and a decoder. Moreover, the stored information in the LUT needs extra steps to be downloaded from external ROM. Whereas the proposed design just needs (1M+2T)*4, the voltage tuning block and the decoder. So the proposed

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCAD.2017.2681079, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems



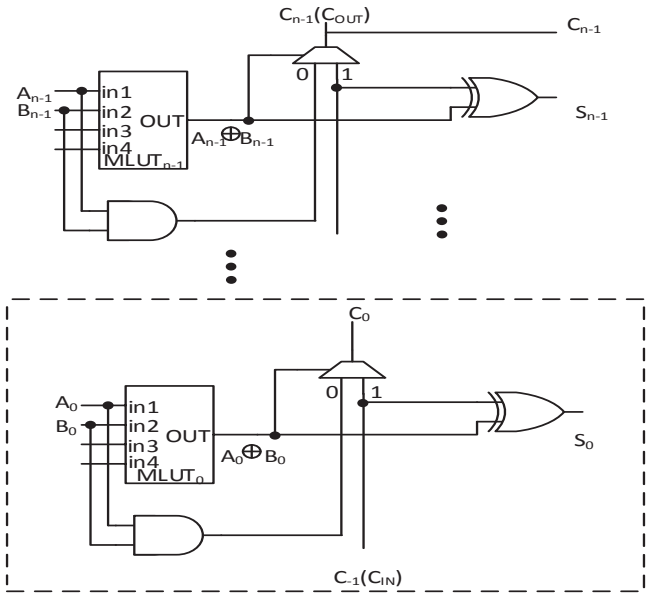


Fig. 7. Schematic diagram of n-bit full adder.

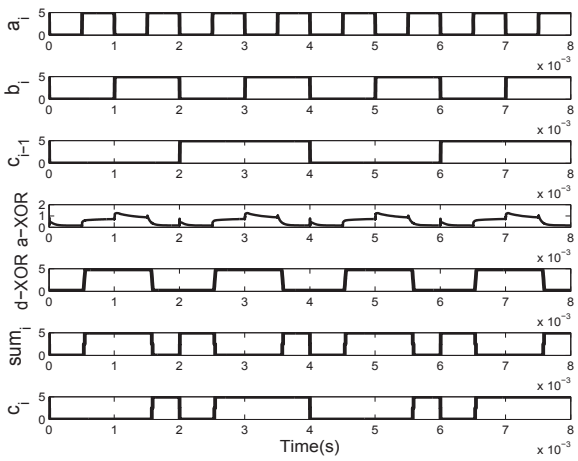


Fig. 8. Simulation results of one-bit full adder. The $a_i$ ,$b_i$ and $c_{i-1}$ are inputs, a-XOR is the analog output of MLUT with XOR logic function (a-XOR also is the voltage before the comparator), d-XOR is the digital output of the specific MLUT (d-XOR also is the voltage after the comparator), $sum_i$ and $c_i$ are outputs. All vertical coordinate unit is Volts.

method saves about 12T in the two-input LUT. In the $n$-input LUT the proposed method at least saves about $3* 2^n$ transistors. Additionally, the proposed design can retain the stored information in the memristor cell when the power is off, which is not easy to realize using the conventional design. These two advantages ensure that the proposed design facilitates the novel FPGA design development.

## V. Conclusion

The detailed memristor-based LUT design is put forward in this paper. The circuit and sub-circuit performance are analyzed. Compared with the previous literatures, this design shows the advantages of taking smaller area overhead and eliminating external ROM. We are confident that this research will facilitate the novel FPGA technology development. Future work will focus on the memristor analog features aiming to achieve much greater improvements on the FPGA design.

## References

[1] L. O. Chua, "Memristor-the missing circuit element," *IEEE Trans. Circuit Theory*, vol. 18, no. 5, Sep. 1971.
[2] D. B. Strukov, G. S. Snider, D. R. Stewart, and R. S. Williams, "The missing memristor found," *Nature*, vol. 453, pp. 80-83, May 2008.
[3] P. O. Vontobel, W. Robinett, P. J. Kuekes, D. R. Stewart, J. Straznicky, and R. S. Williams, "Writing to and reading from a nano-scale crossbar memory based on memristors," *Nanotechnology*, vol. 20, no. 42, p. 425204, 2009.
[4] M. A. Zidan, H. Omran, A. Sultan, H. Fahmy, and K. N. Salama, "Compensated Readout for High-Density MOS-Gated Memristor Crossbar Array,", *IEEE Trans. Nanotechnol.*, vo. 14, no. 1, pp. 3-6, 2015.
[5] M. Zangeneh and A. Joshi, "Design and optimization of nonvolatile multibit 1T1R resistive RAM," *IEEE Trans. Very Large Scale Integr. (VLSI),* vol. 22, no. 8, pp. 1815-1828, 2014.
[6] B. Mohammad, D. Homouz, and H. Elgabra, "Robust hybrid memristor-cmos memory: modeling and design," *IEEE Trans. Very Large Scale Integr. (VLSI),* vol. 21, no. 11, pp. 2069-2079, 2013.
[7] J. Borghetti, G. S. Snider, P. J. Kuekes, J. J. Yang, D. R. Stewart, and R. S. Williams, "Memristive switches enable stateful logic operations via material implication," *Nature*, vol. 464, no. 7290, pp. 873-876, 2010.
[8] K. H. Kim, S. Gaba, D. Wheeler, J. M. Cruz-Albrecht, T. Hussain, N. Srinivasa, and W. Lu, "A functional hybrid memristor crossbar-array/CMOS system for data storage and neuromorphic applications," *Nano Lett.*, vol. 12, no. 1, pp. 389-395, 2011.
[9] 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," *Nat. Nanotechnol.*, vol. 3, no. 7, pp. 429-433, 2008.
[10] Q. Xia *et al.*, "Memristor-CMOS hybrid integrated circuits for reconfigurable logic," *Nano Lett.*, vol. 9, no. 10, pp. 3640-3645, 2009.
[11] G. S. Snider, "Architecture and methods for computing with reconfigurable resistor crossbar," U.S. Patent 7,203, 789 (2007).
[12] Spartan-6 FPGA Configurable Logic Block User Guide. 2010.
[13] E. Ahmed and J. Rose, "The effect of LUT and cluster size on deep-submicron FPGA performance and density," *IEEE Trans. Very Large Scale Integr. (VLSI),* vol. 12, no. 3, pp. 288-298, 2004.
[14] I. Kuon and J. Rose, "Area and delay trade-offs in the circuit and architecture design of FPGAs," in *Proc. Int. Symp. Field-Program. Gate Arrays*, 2008, pp. 149-158.
[15] T. N. Kumar, H. A. F.Almurib, and F.Lombardi, "A novel design of a memristor-based look-up table (LUT) for FPGA," 2014 IEEE Asia Pacific Conference on Circuits and Systems, 2014, pp. 703-706.
[16] W. Wang, T. T. Jing, and B. Butcher, "FPGA based on integration of memristors and CMOS devices," in Proc. IEEE Int. Symp. Circuits Syst., 2010, pp. 1963-1966.
[17] J. Cong and B. Xiao, "mrFPGA: A novel FPGA architecture with memristor-based reconfiguration," 2011 IEEE/ACM Int. Symp. on Nanoscale Architectures, 2011, pp. 1-8.
[18] J. Wu and M. Choi, "Memristor lookup table (MLUT)-based asynchronous nanowire crossbar architecture," 2010 10th IEEE Conference on Nanotechnology, 2010, pp. 1100-1103.
[19] R. E. Pino and Y. K.Pino, "Reconfigurable memristor-based computing logic," U.S. Patent 8,427,203 (2013).
[20] Y. Ho, G. M. Huang, and P. Li, "Dynamical Properties and Design Analysis for Nonvolatile Memristor Memories," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 58, vo. 4, Apr. 2011.
[21] Z. Biolek, D. Biolek, and V. Biolova, "SPICE Model of Memristor with Nonlinear Dopant Drift," *Radioengineering*,vol. 18, no. 2, pp. 210-214, 2009.
[22] S. Smaili and Y. Massoud, "Memristor state to logic mapping for optimal noise margin in memristor memories," in *Proc. IEEE Int. Conf. Nano.*, 2014, pp. 291-295.
[23] S. Kvatinsky, E. G. Friedman, A. Kolodny, and U. C. Weiser, "The desired memristor for circuit designers," *IEEE Circuits Syst. Mag.*, vol. 13, no. 2, pp. 17-22, 2013.
[24] 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 Trans. Very Large Scale Integr. (VLSI),* vol. 22,no. 10, pp. 2054-2066, 2014.