

# A Power-Efficient Reconfigurable Architecture Using PCM Configuration Technology

Ali Ahari<sup>1</sup>, Hossein Asadi<sup>1</sup>, Behnam Khaleghi<sup>1</sup>, and Mehdi B. Tahoori<sup>2</sup>

<sup>1</sup> *Department of Computer Engineering, Sharif University of Technology, Tehran*

<sup>2</sup> *Chair of Dependable Nano Computing, Karlsruhe Institute of Technology, Karlsruhe*

**Abstract**—Promising advantages offered by resistive *Non-Volatile Memories* (NVMs) have brought great attention to replace existing volatile memory technologies. While NVMs were primarily studied to be used in the memory hierarchy, they can also provide benefits in *Field-Programmable Gate Arrays* (FPGAs). One major limitation of employing NVMs in FPGAs is significant power and area overheads imposed by the *Peripheral Circuitry* (PC) of NVM configuration bits. In this paper, we investigate the applicability of different NVM technologies for configuration bits of FPGAs and propose a power-efficient reconfigurable architecture based on *Phase Change Memory* (PCM). The proposed PCM-based architecture has been evaluated using different technology nodes and it is compared to the SRAM-based FPGA architecture. Power and *Power Delay Product* (PDP) estimations of the proposed architecture show up to 37.7% and 35.7% improvements over SRAM-based FPGAs, respectively, with less than 3.2% performance overhead.

## I. INTRODUCTION

In the past decade, *Field-Programmable Gate Arrays* (FPGAs) have gained popularity in wide range of applications due to fast time-to-market, possible design update, and fast reconfiguration [1], [2]. A commonly used FPGA is based on *Static Random Access Memory* (SRAM) where the chip configuration state is stored in SRAM cells known as configuration bits. At nanoscale, SRAM technology faces some important challenges such as high leakage power, susceptibility to particle strikes, and low bit density. State-of-the-art *SRAM-based FPGAs* (SFPGAs) consist of more than  $10^8$  SRAM bits which makes them heavily influenced by negative effects of SRAM technology. For example, with 1000 *Failure-In-Time* (FIT) per Mbit for SRAM technology [1], the largest SFPGAs will suffer from more than  $10^5$  FIT per device, where one FIT equals to one failure in a billion hours of device operation. Recent studies demonstrate that high leakage power and low density of SRAM cells have resulted in significant area, performance, and power gap between *Application-Specific Integrated Circuits* (ASICs) and SFPGAs [2].

Recently, numerous efforts have been made to reduce the gap between SFPGAs and ASICs. One promising solution is using emerging *Non-Volatile Memories* (NVMs) as a candidate alternative of SRAM cells in SFPGAs [3]–[7]. Emerging NVMs such as Flash, *Spin-Transfer Torque Magnetic Random Access Memory* (STT-MRAM), and *Phase-Change Memory* (PCM) offer quite higher density, lower leakage power, and more immunity to particle strikes than SRAM counterparts. In addition, as opposed to SFPGAs, NVM-based FPGAs do not require an additional on-chip or off-chip non-volatile storage such as EEPROM or Flash memory to store configuration bits. This reduces the chip area, removes the boot-up time, and alleviates design complexity at the chip/board level.

Despite appealing features of NVMs, there are several limitations to employ NVMs in FPGAs. Unlike SRAMs, NVMs may impose high write-power consumption and high write-latency during system reconfiguration. In addition, poor compatibility with CMOS fabrication process and high overhead of NVMs peripherals can further limit their usage in FPGAs. Lastly, due to limited endurance of emerging NVMs, NVM-based FPGAs will suffer from limited number of reconfigurations.

Previous studies have examined using various NVMs as candidate replacements for SRAM cells in SFPGAs. The NVMs in these studies can be classified based on the following technologies: 1) antifuse [8], 2) Flash [7], 3) PCM [9], [6] 4) STT-MRAM [4], [5] and 5) memresistor [10] and *Resistive RAM* (RRAM) [11]. Antifuse-based FPGAs are one-time programmable and do not support reconfiguration which makes them out of scope in reconfiguration systems. On the other hand, more recent emerging NVM technologies such as RRAM and memresistor are not mature enough to be considered as a replacement for SRAM cells in the near future and are not discussed in this work. According to several industrial reports [12], [13], promising mature NVM technologies are Flash, PCM, and STT-MRAM and in the rest of this paper, by NVM we mean these memory technologies.

Previous NVM-based FPGA architectures have been proposed to replace conventional SRAM configuration bits as well as *Flip-Flops* (FFs) and *Block RAMs* (BRAMs) with NVM cells. Despite significant improvements in area and power consumption achieved by these NVM-based FPGAs [3], [4], [5], the specialized *Peripheral Circuitry* (PC) still remains a bottleneck for power and area.

In this paper, we investigate the applicability of various NVM technologies in FPGAs. The proposed study reveals that although STT-MRAM is a more promising alternative than PCM and Flash in regular structures such as cache and main memory [14], their usage in FPGAs is more challenging due to power inefficiency of STT-MRAM PCs in irregular structures such as configuration bits. To the best of our knowledge, this is the first work addressing the power limitation of PCs in NVM-based FPGAs. Based on this study, a power-efficient PCM-based reconfigurable architecture is proposed. To address high power consumption in NVM PCs which could highly affect the overall power consumption, a power-efficient PC to convert PCM state to the equivalent voltage levels is proposed, keeping consistency of PCM configuration bits with the other parts.

The proposed architecture has been evaluated using different technology nodes and it is compared with previous NVM-based and SRAM-based FPGA architectures. The results demonstrate that as compared to SFPGAs, the proposed archi-

ture can improve power and *Power Delay Product* (PDP) up to 37.7% and 35.7%, respectively with negligible (3.2%) performance overhead.

The rest of this paper is organized as follow. Sec. II reviews previous NVM-based FPGAs and discusses the shortcomings of the previous work. Sec. III presents our study on applicability of NVMs in FPGAs and then presents the proposed architecture. Sec. IV reports the experimental results. Finally, Sec. V concludes the paper and presents the future work.

## II. PREVIOUS WORKS

In nanoscale technology nodes, the susceptibility to particle strike and high leakage power emerge as major concerns for conventional SRAM-based FPGAs [1], [15]. On the other hand, immunity of NVM cells against particle strikes, high density, and low static power consumption are the attractive features to investigate NVM-based FPGA architectures. In this section, we discuss previously proposed FPGA architectures employing Flash, STT-MRAM, and PCM technologies.

### A. Flash-based FPGAs

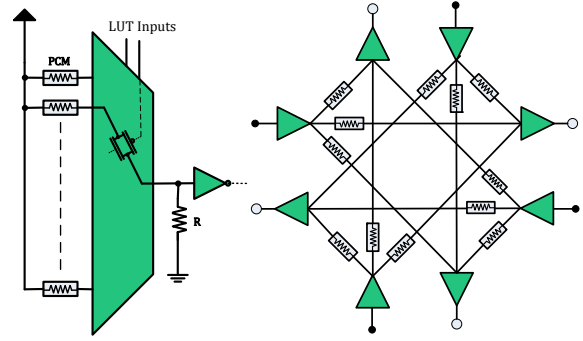
Flash technology is one of the most mature NVM technologies which is widely used in the industrial applications. Immunity of Flash cells against particle strikes and their lower leakage power as compared to SRAM cells have motivated researchers to propose several Flash-based FPGAs [7], [16]. There are also several Flash-based commercial FPGAs such as Actel's ProAsic series [17] and Lattice XP2 family [18] which employ flash memories to store configuration bits.

Despite the reliability advantages of Flash-based FPGAs, they suffer from several limitations. Since flash technology has a low endurance as compared to other NVMs such as PCM and STT-MRAM, the lifetime of Flash-based FPGAs is very limited as compared to other NVM-based FPGAs. Implementing write-intensive memory elements such as flip-flops in [7] further limits the lifetime of proposed Flash-based FPGAs. Furthermore, in-place update is not supported in flash memories and each write operation should be proceeded by an erase operation. Due to the limitation of block-level erase in flash technology, fine-grained bit-level reconfiguration is not supported in Flash-based FPGAs. Finally, integrating CMOS with Flash memories demands higher cost than integrating other NVMs such as PCM and STT-MRAM with CMOS technology [3].

### B. STT-MRAM-based FPGAs

STT-MRAM is a resistive memory and represents stored value by its resistance levels. It also offers one of the best scalability characteristics among the MRAM technologies [19]. The main difficulty of using resistive memories in configuration memories is the required PC to convert the cell resistance to the equivalent voltage level.

There are several proposed STT-MRAM-based FPGAs that have tried to utilize STT-MRAM cells in the configuration memories of FPGAs [4], [5], [20]. Zhao et. al. proposed an STT-MRAM-based FPGA with specialized PCs [4], [20]. The proposed architecture, however, suffers from high power consumption due to static leakage current. It also imposes significant area overhead. The proposed PC in [4] is implemented by dynamic logic. Although a permanent evaluation of configuration bit states is required in FPGAs, the evaluation



(a) Proposed LUT by [3]. (b) Proposed SB by [3].  
Fig. 1. PCs for LUTs and SBs [3]

process is not permitted in a dynamic logic during the pre-charge phase. This makes the proposed PC in [4] inapplicable in FPGAs.

Paul et. al. proposed another STT-MRAM-based FPGA in [5] which employs STT-MRAM cells only in the configuration memories of *Look-Up Tables* (LUTs). The improvement offered by [5] is very limited as LUT configuration bits contribute to less than 10% of the total number of configuration bits in FPGAs. The proposed PC in [5] is basically a voltage divider circuit which suffers from high leakage power. Furthermore, weak logic states in the buffers of the proposed PC can lead to high short-circuit power consumption.

### C. PCM-based FPGAs

PCM is one of the mature NVMs which is reached to high volume production in the recent years [12]. There have been several proposed FPGA architectures such as [6], [21], and [3] which employ PCM in the configuration bits. Similar to STT-MRAM, PCM is a resistive memory and requires a specialized PC in the configuration memory of FPGAs. However, except the proposed PC by Gaillardon et. al. [3], design issues of PC have been overlooked in the previous PCM-based FPGA architectures.

Gaillardon et. al. have proposed two PCs to be used in *Switch Boxes* (SBs) and LUTs of PCM-based FPGAs, as shown in Fig. 1. In this proposed LUT, according to the inputs of the multiplexer, the corresponding PCM cell connects through a transmission gate to the output of the multiplexer. At the output of the multiplexer, a resistor is used to convert the PCM state to the voltage level and furthermore, an inverter is employed to produce the required voltage levels. The main advantage is that always only one PCM cell is active, hence, other PCM cells which are not selected by the multiplexer are inactive and do not consume any power.

The proposed PC for LUTs in [3], however, suffers from limited delay and power efficiency. The delay is directly proportional to the amount of resistance in the critical path of the circuit. In order to reduce the delay of this circuit, one should reduce the amount of the resistance which is the sum of resistances of the PCM cell, the transmission gate, and the resistor at the multiplexer output. On the other hand, the power consumption of the circuit is inversely proportional to the cell resistance. Hence, with decreasing the resistance value, the power consumption will increase. Consequently, decreasing the critical path delay will lead to an increased power consumption and vice versa.

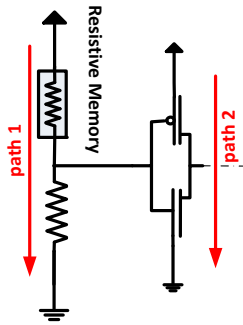


Fig. 2. Resistive memory PC model

In addition, the proposed PC by [3] for SBs, demonstrated in Fig. 1(b), suffers from high leakage current. Unlike the conventional SBs, PCM cells in this SB are used directly instead of pass transistors to connect or disconnect a path between two terminals of the SB. The authors have proposed to configure PCM cells to a low resistance state, when a connection is established between two terminals and in other cases, the PCM cells are configured to a high resistance state. Using this scheme eliminates the need for the pass transistors in the SBs. However, in case a voltage difference exists between two terminals of SB and the PCM between them is configured to a high resistance state, a significant leakage current passes through the PCM cell.

### III. PROPOSED ARCHITECTURE

In order to propose a power-efficient FPGA architecture based on NVM technologies, designers encounter several important design decisions. In particular, the power efficiency of NVM technology used as the configuration bits, the power consumption of the NVM PCs, and the integration of the NVM PCs with the FPGA resources are of decisive design importance. In this section, in order to select the effective NVM technology for configuration bits of the proposed FPGA architecture, the applicability of different NVM technologies in FPGAs is studied. Then, to address the high power consumption of PCs, a power-efficient technique is proposed for configuration bits of FPGA. Finally, a PCM-based architecture which employs the proposed PC is introduced.

#### A. Applicability of NVMs in FPGAs

To investigate the applicability of various NVMs in FPGAs with respect to power consumption, we focus on three most mature NVM technologies: Flash, STT-MRAM, and PCM. Although PCM and STT-MRAM do not have the maturity of Flash technology, they offer more promising power, performance, and endurance characteristics [22]. As discussed in Sec. II-A, in addition to the limited number of reliable reconfigurations in the Flash-based FPGAs, partial reconfiguration is not fully supported due to erase-before-update limitation of Flash technology. Furthermore, the integration of CMOS and Flash technology has higher costs than integrating CMOS with PCM or STT-MRAM technology [3]. Hence, in this work we focus on two emerging non-volatile technologies (i.e., STT-MRAM and PCM) to be used in the configuration memory of FPGAs.

PCM and STT-MRAM are both resistive memories and represent the stored value by their resistance. The main difficulty of using resistive memories in the configuration memory of FPGAs is the required PC to convert the cell resistance to

the equivalent voltage level. In regular array memory structures such as cache and main memory, the PC usually employs a sense amplifier which could be shared by multiple cells. In contrary, configuration bits in FPGAs are scattered throughout the FPGA die and have a non-array structure. Additionally, a permanent read from all configuration bits is required after FPGA power-up. Therefore, the PC overhead could not be shared among multiple cells in FPGAs and each configuration bit requires a dedicated PC. Consequently, using a PC structure same as the PCs used in regular array memory structures could impose significant energy and area overheads in FPGAs. As a result, several specialized PCs are proposed for PCM- and STT-MRAM-based FPGAs such as [3]–[5], [20], [23].

All of the previously proposed PCs for PCM- and STT-MRAM-based FPGAs can be modeled by a simple voltage divider circuit as demonstrated in Fig. 2. In the rest of this section, this model is used to investigate the applicability of PCM and STT-MRAM technologies in FPGAs.

1) *Applicability of STT-MRAMs in FPGAs:* Using the voltage divider scheme in the proposed PCs for STT-MRAM-based FPGAs comes with the following shortcomings. First, STT-MRAM cells do not have high resistance values even in their high resistance state ( $R_H$ ) [22]. Consequently, using them in a voltage divider circuit can lead to significant leakage current in path 1 (Fig. 2). Second, the required buffer or inverter circuit at the output imposes significant short-circuit power consumption through path 2 (Fig. 2). The ratio of high resistance state to the low resistance state ( $R_H/R_L$ ) is typically low in STT-MRAM technology (less than 4 in the room temperature [24]). Consequently, a buffer or inverter is required at the output of the voltage divider in order to provide appropriate voltage levels for the next stages. Weak logic states in the input makes the pull-up and pull-down transistors of the buffer/inverter weakly *on* or *off* all the time. This will impose short-circuit power overheads. Table I shows the distribution of static power in the proposed PC by [5] which is modeled by the voltage divider circuit described in Fig. 2 and it is compared to a single SRAM cell and an inverter. It is shown that the static power consumptions through path 1 and path 2 are orders of magnitude more than the total static power consumption of a single SRAM cell.

TABLE I. STATIC POWER CONSUMPTION AND SIMULATION PARAMETERS

Circuit	Static Power (Watt)			Simulation Parameters
	Path 1	Path 2	Total	
Inverter	-	-	1.0E-08	W/L: 3/2, V <sub>ss</sub> :1V, 45nm [31]
SRAM	-	-	7.2E-08	W/L: same as [25], V <sub>ss</sub> :1V, 45nm
PC [5]	5.6E-05	2.1E-06	5.8E-05	$R_H:6K\Omega$ , $R_L:2K\Omega$ , V <sub>ss</sub> :1V

Another shortcoming of STT-MRAM technology which limits its application in FPGAs is *Thermal Activation* phenomena. Thermal activation occurs in STT-MRAM cells when a read current which is far less than a write current is applied for a certain amount of time (which could be even less than 1000ns [26]). In this case, the probability of a change in the state of STT-MRAM cell will be very high. This makes STT-MRAM an inappropriate technology for configuration memories of FPGAs where a permanent read current to read the state of configuration memories is required (configuration bits are mostly read and seldom written).

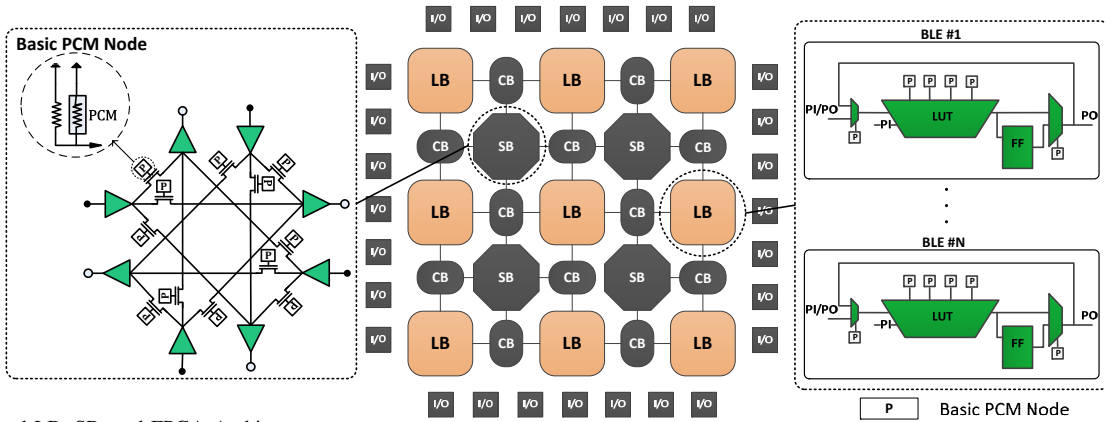


Fig. 3. Proposed LB, SB, and FPGA Architecture

2) *Applicability of PCM in FPGAs*: Unlike STT-MRAM,  $R_H/R_L$  in the PCM technology could be as high as  $10^4$  [27]. This eliminates the need for an inverter or a buffer at the output of the voltage divider and as a result, reduces the short-circuit power. Furthermore, PCM offers high  $R_H$  (more than  $1000M\Omega$  [28]) which could further reduce the leakage power compared to STT-MRAM. In addition, no switching occurs in the state of the PCM cells at the currents lower than the write current. Unlike STT-MRAM, the direction of the applied current is also not important in the write operations for PCM cells. This simplifies designing PC for PCM cells.

These advantageous features of PCM make it an effective NVM technology to be used in the configuration memory of FPGAs. Since *Multi-Level Cells* (MLCs) in PCMs suffer from resistance drift phenomena [29], in the proposed architecture we use *Single-Level Cells* (SLCs). Resistance drift is the increase in the resistance of PCM cells over time and is attributed to structural relaxation phenomena [29]. Since high resistance states have more resistance drift than low resistance states, it leads to reliability issues in MLCs after a while.

### B. PCs and Architecture

Fig. 3 demonstrates an overview of the proposed FPGA architecture. Same as the conventional SFPGAs, the proposed architecture consists of an array of *Logic Blocks* (LBs) that are connected to each other through programmable SBs and CBs. Each LB also consists of several *Basic Logic Elements* (BLEs). As demonstrated in Fig. 3, the proposed BLE consists of an LUT, configurable multiplexers, and a FF.

In the proposed SB and LUT structures which are similar to those in the conventional SFPGAs, the SRAM cells in the conventional structures are replaced with the basic PCM node presented in Fig. 3. Despite static leakage current in each basic PCM node, the simulation results demonstrate that the overall leakage power is less than that of the SRAM-based structures. The proposed basic PCM node consists of a resistor connected to a PCM cell. This PCM node is, in fact, a simple voltage divider circuit. This circuit is used to convert the PCM state to the equivalent voltage level. The resistor value is opted to reduce the leakage current and also to provide the appropriate voltage level at the output. Since the PCM technology offers high  $R_H/R_L$  values, there is no further need for a buffer or inverter at the output of the voltage divider. This further reduces the static power by avoiding the short-circuit leakage in the buffer/inverter.

Unlike the LUT proposed by [3] which employs transmission gates, the proposed LUT is implemented by the conventional CMOS technology. Consequently, each PCM cell is connected to the gate terminal of the transistor instead of the source and drain terminals. The gate to drain/source resistance is much more than the drain to source resistance in the CMOS technology [30]. As a result, the leakage power will be further reduced in the proposed LUT as compared to the LUT proposed by [3].

In addition, the proposed SB provides a connection between two terminals by a transmission gate which could be configured to be on or off. The resistance of the transmission gate in the off state is considerably higher than the high resistance state of PCM technology used in [3]. Therefore, the leakage power of the proposed SB will be less than the leakage power of the SB proposed by [3]. In order to perform a write operation on PCM cells, the same circuitry as presented in [3] could be used. This circuitry almost has the same overhead as the circuit required for programming SRAM cells in SFPGAs.

## IV. EXPERIMENTAL SETUP AND RESULTS

In this section, the functionality of the proposed PC is verified by simulating a simple 7-bit parity generator circuit implemented by the proposed architecture. Then, in order to evaluate the proposed PCM-based FPGA, we will compare our proposed architecture with the conventional SRAM-based FPGA and the PCM-based FPGA proposed by Gaillardon et. al. [3]. In the experiments, it is assumed that each LB consists of four BLEs and a 6-input LUT is used in all BLEs. The proposed architecture and the baseline architectures are synthesized with Design Compiler first. Then, the SPICE netlist is extracted from Design Compiler and imported to the Hspice simulation.

In our experiments, the technology trend is explored for the proposed architecture as well as the previously proposed architectures. To this end, technology libraries of 130, 90, and 45nm are obtained from [31]. Average delay, power, and *Power Delay Product* (PDP) are then extracted from Hspice reports for 20 largest MCNC benchmark circuits. We use the PCM cells proposed in [28] which offer  $1000M\Omega$  and  $0.5M\Omega$  resistance levels in high and low resistance states, respectively. In addition,  $40M\Omega$  is chosen as the resistance level in the proposed basic PCM node. This resistance level could be provided by a PCM cell proposed in [32] which is configured to a high resistance state at the fabrication time.



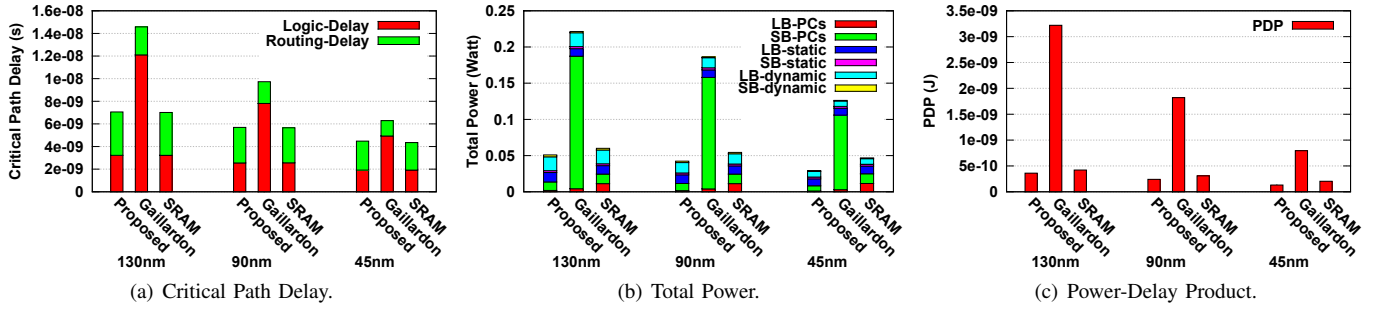


Fig. 5. Proposed Architecture vs. Baseline Architectures

### A. Functional Verification

Fig. 4 demonstrates the implementation of a 7-bit even parity generator circuit using the proposed architecture. In this implementation, two LUTs are connected through a SB to verify the functionality of both the proposed LUT and SB. The configuration bits implemented by the proposed basic PCM node are configured to logical 0 and 1 states as shown in Fig. 4. The simulation of the circuit in Hspice demonstrates the correct functionality of the implemented circuit in the proposed architecture. Hspice simulation results are illustrated in Fig. 4 for an 8ns period of circuit operation.

### B. Critical Path Delay

Fig. 5(a) demonstrates the critical path delay of the proposed architecture as compared to the baseline architectures. On average, the proposed architecture improves performance by 51.5%, 41.6%, and 28.8% in 130, 90, and 45nm technologies, respectively, as compared to the proposed architecture by Gaillardon. The proposed architecture has also a comparable performance as SRAM-based architecture. The proposed ar-

chitecture, however, imposes less than 0.7%, 0.5%, and 3.2% performance overhead in 130, 90, and 45nm technologies, respectively, as compared to the SRAM-based architecture. Since the basic structure of the proposed structures is the same as the SRAM-based counterparts, similar system performance can be achieved. The negligible difference is caused by different performance characteristics between the proposed basic PCM node and an SRAM cell.

### C. Power

Fig. 5(b) illustrates the total power in the three architectures which is the sum of the power consumptions of the PCs used in LBs and SBs, the static power, and the dynamic power of LBs and SBs. As shown in Fig. 5(b), the static power consumption is directly influenced by the architecture of FPGA, however, the dynamic power consumption mainly depends on the design implemented on the FPGA. Therefore, the proposed architecture and the baselines almost have the same dynamic power consumption and the main difference is in the static power consumption. Furthermore, the results demonstrate the considerable impact of the power consumption of PCs on the total power consumption. The proposed architecture reduces the total power consumption by 15.2%, 22.2%, and 37.7% in 130, 90, and 45nm technologies, respectively, as compared to the SRAM-based FPGA. This means that the benefit of the proposed architecture is pronounced with technology scaling. In addition, the total power consumption is reduced in the proposed architecture by 77.0%, 77.3%, and 76.8% in 130, 90, and 45nm technologies, respectively, as compared to the proposed FPGA by Gaillardon.

As shown in Fig. 5(b), using PCM technology by itself does not guarantee the power efficiency and even can impose significant power consumption overheads. This reveals the important role of power efficiency of PCs in NVM-based FPGAs. Furthermore, the results indicate the potential of the proposed architecture to reduce the power consumption as the technology size decreases.

### D. Power-Delay Product

Fig. 5(c) shows the PDP of the proposed architecture as compared to the baselines. Since the proposed architecture almost has the same performance as the SRAM-based architecture, the significant power reduction leads to a significant PDP reduction up to 14.6%, 21.8%, and 35.7% in 130, 90, and 45nm technologies, respectively. In addition, since the proposed architecture in [3] suffers from high power consumption, its PDP is 8.9X, 7.5X, and 6X more than the proposed architecture in 130, 90, and 45nm technologies, respectively.

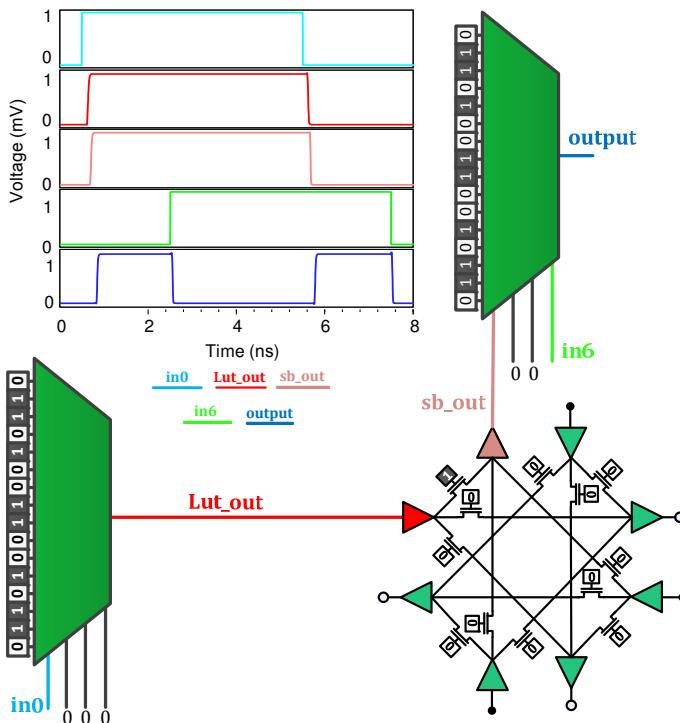


Fig. 4. Functional verification scenario and results

Prominent reduction in PDP of the proposed architecture as compared to the baselines shows the potential of the proposed architecture to reduce the overall energy consumption. This could be a motivation for the next generation of energy efficient FPGA architectures. Furthermore, the proposed architecture could fill the power consumption gap between ASICs and FPGAs and provide new opportunities for the FPGA market.

## V. CONCLUSION AND FUTURE WORK

In this paper, we proposed a FPGA architecture taking the advantage of PCM technology. To address the power issue of PCs in PCM cells, we proposed a power-efficient PC for SBs and LUTs. The results showed that the proposed architecture can improve power consumption and PDP up to 37.7% and 35.7%, respectively, with minimal performance overhead. Since PCM cells have a limited endurance, as a future work, a heterogeneous FPGA architecture taking the advantage of both PCM and SRAM technologies will be studied. To this end, we will examine the bit-change probability of different FPGA resources such as LUTs and SBs to enhance the number of reconfigurations over device lifetime by using PCM cells only in less write intensive configuration memories.

## REFERENCES

- [1] H. Asadi and M. B. Tahoori, "Analytical techniques for soft error rate modeling and mitigation of fpga-based designs," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, vol. 15, no. 12, pp. 1320–1331, 2007.
- [2] I. Kuon and J. Rose, "Measuring the gap between fpgas and asics," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 26, no. 2, pp. 203–215, 2007.
- [3] P. Gaillardon, D. Sacchetto, H. Ben Jamaa, G. Betti Beneventi, L. Perniola, F. Clermidy, I. O'Connor, and G. De Micheli, "Design and architectural assessment of 3-d resistive memory technologies in fpgas," *Nanotechnology, IEEE Transactions on*, vol. 12, no. 1, pp. 40–50, 2013.
- [4] W. Zhao, E. Belhaire, C. Chappert, and P. Mazoyer, "Spin transfer torque (stt)-mram-based runtime reconfiguration fpga circuit," *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 9, no. 2, p. 14, 2009.
- [5] S. Paul, S. Mukhopadhyay, and S. Bhunia, "A circuit and architecture codesign approach for a hybrid cmos-sttram nonvolatile fpga," *Nanotechnology, IEEE Transactions on*, vol. 10, no. 3, pp. 385–394, 2011.
- [6] Y. Chen, J. Zhao, and Y. Xie, "3d-nonfar: three-dimensional non-volatile fpga architecture using phase change memory," in *Proceedings of the 16th ACM/IEEE international symposium on Low power electronics and design*. ACM, 2010, pp. 55–60.
- [7] D. Choi, K. Choi, and J. Villasenor, "New non-volatile memory structures for fpga architectures," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, vol. 16, no. 7, pp. 874–881, 2008.
- [8] J.-J. Wang, B. Cronquist, B. Sin, J. Moriarta, and R. Katz, "Antifuse fpga for space applications," in *RADECS*, vol. 97, 1997, p. 11.
- [9] K. Chen, L. Krusin-Elbaum, D. Newns, B. Elmegreen, R. Cheek, N. Rana, A. Young, S. Koester, and C. Lam, "Programmable via using indirectly heated phase-change switch for reconfigurable logic applications," *Electron Device Letters, IEEE*, vol. 29, no. 1, pp. 131–133, 2008.
- [10] J. Cong and B. Xiao, "mrfpga: A novel fpga architecture with memristor-based reconfiguration," in *Nanoscale Architectures (NANOARCH), 2011 IEEE/ACM International Symposium on*. IEEE, 2011, pp. 1–8.
- [11] —, "Fpga-rr: an enhanced fpga architecture with rram-based reconfigurable interconnects," in *Proceedings of the ACM/SIGDA international symposium on Field Programmable Gate Arrays*. ACM, 2012, pp. 268–268.
- [12] H. P. Wong, S. Raoux, S. Kim, J. Liang, J. P. Reifenberg, B. Rajendran, M. Asheghi, and K. E. Goodson, "Phase change memory," *Proceedings of the IEEE*, vol. 98, no. 12, pp. 2201–2227, 2010.
- [13] A. Makarov, V. Sverdlov, and S. Selberherr, "Emerging memory technologies: Trends, challenges, and modeling methods," *Microelectronics Reliability*, vol. 52, no. 4, pp. 628–634, 2012.
- [14] A. Nigam, C. Smullen, V. Mohan, E. Chen, S. Gurumurthi, and M. R. Stan, "Delivering on the promise of universal memory for spin-transfer torque ram (stt-ram)," in *Low Power Electronics and Design (ISLPED) 2011 International Symposium on*. IEEE, 2011, pp. 121–126.
- [15] N. Ekekwe, "Power dissipation and interconnect noise challenges in nanometer cmos technologies," *Potentials, IEEE*, vol. 29, no. 3, pp. 26–31, 2010.
- [16] K. JoonHan, N. Chan, S. Kim, B. Leung, V. Hecht, and B. Cronquist, "A novel flash-based fpga technology with deep trench isolation," in *Non-Volatile Semiconductor Memory Workshop, 2007 22nd IEEE*. IEEE, 2007, pp. 32–33.
- [17] Actel-Corporation, "Proasic3 flash family fpgas handbook," 2011.
- [18] Lattice-Semiconductor, "Lattice xp2 family handbook," 2010.
- [19] G. Sun, X. Dong, Y. Xie, J. Li, and Y. Chen, "A novel architecture of the 3d stacked mram l2 cache for cmps," in *High Performance Computer Architecture, 2009. HPCA 2009. IEEE 15th International Symposium on*. IEEE, 2009, pp. 239–249.
- [20] W. Zhao, E. Belhaire, Q. Mistral, E. Nicolle, T. Devolder, and C. Chappert, "Integration of spin-ram technology in fpga circuits," in *Solid-State and Integrated Circuit Technology, 2006. ICSICT'06. 8th International Conference on*. IEEE, 2006, pp. 799–802.
- [21] P. Gaillardon, M. Ben-Jamaa, G. Beneventi, F. Clermidy, and L. Perniola, "Emerging memory technologies for reconfigurable routing in fpga architecture," in *Electronics, Circuits, and Systems (ICECS), 2010 17th IEEE International Conference on*. IEEE, 2010, pp. 62–65.
- [22] (2012) International technology roadmap for semiconductors (ITRS). [Online]. Available: <http://www.itrs.net/Links/2012ITRS/Home2012.htm>
- [23] S. Paul, S. Mukhopadhyay, and S. Bhunia, "Hybrid cmos-sttram non-volatile fpga: Design challenges and optimization approaches," in *Proceedings of the 2008 IEEE/ACM International Conference on Computer-Aided Design*. IEEE Press, 2008, pp. 589–592.
- [24] S. Parkin, C. Kaiser, A. Panchula, P. M. Rice, B. Hughes, M. Samant, and S.-H. Yang, "Giant tunnelling magnetoresistance at room temperature with mgo (100) tunnel barriers," *Nature materials*, vol. 3, no. 12, pp. 862–867, 2004.
- [25] S. Khandelwal, S. Akashe, and S. Sharma, "Reducing leakage power for sram design using sleep transistor," *ACTA PHYSICA POLONICA A*, vol. 123, no. 2, pp. 185–187, 2013.
- [26] Z. Diao, Z. Li, S. Wang, Y. Ding, A. Panchula, E. Chen, L.-C. Wang, and Y. Huai, "Spin-transfer torque switching in magnetic tunnel junctions and spin-transfer torque random access memory," *Journal of Physics: Condensed Matter*, vol. 19, no. 16, p. 165209, 2007.
- [27] L. Perniola, V. Sousa, A. Fantini, E. Arbaoui, A. Bastard, M. Armand, A. Fargeix, C. Jahan, J.-F. Nodin, A. Persico *et al.*, "Electrical behavior of phase-change memory cells based on gete," *Electron Device Letters, IEEE*, vol. 31, no. 5, pp. 488–490, 2010.
- [28] R. I. Alip, R. Kobayashi, Y. L. Zhang, Z. Mohamad, Y. Yin, and S. Hosaka, "A novel phase change memory with a separate heater characterized by constant resistance for multilevel storage," *Key Engineering Materials*, vol. 534, pp. 136–140, 2013.
- [29] W. Zhang and T. Li, "Helmet: A resistance drift resilient architecture for multi-level cell phase change memory system," in *Dependable Systems & Networks (DSN), 2011 IEEE/IFIP 41st International Conference on*. IEEE, 2011, pp. 197–208.
- [30] P. F. Butzen and R. P. Ribas, "Leakage current in sub-micrometer cmos gates," *Universidade Federal do Rio Grande do Sul*, pp. 1–28, 2007.
- [31] (2013) Predictive technology model (ptm). [Online]. Available: <http://ptm.asu.edu/>
- [32] F. Xiong, A. D. Liao, D. Estrada, and E. Pop, "Low-power switching of phase-change materials with carbon nanotube electrodes," *Science*, vol. 332, no. 6029, pp. 568–570, 2011.