Advancing Hardware Security Using Polymorphic and Stochastic Spin-Hall Effect Devices

Satwik Patnaik[†*], Nikhil Rangarajan[†*], Johann Knechtel[‡], Ozgur Sinanoglu[‡], and Shaloo Rakheja[†]
[†] Tandon School of Engineering, New York University, New York, USA
[‡] Division of Engineering, New York University Abu Dhabi, Abu Dhabi, United Arab Emirates
{sp4012, nikhil.rangarajan, johann, ozgursin, shaloo.rakheja}@nyu.edu

***Abstract*—Protecting intellectual property (IP) in electronic circuits has become a serious challenge in recent years. Logic locking/encryption and layout camouflaging are two prominent techniques for IP protection. Most existing approaches, however, particularly those focused on CMOS integration, incur excessive design overheads resulting from their need for additional circuit structures or device-level modifications. This work leverages the innate polymorphism of an emerging spin-based device, called the giant spin-Hall effect (GSHE) switch, to simultaneously enable locking and camouflaging within a single instance. Using the GSHE switch, we propose a powerful primitive that enables cloaking all the 16 Boolean functions possible for two inputs. We conduct a comprehensive study using state-of-the-art Boolean satisfiability (SAT) attacks to demonstrate the superior resilience of the proposed primitive in comparison to several others in the literature. While we tailor the primitive for deterministic computation, it can readily support stochastic computation; we argue that stochastic behavior can break most, if not all, existing SAT attacks. Finally, we discuss the resilience of the primitive against various side-channel attacks as well as invasive monitoring at runtime, which are arguably even more concerning threats than SAT attacks.**

I. Introduction

With the advent of globalization affecting the design and manufacturing process of integrated circuits (ICs), hardware security has emerged as a critical concern. The exposure to various adversaries, which may reverse engineer (RE) ICs, counterfeit them, steal their intellectual property (IP), inject hardware Trojans, leak and/or extract sensitive data at runtime has escalated [1]. Next, we briefly review IP protection schemes and attacks in general.

IC camouflaging seeks to mitigate RE attacks, wherein the layout-level appearance of the IC is altered such that it becomes intractable to decipher its underlying functionality and IP. For CMOS integration, various techniques have been proposed, e.g., look-alike gates [2], threshold-dependent camouflaging [3], [4], and obfuscated interconnects [5].

Logic locking/encryption obfuscates the IP functionality rather than the device-level layout [6], [7]. The so-called key gates are carefully tailored into the IP/chip, where only the correct key can "unlock" the original functionality.

Analytical attacks targeting camouflaged (or locked) ICs were initially introduced in [8], [9]. These attacks are based on Boolean satisfiability (SAT) and the fact that a small set of discriminating input patterns (DIPs) may suffice to infer the camouflaged functionality (or locking key). Several SAT-attack resilient techniques were recently proposed [6], [7], [10]; however, most of these techniques are still vulnerable to advanced analytical attacks such as [11]–[13].

Physical attacks range from non-invasive (e.g., power side-channel attacks) and semi-invasive (e.g., localized fault-injection attacks) to invasive attacks (e.g., RE, microprobing the frontside/backside) [14]. Such attacks are also promising for extracting sensitive data at runtime, even from secured chips, e.g., [15], [16].

Emerging devices including, e.g., nanowire transistors, carbon-based or spin-based devices, may offer lower power dissipation and higher integration density compared to their CMOS counterparts [17]. Additionally, emerging devices can augment the CMOS technology to improve hardware security [18]–[20]. The most promising aspect of many emerging devices is *polymorphism*: a polymorphic gate can readily implement different Boolean functions at runtime, where the functionality is determined by an internal/external control mechanism [20]. It is important to note that polymorphic gates can inherently support both camouflaging and locking due to the following reasons. First, owing to their uniform device-level layout, the actual function of a polymorphic gate is hard to determine, particularly when optical-imaging-based RE techniques are used. Second, the actual function is dependent on the control input, which can act as a key input.

In this work, we use the giant spin-Hall effect (GSHE) switch, first proposed in [21], to build polymorphic gates for advanced protection. More specifically, we leverage the GSHE switch recently designed and analyzed by Rangarajan *et al.* [22] in the context of probabilistic computing. We emphasize that the notions of locking and camouflaging are interchangeable in this work due to the polymorphic nature of the proposed primitive, unlike for CMOS-centric approaches. The contributions of this work can be summarized as follows.

1) We leverage a polymorphic, GSHE-based device to propose a versatile security primitive. The primitive provides strong camouflaging capabilities—given two inputs, all 16 possible Boolean functions can be cloaked within a single instance. We elaborate on the device as well as the proposed primitive in detail in Sec. III.
2) We analyze the protection provided by the primitive against attacks such as imaging- and electron-microscopy-based RE, side-channel attacks, and analytical SAT attacks (Sec. V). As for SAT attacks, a comprehensive study is conducted and benchmarked against prior state-of-the-art techniques. Immunity to SAT attacks

[*] S. Patnaik and N. Rangarajan contributed equally.

for probabilistic computing, directly supported by the primitive, is also discussed.

3) We outline the prospects of hybrid CMOS-GSHE designs for industrial benchmarks. We observe that delay-aware protection can provide strong resilience (against SAT attacks) with negligible layout overheads.

II. Background: Prior Art and Limitations

In [23], the authors implemented a low-power and versatile gate using a GSHE-based magnetic tunnel junction (MTJ) as the basic switching element. However, this device is not explicitly tailored for security; it is unable to support logic locking by itself, as it is not truly polymorphic. More concerning is the limitation to only four possible Boolean functions, which renders this primitive weak against SAT attacks (Sec. V).

Alasad *et al.* [24] use all-spin logic (ASL) to design three different security primitives, supporting three sets of camouflaged functionalities: INV/BUF, XOR/XNOR, and AND/NAND/OR/NOR. The layouts of the three primitives are unique; they can be readily distinguished by imaging-based RE tools, which also eases subsequent SAT attacks (Sec. V).

Winograd *et al.* [25] introduced a spin-transfer torque (STT)-based reconfigurable lookup table (LUT), explicitly addressing hardware security. However, their approach falls short in terms of resilience against SAT attack. (Note that the authors did not report on any SAT attack themselves.) We protect the *s38584* benchmark according to their technique and observe that the protected layout can be decamouflaged in less than 30 seconds on average (over 100 runs of camouflaging and SAT attacks). This weak resilience stems from the limited use of their STT-LUT primitive to curb power, performance, and area (PPA) overheads.

As for CMOS-centric camouflaging, most schemes incur a high layout cost. For example, the look-alike NAND-NOR-XOR gate proposed by Rajendran *et al.* [2] induces 4× area, 5.5× power, and 1.6× delay (compared to a regular two-input NAND gate) whereas the threshold-dependent full-chip camouflaging as proposed in [4] still induces overheads of 14%, 82%, and 150% in PPA, respectively. As a result, most schemes are limited to a cost-constrained and selective application, which has severe implications for security (Sec. V).

III. Device-Level Design of Spin-Based Primitive

Protection schemes based on emerging devices can be competitive, even when compared to regular CMOS. While the GSHE switch leveraged in this work is still in the nascent stage of fabrication [26], it is nevertheless promising because of its small scale and low power (Section III-B). As for the relatively large delay, the GSHE-based primitive is still applicable without inducing significant delay overheads (Sec. V).

A. Structure and Operating Principle of the GSHE Switch

The GSHE switch, which is at the heart of the proposed primitive, is shown in Fig. 1. Above the heavy metal spin-Hall layer (purple, bottom) are the write (W; red, bottom) and read (R; red, top) nanomagnets (NM). These nanomagnets (W-NM and R-NM) exhibit a negative mutual dipolar coupling. On top of the R-NM sit two fixed ferromagnetic layers (dark green) with anti-parallel magnetization directions.

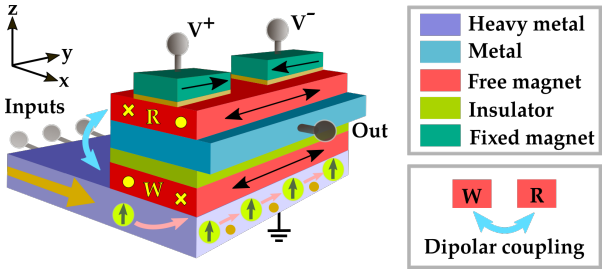

Fig. 1. Structure of the GSHE switch. The concept is derived from [22], but here we adopt a stacked integration to maximize the dipolar coupling.

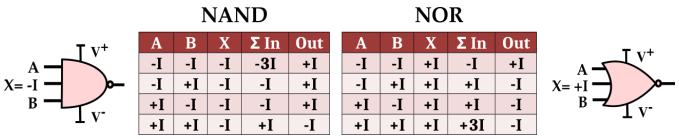

Fig. 2. The current-centric truth tables for NAND and NOR functionalities, with inputs A and B (X is a control signal). As always the case for our GSHE-based primitive, logic 1/0 is represented by an output current +I/-I.

Applying a charge current to the bottom layer (large golden arrow in Fig. 1) results in spin accumulation of one polarity (green spin-up spheres) in the transverse direction (pink arrows) [22]. This spin-polarized current then imparts a spin-transfer torque (STT) to the W-NM [27]. The STT switches the W-NM from one stable state to the other which, in turn, switches the R-NM in the opposite sense.[1] Now, the magnetization direction of the R-NM will be parallel to one of the fixed ferromagnets on top and anti-parallel to the other. The parallel path offers a lower resistance for a charge current passing from/to the respective top contact to/from the output terminal. This read-out phase commences once voltages are applied to the top contacts (V^+ and V^-). Depending on the polarity of the voltage applied to the low-resistance path, the output current either flows inward or outward—this represents the binary result of the GSHE switch operation (see Fig. 2).

B. Characterization and Comparison of the GSHE Switch

The conceptual layout of the GSHE switch (Fig. 3) is drawn based on the design rules for beyond-CMOS devices [17], i.e., in units of maximum misalignment length λ. The area of the GSHE switch is accordingly estimated to be $0.0016\mu\text{m}^2$. The material parameters are given in Table I. Notably, a spin current (I_S) of at least 20μA is required in this work to guarantee a deterministic switching behavior.

The performance of the switch is determined by the nanomagnetic dynamics, which is simulated using the stochastic Landau-Lifshitz-Gilbert-Slonczewski equation [29]. Three simulated delay distributions are illustrated in Fig. 4. For the propagation delay of the primitive, we subsequently assume a mean delay of 1.55 ns obtained for $I_S = 20$ μA.

The power dissipation for the read-out phase is derived according to the equivalent circuit shown in Fig. 3 (inset). Using the following equations and the parameters listed in Table I, the power dissipation of the GSHE switch (including leakage) is derived as 0.2125 μW.

$$P = \frac{{V_{OUT}}^2}{r} + (V_{SUP} - V_{OUT})^2 G_P + (V_{OUT} + V_{SUP})^2 G_{AP}$$

[1]That is because in the presence of negative magnetic dipolar coupling, the minimum energy state is the one in which the W and R nanomagnets are anti-parallel to each other [21].

TABLE I
MATERIAL PARAMETERS OF THE GSHE SWITCH

Parameter	Value
Volume of nanomagnets (NM)	$(28 \times 15 \times 2)$ nm^3 [22]
Saturation magnetization M_s of NM	10^6 A/m (W-NM) [22] 5×10^5 A/m (R-NM) [22]
Uniaxial energy density K_u of NM	2.5×10^4 J/m^3 (W-NM) [22] 5×10^3 J/m^3 (R-NM) [22]
Spin current I_S, determ. switching	20 μA [22]
Resistance area product RAP	1 $\Omega\mu$m^2 [28]
Tunneling magnetoresistance TMR	170% [28]
Parallel conductance G_P	420 μS
Anti-parallel conductance G_{AP}	155.6 μS
Resistivity of heavy metal (HM) ρ	$5.6 \times 10^{-7}\Omega$–m
Spin-Hall angle θ_{SH} of HM	0.4
Thickness t_{HM} of HM	1 nm
Internal gain β of HM $\beta = \theta_{SH} \times (w_{NM}/t_{HM})$	$0.4 \times (15\text{ nm}/1\text{ nm})$ $= 6$
Resistance r of HM	$\approx 1\ k\Omega$

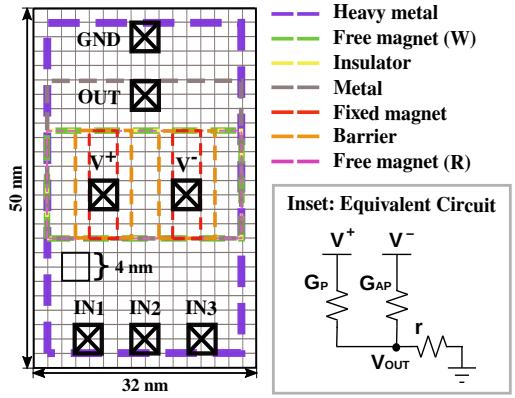

Fig. 3. The conceptual layout of the GSHE switch (main part), and the equivalent circuit (inset, derived from [21]). The power dissipation of the latter is dictated by the resistance r of the heavy metal as well as the conductances of the anti-parallel, high-resistance path (G_{AP}) and the parallel, low-resistance path (G_P) build up by the fixed ferromagnets.

$$V_{SUP} = \left|V^{+/-}\right| = \left(\frac{I_S}{\beta}\right)\left(\frac{1 + r(G_P + G_{AP})}{G_P - G_{AP}}\right);\ V_{OUT} = \frac{I_S\ r}{\beta}$$

$$\frac{G_P}{G_{AP}} = 1 + TMR;\ G_P = \frac{A(nanomagnets)}{RAP}$$

In Table II, we compare the metrics of the GSHE switch against those of existing devices, including ones that are not necessarily security-oriented. The switch is superior in terms of energy/power but is limited in terms of delay. As for security, the number of possible functions is the relevant metric; here, the GSHE switch significantly outperforms prior art. Moreover, a delay-aware application can provide adequate security without any significant overheads (Sec. V).

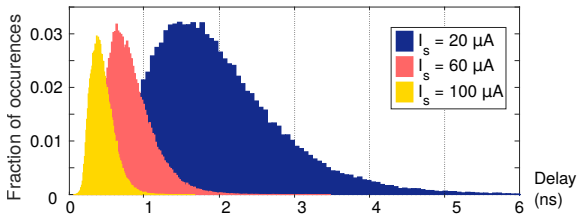

Fig. 4. Delay distributions for the GSHE switch at various spin currents (I_S). The distributions are obtained from 100,000 simulations. Although the delays incurred in switching are stochastic, the switching process itself is still deterministic. Note that the spread and mean delay diminish with increasing I_S, however, at the cost of higher power dissipation.

TABLE II
COMPARISON OF SELECTED EMERGING-DEVICE PRIMITIVES

Publication	# Functions	Energy	Power	Delay
[19] SiNW	NAND/NOR	0.05–0.1 fJ	1.13–1.77 μW	42–56 ps
[24, a] ASL	NAND/NOR/AND/OR	0.58 pJ	351.52 μW	1.65 ns
[24, b] ASL	XOR/XNOR	1.16 pJ	351.52 μW	3.3 ns
[24, c] ASL	INV/BUF	0.13 pJ	342.11 μW	0.38 ns
[30] DWM	AND/OR	67.72 fJ	60.46 μW	1.12 ns
[20] DWM	NAND/NOR/XOR/XNOR/ AND/OR/INV	N/A	N/A	N/A
[23] GSHE	AND/OR/NAND/NOR	N/A	N/A	N/A
[25] STT	NAND/NOR/XOR/ XNOR/AND/OR	N/A	N/A	N/A
This work	**All 16**	**0.33 fJ**	**0.2125 μW**	**1.55 ns**

C. Security Primitive: Cloaking of all 16 Boolean Functions

All 16 possible Boolean functions implemented by the proposed primitive are illustrated in Fig. 5. To realize NAND/NOR, e.g., three charge currents are fed into the bottom layer of the GSHE switch at once: two currents represent the logic signals A and B, and the third current (X) acts as the "tie-breaking" control input (recall Fig. 2). For the XOR/XNOR functionalities, one signal is provided as input current, whereas the other signal and its inverse are provided as input voltages at the V^+ and V^- terminals of the fixed ferromagnets.[2] Swapping the voltage polarities switches between the complementary functions.

Note that three wires are used for the input terminal for all 16 Boolean gates (recall Fig. 3); this renders the layout of the primitive indistinguishable for optical-imaging-based RE, irrespective of the actual functionality. As such, some gates will require dummy wires. Depending on the threat model and concept for chip-level implementation (Sec. IV), one may implement these dummy wires using RE-resilient interconnects in the BEOL [5], or with the help of additional MUXes and key bits to seemingly switch between real/dummy wires at the FEOL. Similar protection is required for the assignment of the different input voltages and control signals.

Finally, in addition to the 16 functions illustrated in Fig. 5, we can readily extend our primitive to cloak latches and flip-flops, by applying the clock signal to the fixed ferromagnets' terminals. Besides, the primitive can readily implement multi-input gates (i.e., >2 signal inputs) as well.

IV. THREAT MODEL AND CONCEPT FOR SECURE CHIP-LEVEL IMPLEMENTATION

We assume the fab and the end-user to be untrusted; the ultimate goal for any adversary is to understand the true functionality of a camouflaged/locked chip. Our threat model represents a notable advancement over prior work related to camouflaging, where the IP holder traditionally *must* trust the fab because of the device/circuit-level protection mechanism.

[2]Toward this end, magneto-electric transducers [31] may be placed in the interconnects. Such transducers can be tailored for uniform, indistinguishable layouts, and can be used to convert (i) charge currents to their reverse (+I to -I, or B to B'), (ii) voltages to charge currents (high/low voltage to +/-I), and (iii) charge currents to voltages (+/-I to high/low voltages).

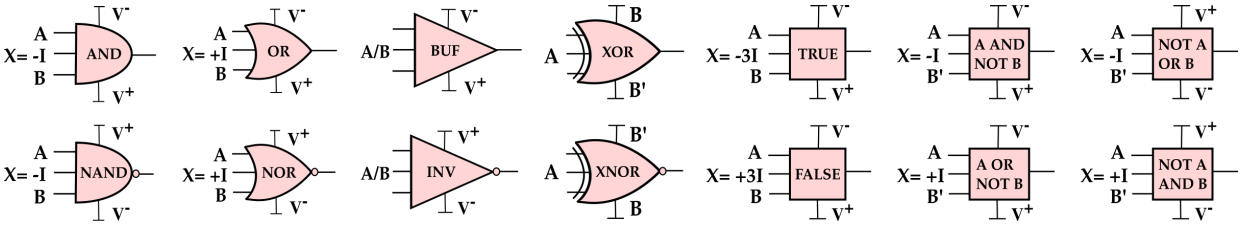

Fig. 5. All 16 possible Boolean functionalities for two inputs, A and B, implemented using the proposed primitive. If required, X serves as control signal, not as regular input. Note that BUF and INV capture two functionalities each.

To hinder fab-based adversaries, we outline two equally promising options for secure implementation: either (a) leverage split manufacturing [32] or (b) provision for a tamper-proof memory. For option (a), the wires for the control inputs and the ferromagnet terminals shall remain protected from the untrusted FEOL fab. Hence, these wires have to be routed at least partially through the BEOL, which must be manufactured by a separate, trusted fab. For option (b), the tamper-proof memory holds a secret key that defines (using some additional circuitry) the correct assignment of control inputs and voltages for all devices. The key must be loaded (by the IP holder) into the memory only after fabrication.

A malicious end-user can obtain the design specifics of the chip through RE and side-channel attacks. She/he can also use a working chip as an oracle for analytical attacks. In the remainder of this paper, we focus on malicious end-user.

V. Security Analysis

A. Study on Large-Scale IP Protection Against SAT Attacks

Setup: We model the proposed primitive and those of selected prior art [2], [3], [19], [20], [23]–[25], [35] as outlined in [9], [36]. Although the proposed primitive also supports locking, here we contrast it only to camouflaging primitives; logic locking and camouflaging are transformable notions without loss of generality [36]. Note that we also contrast to CMOS-centric techniques; this is meaningful as any scheme hinges on the number and composition of their cloaked functionalities [8], [9], not their implementation (i.e., at least for analytical attacks). For a fair evaluation, the same set of gates are protected—gates are randomly selected once for each benchmark, memorized, and then reapplied across all techniques. We evaluate all techniques against powerful SAT attacks [8], [12], [37], run on Intel Xeon server (2.3 GHz, 4 GB per task allowed). The time-out ("t-o") is set to 48 hours.

Benchmarks: We conduct our experiments on traditional benchmarks suites (*ISCAS-85*, *MCNC*, and *ITC-99*), on the large-scale *EPFL suite* [33], and on the industrial *IBM superblue* circuits [34] (Table III). For the *IBM superblue* circuits, we leverage [38] to synthesize and generate the layouts for further analysis. As for SAT attacks, we pre-process the sequential circuits (*IBM superblue*) as follows: the inputs (and outputs) of all flip-flops become primary outputs (and inputs); thereafter, the flip-flops are removed. (Doing so is essential to mimic access to scan chains for the SAT attacks [9].)

TABLE III
Characteristics of Synthesized Benchmarks (Italics: *EPFL Suite* [33]; Bold: *IBM Superblue Suite* [34])

Benchmark	Inputs	Outputs	Gates	Benchmark	Inputs	Outputs	Gates
aes_core	789	668	39,014	*log2*	32	32	51,627
b14	277	299	11,028	**sb1**	8,320	13,025	856,403
b21	522	512	22,715	**sb5**	11,661	9,617	741,483
c7552	207	108	4,045	**sb10**	10,454	23,663	1,117,846
ex1010	10	10	5,066	**sb12**	1,936	4,629	1,523,108
pci_bridge32	3,520	3,528	35,992	**sb18**	3,921	7,465	659,511

On provably secure versus large-scale schemes: Contrary to *provably secure* schemes such as [6], [7], [10], one may find it difficult to engage in "plain" but large-scale camouflaging. The key reason for this concern is that the solution space C—covering all possible functionalities of a camouflaged design and thereby defining the computational efforts for SAT attacks—is hard to quantify precisely [8]–[10]. More specifically, C depends primarily on (i) the number and the composition of functions cloaked by each primitive and (ii) the number and selection of gates protected with a primitive. Recall that prior art is limited in both (i) and (ii) by cost considerations. In contrast, thanks to the innate polymorphism of the proposed primitive, we are unbound toward large-scale and even full-chip camouflaging. Moreover, the primitive cloaks all 16 possible functionalities. Intuitively, our scheme should thus impose maximal efforts for SAT attacks. We believe that this renders our scheme competitive on par with provably secure techniques, and we substantiate this statement with a comprehensive study below.

Results: Table IV contrasts the resilience (against [8], [37]) of all considered schemes for large-scale application. For the same number of gates protected, we observe that the more functions a primitive can cloak, the more resilient it becomes in practice. More importantly, the runtimes required for de-camouflaging (if possible at all), tend to scale exponentially with the percentage of gates being camouflaged.

Our primitive induces by far the highest efforts across all benchmarks. Except for *ex1010*, none of the benchmarks could be resolved within 48 hours once we protect 20% or more of all gates. To confirm this superior resilience, we conducted further attacks running for 240 hours for full-chip protection using the proposed primitive—the designs could still not be resolved. Moreover, we also observe some computational failures;[3] this hints on another practical limitation w.r.t. scalability for SAT attacks, as one can reasonably expect [9].

Besides the attacks of [8], [37], we also leverage *Double DIP* [12]. The key advancement of this attack is that it rules out at least two incorrect keys in each iteration. Conducting the very same set of experiments as in Table IV, we observe that the runtimes are on average higher across all benchmarks. For example, decamouflaging *aes_core* (for 10% protection using our primitive) requires ≈7 hours using [8], but ≈15 hours us-

[3] E.g., "*internal error in 'lglib.c': more than 134,217,724 variables*".

TABLE IV
RUNTIME FOR OUR SAT ATTACKS (USING [8], [37]), IN SECONDS (TIME-OUT T-O IS 172,800 SECONDS, I.E., 48 HOURS)

Benchmark	10% IP Protection							20% IP Protection						
	[2] (3)*	[3], [25] (6)*	[19] (4)*†	[24, c], [35] (2)*	[23], [24, a] (4)*	[20] (7+1)*‡	Our (16)*	[2]	[3], [25]	[19]†	[24, c], [35]	[23], [24, a]	[20]‡	Our
aes_core	610	4,710	890	132	536	6,229	25,890	4,319	41,844	11,306	407	9,432	t-o	t-o
b14	2,078	20,603	11,465	6,884	17,634	27,438	60,306	56,155	t-o	64,145	8,426	t-o	t-o	t-o
b21	7,813	162,324	45,465	3,977	24,035	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o
c7552	37	210	74	12	66	371	2,289	169	14,575	1,153	110	1,327	172,548	t-o
ex1010	62	215	82	12	73	295	922	171	1,047	274	38	250	1,310	4,701
log2	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o
pci_bridge32	1,119	t-o	9,011	1,325	2,690	t-o	t-o	54,577	t-o	t-o	t-o	t-o	t-o	t-o
	30% IP Protection							40–100% IP Protection§						
aes_core	17,148	t-o	31,601	2,020	26,498	t-o	t-o	t-o	t-o	t-o	8,206	t-o	t-o	t-o
b14	56,787	t-o	t-o	38,495	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o
b21	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o
c7552	1,786	t-o	t-o	766	t-o	t-o	t-o	t-o	t-o	t-o	41,721	t-o	t-o	t-o
ex1010	448	4,357	938	87	719	11,736	24,727	1,703	t-o	129,290	169—7,073§	1,950	t-o	t-o
log2	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o
pci_bridge32	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o	t-o

*Number of cloaked functions; refer to Table II or the related publication for the actual sets of cloaked functions. Prior art covering the same set is grouped into one column. †Here we refer to the camouflaging primitive, not the polymorphic gate reported on in Table II. ‡Here we also assume BUF to be available. §The benchmark ex1010 can be resolved for 100% IP protection, when the primitives of [24, c], [35] are used. The related runtime range is for 40–100% protection; all other runtimes are for 40% protection.

ing [12].[4] This finding suggests that large-scale camouflaging can be indeed on par with provably secure schemes.

Independent of our study, note that some prior art (e.g., [25]) proposed cost-limited protection schemes. Here we have demonstrated that an overly limited protection cannot withstand powerful SAT attacks (also recall Sec. II for [25]).

Next, we outline the prospects for camouflaging of industrial circuits. Recall that the delay of the GSHE switch is considerably higher when compared to CMOS (Sec. III). Interestingly, large-scale circuits typically exhibit biased distributions of delay paths, with most paths having short delays but few paths having dominant, critical delays (Fig. 6). In an experimental study on those *IBM superblue* circuits, we replace CMOS gates in the non-critical paths with the GSHE-based primitive such that no delay overheads can be expected.[5] On an average, we can camouflage 5–15% of all gates this way. Conducting SAT attacks [8], [37] on those protected designs, we observe that they cannot be resolved within 240 hours; in fact, most runs incur similar failures as discussed above. This indicates that the proposed primitive can help to strongly protect industrial circuits without excessive layout (PPA) overheads.

B. On Stochastic Switching to Hinder SAT Attacks

So far we leveraged the primitive in the context of classical, deterministic computation. Note, however, that the underlying GSHE switch supports tunable probabilistic computation [22]. Interestingly, the implications of probabilistic computation on hardware security are largely unexplored.

Recall the general principle of SAT attacks, i.e., to carefully apply input patterns on a working chip and to observe the output patterns, throughout multiple sampling iterations, until the correct assignment for all key bits can be derived (by ruling out incorrect keys via disagreement). Now consider a scenario where the GSHE switch (or any probabilistic device, for that matter) is tuned for 95% accuracy. This implies that 5% of the patterns observed by the SAT attack are incorrect.

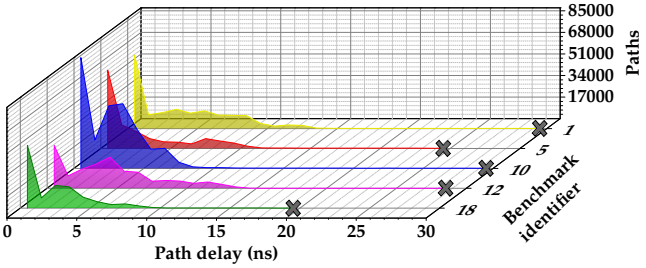

Fig. 6. Delay distributions of selected *IBM superblue* circuits. The paths with the longest, critical delays are marked by crosses for clarity.

We believe that most if not all proposed SAT attacks will fail in such scenarios.[6] That is because they have not been tailored to account for incorrect output patterns. Even if they were, distinguishing incorrect patterns from correct ones is difficult when only given a "probabilistic black-box oracle." Naturally, one might want to leverage machine learning (ML) toward this end. We argue, however, it remains to be seen whether ML-based attacks will be sufficiently robust and capable. Here we like to point out that (i) the GSHE switch experiences thermally induced stochasticity [22], (ii) the error rate for any switch can be tuned individually, and (iii) those individual distributions superpose with each other while they propagate throughout the entire design, resulting in stochastically correlated behavior at the primary outputs.

C. Preventing Reverse Engineering and Side-Channel Attacks

Layout identification and read-out attacks: Recall that the layout of the proposed primitive is uniform (Sec. III), hence indistinguishable for optical-imaging-based RE. A more sophisticated attacker might, however, leverage electron microscopy (EM) for identification and read-out attacks [16].

[4]Due to lack of space, we refrain from providing all detailed results on [12].

[5]We anticipate such hybrid designs to be practical, given the CMOS-compatible manufacturing of spin-based devices [39]. The main focus of this work, however, is hardware security, not circuit design. Hence, to mimic hybrid designs, we replace the delay numbers of selected CMOS gates in non-critical timing paths with that of the GSHE switch, i.e., 1.55 ns.

[6]The most promising contender here is arguably *AppSAT* [11], which is based on the probably-approximately-correct (PAC) paradigm. The attack as outlined in [11], however, requires a consistent solution space regarding the input-output queries—probabilistic computation violates this assumption. The attack was not available to us for an experimental study at this time.

While such attacks are yet to be demonstrated on switching devices at runtime, we believe that the proposed primitive can prevent them. First, the dimensions of the GSHE switch are significantly smaller than CMOS devices, which is a challenge regarding the spatial resolution for EM-based analysis [16]. Second, the primitive is truly polymorphic, i.e., its functionality can be switched at runtime; see also next.

Polymorphism at the chip-level: Given truly polymorphic gates and some circuitry to judiciously switch the functionalities of gates, we can implement *runtime polymorphism* at the chip-level. Then, internal functionalities are not static (possibly even for static input patterns), whereupon an RE-centric attacker is bound to misinterpret parts of the layout—it is virtually impossible to resolve all dynamic features on full-chip scale at once.[7] Independent of RE threats, runtime polymorphism at the chip-level can also enable dynamic protection, e.g., as recently proposed by Koteshwara *et al.* [40]. Their idea is to alter the key dynamically, thereby rendering runtime-intensive attacks incapable (SAT attacks in particular).

Photonic side-channel attacks: While CMOS devices emit photons during operation, making them vulnerable to powerful attacks such as [41], the GSHE switch itself does not emit any photons. The fundamentally different switching principle hence makes the proposed primitive inherently resilient to read-out attacks based on photons. Still, we caution that an assessment against such attacks shall be performed in future.

Magnetic and temperature attacks: Ghosh *et al.* [18] outlined attacks on spintronic (memory) devices using magnetic fields and temperature curves. The design of the GSHE switch shall ensure a robust coupling between the W and R nanomagnets [22]. This would naturally be disturbed by any external magnetic fields. Hence, an attacker leveraging a magnetic probe may induce stuck-at-faults which are, however, hardly controllable due to multiple factors (very small size of switches, accordingly large magnetic fields required for the probe, state of W and R magnetizations, the orientation of the fixed magnets, voltage polarities on the fixed magnets). This implies that sensitization attacks such as [2] will be difficult, if practical at all. Regarding temperature-driven attacks, note that the retention time of the switch will be impacted. The resulting disturbances, however, are likely stochastic due to the inherent thermal noise in the nanomagnets.

VI. Conclusion

We explore the security aspects of the GSHE switch: a versatile spin-based polymorphic device which can support both camouflaging and logic locking. Through a comprehensive study using SAT attacks, we show the strong resilience of our deterministic primitive as compared to prior art. We further discuss the resilience of our primitive against various classes of side-channel attacks. Finally, we lay the foundations for promising security concepts: truly polymorphic behavior at runtime, and stochastic behavior to thwart analytical attacks.

Acknowledgements

This work was carried out in part on the High Performance Computing resources at New York University Abu Dhabi.

[7] In [16], e.g., it took 50 ns to read-out one pixel of one memory cell, which is well above the 1.55 ns speed of the GSHE device.

References

[1] M. Rostami, F. Koushanfar, and R. Karri, "A primer on hardware security: Models, methods, and metrics," *Proc. IEEE*, vol. 102, no. 8, pp. 1283–1295, 2014.
[2] J. Rajendran, M. Sam, O. Sinanoglu, and R. Karri, "Security analysis of integrated circuit camouflaging," in *Proc. CCS*, 2013, pp. 709–720.
[3] I. R. Nirmala, D. Vontela, S. Ghosh, and A. Iyengar, "A novel threshold voltage defined switch for circuit camouflaging," in *Proc. ETS*, 2016, pp. 1–2.
[4] B. Erbagci, C. Erbagci, N. E. C. Akkaya, and K. Mai, "A secure camouflaged threshold voltage defined logic family," in *Proc. HOST*, 2016, pp. 229–235.
[5] S. Patnaik, M. Ashraf, J. Knechtel, and O. Sinanoglu, "Obfuscating the interconnects: Low-cost and resilient full-chip layout camouflaging," in *Proc. ICCAD*, 2017.
[6] M. Yasin, B. Mazumdar, J. J. V. Rajendran, and O. Sinanoglu, "SARLock: SAT attack resistant logic locking," in *Proc. HOST*, 2016, pp. 236–241.
[7] Y. Xie and A. Srivastava, "Mitigating SAT attack on logic locking," in *Proc. CHES*, 2016, pp. 127–146.
[8] P. Subramanyan, S. Ray, and S. Malik, "Evaluating the security of logic encryption algorithms," in *Proc. HOST*, 2015, pp. 137–143, see also [37].
[9] M. E. Massad, S. Garg, and M. V. Tripunitara, "Integrated circuit (IC) decamouflaging: Reverse engineering camouflaged ICs within minutes," in *Proc. NDSS*, 2015, pp. 1–14.
[10] M. Li *et al.*, "Provably secure camouflaging strategy for IC protection," in *Proc. ICCAD*, 2016, pp. 28:1–28:8.
[11] K. Shamsi, M. Li, T. Meade, Z. Zhao, D. Z. Pan, and Y. Jin, "AppSAT: Approximately deobfuscating integrated circuits," in *Proc. HOST*, 2017, pp. 95–100.
[12] Y. Shen and H. Zhou, "Double DIP: Re-evaluating security of logic encryption algorithms," in *Proc. GLSVLSI*, 2017, pp. 179–184.
[13] X. Xu, B. Shakya, M. M. Tehranipoor, and D. Forte, "Novel bypass attack and BDD-based tradeoff analysis against all known logic locking attacks," in *Proc. CHES*, 2017.
[14] H. Wang, Q. Shi, D. Forte, and M. M. Tehranipoor, "Probing attacks on integrated circuits: Challenges and research opportunities," *IEEE Design & Test*, vol. 34, no. 5, pp. 63–71, 2017.
[15] S. Skorobogatov and C. Woods, "In the blink of an eye: There goes your AES key," in *IACR Crypt. ePrint Arch.*, no. 296, 2012.
[16] F. Courbon, S. Skorobogatov, and C. Woods, "Direct charge measurement in floating gate transistors of flash EEPROM using scanning electron microscopy," in *Proc. ISTFA*, 2016, pp. 1–6.
[17] D. E. Nikonov and I. A. Young, "Overview of beyond-CMOS devices and a uniform methodology for their benchmarking," *Proc. IEEE*, vol. 101, no. 12, pp. 2498–2533, 2013.
[18] S. Ghosh, "Spintronics and security: Prospects, vulnerabilities, attack models, and preventions," *Proc. IEEE*, vol. 104, no. 10, pp. 1864–1893, 2016.
[19] Y. Bi *et al.*, "Emerging technology-based design of primitives for hardware security," *J. Emerg. Tech. Comp. Sys.*, vol. 13, no. 1, pp. 3:1–3:19, 2016.
[20] F. Parveen, Z. He, S. Angizi, and D. Fan, "Hybrid polymorphic logic gate with 5-terminal magnetic domain wall motion device," in *Proc. ISVLSI*, 2017, pp. 152–157.
[21] S. Datta, S. Salahuddin, and B. Behin-Aein, "Non-volatile spin switch for boolean and non-boolean logic," *Applied Physics Letters*, vol. 101, no. 25, p. 252411, 2012.
[22] N. Rangarajan, A. Parthasarathy, N. Kani, and S. Rakheja, "Energy-efficient computing with probabilistic magnetic bits–performance modeling and comparison against probabilistic CMOS logic," *IEEE Trans. on Magnetics*, vol. 53, no. 11, pp. 1–10, 2017.
[23] Y. Zhang, B. Yan, W. Wu, H. Li, and Y. Chen, "Giant spin hall effect (GSHE) logic design for low power application," in *Proc. DATE*, 2015, pp. 1000–1005.
[24] Q. Alasad, J. Yuan, and D. Fan, "Leveraging all-spin logic to improve hardware security," in *Proc. GLSVLSI*, 2017, pp. 491–494.
[25] T. Winograd, H. Salmani, H. Mahmoodi, K. Gaj, and H. Homayoun, "Hybrid STT-CMOS designs for reverse-engineering prevention," in *Proc. DAC*, 2016, pp. 88–93.
[26] A. V. Penumatcha, C.-C. Lin, V. Q. Diep, S. Datta, J. Appenzeller, and Z. Chen, "Impact of scaling on the dipolar coupling in magnet–insulator–magnet structures," *IEEE Trans. on Magnetics*, vol. 52, no. 1, pp. 1–7, 2016.
[27] J. C. Slonczewski, "Current-driven excitation of magnetic multilayers," *Journal of Magnetism and Magnetic Materials*, vol. 159, no. 1-2, pp. L1–L7, 1996.
[28] H. Maehara *et al.*, "Tunnel magnetoresistance above 170% and resistance–area product of 1 $\Omega(\mu m)^2$ attained by in situ annealing of ultra-thin MgO tunnel barrier," *Applied Physics Express*, vol. 4, no. 3, p. 033002, 2011.
[29] M. d'Aquino, C. Serpico, G. Coppola, I. Mayergoyz, and G. Bertotti, "Midpoint numerical technique for stochastic landau-lifshitz-gilbert dynamics," *Journal of Applied Physics*, vol. 99, no. 8, p. 08B905, 2006.
[30] K. Huang and R. Zhao, "Magnetic domain-wall racetrack memory-based nonvolatile logic for low-power computing and fast run-time-reconfiguration," *Trans. VLSI Syst.*, vol. 24, no. 9, pp. 2861–2872, 2016.
[31] S. Manipatruni, D. E. Nikonov, R. Ramesh, H. Li, and I. A. Young. (2015) Spin-orbit logic with magnetoelectric nodes: A scalable charge mediated nonvolatile spintronic logic. [Online]. Available: https://arxiv.org/abs/1512.05428
[32] C. McCants, "Trusted integrated chips (TIC)," Intelligence Advanced Research Projects Activity (IARPA), Tech. Rep., 2011.
[33] L. Amarù. (2015) Majority-inverter graph (MIG) benchmark suite. [Online]. Available: http://lsi.epfl.ch/MIG
[34] N. Viswanathan, C. J. Alpert, C. Sze, Z. Li, G.-J. Nam, and J. A. Roy, "The ISPD-2011 routability-driven placement contest and benchmark suite," in *Proc. Int. Symp. Phys. Des.*, 2011, pp. 141–146.
[35] J. Zhang, "A practical logic obfuscation technique for hardware security," *Trans. VLSI Syst.*, vol. 24, no. 3, pp. 1193–1197, 2016.
[36] M. Yasin and O. Sinanoglu, "Transforming between logic locking and IC camouflaging," in *Proc. Des. Test Symp.*, 2015, pp. 1–4.
[37] P. Subramanyan. (2017) Evaluating the security of logic encryption algorithms. [Online]. Available: https://bitbucket.org/spramod/host15-logic-encryption
[38] A. B. Kahng, H. Lee, and J. Li, "Horizontal benchmark extension for improved assessment of physical CAD research," in *Proc. GLSVLSI*, 2014, pp. 27–32. [Online]. Available: http://vlsicad.ucsd.edu/A2A/
[39] S. Matsunaga *et al.*, "Fabrication of a nonvolatile full adder based on logic-in-memory architecture using magnetic tunnel junctions," *Applied Physics Express*, vol. 1, no. 9, p. 091301, 2008.
[40] S. Koteshwara, C. H. Kim, and K. K. Parhi, "Key-based dynamic functional obfuscation of integrated circuits using sequentially-triggered mode-based design," *Trans. Inf. Forens. Sec.*, vol. 13, no. 1, pp. 79–93, 2018.
[41] A. Schlösser, D. Nedospasov, J. Krämer, S. Orlic, and J.-P. Seifert, "Simple photonic emission analysis of AES," in *Proc. CHES*, 2012, pp. 41–57.