Received 8 January 2015; revised 12 March 2015; accepted 30 March 2015. Date of publication 9 April 2015; date of current version 19 June 2015.
The review of this paper was arranged by Editor M. Östling.

Digital Object Identifier 10.1109/JEDS.2015.2421301

Silicon Carbide (SiC) Nanoelectromechanical Antifuse for Ultralow-Power One-Time-Programmable (OTP) FPGA Interconnects

TINA HE (Student Member, IEEE), FENGCHAO ZHANG (Student Member, IEEE), SWARUP BHUNIA (Senior Member, IEEE), AND PHILIP X.-L. FENG (Senior Member, IEEE)

Department of Electrical Engineering and Computer Science, Case School of Engineering, Case Western Reserve University, Cleveland, OH 44106, USA

CORRESPONDING AUTHOR: P. X.-L. FENG (e-mail: philip.feng@case.edu)

This work was supported in part by the National Science Foundation under Grant CCF-1116102, and in part by Defense Advanced Research Program Agency under Grant D11AP00292.

ABSTRACT We report a new nanoscale antifuse featuring low-power and high-programming speed, by employing silicon carbide (SiC) nanoelectromechanical systems (NEMS). We show that the SiC NEMS antifuses can enable ultralow-power one-time-programmable (OTP) field-programmable gate arrays (FPGAs) with characteristics promising for security-sensitive and harsh-environment applications. The SiC NEMS antifuses offer minimal leakage, low-programming voltage (down to ~1.5 V), ideally abrupt transient, high on/off ratios ($>10^7$) and high-current carrying ability ($>10^6$ A/cm^2), and very small footprints (~1 μm^2 to ~0.1 μm^2 per device). We further describe new designs of antifuses, simulate FPGA benchmarking circuits based on experimentally demonstrated practical NEMS antifuses, and compare their advantageous performance with state-of-the-art conventional antifuse FPGAs. We also demonstrate a SiC NEMS antifuse-based OTP memory cell with a read margin of $>10^6$.

INDEX TERMS Antifuse, field-programmable gate array (FPGA), nanoelectromechanical systems (NEMS), silicon carbide (SiC), programming voltage, ultralow power.

I. INTRODUCTION

FPGAs are integrated circuits that are pre-fabricated to be electrically programmed and configured in the field to serve individual users different needs for digital circuit or system [1]. The key feature of FPGAs compared to application specific integrated circuits (ASICs) is the reconfigurability or programmability. The FPGA is able to implement a new function on the chip after it is fabricated by the manufacturers. Because of this feature, FPGAs are often preferred in situations that call for low volume units with lower cost and shorter time to deployment, as compared to ASICs which normally would require more time and investment to develop a prototype. However, in order to have a fully flexible circuit that can be configured to satisfy all the needs from various users, today's FPGAs are significantly larger, slower, and more power consuming than their ASIC counterparts. At the device level, the fundamental cause of these limitations lies in the structure of the FPGAs.

An FPGA typically consists of three components: programmable logic blocks which implement various logic functions, programmable routing that connects these logic functions, and I/O blocks that are connected to logic blocks through routing interconnect and make input/output of chip connection. Among the three, the programmable routing interconnect comprises of almost 90% of the total chip area, and thus contributes to most of the delay, area and power consumption. Typically an FPGA uses 20 to over 100 interconnects per logic gate to link logic blocks [2].

Antifuses are among the commonly used interconnect devices for FPGAs. An antifuse is a one-time-programmable (OTP), two-terminal device that has high initial resistance (e.g., open circuit) till a programming voltage

changes it to low resistance in an irreversible process [2]–[6]. It has been extensively employed as an economical and convenient solution in complex logic ICs for improved functionality and flexibility, and thus widely used in non-volatile memories and OTP secure FPGAs [7]–[10]. Compared to other FPGA programmable interconnect devices such as SRAM-controlled pass transistors [11] and EPROMs [12], antifuse interconnects offer smaller size, faster programming, lower programmed resistance and lower parasitic capacitance [13]–[15]. In particular, antifuse-based FPGAs offer higher security, stemming primarily from their non-volatility [14]. It secures a design due to piracy and difficulty in determining the state of an antifuse that protects against direct physical attack [16]. Fig. 1 presents the architecture of a commercial antifuse FPGA. The major components are: core tiles, which contains the logic blocks, RAMs and data buffering for data storage, and I/O structure for data input and output. The core tiles are linked by antifuses as interconnects.

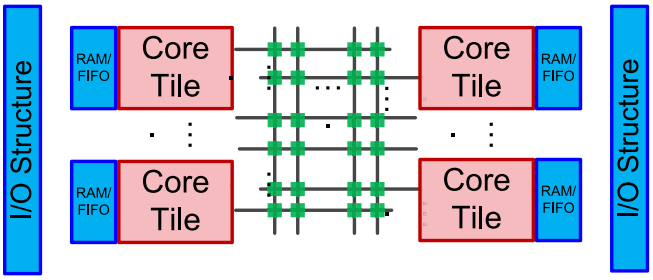


FIGURE 1. Schematic architecture of Microsemi Axcelerator antifuse FPGA. The major components include core tiles containing the logic blocks, RAMs, and data buffering for storage and I/O structure for inputs and outputs. Green squares are antifuses. FIFO stands for first-in and first-out.

While antifuse-based FPGAs are currently among the most secure programmable devices available [16], limitations and challenges remain in high leakage power, increasing security requirements (e.g., against potential attack in the form of reprogramming), and scalability to advanced technology nodes. Many critical defense and aerospace applications drive great demands for devices with higher programming speed, lower power, increasing tolerance to radiation [17], [18] and harsh environment, and higher resistance to attacks.

A conventional antifuse consists of a dielectric layer sandwiched between two electrodes as shown in Fig. 2(a). Commonly used implementations include n+ diffusion/oxide-nitride-oxide (ONO) dielectrics/poly-silicon [3], [19], metal-to-metal structures [20]–[22], and amorphous silicon/metal structures [23], [24].

Spurred by advancements in reliably fabricating and manipulating mechanically active nanostructures, recently nanoelectromechanical systems (NEMS) have been explored as logic building blocks, including single switches, logic gates [25], and nonvolatile memories [26], [27]. NEMS antifuses offer intrinsically strong and ideal insulation at 'off'

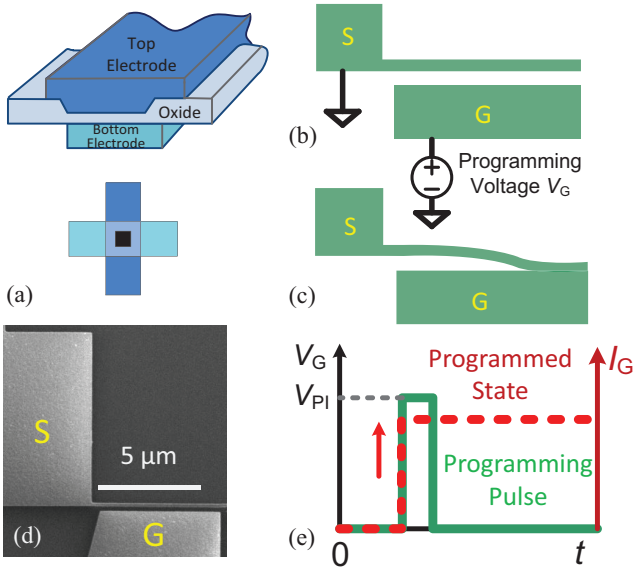


FIGURE 2. Nanoelectromechanical systems (NEMS)-enabled antifuse design and operation principle. (a) Conventional antifuse device structure and top view. (b) Unprogrammed NEMS antifuse and the programming scheme. (c) Programmed NEMS antifuse, with S permanently connected to G upon the first programming event. (d) SEM image of a SiC NEMS antifuse. (e) Expected *I* curve (red dashed line) as we apply the gate voltage V_G (green solid line) in programming cycle. V_{PI} is the minimum programming voltage required.

state with air gaps separating electrodes, instant programming via abrupt switching, low programming voltage, and robust non-volatile programmed state. In particular, SiC NEMS exploit the outstanding thermal and mechanical properties of this technologically important material for reliable performance even in harsh environments (e.g., high temperature, radiation) [25]. Established thin-film technologies can make SiC NEMS on various substrates [28]. SiC is especially suited for NEMS *antifuses* also thanks to its exceptional ability in carrying high current; and SiC can hardly be *fused* even at very high current levels. In this work, we demonstrate very low voltage SiC NEMS antifuses for the first time, and show significant enhancement in energy efficiency in FPGA designs enabled by such NEMS antifuses.

II. SiC NEMS ANTIFUSE STRUCTURE, FABRICATION, AND OPERATING PRINCIPLES

The NEMS antifuse is based on electrostatically actuated SiC nanocantilevers illustrated in Fig. 2(b) and (c). It is a two-terminal device in lateral configuration, with a fixed electrode gate (G) for programming and a movable nanocantilever as source (S), separated by an air gap. At the initial (unprogrammed) state, G and S form an open circuit. A programming voltage applied between G and S actuates the device to connect G and S (i.e., to *program* the antifuse). In contrast to three-terminal NEMS logic switches [25], in antifuses we *exploit* adhesion forces to keep G and S connected *after programming* (which must be *avoided* in three-terminal NEMS logic switches [25]).

The SiC NEMS antifuse fabrication starts with a 4-inch Si wafer with 500nm thermally-grown SiO_2 and 500nm thick polycrystalline SiC (poly-SiC) deposited using low-pressure chemical vapor deposition (LPCVD), as illustrated in Fig. 3(a). The deposition of SiC is done using a customized hot-wall horizontal furnace at temperature of 900°C and the SiC film is heavily doped via using NH_3 gas [28]. We then sputter PMMA as a mask and define the pattern with wafer-scale electron beam lithography (EBL) and transfer the pattern using reactive ion etching (RIE). Finally we release the suspended cantilever beam with a high-yield process of vapor hydrofluoric acid (HF) etching of SiO_2.

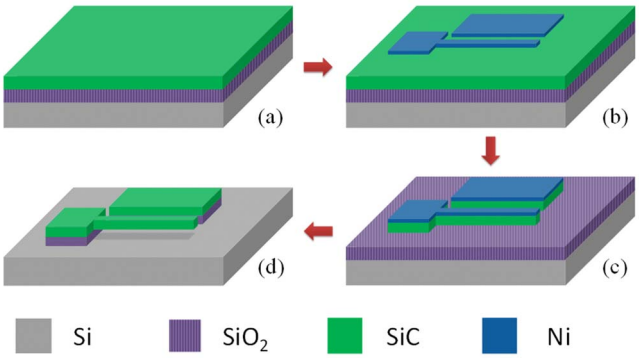


FIGURE 3. Nanofabrication of NEMS antifuse. (a) Poly-SiC deposited on 500 nm SiO_2 on a 4 In Si wafer. (b) 40 nm Ni layer after EBL and lift-off as the etching mask for SiC. (c) Reactive ion etching of SiC. (d) Ni removal and vapor HF etching of SiO_2 to suspend the nanocantilever beam.

In device operation, the movable cantilever beam is usually grounded at its clamping point to provide a voltage potential reference for the program electrode G. The device can be approximately modeled as a parallel plate capacitor, with one plate fixed and the other attached to a spring. The capacitance of this parallel plate C_{act} is given by

$$C_{act} = \frac{\varepsilon A_{act}}{g_{GS} - x}, \tag{1}$$

where ε is the permittivity of air, A_{act} is the overlapping area of the two plates, x is the distance the plate moves and g_{GS} is the as-fabricated gap between the two plates. When a voltage V_G is applied between the two plates, the total force acting on the movable plate is given by

$$F_{tot} = \frac{1}{2}\frac{\varepsilon A_{act}}{(g_{GS} - x)^2} V_G^2 - k_{eff} x. \tag{2}$$

The device behavior can be understood by its electrostatic coupling via the air gap capacitor. As the applied gate voltage V_G increases to the pull-in voltage V_{PI} or beyond, the NEMS antifuse is programmed. The voltage is given by

$$V_{PI} \approx \left(8k_{eff} g_{GS}^3 / 27\varepsilon t_B L_G\right)^{1/2}, \tag{3}$$

where k_{eff} is the effective stiffness of the cantilever, g_{GS} the gap between G and S, ε the dielectric constant, t_B the thickness of the cantilever and L_G the length of the actuation gate, respectively. After programming (as V_G returns to 0) the total force acting on the cantilever is $F_a - F_{M,max} = 0$, where F_a is the contact adhesion force and $F_{M,max} = k_{eff} g_{GS}$ is the mechanical restoring force when the contact is made. The switching time is estimated by

$$\tau_s \approx \sqrt{27/2}\left(V_{on}/\omega_0 V_G\right), \tag{4}$$

where ω_0 is the fundamental-mode resonance frequency of the cantilever [29], V_{PI} the pull-in/programming voltage and V_G the actual applied gate voltage.

III. DEVICE CHARACTERIZATION AND VARIATION

The electrical characteristics of SiC NEMS antifuses will be described in this Section. We study the programming of NEMS antifuse and compare the initial state with programmed state. The characteristics of the SiC NEMS antifuses are investigated by high-precision I-V measurements using a source measurement unit (SMU) shown in Fig. 4(a).

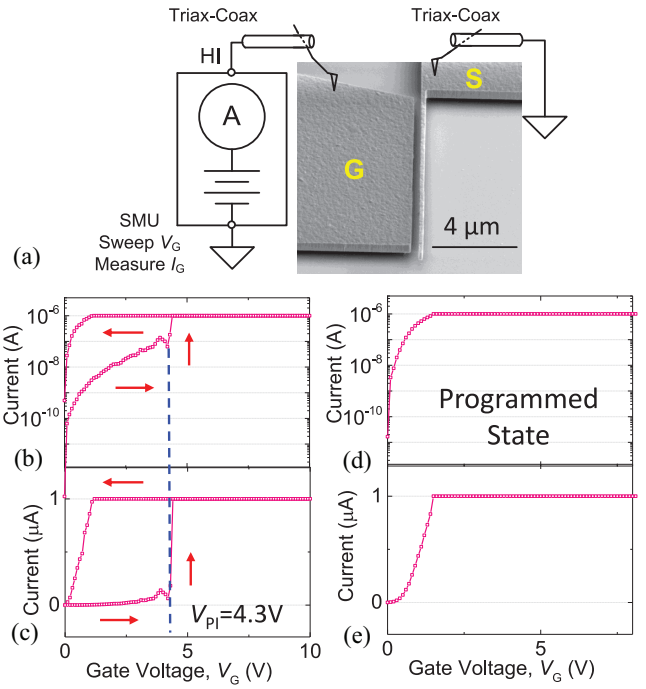


FIGURE 4. SiC NEMS antifuse programming cycles. Data shows the first and second cycle of the device. (a) Schematic of the measurement system with SEM image of a NEMS antifuse. (b) and (c) Programming cycle of NEMS antifuse: sweeping the gate voltage V_G to above V_{PI} and back. (d) and (e) Apply the same sweeping gate voltages after the programming cycle showing the connection is irreversible. Currents are plotted in (b) and (d) in logarithmic scale, and in (c) and (e) in linear scale, respectively.

A. PROGRAMMING OF NEMS ANTIFUSES

Prior to programming, the NEMS antifuse is in its initial state which is 'open', with high resistance of >10GΩ. The current increases abruptly at $V_{PI} = 4.3$V, showing instant programming via NEMS contact. As V_G sweeps back to 0V, the antifuse stays connected with on-state current $I_{on} \geq 1\mu$A, which is the current measurement compliance

we manually set. We sweep V_G again after the programming cycle on the NEMS antifuse whose state has been changed to 'programmed', and observe resistive behavior from the *I-V* curve in Fig. 4(d) and (e).

The NEMS antifuse can have very low programming voltage and high on/off ratios. Fig. 5(a) and (b) shows a NEMS antifuse with ultralow programming voltage at $V_{PI} = 1.6$V. The non-volatility is demonstrated as it stays connected with high I_{on} ($\geq 1\mu$A) as V_G sweeps back to 0V. The gate leakage in this particular device is not fundamental to this type of devices, but is caused by local defects in the SiO_2 layer underneath this specific SiC device (which happens to be on top of a region of non-ideal SiO_2 with defects). Fig. 5(c) shows the programming cycle of another antifuse with a low programming voltage $V_{PI} = 2.2$V and a lower measurement compliance of 100nA.

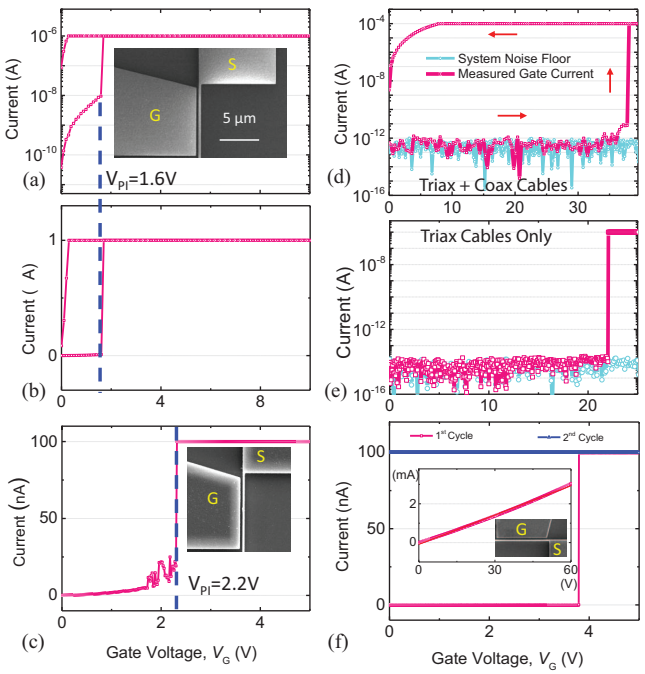


FIGURE 5. SiC NEMS antifuse programming cycle. The current is plotted in (a) logarithmic and (b) linear scale. The current limit (compliance) is set to be 1 μA. Inset in (a) shows an SEM image of the SiC NEMS antifuse. (c) Another antifuse device with low programming voltage, tested with a current limit set at 100 nA. (d) SiC NEMS antifuse with higher measurement compliance (10^{-4}A) showing no gate leakage and on/off ratio of $> 10^8$, the blue trace is the system noise floor measured when no antifuse device is being probed (testing probes not contacting the device). (e) Programming cycle of a SiC NEMS antifuse showing the change from initial state and that the leakage is below the system noise floor (~10 fA). (f) SiC NEMS antifuse first and second cycle with 10^{-7} A current limit. Inset in (f) shows the antifuse connection after 24 months of inactivity (stored in moderate vacuum) and the SEM image of this device.

For the true off-state/leakage current measurement, we establish the noise floor of the measurement system by calibrating the system. We first lift up the testing probes and make sure there is no contact between the probes and the device, and perform a voltage sweep while recording the current. This gives us the noise floor of the measurement system. Then we put down the testing probes to engage the device in measurement, and perform the same voltage sweep and record the current. By comparing the data from these two measurements we can obtain the off-state/leakage current.

Fig. 5(d) and (e) demonstrate another two NEMS antifuses without any measurable gate leakage, where the light blue traces are the system noise floor measured with the probes lifted up. The noise floor level for connections with triax and coax cables is 10^{-14}~10^{-13}A [Fig. 5(d)], and 10^{-16}~10^{-15}A for triax cables only [Fig. 5(e)]. The high measurement compliance of 10^{-4}A in the measurement plotted in Fig. 5(d) suggests a current density of $> 10^5$A/cm^2. The leakage current is below the noise floor of the measurement system (~10^{-13}A–10^{-14}A, depending on cables). Fig. 5(f) is the programming and second cycle of a NEMS switch with 100nA of compliance. The inset of Fig. 5(f) confirms the connection of this already programmed device after 24 months of inactivity, showing an R_{on} of 20kΩ, demonstrating long-term stability of the antifuse.

To study the on/off ratio between unprogrammed and programmed device, we apply the same stress voltage sweep on an unprogrammed NEMS antifuse and an already programmed antifuse and the currents are plotted in Fig. 6. Under the same applied stress voltage, the on/off current ratio is $>10^7$, showing that NEMS antifuses provide excellent insulation when unprogrammed and stable connection after being programmed. This margin leaves a sufficient space for designers to define the on/off state using these devices in logic circuits and FPGAs.

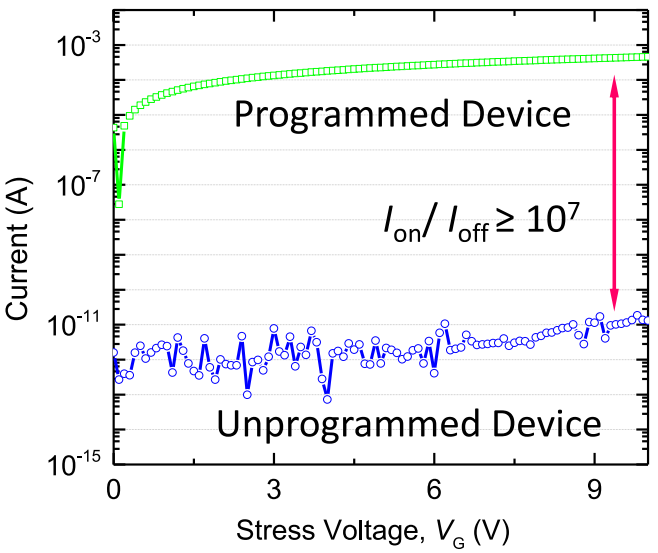


FIGURE 6. Stress voltage sweep on programmed and unprogrammed NEMS antifuse showing a large on/off ratio between programmed and unprogrammed interconnect devices.

B. NEMS ANTIFUSE BREAKDOWN AND HIGH CURRENT DENSITY

For conventional antifuses, the breakdown of the dielectrics has significant impact on reliability and programming, and is heavily studied and characterized [18]–[24]. Different from

the conventional semiconductor antifuses, the breakdown of NEMS antifuse usually happens when the NEMS structure raptures or breaks physically when the current it is bearing exceeds a certain critical level.

We study the breaking down mechanism of NEMS antifuse by sweeping the stress voltage on programmed NEMS antifuses and raise the measurement compliance so that the current can increase to a higher level until the device breaks down. Fig. 7 shows the breaking down processes of three devices. The programmed antifuse exhibits resistor-like behavior in *I-V* characterization, where the current increases linearly as the stress voltage sweeps up. When the current reaches the breakdown level, it starts to decrease to 0A, and remains unchanged as the stress voltage sweeps up and down. The highest current passing through the antifuse interconnect before it breaks down, $I_{\text{on,max}}$, is 0.13mA to 0.33mA for the three devices tested. The current density the device can carry is calculated with:

$$J_{\text{c,max}} = \frac{I_{\text{on,max}}}{A_{\text{B}}} = \frac{I_{\text{on,max}}}{w_{\text{B}} t_{\text{B}}}, \tag{5}$$

where A_{B} is the cross section area of the cantilever beam and w_{B} and t_{B} are the beam width and thickness, respectively. The highest current density the device can carry is at $J_{\text{c,max}} \sim 10^6 \text{A/cm}^2$.

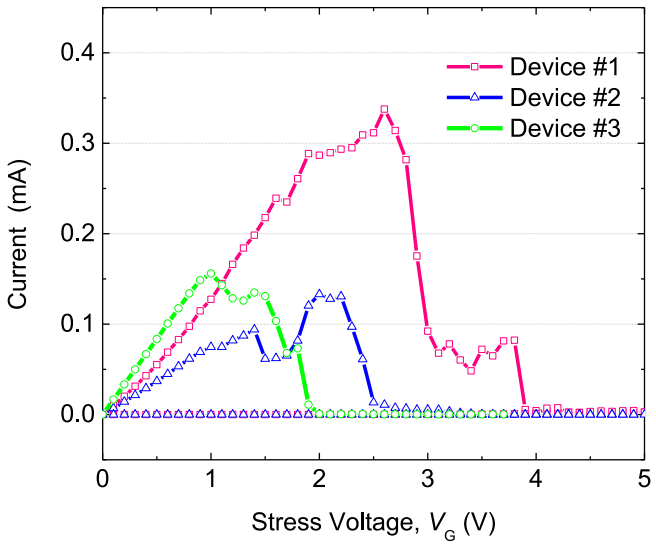


FIGURE 7. Programmed NEMS antifuses breakdown process under high currents.

C. DEVICE VARIATION

We have tested ~30 SiC NEMS antifuses with different dimensions. Among these devices, V_{PI} varies from 1.6V to 35V, and I_{on} can be as high as >0.1mA (with compliance). Fig. 8 presents the programming voltage variation of the NEMS cantilever devices. The minimum voltage required to program a NEMS antifuse is the pull-in voltage (V_{PI}) of the cantilever beam. Based on the analysis in Section II, the programming voltage will decrease as cantilever beam length increases; and large actuation gap will cause the programming voltage to increase, as shown in the color map in Fig. 8(a).

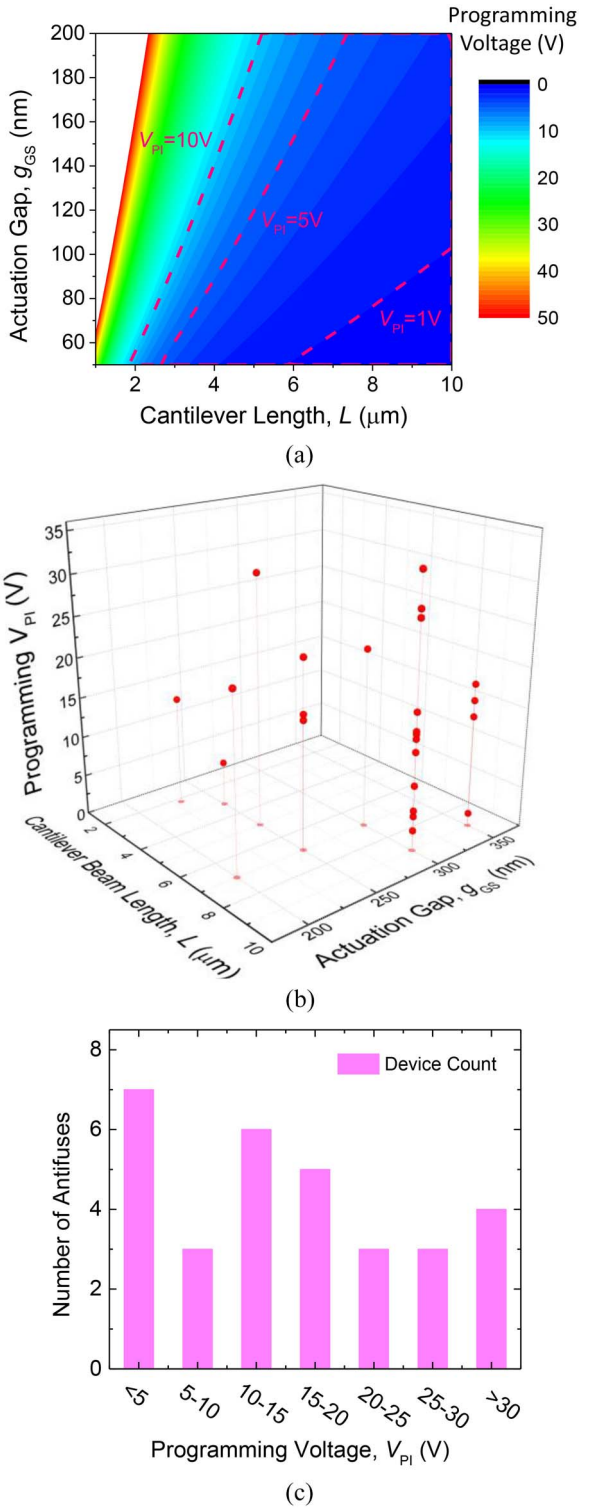


FIGURE 8. Programming voltage versus cantilever geometry. (a) Color map of calculated programming voltage versus actuation gap and cantilever length. (b) Measured programming voltage with designed actuation gap and cantilever length. (c) Histogram of measured programming voltage distribution across ~30 NEMS antifuse devices.

The measured programming voltages versus designed cantilever beam lengths and actuation gaps are illustrated in Fig. 8(b). Some NEMS antifuses with same designed parameters such as beam length and actuation gap may have different programming voltages. The possible causes are fabrication variations, and non-idealities in SiC film uniformity and oxide quality. The distribution statistics of the measured programming voltages are shown in Fig. 8(c).

D. IMPROVED DESIGN OF NEMS ANTIFUSE

Based on the above experimental results, we further describe designs of future-generation NEMS antifuses for improving performance and functions. One of the design improvements we propose is to engineer the beam shape to increase the adhesion between G and S, thus to minimize the on-state resistance R_{on} under the same actuation load. Low R_{on} is desired to minimize delay between logic modules. The design in Fig. 9(a) and (b) would reduce R_{on} by engineering the stiffness and the contact area.

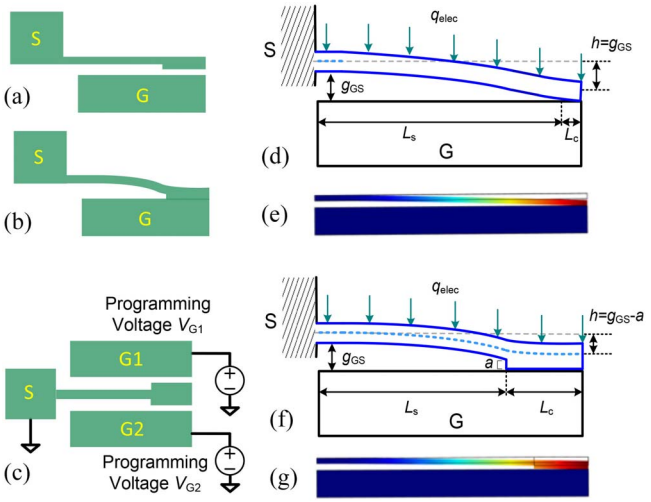


FIGURE 9. Improved NEMS antifuse design. (a) and (b) NEMS antifuse for reduced R_{on}. (c) Two-directional programmable NEMS antifuse for high security. (d) Original design: cantilever beam deflection profile at programmed state. (e) COMSOL simulation of beam deflection profile with applied gate voltage $V_G = 20$ V. (f) Improved design: hammerheaded cantilever beam deflection profile at programmed state. (g) COMSOL simulation of beam deflection profile with applied gate voltage $V_G = 20$ V.

For a uniformly shaped cantilever beam as in the original design, which is a commonly used mechanical structure in switching devices, the surface adhesion mechanism has been studied using various approaches. In general, for a MEMS/NEMS structure, the van der Waals force between the two contact surfaces plays a critical role in adhesion. Studies have linked adhesion force with surface roughness [30], and a first order estimation of the van der Waals force gives:

$$F_v = \frac{H_a \times A_c}{6\pi d^3}, \tag{6}$$

where F_v is the van der Waals force, H_a is the Hamaker constant, A_c is the contact area and d is the separation between the two surfaces. For a given device, the larger the contact area, the more adhesion there is between the two surfaces. Hence for the improved design we employ a hammerheaded shape for the cantilever beam to increase the contact area at the same amount of electrostatic load q_{elec}.

One approach to analyze cantilever deflections in correlation with the adhesion versus applied load is to analyze the energy release rate [31]. In this approach we study a cantilever beam in its adhered condition. Consider a uniform cantilever beam with no hammerhead in the original NEMS antifuse design, the external load q_{elec} is the electrostatic force that brings the cantilever tip to contact G, and the beam deflection profile is shown in Fig. 9(d). Using the definition in fracture mechanics, the unattached portion of the cantilever beam, which is the length from the clamping point to the contact point, can be seen as the crack length, L_s; and the length of the attached portion of the beam is the contact length, L_c. The adhesion energy is determined when the energy release rate reaches a static equilibrium with no external load applied:

$$G_o = \frac{18Dg_{GS}^2}{L_s^4} = \Gamma, \tag{7}$$

where Γ is the surface adhesion energy per unit area, g_{GS} is the as-fabricated gap between the cantilever and the gate electrode, and D the flexural rigidity per unit length, which is given by:

$$D = \frac{E_Y I}{t_B} = \frac{E_Y w_B^3}{12}, \tag{8}$$

where E_Y is the Young's modulus of the material and t_B is the thickness of the SiC film and thus the thickness of the cantilever. $I = w_B^3 t_B/12$ is the cantilever moment of inertia. The total contact area $A_c = L_c \times t_B$. Follow the analytical calculations in [30], the relationship between external uniform load and beam deflection profile can be described as follows: given the same Γ, higher q_{elec} is needed to reach smaller L_s and greater L_c for the uniform cantilever beam design.

The hammerheaded design illustrated in Fig. 9(f) gives the contact length L_c increases and crack length L_s decreases with the same electrostatic load q_{elec} compared to the uniform cantilever beam design. In other words, in order to reach the beam deflection that gives the same contact length L_c, the electrostatic load q_{elec} must be much higher in a uniform (no hammerhead) cantilever beam. If we define h as the distance the tip of cantilever beam has to deflect to make a contact with G, for uniform cantilever beam with no hammerhead, h equals the as-fabricated gap g_{GS}. In the case of cantilever beam of exact same dimension but with hammerhead, h becomes g_{GS}-a, where a is the depth of the hammerheaded structure. Because of the added structure that shortens the distance between the cantilever and G, the smallest electrostatic load required to make a contact reduces. To further illustrate the advantage of the hammerheaded design, we perform finite element method (FEM) simulation

in COMSOL on both designs under the same electrostatic load and the beam deflection profiles are shown in Fig. 9(e) and (g). Both designs use a cantilever beam of 10μm in length, 200nm in width and 500nm in thickness. The initial gap between the cantilever and gate electrode is g_{GS} = 250nm. The hammerhead depth a as illustrated in Fig. 9(f) is 100nm and the length is $L_c = 2\mu m$, with the same thickness t_B = 500nm as the cantilever beam. We keep all the simulation conditions the same and apply gate voltage $V_G = 20V$ to actuate both cantilevers. The improved design offers increased contact area under the same electrostatic load and thus lead to lower R_{on} and higher (more secured) adhesion.

We also propose improved design of NEMS with increased hardware security. One key to antifuse-based security is that once an antifuse is programmed to be connected, the state cannot change under potential attacks. One attack an antifuse may encounter is reprogramming. Conventional antifuses have two states: open circuit or conductive ('on'). This leaves an opportunity for hardware attackers to reprogram the unconnected antifuses and compromise the FPGA. Our NEMS antifuse design in Fig. 8(c) addresses this issue by introducing a third state. It has two gates for programming in both directions. In case an antifuse should remain open we can program it to permanently connect to the electrode on the other side; since this connection is irreversible, the device is thus immune to attempts of reprogramming attacks.

IV. NEMS ANTIFUSE FPGA DESIGN AND SIMULATION

In this Section we discuss antifuse-based FPGA designs and simulations using specialized software for the FPGA-based logic design, synthesis and simulation. We first compare NEMS antifuse with conventional antifuse at single device level, then we simulate and analyze the system-level power performance in a large scale system on FPGA platform.

A. COMPARISON WITH CONVENTIONAL ANTIFUSE

The NEMS antifuse is a promising candidate for OTP FPGA thanks to its minimal leakage power, high programming speed and small and scalable volume. Table 1 summaries the comparison of the SiC NEMS antifuse *versus* conventional antifuse at the single device level.

The leakage current I_{leak} for unprogrammed antifuses is obtained at the same conditions. The NEMS and conventional antifuses are both tested with stress voltage 3.6V at 25°C. Note that for some of the antifuse devices fabricated, some leakage comes from the defects in oxide layer, which can be alleviated by using high-quality oxide in the fabrication. The ultimate leakage current level for NEMS antifuse is well below the noise floor of the measurement system as demonstrated in Section III A.

The program voltage of NEMS antifuse can be as low as 1.6V and can be scaled down by scaling the actuation gaps. A <50nm actuation gap is reported [29] with similar fabrication techniques. The active area for NEMS antifuses is estimated by including the actuation gap, gate and cantilever beam width but no connecting pads, which are only needed for accessing individual devices in the characterization stage. For large network of antifuses the pads can be eliminated to achieve large scale integration. The device volume is calculated by multiplying the active area and the SiC film thickness.

TABLE 1. Comparison of NEMS antifuse and conventional antifuse.

Properties / Features	NEMS Antifuse	Conventional Antifuse
Leakage (Unprogrammed)*	≤0.1–1pA/μm² **	20pA/μm² [20]
Program Speed/Time	1–300ns/bit	50μs/bit–50ms/bit
Program Voltage	1–10V	5–15V [32]
Highest Current Density	10^5–10^6 A/cm²	800-10^6A/cm² [33]
Operating Temperature	25-500°C	25–125°C
Radiation Resistance	High	Low [17]
Active Area***	0.01–2.5μm²	0.4–21μm² [5]
Device Volume	0.01–1μm³	0.1–3μm³[34]
Bi-Directionality?	Yes	No

*Leakage current obtained with 3.6V stress voltage at 25°C.
**Ultimate leakage current is well below the noise floor of measurement system.
***Active areas do not include connecting pads for both NEMS and conventional devices, for fair comparison. NEMS cantilevers can already be made with dimensions of $L \times w \times t$=1μm×50nm×50nm and smaller, with coupling gaps <50nm [29].

B. PROGRAMMING SPEED/TIME

The programming time of the NEMS antifuse is the time it takes to switch the cantilever beam with applied programming voltage. We estimate the intrinsic speed of the NEMS cantilever switching time τ_s using equation (4) to be ~29ns for a switch with a fundamental-mode resonance at $f_0 = 20$MHz. Direct measurements of the resonance frequencies of the SiC NEMS and the speeds they make contacts with gate or drain electrodes are demonstrated in [35]. We drive the cantilever near its resonance frequency and monitor the tip contact optically with laser interferometry. We observe clear contact in the experiment when the device is operating at its resonance frequency and the measured resonance frequencies agree with the theoretical estimation. For example, a resonance frequency of $f_0 \approx 19.73$MHz has been measured from a 4μm-long SiC NEMS cantilever, with a switching time (or speed) of $\tau_s \approx 30$ns for making fast contact with a nearby electrode. If this device is used as an antifuse, its programming time would be $\tau_s \approx 30$ns. The programming speed can be further accelerated by scaling down the device dimensions and also by using higher gate overdrive.

The NEMS antifuse technology, at this early stage of research, has not yet advanced to large-scale arrays for system-level measurements. However, the unique properties of the NEMS antifuse make it a promising candidate for highly secure, harsh-environment adaptable logic applications including OTP FPGAs. Further study and improved designs of the NEMS structure will improve the function and reliability for large-scale integrated arrays of NEMS

antifuses, and future scaling is possible with the advances in fabrication technologies.

C. ANTIFUSE-BASED FPGA SIMULATION

We simulate and compare the performance from two types of antifuse-based FPGA circuits: one consists of conventional antifuses from a commercial manufacturer and the other uses NEMS antifuses. To simulate antifuse-based FPGA we first look into its basic architecture as illustrated in Fig. 1.

The computing functionality of FPGA is provided by its programmable logic blocks and these blocks connect to each other using programmable routing network. This programmable routing network provides routing connections among logic blocks and I/O blocks to implement any user-defined circuit. The routing interconnect of an FPGA consists of wires and programmable switches to form the required connection. These programmable switches are configured using the programmable technology such as static memory [36], flash [37] and antifuse [38].

FPGA programming and configuration are based on computer aided design (CAD). The application design is described in hardware description language (HDL), which is converted to a stream of bits and eventually programed on the FPGA, as shown in the flow in Fig. 10(a). The process of converting a circuit description into a format that can be loaded into an FPGA can be roughly divided into 5 distinct steps: synthesis, technology mapping, mapping, placement and routing. The FPGA CAD tools then generates a bitstream to output to the FPGA. The state of these bits configure the state of the memory bits in an FPGA and determines the logical function that the FPGA implements [1]. For varying requirements, a portion of FPGA can be partially reconfigured while the rest of an FPGA is still running. Future updates in the final product can be easily upgraded by simply downloading a new application bitstream file to the FPGA.

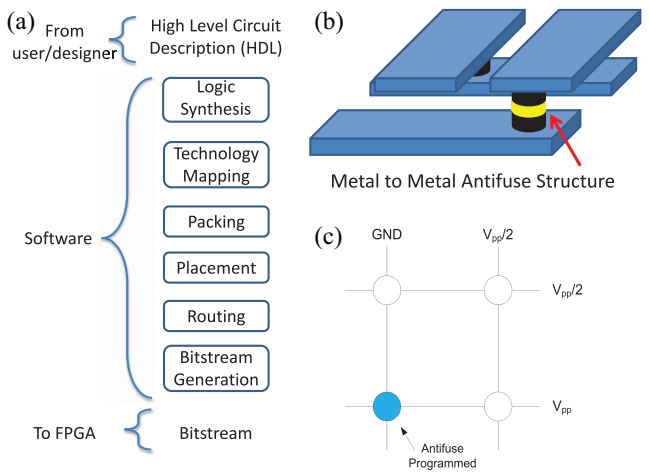


FIGURE 10. (a) Software flow of FPGA configuration. (b) Accelerator family interconnect elements. (c) Programming scheme of applied voltages on Accelerator family antifuses.

In our simulations, we focus on evaluating the *energy efficiency improvement* of NEMS antifuse FPGA. We simulate the power consumption with three benchmark circuits using a commercial antifuse FPGA and compare with the SiC NEMS antifuse FPGA. The commercial antifuse FPGA we choose is from the latest Axcelerator antifuse FPGA family offered by Microsemi, which is based upon 0.15μm and seven-layers of metal CMOS antifuse process technology. The Axcelerator family uses a metal-to-metal antifuse programmable interconnect element that resides between the upper two layers of metal [Fig. 10(b)]. This nonvolatile antifuse technology, known as FuseLock technology, provides excellent protection against design pirating and cloning. Fig. 10(c) shows the voltage scheme for programming the antifuses [13]. Consider an array of antifuses at the intersection of some horizontal and vertical segments, the programming of a certain antifuse is achieved by applying a programming voltage V_{pp} across it. This is done by precharging all segments to an intermediate voltage ($\sim V_{pp}/2$). Then a selected vertical segment is grounded and a selected horizontal segment is driven to V_{pp}. Other segments are left at $V_{pp}/2$. Only the single antifuse at the intersection of the selected segments sees the full V_{pp}.

For the logic function blocks, the Axcelerator family antifuse FPGA provides 2 types of logic modules: the register cell (R-cell) and the combinatorial cell (C-cell) as shown in Fig. 11 [39]. The inputs and outputs of the C-cell are as follows: D0-D3 are data inputs and A0, A1, B0, B1 are select inputs. User signals can be routed to any one of these inputs. Any of the C-cell inputs (D0-D3, A0, A1, B0 and B1) can be tied to clock signal as well. DB input can be used to drive a complement signal of any of the inputs to the C-cell. Carry connect signal can be used to combine two

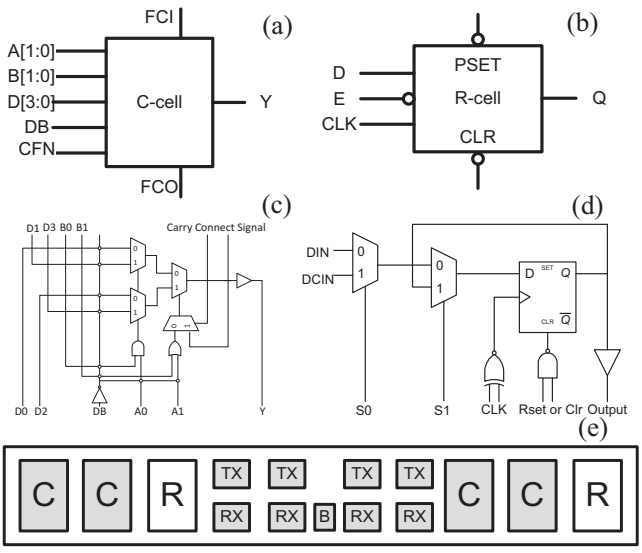


FIGURE 11. Schematic of the basic logic blocks in Axcelerator antifuse FPGA. (a) C-cell. (b) R-cell. (c) and (d) Gate-level schematics of C-cell and R-cell with input and output signals, respectively. (e) SuperCluster block comprised of C-cells, R-cells, and transmit and receive routing buffers, in Axcelerator antifuse FPGA.

TABLE 2. Power performance comparison of AX1000 FBGA896 between NEMS antifuse FPGA and conventional antifuse FPGA.

Power / Circuit	Number of Antifuses	Programming Power (whole circuit)		Operation Power* (all numbers are for the whole circuit simulated)					
				Switching ($P_{dyam} = C_{on} V_{DD}^2 f$)		Leakage ($P_{leak} = I_{off} \times V_{DD}$)		Total ($P_{tot} = P_{dyna} + P_{leak}$)	
		Conventional Antifuse $P_{prog} = I_{pp} \times V_{pp}$	NEMS Antifuse $P_{prog} = \frac{C_{act} V_{PI}^2}{2\tau_s}$	Conventional Antifuse	NEMS Antifuse	Conventional Antifuse	NEMS Antifuse**	Conventional Antifuse	NEMS Antifuse
C432	684	34–103mW	0.7–34μW	0.182–0.223mW	0.68–0.75μW	0.17–18.9μW	1.71–1.89nW	0.35–19.12mW	~0.68–0.75μW
C880	1299	65–195mW	1.5–65μW	0.348–0.425mW	1.3–1.43μW	0.17–18.85μW	1.70–1.85nW	0.51–19.28mW	~1.3–1.43μW
C1908	1200	60–120mW	1.4–60μW	0.808–0.987mW	1.2–1.32μW	0.17–18.86μW	1.70–1.86nW	0.97–19.84mW	~1.2–1.32μW

*Operation power simulated with operating voltage V_{DD} = 1.425V—1.575V, and operation frequency f = 1MHz.
**Ultimate leakage current is well below the noise floor of measurement system.

C-cell to fulfill up to 5 inputs logic function. Y is the output of the cell.

The R-cell contains a flip-flop featuring asynchronous clear, asynchronous preset and active-low enable control signals. Two C-cells, a single R-cell, two Transmit (TX), and two Receive (RX) routing buffers form a Cluster, while two Clusters comprise a SuperCluster [Fig. 11(e)], which works as the fundamental logic blocks in Axcelerator antifuse FPGAs [39].

To simulate the performance of antifuse FPGA, we use Libero IDE integrated design manager which integrates design tools while guiding users through the design software flow. The selected device in this simulation is AX1000 with FPGA 896 package silicon die. Fig. 12 shows the FPGA architecture with interconnects and the simulated circuits. Firstly, the benchmark circuit verilog HDL file is generated in HDL Editor. In this work, ISCAS-85 C432, C880 and C1908 are generated as the benchmark circuits by verilog HDL. Then the benchmark source file is compiled by a synthesis tool, Synplify, where the Netlist of the benchmark is synthesized. After the synthesis, a post-synthesis file is generated which can be used in the next step for mapping, placement and routing. Mapping and placement involve deciding where to place all components, circuitry, and logic elements in a constraint space. This is followed by routing, which decides the exact design of all the wires needed to connect the placed components. In this step, the proper interconnections of antifuses are selected to be programmed based on the timing and area restriction and optimization. This step must also implement all the desired connections while following the rules and limitations of the manufacturing process. After the placement, the detailed power consumption of the design implemented FPGA can be obtained from the power report generated by the simulation tool. A list of antifuses to be programmed is generated and downloaded to a programming station into which the FPGA is plugged. Number of antifuses is counted after the routing to derive the total power consumption, the details are discussed in the next subsection.

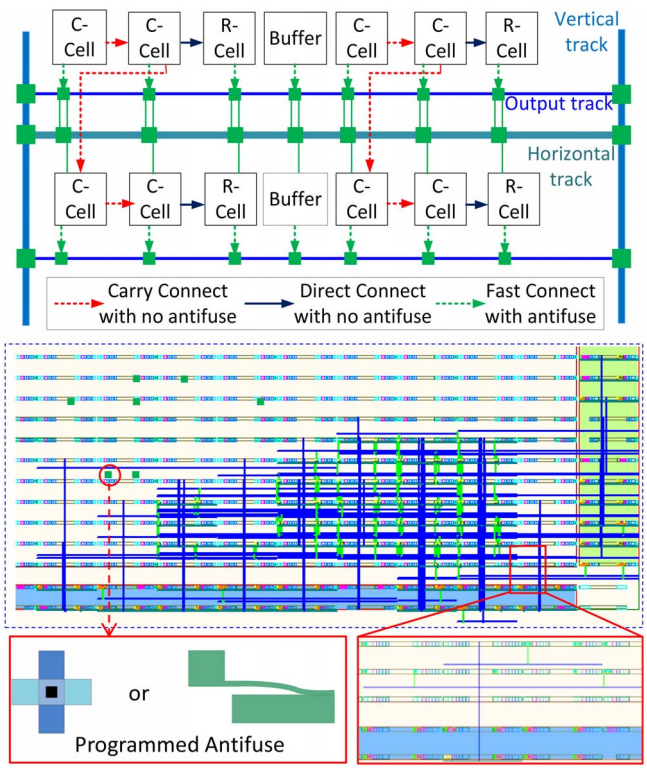


FIGURE 12. Axcelerator antifuse FPGA simulation architecture and layout of a mapped benchmark circuit. Top: FPGA architecture with logic modules (C-cell, R-cell, and buffers) and interconnections. Middle: layout of benchmark circuit after routing. Bottom: zoomed-in view of the programmed antifuses and circuit connections.

We examine the programming and operation power (both switching and leakage) with the three benchmark circuits. C432 is a 27 channel interrupt controller with 36 inputs, 7 outputs and 160 gates. C880 is an 8-bit arithmetic logic unit (ALU) with 60 inputs, 26 outputs and 383 gates. C1908 is a 16-bit single-error-correcting and double-error-detecting circuit with 33 inputs, 25 outputs and 880 gates. Considering the number of antifuse interconnection must be counted manually, the number of the gates of each benchmark cannot be too large. Table 2 describes the power consumption comparison between the antifuse interconnection and NEMS interconnection. The power consumption data of operation power of antifuse interconnection are obtained by the previously described process utilizing the 3 different ISCAS-85 benchmark circuits. The software utilized to analyze the power consumption called SmartPower is provided by Microsemi development software. By choosing the environment conditions, such as the temperature and supply voltage, we can obtain the power consumption taking into account of the working environment influence.

D. POWER ANALYSIS OF NEMS VERSUS CONVENTIONAL ANTIFUSE FPGA

NEMS antifuse provides great advantage in power consumption for FPGAs. The conventional antifuse FPGA used in the simulation is AX1000 896 from Axcelerator family manufactured by Microsemi. It uses metal-to-metal antifuses and the programming process is given in [40]. A programming pulse with voltage V_{prg} (5–15V) is applied while monitoring the current I_{prg}. This current is typically $< 10\mu$A until an antifuse is programmed. So the programming power for conventional antifuse is estimated to be

$$P_{\text{prg}} = I_{\text{prg}} V_{\text{prg}}, \tag{9}$$

where I_{prg} and V_{prg} are the programming current and voltage, respectively. The calculated programming power for conventional antifuse is 50–150μW per antifuse.

The programming power for NEMS antifuse is the power needed to capacitively actuate the NEMS cantilever:

$$P_{\text{prg}} = C_{\text{act}} V_{\text{PI}}^2/(2\tau_s). \tag{10}$$

Here C_{act} is the actuation capacitance given by

$$C_{act} = \varepsilon A_{act}/g_{GS} \tag{11}$$

where A_{act} is the actuation area, given by

$$A_{act} = t_B L_G. \tag{12}$$

For a NEMS cantilever beam with $L_{\text{B}} \times w_{\text{B}} \times t_{\text{B}} = 10\mu\text{m} \times 200\text{nm} \times 500\text{nm}$, actuation gate length $L_{\text{G}} = 10\mu$m, as-fabricated gap $g_{\text{GS}} = 300$nm, and $V_{\text{PI}} = 1.5$V–10V, the switching time $\tau_s \approx 150$ns and the programming power is $P_{prg} \approx 1.125$–50nW.

The switching power for conventional antifuse FPGA for the given benchmark circuits is given by the simulation tool SmartPower, with the following conditions to operate the logic circuits: operation voltage $V_{\text{DD}} = 1.425$–1.575V and clock frequency $f_{\text{clock}} = 1$MHz.

To estimate the dynamic power of NEMS antifuses, we model each programmed NEMS antifuse with a resistor R_{on} and a capacitor C_{on}. R_{on} is the on-resistance of the programmed NEMS antifuse and C_{on} is the capacitor formed by the SiC layer of the NEMS antifuse, the oxide beneath the SiC layer and the Si substrate. C_{on} can be estimated by:

$$C_{\text{on}} = \frac{\varepsilon_{\text{ox}}}{t_{\text{ox}}}\frac{2}{3}A_{\text{NEMS}} + \frac{\varepsilon_{air}}{t_{\text{ox}}}\frac{1}{3}A_{\text{NEMS}}, \tag{13}$$

where A_{NEMS} is the NEMS antifuse active area, ε_{ox}. and ε_{air} are the dielectric permittivity of SiO_2 and air, respectively, and t_{ox} is the thickness of the oxide layer. The capacitor is formed by two fixed terminals (G and S) with oxide underneath in parallel with a suspended cantilever beam, with only air between the beam and the substrate. Each of the terminals accounts for 1/3 of the active area of the NEMS antifuse and the cantilever beam is the rest 1/3 of the active area. Switching power for NEMS antifuse is

$$P_{\text{dyn}} = C_{\text{on}} V_{\text{DD}} f_{\text{clock}}. \tag{14}$$

Using the same V_{DD} and f_{clock}, we calculate the switching power per antifuse to be $P_{\text{dyn}} \approx 1$–1.1nW. The switching power of the benchmark circuits are calculated using $P_{\text{dyn}} \times$ number of programmed antifuses.

Leakage power comes from all the *unprogrammed* antifuses in the FPGA. The leakage power for each unprogrammed antifuse is given by

$$P_{\text{leak}} = I_{\text{off}} V_{\text{DD}}. \tag{15}$$

Using $I_{\text{off}} = 1$–100pA [20], and same V_{DD} as in the simulations mentioned above, we have $P_{\text{leak}} = 1.425$–157.5pW for the conventional antifuse. The AX1000 has $\sim 29 \times 10^6$ antifuse switches and 9 logic tiles. Assuming we only use one logic tile, the number of antifuses is $\sim 3 \times 10^6$. So the unprogrammed antifuse is 3×10^6 minus the number of programed antifuses for each benchmark circuit.

We use the noise floor of the measurement system as the upper limit of leakage current for NEMS antifuses. So with $I_{\text{off}} = 10$fA and same V_{DD}, the leakage power per NEMS antifuse is $P_{\text{leak}} = I_{\text{off}} V_{\text{DD}} = 14.25$–15.75fW. The number of unprogrammed antifuses is the same as in the case of the conventional FPGA.

V. ANTIFUSE-BASED OTP MEMORY CELL

The logic blocks in Microsemi antifuse FPGA are based on multiplexers and with one AND gate and one OR gate. With the combination of multiplexer based circuit and the carry connect signal communicating with another C-cell, the Axcelerator device can implement more than 4,000 combinatorial functions of up to five inputs. For the configuration bits of C-cell, a low resistance permanent link is formed to connect the configuration lines to either logic '0' or logic '1' [39], [41].

However, there is another way of construct basic logic blocks for FPGAs by using a configurable logic block (CLB), made of NAND gates [42], an interconnection of multiplexors [43], and lookup tables (LUTs) [44]. Commercial vendors use LUT-based CLBs to provide basic logic and storage functionality. A CLB can comprise of a single basic logic element or a cluster of locally connected basic logic elements. The CLBs in an FPGA are arranged in a two dimensional grid and are interconnected by programmable routing resources [45]. I/O blocks are arranged at the periphery of the grid and they are also connected to the programmable routing interconnect. Logic blocks based on LUTs can implement much more complex combinational functions with much more inputs than the ones based on multiplexer. Furthermore LUT-based logic blocks provide a good trade-off between too fine-grained and too coarse-grained logic blocks. Static random

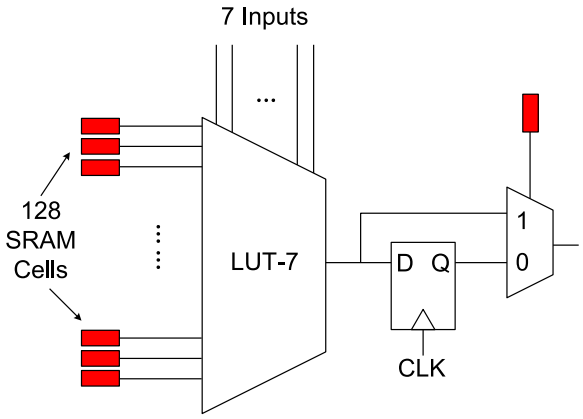


FIGURE 13. **LUT-based C-cell logic blocks. The red blocks are the memory cells used to store information which can be replaced by NEMS OTP memory.**

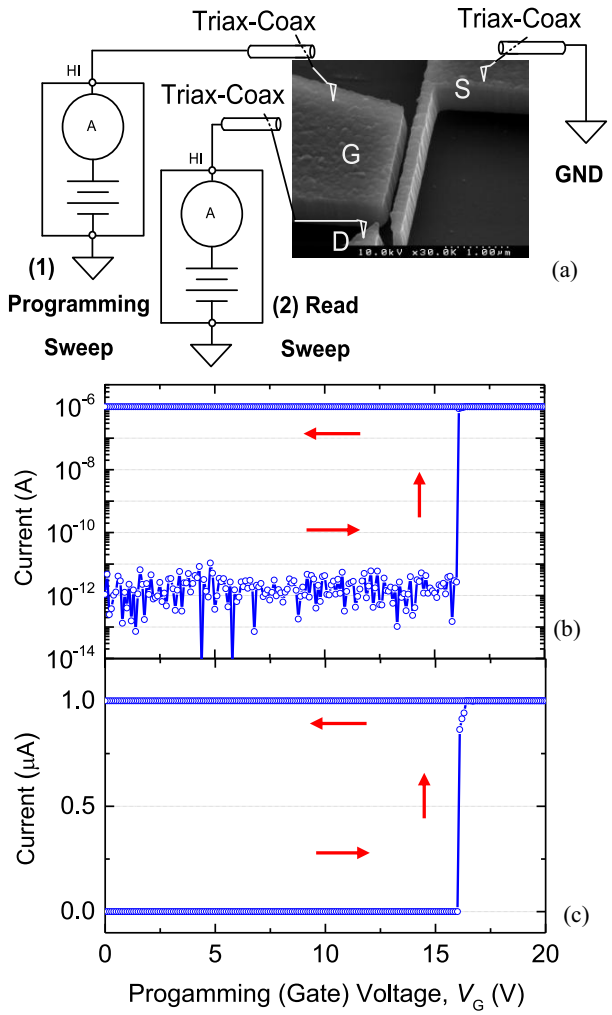


FIGURE 14. **NEMS OTP memory cell characterization. (a) Programming scheme (1) and read scheme (2). Inset of (a) is an SEM image of a NEMS OTP memory cell. (b) and (c) Programming cycle through G with scheme (1), with current in logarithmic and linear scale, respectively.**

access memory (SRAM) cells are used to store the data in the LUTs that are typically used in SRAM-based FPGAs to implement logic functions, as shown in Fig. 13. However, SRAMs are volatile and need to be reprogrammed each time before implement the logic functions which is not compatible in the antifuse FPGA.

If the SRAM can be replaced by OTP memory, then the logic blocks based on LUT is a potential candidate to work as the logic blocks implemented in OTP FPGA – in our case is the SiC NEMS antifuse FPGA. The OTP memory can be programmed in the same time with the interconnects. SiC NEMS antifuses can be used to build OTP memory cell in a three-terminal configuration. In this configuration we use G to program the memory cell and D to read out the state of the memory. If we define the initial state before programming as '1', in which the cantilever beam is not contacting D, the programmed state is '0', where the cantilever stays connected with D. A read sweep on an unprogrammed memory cell will exhibit high resistance and thus very small current, and the same read sweep on a programmed memory cell will have low resistance and a high current.

Fig. 14 presents this configuration and the electrical characterization of the memory cell. At its initial state, the memory cell remains unprogrammed, defined as logic '1'. The programming of the memory cell is through G, as shown in scheme (1) in Fig. 14(a). Upon programming, the state of the memory cell changes to logic '0', which can be verified by the reading through D in scheme (2). Data from the programming cycle through G in Fig. 14(b) and (c) show that this NEMS antifuse-based OTP memory cell is programmed at $V_{PI} = 15.3$V.

The read current margin between programmed state and unprogrammed state of the same NEMS antifuse OTP memory is measured to be more than 6 orders of magnitude, as shown in Fig. 15. This broad range provides the circuit designer with a good design margin by using NEMS antifuse-based OTP memory cell.

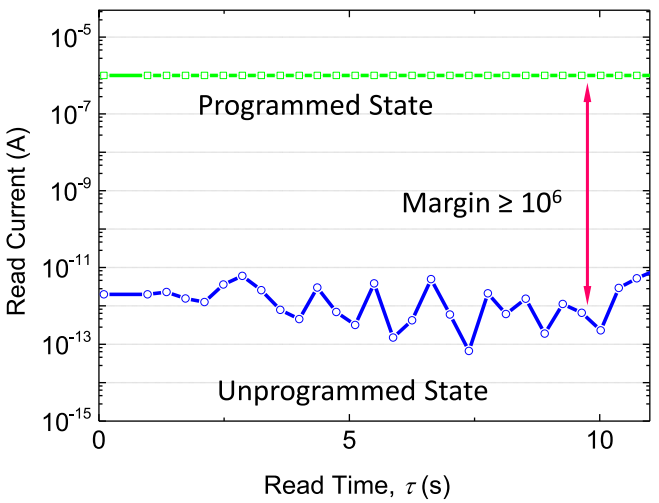


FIGURE 15. **Read sweep before/after programming cycle in Fig. 14(b) and (c) showing large read margin between the programmed/unprogrammed memory cell, exhibiting a read margin of over six orders of magnitude.**

VI. CONCLUSION

We have developed a new type of robust antifuse based on SiC NEMS to serve as interconnect building blocks for secure OTP FPGAs. We demonstrate the devices can have very low programming voltage, high current density, abrupt switching, small footprints and long term stability. Simulations of benchmark circuits with a commercial antifuse FPGA design tool show that NEMS antifuses offer significant improvement in power consumption. New designs and dimensional scaling will further enable smaller devices with higher performance. We have also demonstrated, with experiments, a SiC NEMS-based OTP memory cell with separated programming and reading path, and $>10^6$ current margin for reading. The unique properties and advantages of NEMS antifuse open many opportunities for designing low-power, high security and harsh environment operable logic circuits and systems.

With the advancement in fabrication technology, integration of the emerging NEMS devices with mainstream integrated circuit technologies is feasible and promising. We envision the following strategies in realizing a secure NEMS antifuse-based FPGA. First, with the 'MEMS/NEMS-first' approach [46], it is possible to fabricate NEMS antifuses as interconnects and have the logic modules implemented in conventional CMOS technology, as simulated in this work. For fully high-temperature circuits, SiC JFET technology can be employed to realize the logic functions [47], [48], with the options of either integrating the JEFT chip with the NEMS chip in a flip-chip or system-in-package (SiP) fashion, or monolithic co-fabrication of SiC JFETs and NEMS building blocks on the same chip. Furthermore, we have already demonstrated fundamental logic functions realized by NEMS switching devices [25], leading to the possibility of an entirely mechanical reconfigurable logic paradigm that utilizes NEMS to realize logic functions, and NEMS antifuses as interconnects. This solution, once fully realized, shall provide high security for high temperature and harsh environment operations.

REFERENCES

[1] U. Farooq, Z. Marrakchi, and H. Mehrez, "FPGA architectures: An overview," in *Tree-based Heterogeneous FPGA Architectures*. New York, NY, USA: Springer, 2012, pp. 7–48.

[2] C. Hu, "Interconnect devices for field programmable gate array," in *IEDM Tech. Dig.*, San Francisco, CA, 1992, pp. 591–594.

[3] E. Hamdy *et al.*, "Dielectric based antifuse for logic and memory ICs," in *IEDM Tech. Dig.*, San Francisco, CA, USA, 1988, pp. 786–789.

[4] D. Liu, K. Chen, H. Tigelaar, J. Paterson, and S. Chen, "Scaled dielectric antifuse structure for field-programmable gate array applications," *IEEE Electron Device Lett.*, vol. 12, no. 4, pp. 151–153, Apr. 1991.

[5] S. Wang, G. Misium, J. Camp, K. Chen, and H. Tigelaar, "High performance metal/silicide antifuse," *IEEE Electron Device Lett.*, vol. 13, no. 9, pp. 471–473, Sep. 1992.

[6] G. Zhang *et al.*, "An electro-thermal model for metal-oxide-metal antifuses," *IEEE Trans. Electron Devices*, vol. 42, no. 8, pp. 1548–1558, Aug. 1995.

[7] F. Li, X. Yang, A. T. Meeks, J. T. Shearer, and K. Y. Le, "Evaluation of SiO_2 antifuse in a 3D-OTP memory," *IEEE Trans. Device Mater. Rel.*, vol. 4, no. 3, pp. 416–421, Sep. 2004.

[8] J. Kim and K. Lee, "Three-transistor one-time programmable (OTP) ROM cell array using standard CMOS gate oxide antifuse," *IEEE Electron Device Lett.*, vol. 24, no. 9, pp. 589–591, Sep. 2003.

[9] H.-K. Cha *et al.*, "A 32-KB standard CMOS antifuse one-time programmable ROM embedded in a 16-bit microcontroller," *IEEE J. Solid-State Circuits*, vol. 41, no. 9, pp. 2115–2124, Sep. 2006.

[10] H. Ito and T. Namekawa, "Pure CMOS one-time programmable memory using gate-ox anti-fuse," in *Proc. IEEE Custom Integr. Circuits Conf.,* Orlando, FL, USA, 2004, pp. 469–472.

[11] S. Trimberger, "Effects of FPGA architecture on FPGA routing," in *Proc. ACM/IEEE Design Autom. Conf.*, San Francisco, CA, USA, 1995, pp. 574–578.

[12] S. Wong, H. So, J. Ou, and J. Costello, "A 5000-gate CMOS EPLD with multiple logic and interconnect arrays," in *Proc. Custom Integr. Circuits Conf.*, San Diego, CA, USA, 1989, pp. 5.8.1–5.8.4.

[13] J. Greene, E. Hamdy, and S. Beal, "Antifuse field programmable gate arrays," *Proc. IEEE*, vol. 81, no. 7, pp. 1042–1056, Jul. 1993.

[14] S. Chiang *et al.*, "Antifuse structure comparison for field programmable gate arrays," in *IEDM Tech. Dig.*, San Francisco, CA, USA, 1992, pp. 611–614.

[15] S. Hauck, "The roles of FPGAs in reprogrammable systems," *Proc. IEEE*, vol. 86, no. 4, pp. 615–638, Apr. 1998.

[16] *Understanding Actel Antifuse Device Security*, Microsemi Corp., Aliso Viejo, CA, USA, Jan. 2004. [Online]. Available: http://www.actel.com/documents/AntifuseSecurityWP.pdf

[17] R. Katz *et al.*, "Radiation effects on current field programmable technologies," *IEEE Trans. Nucl. Sci.*, vol. 44, no. 6, pp. 1945–1956, Dec. 1997.

[18] R. Katz *et al.*, "Current radiation issues for programmable elements and devices," *IEEE Trans. Nucl. Sci.*, vol. 45, no. 6, pp. 2600–2610, Dec. 1998.

[19] S. Chiang *et al.*, "Oxide-nitride-oxide antifuse reliability," in *Proc. IEEE Rel. Phy. Symp.*, New Orleans, LA, USA, 1990, pp. 186–192.

[20] C.-C. Shih *et al.*, "Characterization and modeling of a highly reliable metal-to-metal antifuse for high-performance and high-density field-programmable gate arrays," in *Proc. IEEE Rel. Phys. Symp.*, Denver, CO, USA, 1997, pp. 25–33.

[21] S. S. Cohen, J. I. Raffel, and P. W. Wyatt, "A novel double-metal structure for voltage-programmable links," *IEEE Electron Device Lett.,* vol. 13, no. 9, pp. 488–490, Sep. 1992.

[22] J. S. Lee and Y. H. Lee, "Metal-to-metal antifuse with amorphous Ti-rich barium titanate film and silicon oxide film," *Solid-State Electron.*, vol. 43, pp. 469–472, Mar. 1999.

[23] *Reliability of the Amorphous Silicon Antifuse*, Quick Logic, Sunnyvale, CA, USA, 2008.

[24] K. Gordon and R. Wong, "Conductive filament of the programmed metal electrode amorphous silicon antifuse," in *IEDM Tech. Dig.*, Washington, DC, USA, 1993, pp. 27–30.

[25] T. He *et al.*, "Silicon carbide (SiC) nanoelectromechanical switches and logic gates with long cycles and robust performance in ambient air and at high temperature," in *Proc. Int. Electron Devices Meeting*, Washington, DC, USA, 2013, paper no. 4.6, pp. 108–111.

[26] K. Akarvardar and H.-S. P. Wong, "Ultralow voltage crossbar nonvolatile memory based on energy-reversible NEM switches," *IEEE Electron Device Lett.*, vol. 30, no. 6, pp. 626–628, Jun. 2009.

[27] V. Ranganathan *et al.*, "Nanomechanical non-volatile memory for computing at extreme," in *Proc. IEEE/ACM Int. Nanoscale Archit. Symp.*, Brooklyn, NY, USA, 2013, pp. 44–45.

[28] J. Trevino, X.-A. Fu, M. Mehregany, and C. A. Zorman, "Low-stress, heavily-doped polycrystalline silicon carbide for MEMS applications," in *Proc. IEEE 18th Int. Conf. Micro Electro Mech. Syst. (MEMS)*, Miami, FL, USA, 2005, pp. 451–454.

[29] X. L. Feng, M. H. Matheny, C. A. Zorman, M. Mehregany, and M. L. Roukes, "Low voltage nanoelectromechanical switches based on silicon carbide nanowires," *Nano Lett.*, vol. 10, no. 8, pp. 2891–2896, Aug. 2010.

[30] F. W. Delrio *et al.*, "The role of van der Waals forces in adhesion of micromachined surfaces," *Nat. Mater.*, vol. 4, pp. 629–634, Aug. 2005.

[31] J. A. Knapp and M. P. de Boer, "Mechanics of microcantilever beams subject to combined electrostatic and adhesive forces," *IEEE J. Microelectromech. Syst.*, vol. 11, no. 6, pp. 754–764, Nov. 2002.

[32] E. Ebrard, B. Allard, P. Candelier, and P. Waltz, "Review of fuse and antifuse solutions for advanced standard CMOS technologies," *Microelectron. J.*, vol. 40, no. 12, pp. 1755–1765, Dec. 2009.

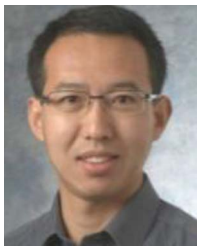
[33] M. Crowley *et al.*, "512 Mb PROM with 8 layers of antifuse/diode cells," in *ISSCC Tech. Dig.*, San Francisco, CA, USA, 2003, pp. 284–493.
[34] N. Patil, D. Das, E. Scanff, and M. Pecht, "Long term storage reliability of antifuse field programmable gate arrays," *Microelectron. Rel.*, vol. 53, no. 12, pp. 2052–2056, Dec. 2013.
[35] T. He, J. Lee, Z. Wang, and P. X.-L. Feng, "Interrogating contact-mode silicon carbide (SiC) nanoelectromechanical switching dynamics by ultrasensitive laser interferometry," in *Proc. IEEE 27th Int. Conf. Micro Electro Mech. Syst.* (*MEMS*), San Francisco, CA, USA, 2014, pp. 1079–1082.
[36] W. Carter, K. Duong, R. Freeman, and S. Sze, "A user programmable reconfiguration gate array," in *Proc. IEEE Custom Integr. Circuits Conf.*, Rochester, NY, USA, 1986, pp. 233–235.
[37] D. C. Guterman, I. H. Rimawi, T. L. Chiu, R. Halvorson, and D. McElroy, "An electrically alterable nonvolatile memory cell using a floating-gate structure," *IEEE Trans. Electron Devices*, vol. 26, no. 4, pp. 576–586, Apr. 1979.
[38] J. Birkner *et al.*, "A very-high-speed field programmable gate array using metal-to-metal antifuse programmable elements," *Microelectron. J.*, vol. 23, no. 7, pp. 561–568, Nov. 1992.
[39] *Axcelerator Family FPGA Datasheet*, Microsemi *Corp., Aliso Viejo*, CA, USA, Mar. 2012.
[40] *Testing and Programming Actel Field Programmable Gate Arrays (FPGAs)*, Actel Corp., Mountain View, CA, USA, Apr. 1996.
[41] I. Kuon, R. Tessier, and J. Rose, "FPGA architecture: Survey and challenges," *Found. Trends Electron. Design Autom.*, vol. 2, pp. 135–253, Feb. 2008.
[42] *ERA60100 Electrically Reconfigurable Array Data Sheet*, Plessey Semicond. Ltd., Devon, U.K., 1990.
[43] A. E. Gamal, "An architecture for electrically configurable gate arrays," *IEEE J. Solid-State Circuits*, vol. 24, no. 2, pp. 394–398, Apr. 1989.
[44] H. Michinishi, *et al*., "A test methodology for interconnect structures of LUT-based FPGAs," *Proc. 5th Asian Test Symp.*, Hsinchu , Taiwan, 1996, pp. 68–74.
[45] V. Betz, A. Marquardt, and J. Rose, *Architecture and CAD for Deep-Submicron FPGAs*. New York, NY, USA: Kluwer Academic, Jan. 1999.
[46] G. K. Fedder, R. T. Howe, T.-J. K. Liu, and E. P. Quevy, "Technologies for cofabricating MEMS and electronics," *Proc. IEEE*, vol. 96, no. 2, pp. 306–322, Feb. 2008.
[47] C.-W. Soong, A. C. Patil, S. L. Garverick, X. Fu, and M. Mehregany, "550°C integrated logic circuits using 6H-SiC JFETs," *IEEE Electron Device Lett.*, vol. 33, no. 10, pp. 1369–1371, Oct. 2012.
[48] S. L. Garverick, C.-W. Soong, and M. Mehregany, "SiC JFET integrated circuits for sensing and control at temperatures up to 600°C," in *Proc. IEEE EnergyTech*, Cleveland, OH, USA, May 2012, pp. 1–6.

TINA HE (S'10) received the B.S. degree in electrical engineering from Fudan University, Shanghai, China, in 2008. She is currently pursuing the Ph.D. degree with the Department of Electrical Engineering and Computer Science, Case Western Reserve University, Cleveland, OH, USA. Her research interests include design, fabrication, and electrical measurement of SiC NEMS/MEMS for harsh environment applications. She was a recipient of the Keithley Graduate Fellowship in 2012 and the Best Student Paper Award at the IEEE 8th International Conference on Nano/Micro Engineered and Molecular Systems 2013 (NEMS 2013).

FENGCHAO ZHANG (S'13) received the B.E. degree in optoelectronic information engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2012. He is currently pursuing the Ph.D. degree with the Department of Electrical Engineering and Computer Science, Case Western Reserve University. His research interests include hardware security, ultralow-power very large-scale integration design and architecture, nanoelectromechanical systems-based logic devices, circuits, and architectures.

SWARUP BHUNIA (S'01–M'06–SM'11) received his B.E. (Hons.) from Jadavpur University, Kolkata, India, and the M.Tech. degree from the Indian Institute of Technology (IIT), Kharagpur. He received his Ph.D. from Purdue University, IN, USA, in 2005.

He is the T. and A. Schroeder Associate Professor of Electrical Engineering and Computer Science, Case Western Reserve University. He has worked in the semiconductor industry on RTL synthesis, verification, and low-power design for about three years. His research interests include low power and robust design, hardware security and protection, adaptive nanocomputing, and novel test methodologies. He has over ten years of research and development experience with over 150 publications in peer-reviewed journals and premier conferences in the area of very large-scale integration design (VLSI), CAD, and test techniques. He was a recipient of the IBM Faculty Award (2013), the NSF CAREER Award (2011), the SRC Technical Excellence Award (2005), and the Best Paper Award at VLSI Design 2012, ICCD 2004, LATW 2003, ASP-DAC 2006, and Hardware Oriented Trust and Security Symposium (HOST) 2010. He has served as a Guest Editor for the IEEE Design & Test of Computers (2010). He has served on the Editorial Board of the *Journal of Low Power Electronics* and the Technical Program Committee of Design Automation and Test in Europe (2006)–(2010), Hardware Oriented Trust and Security Symposium (2008)–(2010), the IEEE/IFIP International Conference on VLSI (VLSI SOC 2008), Test Technology Educational Program (2006)–(2008), the International Symposium on Low Power Electronics and Design (2007) and (2008), the IEEE/ACM Symposium on Nanoscale Architectures (2007)–(2010), and the IEEE International Conference on VLSI (ISVLSI 2008–2010), the International Conference of VLSI Design as a Track Chair (2010), and the Program Committee of the International Online Test Symposium (2005). He has given tutorials on low-power and robust design and test in premier conference including the International Test Conferences (2009), VLSI Test Symposium (2010), and Design Automation and Test in Europe (2009).



PHILIP X.-L. FENG (S'03–M'08–SM'14) is an Assistant Professor in Electrical Engineering & Computer Science at Case School of Engineering, Case Western Reserve University. His research is primarily focused on nanoscale devices and systems. Prior to joining the faculty at Case, Feng was with the Kavli Nanoscience Institute, California Institute of Technology (Caltech), where he served as a Staff Scientist and a Co-Principal Investigator from 2007 to 2010. He received his Ph.D. from Caltech in 2007 for developing ultra high frequency (UHF) nanoelectromechanical systems (NEMS) with low-noise technologies for real-time single-molecule sensing. Feng was one of the 81 young engineers selected to participate in the National Academy of Engineering (NAE) 2013 U.S. Frontier of Engineering (USFOE) Symposium. Subsequently, he received the NAE Grainger Foundation Frontiers of Engineering (FOE) Award in 2014. He is an recipient of the National Science Foundation CAREER Award and 3 Best Paper Awards (together with his advisees, at *IEEE NEMS* 2013, *IEEE Int. Freq. Control Symp.* 2014, and *AVS Int. Symp*. 2014). He is also the recipient of the Case School of Engineering Graduate Teaching Award (2014), and the Case School of Engineering Research Award (2015). His other recent awards and honors include the Mihajlo Mesarovic Award (2013), a T. Keith Glennan Fellowship (2012), an Innovative Incentive Award (2011), and a finalist for the Draper-Fisher-Jurvetson (DFJ) Venture Challenge (2007). He has been serving on the Technical Program Committees for *IEEE IEDM*, *IEEE MEMS*, *Transducers, IEEE IFCS, IEEE NANO*; and he has also served as the Technical Program Chair for the MEMS/NEMS Technical Group at the 61*st* and 62*nd* *American Vacuum Society (AVS) International Symposium & Exhibition*.