Accurate Power Analysis for Near-V_t RRAM-based FPGA

Xifan Tang, Pierre-Emmanuel Gaillardon and Giovanni De Micheli
École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
Email: xifan.tang@epfl.ch

***Abstract*—Resistive Random Access Memory (RRAM)-based FPGA architectures employ RRAMs not only as memories to store the configuration but embed them in the datapaths of programmable routing resources to propagate signals with improved performances. Sources of power consumption have been intensively studied for conventional *Static Random Access Memories* (SRAM)-based FPGAs. However, very limited works focused so far on studying the power characteristics of RRAM-based FPGAs. In this paper, we first analyze the power characteristics of RRAM-based multiplexer at circuit level and then use electrical simulations to study power consumption of RRAM-based FPGA architectures. Experimental results show that RRAM-based FPGAs achieve a *Power-Delay Product* reduced by 50% compared to SRAM-based FPGA at nominal voltage and 20% compared to near-V_t SRAM-based FPGA, respectively.**

I. Introduction

Resistive Random Access Memory (RRAM) technology recently motivated intensive research efforts in exploring novel FPGA architectures [1]–[6]. RRAMs can bring to FPGA technologies both non-volatility and performance enhancements. Most RRAM-based FPGA architectures employ RRAMs as non-volatile standalone memories [1], [2]. Thanks to their *Back-End-of-Line* (BEoL) integration capabilities, RRAMs do not occupy transistor area as SRAMs do, contributing to $7-15\%$ area shrink [2]. More advanced RRAM-based FPGA architectures employ RRAMs as high-performance programmable switches [4]–[6]. When used to propagate signals, RRAMs can reduce up to 75% resistance in datapaths compared to standard transistors, leading to a performance gain from 10% to 55% [4]. Previous works [1]–[6] address that RRAM-based FPGAs reduce $20-65\%$ power consumption. Despite the strong interest of RRAM-based FPGAs for power reduction, there is very limited work on studying precisely their power sources.

FPGA power evaluation tools [7]–[9] based on analytical models are well adapted for SRAM-based FPGAs but are not general enough for RRAM-based FPGAs. Electrical simulation is a general-purposed approach to accurately analyze the power consumption of novel FPGA architectures.

In this paper, we focus on studying the power characteristics of RRAM-based multiplexers with theoretical analysis and electrical simulations. We address the sources of the power consumption of RRAM-based multiplexers and the impact of off-resistance of RRAMs on static power consumption. We discuss the factors that suppress the static power of RRAM-based multiplexers in RRAM-based FPGA architecture. Then, we use electrical simulations to evaluate the power differences between RRAM-based and SRAM-based FPGAs at both nominal voltage and near-V_t regime. Experimental results show that RRAM-based FPGAs, operated in near-V_t, can achieve 50% power reduction without performance loss compared to a standard SRAM-based FPGA. Near-V_t RRAM-based FPGAs achieve a *Power-Delay Product* reduced by 50% compared to SRAM-based FPGA at nominal voltage and 20% compared to near-V_t SRAM-based FPGA, respectively.

This paper is organized as follows. Section II introduces the RRAM technology and the RRAM-based FPGA architecture considered in this paper. Section III discusses the power characteristics of SRAM-based and RRAM-based multiplexers. Section IV analyzes the architecture-level power consumption of SRAM-based and RRAM-based FPGAs. Section V concludes the paper.

II. Background

In this section, we give a brief introduction on RRAM technologies and RRAM-based FPGA architecture.

A. RRAM Technology

Resistive Random Access Memory (RRAM) belongs to the family of emerging *Non-Volatile Memories* (NVMs) [10]. As shown in Fig. 1(a), RRAMs are two-terminal devices and typically consist of three layers: the top electrode, the switching metal oxide and the bottom electrode. RRAMs can be programmed into two stable resistance states, a *Low Resistance State* (LRS) and a *High Resistance State* (HRS) respectively. When a programming voltage is applied between the electrodes, the metal oxide sees a conductivity change which leads to the switch between the resistance states.

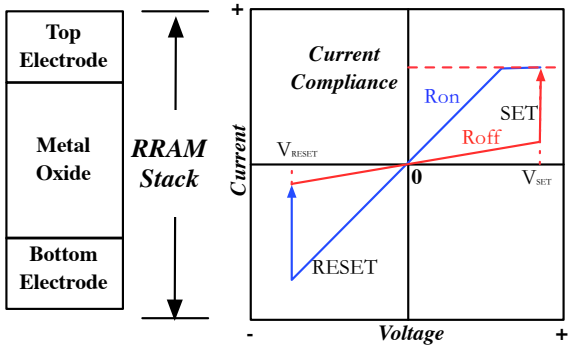

Fig. 1. (a) RRAM structure; (b) RRAM I-V characterization.

In this paper, we only focus on *Bipolar Resistive Switching* (BRS) RRAMs [11] whose I-V characteristics are illustrated in Fig. 1(b). A positive programming voltage triggers a switching event from HRS to LRS, called the **set** process. Conversely, a negative programming voltage triggers a switching event from LRS to HRS, called the **reset** process. Note that during the **set** process, a current compliance is often enforced to avoid a permanent breakdown of the device.

B. RRAM-based FPGA Architecture

Modern island-style FPGA architectures [12] [13] consist of an array of *Configurable Logic Blocks* (CLBs) surrounded by a global routing architecture. The global routing architecture is built with *Connection Blocks* (CBs), which connect CLB pins to routing tracks, and *Switch Boxes* (SBs), which interconnect routing tracks. Inside a CLB, there are a number of *Basic Logic Elements* (BLEs) each of which consists of a fracturable *Look Up Table* (LUT) [14], a D *Flip-flop* (FF), and a output selector (2:1 MUX). Additionally, there is a local routing architecture interconnects BLE pins and CLB pins.

In our vision of an RRAM-based FPGA, the scan-chain SRAMs of LUTs are replaced with a non-volatile version

[15], while the SRAMs along with transmission-gates in programmable routing architectures evolve to RRAMs with programming schemes. When programmed into LRS, the RRAMs propagate the signals through the datapath and achieve a similar functionality as transmission-gates in *on* state. The RRAMs in HRS block signals within the datapath plays the role of transmission-gates in *off* state. Compared to SRAM-based FPGA, RRAM-based FPGA obtains performance gain thanks to the RRAM-based routing elements that typically provide up to 75% less resistances than standard transistors. Additionally, the performance of RRAM-based circuits is not sensitive to the supply voltages, unlike SRAM-based counterparts, and the programming transistor sizing technique can further reduce the area, delay of RRAM-based circuits [6]. At nominal working voltage, previous works [1]–[5] predict $7\% - 15\%$ area shrinks, $10\% - 58\%$ performance gains, and $20\% - 55\%$ power reductions, compared to SRAM-based FPGAs. When operated in near-V_t regime, RRAM-based FPGA can achieve not only high performance as SRAM-based FPGA at nominal voltage but also significant power reduction as SRAM-based FPGA at near-V_t regime [6].

III. Power Analysis on Multiplexers and Discussions

In this section, we analyze the power consumption of standalone SRAM-based and RRAM-based multiplexers. Section III-A introduces the methodology, while Section III-B and III-C study the sources of switching and static power of SRAM-based and RRAM-based multiplexers, respectively.

A. Methodology

We define R_{LRS}/R_{HRS} as the on/off-resistance of RRAMs, and R_{on}/R_{off} as the on/off-resistance of a transistor, respectively. We consider a 45 nm high-performance transistor technology and use PTM models [16]. We consider flexible parameters for the RRAM technology with R_{LRS} ranging from $2k\Omega$ to $5k\Omega$ and R_{HRS} ranging from $1M\Omega$ to $1G\Omega$. These parameters are achievable by technological optimizations [11] and their effect on RRAM-based FPGAs will be studied in the following. RRAM-based multiplexers are derived from [5]. SRAM-based multiplexers are built with transmission-gates, which are more robust in near-Vt regime compared to pass-transistors [17]. SRAM-based multiplexers employ a tree-like structure for a fair comparison with the RRAM-based multiplexers. We study the 2:1 multiplexers in Fig. 2 as an example and focus on analyzing what dominates the switching and static power of RRAM-based multiplexers.

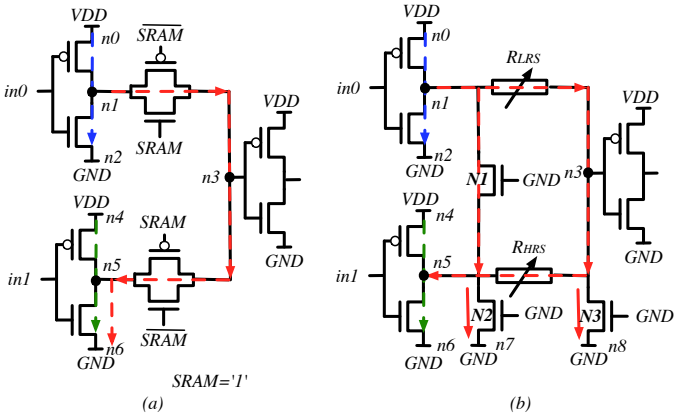

Fig. 2. 2:1 Multiplexers: (a) SRAM-based (b)RRAM-based

B. Switching Power

The switching power of multiplexers is dominated by the internal capacitances and the input activities. The differences in total internal capacitances are determined by the transmission gates in Fig. 2(a) and the programming transistors (**N1, N2, N3**) in Fig. 2(b). The obtention of a low R_{LRS}, which leads to higher routing multiplexer performances, implies the use of programming transistors typically as large as the regular transmission gates [6]. Therefore, there could be almost no differences in the total internal capacitances. Note that the size of the programming transistors and the internal capacitances may be reduced by applying the programming transistor sizing technique in [6]. We simulate one of the worst cases of the multiplexers in Fig. 2 by setting a rising edge at *in0* and a falling edge at *in1*. Table I shows the switch energy of SRAM-based, RRAM-based and RRAM-based sized multiplexers. The differences in switch energy are less than 5%.

TABLE I. Switch energy of 2:1 multiplexers

Multiplexer	Switch Power (μW)	Switch Energy (*fJ*)
SRAM-based	30.53	2.36
RRAM-based	34.50	2.47
Sized RRAM-based	31.29	2.21

C. Static Power

Static power of a multiplexer is dominated by the resistances of the sneak paths from VDD to GND. Take the example of the 2:1 multiplexers in Fig. 2, the worst case of static power happens when SRAM is configured to VDD and *in0* and *in1* are set to GND and VDD, respectively. In such case, there exist three sneak paths from VDD to GND in the SRAM-based multiplexer (Fig. 2(a)) while there are six sneaking paths in the RRAM-based multiplexer (Fig. 2(b)), as listed in Table II. Assume $R_{off} \gg R_{on}$, $R_{off} \gg R_{LRS}$, $R_{HRS} \gg R_{on}$ and $R_{HRS} \gg R_{LRS}$, the static power of the SRAM-based multiplexer is $3VDD^2/R_{off}$ while that of the RRAM-based multiplexer is $6VDD^2/R_{off} + 2VDD^2/R_{HRS}$.

TABLE II. Sneaking paths of 2:1 multiplexers

SRAM-based multiplexer in Fig. 2(a)	
Sneaking paths	Resistances on Sneaking paths
Path 1: *n0→n1→n2*	$R_{off} + R_{on}$
Path 2: *n0→n1→n3→n5→n6*	$R_{off} + 3R_{on}$
Path 3: *n4→n5→n6*	$R_{off} + R_{on}$
RRAM-based multiplexer in Fig. 2(b)	
Sneaking paths	Resistances on Sneaking paths
Path 1:*n0→n1→n2*	$R_{off} + R_{on}$
Path 2:*n0→n1→n3→n8*	$R_{on} + 2R_{LRS} + R_{off}$
Path 3:*n0→n1→n3→n5→n6*	$2R_{on} + R_{LRS} + R_{HRS}$
Path 4:*n0→n1→n3→n5→n7*	$2R_{on} + R_{LRS} + R_{HRS}$
Path 5:*n0→n1→n5→n6*	$2R_{on} + R_{off}$
Path 6:*n0→n1→n5→n7*	$2R_{on} + R_{off}$
Path 7:*n4→n5→n6*	$R_{on} + R_{off}$
Path 8:*n4→n5→n7*	$R_{on} + R_{off}$

In Table III, we analyze the sneak paths and static power of the 2:1 multiplexer for all the possible input patterns. Assuming $R_{HRS} = R_{off}$ (In the considered 45nm technology, R_{off} is $\sim 500M\Omega$.) as an ideal boundary condition, the static power of the RRAM-based multiplexer is $2\times$ its SRAM-based counterpart on average. Similar conclusions can be drawn in other sizes of multiplexers. However, in more realistic conditions where R_{HRS} is lower than R_{off}, the static power can drastically increase. Therefore, the R_{HRS} should be carefully selected to suppress the static power of RRAM-based multiplexers. Fig. 3 presents the electrical simulation results of the average static power of the RRAM-based multiplexer and its sized version by sweeping R_{HRS}. The average static power of RRAM-based multiplexers decreases sharply as R_{HRS} increases. Note that in the range between $500M\Omega$ to $1000M\Omega$, the average static power of RRAM-based multiplexer reaches the estimated bottom line, which is $2\times$ of the SRAM-based.

In the context of RRAM-based FPGAs, only the static power consumed during standard operation time should be considered. Indeed, in sleep mode, RRAM-based FPGAs can

TABLE III. Static power cases of 2:1 multiplexers

2:1 Multiplexers	SRAM-based		RRAM-based	
Case No.: (*in0,in1*)	No. of paths	Analytical static power	No. of paths	Analytical static power
1: (GND, VDD)	3	$3VDD^2/R_{off}$	8	$6VDD^2/R_{off}$ $+2VDD^2/R_{HRS}$
2: (VDD, GND)	3	$3VDD^2/R_{off}$	6	$4VDD^2/R_{off}$ $+2VDD^2/R_{HRS}$
3: (GND, GND)	2	$2VDD^2/R_{off}$	2	$2VDD^2/R_{off}$
4: (VDD, VDD)	2	$2VDD^2/R_{off}$	2	$2VDD^2/R_{off}$
Average	2.25	$2.25VDD^2/R_{off}$	4.5	$3.5VDD^2/R_{off}$ $+VDD^2/R_{HRS}$

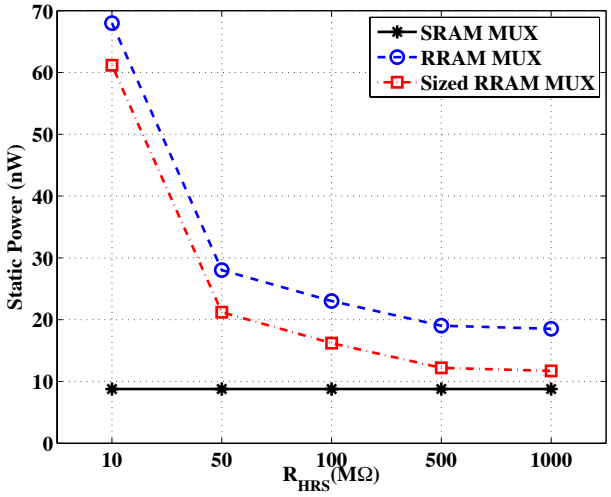

Fig. 3. Impact of R_{HRS} on the static power of the 2:1 SRAM-based and RRAM-based multiplexers

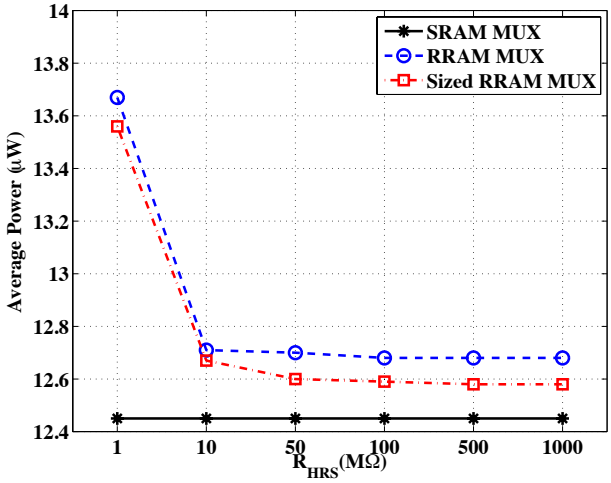

Fig. 4. Impact of R_{HRS} on the average power of the 2:1 SRAM-based and RRAM-based multiplexers with tapered buffer at output

be safely powered off and instantly turned on, reducing to zero the static power. During operation, the dynamic power of a multiplexer is much larger than the static power, limiting the effect of the increased static power contribution due to a low R_{HRS} sharply. Additionally, the use of a tapered buffer at output and a proper sizing of the programming transistors [6] reduce the RRAM contribution to static power. Fig. 4 depicts the average power, i.e. dynamic and static, of the 2:1 multiplexers in Fig. 2 while considering the contribution of the output tapered buffers and realistic output loads. In each simulation, we set the simulation time to the longest critical path of MCNC big20 benchmarks in order to account for a switching event and static periods. When the R_{HRS} is larger than $10M\Omega$, there is only 1% power differences between RRAM-based and SRAM-based 2:1 multiplexers. Therefore, the static power increase of RRAM-based multiplexer is only a marginal effect on the total power.

IV. Architectural Results and Discussions

In this section, we evaluate the power differences between SRAM-based and RRAM-based FPGAs at the architecture level. We describe our methodology in Section IV-A and analyze experimental results in Section IV-B.

A. Methodology

In this paper, we resemble the architecture of an Altera Stratix IV FPGA [18], where each CLB contains $I = 33$ inputs pins and $N = 10$ fracturable 6-input LUTs ($K = 6$). Length-4 uni-directional routing architectures are employed to interconnects Wilton's Switch Blocks, where $F_s = 3$. We set $F_{c,in} = 0.15$ and $F_{c,out} = 0.10$. Routing channel width (W) is set to 120 by adding 20% more routing tracks to the minimum channel width required for the biggest tested benchmark. Fig. 5 illustrates the modified VTR flow [19] for our power analysis method. Logic synthesis tool ABC [20] optimizes the MCNC big20 benchmarks [21] and we use VPR to pack, place and route. Afterwards, a netlist generation engine outputs the SPICE netlists of LUTs, FFs, and MUXes according to the routing results and then we run SPICE simulators to analyze power consumption. We first analyze the area, delay, power and power-delay product of the architectures discussed in [6], namely: (a) SRAM-based FPGA at nominal VDD; (b) SRAM-based FPGA at near-V_t regime; (c) unsized RRAM-based FPGA at near-V_t regime; (d) sized RRAM-based FPGA at near-V_t regime (Consider $R_{HRS} = 100M\Omega$ for RRAM-based FPGAs.) The nominal VDD of the PTM 45nm technology is 1.0V. We consider near-V_t VDD=0.85V, the critical point where near-V_t RRAM-based FPGAs have no delay degradation compared to SRAM-based at nominal VDD. Then we sweep the R_{HRS} and study its impact on RRAM-based FPGA power consumption.

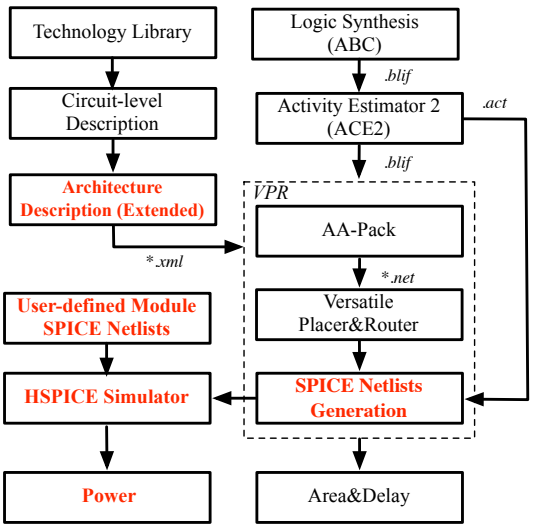

Fig. 5. Modified VTR flow for power estimation.

B. Experimental Results and Discussions

Fig. 6 illustrates the normalized area, delay, power and *Power-Delay Product* (PDP) of the four FPGA architectures. On average, the unsized RRAM-based FPGA uses 14% less area than the SRAM-based FPGAs, while thanks to the sizing

technique, a sized RRAM-based FPGA reduces 27% area compared to SRAM-based FPGAs. On average, the near-V_t SRAM-based FPGA degrades by 36% more delay than the SRAM-based FPGA at nominal VDD, while the performances of RRAM-based FPGAs, which benefit from high-performance RRAM multiplexers, are still competitive with regards to a SRAM-based architecture operated at nominal voltage. Unsized RRAM-based FPGA degrades the delay metric by only 3% on average while the properly sized one sees a 6% delay gain on average, compared to SRAM-based FPGA at nominal voltage. Near-V_t FPGAs reduce by ∼50% the power consumption compared to the SRAM-based FPGA at nominal VDD. With respect to a SRAM FPGA running at near-V_t, a unsized RRAM-based FPGA consumes 6.5% more power due to its larger static power. However, a properly sized RRAM-based FPGA consumes the same power as SRAM-based at near-Vt regime on average, as discussed in Section III-C. In terms of PDP, a unsized RRAM-based FPGA is 10% better than their SRAM-based counterparts, directly resulting from the increased performances and reduced power consumption. The sized RRAM-based FPGA further improves PDP by 10%, as the best among the four architectures. Note that the SRAM-based FPGA yields a better PDP when operated in the near-V_t regime rather than at nominal voltage because of the following reason: At circuit-level, the near-V_t LUTs, FFs and MUXes degrade over 50% compared to nominal voltage, but at architecture-level, the overall delay is optimized by VPR and the degradation is only 36% on average.

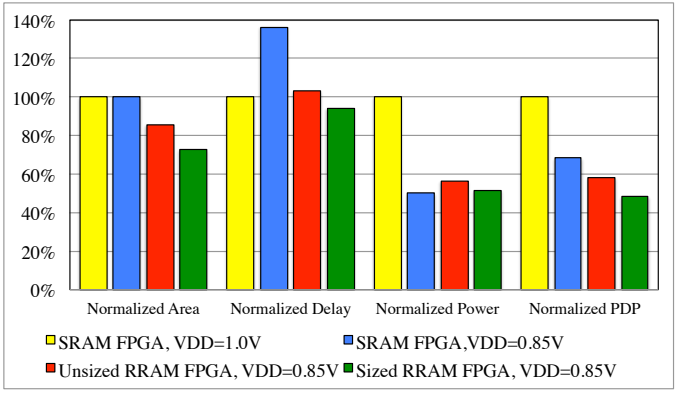

Fig. 6. Normalized area, delay, power and PDP for the considered FPGA architectures on average of MCNC big20 benchmarks.

C. Impact of R_{HRS} on Power Consumption

As explained in Section III-C, the R_{HRS} can influence the power consumption of RRAM-based routing elements. We evaluate in Fig. 7 the impact of R_{HRS} on the average power of the considered FPGA architectures implementing in MCNC big20 benchmarks. Basically, the power consumption of RRAM-based FPGA increases as R_{HRS} decreases. The power differences between RRAM-based and near-V_t SRAM-based FPGAs is within 10% when R_{HRS} is larger than $50M\Omega$. From a architecture prospective, RRAM-based FPGAs require a larger unbound on R_{HRS} than the RRAM-based multiplexers due to the fact that part of the multiplexers in FPGAs are idle at runtime, increasing the share of the static power.

V. Conclusion

In this paper, we analyze the power characteristics of the RRAM-based multiplexers and discuss their differences with respect to their SRAM-based counterparts. Theoretical analysis and electrical simulations show that the static power of RRAM-based multiplexers is sensitive to the off-resistance of the RRAMs and can be reduced by a proper sizing of

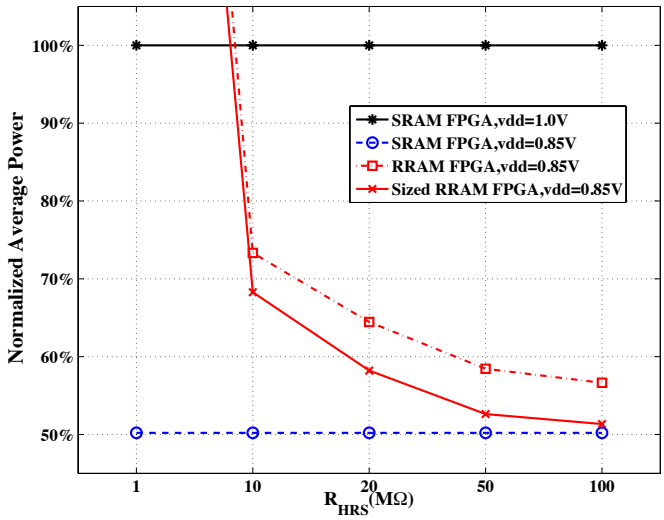

Fig. 7. Normalized power consumption of RRAM-based architecture with different R_{HRS}

the RRAM programming transistors. Architecture-level power analysis shows that near-Vt RRAM-based FPGAs produce 20% better *Power-Delay Product* (PDP) than near-V_t SRAM-based FPGA, and 50% better PDP compared to nominal SRAM-based FPGA.

Acknowledgment

This work was supported by the Swiss National Science Foundation under the project number 200021-146600.

References

[1] Y. Chen *et al.*, *Non-volatile 3D Stacking RRAM-based FPGA*, FPL, pp. 367-372, 2012.
[2] P.-E. Gaillardon *et al.*, *Phase-Change-Memory-based Storage Elements for Configurable Logic*, ICFPT, pp. 17-20, 2010.
[3] S. Tanachutiwat *et al.*, *FPGA Based on Integration of CMOS and RRAM*, TVLSI, Vol. 19, No. 11, pp. 2023-2032, 2010.
[4] J. Cong *et al.*, *FPGA-RPI: A Novel FPGA Architecture With RRAM-Based Programmable Interconnects*, TVLSI, Vol. 22, No. 4, pp. 864-877, 2014.
[5] P.-E. Gaillardon *et al.*, *GMS: Generic Memristive Structure for Non-Volatile FPGAs*, IEEE/IFIP VLSI-SoC, pp. 94-98, 2012.
[6] X. Tang *et al.*, *A High-performance Low-power Near-Vt RRAM-based FPGA*, ICFPT, pp. 207-215, 2014.
[7] K. K. Poon *et al.*, *A Detailed Power Model for Field-Programmable Gate Arrays*, TODAES, Vol. 10, No. 2, pp. 279-302, 2005.
[8] J. B. Goeders *et al.*, *VersaPower: Power Estimation for Diverse FPGA Architectures*, ICFPT, pp. 229 - 234, 2012.
[9] F. Li *et al.*, *Power Modeling and Characteristics of Field Programmable Gate Arrays*, TCAD, Vol. 24, No. 11, pp. 1712-1724, 2005.
[10] G.W. Burr *et al.*, *Overview of Candidate Device Technologies for Storage-Class-Memory*, IBM J. R&D, Vol. 52, No. 4/5, July/Sept. 2008.
[11] H.-S. P. Wong *et al.*, *Metal-Oxide RRAM*, Proceedings of the IEEE, Vol. 100, No. 6, 2012, pp. 1951-1970.
[12] Altera Corporation, *Stratix 10 Advance Information Brief*, July 2015.
[13] Xilinx, *Virtex-7 User Guide DS180 (v1.17)*, May 2015.
[14] M. Hutton *et al.*, *Improving FPGA Performance and Area Using an Adaptive Logic Module*, FPL, 2004, pp. 135-144.
[15] I. Kazi *et al.*, *Energy/Reliability Trade-Offs in Low-Voltage ReRAM-Based Non-Volatile Flip-Flop Design*, TCAS-I, Vol. 61, No. 11, pp. 3155-3164, 2014.
[16] Available online http://ptm.asu.edu/.
[17] C. Chiasson *et al.*, *Should FPGAs Abandon the Pass-gate?*, FPL, 2013, pp. 1-8.
[18] Altera Corporation, *Stratix IV Device Handbook*, July 2008.
[19] J. Rose *et al.*, *The VTR Project: Architecture and CAD for FPGAs from Verilog to Routing*, FPGA, 2012, pp. 77-86.
[20] UC Berkeley, *ABC: A System for Sequential Synthesis and Verification*, Available online. http://www.eecs.berkeley.edu/~alanmi/abc/
[21] S. Yang, *Logic Synthesis and Optimization Benchmarks User Guide Version 3.0*, MCNC, Jan. 1991.