# Via-Switch FPGA: 65-nm CMOS Implementation and Evaluation

Xu Bai, *Senior Member, IEEE*, Naoki Banno, *Member, IEEE*, Makoto Miyamura, Ryusuke Nebashi, Koichiro Okamoto, Hideaki Numata, Noriyuki Iguchi, Masanori Hashimoto, *Senior Member, IEEE*, Tadahiko Sugibayashi, Toshitsugu Sakamoto, and Munehiro Tada, *Fellow, IEEE*

***Abstract*— Offering a combination of low latency, high energy-efficiency, and flexibility, field-programmable gate arrays (FPGAs) suit applications ranging from Internet of Things (IoT) computing to artificial intelligence (AI). The conventional static random access memory (SRAM) FPGAs face severe challenges including large standby power and low logic density due to utilization of SRAM cell and MOS switch for signal routing. In response, researchers have introduced emerging non-volatile (NV) memory technologies to solve standby power issues. However, access transistors used for NV memory cell configuration still consume a large silicon area. In this article, we introduce an NV via-switch (VS) FPGA featuring fully back-end-of-line (BEOL) signal routing and front-end-of-line (FEOL) logic computing for high logic density. The VS fabricated in BEOL is constructed by two Cu atom switches (ASs) for signal routing and two a-Si/SiN/a-Si varistors for AS configuration. We demonstrate the first implementation of the VS-FPGA at 65-nm node and evaluate its performance by various basic applications. 2.6× logic density, 1.5× energy efficiency, and 1.4× operation speed are achieved in comparison with a previous complementary AS (CAS) FPGA in which one access transistor is necessary for each CAS configuration.**

***Index Terms*— Atom switch (AS), cross-point, field programmable gate array (FPGA), nonvolatile (NV), programmable logic, resistive random access memory (RRAM), via-switch (VS).**

## I. Introduction

FIELD-PROGRAMMABLE gate array (FPGAs) are playing an important role in both cloud and edge computations of the Internet of Things (IoT). FPGAs are being deployed as accelerators in data center infrastructure to construct a configurable cloud [1], [2], introduced in various sensors for real-time processing [3], [4], and used for *in situ* artificial intelligence (AI) inference and training in edge devices [5], [6].

A conventional static random access memory (SRAM)-FPGA consists of routing blocks (RBs), logic blocks (LBs), SRAM blocks and an external nonvolatile memory (NVM) block as shown in Fig. 1(a). SRAM cells typically composed of six transistors are programmed to implement logic functions in look-up tables (LUTs) and control MOS switches (pass transistors, transmission gates, or multiplexers) to steer interconnect signals for various applications as shown in Fig. 1(b) [7]. The area of the RBs and SRAM blocks is about four times larger than that of the LBs [8]. Due to this, the chip size of the SRAM-FPGA becomes 35 times larger than that of an application-specific integrated circuit (ASIC) [9]. MOS switches and long interconnection wires result in 3.4–4.6 times lower operation speed than an ASIC [9] and consume 62% more dynamic power [10]. The high leakage power of the SRAM has become another critical issue in the SRAM-FPGA [10]. Moreover, the SRAM-FPGA is volatile and needs to reload their configuration data from the external NVM to the internal SRAM every time after it is powered up [11].

To address the issues concerning the SRAM-FPGA, as shown in Fig. 1(c), researchers started to replace the SRAM by emerging embedded NVM including phase-change memory (PCM) [12], spin-transfer-torque magnetic random access memory (STT-MRAM) [13]–[15], and resistive random access

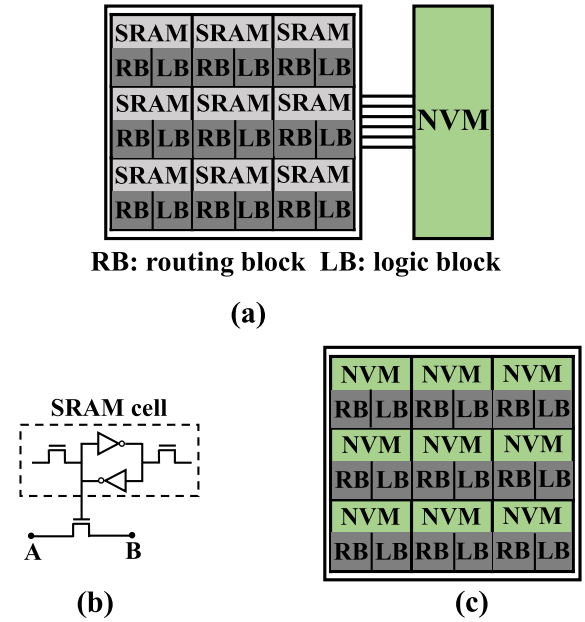



Fig. 1. Structures of conventional SRAM- and NVM-FPGAs. (a) Structure of a conventional SRAM-FPGA [7]. (b) Schematic of an SRAM routing switch [7]. (c) Structure of an NVM-FPGA [12]–[34].

Manuscript received July 3, 2021; revised September 11, 2021; accepted September 26, 2021. Date of publication October 13, 2021; date of current version June 29, 2022. This article was approved by Associate Editor Vivek De. This work was supported by the Core Research for Evolutional Science and Technology (CREST), Japan Science and Technology Agency (JST), under Grant JPMJCR1432. *(Corresponding author: Xu Bai.)*

Xu Bai, Naoki Banno, Makoto Miyamura, Ryusuke Nebashi, Koichiro Okamoto, Hideaki Numata, Noriyuki Iguchi, Tadahiko Sugibayashi, Toshitsugu Sakamoto, and Munehiro Tada are with NanoBridge Semiconductor, Inc., Tsukuba, Ibaraki 305-0047, Japan (e-mail: x-bai@nanobridgesemi.com).

Masanori Hashimoto is with the Graduate School of Informatics, Kyoto University, Kyoto 606-8501, Japan.

Color versions of one or more figures in this article are available at https://doi.org/10.1109/JSSC.2021.3117260.

Digital Object Identifier 10.1109/JSSC.2021.3117260

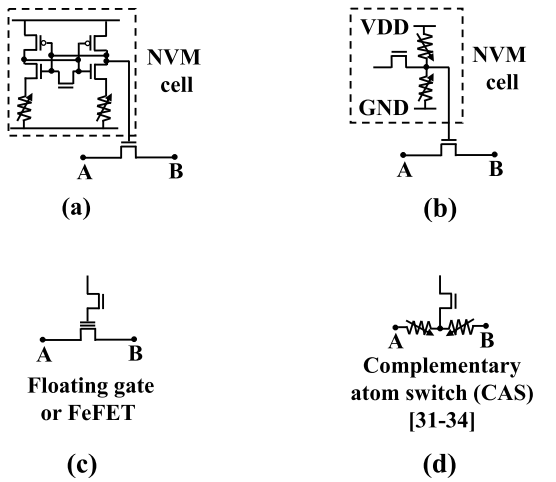



Fig. 2. NV routing switches. (a) Routing switch using an NVM cell in an SRAM-like sensing structure [12]–[16]. (b) Routing switch using a voltage-divider-based NVM cell [17]–[20]. (c) Routing switch using an NV programmable transistor including a floating gate [21]–[23] or an FeFET [24]. (d) Routing switch using an NV programmable switch [25]–[33], a CAS is shown as an example.

memory (RRAM) [16]. Fig. 2(a) shows a routing switch using two NVM cells embedded in an SRAM-like sensing structure. Instant power-on directly using configuration data stored in NVM cells achieves almost zero standby power. However, the additional transistors for providing programmability set limits for advanced scaling.

To maximize the advantages of the non-volatile (NV) routing switch, Ahari *et al.* [17], Tanachutiwat *et al.* [18], Chen *et al.* [19], and Liauw *et al.* [20] introduced a voltage-divider-based configuration cell shown in Fig. 2(b). Liauw *et al.* [20] reported that the voltage-divider RRAM-FPGA achieved 40% smaller die area and 28% lower energy-delay product thanks to shorter interconnection wires. Other researchers have attempted to replace the SRAMs and MOS switches with programmable transistors including floating gates [21]–[23] and ferroelectric field-effect transistors (FeFET) [24] as shown in Fig. 2(c). However, the transistors with high resistance and capacitance on the routing path still result in low operation speed and high dynamic power consumption.

To overcome the problems originating from routing transistors, RRAMs and atom switches (ASs) are directly introduced as routing switches for signal transfer control in [25]–[27] and [28]–[34], respectively. The AS is an NV resistive switch fabricated between back-end-of-line (BEOL) Cu layers and has very small capacitance (∼1/10 of a transistor), low ON-state resistance (∼500 Ω), and high OFF-state resistance (∼250 MΩ) [28]. Its OFF/ON resistance ratio is much larger than the RRAM whose ON-state and OFF-state resistances are ∼10 KΩ and ∼1 MΩ, respectively [24]. The AS-FPGA is first fabricated and presented in [29]. In addition, a complementary AS (CAS), which is composed of two ASs in series as shown in Fig. 2(d), reduces the programming voltage to 2 V and improves OFF-state reliability [30]. The CAS-FPGAs have been fabricated to evaluate area and performance [31]–[34] and achieved 78% area reduction compared with a conventional SRAM-FPGA [31].

On the other hand, the switch footprint has room for improvement. The RRAM routing switch adopts two-transistor-one-RRAM (2T1R) structure [25] or four-transistor-one-RRAM (4T1R) structure [26], [27]. Either the AS or the CAS also requires an access transistor for configuration control. Even though the switch itself has a small footprint and is integrated above the CMOS layer, the access transistors occupy the additional area.

To further exploit the advantage of the NV routing switch, integrating selectors between BEOL metal layers is studied to replace the access transistors for dramatical improvement of logic density [35]–[39]. For pursuing further efficient FPGA implementation, we have proposed a two-varistor one-CAS (2V-1CAS) structure named "via-switch" (VS) [40]–[42] to obtain high functionality with multiple cross-point programming per column or row (multiple fan-outs) of the crossbar switch. The multiple fan-outs functionality facilitates the signal routing and improves its quality in application mapping. As for the FPGA chip design, the CMOS layer under the VS has no layout restrictions, and hence the layout design flexibility improves significantly, which is another advantage in addition to the small footprint.

Previously, we have reported the first implementation of the VS-FPGA [43] and its performance evaluation [44]. This article provides the details of the design of the fabricated VS-FPGA, including device characteristics of the VS, circuit schematics, the explanation of the area efficiency, and discusses evaluation results of the VS-FPGA using additional measurement data.

The remainder of this article is organized as follows. Section II introduces the characteristics of the AS, the varistor, the VS and the VS crossbar switch. Section III describes the circuit schematics of the VS-FPGA and demonstrates its area efficiency. Section IV provides evaluations of the VS-FPGA. Conclusions are presented in Section V.

## II. VIA-SWITCH

### A. Atom Switch

The AS is a kind of cation-type electrochemical NV resistive-change devices [28]. A polymer-solid electrolyte (PSE) is sandwiched between an active (Cu) anode and an inert (Ru) cathode as shown in Fig. 3. When a positive voltage VSET is applied to the Cu electrode, $Cu^+$ is extracted to form a Cu bridge, and the AS is turned on, which is called set operation. On the other hand, when a positive voltage VRESET is applied to the Ru electrode, the Cu bridge is removed in the electrolyte, and the AS is turned off, which is called reset operation. The set and reset operations are repeatable over 1000 times, and both the ON and OFF states are NV.

In the AS-FPGA introduced in [28], an AS crossbar switch was used for signal transfer control. Fig. 4 shows a 2 × 2 AS crossbar switch with two input and two output terminals. The AS located at each crosspoint is turned on or off to control signal transfer from two input terminals to two output terminals. Input signals *A* and *B* are applied to the two input terminals via two buffers, respectively. Let us explain what

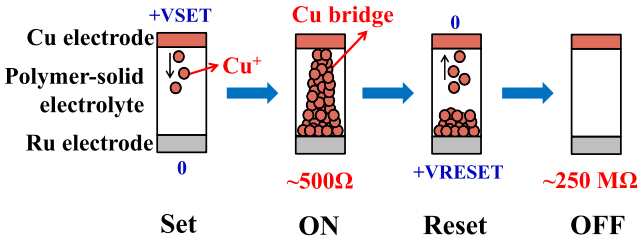



Fig. 3. Set/reset operations of an AS.

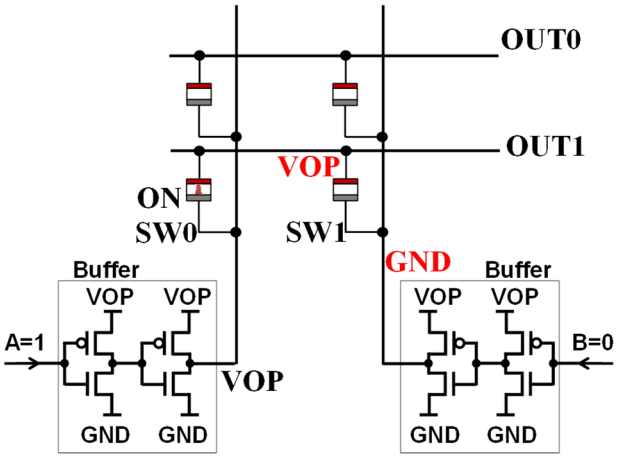



Fig. 4. OFF-state reliability issue in an AS crossbar switch.

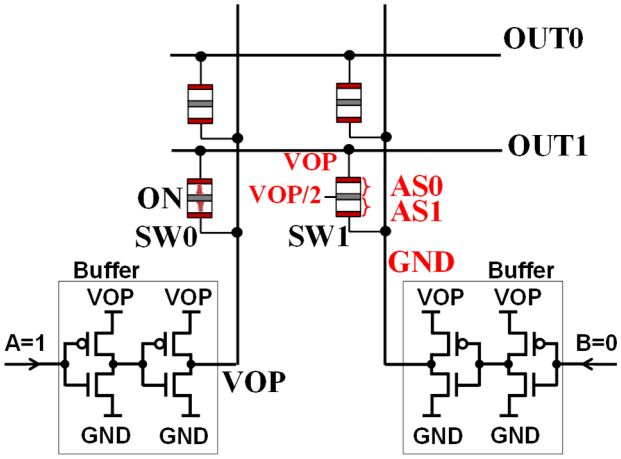



Fig. 5. OFF-state reliability improvement in a CAS crossbar switch.

the problem happens by using a single AS for signal transfer control. If $A$ is "1," $B$ is "0," and the AS SW0 is at ON state, a logic operation voltage VOP and a ground voltage GND are applied to the Cu and Ru electrodes of the OFF-state AS SW1, respectively. According to the off-to-on mechanism of Cu drift in dielectrics [45], even though the voltage level of VOP is much smaller than VSET, the OFF-state SW1 will change into an ON-state AS slowly.

To secure a longer OFF-state lifetime at VOP, we have proposed the CAS composed of two ASs connected in series with opposite direction [30] shown in Fig. 5. In a 2 × 2 CAS crossbar switch, if $A$ is "1," $B$ is "0," the CAS SW0 is at ON-state, VOP and GND are applied to the two Cu electrodes

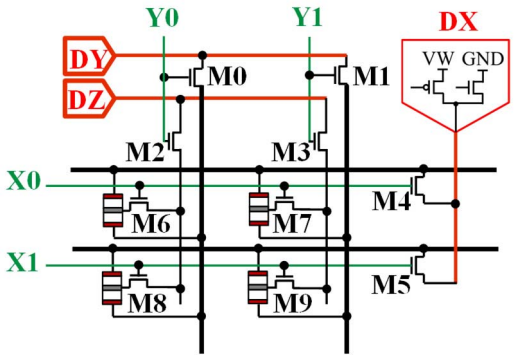



Fig. 6. Previous 1T1CAS crossbar switch with programming drivers.

of the OFF-state CAS SW1, respectively. VOP is divided by AS0 and AS1 in SW1, and therefore, the voltage difference between the two terminals of AS0 or AS1 is VOP/2. The set bias stress voltage is decreased, which contributes to a higher OFF-state reliability of AS0 than that of the single AS SW1 in Fig. 4. Moreover, VOP/2 between the two terminals of AS1 is a reset bias voltage. Therefore, the AS1 maintains the OFF state until time-dependent dielectric breakdown of the electrolyte. Thus, OFF-state lifetime can be dramatically improved and kept more than ten years at 125 °C.

### *B. Varistor*

Let us explain the role of the access transistor in the one-transistor-one-CAS (1T1CAS) structure shown in Fig. 2(d) and the required $I$–$V$ characteristics of the varistor for the access transistor replacement.

Fig. 6 shows a previous 2 × 2 1T1CAS crossbar switch using the access transistors. Peripheral programming drivers DX, DY, and DZ are necessary for programming a target CAS. They are applied to three terminals of a CAS via access transistors M0–M9. The programming driver can supply a programming voltage VW (=VSET or VRESET), GND, or high impedance (HZ). Row address signals (X0, X1) and column address signals (Y0, Y1) control surrounding access transistors M0–M5 at the boundary of the crossbar switch and crosspoint access transistors M6–M9 at each crosspoint to access a target CAS.

The varistors are used to replace the crosspoint access transistors M6–M9. In a programming operation, VW is applied to a target CAS and M6–M9 controlled by X0 and X1 provide programming current. In an application operation, the address signals X0 and X1 are set to "0" to turn off M6–M9 for isolating the sneak paths between CASs on the same column when a data signal at voltage level VOP is transferred through an ON-state CAS. To act in the same role, the varistor should be a nonlinear selector device that provides programming current in the programming operation at the high programming voltage VW and isolates connections between CASs in the application operation at the low logic operation voltage VOP.

Banno *et al.* [40]–[42] introduce the varistor with a-Si/SiN/a-Si stacking layers achieves a high nonlinearity performance of $\sim 10^5$ and has the advantage of high compatibility with a CMOS process. Fig. 7 shows $I$–$V$ characteristics

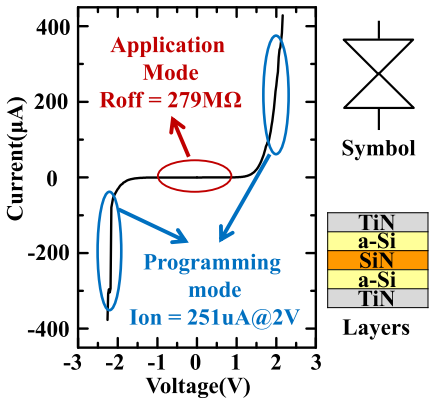



Fig. 7. *I–V* characteristics of an a-Si/SiN/a-Si varistor.

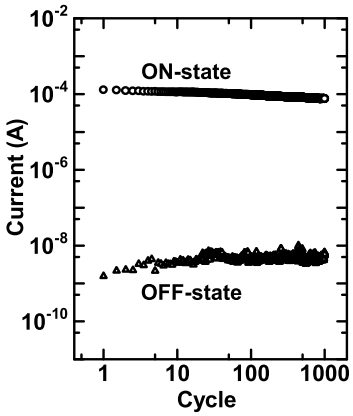



Fig. 8. ON/OFF current characteristics of varistor for $10^3$ cycles.

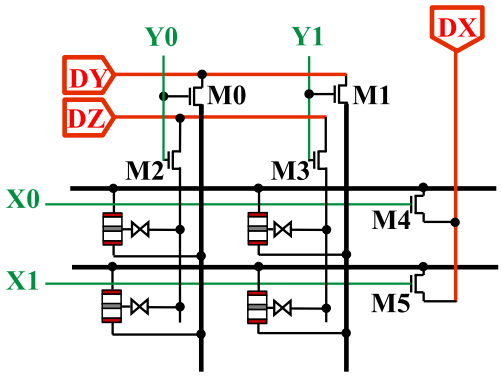



Fig. 9. 1V1CAS crossbar switch with programming drivers.

of the varistor in a 65-nm node BEOL on a 300-mm wafer, which exhibits ON current of 251 $\mu$A at 2 V and OFF resistance of 270 MΩ. Fig. 8 shows the endurance characteristic of the varistor. The high ON/OFF current ratio is confirmed for 1000 cycles, which is enough for the FPGA application.

### C. VS Crossbar Switch

Varistors can directly replace all the crosspoint access transistors for area reduction. However, an address selectivity problem occurs in a programming operation, which causes a programming sneak path problem. Fig. 9 shows a 2 × 2 one-varistor one-CAS (1V1CAS) crossbar switch as an example to explain the programming sneak path problem of the 1V1CAS

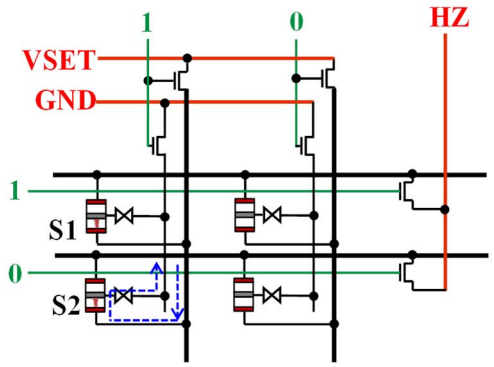



Fig. 10. Programming sneak path in 1V1CAS crossbar switch.

structure. In the previous 2 × 2 1T1CAS crossbar switch as shown in Fig. 6, each CAS is connected to DZ via a crosspoint access transistor controlled by a row address signal and a surrounding access transistor controlled by a column address signal. However, in the 1V1CAS crossbar switch, each CAS is connected to DZ via a varistor, and a surrounding access transistor controlled by a column address signal. Since the varistor is not controlled by a row address signal, all the CASs on the same column are accessed in one programming operation as shown in Fig. 10. To set the AS S1, both (X0, X1) and (Y0, Y1) are set to (1, 0), and DX, DY, and DZ are set to HZ, VSET, and GND, respectively. S2 is also on the set path so that both S1 and S2 are turned on.

To overcome the above programming sneak path problem in the 1V1CAS crossbar switch, the VS structure [40]–[42] was proposed to provide both row and column address selectivity by adding surrounding access transistors and a programming driver in a VS crossbar switch. As shown in Fig. 11(a), different from the 1V1CAS crossbar switch shown in Fig. 9, the middle terminal of a CAS is connected to two varistors at each crosspoint, and two programming drivers DZX and DZY are used to program two ASs in one CAS, respectively. DZX and DZY are connected to the two varistors via existing surrounding access transistors M2 and M3 and added surrounding access transistors M6 and M7. (X0, X1) and (Y0, Y1) control M6 and M7 and M2 and M3, respectively, to make sure that only one VS is selected. Moreover, one of the programming drivers DX and DZX (DY and DYX) must be set to HZ to avoid programming sneak path on the same row (column).

In a programming operation, two steps are necessary for setting/resetting the CAS with two ASs S0 and S1 as shown in Fig. 11(b)–(e). Both (X0, X1) and (Y0, Y1) are set to (1, 0) to apply the programming drivers to S0 and S1. In a set operation of S0 shown in Fig. 11(b), DX, DY, DZX, and DZY are set to VSET, HZ, HZ, and GND, respectively, to form a Cu bridge in S0. In a set operation of S1 shown in Fig. 11(c), DX, DY, DZX, and DZY are set to HZ, VSET, GND, and HZ, respectively, to form a Cu bridge in S1. On the other hand, in a reset operation of S0 shown in Fig. 11(d), DX, DY, DZX, and DZY are set to GND, HZ, HZ, and VRESET, respectively, to remove the Cu bridge in S0. In a reset operation of S1 shown in Fig. 11(e), DX, DY, DZX, and DZY are set to HZ, GND, VRESET, and HZ, respectively, to remove the Cu bridge in S1.

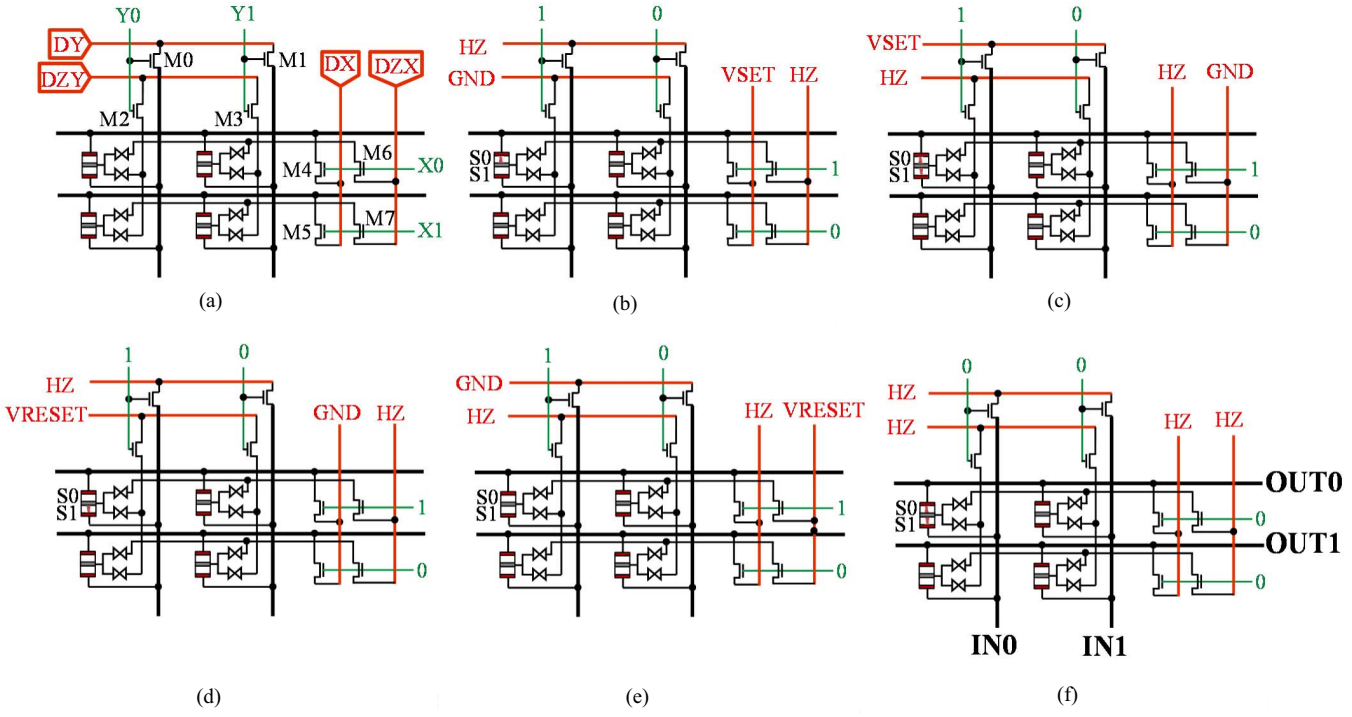



Fig. 11. VS crossbar switch with programming drivers and its programming (set/reset) and application operations. (a) 2V1CAS crossbar switch. (b) Set operation of S0. (c) Set operation of S1. (d) Reset operation of S0. (e) Reset operation of S1. (f) Application operation.

In an application operation shown in Fig. 11(f), the address signals are set to "0" to isolate the signal transfer path from the programming drivers. DX, DY, DZX, and DZY are set to HZ for avoiding source-drain leakage current of surrounding access transistors. Varistors are at a high-resistance state to isolate connections between CASs. S0 and S1, which are at ON state, provide a signal transfer path IN0→OUT0.

Fig. 12 shows a cross-sectional TEM image of a VS which is fabricated between Cu metal layers M4 and M5 and above a CMOS layer. Two series-connected TiN/a-Si/SiN/a-Si/TiN varistor and Ru-alloy/PSE/Cu AS are clearly separated from each other. The VS occupies $48F^2$ whereas its footprint can be reduced to $18F^2$ if four metal layers are used for VS implementation [42]. The VS can be fabricated between any adjacent Cu metal layers. For example, the VS is fabricated between M1 and M2 in [40]–[42] to evaluate its characteristics. In the FPGA application introduced in [43] and [44], M1–M3 are used for CMOS circuits, and therefore the VS is fabricated between M4 and M5.

In research [42], the VS was fabricated in a 65-nm node BEOL to test set and reset operation. Fig. 13 shows the set/reset $I$–$V$ characteristics of a single side of the integrated VS. In set operation of the AS S0, VSET and GND are applied to T0 and T2, respectively. When VSET is increased to 3 V, the Cu bridge is formed in the AS S0 which is turned on. The resistance RON of the ON-state AS S0 is determined by the ON current of the varistor since AS and varistor are connected in series. On the other hand, in the reset operation of S0, GND and VRESET are applied to T0 and T2, respectively. When VRESET is increased to 3 V, the Cu bridge is cutoff, and S0 is

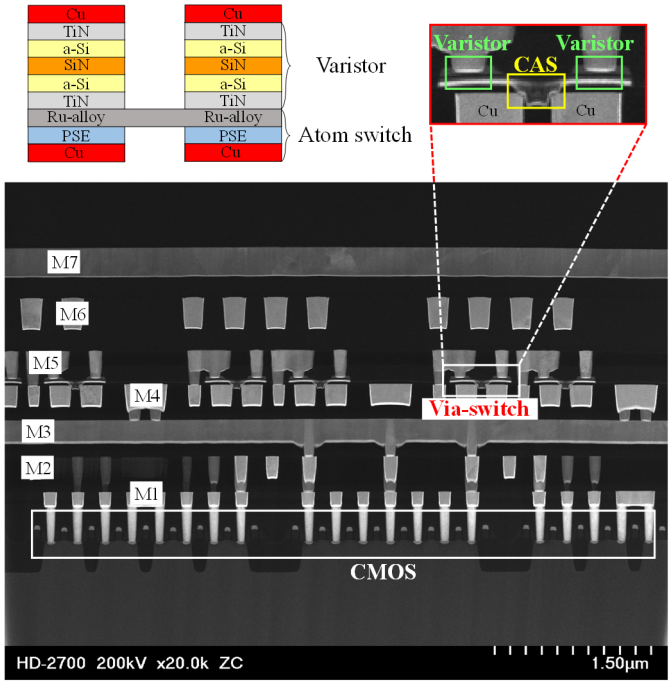



Fig. 12. Cross-sectional TEM image of a VS.

turned off. The retention characteristics depending on the AS have been confirmed in [46]. No failure is observed in the ON-state ASs for 1 h at 260 °C and for 3000 h at 150 °C.

Fig. 14 shows ON- and OFF-state $I$–$V$ characteristics of a VS with leakage current between varistors. In an application operation, $\sim 10^4$ ON/OFF current ratio is large enough to

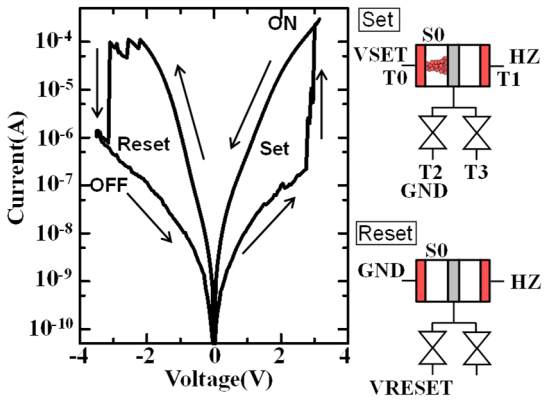



Fig. 13. Set/reset $I$–$V$ characteristics of a VS single side.

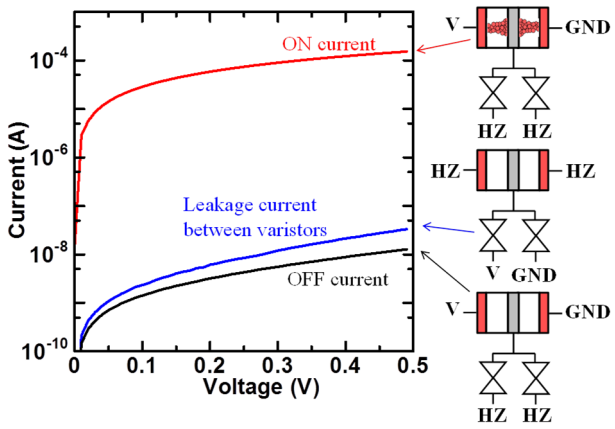



Fig. 14. ON- and OFF-state $I$–$V$ characteristics of a VS with leakage current between varistors.

control a data signal transfer in a VS crossbar switch. It should be noted that the varistor is still under development to improve the nonlinearity for higher ON/OFF current ratio and lower leakage current between varistors.

Next, let us introduce the advantages obtained from the replacement of the access transistors by the varistors. We need to use high voltage (HV) transistors (usually I/O transistors with high breakdown voltage) as the access transistors since VW (around 2V [33]) applied to them is much higher than VOP. In the previous 2 × 2 1T1CAS crossbar switch shown in Fig. 6, the count of the surrounding access transistors M0–M5 is larger than that of the crosspoint access transistors M6–M9. Practically, in a 65-nm CAS-FPGA to implement simple applications such as a 4-bit counter and a 2-bit adder [31], a 19 × 16 1T1CAS crossbar switch requires 304 crosspoint access transistors which are more than 54 surrounding access transistors. Thanks to the replacement of the crosspoint access transistors by the varistors, the access transistor count of the 19 × 16 VS crossbar switch is reduced from 358 to 70 in comparison with the 19 × 16 1T1CAS crossbar switch [31]. The crosspoint access transistor count increases dramatically with the 1T1CAS crossbar switch size increasing. Therefore, their replacement by the varistors leads to further improvement of area efficiency, operation speed, and power efficiency for the large-scale FPGA application.

In Section III, we will introduce the structure of the 65-nm VS-FPGA using the VS crossbar switch [43], [44] in detail and its superiority over the CAS-FPGA using the 1T1CAS crossbar switch in the same 65-nm node [31], [32].

## III. Via-Switch FPGA

### A. Architecture of a VS-FPGA

As shown in Fig. 15, the VS-FPGA is constructed by a 6 × 6 cell array with its periphery circuits including a row decoder, a column decoder, and programming drivers. A cell (such as cell A) is connected to four adjacent cells and their next four cells (eight cells in total) in four directions through unidirectional wires which are superior to bidirectional wires in terms of area, delay, and area-delay product [47]. Row and column address signals are generated by the row and column decoders, respectively, and they control access transistors in each cell to apply the programming drivers DX, DY, DZX, and DZY shown in Fig. 11 to a target VS. The programming drivers are shared for programming the VSs in all the cells, and therefore their area overhead is much smaller than that of the cell array.

Each cell includes a configurable LB, a switch multiplexer (SMUX), and an input multiplexer (IMUX). The LB composed of two four-input LUTs (4-LUTs), two D-flip-flops (DFFs), and two multiplexers (MUXs) implements a logic circuit including a combinational circuit or a sequential circuit. The SMUX is used for signal routing from the last cells to the next cells. On the other hand, the IMUX is used for signal routing from the last cells to the LB.

A VS crossbar switch is used in the SMUX, IMUX, and 4-LUT. Sixteen wires from eight cells and two wires from the LB are connected to the inputs of the SMUX and IMUX. To avoid the through current in the CMOS gates connected to the outputs of the SMUX and IMUX, GND is also applied to the SMUX and IMUX. As a result, either SMUX or IMUX has nineteen inputs in total. The SMUX provides eight outputs to four directions, and the IMUX's eight outputs are applied to the two 4-LUTs. Therefore, the SMUX and IMUX are constructed by a 19 × 16 VS crossbar switch. The 4-LUT consists of a 16:1 MUX and a 2 × 16 VS crossbar switch. The 2 × 16 VS crossbar switch is used as a 16-bit memory array to store logic configuration data. One input is connected to VOP, and the other one is connected to GND. Two VSs on one row form a memory cell. In case that the VS connected to VOP is turned on and the VS connected to GND is turned off, a logic value "1" is stored in the memory cell. Conversely, a logic value "0" is stored. If both VSs are turned off, the memory cell provides HZ. To avoid the through current from VOP to GND, both VSs cannot be turned on simultaneously. Users can configure the sixteen memory cells to implement an arbitrary four-input logic operation in one 4-LUT.

### B. VS-FPGA Cell Versus CAS-FPGA Cell

In this section, we will introduce the VS-FPGA cell using the VS crossbar switch and demonstrate its area efficiency in comparison with the previous CAS-FPGA cell.

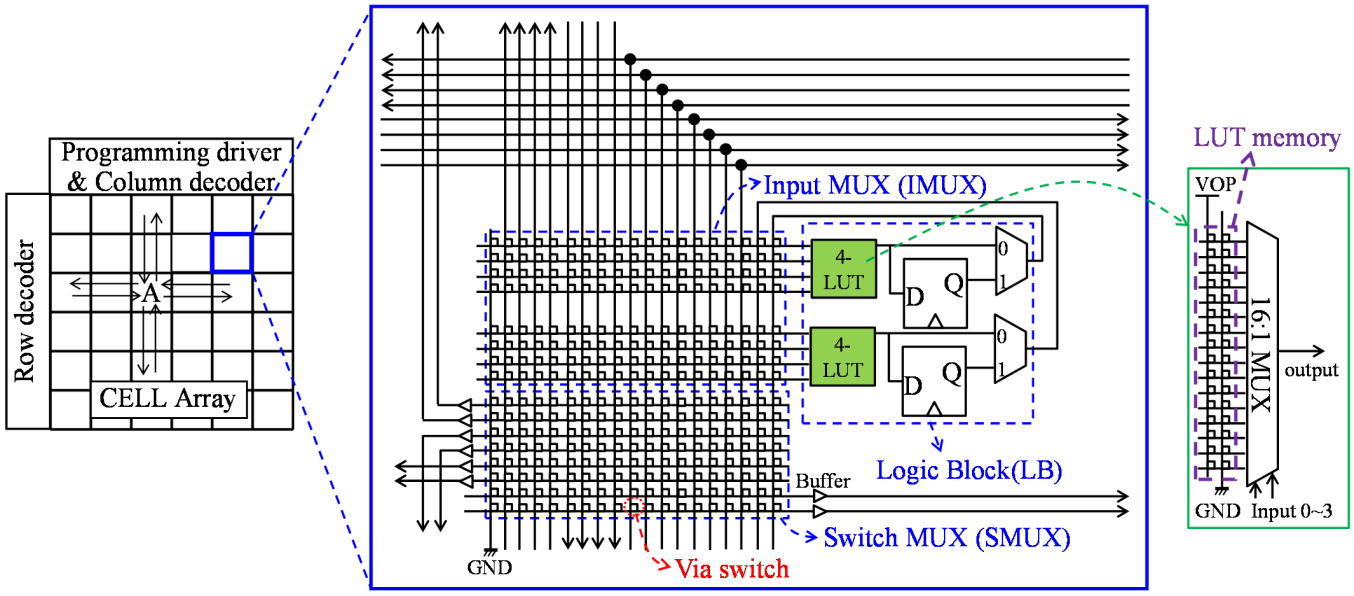



Fig. 15. Architecture of the VS-FPGA.

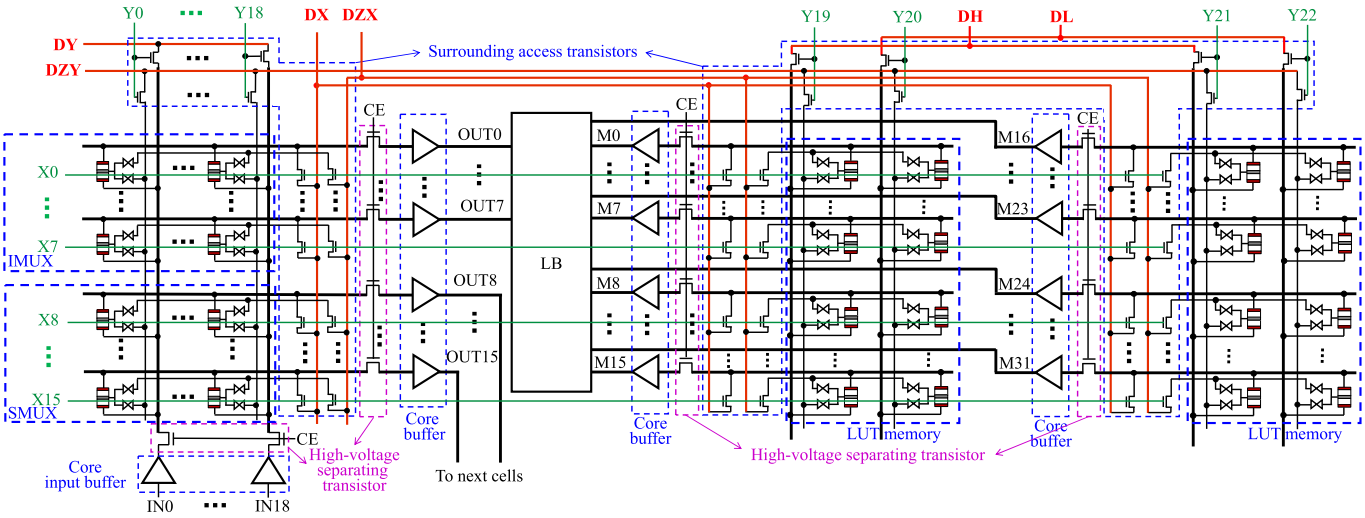



Fig. 16. VS-FPGA cell using VS crossbar switches.

Let us show how to use the VS crossbar switch in the VS-FPGA cell. As shown in Fig. 16, the VS-FPGA cell has a 19 × 16 crossbar switch used as IMUX + SMUX and two 2 × 16 crossbar switches used as LUT memory arrays. In a programming operation, programming drivers DX, DY, DZX, DZY, DH, and DL are applied to CASs via surrounding access transistors controlled by row address signals X0–X15 and column address signals Y0–Y22. In an application mode, DH and DL applied to the two 2 × 16 crossbar switches are set to VOP and GND, respectively, and their connected surrounding transistors are turned on to apply VOP and GND to each memory for providing configuration data to 4-LUTs.

The inputs IN0–IN18 mentioned in Section III-A are connected to the 19 × 16 crossbar switch via HV separating transistors for isolating the core input buffers from the crossbar switch in the programming operation. An enable signal CE used to control the HV separating transistors is set to high in an application operation to enable signal transfer between cells and low in a programming operation to avoid collision of core input buffers and the programming driver DY, respectively. The outputs of the 19 × 16 crossbar switch and two 2 × 16 crossbar switches are connected to HV separating transistors since the voltage level of their inputs becomes VW in the programming operation which is higher than the breakdown voltage of the core transistor.

Fig. 17 shows the previous CAS-FPGA cell using the 1T1CAS crossbar switch. The difference between the VS-FPGA cell and the previous CAS-FPGA cell is that one more programming driver DZX is provided to avoid the programming sneak path mentioned in Section II, and the horizontal surrounding access transistors are increased twice. However, thanks to the replacement of the crosspoint

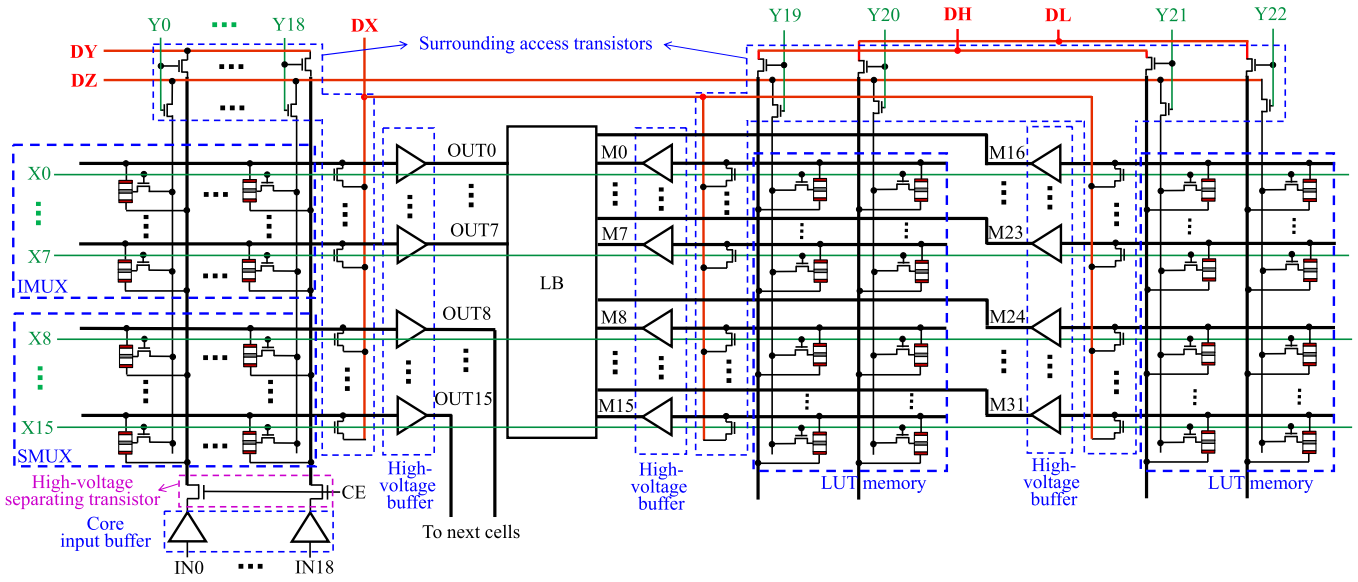



Fig. 17. Previous CAS-FPGA cell using 1T1CAS crossbar switches.

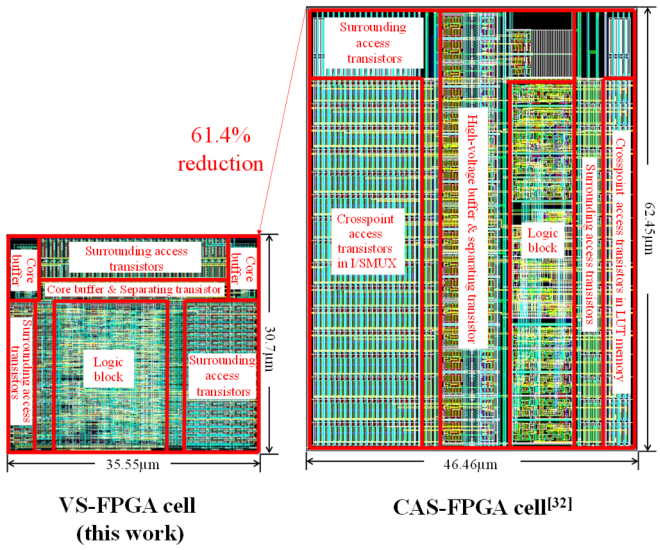
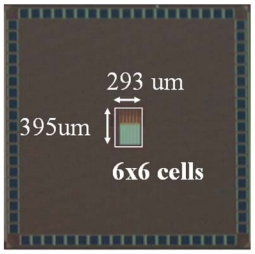



Fig. 18. Area comparison by layouts of the VS-FPGA and previous CAS-FPGA cells.




| Technology | 65-nm CMOS (for FEOL and M1-M4) + Post process (for VS and M5-M7) |
|---|---|
| # of cells | 6 x 6 |
| One cell | 35.55 x 30.7um² |
| VS size | 8x6 F² |

Fig. 19. Die micrograph and specification of VS-FPGA.

access transistors used in the previous CAS-FPGA cell by the varistors in BEOL, the total access transistor count is reduced by 69% (462 → 142). Moreover, we use the HV separating transistors and core buffers to replace the HV buffers connected to the outputs of the 19 × 16 crossbar switch and two 2 × 16 crossbar switches for area reduction.

As reported in [31], the CAS-FPGA cell achieved 78% area reduction compared with an equivalent SRAM-FPGA cell. Let us compare the areas of the CAS-FPGA and VS-FPGA cells by their layouts shown in Fig. 18. 55% area is consumed by the crosspoint access transistors in the CAS-FPGA cell. In the VS-FPGA cell, the crosspoint access transistors are replaced by the varistors fabricated above the LB, core buffers, HV separating transistors, and surrounding access transistors. Therefore, no area is consumed by the crosspoint access transistors, and the floorplan is changed. Moreover, high-voltage buffers are replaced by HV separating transistors and core buffers, which leads to further area reduction. As a result, the area of the VS-FPGA cell is reduced by 61.4% in comparison with that of the CAS-FPGA cell. Furthermore, its area is reduced to 8.3% compared with the SRAM-FPGA cell [43]. Logic density is calculated by the number of 4-LUTs per area. The VS-FPGA achieves 2.6× and 12× logic density of the CAS-FPGA and SRAM-FPGA, respectively.

## IV. Evaluation of VS-FPGA

Fig. 19 shows the die micrograph and specifications of a chip fabricated in 65-nm CMOS, where front-end-of-line (FEOL) and M1–M4 are fabricated in a commercial fab and VS, M5, M6, and M7 (semi-global) are processed by ourselves. The die includes 6 × 6 cells, peripheral circuits including a controller, programming drivers, and address decoders

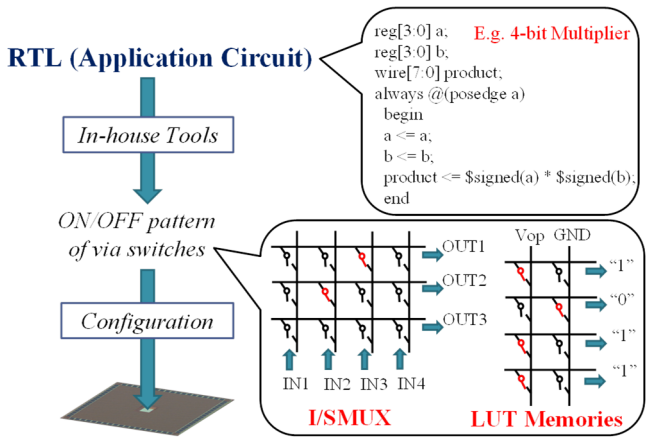



Fig. 20. VS-FPGA design flow.

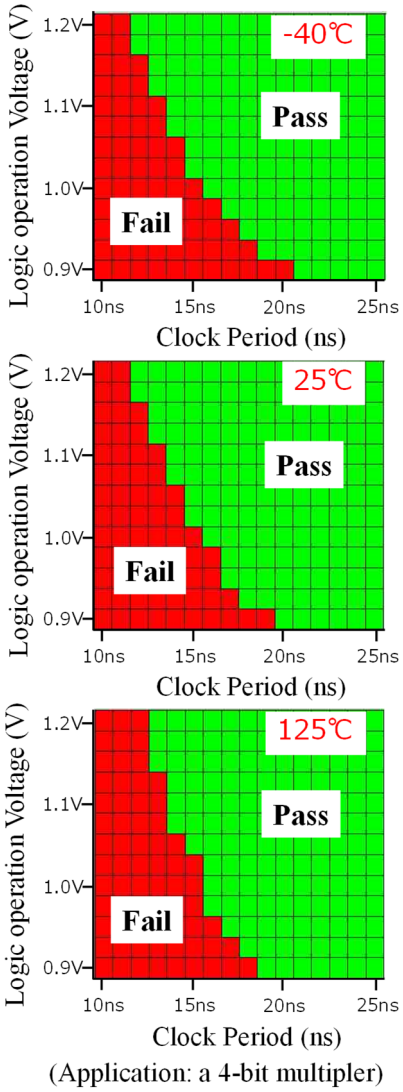



Fig. 21. Measured Shmoo plots of a 4-bit multiplier implemented on the fabricated VS-FPGA at −40 °C, 25 °C and 125 °C.

in 293 × 395 $\mu$m$^2$. Note that the area of peripheral circuits is negligible for larger cell arrays. It is estimated that the area of peripheral circuits is less than 1% of the whole area in the case of a 1000 × 1000 cell array.

Fig. 20 shows a flow to implement an application circuit in the VS-FPGA using in-house tools including LUT mapping, packing, placement, routing, and bitstream generation tools. The application circuit is described in terms of registers and logic operations called register-transfer level (RTL). The RTL file passes through technology-independent logic optimization (Synthesis) by a Synopsys design compiler (DC) tool. The LUT mapping tool maps the gate-level netlist generated by the Synopsys DC tool into LUTs. The packing tool which has the same algorithm with the conventional T-VPACK (timing driven packing) tool [48] clusters LUTs and DFFs together into LBs. The placement tool developed based on the simulated annealing algorithm [48] implements timing driven location of LBs. The routing tool developed based on the pathfinder algorithm [48] optimizes data transfer path for a small delay time. The bitstream generation tool generates a bitstream file that contains VS ON/OFF information for VS-FPGA configuration. Finally, VSs are configured by controlling programming drivers and address signals according to the generated VS ON/OFF bitstream.

We measure the fabricated VS-FPGA under automotive temperature grade (−40 °C–125 °C) [49] to demonstrate high reliability of the VS-FPGA in a harsh environment. An area-minimized 4-bit multiplier (MPY4) is implemented on the fabricated VS-FPGA with VS programming. Fig. 21 shows Shmoo plots of the MPY4 on the fabricated VS-FPGA at −40 °C, 25 °C (room temperature), and 125 °C, respectively. The plots confirm the correct operation of the MPY4 at logic operation voltage range 0.9–1.2 V under the automotive temperature grade. It achieves 83-MHz operation at the standard supply voltage 1.2 V of 65-nm technology node and the room temperature 25 °C.

In the SRAM-FPGA, the ON resistance variation of the MOS switch used for signal transfer control affects the signal delay in high temperatures because the electron transport suffers from phonon scattering at high temperature. However, the Cu bridge formed in the ON-state VS is almost not affected by temperature [50]. The research in [51] evaluates the impact of temperature on the delay of the SRAM-FPGA. Its delay increases up to 47% for 0 °C→100 °C. The delays at −40 °C, 25 °C, and 125 °C of the MPY4 on the fabricated VS-FPGA at 1.2 V are shown in Fig. 22. The delay of the VS-FPGA increases up to only 7% for −40 °C→125 °C thanks to the replacement of the MOS switch by the VS for signal transfer control.

To demonstrate the performance advantages of the VS-FPGA, we evaluate the energy per cycle, delay, and energy-delay product (EDP) of the VS-FPGA and previous CAS-FPGA by implementing three basic applications including a 16-bit counter (CNT16), a 24-bit linear feedback shift register (LFSR24), the MPY4 as shown in Fig. 23. The EDP is a metric that combines measures of energy and delay [52]. The CNT16 and MPY4 are evaluated at the standard supply voltage 1.2 V of 65-nm node technology. The LFSR24 is

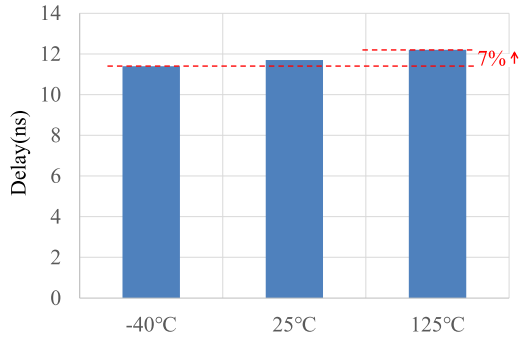



Fig. 22. Impact of temperature on the delay of the MPY4 implemented on the fabricated VS-FPGA.

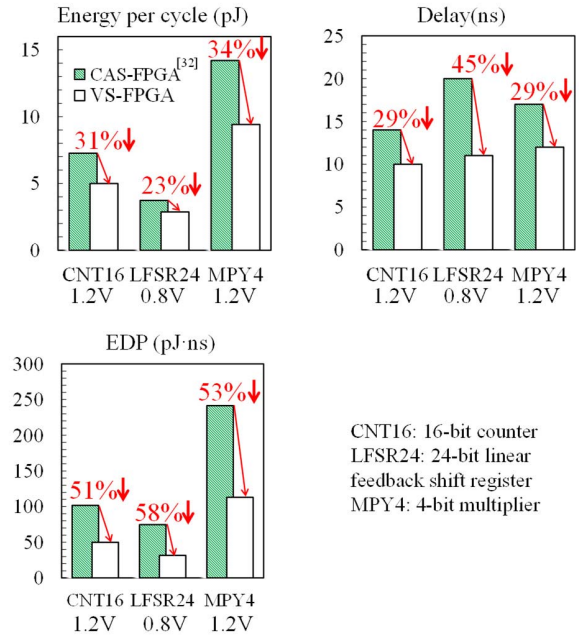



Fig. 23. Energy per cycle, delay, and energy-delay product (EDP) comparisons between CAS- and VS-FPGAs.

TABLE I

SUMMARY OF PERFORMANCE COMPARISON

| | CAS-FPGA[32] | | VS-FPGA (This work) | | SRAM-FPGA[32] |
|---|---|---|---|---|---|
| Switch | CAS + Access Tr. | | Via switch | | SRAM + Pass Tr. |
| Process node | 65 nm | | 65 nm | | 65 nm |
| Cell Area (um$^2$) | 2,827 | 61%↓ | 1,091 | 92%↓ | 13,144 |
| Logic operation voltage(V) | 1.0 | | 1.0 | | 1.0 |
| Delay(ns) | 21 | 24%↓ | 16 | 80%↓ | 80 |
| Energy per cycle (pJ) | 8.33 | 25%↓ | 6.27 | 70%↓ | 20.7 |
| EDP(pJ·ns) | 175 | 45%↓ | 97 | 94%↓ | 1659 |

(Application: a 4-bit multiplier)

evaluated at 0.8 V due to the delay measurement limit of the equipment that we used. In comparison with the previous CAS-FPGA, the VS-FPGA has shorter interconnection wires between cells due to its dramatically reduced area. Smaller resistance and capacitance of the shortened interconnection wire result in both the energy and delay reduction. As a result, the energies per cycle of the CNT16, LFSR24, and MPY4 implemented on the VS-FPGA are reduced by 31%, 23%, and 34%, respectively, in comparison with those of the CNT16, LFSR24, and MPY4 implemented on the previous CAS-FPGA [32]. On the other hand, the delays are reduced by 29%, 45%, and 29%, respectively. Totally, the EDPs are reduced by 51%, 58%, and 53%, respectively.

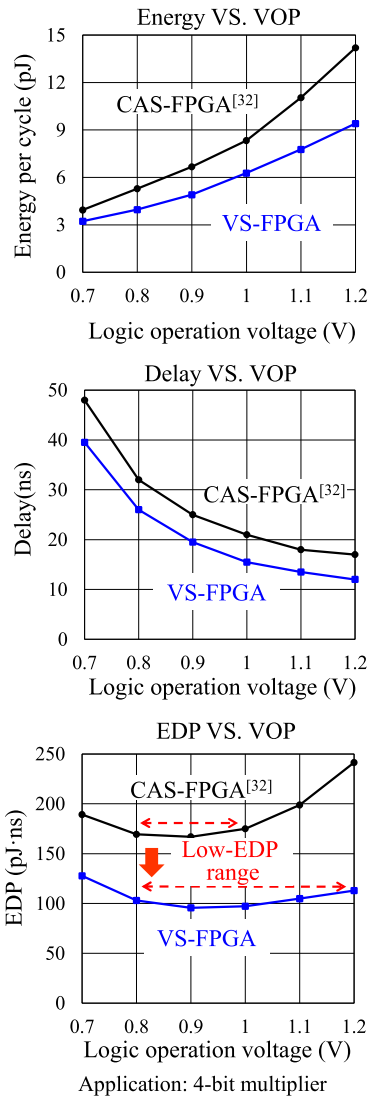



Fig. 24. Energy per cycle, delay, and EDP versus operation voltage of the MPY4 implemented on the CAS- and VS-FPGAs.

The performance comparison of the MPY4s implemented on the VS-FPGA, previous CAS-FPGA [32], and equivalent SRAM-FPGA [32] is summarized in Table I. At logic operation voltage 1.0 V, delay, energy per cycle, and EDP of the VS-FPGA are reduced by 24%, 25%, and 45%, respectively, in comparison with the previous CAS-FPGA. On the other hand, delay, energy per cycle, and EDP of the VS-FPGA are reduced by 80%, 70%, and 94%, respectively, in comparison with the equivalent SRAM-FPGA.

Fig. 24 shows energy per cycle, delay, and EDP versus logic operation voltage of the MPY4 implemented on the CAS- and

VS-FPGAs. A higher logic operation voltage reduces the delay but elevates the energy per cycle. It is worth evaluating the logic operation voltage dependence of the EDP to find an optimum logic operation voltage [52]. Both the CAS- and VS-FPGAs achieve minimum EDP at 0.9 V. The VS-FPGA attains twice wider operation voltage range 0.8–1.2 V of low-EDP (EDP $<$ 1.1 $\times$ $\text{EDP}_{\text{min}}$) than that of 0.8–1.0 V in the CAS-FPGA thanks to shortened interconnect wires.

## V. Conclusion

For the first time, an NV VS-FPGA was fabricated by a 65-nm CMOS process. The VS integrated between BEOL Cu metal layers (M4 and M5) is constructed by two ASs for signal routing and two varistors for AS configuration. Its utilization in routing matrixes for signal routing and memories for logic operations leads to 2.6$\times$ and 12$\times$ logic density improvement compared with the previous CAS-FPGA with access transistors and the conventional SRAM-FPGA with MOS switches, respectively. Correct operation and small delay variation of 7% under automotive temperature grade (−40 °C∼125 °C) were confirmed by testing the fabricated chip. The silicon results show that the VS-FPGA % energy per cycle reduction, 29% delay reduction, and 53% energy-delay product reduction in comparison with the previous CAS-FPGA at a standard operation voltage.

To improve the operation speed of the VS-FPGA, we are developing a new varistor with higher programming current to reduce ON-state resistance of the AS. Also, a VS bidirectional interconnect structure without using tristate buffers is being developed to achieve further area efficiency improvement [53], [54]. Furthermore, a near-memory computing-oriented VS-FPGA is being developed for AI applications [43].

## Acknowledgment

A part of the device processing was operated by the National Institute of Advanced Industrial Science and Technology (AIST), Tsukuba, Japan.

## References

[1] F. Turan, S. S. Roy, and I. Verbauwhede, "HEAWS: An accelerator for homomorphic encryption on the Amazon AWS FPGA," *IEEE Trans. Comput.*, vol. 69, no. 8, pp. 1185–1196, Aug. 2020.

[2] H. Artail *et al.*, "Speedy cloud: Cloud computing with support for hardware acceleration services," *IEEE Trans. Cloud Comput.*, vol. 7, no. 3, pp. 850–865, Jul. 2019.

[3] L. Zhao, I. Matsuo, Y. Zhou, and W.-J. Lee, "Design of an industrial IoT-based monitoring system for power substations," in *Proc. IEEE/IAS 55th Ind. Commercial Power Syst. Tech. Conf. (I&CPS)*, Calgary, AB, Canada, May 2019, pp. 1–6.

[4] M. Urbina, T. Acosta, J. Lazaro, A. Astarloa, and U. Bidarte, "Smart sensor: SoC architecture for the industrial Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6567–6577, Aug. 2019.

[5] W.-C. Fang, K.-Y. Wang, N. Fahier, Y.-L. Ho, and Y.-D. Huang, "Development and validation of an EEG-based real-time emotion recognition system using edge AI computing platform with convolutional neural network system-on-chip design," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 9, no. 4, pp. 645–657, Dec. 2019.

[6] A. Shawahna, S. M. Sait, and A. El-Maleh, "FPGA-based accelerators of deep learning networks for learning and classification: A review," *IEEE Access*, vol. 7, pp. 7823–7859, 2019.

[7] I. Kuon, R. Tessier, and J. Rose, "FPGA architecture: Survey and challenges," *Found. Trends Electron. Des. Autom*, vol. 2, no. 2, pp. 135–253, Feb. 2008.

[8] F.-L. Yuan, C. C. Wang, T.-H. Yu, and D. Marković, "A multi-granularity FPGA with hierarchical interconnects for efficient and flexible mobile computing," *IEEE J. Solid-State Circuits*, vol. 50, no. 1, pp. 137–149, Jan. 2015.

[9] 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.

[10] T. Tuan, A. Rahman, S. Das, S. Trimberger, and S. Kao, "A 90-nm low-power FPGA for battery-powered applications," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 26, no. 2, pp. 296–300, Feb. 2007.

[11] S. Ma and P. Ampadu, "Self-decompressing FPGA bitstreams," in *Proc. IEEE 62nd Int. Midwest Symp. Circuits Syst. (MWSCAS)*, Dallas, TX, USA, Aug. 2019, pp. 247–250.

[12] K. Huang, Y. Ha, R. Zhao, A. Kumar, and Y. Lian, "A low active leakage and high reliability phase change memory (PCM) based non-volatile FPGA storage element," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 61, no. 9, pp. 2605–2613, Sep. 2014.

[13] R. Rajaei, "Radiation-hardened design of nonvolatile MRAM-based FPGA," *IEEE Trans. Magn.*, vol. 52, no. 10, pp. 1–10, Oct. 2016.

[14] R. Zand and R. F. DeMara, "MRAM-enhanced low power reconfigurable fabric with multi-level variation tolerance," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 66, no. 12, pp. 4662–4672, Dec. 2019.

[15] M. Natsui *et al.*, "A 47.14-$\mu$W 200-MHz MOS/MTJ-hybrid nonvolatile microcontroller unit embedding STT-MRAM and FPGA for IoT applications," *IEEE J. Solid-State Circuits*, vol. 54, no. 11, pp. 2991–3004, Nov. 2019.

[16] O. Turkyilmaz *et al.*, "RRAM-based FPGA for 'normally off, instantly on' applications," in *Proc. IEEE/ACM Int. Symp. Nanosc. Architectures (NANOARCH)*, Amsterdam, The Netherlands, Jul. 2012, pp. 101–108.

[17] A. Ahari, H. Asadi, B. Khaleghi, and M. B. Tahoori, "A power-efficient reconfigurable architecture using PCM configuration technology," in *Proc. Design, Automat. Test Eur. Conf. Exhib. (DATE)*, Dresden, Germany, Mar. 2014, pp. 1–6.

[18] S. Tanachutiwat, M. Liu, and W. Wang, "FPGA based on integration of CMOS and RRAM," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 19, no. 11, pp. 2023–2032, Nov. 2011.

[19] Y.-C. Chen, W. Wang, H. Li, and W. Zhang, "Non-volatile 3D stacking RRAM-based FPGA," in *Proc. 22nd Int. Conf. Field Program. Log. Appl. (FPL)*, Oslo, Norway, Aug. 2012, pp. 367–372.

[20] Y. Y. Liauw, Z. Zhang, W. Kim, A. E. Gamal, and S. S. Wong, "Nonvolatile 3D-FPGA with monolithically stacked RRAM-based configuration memory," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2012, pp. 406–408.

[21] J. Greene *et al.*, "A 65 nm flash-based FPGA fabric optimized for low cost and power," in *Proc. 19th ACM/SIGDA Int. Symp. Field Program. Gate Arrays (FPGA)*, Feb. 2011, pp. 87–95.

[22] J.-J. Wang *et al.*, "A novel 65 nm radiation tolerant flash configuration cell used in RTG4 field programmable gate array," *IEEE Trans. Nucl. Sci.*, vol. 62, no. 6, pp. 3072–3079, Dec. 2015.

[23] K. Zaitsu, K. Tatsumura, M. Matsumoto, M. Oda, and S. Yasuda, "Non-volatile programmable switch with adjacently integrated flash memory and CMOS logic for low-power and high-speed FPGA," *IEEE Trans. Electron Devices*, vol. 62, no. 12, pp. 4009–4014, Dec. 2015.

[24] X. Chen, K. Ni, M. T. Niemier, Y. Han, S. Datta, and X. S. Hu, "Power and area efficient FPGA building blocks based on ferroelectric FETs," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 66, no. 5, pp. 1780–1793, May 2019.

[25] B. Khaleghi and H. Asadi, "A resistive RAM-based FPGA architecture equipped with efficient programming circuitry," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 65, no. 7, pp. 2196–2209, Jul. 2018.

[26] X. Tang, E. Giacomin, G. De Micheli, and P.-E. Gaillardon, "Circuit designs of high-performance and low-power RRAM-based multiplexers based on 4T(ransistor)1R(RAM) programming structure," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 64, no. 5, pp. 1173–1186, May 2017.

[27] X. Tang, E. Giacomin, P. Cadareanu, G. Gore, and P.-E. Gaillardon, "A RRAM-based FPGA for energy-efficient edge computing," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, Grenoble, France, Mar. 2020, p. 144.

[28] M. Tada, K. Okamoto, T. Sakamoto, M. Miyamura, N. Banno, and H. Hada, "Polymer solid-electrolyte switch embedded on CMOS for nonvolatile crossbar switch," *IEEE Trans. Electron Devices*, vol. 58, no. 12, pp. 4398–4406, Dec. 2011.

[29] M. Miyamura *et al.*, "Programmable cell array using rewritable solid-electrolyte switch integrated in 90 nm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2011, pp. 228–229.
[30] M. Tada *et al.*, "Improved off-state reliability of nonvolatile resistive switch with low programming voltage," *IEEE Trans. Electron Devices*, vol. 59, no. 9, pp. 2357–2362, Sep. 2012.
[31] M. Miyamura *et al.*, "First demonstration of logic mapping on non-volatile programmable cell using complementary atom switch," in *IEDM Tech. Dig.*, San Francisco, CA, USA, Dec. 2012, pp. 10.6.1–10.6.4.
[32] M. Miyamura *et al.*, "Low-power programmable-logic cell arrays using nonvolatile complementary atom switch," in *Proc. 15th Int. Symp. Qual. Electron. Design*, Santa Clara, CA, USA, Mar. 2014, pp. 330–334.
[33] X. Bai *et al.*, "A low-power cu atom switch programmable logic fabricated in a 40 nm-node CMOS technology," in *Proc. Symp. VLSI Technol.*, Kyoto, Japan, Jun. 2017, pp. T28–T29.
[34] R. Nebashi *et al.*, "High-density and fault-tolerant cu atom switch technology toward 28 nm-node nonvolatile programmable logic," in *Proc. IEEE Symp. VLSI Technol.*, Honolulu, HI, USA, Jun. 2018, pp. 127–128.
[35] L. Zhang *et al.*, "Ultrathin metal/amorphous-silicon/metal diode for bipolar RRAM selector applications," *IEEE Electron Device Lett.*, vol. 35, no. 2, pp. 199–201, Feb. 2014.
[36] W. Lee *et al.*, "Varistor-type bidirectional switch ($J_{MAX}$>$10^7$ A/cm$^2$, selectivity∼104) for 3D bipolar resistive memory arrays," in *Proc. Symp. VLSI Technol. (VLSIT)*, Honolulu, HI, USA, Jun. 2012, pp. 37–38.
[37] E. Cha *et al.*, "Nanoscale (∼10 nm) 3D vertical ReRAM and $NbO_2$ threshold selector with TiN electrode," in *IEDM Tech. Dig.*, Washington, DC, USA, Dec. 2013, pp. 10.5.1–10.5.4.
[38] Q. Luo *et al.*, "Cu BEOL compatible selector with high selectivity (>107), extremely low off-current (∼pA) and high endurance (>1010)," in *IEDM Tech. Dig.*, Dec. 2015, pp. 10.4.1–10.4.4.
[39] K. Okamoto *et al.*, "Bidirectional TaO-diode-selected, complementary atom switch (DCAS) for area-efficient, nonvolatile crossbar switch block," in *Proc. Symp. VLSI Technol.*, Kyoto, Japan, Jun. 2013, pp. T242–T243.
[40] N. Banno *et al.*, "A novel two-varistors (a-Si/Sin/a-Si) selected complementary atom switch (2 V-1CAS) for nonvolatile crossbar switch with multiple fan-outs," in *IEDM Tech. Dig.*, Dec. 2015. Washington, DC, USA, pp. 2.5.1–2.5.4.
[41] N. Banno *et al.*, "50×20 crossbar switch block (CSB) with two-varistors (a-Si/SiN/a-Si) selected complementary atom switch for a highly-dense reconfigurable logic," in *IEDM Tech. Dig.*, San Francisco, CA, USA, Dec. 2016, pp. 16.4.1–16.4.4.
[42] N. Banno *et al.*, "Low-power crossbar switch with two-varistor selected complementary atom switch (2 V-1CAS; via-switch) for nonvolatile FPGA," *IEEE Trans. Electron Devices*, vol. 66, no. 8, pp. 3331–3336, Aug. 2019.
[43] M. Hashimoto *et al.*, "Via-switch FPGA: 65 nm CMOS implementation and architecture extension for al applications," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2020, pp. 502–504.
[44] X. Bai *et al.*, "1.5× energy-efficient and 1.4× operation-speed via-switch FPGA with rapid and low-cost ASIC migration by via-switch copy," in *Proc. IEEE Symp. VLSI Technol.*, Honolulu, HI, USA, Jun. 2020, pp. 1–2.
[45] N. Suzumura *et al.*, "A new TDDB degradation model based on Cu ion drift in Cu interconnect dielectrics," in *Proc. IEEE Int. Rel. Phys. Symp.*, San Jose, CA, USA, Mar. 2006, pp. 484–489.
[46] N. Banno *et al.*, "A fast and low-voltage cu complementary-atom-switch 1 Mb array with high-temperature retention," in *Symp. VLSI Technol. (VLSI-Technology), Dig. Tech. Papers*, Jun. 2014, pp. 1–2.
[47] G. Lemieux, E. Lee, M. Tom, and A. Yu, "Directional and single-driver wires in FPGA interconnect," in *Proc. IEEE Int. Conf. Field-Program. Technol.*, Brisbane, NSW, Australia, Dec. 2004, pp. 41–48.
[48] J. Luu *et al.*, "VPR 5.0: FPGA CAD and architecture exploration tools with single-driver routing, heterogeneity and process scaling," *ACM Trans. Reconfigurable Technol. Syst.*, vol. 4, no. 4, pp. 1–23, Dec. 2011.
[49] (Mar. 2021). *The Automotive-Grade Device Handbook*. Intel corporation. [Online]. Available: https://www.intel.com/content/dam/www/programmable/ us/en/pdfs/literature/hb/auto/automotive_handbook.pdf
[50] M. Miyamura *et al.*, "Nanobridge-based FPGA in high-temperature environments," *IEEE Micro*, vol. 37, no. 5, pp. 32–42, Oct. 2017.
[51] B. Khaleghi and T. S. Rosing, "Thermal-aware design and flow for FPGA performance improvement," in *Proc. Design, Automat. Test Eur. Conf. Exhib. (DATE)*, Mar. 2019, pp. 342–347.
[52] V. Gaudet, "Chapter 4.1. Low-power design techniques for state-of-the-art CMOS technologies," in *Recent Progress in the Boolean Domain*. Newcastle Upon Tyne, U.K.: Cambridge Scholars Publishing, 2013, pp. 187–212.
[53] H. Ochi *et al.*, "Via-switch FPGA: Highly dense mixed-grained reconfigurable architecture with overlay Via-switch crossbars," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 26, no. 12, pp. 2723–2736, Dec. 2018.
[54] R. Doi, J. Yu, and M. Hashimoto, "Sneak path free reconfiguration with minimized programming steps for via-switch crossbar-based FPGA," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 39, no. 10, pp. 2572–2587, Oct. 2020.

**Xu Bai** (Senior Member, IEEE) received the B.E. degree in communication engineering from Southwest Jiaotong University, Chengdu, China, in 2008, and the M.S. and Ph.D. degrees in information sciences from Tohoku University, Sendai, Japan, in 2011 and 2014, respectively.

He joined NEC Corporation, Tokyo, Japan, in 2014, where he was involved in research and development of nonvolatile memory and programmable logic device. He is currently a Senior Researcher with NanoBridge Semiconductor, Inc., Tsukuba, Japan. His research interests include nonvolatile memory, programmable logic device, and new concept VLSI architectures, including multiple-valued VLSI architecture and compute-in-memory architecture.

Dr. Bai is a member of the Institute of Electronics, Information and Communication Engineers (IEICE), Japan. He received the IEEE Sendai Student Award in 2010, the Excellent Student Poster Award at 22nd International Workshop on Post-Binary ULSI Systems in 2013, and the Young Researcher Award at 48th International Conference on Solid State Devices and Materials in 2015.

**Naoki Banno** (Member, IEEE) received the B.S. and M.S. degrees from Waseda University, Tokyo, Japan, in 2002 and 2004, respectively, and the Ph.D. degree from The University of Tokyo, Tokyo, in 2012.

He was with NEC Corporation, Tokyo, from 2004 to 2021. He is currently the Manager of NanoBridge Semiconductor, Inc., Tsukuba, Japan.

**Makoto Miyamura** received the B.E. and M.E. degrees from The University of Tokyo, Tokyo, Japan, in 2000 and 2002, respectively.

He joined NEC Corporation, Tokyo, in 2002. From 2008 to 2009, he was a Visiting Researcher with Stanford University, Stanford, CA, USA. He is currently a Principal Researcher with NanoBridge Semiconductor, Inc., Tsukuba, Japan. His research interests include static random access memory (SRAM) variability, nonvolatile memory, and energy-efficient circuit design.

**Ryusuke Nebashi** received the B.E. and M.E. degrees in applied physics and physico-informatics from Keio University, Tokyo, Japan, in 2002 and 2004, respectively.

He joined NEC Corporation, Tokyo, in 2004, where he was involved with the development of nonvolatile memory and programmable logic device. He is currently a Principal Researcher with NanoBridge Semiconductor, Inc., Tsukuba, Japan.

**Koichiro Okamoto** received the B.S., M.S., and Ph.D. degrees in applied chemistry from The University of Tokyo, Tokyo, Japan, in 2004, 2006, and 2009, respectively.

He joined NEC Corporation, Tokyo, in 2009. He is currently a Senior Researcher with NanoBridge Semiconductor, Inc., Tsukuba, Japan. His research interests include nonvolatile memory and integration technology.

**Hideaki Numata** received the B.S. and M.S. degrees from Nagoya University, Nagoya, Japan, in 1988 and 1990, respectively.

He joined NEC Corporation, Tokyo, Japan, in 1990. He is currently a Principal Researcher with NanoBridge Semiconductor, Inc., Tsukuba, Japan. His research interests include nonvolatile memory, sensing device, and integration technology.

**Noriyuki Iguchi** received the A.S. degree from NEC Industrial Technology Junior College, Kawasaki, Japan, in 1991.

He joined NEC Corporation, Tokyo, Japan, in 1990. He is currently a Researcher with NanoBridge Semiconductor, Inc., Tsukuba, Japan.

**Masanori Hashimoto** (Senior Member, IEEE) received the B.E., M.E., and Ph.D. degrees in communications and computer engineering from Kyoto University, Kyoto, Japan, in 1997, 1999, and 2001, respectively.

He is currently a Professor with the Graduate School of Informatics, Kyoto University. His current research interests include design for manufacturability and reliability, timing and power integrity analysis, reconfigurable computing, soft error characterization, and low-power circuit design.

Dr. Hashimoto was on the technical program committees of international conferences, including Design Automation Conference (DAC), International Conference on Computer Aided Design (ICCAD), International Test Conference (ITC), Symposium on VLSI Circuits, Asia and South Pacific Design Automation Conference (ASP-DAC), and Design, Automation and Test in Europe Conference (DATE). He serves/served as the Editor-in-Chief for *Microelectronics Reliability* (Elsevier) and an Associate Editor for IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I: REGULAR PAPERS, and *ACM Transactions on Design Automation of Electronic Systems* (TODAES).

**Tadahiko Sugibayashi** received the B.S. and M.S. degrees in material science from Osaka University, Osaka, Japan, in 1984 and 1986, respectively.

In 1986, he joined NEC Corporation, Tokyo, Japan, where he was involved in memory LSI design. He is currently the CEO of NanoBridge Semiconductor, Inc., Tsukuba, Japan. He is also involved in the development of ultralow power circuits utilizing next-generation nonvolatile devices.

Mr. Sugibayashi is a member of the Institute of Electronics, Information and Communication Engineers (IEICE), Japan.

**Toshitsugu Sakamoto** received the M.S.E.E. and Ph.D. degrees from Osaka University, Osaka, Japan, in 1991 and 1996, respectively.

After joining NEC Corporation, Tokyo, Japan, in 1991, he has worked on nanoelectronic devices such as hot electron transistors, single electron devices, and nano-electromechanical devices. From 1999 to 2000, he was a Visiting Researcher with California Institute of Technology, Pasadena, CA, USA. He is currently the Chief Technology Officer (CTO) of NanoBridge Semiconductor, Inc., Tsukuba, Japan. His current focus is on atom switch that will be used in switching elements of programmable logic devices.

**Munehiro Tada** (Fellow, IEEE) received the M.S. (Hons.) and Ph.D. degrees from Keio University, Tokyo, Japan, in 1999 and 2007, respectively.

In 1999, he joined NEC Corporation, Tokyo, Japan. He was a Visiting Scholar with Stanford University, Stanford, CA, USA, in 2008. He is currently a VP Engineer at NanoBridge Semiconductor, Inc., Tsukuba, Japan. His current research interests include ultralow power device, circuit, systems, and applications of emerging technologies.

Dr. Tada is a fellow of Japan Society of Applied Physics (JSAP).