Logic Locking in Single Flux Quantum Circuits

Tahereh Jabbari, *Student Member, IEEE*, Gleb Krylov, *Student Member, IEEE*, and Eby G. Friedman, *Fellow, IEEE*

***Abstract*—The hardware security of RSFQ circuits has become an issue of growing importance for prospective exascale computing systems. Hardware security in RSFQ circuits is particularly relevant to large scale data centers operating with sensitive information. The number of fabrication facilities for superconductive niobium-based technology is limited, and the supply chain for distributing fabricated circuits can be compromised. Logic locking is widely used in modern CMOS circuits to enhance security by masking the functionality of the circuit using a secret key. If an attacker possesses a physical circuit secured by logic locking, the attacker would be unable to determine the intended function. In this paper, a novel methodology for logic locking is proposed for SFQ circuits. Mutual inductances are used to apply additional currents to some or all of the logic gates. These currents behave as keys to access the functionality of the SFQ logic elements. In the proposed technique, only after an additional correct current is applied to all of the locked gates will the circuit produce the proper output. By using inductors with a positive and negative mutual inductance connected to the internal inductances of the gates, the range of current required to unlock a secured circuit is greatly narrowed. In this work, the operation of this proposed logic locking technique is demonstrated using modified SFQ OR gates to enable security while maintaining proper functionality. Less than 4% area overhead is achieved.**

***Index Terms*—Hardware security, logic locking, reverse engineering, single flux quantum, superconductive integrated circuits, superconductive digital electronics.**

I. Introduction

SUPERCONDUCTIVE rapid single flux quantum (RSFQ) technology is one of the most promising beyond-CMOS technologies for ultra-low power and ultra-high speed digital applications [1]. Significant development in the design and fabrication of superconductive electronics has resulted in device densities exceeding 600000 Josephson junctions/cm^2. Josephson junctions (JJs) in RSFQ circuits propagate SFQ pulses through logic gates operating at switching speeds on the order of a picosecond, while dissipating power below 10^{-19} J [2]–[7]. An RSFQ-based arithmetic logic unit has been demonstrated to operate at frequencies approaching 80 GHz with an 8 bit RSFQ datapath [8], [9].

Prospective exascale computing systems based on VLSI complexity SFQ circuits are expected to be used for sensitive tasks. Hardware security for superconductive technology and novel techniques for providing trustworthy hardware based on RSFQ circuits are therefore necessary. Hardware security methodologies for this technology are currently not well established. Recent progress in the fabrication and design of RSFQ circuits emphasizes the need for hardware security techniques targeting SFQ circuits. Furthermore, SFQ technology exhibits unique advantages and challenges, which should be considered when applying hardware security techniques.

One widely used hardware security technique in modern CMOS circuits is logic locking [10], [11]. Logic locking introduces modifications into a circuit to prevent piracy, counterfeiting, reverse engineering, and overproduction. Logic locking hides and locks the functionality of a circuit. A valid key is required for correct functionality. Applying an incorrect key on a locked circuit produces incorrect or seemingly random behavior. Even if an attacker obtains a physical copy of a circuit, reverse engineering the circuit layout does not allow the attacker to determine the intended behavior.

In this paper, a logic locking technique for RSFQ circuits is proposed and demonstrated on a modified RSFQ OR gate. The principles of hardware security, logic locking, and possible attack paths on circuits secured by this technique are briefly reviewed in Section II. Countermeasures against these types of attacks on SFQ circuits are discussed in Section III. A logic locking technique for SFQ circuits is described in Section IV. The hardware cost of the proposed technique is also discussed in this section. The paper is concluded in Section V.

II. Background

The principles of hardware security and logic locking are, respectively, introduced in section II-A and II-B. Related attack paths on logic locking in CMOS technology are also discussed in section II-C.

A. Principles of Hardware Security

Due to the increasing complexity of modern systems-on-chip with advanced fabrication capabilities and higher manufacturing costs, many companies have become fabless [12]. These fabless companies design the integrated circuits, utilizing an external

Manuscript received November 27, 2020; revised January 25, 2021; accepted March 5, 2021. Date of publication March 11, 2021; date of current version April 2, 2021. This work was supported by the Department of Defense (DoD) Agency-Intelligence Advanced Research Projects Activity (IARPA) through the U.S.Army Research Office under Contract W911NF-17-9-0001. *(Corresponding author: Tahereh Jabbari.)*

The authors are with the Department of Electrical and Computer Engineering, University of Rochester, Rochester, NY 14627 USA (e-mail: tjabbari@ur.rochester.edu; gkrylov@ur.rochester.edu; friedman@ece.rochester.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TASC.2021.3065301.

Digital Object Identifier 10.1109/TASC.2021.3065301

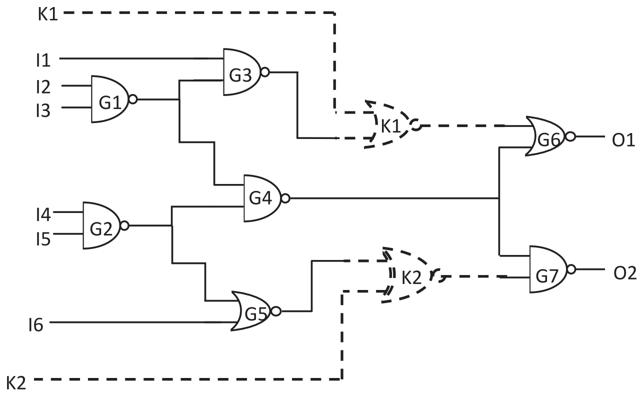

Fig. 1. Circuit utilizing logic locking with two key gates, K_1 and K_2.

foundry for manufacturing. Although distributing the manufacturing process to an external foundry reduces costs, this process also introduces security vulnerabilities into the design flow [13].

With increasingly distributed IC design flows, hardware is vulnerable to a number of attack paths, such as counterfeiting, reverse engineering (RE), and intellectual property (IP) piracy. An attacker may insert a hardware trojan inside a fabricated circuit – an external component to perform additional malicious operations. Other vulnerabilities that can be introduced during the distributed manufacturing process are IC counterfeiting [14], theft of IC masks [15], and overproduction of ICs [16].

Industries annually lose up to $4 billion due to IP violations in semiconductor technology [10]. Hardware security has been established as an important field to mitigate the risks of piracy, counterfeiting, reverse engineering, and side channel attacks [17]. If the functionality of an IC can be hidden while the IC passes through the different, potentially untrustworthy phases of the design flow, these attacks can be thwarted [16], [18]. It is therefore important to the IC design company to protect this design flow. Counterfeiting is typically thwarted by IC camouflaging [19] or logic locking to prevent RE, or by including a watermark to identify counterfeit ICs. Logic locking also provides protection against piracy and overproduction attacks.

B. Logic Locking

Logic locking hides the correct functionality of a circuit by introducing additional gates within the original design. In this technique, a set of key gates, key inputs, and an on-chip memory are introduced into the design to prevent attacks from the supply chain and untrusted foundries. The key gates use AND/OR gates, XOR/XNOR gates, MUX gates, and look up tables (LUT) [20]. An example of a locked circuit with key gates is shown in Fig. 1. The key inputs are K_1 and K_2 which connect to the key gates. The correct output is produced only if the correct value of the keys are applied [16]. An incorrect key used with a logic locked design causes incorrect or random operation.

Since the correct key is known only to the designer, the foundry cannot utilize any copies or overproduce and sell additional ICs without these secret keys. If the number of key values is sufficiently large, manual brute force insertion of different keys is infeasible. Furthermore, this technique prevents an external attacker from analyzing the structural behavior of the design even if a copy of the secured circuit is obtained.

C. Threat Model of Attacks on Logic Locking

The primary objective of an attack on a circuit secured with logic locking is to determine the correct value of the secret keys to decipher the functional netlist of the ICs. If the keys are determined and the design is deciphered, the correct place to insert an undetected hardware trojan can also be determined.

Different input patterns can be applied to both the circuit and the key inputs in a brute force manner. The output of these patterns can be used to discover the correct keys. In this attack, both the locked netlist and the design of the circuit are required. The netlist can be obtained from reverse engineering a GDSII layout file, masks, or an activated functional IC. With complex circuits and a large number of key inputs, these attacks become less efficient.

Reverse engineering poses another major challenge to hardware security. Reverse engineering is the process of analyzing the layout and functionality of a system to extract the gate-level netlist. A typical reverse engineering attack requires several steps to extract the netlist. The initial step of a RE attack is a product teardown to identify the external characteristics of the product and package ($e.g.$, the pin arrangement). The next step – system level RE – analyzes the operations, functions, and timing characteristics of the interconnect paths. In the following step – process analysis – the structure and materials used for fabrication are examined. In the final step – circuit extraction – the gate-level schematic and netlist of the design are extracted. The cost and time necessary for RE attacks significantly increase with each step [21]. RE can be used to obtain confidential information about the design to recreate the gate level netlist, allowing counterfeit ICs to be built among other nefarious schemes.

Due to the relatively simple layout structure and limited number of devices in VLSI complexity SFQ systems, as compared to CMOS systems, reverse engineering is a serious hardware threat to SFQ circuits. The importance of hardware security in RSFQ circuits is emphasized by one of the primary prospective applications of these circuits – large scale data centers typically operating with sensitive information. In the following sections, countermeasures to these attacks applicable to RSFQ circuits are discussed.

III. Attack Countermeasures in SFQ Circuits

IC camouflaging and logic locking aim to thwart the threat of reverse engineering attacks on hardware. IC camouflaging and logic locking are, respectively, a layout technique and a circuit technique. The choice between IC camouflaging and logic locking depends upon the access of the expected attackers to the necessary resources. Both techniques, however, can be simultaneously used in an SFQ circuit.

Camouflaging techniques were first proposed for RSFQ circuits in [19], utilizing dummy Josephson junctions and camouflaged SFQ gates. These techniques are briefly described in section III-A. Logic locking for RSFQ circuits – a novel approach to thwart RE attacks, is proposed in section III-B. A modified SFQ OR gate is described in section III-C to demonstrate logic locking, as described in section III-B.

A. IC Camouflaging in SFQ Circuits

IC camouflaging in SFQ circuits is a layout technique that obstructs the reverse engineering process by introducing dummy JJs into a layout. A dummy JJ is designed to generate the identical top view image of a layout as a standard JJ [19], while behaving as a resistor. Distinguishing between a real JJ and a dummy JJ is difficult with RE attacks, which typically utilize delayering and analysis of the top view image of the layout [22], [23]. RE can only detect these dummy JJs by slicing an IC and analyzing a side view image of the layout. Slicing the die to detect dummy JJs is highly challenging in SFQ circuits due to the expected large number of JJs in large scale SFQ circuits, and the small difference in the thickness of the tunneling barrier between a real and dummy JJ.

In camouflaged SFQ cells, both real and dummy JJs are used. When JJs are necessary for gate operation, the layout is changed and dummy JJs are replaced with real JJs. This technique relies on making these JJs indistinguishable to the attacker, who extracts an incorrect netlist.

IC camouflaging in SFQ thwarts RE attacks by introducing camouflaged cells into a standard cell library along with the regular cells. The rest of the layout and synthesis process remains unchanged. A large camouflaged SFQ circuit consists of camouflaged and regular gates with indistinguishable layouts. Camouflaged RSFQ flip flops and AND/OR gates are proposed in [19]. Camouflaged AND/OR SFQ gates operate as either a two-input AND gate or a two-input OR gate. A camouflaged RSFQ DFF is designed to behave as a Josephson transmission line (JTL) despite a layout identical to a standard RSFQ D flip flop (DFF).

IC camouflaging increases the effort necessary for hardware RE attacks. Camouflaged gates, however, significantly increase the area, power, and delay of a circuit as compared to using only standard gates. A tradeoff therefore exists between security and cost as in most hardware security approaches.

B. Logic Locking in SFQ Circuits

Logic locking complicates attacks, thereby improving the security of SFQ circuits. Existing CMOS logic locking techniques rely on introducing additional gates, look up tables, and external inputs into the design [20]. Logic locking can be similarly applied in RSFQ circuits without additional modifications. The necessary gates, however – typically XOR/XNOR and multiplexers – are expensive in terms of RSFQ circuit area. LUTs also require significant area. The pinout limitations of modern superconductive ICs also severely limit the size of the secret key, compromising security.

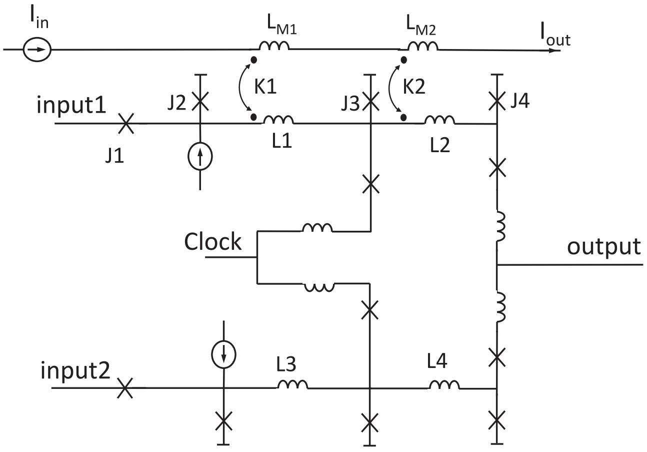

Fig. 2. Proposed OR gate with mutual inductances to apply the secret key current. $I_{in} = 250\mu$ A, L1 = L3 = 15.1 pH, L2 = L4 = 3.8 pH, and $L_{M1} = L_{M2} = 1$ pH (in a 10 kA/cm^2 technology).

A different methodology for logic locking in SFQ circuits is proposed here. Rather than applying a data key, a specific current magnitude is used as the secret key. This current is applied to specific inductances within specific gates. These locked gates exhibit incorrect operation when no current or a different current is supplied. The internal parameters of the gates are modified, and different mutual inductors are introduced, coupling the key current to the gates. By increasing the number of locked gates relying on this key current and varying the coupling direction, the range of key currents required for correct operation is narrowed, enhancing the security. In the following section, this proposed logic locking technique is evaluated in terms of the security of SFQ circuits.

C. Modified OR Gate for Logic Locking

A modified OR gate is shown in Fig. 2. Mutual inductances are used to apply the additional secret key current to unlock the correct functionality of the OR gate. In this circuit, the mutual inductance between L1 and L_{M1} and between L2 and L_{M2} are used to apply the key currents.

The coupling coefficient K_n of the inductances changes the current through the internal gate inductances [24]. Due to the small current in the key lines, the effects of the key current on other circuit components are negligible as compared to the bias lines. To prevent any additional inductive coupling, the key lines can be placed farther from any sensitive circuit components. L1 and L2 are arbitrarily chosen as coupled inductors in the OR gate. Other gate inductors can also be used. The magnitude of the current within these inductors should be carefully chosen to maintain correct operation of the OR gate. L1 controls the current within one of the state storage loops within the OR gate. L2 also affects the current within the state storage loop, as well as switching junction J3. A range of the coupling coefficient between L1 and L_{M1} and between L2 and L_{M2} are, respectively, $-0.45 < K_1 < 0.45$ with zero coupling between L2 and L_{M2}, and $-0.6 < K_2 < 0.6$ with zero coupling between L1 and L_{M1}.

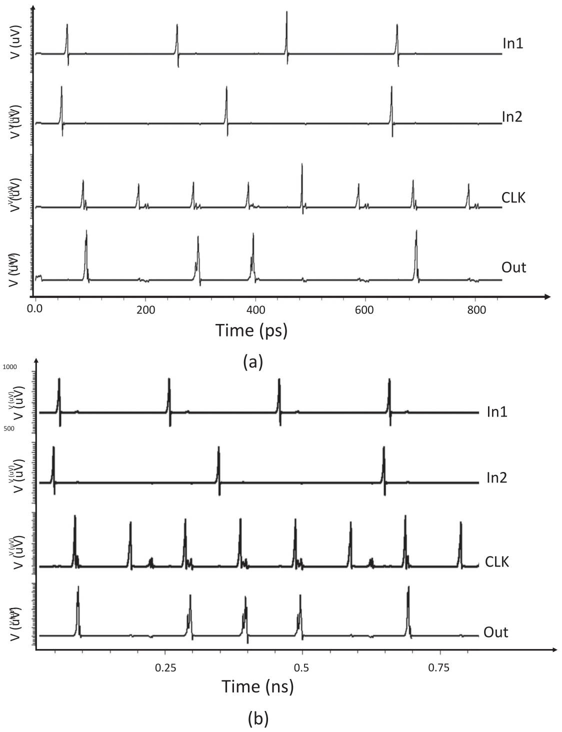

Fig. 3. OR gate operation, (a) with incorrect current key currents with K1 = 0.5 and K2 = 0.5, and (b) with correct current key currents with K1 = 0.3 and K2 = 0.3.

To unlock this OR gate, an attacker needs to determine the correct value of the key current. Correct outputs are only produced when correct currents are supplied. Incorrect key currents coupled to the inductances L_1 and L_2 produce incorrect or random circuit behavior. These incorrect key currents change the bias conditions of the SFQ storage loop by changing the current in L_1 and L_2. Incorrect operation of an SFQ OR gate is shown in Fig. 3(a). The circuit incorrectly produces an output after the second and third pulse of input 1 (see Fig. 3(a)). The locked circuits only produce correct outputs when the appropriate magnitude of the key currents is applied to the mutual inductors with a coupling coefficient K_n. Correct operation of the circuit is shown in Fig. 3(b).

IV. Security Characteristics of Logic Locking

An analysis of the security characteristics of the proposed technique is presented in section IV-A. The area of the proposed logic locking technique using the modified OR gate is quantified in section IV-B.

A. Analysis of Security Characteristics

To increase the security of the proposed technique, the range of coupling coefficients for an OR gate is evaluated. By changing the coupling coefficient K_n, different fractions of the key current can be applied to the gates through the inductances. The range of K_2 within a modified OR gate for different values of K_1 is shown in Fig. 4. Each range of K2 has a specific range for the additional key currents. The range of additional current is listed in Table I. The key current margins are described as margins of K2. To unlock the circuit, it is necessary to determine the correct value of K_1, range of K_2, and range of key current. With $K_1 =$ 0.3, the range of coupling coefficient K_2 is $0.25 < K_1 < 0.35$. For smaller K_1, the circuit exhibits a large range of K_2, resulting in lower security as compared to a higher K_1. A narrower range of K_n increases the effort by the attacker to determine the secret key current.

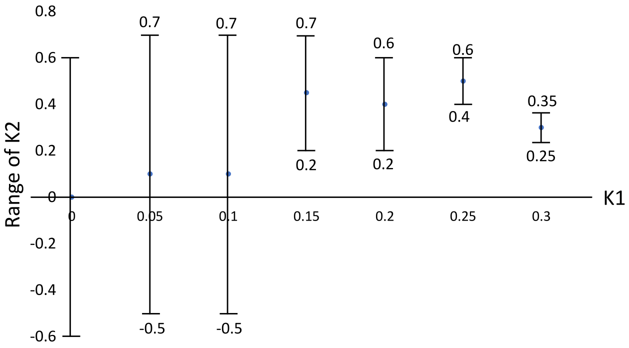

Fig. 4. Security characteristics of an OR gate for different ranges of the coupling coefficient.

TABLE I
Range of Key Currents for Different Ranges of the Coupling Coefficients

		Range of current through L1	**Range of current through L2**
K1 = 0	K2 = 0	9 μA	-33 μA
K1 = 0	-0.6 <K2 <0.6	5 μA to 13 μA	-73 μA to 5 μA
K1 = 0.1	-0.5 <K2 <0.7	12 μA to 19 μA	-66 μA to 12 μA
K1 = 0.2	0.2 <K2 <0.6	21 μA to 24 μA	-17 μA to 7 μA
K1 = 0.3	0.25 <K2 <0.35	27 μA to 28 μA	-13 μA to -7 μA

Manufacturing process variations is a challenging issue in all large scale integrated circuits. A significant tradeoff exists between circuit yield and security. To maintain proper functioning of a circuit secured by logic locking, the range of effective key currents should be wider than any expected bias variations caused by manufacturing and the bias distribution network [25], [26]. Process variations can improve the overall security of the logic locked system, protecting the circuit. Unlike the intended user of an IC, correct operation of an IC is not known to the attacker, which inhibits a brute force key selection attack.

Multiple locked gates can be connected to the same source of key current. These gates utilize different magnitudes and directions of inductive coupling with only a small overlap in the operational range of the key current, providing greater security. In this way, the magnitude and precision of the key currents can be increased in case of greater manufacturing variations.

B. Hardware Cost

An important tradeoff exists in the proposed logic locking technique between hardware security and physical area. The area overhead of the logic locked OR gate described here is approximately 20%. ISCAS'85 benchmark circuits are used to characterize the area overhead of the proposed technique when

TABLE II
BENCHMARK CHARACTERISTICS (*ISCAS'85 BENCHMARK CIRCUITS* [27])

Benchmark	# Gates	# OR gates	Area overhead with 10% locked OR gates	Area overhead with 20% locked OR gates
c880	383	90	0.5%	1%
c2670	1,193	89	0.15%	0.3%
c3540	1,669	160	0.2%	0.4%
c5315	2,406	241	0.2%	0.4%
c6288	2,406	2,128	1.77%	3.6%
c7552	3,512	298	0.17%	0.3%

applied to large scale circuits. In the benchmark circuits listed in Table II, the OR and NOR gates are replaced with locked OR gates to produce a narrow range of the correct key current. The number of OR gates within each benchmark circuit is listed in Table II. Only a few locked OR gates are necessary to affect the output. The area overhead for these benchmark circuits is also listed in Table II, assuming 10% and 20% of the OR gates are replaced by locked OR gates. In the c6288 benchmark circuit with a large number of OR gates, 20% of the OR gates are replaced with locked OR gates. The area overhead is approximately 3.6%. The additional area to logic lock the c6288 benchmark circuit is therefore fairly small. The area overhead is greater if additional locked gates are used to further increase the security of the circuit.

V. CONCLUSION

A primary prospective application of RFSQ circuits – large scale data centers operating with sensitive information – emphasizes the importance of hardware security in RSFQ circuits. A hardware security approach for SFQ circuits – logic locking– is proposed herein. Logic locking is a well known technique widely used to secure CMOS circuits. Although this technique can be applied to SFQ circuits without modifications, standard approaches require costly gates and additional input pins. A novel way to provide a secret key for logic locking is proposed. Standard RSFQ gates are modified to depend on a secret key current to maintain correct functionality. Mutual inductors are used to couple a specific additional positive or negative current into the locked gate from the key current. The efficacy of the proposed logic locking technique is characterized by the number of logic locked gates. The additional area of the locked OR gates is evaluated with ISCAS'85 benchmark circuits. The area overhead for circuits with a large number of logic locked OR gates is below 4%.

ACKNOWLEDGMENT

The content of the information does not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred.

REFERENCES

[1] K. K. Likharev and V. K. Semenov, "RSFQ logic/memory family: A new Josephson-junction technology for sub-terahertz-clock-frequency digital systems," *IEEE Trans. Appl. Supercond.*, vol. 1, no. 1, pp. 3–28, Mar. 1991.

[2] V. K. Semenov, Y. A. Polyakov, and S. K. Tolpygo, "New AC-Powered SFQ digital circuits," *IEEE Trans. Appl. Supercond.*, vol. 25, no. 3, Jun. 2015, Art. no. 1301507.

[3] T. V. Duzer and C. W. Turner, *Principles of Superconductive Devices and Circuits*, 2nd ed., 1981.

[4] K. Gaj, Q. P. Herr, V. Adler, A. Krasniewski, E. G. Friedman, and M. J. Feldman, "Tools for the computer-aided design of multigigahertz superconducting digital circuits," *IEEE Trans. Appl. Supercond.*, vol. 9, no. 1, pp. 18–38, Mar. 1999.

[5] C. J. Fourie, "Digital superconducting electronics design tools—Status and roadmap," *IEEE Trans. Appl. Supercond.*, vol. 28, no. 5, Aug. 2018, Art. no. 1300412.

[6] T. Jabbari, G. Krylov, S. Whiteley, E. Mlinar, J. Kawa, and E. G. Friedman, "Interconnect routing for large-scale RSFQ circuits," *IEEE Trans. Appl. Supercond.*, vol. 29, no. 5, Aug. 2019, Art. no. 1102805.

[7] T. Jabbari, G. Krylov, S. Whiteley, J. Kawa, and E. G. Friedman, "Repeater insertion in SFQ interconnect," *IEEE Trans. Appl. Supercond.*, vol. 30, no. 8, Dec. 2020, Art. no. 5400508.

[8] T. V. Filippova *et al.*, "20 GHz operation of an asynchronous wave-pipelined RSFQ arithmetic-logic unit," *Phys. Procedia*, vol. 36, pp. 59–65, Sep. 2012.

[9] J. Y. Kim and J. H. Kang, "High frequency operation of a rapid single flux quantum arithmetic and logic unit," *J. Korean Phys. Soc.*, vol. 48, no. 5, pp. 1004–1007, May 2006.

[10] M. Yasin, J. J. Rajendran, O. Sinanoglu, and R. Karri, "On improving the security of logic locking," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 35, no. 9, pp. 1411–1424, Sep. 2016.

[11] G. Di Crescenzo, A. Sengupta, O. Sinanoglu, and M. Yasin, "Logic Locking of Boolean Circuits: Provable Hardware-Based Obfuscation from a Tamper-Proof Memory," *Int. Conf. Innovative Security Solutions Inf. Technol. Comm.*, Romania, Nov. 2018.

[12] J. Hurtarte, E. Wolsheimer, and L. Tafoya, *Understanding Fabless IC Technology*. Newnes, 1st ed., MA, USA, Aug. 2007.

[13] I. Polian, G. T. Becker, and F. Regazzoni, "Trojans in early design steps - an emerging threat," *Proc. Conf. Trustworthy Manuf. Utilization Secure Devices*, Nov. 2016, pp. 1–6.

[14] M. M. Tehranipoor, U. Guin, and D. Forte, *Counterfeit Integrated Circuits*. Springer, Feb. 2015, pp. 15–36.

[15] T. Huffmire *et al.*, "Managing security in FPGA-Based embedded systems," *IEEE Des. Test Comput.*, vol. 25, no. 6, pp. 590–598, Dec. 2008.

[16] J. A. Roy, F. Koushanfar, and I. L. Markov, "EPIC: Ending piracy of integrated circuits," in *Proc. IEEE/ACM Design, Automat. Test Conf. Europe*, Apr. 2008, pp. 1069–1074.

[17] P. Prinetto and G. Roascio, "Hardware security, vulnerabilities, and attacks: A comprehensive taxonomy," *Proc. Italian Conf. Cybersecurity*, Feb. 2010, pp. 177–189.

[18] J. A. Roy, F. Koushanfar, and I. L. Markov, "Ending piracy of integrated circuits," *Computer*, vol. 43, no. 10, pp. 30–38, Oct. 2010.

[19] H. Kumar, T. Jabbari, G. Krylov, K. Basu, E. G. Friedman, and R. Karri, "Toward increasing the difficulty of reverse engineering of RSFQ circuits," *IEEE Trans. Appl. Supercond.*, vol. 30, no. 3, Apr. 2020, Art. no. 1700213.

[20] T. Thangam, G. Gayathri, and T. Madhubala, "A novel logic locking technique for hardware security," in *Proc. IEEE Int. Conf. Electrical, Instrum. Commun. Eng.*, Dec. 2017, pp. 1–7.

[21] R. Torrance and D. James, "The state-of-the-art in semiconductor reverse engineering," in *Proc. ACM/EDAC/IEEE Des. Automat. Conf.*, Aug. 2011, pp. 333–338.

[22] 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, Aug. 2014.

[23] B. Erbagci, C. Erbagci, N. E. C. Akkaya, and K. Mai, "A secure camouflaged threshold voltage defined logic family," in *Proc. IEEE Int. Symp. Hardware Oriented Secur. Trust*, Jun. 2016, pp. 229–235.

[24] G. Krylov and E. G. Friedman, "Design for testability of SFQ circuits," *IEEE Trans. Appl. Supercond.*, vol. 27, no. 8, Dec. 2017, Art. no. 1302307.

[25] G. Krylov and E. G. Friedman, "Design methodology for distributed large-scale ERSFQ bias networks," *IEEE Trans. Very Large Scale Integration Syst.*, vol. 28, no. 11, pp. 2438–2447, Nov. 2020.

[26] G. Krylov and E. G. Friedman, "Bias distribution in ERSFQ VLSI circuits," in *Proc. IEEE Int. Symp. Circuits Syst.*, Oct. 2020, pp. 1–5.

[27] F. Brglez and H. Fujiwara, "A neutral netlist of 10 combinational benchmark circuits," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 1985, pp. 685–698.