IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 23, 2021, accepted June 3, 2021, date of publication June 24, 2021, date of current version July 2, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3092167

A Survey of FPGA Logic Cell Designs in the Light of Emerging Technologies

SHUBHAM RAI[1], (Member, IEEE), PALLAB NATH[1], ANSH RUPANI[1], SANTOSH KUMAR VISHVAKARMA[2], (Member, IEEE), AND AKASH KUMAR[1], (Senior Member, IEEE)

[1]Chair for Processor Design, Technische Universität, 01062 Dresden, Germany
[2]Indian Institute of Technology Indore, Indore 453552, India

Corresponding author: Shubham Rai (shubham.rai@tu-dresden.de)

This work was supported by the German Research Foundation (DFG) through SecuReFET under Project 439891087.

ABSTRACT The functional component for an FPGA is the logic element which enables it to adapt to various hardware descriptions. This behavior is mostly due to the MUX-like functional flexibility provided by these logic elements or cells. However, in recent years, decelerating transistor sizing as per Moore's law has led to diminishing power, area and delay returns over cost. Hence, the idea of venturing into FPGA logic cell designs based on emerging technologies is becoming not merely attractive, but even inescapable. The present work surveys various conventional and non-conventional logic cell designs proposed in the literature by identifying four logic design families, namely LUT-based, cone-based, matrix/cluster-based and transistor array-based. We then carry out a detailed comparison at two levels - a quantitative comparison based on metrics like power, delay and area which govern the overall performance of various FPGA architectures and secondly a qualitative comparison on factors which are important considering the ease of mainstream adoption. We highlight the importance of introducing and co-optimizing novel devices and architectures to maximize the overall FPGA performance.

INDEX TERMS FPGA, logic cell, emerging technologies.

I. INTRODUCTION

Since 1984, the year Field Programmable Gate Arrays (FPGAs) were introduced, they are becoming ever more popular with researchers and the industry due to their very low design turn-over time. FPGAs provide the freedom to adapt the hardware as per application requirement. They boast of features like compile time and runtime reconfigurability, short time-to-market, easy prototyping etc. which make them a reliable and viable option in digital systems applications [2].

A typical FPGA architecture consists of the following primary components – (i) Logic cells for implementing logic; (ii) DSP blocks to accelerate complex arithmetic operations; (iii) Block RAMs for facilitating dedicated on-chip memory; (iv) Transceivers for high speed data transfer; (v) IO Blocks at periphery for external connections; and (vi) Interconnects and routing infrastructure to allow communication and transfer of signals between different blocks. Although, all the components play a role in deciding the overall efficiency of an FPGA architecture, this work focusses on the logic cell design, as it is the pivotal component that implements synthesized logic functions. The mainstream FPGAs rely on look-up-table (LUT)-based architectures [3]. These LUTs are the basic logic cells and are primarily based on static RAMs and multiplexers. A *k*-input LUT is capable of implementing any function of up to *k* inputs. However, this functional completeness and flexibility comes at the cost of compromised delay, area and power consumption as compared to Application Specific Integrated Circuits (ASICs) [4]. These problems are getting further aggravated because of technology scaling Esmaeilzadeh *et al.* [1]. Though technology scaling for Complementary Metal Oxide Semiconductor (CMOS) continues, the area, delay and power benefits over cost are getting further skewed and this calls for rethinking the FPGAs' logic cell architecture. For instance, the power consumption takes the maximum toll, as evident by the fact that with an operating voltage of 0.6V, around 50% of total energy consumption is wasted as leakage dissipation in modern FPGAs [5].

In order to circumvent CMOS scaling and the various challenges that come with it, researchers have proposed many architectural modifications and system-level techniques which either focus on a single metric like area, power

The associate editor coordinating the review of this manuscript and approving it for publication was Ye Zhou.

and delay or a particular combination of them. Many take a fresh perspective by involving emerging technologies like memristors [6]–[8], spintronics [9] and controllable-polarity transistors based on materials like silicon [10]–[13] and germanium [14] nanowires, carbon nanotubes [15] and 2D materials like WSe_2 [16]. Nikonov and Young [17] demonstrate the promises of beyond-CMOS devices by benchmarking them and helping researchers seek methods of improving their power versus performance.

This paper surveys various FPGA logic cell designs and techniques proposed in literature and discusses how emerging technologies along with novel micro-architectures aim to mitigate the performance losses. We further investigate how different proposed designs favor commercial aspects like computer-aided design (CAD) adaptability and scalability. In the present work, we have focused on covering all major FPGA logic cell designs and hence, discussion on routing and interconnects is beyond its current scope.

II. FPGA LOGIC CELL DESIGN

In this section, we discuss the evolution of CMOS-based FPGA design starting from its earliest days. Search for programmable hardwares started way back in 1980s. The initial Gate Arrays such as sea-of-gates [18] provided enough flexibility for their time in low-volume production scenarios but were pursued less due to being only one-time-programmable. Then, came multiplexer-based (MUX-based) FPGAs as MUXes were capable of implementing any logic functions. They paved the way for more capable memory-based LUTs. Here we discuss various logic cells proposed, focusing on the incremental changes, starting from the initial MUX/LUT-based designs.

A. LUT-BASED FPGAs

Look-up-Tables (LUTs) use the underlying concept as memories followed by a MUX-tree so that all the possible outputs of a function are stored in SRAMs. The output is then selected by function inputs (which are connected to the select lines) through a tree of MUXes. LUTs are simpler than the MUX-based designs due to the fact that configuration only means rewriting the content of the SRAMs, which in the latter case, means changing the connections to MUX inputs as well as the select lines. Following is a discussion on LUTs while classifying them into two broad classes:

Fixed-size LUTs: Xilinx employed LUT-based logic cell design, primarily consisting of 4-input and 6-input LUTs [19]. A generic LUT is shown in FIGURE 1a, which consists of a tree of pass-transistor-based MUXes whose select lines are driven by the LUT inputs. The configuration bits are stored in SRAMs which drive the output of the LUT upon selection by the inputs.

Fracturable LUTs: It is rare that LUTs with more than 4 or 6 inputs are used for implementing FPGAs [20]. Feng *et al.* [21] proposed the S44 architecture (shown in FIGURE 1b) and demonstrated that the depth properties of a 7-LUT can be achieved without having to pay the heavy

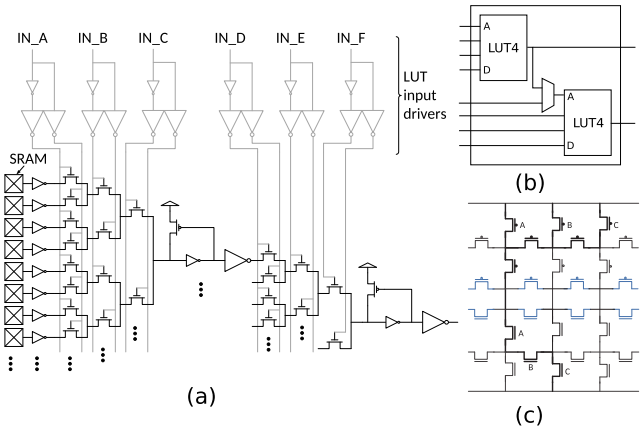

FIGURE 1. Logic cell designs proposed in the literature (a) A generic pass-transistor logic based LUT architecture [26] (b) S44 architecture [21] (c) A NAND3 implemented in an FPTA shown by transistors in bold [27].

area price. The proposed design is a 7-input structure composed of two tightly coupled 4-input LUTs. While it cannot implement all the 7-input functions, it can implement almost all 5-input functions, 98% of all 6-input functions and 75% of all 7-input functions. The S44 cell saves on the number of logic levels in the critical path, translating to better delay and a significant decrease in area compared to 6-LUT for industrial benchmarks. Similarly, Altera's *Adaptive Logic Module (ALM)* [22] can implement either one 6-input function or two 5-input functions or four 4-input functions. An ALM contains a 6-LUT that can be fractured into two 5-LUTs. In the case of two 5-LUTs in a single ALM, there must be no more than 8 unique inputs, so that a pair of 5-LUTs must share at least two inputs. In arithmetic modes, the 6-LUT can be used as four 4-LUTs, with two pairs of 4-LUT providing inputs to an in-built 2-bit adder.

B. FINE-GRAINED HETEROGENOUS ARCHITECTURE BASED FPGAs

Ebrahimi *et al.* [23] motivated their work on the fact that on average, 97% of the functions in industrial or standard benchmark circuits are homomorphic, i.e. they belong to any one among a few NPN classes.[1] They showed that these classes of functions can be efficiently implemented with power/performance-aware *Reconfigurable Hard Logic (RHL)*. It was shown that 3-input functions are the ones that cannot be efficiently mapped onto an RHL, and hence a 3-LUT is used to implement them, resulting in a heterogeneous architecture as shown in FIGURE 2. Their primary motive was to keep the static power dissipation of dark silicon in check. At the cost of area, they use additional power-gating logic and shared configuration SRAMs. It involves the isolation of the targeted logic from GND by intermediate transistors driven by an SRAM. They also power-gate the configu-

[1] An NPN (Negation-Permutation-Negation) class is a set of Boolean functions derived from each other by permuting and complementing the inputs, and complementing the output [24].

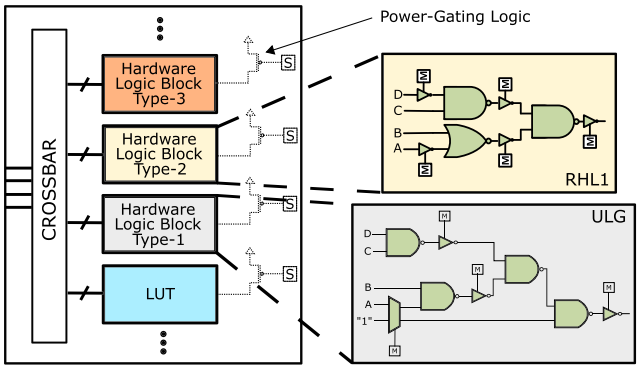

FIGURE 2. Hybrid logic blocks with LUT and reconfigurable hardware logic [23]–[25].

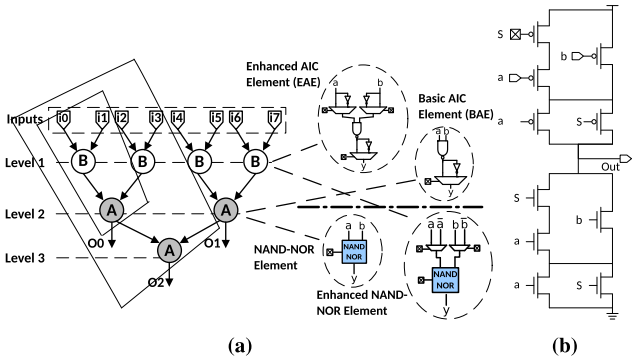

FIGURE 3. (a) And-inverter cone showing AIC (up) [28] & NAND-NOR (down) [29] cell. The square shows how outputs can be tapped (b) NAND-NOR cell [29].

ration SRAMs so that a few shared ones can be powered-off when not required for implementing a function within the logic cell. Luo *et al.* [25] demonstrated a similar architecture which has *Universal Logic Gates (ULGs)* (analogous to RHLs) and LUTs in a certain ratio in a *Configurable Logic Block* (CLB), as shown in [25]. They determined the ratio of LUT:ULG which gives delay, power or area centric results.

C. FIELD PROGRAMMABLE TRANSISTOR ARRAY (FPTA)

Tian *et al.* [27] reported better area utilization figures with a fine-grained implementation of functions onto a programmable transistor array. FPTAs employ configurable array of transistors which are regularly-arranged into rows and columns. We know that the (CLB) of FPGAs rely on *Look-up Tables (LUTs)* for implementing combinational functions along with a flip-flop for sequential ones. In other words, an FPGA might allot an entire LUT to implement a relatively small logic function, while an FPTA efficiently configures only the required number of columns. An implementation of NAND3 in an FPTA is shown in FIGURE 1c. Another trait of this fabric and design style is its ability to be quickly reconfigured within a single cycle of operation by changing the inputs to the configuration transistors.

D. AND-INVERTER CONE/NAND-NOR CONE

Inspired by recent trends in logic synthesis, Parandeh-Afshar *et al.* [28] proposed that basic blocks based on *And-Inverter Graphs (AIGs)* provide a better compromise between hardware complexity, delay, flexibility and input and output counts. The authors call this new logic block as *And-Inverter Cone (AIC)* (see FIGURE 3a). It is a simple circuit where arbitrary AIGs can be naturally mapped. They have the following promising features as opposed to LUTs: (1) the AIC has more number of inputs and outputs, which allows it to implement larger, multi-output functions; (2) the AIC structure is similar to the regular expression of boolean algebra, which allows it to satisfy the logic synthesis requirements and abide by its optimizations for an improved performance; (3) the AIC's area and delay increases linearly and logarithmically with the number of inputs, as opposed to the exponential and linear increase in the case of LUTs, respectively; (4) intermediate results can be directly reused through the intermediate outputs (known as side outputs) of the AIC, reducing logic duplication and improving the overall circuit area. The AIC cell along with the cone is shown in FIGURE 3a.

A crucial point of discussion of the AIC cell is that the propagation delay of the cell depends upon its configuration, which is rightly pointed out by the authors in [30]. In combinational circuits, different arrival times of signals that might otherwise transmit simultaneously, lead to signal competition and might cause glitches. Glitches often lead to instability and errors in the circuit. To tackle this issue, Huang *et al.* [29] improved upon the AIC cell with a new NAND-NOR cell as shown in FIGURE 3b which substitutes the AIC elements as shown in FIGURE 3a. The new cell was shown to have significantly lesser delay discrepancy (64% less) among its signal paths for different configurations and also has lesser number of transistors. This is made possible by the novel transistor-level design of the NAND-NOR cell and some architectural modifications with the input crossbars. One of the major changes is that the inverted inputs to the *Level 1* of the cone are provided by external *Delay-balanced Dual-phased Multiplexer (DDM)* crossbars. This alone leads to significant area and delay reduction, compared to the crossbar used with AIC cone.

III. NOVEL FPGA LOGIC CELL DESIGNS BASED ON EMERGING TECHNOLOGIES

The application of novel devices in FPGA design has picked up pace in the last decade with many technologies being introduced having different traits. One of the earliest work using emerging nanotechnologies for FPGA architectures was done by DeHon and Wilson [31] and DeHon [32]. He proposed gate arrays using nanowires for implementing programmable hardware architectures. Since then, many emerging technologies like ambipolar *Carbon Nanotubes Field Effect Transistors* (CNTFETs) [15] and *Silicon and Germanium Nanowires* SiNW [10], [11] or GeNW [33] transistors which are known for their reconfigurability and low static power dissipation,

memristors for their ability to act as both switch and storage element and spintronics for their overall power-efficiency have shown interesting results for FPGA architectures. They are used by researchers for proposing novel FPGA logic designs. In this section, we discuss the most prominent approaches.

A. LOGIC CELLS BASED ON RECONFIGURABLE CNTFETs

Jabeur *et al.* [34] proposed two 2-input cells designed to perform reconfigurable operations by exploiting the ambipolar property of double-gate CNTFETs. One of the cells is called SRC (Static Reconfigurable Cell) while the other is DRC (Dynamic/Domino Reconfigurable Cell). Both are capable of performing all the 16 functions (with 2 inputs) exploiting a specific correlation between inputs and configuration signals. They report upto 2X delay improvements with both static and dynamic logic cells. Cheng *et al.* [35] used reconfigurable logic cells based on ambipolar CNTFETs, which implement a subset of 2 or 3 input functions, and then arranged them in unique topologies to achieve functional completeness. This *fine-grained* approach showed better area utilization compared to a baseline LUT-based implementation.

B. NOVEL FINE-GRAINED LOGIC CELL AND THEIR CLUSTER BASED ON RECONFIGURABLE SiNW TRANSISTORS

Gaillardon *et al.* [36] investigated the implementation of FPGAs with fine-grained cells based on controllable-polarity transistors. They used a novel architecture which uses dynamic logic for logic cell architecture and is capable of implementing a major portion of all 2-input functions. The micro-architecture is a SiNW re-implementation of DRC. Using a $k \times k$ matrix-based *Basic Logic Element* (BLE) they compare their implementation with k-input LUTs and show improvements. The logic cell and the cluster are shown in FIGURE 4a and FIGURE 4b, respectively.

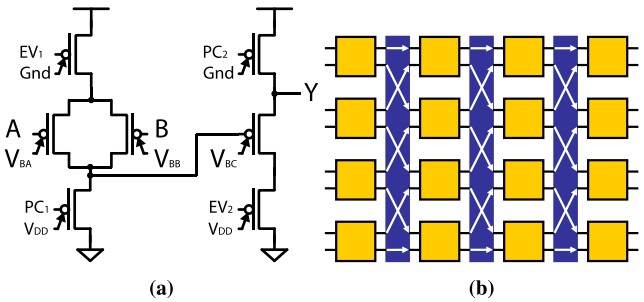

FIGURE 4. (a) Fine-grained logic cell based on SiNW and (b) 4 × 4 MCluster based on the fine-grained logic cells [36].

However, there were certain limitations related to the proposed design: The major concern was the use of domino logic which is highly susceptible to noise. The proposed logic cell relies on a 4-phase pseudo clocking signal which consists of two precharge (pc) and two evaluation (ev) signals. Each logic cell is connected to these four signals, which are always switching in synchronization leading to higher power dissipation. Intermediate D-flipflops are needed to support pre-charge and evaluation of each domino stage. Moreover, the delay of domino logic cells have been shown to change almost more than twice as much as compared to static logic with process variations [37]. Since an n-input logic cell is of size $n \times n$, it leads to an implicit logic excess. As the functions mapped onto a cell are convergent in nature, most of the logic nodes in the matrix would be configured as buffers or even left unused. On the other hand, circuits based on dynamic logic are known to be much faster and functionally dense as compared to static logic ones. This allowed the authors to implement eight 2-input functions with a 7-transistor cell.

C. ALL-SPINTRONIC NAND-NOR CELL-BASED FPGA

Among many emerging device technologies, spin-based devices, commonly referred to as spintronics, show promise because they have good scalability, non-volatility, and ultra-low energy [38]. Williams and Lin [9] aimed at fully extracting the benefits of new spin-based device technology. Specifically, they exploited the unique characteristics of a domain-wall logic device called the *mCell* [39] to achieve a direct mapping to NAND-NOR logic and proposed a high-throughput non-volatile alternative to LUT-based CMOS reconfigurable logic. As shown in FIGURE 5, their NAND-NOR cell consists of two programmable inverters. The inverters have a unique property that they exhibit similar delay in both inverting and non-inverting mode. This modification itself helps them in reducing the delay discrepancy between the best and worst-case configurations of the cell by more than 50%. Their simulation results show that, for a similar logic capacity, the NAND-NOR FPGA design with *mCell* devices excels across all metrics when compared to the CMOS-based NAND-NOR FPGA design. It is to be noted that using domain-wall devices as a drop-in replacement in a CMOS-style design may not be promising owing to the relatively low switching ratio inherent in domain-wall devices [39]. It requires design methods which can tolerate low switching ratios. In this light, Fan *et al.* used threshold logic approach to achieve almost twice the functionality for the same device count [40].

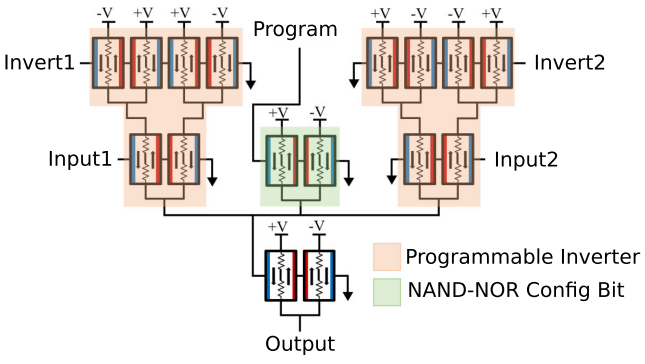

FIGURE 5. Spintronic-based NAND-NOR cell [9].

D. MEMRISTOR-BASED LUT CELL

The memristor emerged as the fourth passive element after resistor, capacitor and inductor when it was postulated by Chua [42] and then realized by Strukov *et al.* [43] using a nano scale thin film of titanium dioxide sandwiched between two platinum electrodes. Subsequent research on memristors [44]–[49] advocate it as a potential replacement for conventional memories due to its higher density, non-volatility, lower power consumption and faster read speeds. Xia *et al.* [50] successfully fabricated and demonstrated memristor-CMOS hybrid integrated circuits. Cong and Xiao [8] showed significant area savings on the existing CMOS-compatible memristor fabrication process by using memristors for programmable interconnects and laying them over logic blocks. Sampath *et al.* [51] showed memristor-based routing crossbars while Guo *et al.* [52] demonstrated a memristor-based logic cell to be a potential candidate for LUT implementation.

Kumar *et al.* [6] and Almurib *et al.* [53] proposed a memristor-based cell for an LUT of an FPGA. The authors implemented arrays of size 4 × 4 and 6 × 6 (equivalent to 4-LUT and 6-LUT) and compared them with the memory *cross-point* implementation and schemes in [44]–[46]. They achieved substantial delay improvement over the baseline. When compared to SRAM-based designs, the memristor-based LUTs are generally faster in terms of READ, but suffer in terms of WRITE [7]. In its defense, it can be argued that we WRITE to an FPGA once for initial configuration, and, infrequently for reconfiguration, but subsequently, it is READ many times.

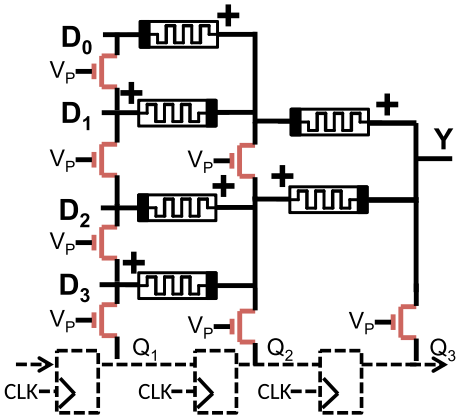

FIGURE 6. Memristor-based GMS cell [41].

Gaillardon *et al.* [41] proposed a *Generic Memristive Structure (GMS)* cell, as shown in FIGURE 6 and used it to replace the pass-transistor-based MUXes in LUTs. Additionally, they used the *GMS* to replace the routing MUXes. As a result, improvements in delay and area are observed along with a reduction in power as leakage in memristor-based MUXes is almost non-existent.

IV. COMPARISON

In this section, we carry out a detailed comparative analysis among the surveyed logic cell designs. Further, we present an overview on various emerging technologies by comparing them in the context of their suitability for implementing FPGA architectures. We also point out at methods to better embrace emerging technologies as they mature for FPGA architectures. Finally, we qualitatively summarize all the designs and discuss about factors which are important for their scalability and commercial viability.

A. SETUP

We group the surveyed logic cell designs into four families based on their architectures and design topologies – matrix/cluster-based, cone-based, LUT-based and transistor array-based. Most of the works on novel logic cell designs either compare their proposed design with the corresponding similar input LUT or show a benchmark level evaluation or both. We have compiled all these evaluations in TABLE 1 and FIGURE 7. TABLE 1 reports the area, delay and power numbers as quoted in respective works. While TABLE 1 shows individual logic cell-level comparisons, FIGURE 7 shows a quantitative comparison among all the designs in terms of area and delay (and power for available data) over the MCNC benchmarks.[2] All the values are normalized to a 4-LUT. The normalization calculations are done based on the 4-LUT vs. 6-LUT comparison reported in [21]. The background color in FIGURE 7 signifies that a design is better as its gradient darkens, thus indicating better area-delay product.

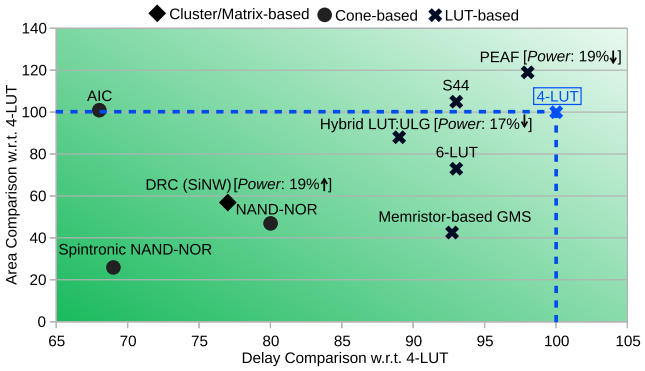

FIGURE 7. Comparison among all designs in terms of area, delay and power over MCNC benchmarks, normalized to a 4-LUT (Available power figures within braces).

B. QUANTITATIVE ANALYSIS

1) CLUSTER/MATRIX-BASED

For the cluster/matrix-based designs employing logic cells constructed using reconfigurable CNTFETs, it can be seen from TABLE 1 that SRC and DRC cells have 50% and 58% delay improvement respectively as compared to 2-LUT based

[2]We have shown evaluation over MCNC benchmarks because it was the most common denominator across the proposed logic cell designs. Discussion about works which have carried out evaluation over VTR or other benchmarks has been done in the text.

TABLE 1. Quantitative comparison among proposed logic cell designs in terms of delay, area and power (↓: less, ↑: more).

Family	Design	Results shown for	Compared with	Delay	Area	Power
Cluster/Matrix-based	CNTFET-based (Static) [34]	SRC cell (CNTFET)	2-LUT (CNTFET)	50% ↓	No difference	5% ↓
	CNTFET-based (Dynamic) [34]	DRC cell (CNTFET)	2-LUT (CNTFET)	58% ↓	50% ↑	26% ↑
Cone-based	And-Inverter Cone [28]	6-level AIC cell (CMOS)	6-LUT cell (CMOS)	20% ↓	138% ↑	—
	NAND-NOR Cone [29]	6-level NAND-NOR cell	6-LUT cell (CMOS)	30% ↓	—	—
	Spintronic NAND-NOR [9] Cone	6-level Spintronic NAND-NOR Cluster	6-level CMOS NAND-NOR Cluster	17% ↓	—	57% ↓
LUT-based	NPN Classification-based Hybrid LUT [23]	RHL1, RHL2, RHL3	4-LUT (CMOS)	55% ↓, 39% ↓, 43% ↓	73% ↓, 57% ↓, 55% ↓	84% ↓, 77% ↓, 71% ↓
	Memristor-based LUT [6]	4x4, 6x6 matrix (4x4=4-LUT, 6x6=6-LUT)	[44, 45, 46]	97% ↓, 97% ↓	—	Energy: 56% ↓, 60% ↓

on CNTFET. It is due to their minimalistic design using ambipolar devices and the fast switching characteristics of CNTFETs.

Progressing with the cluster-based approach, for the DRC logic cell re-implemented with SiNW transistors, the delay results draw parallels with the results reported for DRC with CNTFETs. As shown in FIGURE 7, the novel matrix-based cluster design helps in offsetting the area for a 43% advantage over CMOS based 4-LUT. The delay is 23% less than the baseline 4-LUT-based FPGA. Again, the use of dynamic logic along with the fact that SiNW transistors dissipate higher dynamic power, leads to a 19% power penalty. However, impact of intra-cellular routing was not discussed in their evaluation. Given the fact that interconnect and fanout can account for up to 90% of total FPGA power [54], it is an important parameter for evaluation of FPGA performance.

2) CONE-BASED

As shown in TABLE 1, a one-to-one comparison between just the 6-level AIC cell and 6-LUT show that the AIC has 20% less delay but the area is more than 2X. But a one-to-one comparison is not indicative of the actual performance of the logic cells as the LUT has 6 inputs while a 6-level AIC has $2^6 = 64$ inputs. As illustrated in FIGURE 7, evaluation over MCNC benchmarks shows that 6-level AIC has a 32% advantage in terms of delay, while the area is comparable. This advantage is due to two primary reasons. First, the ease with which a function graph with many inputs can be mapped onto a cone and second, the ability to tap-out intermediate outputs, which discourages logic duplication and thus, keeps area in check.

As reported in [30], the delay glitch caused by the delay variation in the cone proposed in [28] may induce additional dynamic power consumption and affect the overall stability of the circuit. The use of the NAND-NOR cell proposed by Huang *et al.* [29] along with the use of DDM crossbars contribute to achieving lesser delay discrepancy among the configurations of the cone. As shown in TABLE 1, the 6-level NAND-NOR cell has 30% lesser delay compared to a baseline 6-LUT based implementation. Results with MCNC and VTR benchmarks show a 14% and 3% improvement in delay and 35% and 18% improvement in area, respectively, when compared to a 6-LUT implementation.[3] Compared to 6-AIC, it is 13% and 4% faster and has 21% and 15% lesser area footprint for MCNC and VTR benchmarks respectively. When normalized to a 4-LUT, it is 20% faster and 53% smaller in terms of area over the MCNC benchmarks, as shown in FIGURE 7. However, the authors do not discuss the power dissipation of the cone-based designs. It can be ascertained that the power due to configuration SRAMs would be less than LUTs because for an n-input cone, there are only n-1 SRAMs. In the case of LUTs, an n-input cell has 2^n SRAMs, which add to the power overhead. However, despite the improvements over AIC, NAND-NOR cones still have an input-to-output delay variance of upto 20% [9], and the glitches caused by it are not in the favor of power as it might cause extra leakage dissipation. Additionally, the delay variance burdens the FPGA timing analysis CAD tools and might make the idea of achieving dynamic reconfigurability with this architecture a challenge.

For the *all-spintronic* implementation of the NAND-NOR cone as proposed in [9], it can be seen from TABLE 1 that a 6-level all-spintronic NAND-NOR cluster has 17% lesser delay and a radical 57% lesser power dissipation as compared to a CMOS implementation of 6-level NAND-NOR. Reported results over MCNC and VTR benchmarks show a delay improvement of 26% and 14% while an area improvement of 65% and 55%, respectively when compared to CMOS-based 6-LUT implementation. It also decreases the delay variance between configurations by 59% compared to the baseline implementation. After normalizing the area and delay to a 4-LUT, it can be seen in FIGURE 7, that the delay is 31% less and area is a radical 73% less. Owing to the improvements, it eases the burden on FPGA timing analysis and paints a promising picture for its use in applications

[3]The improvements reported in [28]–[30] are shown as geometric means.

requiring dynamic reconfiguration. Moreover, since the output signals in spintronics is current-based, a serial fanout scheme is required to send the same amount of current in downstream gates, rather than the large tapered buffers needed in CMOS. This helps to save overall circuit energy consumption.

3) LUT-BASED

FIGURE 7 shows the delay, power and area figures for conventional 6-LUT as compared to a 4-LUT over the MCNC benchmarks. It is proven that 4-LUT and 6-LUT are the most-used LUT sizes across all FPGAs owing to their optimal area and performance figures respectively. For the S44 architecture, the delay over MCNC and industrial benchmarks for S44 is 7% and 3% less as compared to a 4-LUT-based, respectively. However, as seen in FIGURE 7, the area over MCNC benchmarks increase by 5%. The non-routability of the intra-cell connections is cited as the reason for this [55]. The area over industrial benchmarks is 4% less as compared to 4-LUT. This is attributed to the presence of less logic near the critical path which might otherwise trigger logic duplication. The authors do not report any power figures but indicate that the S44 cell would have a static power advantage over 6-LUT as static power tends to correlate with area— and S44 has a smaller area as compared to 6-LUT.

The power-gating approach with RHLs taken by Ebrahimi *et al.* [23] helps reduce the total power dissipation over MCNC, VTR and IWLS benchmarks by an average of 19%, when compared to a 4-LUT design. Due to the minimalistic transistor footprint of the RHLs, the delay over the same benchmarks fared 2% better than the implementation on conventional 4-LUT. However, as shown in FIGURE 7, the area takes a toll, and is 19% more than the baseline. Similarly, the LUT-ULG hybrid approach proposed by Luo *et al.* [25] achieve 11%, 10% and 17% better figures for delay, area and power respectively, when compared to 4-LUT. The delay figure is for a LUT:ULG ratio of 3:7. This is an optimal ratio because a ratio smaller than this (1:9 or 2:8) would mean fewer LUTs in a CLB, which might lead to the use of extra CLBs to accommodate the need for more LUTs in a circuit. A ratio higher than 3:7 means that the percentage of ULGs in the total implemented circuit decreases, which leads to diminishing delay benefits. Due to the same reasons, the area and power figures are for a LUT:ULG ratio of 4:6.

As can be seen in TABLE 1, the memristor-based LUT [7] has a READ delay improvement of 97% as compared to the *cross-point* memory structures and scheme used in [44]–[46]. The energy dissipation for a READ operation on the LUT is 56% and 60% less for 4×4 and 6×6 matrices respectively. However, there is no mention of area. But it can be speculated that the area would be comparable, or even lesser, which is decided by two opposing factors– 1) The controller circuit required to WRITE/READ into the memory cells and the presence of amplifiers which have passive components like resistors and capacitors, and 2), the density of laying memristor cells onto the substrate [56]. The former pushes area margins higher while the latter brings it lower. Similarly, the GMS-based implementation by [41] shows 7% better delay and 58% lesser area compared to 4-LUT, over MCNC benchmarks (as shown in FIGURE 7).

4) TRANSISTOR-ARRAY BASED

The Field Programmable Transistor Array (FPTA) proposed by Tian *et al.* [27] is shown to have a 15% lower area utilization compared to the baseline architecture of Altera Stratix EP1S10, as shown in TABLE 1.

FIGURE 7 gives an overall picture for all the four design families in terms of delay and area with respect to our baseline (4-LUT). We can see that most of the design families are within the blue-dotted rectangle which show better area or delay numbers or both as compared to 4-LUT. Concluding remarks for power, delay and area metrics are as follows:

a: POWER

Power is at the peak of all concerns which researchers try to tackle, with the modern devices becoming more and more mobile and ubiquitous. Especially with the number of transistors that are packed into a single die already in the billions, static power dissipation is becoming a bigger concern as compared to dynamic power. From FIGURE 8, it can be observed that hybrid-logic-based, memristor and spintronic-based cells are the primary power-centric designs. More significant power savings are shown inherently by spintronic and memristor-based cells.

b: DELAY

It is to be noted that not all the designs are delay centric, and the ones which are, do so by either employing novel cone-based architecture or using devices like CNTFETs and memristors. The best delay improvement is shown by the cone-based architectures especially spintronic NAND-NOR (as evident from FIGURE 7). This is primarily because they use lesser memory elements like SRAMs compared to LUTs. On the other hand, hybrid-logic [23], [25] showed comparable performance as compared to 4-LUTs.

c: AREA

Area is however, a very loosely-defined criterion. Logic density is a more crucial metric which researchers are targeting by proposing functionally dense fine-grained cells. Again, among all the designs compared in terms of area, spintronic NAND-NOR is the most noteworthy (see FIGURE 7). Special techniques like the use of intermediate outputs which are possible using fracturable LUTs and AIC architectures are specially useful in saving area. The memristor-based GMS cell trails closely behind due to the presence of elaborate programming circuitry, but is nonetheless promising. Since in a reconfigurable logic, up to 40% area is dedicated to the storage of configuration signals [57], the functional overlap of logic with configuration in memristors leads to savings in area.

C. OVERVIEW OF LOGIC CELLS BASED ON EMERGING TECHNOLOGIES

From FIGURE 7, it is apparent that most of the gains in terms of area, power and delay are in the case when emerging technologies are used. On the basis of quantitative results, we have compiled a high-level comparison among various emerging technologies used for FPGA logic cell design in TABLE 2. Recently, a 16-bit RISC-based processor has been demonstrated using CNTFETs [58]. Among the ambipolar technologies, CNTFETs show faster performance but often show higher power consumption as compared to CMOS. The biggest problem is with their adoption into mainstream electronics because of their instability issues at room temperature and in non-vacuum conditions [59].

TABLE 2. Qualitative comparison among emerging technologies w.r.t. CMOS (Legend: Better 🟢, Similar 🟡, Worse 🔴).

Emerging Technologies	Delay	Power	Area	Volatility	Ease of adoption
Ambipolar CNTFETs	🟢	🔴	🟡	Volatile	🔴
Ambipolar SiNW RFETs	🟡	🟡	🟡	Volatile	🟢
Spin-based devices	🔴	🟢	🟢	Non-volatile	🔴
Memristors	🟢	🟢	🟡	Non-volatile	🟢

SiNW based FETs are more readily adoptable because they follow the same top-down manufacturing process as CMOS [60], [61]. However, SiNW FET are Schottky-contact based devices and hence tend to be slower as compared to CMOS [62]. In their current state, although they are frugal in static power dissipation, they end up dissipating more dynamic power due to the presence of more parasitic capacitances. However, it might be rewarding to pursue voltage scaling and multi-threshold techniques [63] to further curtail power in these devices. The above two technologies are still charge-based and the movement of carrier charges defines the logic function.

In case of spin-based devices and memristors, the operating mechanism is different, with the former being current-based. In case of spin-based devices, the information is encoded in the spin of the electron. This requires taking care of many aspects [64] for it to be a viable option. Hence, naturally as a technology, they are complicated and expensive to be built commercially. On the other hand, memristors are easier to adopt and there have been works which demonstrate fabricated reconfigurable logic circuits based on CMOS compatible processes [50]. Thus, memristors show better promise in the near future as they have much better delay and power figures as compared to CMOS.

D. QUALITATIVE SUMMARY

Each of the techniques mentioned in the previous section either targets overall performance gains compared to the conventional CMOS counterpart or focuses on improving upon a specific metric. They also have various degrees of CAD complexity and ease of adaptation. A qualitative comparison among all the designs can be seen in FIGURE 8.

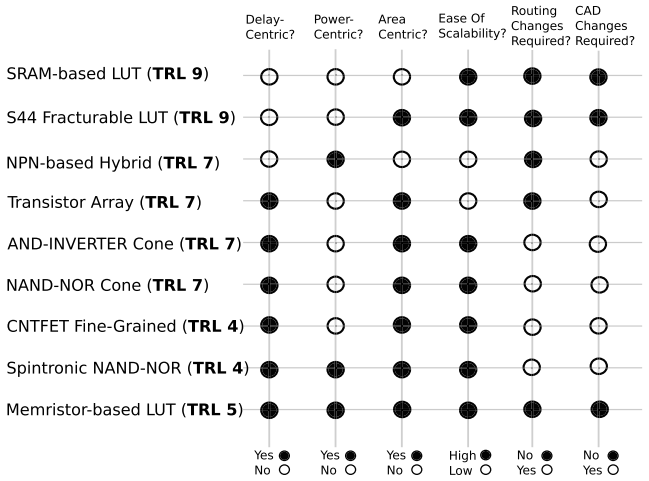

FIGURE 8. A qualitative comparison of all designs on parameters affecting mainstream adoption for FPGA architectures.

1) SCALABILITY

All logic designs do not scale with the same ease as an LUT-based design. Designs like NPN-based and memristor-based are supposed to scale up in a manner similar to LUTs, cones based on CMOS or spintronics scale based on the number of levels and inputs, while the CNTFET-based cells scale in a grid-like fashion. NPN-based hybrid designs don't scale well (see FIGURE 8) because with higher number of inputs, the NPN classes increase exponentially and it becomes challenging to find suitable RHLs that cover a good subset of them. With transistor arrays [27], the configuration overhead of each transistor in each column grows beyond viability with scaling. The intra-cellular routing wires also factor in when the cell scales up. This is more pronounced in matrix/cluster-based designs, which, in the worst case, might downplay the benefits of the finer logic grains.

2) CAD MODIFICATION

Novel architectures like AIC/NAND-NOR cones and matrix-like cluster need novel mapping algorithms as the individual units/cells implement only a subset (without input correlation) of the function space for a given number of inputs. In case of the matrix-based approach, it is essential to map an n-LUT's functions to an equivalent matrix of size k $\times$ k, where each logic cell implements a subset of all n-input functions. In case of the cone-based approach, the function-graph of a logic needs to be mapped onto the cone in a depth-constrained manner [65], where each logic cell can only switch between NAND and NOR functionality. For emerging technologies, steps such as physical synthesis will play a role and need more exploration. Among memristors and spintronics, the former is closer to adoption backed by fabricated demonstrations and measurement results, while spintronics is still in its infancy.

3) ROUTING ARCHITECTURE MODIFICATION

Most of the logic cell design families like the cluster-based and LUT-based (including NPN-based, memristor-based and S44) use conventional input crossbar MUXes (fully-connected or depopulated). However, the conical design with NAND-NOR proposed by Huang *et al.* [29] use a special Dual-phased Delay-balanced Multiplexer (DDM). Also, memristor-based implementation need additional READ/WRITE-control circuitry. This is closely related to routing algorithms in FPGAs as they need to mould themselves according to the capabilities of a certain design topology. For instance, with an appropriate control circuitry, a memristor-based crossbar can be used for both routing and logic implementation.

E. TECHNOLOGY READINESS LEVELS

In order to compare all the proposed logic cell designs, we have used the concept of *Technology readiness levels* (TRLs) [66] as a recognized figure of merit. We have added TRLs to each of the individual technology in FIGURE 8. Of these, the first two i.e. SRAM-based LUTs and S44 fracturable LUT are at TRL-9 which implies that these are in the class of ''actual system proven in operational environment''. These logic cells are contemporary solutions in commercial FPGAs. The next four are at TRL-7 which implies ''system prototype demonstration in operational environment'' as they are CMOS-based and hence are fabrication-ready. However, an actual system has not been demonstrated. The CNTFET-based logic cells (cluster-based logic cells) or spin-based are based on emerging nanotechnologies. These logic cells are based on models developed in lab and hence are not fully mature in terms of commercial adoption. Finally, memristor, as a technology is at a more advanced state (TRL-5) and are also available as components from industry [67]. However, LUTs based on memristors are based on available memristor technology models and a full-fledged system has not been yet demonstrated.

V. CONCLUSION AND FUTURE DIRECTIONS

Emerging technologies and novel architectures indeed provide opportunities to further increase FPGA performance and keep Moore's Law alive. However, there is no winning formula to a design that is better than the baseline in terms of delay, power and area while also being cost-effective and easy to adapt and implement. We can deduce from the existing work that while power-gating with *hard-logics* is still keeping CMOS alive (in terms of power), adopting and co-optimizing novel devices like SiNW, spintronics and memristors into architectures like hybrid clusters and cones holds promise for further improving power figures. However, there are a number of caveats which prevent their blind adoption. Memristors, on the other hand, open up an entirely new paradigm of *In-Memory Computing* [68], but come with their own set of challenges like mass-integration into the current fabrication flow, *sneak-currents* affecting the READ and WRITE characteristics, and ringing-effect associated with RESTORE after READ [6].

As device technology matures, researchers will come up with more optimized CAD tools that better exploit their capabilities. The target is to employ the emerging technologies and optimize them to achieve holistic system-level efficiency and complement current methods like partial reconfiguration. While the search for the *universal switch/memory* continues, pursuing heterogeneous architectures, by employing each technology's forte, could be the most promising step towards the future.

REFERENCES

[1] H. Esmaeilzadeh, E. Blem, R. St. Amant, K. Sankaralingam, and D. Burger, ''Dark silicon and the end of multicore scaling,'' in *Proc. 38th Annu. Int. Symp. Comput. Archit. (ISCA)*, 2011, pp. 365–376.
[2] D. Chinnery and K. Keutzer, *Closing the Gap Between ASIC & Custom: Tools and Techniques for High-Performance ASIC Design*. Springer, 2002.
[3] V. Betz, J. Rose, and A. Marquardt, *Architecture and CAD for Deep-Submicron FPGAs*, vol. 497. Springer, 2012.
[4] I. Kuon and J. Rose, ''Measuring the gap between FPGAs and ASICs,'' *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 26, no. 2, pp. 203–215, Feb. 2007.
[5] H. Qi, O. Ayorinde, and B. H. Calhoun, ''An energy-efficient near/sub-threshold FPGA interconnect architecture using dynamic voltage scaling and power-gating,'' in *Proc. FPT*, 2016, pp. 20–27.
[6] T. Nandha Kumar, H. A. F. Almurib, and F. Lombardi, ''On the operational features and performance of a memristor-based cell for a LUT of an FPGA,'' in *Proc. 13th IEEE Int. Conf. Nanotechnol. (IEEE-NANO)*, Aug. 2013, pp. 71–76.
[7] T. N. Kumar, H. A. F. Almurib, and F. Lombardi, ''A novel design of a memristor-based look-up table (LUT) for FPGA,'' in *Proc. IEEE Asia Pacific Conf. Circuits Syst. (APCCAS)*, Nov. 2014, pp. 703–706.
[8] J. Cong and B. Xiao, ''MrFPGA: A novel FPGA architecture with memristor-based reconfiguration,'' in *Proc. IEEE/ACM Int. Symp. Nanosc. Archit.*, Jun. 2011, pp. 1–8.
[9] S. M. Williams and M. Lin, ''Architecture and circuit design of an all-spintronic fpga,'' in *Proc. ISFPGA*, 2018, pp. 41–50.
[10] A. Heinzig, S. Slesazeck, F. Kreupl, T. Mikolajick, and W. M. Weber, ''Reconfigurable silicon nanowire transistors,'' *Nano Lett.*, vol. 12, no. 1, pp. 119–124, Jan. 2012.
[11] M. De Marchi, D. Sacchetto, S. Frache, J. Zhang, P.-E. Gaillardon, Y. Leblebici, and G. De Micheli, ''Polarity control in double-gate, gate-all-around vertically stacked silicon nanowire FETs,'' in *IEDM Tech. Dig.*, Dec. 2012, pp. 4–8.
[12] S. Rai, A. Rupani, D. Walter, M. Raitza, A. Heinzig, T. Baldauf, J. Trommer, C. Mayr, W. M. Weber, and A. Kumar, ''A physical synthesis flow for early technology evaluation of silicon nanowire based reconfigurable FETs,'' in *Proc. DATE*, 2018, pp. 605–608.
[13] G. Gore, P. Cadareanu, E. Giacomin, and P.-E. Gaillardon, ''A predictive process design kit for Three-Independent-Gate field-effect transistors,'' in *Proc. IFIP/IEEE 27th Int. Conf. Very Large Scale Integr. (VLSI-SoC)*, Oct. 2019, pp. 172–177.
[14] J. Trommer, A. Heinzig, S. Slesazeck, U. Mühle, M. Löffler, D. Walter, C. Mayr, T. Mikolajick, and W. Weber, ''Reconfigurable germanium transistors with low source-drain leakage for secure and energy-efficient doping-free complementary circuits,'' in *Proc. DRC*, 2017, pp. 1–2.
[15] Y.-M. Lin, J. Appenzeller, J. Knoch, and P. Avouris, ''High-performance carbon nanotube field-effect transistor with tunable polarities,'' *IEEE Trans. Nanotechnol.*, vol. 4, no. 5, pp. 481–489, Sep. 2005.
[16] G. V. Resta, S. Sutar, Y. Balaji, D. Lin, P. Raghavan, I. Radu, F. Catthoor, A. Thean, P.-E. Gaillardon, and G. de Micheli, ''Polarity control in WSe2 double-gate transistors,'' *Sci. Rep.*, vol. 6, no. 1, Sep. 2016.
[17] D. E. Nikonov and I. A. Young, ''Benchmarking of beyond-CMOS exploratory devices for logic integrated circuits,'' *IEEE J. Explor. Solid-State Comput. Devices Circuits*, vol. 1, pp. 3–11, Dec. 2015.
[18] P. Duchene and M. J. Declercq, ''A highly flexible sea-of-gates structure for digital and analog applications,'' *IEEE J. Solid-State Circuits*, vol. 24, no. 3, pp. 576–584, Jun. 1989.

[19] Xilinx. (2012). *Virtex-5 FPGA User Guide*. [Online]. Available: https://www.xilinx.com/support/documentation/user_guides/ug190.pdf

[20] S. Singh, J. Rose, P. Chow, and D. Lewis, "The effect of logic block architecture on FPGA performance," *IEEE J. Solid-State Circuits*, vol. 27, no. 3, pp. 281–287, Mar. 1992.

[21] W. Feng, J. Greene, and A. Mishchenko, "Improving FPGA performance with a s44 LUT structure," in *Proc. ISFPGA*, 2018.

[22] Altera. (2006). *FPGA Architecture White Paper*. [Online]. Available: https://www.intel.com/content/dam/www/programmable/us/en/pdfs/literature/wp/wp-01003.pdf

[23] Z. Ebrahimi, B. Khaleghi, and H. Asadi, "PEAF: A power-efficient architecture for SRAM-based FPGAs using reconfigurable hard logic design in dark silicon era," *IEEE Trans. Comput.*, vol. 66, no. 6, pp. 982–995, Jun. 2017.

[24] G. D. Micheli, *Synthesis and Optimization of Digital Circuits*, 1st ed. New York, NY, USA: McGraw-Hill, 1994.

[25] T. Luo, H. Liang, W. Zhang, B. He, and D. Maskell, "A hybrid logic block architecture in FPGA for holistic efficiency," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 64, no. 1, pp. 71–75, Jan. 2017.

[26] C. Chiasson and V. Betz, "COFFE: Fully-automated transistor sizing for FPGAs," in *Proc. Int. Conf. Field-Program. Technol. (FPT)*, Dec. 2013, pp. 34–41.

[27] J. Tian, G. R. Reddy, J. Wang, W. Swartz, Y. Makris, and C. Sechen, "A field programmable transistor array featuring single-cycle partial/full dynamic reconfiguration," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, 2017, pp. 1336–1341.

[28] H. Parandeh-Afshar, H. Benbihi, D. Novo, and P. Ienne, "Rethinking FPGAs: Elude the flexibility excess of LUTs with and-inverter cones," in *Proc. ISFPGA*, 2012, pp. 119–128.

[29] Z. Huang, X. Wei, G. Zgheib, W. Li, Y. Lin, Z. Jiang, K. Tu, P. Ienne, and H. Yang, "NAND-NOR: A compact, fast, and delay balanced FPGA logic element," in *Proc. ISFPGA*, 2017, pp. 135–140.

[30] G. Zgheib, L. Yang, Z. Huang, D. Novo, H. Parandeh-Afshar, H. Yang, and P. Ienne, "Revisiting and-inverter cones," in *Proc. ISFPGA*, 2014, pp. 45–54.

[31] A. DeHon and M. J. Wilson, "Nanowire-based sublithographic programmable logic arrays," in *Proc. ACM/SIGDA 12th Int. Symp. Field Program. Gate Arrays (FPGA)*, New York, NY, USA, 2004, pp. 123–132, doi: 10.1145/968280.968299.

[32] A. DeHon, "Design of programmable interconnect for sublithographic programmable logic arrays," in *Proc. ACM/SIGDA 13th Int. Symp. Field-Program. Gate Arrays (FPGA)*, New York, NY, USA, 2005, pp. 127–137, doi: 10.1145/1046192.1046210.

[33] J. Trommer, A. Heinzig, A. Heinrich, P. Jordan, M. Grube, S. Slesazeck, T. Mikolajick, and W. M. Weber, "Material prospects of reconfigurable transistor (RFETs)—From silicon to germanium nanowires," *MRS Proc.*, vol. 1659, no. 1, pp. 225–230, 2014.

[34] K. Jabeur, N. Yakymets, I. O'Connor, and S. Le-Beux, "Fine-grain reconfigurable logic cells based on double-gate cntfets," in *Proc. 21st Ed. Great Lakes Symp. Great Lakes Symp. (VLSI)*, 2011, pp. 19–24.

[35] K. Cheng, S. Le Beux, and I. O'Connor, "Hybrid topologies for reconfigurable matrices based on nano-grain cells," in *Proc. IEEE Int. Conf. Rebooting Comput. (ICRC)*, Nov. 2017, pp. 1–8.

[36] P.-E. Gaillardon, X. Tang, G. Kim, and G. De Micheli, "A novel FPGA architecture based on ultrafine grain reconfigurable logic cells," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 23, no. 10, pp. 2187–2197, Oct. 2015.

[37] M. Alioto, G. Palumbo, and M. Pennisi, "Understanding the effect of process variations on the delay of static and domino logic," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 18, no. 5, pp. 697–710, May 2010.

[38] S. Manipatruni, D. E. Nikonov, and I. A. Young, "Material targets for scaling all-spin logic," *Phys. Rev. A, Gen. Phys. Appl.*, vol. 5, no. 1, Jan. 2016, Art. no. 014002.

[39] D. Morris, D. Bromberg, J.-G. Zhu, and L. Pileggi, "mLogic: Ultra-low voltage non-volatile logic circuits using STT-MTJ devices," in *Proc. DAC Design Autom. Conf.*, 2012, pp. 486–491.

[40] D. Fan, M. Sharad, and K. Roy, "Design and synthesis of ultralow energy spin-memristor threshold logic," *IEEE Trans. Nanotechnol.*, vol. 13, no. 3, pp. 574–583, May 2014.

[41] P.-E. Gaillardon, D. Sacchetto, S. Bobba, Y. Leblebici, and G. De Micheli, "GMS: Generic memristive structure for non-volatile FPGAs," in *Proc. IEEE/IFIP 20th Int. Conf. VLSI System-on-Chip (VLSI-SoC)*, Oct. 2012, pp. 94–98.

[42] L. Chua, "Memristor–The missing circuit element," *IEEE Trans. Circuit Theory*, vol. CT-18, no. 5, pp. 507–519, May 1971.

[43] D. B. Strukov, G. S. Snider, D. R. Stewart, and R. S. Williams, "The missing memristor found," *Nature*, vol. 453, no. 7191, p. 80, 2008.

[44] Y. Ho, G. M. Huang, and P. Li, "Dynamical properties and design analysis for nonvolatile memristor memories," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 58, no. 4, pp. 724–736, Apr. 2011.

[45] N. Z. Haron and S. Hamdioui, "On defect oriented testing for hybrid CMOS/Memristor memory," in *Proc. Asian Test Symp.*, Nov. 2011, pp. 353–358.

[46] C. Xu, X. Dong, N. P. Jouppi, and Y. Xie, "Design implications of memristor-based RRAM cross-point structures," in *Proc. Design, Autom. Test Eur.*, Mar. 2011, pp. 1–6.

[47] P. Junsangsri and F. Lombardi, "A novel hybrid design of a memory cell using a memristor and ambipolar transistors," in *Proc. 11th IEEE Int. Conf. Nanotechnol.*, Aug. 2011, pp. 748–753.

[48] A. Chen, "Accessibility of nano-crossbar arrays of resistive switching devices," in *Proc. 11th IEEE Int. Conf. Nanotechnol.*, Aug. 2011, pp. 1767–1771.

[49] M. M. Ziegler and M. R. Stan, "Design and analysis of crossbar circuits for molecular nanoelectronics," in *Proc. 2nd IEEE Conf. Nanotechnol.*, Aug. 2002, pp. 323–327.

[50] Q. Xia, W. Robinett, M. W. Cumbie, N. Banerjee, T. J. Cardinali, J. J. Yang, W. Wu, X. Li, W. M. Tong, D. B. Strukov, G. S. Snider, G. Medeiros-Ribeiro, and R. S. Williams, "Memristor—CMOS hybrid integrated circuits for reconfigurable logic," *Nano Lett.*, vol. 9, no. 10, pp. 3640–3645, Oct. 2009.

[51] M. Sampath, P. S. Mane, and C. Ramesha, "Hybrid CMOS-memristor based FPGA architecture," in *Proc. Int. Conf. VLSI Syst., Archit., Technol. Appl. (VLSI-SATA)*, 2015, pp. 1–6.

[52] Y. Guo, X. Wang, and Z. Zeng, "A compact memristor-CMOS hybrid Look-Up-Table design and potential application in FPGA," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 36, no. 12, pp. 2144–2148, Dec. 2017.

[53] H. A. F. Almurib, T. N. Kumar, and F. Lombardi, "A memristor-based LUT for FPGAs," in *Proc. 9th IEEE Int. Conf. Nano/Micro Engineered Mol. Syst. (NEMS)*, Apr. 2014, pp. 448–453.

[54] Y. Lin, F. Li, and L. He, "Routing track duplication with fine-grained power-gating for FPGA interconnect power reduction," in *Proc. Conf. Asia South Pacific Design Autom.*, 2005, pp. 645–650.

[55] S. Ray, A. Mishchenko, N. Een, R. Brayton, S. Jang, and C. Chen, "Mapping into LUT structures," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, Mar. 2012, pp. 1579–1584.

[56] P. O. Vontobel, W. Robinett, P. J. Kuekes, D. R. Stewart, J. Straznicky, and R. S. Williams, "Writing to and reading from a nano-scale crossbar memory based on memristors," *Nanotechnology*, vol. 20, no. 42, Oct. 2009, Art. no. 425204.

[57] M. Lin, A. El Gamal, Y.-C. Lu, and S. Wong, "Performance benefits of monolithically stacked 3-D FPGA," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 26, no. 2, pp. 216–229, Feb. 2007.

[58] G. Hills, C. Lau, A. Wright, S. Fuller, M. D. Bishop, T. Srimani, P. Kanhaiya, R. Ho, A. Amer, Y. Stein, D. Murphy, Arvind, A. Chandrakasan, and M. M. Shulaker, "Modern microprocessor built from complementary carbon nanotube transistors," *Nature*, vol. 572, no. 7771, pp. 595–602, Aug. 2019.

[59] I. O'Connor, J. Liu, F. Gaffiot, F. Pregaldiny, C. Lallement, C. Maneux, J. Goguet, S. Fregonese, T. Zimmer, L. Anghel, T.-T. Dang, and R. Leveugle, "CNTFET modeling and reconfigurable logic-circuit design," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 54, no. 11, pp. 2365–2379, Nov. 2007.

[60] M. Simon, A. Heinzig, J. Trommer, T. Baldauf, T. Mikolajick, and W. M. Weber, "Bringing reconfigurable nanowire FETs to a logic circuits compatible process platform," in *Proc. IEEE Nanotechnol. Mater. Devices Conf. (NMDC)*, Oct. 2016, pp. 1–3.

[61] M. Simon, A. Heinzig, J. Trommer, T. Baldauf, T. Mikolajick, and W. M. Weber, "Top-down technology for reconfigurable nanowire FETs with symmetric on-currents," *IEEE Trans. Nanotechnol.*, vol. 16, no. 5, pp. 812–819, Sep. 2017.

[62] J. Zhang, X. Tang, P.-E. Gaillardon, and G. De Micheli, "Configurable circuits featuring dual-threshold-voltage design with three-independent-gate silicon nanowire FETs," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 61, no. 10, pp. 2851–2861, Oct. 2014.

[63] J. Zhang, M. De Marchi, D. Sacchetto, P.-E. Gaillardon, Y. Leblebici, and G. De Micheli, "Polarity-controllable silicon nanowire transistors with dual threshold voltages," *IEEE Trans. Electron Devices*, vol. 61, no. 11, pp. 3654–3660, Nov. 2014.
[64] S. A. Wolf, D. Awschalom, R. Buhrman, J. Daughton, S. Von Molnar, M. Roukes, A. Y. Chtchelkanova, and D. Treger, "Spintronics: A spin-based electronics vision for the future," *Science*, vol. 294, no. 5546, pp. 1488–1495, Nov. 2001.
[65] Z. Jiang, G. Zgheib, C. Y. Lin, D. Novo, Z. Huang, L. Yang, H. Yang, and P. Ienne, "A technology mapper for depth-constrained FPGA logic cells," in *Proc. 25th Int. Conf. Field Program. Log. Appl. (FPL)*, 2015, pp. 1–8.
[66] M. Héder, "From NASA to EU: The evolution of the TRL scale in public sector innovation," *Innov. J.*, vol. 22, no. 2, pp. 1–23, 2017.
[67] A. Vera-Tasama, M. Gomez-Cano, and J. I. Marin-Hurtado, "Memristors: A perspective and impact on the electronics industry," in *Proc. Latin Amer. Electron Devices Conf. (LAEDC)*, vol. 1, Feb. 2019, pp. 1–4.
[68] S. Hamdioui, L. Xie, H. A. D. Nguyen, M. Taouil, K. Bertels, H. Corporaal, H. Jiao, F. Catthoor, D. Wouters, L. Eike, and J. van Lunteren, "Memristor based computation-in-memory architecture for data-intensive applications," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, San Jose, CA, USA, 2015, pp. 1718–1725. [Online]. Available: http://dl.acm.org/citation.cfm?id=2757012.2757210

SHUBHAM RAI (Member, IEEE) received the B.Engg. degree in electrical and electronic engineering and the M.Sc. degree in physics from the Birla Institute of Technology and Science, Pilani, India, in 2011. He is currently pursuing the Ph.D. degree with Technische Universität, Dresden, Germany. His research interests include circuit design for reconfigurable nanotechnologies and their logical applications.

PALLAB NATH received the bachelor's degree in computer science and engineering from the National Institute of Technology Meghalaya and the master's degree in VLSI Design and Nanoelectronics from the Indian Institute of Technology Indore. During his master's degree, in 2018, he was a Guest Researcher at the Chair for Processor Design, TU Dresden, wherein, he worked on developing novel logic leveraging emerging reconfigurable technologies. His research interests include beyond-Moore computing paradigms, and systems, architecture and security exploration for contemporary AI workloads.

ANSH RUPANI received the B.E. degree in electrical and electronics engineering from the Birla Institute of Technology and Science, Pilani, Hyderabad Campus, India, in 2018. He is currently pursuing the M.S. degree in distributed systems engineering with Technische Universität, Dresden, Germany. He is working as a Student Research Assistant at the Chair for Processor Design, TU Dresden. His prior research interests include EDA for emerging reconfigurable technologies and exploiting such technologies for hardware security.

SANTOSH KUMAR VISHVAKARMA (Member, IEEE) received the Ph.D. degree from the Indian Institute of Technology Roorkee, India, in 2010. From 2009 to 2010, he was with the University Graduate Center, Kjeller, Norway, as a Postdoctoral Fellow under European Union Project COMON. He is currently an Associate Professor with the Department of Electrical Engineering, Indian Institute of Technology Indore, India, where he is leading the Nanoscale Devices and VLSI Circuit and System Design Laboratory. His current research interests include nanoscale devices, reliable SRAM memory designs, and configurable circuits design for IoT application.

AKASH KUMAR (Senior Member, IEEE) received the joint Ph.D. degree in electrical engineering in embedded systems from the University of Technology (TUe), Eindhoven, and the National University of Singapore (NUS), Singapore, in 2009. From 2009 to 2015, he was with the National University of Singapore. He is currently a Professor at Technische Universität Dresden (TUD), Germany, where he is directing the Chair for Processor Design. His current research interests include design, analysis, and resource management of low-power and fault-tolerant embedded multiprocessor systems.