Indian Journal of Science and Technology, Vol 9(5), DOI: 10.17485/ijst/2016/v9i5/87151, February 2016
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

A Neoteric FPGA Architecture with Memristor Based Interconnects for Efficient Power Consumption

A. Karthikeyan*, J. Arunarasi and A. Arul Mary

Department of Electronics and Communication Engineering, Vel Tech Multitech, Avadi, Chennai – 600062, Tamil Nadu, India; a.karthik1982@gmail.com, arunarasij@veltechmultitech.org, arulmary@veltechmultitech.org

Abstract

Objective: This paper discusses different ways of power consumption in FPGA and the earlier SRAM based power reduction techniques with its limitations and demonstrates the methodology adapted by the proposed neoteric Memristor based FPGA architecture. The programmable interconnects of Memristor based FPGA architecture uses the newly found circuit element, i.e. Memristor instead of the SRAM based interconnects as in the usual FPGA architecture, resulting in significant reduction of overall power consumption, area, propagation delay which in turn increases the speed of FPGA than the other conventional SRAM based FPGA architecture. **Statistical Methods:** The study on the various architectures of FPGA aiming at minimizing the power consumption with the application of different techniques is carried out earlier. One such technique is the SRAM power reduction technique which has got its own limitations such as low density, longer routing path, increased propagation delay, increased peak power consumption rate as well as high value of average power consumption. This paper proposes a low area with the help of Memristor in order to such a minimal routing path aiming to reduce the drawbacks as identified in SRAM architecture. **Findings:** A simulation work has been carried out with the help of HSPICE and the results prove that the peak power consumption, average power consumption, propagation delay are reduced to a larger extent than that of the reported SRAM architecture. **Applications/Improvements**: Further it is recorded that the Memristor based FPGA architecture suits better for the applications where there is minimal energy requirement on wireless network scenarios.

Keywords: CMOS Technologies, Field-Programmable Gate Arrays (FPGA), Low Power Consumption, Memristor, SRAM, Very Large Scale Integrated Circuits

1. Introduction

As a result of the developing technology, in recent times the low power very large scale circuits especially in the case of Field-Programmable Gate Arrays (FPGA's) are gaining prominence. FPGA's are more popular in use than the Application Specific Integrated Circuits (ASIC's)[1,2]. But the main problem with the FPGA is that it is less power efficient than the ASIC[3]. Studies on FPGA show that they are between 9 and 12 times less power efficient than the ASIC's[4]. Despite the several advantages of FPGA like its programmability and flexibility, it suffers certain shortcomings in relation to the performance of ASIC's due to its high power consumption. These differences lead to an area, performance and power consumption gap between ASIC and FPGAs. These gaps have been studied extensively[5,6] and even more deeply in the later studies. So the important goal in the recent time is to design or modify the conventional FPGA architecture with low power consumption, which will in turn increase the performance, area and speed of these FPGA's. Generally the power consumed in the FPGA consists of both the static and dynamic components. The static power consumption is only about 10% of the total power consumed in an FPGA[7]. Further, the known techniques for reducing this static power consumption such as programmable supply voltage, multiple transistor threshold voltages, multiple gate oxide thicknesses and power gating, have already been used in the conventional FPGA's [8].

On the other hand, the dynamic power constitutes about 90% of the total power consumed in FPGA's and

*Author for correspondence

it is considered to be the main reason for these FPGA's power inefficiency. This high power consumption is due to the increase in the programming of FPGA's, including the presence of the pass transistor switches, buffers, multiplexers and the configuration of its memory used in the routing fabric which occupies 50%-80% of the total area. This results in much more longer routing paths than the ASIC implementations and hence increases the loading of the capacitors of these conventional FPGA's, because of this 60%-80% of the total FPGA's dynamic power is being consumed in these programmable routing fabrics[9,10].

Also the major power consumption in FPGA is due to its programmable routing structure. In the programmable routing structure, due to the presence of the programmable logic blocks and the programmable interconnects, more amount of the power is being consumed, which results for about 90% of the total area, about 85% of the total power consumption and about 80% of the total delay.

The conventional FPGA's suffers a lot from its programmable interconnects due to major use of the Static Random Access Memory (SRAM) based programming interconnects. This is because one SRAM cell contains about six transistors, but can store only one bit data. The low density of SRAM-based storage increases the area of FPGA programmability which in turn leads to longer routing paths and larger interconnect delay which has direct impact on the overall performance of the FPGA. In addition to this, SRAM is a volatile memory which consumes excessive power consumption during the stand-by mode and the hold mode. On the other end, the problem with the DRAM is that, it stores the data as electric charge which has to be refreshed time to time[10].

This paper is organized as follows: Section 2 describes the power consumption in FPGA. Section 3 describes the conventional SRAM based FPGA. Section 4 describes the Memristor based FPGA. Section 5 describes about modeling a stable Memristor. Section 6 describes about Memristor parameter analysis. Section 7 describes the implementation and the simulation results. Finally, Section 8 summarizes the paper and proposes the future work.

2. Power Consumption in FPGA

Power consumption in FPGA is mainly due to the static and dynamic power consumption, which in turn is due to the conventional structure of the FPGA .

2.1 Structure of a Conventional FPGA

It is made of several arrays of blocks and each block consists of one Logic Block (LB), two Connection Blocks (CB) and one Switch Block (SB). Each Logic Block contains a cluster of Basic Logic Elements (BLE's); the Lookup Tables (LUT's) for providing the logic functions. LB's are connected to the routing channels through CB's and the routing channels are connected with each other through the SB's.

The selector pins of each multiplexer are connected to the SRAM cell made of 6-transistors for their connectivity with much larger area and higher complexity compared to the direct interconnects of the ASIC. The programming technology for the logic and interconnect resources are the SRAM. The SRAM based FPGA's offer in-circuit reconfigurability though being volatile.

2.2 Sources of Power Dissipation in FPGA

The main sources of the power dissipation in the FPGA are the static, dynamic and the memory cell power consumption. FPGA's consume more power than the ASIC because of the presence of a large number of transistors which though provides flexibility causes both the static and dynamic power dissipation. The SRAM cells have to be slowed down and their leakage have to be reduced in order to reduce the total power consumption of FPGA. Furthermore, SRAM cells constitute a large space of the FPGA's total area which makes the wire lengths in FPGA's longer. These interconnects thus increases the high capacitive load in FPGAs, making it as the primary source of dynamic power dissipation.

2.3 Earlier Power Reduction Techniques

The earlier techniques that are commonly used to reduce the power consumption of the conventional FPGA are as follows:

2.3.1 Architectural Level Power Reduction Techniques

The architectural level power reduction techniques are fine grained-VDD, leakage reduction in FPGA routing multiplexers, low power programmable FPGA routing circuitry, clock gating power reduction technique, power gating and sub threshold FPGA.

2.3.2 Circuit Level Power Reduction Techniques

The circuit level power reduction techniques are a dual-threshold FPGA routing design, programmability of

VDD, input vector reordering, FPGA leakage power reduction using CLB- clustering and Dual Threshold Transistor Stacking (DTTS).

2.3.3 Device Level Power Reduction Techniques

The device level power reduction techniques are leakage power reductions from Tunnel FETs (TFETs) and carbon nano tube SRAM design.

The key problems in the related works are mainly due to the presence of the SRAM based interconnects in the FPGAs architecture which results in high power consumption. These related works are mainly focused on reducing the number of transistors in the SRAM. However, almost the same amount of power is being consumed once again, because of the access time, density and power dissipation of the SRAM in the FPGA.

3. Conventional SRAM based FPGA

The Conventional FPGA's are made of SRAM based programmable interconnects. The major power consumption in FPGA is due to its programmable routing structure made of SRAM. The main reason is that these SRAM memory cells are made up of six transistors.

The energy consumption in the SRAM includes two components: dynamic energy consumption and static energy consumption. Static energy consumption, consumed due to leakage current in SRAM and dynamic energy consumption consumed due to the charging and de-charging of capacitance during read and write operation. It is been found that the programmable interconnects in FPGA accounts for about 90% of the total area, about 80% of the total delay and about 85% of the total power consumption.

4. Memristor based FPGA

Ultra-Low Energy Sub-threshold FPGA which uses Single-VDD was proposed[11]. Minimum Energy Analysis and Experimental Verification of a Latch-based subthreshold FPGA was proposed[12]. In this paper the proposed methodology is aimed to reduce the power consumption of FPGA by using the Memristor based FPGA architecture. The methodology used here is the neoteric Memristor based FPGA architecture. Memristor based FPGA uses Memristor interconnects rather than programmable interconnects made of SRAM memory cells. Dnamically supporting ontrolled power gating which uses An FPGA architecture was given[13]. Proposed methodology is made of 3-D crossbar-based memory architecture with the 3D die-stacking process and also provides capacitance shielding from unused routing path. FPGA power reduction using gating architectures was given[14].

Low-Power programmable FPGA which uses routing circuitry was proposed[15]. FPGA routing multiplexers which uses Leakage reduction was given[16]. The 95% Leakage reduced FPGA using zigzag power-gating, Dual-VTH/VDD and Micro VDD hopping was given[17]. There are a number of studies which uses SPICE modeling of Memristors was proposed[18]. The application of Memristors in modeling unconventional devices shows that Memristors are useful even if they are used as a conceptual tool of analysis was given[19]. These systems are unconventional in the sense that while they behave like resistive devices was proposed[20].

The programmable interconnects of Memristor based FPGA architecture uses the newly found circuit element, i.e. Memristors instead of the SRAM based interconnects as in the usual FPGA architecture, resulting in significant reduction of overall area power consumption, area, interconnect delay and increases the speed of the FPGA. Compared to the existing system, it does not occupy more area because Memristor is a nano device. The main highlight of this approach is that it employs the Resistive Random Access Memory (RRAM) instead of the SRAM.

4.1 Memristor

Memristor is a fourth fundamental passive 2 terminal linear Resistor with Memory (RRAM). Memristor is a semiconductor thin film sandwiched between two metal contacts with a total length of D of TiO_2 film and it consists of doped low resistance and undoped high resistance regions and memristance arises naturally in nanoscale systems in which solid-state electronic and ionic transport are coupled under an external bias voltage was given[21].

Moreover, Memristors provide new paradigms in Application Specific Integrated Circuits (ASICs) and Field Programmable Gate Arrays (FPGAs). A significant reduction in area with an unprecedented memory capacity and device density are the potential advantages of Memristors for Integrated Circuits (ICs) was proposed[22] and architectural innovations in Virtex-5 devices to ensure that static power consumption is minimized and that the dynamic power benefits of moving to a new process node are fully realized in 65 nm FPGAs was given[23]. The amount of

power reduction depend not only on supply voltage but also on the threshold voltage to sustain the reduction of component delay, which is crucial for high speed digital circuit design was given[24].

The physical structure of Memristor is shown in Figure 1(a) and its equivalent model is shown in Figure 1(b). Resistance can be changed by changing the direction of applied voltage or current. Resistance increases when current flows in one direction and decreases when it flows in the other direction. Memristance arises naturally in nanoscale systems in which solid-state electronic and ionic transport are coupled under an external bias voltage.

When applied external potential is removed, Memristor remains in the last state. Memristor is capable of memorizing its own resistance, different resistance values can be used to represent different values of data.

The Memristive systems[25] can be described as follows:

$$M(x) = R_{ON}x + R_{OFF}(1-x) \quad (1)$$

Where, x is the state variable which has to satisfy the boundary conditions of 0 and 1 and 'M(x)' is the Memristance of a Memristor.

The non linear Memristance (M) is a function of charge (q). M depends on the history of current passing through the Memristor element.

This makes the Memristor act like a linear resistor with memory.

5. Modeling a Stable Memristor

In the Resistive RAM (RRAM) structure, the ability to selectively access one device without disturbing the other is the most difficult one. In an array, the problem of sensing the resistive memory is too high. Consequently when the leakage current is too high it would result in excessive voltage drops. This requires tuning the voltage level properly in the crossbar array for the adaptive read, write and erase method by the Memristor memory. One way to limit this effect is to allow for an adjustable reference resistor and design for specific leakage tolerance or by using a transistor or a diode which will help to stabilize the Memristor when it is used in an array. Figure 2 shows the structure of 1 transistor 1 Memristor model.

6. Memristor Parameter Analysis

The Memristor parameter analysis done in this paper is of four types - peak power, average power, propagation delay and the number of transistor calculations. Design exploration of hybrid CMOS and Memristor circuit by New Modified Nodal Analysis was given[26]. Novel Si-Tunnel FET based SRAM Design for Ultralow-Power 0.3V VDD applications was proposed[27]. FPGA leakage power reduction using CLB-clustering technique was given[28] and Input

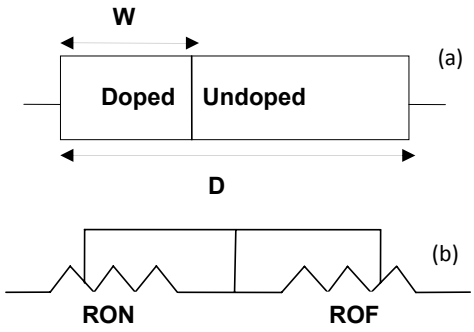

Figure 1. **(a)** Structure of Memristor. Memristor is a semiconductor thin film sandwiched between two metal contacts with a total length of D of TiO_2 film and it consists of doped low resistance of length of W and undoped high resistance regions. **(b).** Equivalent structure of Memristor. Memristor operates in one of two nonvolatile resistive states (ON or OFF). Here RON and ROFF represent the low and the high resistance.

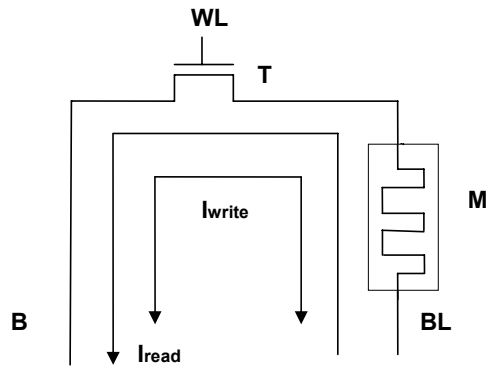

Figure 2. Structure of 1 Transistor 1 Memristor Model. This cell is made of one access control Transistor (T) and memory storage Memristor (M) has one memory cell and its possible interface connections to other cells as well as the IO circuitry. M is connected to the Bit Line Bar (BLB) and the WL control transistor that is connected to the BL line.

vector reordering for leakage power reduction in FPGAs was given[29]. FPGA power reduction which uses Field programmability of supply voltages was given[30]. Design for sub-threshold leakage reduction using A dual-threshold FPGA routing was propsed[31]. Monolithically Integrated Graphene Interconnects which uses demonstration of a Subthreshold FPGA was reported[32].

6.1 Power

Power calculations are done to calculate the amount of static power and dynamic power dissipated. In this paper the power calculations is carried through two types: the peak power and the average power calculation. The peak power is the rate of energy flow in each and every pulse whereas the average power is the rate of energy flow averaged over one full period. The amount of power dissipation along this peak and average power constitutes the total power dissipation. In this paper, the power consumption is got from the transient analysis of the HSPICE simulation tool.

$$Ppeak = E/\Delta t \quad (2)$$

$$Pavg = E/T \quad (3)$$

Where, 'E' is the energy flow and't' is the time period.

6.2 Propagation Delay

Transistors within the gates take a finite amount of time to switch (i.e.) a change on the input of a gate takes a finite amount of time to cause a change on the output. This time is known as Propagation Delay. If the structure is small then the switching times will be faster. In this paper the amount of the propagation delay is got from the transient analysis of the HSPICE simulation tool.

6.3 Number of Transistors

The number of transistors calculation is very important because this determines the density of the circuit and the overall area of the chip, which in turn relates to the power consumption also. Hence the number of transistors is also calculated in this work.

7. Implementation

The SRAM and Memristor parameter analysis is done by writing the HSPICE coding for both separately and the executing that coding in the HSPICEA-2007.09 (HspuiA-2007.09). There after simulation is carried and the parameters are obtained from the transient analysis of both SRAM and Memristor.

Later the output wave form of the SRAM and Memristor is got from the CosmosScope of CosmosScope(TM) Z-2007.03-SPI from Synopsys.

7.1 SRAM Parameter Analysis

SRAM parameter analysis is of two types: SRAM cell and SRAM cell in hold mode. In the output of the SRAM cell, there are three lines; they are the word line (wl), the bit line (bl) and the read output (Voutr). The "Voutr" line will be high, only when both the "bl" and "wl" lines are high. For reading a value from the SRAM, the word line (wl) should be kept high.

7.1.1 SRAM 6t Cell

For enabling the read operation or the write operation, enable the word line signal.

If the word line signal is near to zero then the SRAM cell will be in the standby mode.

For the write operation, both the bit lines are enabled high along with the word line.

For read operation, either of bit line is kept low and the other is kept high.

From Figure 3 it is seen that in the voltage graph the value written in "Voutr" line will be high, only when both the "bl" and "wl" lines are high. For reading a value from the SRAM, the word line (wl) should be kept high.

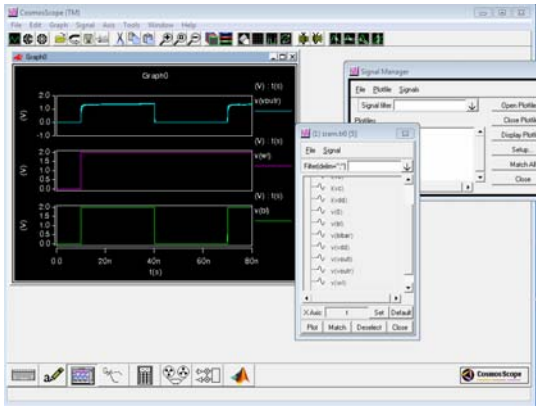

Figure 3. Read and write operation of SRAM Cell, where ——— - Voutr line, ——— - wl line, ——— - bl line. The "Voutr" line will be high, only when both the "bl" and "wl" lines are high.

Further this will come back to the reference value if "wr" and "rd" are low.

From the Figure 4 it is shown that SRAM's transient analysis gives the value of propagation delay, average and peak power values for a SRAM cell as the parameter output from the HSPICE simulation tool.

7.1.2 SRAM in Hold Mode

For SRAM in hold mode: When WL = 0; both NMOS will be in OFF state, SRAM is in HOLD mode.

From the Figure 5 it is shown that in the voltage graph "Voutr" line will be high, only when both the "bl" and "wl" lines are high. Again when "wl" goes low, "Voutr" holds the last output value waveform of the SRAM cell in hold mode.

For enabling read or write operation one needs to enable the word line signal. Here in the output of the SRAM cell in the hold mode, there are three lines; they are the word line (wl), the bit line (bl) and the read output (Voutr).

For the SRAM in the hold mode, whenever "bl" and "wl" will be high only then "Voutr" will be high. Again when "wl" goes low, "Voutr" holds the last output value (high state) that is read because to read a value from the SRAM, the word line (wl) should be kept high always.

```
******  transient analysis              tnom=  25.000 temp=  25.000
******
  propagation_delay=  1.5117E-10  targ=  4.0301E-08   trig=  4.0150E-08
  avgpower=  3.1008E-04  from=  1.0000E-09     to=  8.0000E-08
  peakpower=  2.3323E-03     at=  4.0304E-08
              from=  1.0000E-09     to=  8.0000E-08
```

Figure 4. Transient analysis of SRAM cell. SRAM's transient analysis gives the value of propagation delay, average and peak power values for a SRAM cell as the parameter output from the HSPICE simulation tool.

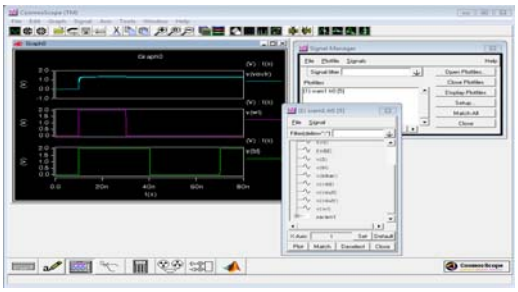

Figure 5. Read and write operation of SRAM Cell in Hold State, where — - Voutr line, — - wl line, — - bl line. When "wl" goes low, "Voutr" holds the last output value (high state) that is read because to read a value from the SRAM, the word line (wl) should be kept high always.

7.2 Memristor Parameter Analysis

As discussed earlier, the physical structure of Memristor is in such a way that the resistance can be changed by changing the direction of applied voltage or current. Resistance can be changed by changing the direction of applied voltage or current.

Resistance increases when current flows in one direction and decreases when it flows in the other direction. When applied external potential is removed, Memristor remains in the last state. That is why the Memristor is capable of memorizing its own resistance. Different resistance values can be used to represent different values of data.

In the Memristor output there are four lines, write line (wr), read line (rd), output data line (d) and combinational word line (comp). Here "comp" acts like the word line, only if the "comp" line is high, Memristor will write the value (i.e) "wr" will be high.

From the Figure 6 it is seen that in the voltage graph "comp" becomes high only when "wr" and "rd" becomes high. Further this will come back to the reference value if "wr" and "rd" are low.

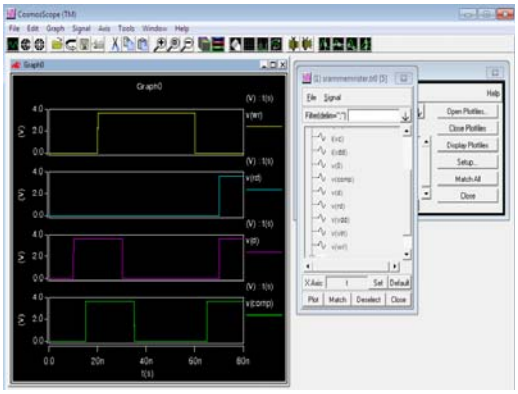

Figure 6. Read and write operation of Memristor Cell, where — - wr line (the write line), — - d line (output data line), — - rd line (the read line), — - comp line (the combinational word line). This value that is written in "wr" will be read only when "comp" becomes high (i.e.) after "wr" becomes high and again whenever "comp" becomes high, "rd" becomes high. Finally this read value is got as the output value.

From the Figure 7 it is shown that Memristor transient analysis gives the value of propagation delay, average and peak power values for a Memristor cell as the parameter output from the HSPICE simulation tool.

From the transient analysis of the Memristor there is a decrease in average power consumption, peak power consumption and the propagation delay for the increase in voltage in the two terminals Memristor.

From the Figures 4 and 7 (i.e.) from the transient analysis of the SRAM and the Memristor, the following SRAM and Memristor analysis is done:

Figure 8 represents the average power consumption for SRAM vs. Memristor. From this figure it is clearly evident that the average power consumption increases in SRAM architecture for the increase in voltage, whereas the Memristor shown the decrease in average power consumption for the increase in voltage. For example the potential of 2.5v the average power consumption is only 20μw for Memristor whereas the SRAM it is 350μw. Thus it is proved that Memristor is the best architecture for less average power consumption.

Figure 9 represents the propagation delay for SRAM vs. Memristor. From this figure it is found that the propagation delay increases in SRAM architecture for the increase in voltage, whereas the Memristor exhibited the decrease propagation delay for the increase in voltage. For example the potential of 2.5v the propagation delay is nearly 10ns for Memristor, whereas for the SRAM it is nearly 465ns. Thus it is proved that Memristor is the best architecture for less propagation delay.

Figure 10 represents the propagation delay for SRAM vs. Memristor. From this figure it is found that the propagation delay increases in SRAM architecture for the increase in voltage, whereas the Memristor exhibited the decrease propagation delay for the increase in voltage. For example the potential of 2.5v the propagation delay is nearly 10ns for Memristor, whereas for the SRAM it is nearly 465ns. Thus it is proved that Memristor is the best architecture for less propagation delay.

```
propagation_delay= -5.0000E-08  targ=  2.0050E-08   trig=  7.0050E-08
avgpower=  1.3189E-05  from=  1.0000E-09     to=  8.0000E-08
peakpower=  1.5264E-04     at=  1.5100E-08
           from=  1.0000E-09    to=  8.0000E-08
```

Figure 7. Transient analysis of Memristor cell. The transient analysis of the Memristor gives the value of propagation delay, average and peak power values for a Memristor cell as the parameter output from the HSPICE simulation tool.

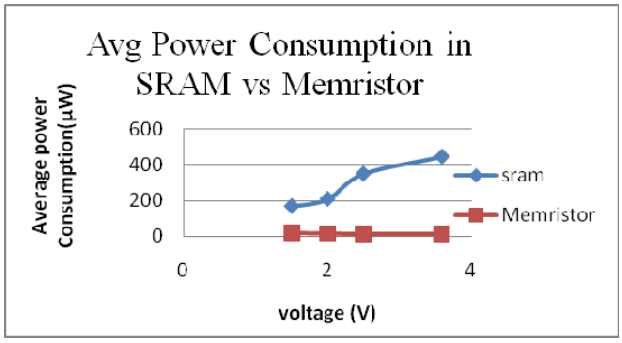

Figure 8. Average power consumption in SRAM vs. Memristor. Average power consumption increases in SRAM architecture for the increase in voltage, whereas the Memristor shown the decrease in average power consumption for the increase in voltage.

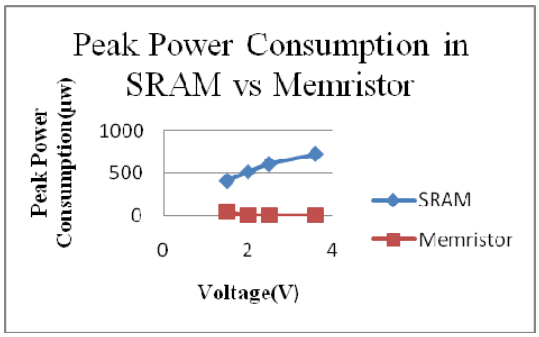

Figure 9. Peak Power Consumption in SRAM vs. Memristor. Peak power consumption increases in SRAM architecture for the increase in voltage, whereas the Memristor shown the decrease in peak power consumption for the increase in voltage.

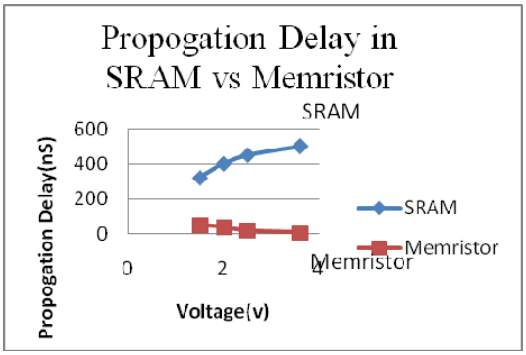

Figure 10. Propagation delay in SRAM vs. Memristor. Propagation delay increases in SRAM architecture for the increase in voltage, whereas the Memristor shown the decrease propagation delay for the increase in voltage.

8. Conclusion and Future Work

Implementing the Memristor interconnects in the neoteric Memristor based FPGA architecture rather than the conventional SRAM interconnects based architecture consumes less power and very less propagation delay. Moreover it does not occupy more area since Memristor is a nano device and hence has high density. The main highlight of this approach is that it employs the Resistive Random Access Memory instead of the Static Random Access Memory. From the simulation it is proved that the average power consumption for Memristor is always lesser than the existing SRAM architecture. Further a consistency in average power is also recorded as an output of simulation. It is also proved that the peak power consumption for Memristor is very less when compared to SRAM architecture for any selected voltage. The difference between peak power consumption for Memristor and SRAM is about 300 to 500 μW. Next the difference between average power consumption for Memristor and SRAM is about 200 to 400 μW. As a third achievement it is also proved that propagation delay for proposed Memristor records very less time in terms of 0 to 40 ns whereas the existing SRAM architecture the recorded propagation delay is about 320 to 500 ns. So, it is obvious that the proposed Memristor architecture has very less propagation delay than SRAM.

The Memristor based architecture presented in this paper could be extended to fabricate with the help of FPGA and real-time experiment can be carried out to prove the results obtained through simulation. Further, Memristor based security and Neuromorphic application circuits in FPGA can also be done as a future enhancement to this work.

9. References

1. George V. Low energy field-programmable gate array. Berkeley: University of California; 2000.
2. Li F, Lin Y, He L, Chen D. Power modeling and characteristics of field programmable gate arrays. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. 2005 Nov; 24(11):1712–24.
3. Shang L, Kaviani AS, Bathala K. Dynamic power consumption in Virtex-II FPGA family. Proc 2002 ACM/SIGDA 10th International Symposium on Field-Programmable Gate Arrays; 2002. p. 157–64.
4. Kuon I, Rose J. Measuring the gap between FPGAs and ASICs. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. 2007 Feb; 26(2):203–15.
5. Chinnery D, Keutzer K. Closing the gap between ASIC and custom tools and techniques for high-performance ASIC design. Norwell, MA: Kluwer; 2002.
6. Chinnery DG, Keutzer K. Closing the power gap between ASIC and custom: An ASIC perspective. Proc 42nd Design Automation Conference; 2005 Jun 13-17. p. 275–80.
7. Stratix III Programmable Power. 2001. Available from: https://www.altera.com/en_US/pdfs/literature/wp/wp-01006.pdf
8. Tuan T, Kao S, Rahman A, Das S, Trimberger S. A 90 nm low power FPGA for battery-powered applications. Proceedings 2006 ACM/SIGDA 10th International Symposium on Field-Programmable Gate Arrays; 2006. p. 3–11.
9. Zhang Z, Delgado-Frias JG. Delgado-Frias. Carbon nano tube SRAM design with metallic CNT or removed metallic CNT tolerant approaches. IEEE Transactions on Nanotechnology. 2012 Jul; 11(4):788–98.
10. George V. Low energy field-programmable gate array. Berkeley: University of California; 2000.
11. Sankaranarayanan R, Guthaus MR. A single-VDD ultra-low energy subthreshold FPGA. IEEE/IFIP 20th International Conference on VLSI and System-on-Chip, 2012 (VLSI-SoC); Santa Cruz, CA, USA. 2012 Oct 7-10. p. 219–24.
12. Grossmann PJ, Lesser ME, Onabajo M. Minimum energy analysis and experimental verification of a latch-based subthreshold FPGA. IEEE Transactions on Circuits and Systems II: Express Briefs. 2012 Dec; 59(12):942–6.
13. Bsoul AAM, Wilton SJE. An FPGA architecture supporting dynamically controlled power gating. 2010 International Conference on Field-Programmable Technology (FPT); Beijing. 2010 Dec 8-10. p. 1–8.
14. Huda S, Mallick M, Anderson JH. Clock gating architectures for FPGA power reduction. International Conference on Field Programmable Logic and Applications, FPL'09; Prague. 2009 Aug 31-Sep 2. p. 112–8.
15. Anderson JH, Najim FN. Low-power programmable FPGA routing circuitry. IEEE Transactions on Very Large Scale Integration (VLSI) Systems. 2009 Aug; 17(8):1048–60.
16. Hasan M, Kureshi AK, Arslen T. Leakage reduction in FPGA routing multiplexers. IEEE International Symposium on Circuits and Systems, ISCAS'09; Taipei. 2009 May 24-27. p. 1129–32.
17. Tran CQ, Kawaguchi H, Sakurai T. The 95% leakage reduced FPGA using zigzag power-gating, Dual-VTH/VDD and Micro VDD hopping. 2005 Asian Solid-State Circuits Conference; Hsinchu. 2005 Nov. p. 149–52.
18. Abdalla H, Pickett MD. SPICE Modeling of Memristors. IEEE International Symposium on Circuits and Systems; Rio de Janeiro. 2011 May 15-18. p.1832–5.
19. Chua LO. Memristor - The missing circuit element. IEEE Transactions on Circuit Theory. 1971 Sep; 18(5):507–19.

20. Chua LO, Kang SM. Memristive devices and systems. Proceedings of the IEEE. 1976 Feb; 64(2):209–23.
21. Strukov DB, Snider GS, Stewart DR, Williams RS.The missing Memristor found. Nature. 2008 May; 453(7191):80–3.
22. Kavehei O, Iqbal A, Kim YS, Eshraghian K, Al-Sarawi SF, Abbott D. The Fourth element: Characteristics, modelling and electromagnetic theory of the Memristor. Proceedings of Royal Society A; 2010. p. 2175–202.
23. Curd D. Power consumption in 65 nm FPGAs. 2007. Available from: http://www.xilinx.com/support/documentation/white_papers/wp246.pdf
24. Udaiyakumar R, Sankaranarayanan K. Dual Threshold Transistor Stacking (DTTS) - A novel technique for static power reduction in nano-scale CMOS circuits. Linknovate. 2012; 72(2):184–94.
25. Mohammad B, Homouz D, Elgabra H. Robust hybrid memristor - CMOS Memory: Modeling and design. IEEE Transactions on Very Large Scale Integration Systems, VLSI. 2013 Nov; 21(11):2069–79.
26. Fei W, Yu H, Zhang W, Yeo KS. Design exploration of hybrid CMOS and Memristor circuit by new modified nodal analysis. IEEE Transactions on Very Large Scale Integration Systems VLSI. 2012 Jun; 20(6):1012–25.
27. Singh J, et al. Novel Si-Tunnel FET based SRAM design for Ultralow-Power 0.3V VDD Applications. Proceedings of the 2010 Asia and South Pacific Design Automation Conference; Taipei. 2010 Jan 18-21. p. 181–6.
28. Tohidi MM, Masoumi N. FPGA leakage power reduction using CLB-clustering technique. 2010 3rd International Nanoelectronics Conference (INEC); Hong Kong. 2010 Jan 3-8. p. 637–8.
29. Hassan H, Anis M, Elmasry M. Input vector reordering for leakage power reduction in FPGAs. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. 2008 Sep; 27(9):1555–64.
30. Li F, Lin Y, He L. Field programmability of supply voltages for FPGA power reduction. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. 2007 Apr; 26(4):752–64.
31. Jaramillo-Ramirez R, Anis M. A dual-threshold FPGA routing design for subthreshold leakage reduction. IEEE International Symposium on Circuits and Systems, ISCAS'07; New Orleans, LA. 2007 May 27-30. p. 3724–7.
32. Lee KJ, Park H, Kong J, Chandrakasan AP. Demonstration of a subthreshold FPGA using monolithically integrated graphene interconnects. IEEE Transactions on Electron Devices. 2013 Jan; 60(1):383–90.