Rasit O. Topaloglu *Editor*

Design Automation of Quantum Computers

Springer

Hardware Security of SFQ Circuits

Tahereh Jabbari, Yerzhan Mustafa, Eby G. Friedman, and Selçuk Köse

1 Principles of SFQ Logic

The fundamental principles of single flux quantum (SFQ) logic are described in this section. The operation of a Josephson junction (JJ) is described in Sect. 1.1. SFQ logic is explained in Sect. 1.2. The principles of hardware security of SFQ circuits are described in Sect. 1.3.

1.1 Josephson Junctions

Superconductive materials exhibit zero electrical resistance when cooled below a temperature known as the critical temperature T_C [1]. The Josephson effect is described as quantum tunneling in a superconductor across a thin insulator barrier by overlap of the wave function of a Cooper pair in two superconductive layers [2]. The operation of a JJ is based on this effect. A JJ which consists of two superconductive niobium layers separated by a thin layer of oxide [3] is the primary active device in superconductive electronics. A JJ loses superconductivity when the bias current, temperature, or magnetic field exceeds, respectively, a critical current I_C, critical temperature T_C, or critical magnetic field B_C. The structure of a JJ is illustrated in Fig. 1a. JJs are modeled as a resistively and capacitively shunted junction (RCSJ) which is a depicted in Fig. 1b. The I–V characteristics of a junction are shown in Fig. 1c.

T. Jabbari (✉) · Y. Mustafa · E. G. Friedman · S. Köse
Department of Electrical and Computer Engineering, University of Rochester, Rochester, NY, USA
e-mail: tjabbari@ur.rochester.edu; ymustafa@ur.rochester.edu; friedman@ece.rochester.edu; selcuk.kose@rochester.edu

R. O. Topaloglu (ed.), *Design Automation of Quantum Computers*,
https://doi.org/10.1007/978-3-031-15699-1_7

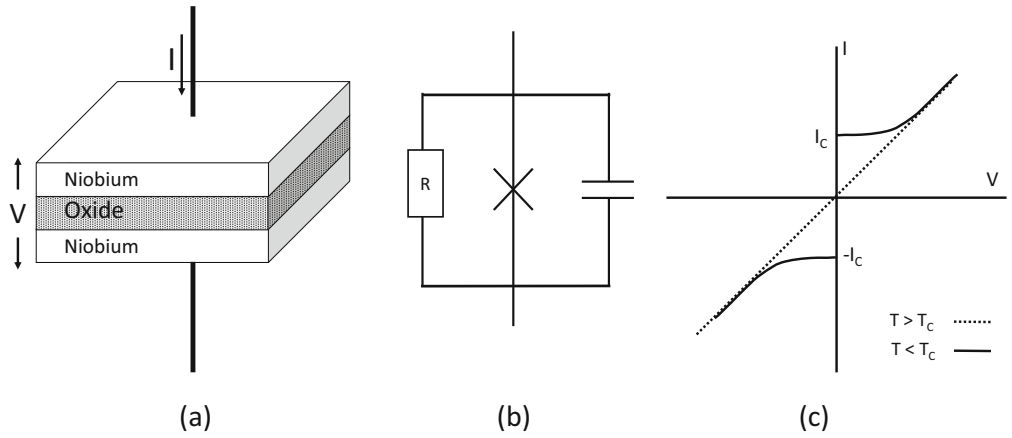

Fig. 1 Josephson junction, (**a**) structure, (**b**) RCSJ model, and (**c**) I–V relationship

1.2 SFQ Logic

An SFQ circuit consists of JJs and inductors. In SFQ circuits, information is transferred in the form of picosecond duration voltage pulses $V(t)$ within a quantized area [4–6]. Elementary logic gates in this circuit family can generate, pass, store, and reproduce picosecond voltage pulses. Switching a JJ is described as a 2π change in phase, producing a voltage pulse equal to a quantum of flux ($\phi_\circ = 2.07 \times 10^{-15}$ $V{\cdot}s$) [7],

$$\int V(t)dt = \phi_\circ \equiv \frac{h}{2e}\,, \tag{1}$$

where $\phi_\circ$ is a single flux quantum, and h and e are, respectively, the Planck constant and electron charge.

The existence of a flux quantum represents a logic '1,' whereas the absence of a pulse is a '0.' In SFQ circuits, a clock signal is required for most logic gates (except for splitters, Josephson transmission lines (JTLs), buffers, and mergers [6, 8–10]). In these clockless gates, the propagation delay is the delay of the output with respect to the input. Alternatively, in clocked gates, the incoming SFQ pulse changes the internal state of an SFQ gate but does not change the output. The output changes only when a clock pulse arrives at a gate. The propagation delay is measured as the time elapsed after arrival of the clock pulse. SFQ gates are, therefore, similar to CMOS logic gates combined with an edge triggered flip flop.

1.3 Hardware Security of SFQ Circuits

VLSI complexity superconductive SFQ systems is one of the most promising beyond CMOS technologies for ultra-low power and ultra-high speed digital

applications [5, 6]. Significant developments in the design and fabrication of superconductive electronics have resulted in device densities exceeding 600,000 Josephson junctions/cm^2 [11, 12]. Josephson junctions in SFQ circuits propagate an SFQ pulse through logic gates operating at switching speeds on the order of picoseconds, while dissipating power below 10^{-19} J [4, 13–18]. An SFQ-based arithmetic logic unit has been demonstrated to operate at frequencies approaching 80 GHz with an 8 bit SFQ datapath [19, 20].

Prospective exascale computing systems based on VLSI complexity SFQ circuits are expected to be used not only for high performance computing but also for critical security tasks. Hardware security for superconductive technology [21, 22] and novel techniques for providing trustworthy hardware based on SFQ circuits are therefore necessary. Security aware design methodologies for this technology are currently not well established. Recent progress in the fabrication and design of SFQ circuits strengthens the need for hardware security techniques targeting SFQ circuits. Furthermore, SFQ technology exhibits unique advantages and challenges, which should be considered when developing these hardware security techniques.

Due to the increasing complexity of modern systems-on-chip with advanced fabrication capabilities and higher manufacturing costs, many semiconductor companies have become fabless [23]. These fabless companies design the integrated circuits in-house while utilizing external foundries for fabrication, manufacturing, and integration. Although the cost of the IC production supply chain can be reduced by outsourcing certain processes to external foundries, this process also introduces security vulnerabilities into the systems integration design flow.

With an increasingly distributed IC production supply chain, different stages of the supply chain have become vulnerable to a number of attack vectors, such as counterfeiting, reverse engineering (RE), and intellectual property (IP) piracy. An attacker may insert a hardware Trojan at any point during the design, fabrication, manufacturing, or integration of an IC—either as on-chip circuitry or as an external component to perform additional malicious operations. Other vulnerabilities that can be introduced during the IC production supply chain are IC counterfeiting [21, 24], theft of IC masks [25], overproduction of ICs [26], and insertion of hardware Trojans [27].

Technology companies annually lose up to $4 billion due to IP violations in semiconductor technology [28, 29]. Hardware security has been established to mitigate the risks of piracy, counterfeiting, reverse engineering, and side-channel attacks [30]. 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 potentially be thwarted. It is therefore important to an IC design company to protect this design flow. Counterfeiting is typically thwarted by IC camouflaging [22] 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.

Reverse engineering poses a major challenge to hardware security. RE is the process of analyzing the layout and functionality of a system to extract the gate-level netlist. RE can be performed as a non-invasive attack or as an invasive attack. Non-invasive RE attacks can be performed in combination with side-channel attacks where an attacker collects certain side-channel emanations such as power consumption [31–33], electromagnetic (EM) signals [34], or timing information to deduce the functionality of a circuit. In non-invasive RE attacks, an attacker does not leave an obvious footprint, making the attack difficult to detect. Alternatively, an invasive RE attack requires more advanced imaging and circuit analysis tools and may require several steps to extract the netlist. Invasive attacks typically can recover the netlist more accurately than non-invasive attacks. The initial step of an invasive RE attack is 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 [35]. 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.

IC camouflaging and logic locking, respectively, a layout technique and a circuit technique [21, 22], are widely used to mitigate the threat of RE attacks on hardware. 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. IC camouflaging in SFQ circuits obstructs the reverse engineering process by introducing dummy (i.e., redundant) JJs into a layout [22]. In camouflaged SFQ cells, normal and dummy JJs are both used. When certain JJs are necessary to maintain correct logical operation, the layout is slightly changed and dummy JJs are replaced with normal JJs. This technique relies on making these JJs indistinguishable to the attacker, who extracts an incorrect netlist. Distinguishing between a required JJ and a dummy JJ is difficult with RE attacks, which typically utilize delayering and analysis of the top view image of the layout. RE can only distinguish the 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 systems, and the small difference in the thickness of the tunneling barrier between a normal and dummy JJ [22]. 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 without the valid key.

2 Design of SFQ Camouflage Cells

IC camouflaging in SFQ thwarts RE attacks by introducing camouflaged cells and dummy JJs along with regular cells into a standard cell library. In this section, dummy JJs, camouflaged SFQ AND/OR gates, and camouflaged SFQ flip flops are reviewed. The structure of a dummy JJ to thwart RE is introduced in Sect. 2.1. The use of dummy JJs in SFQ AND/OR gates is described in Sect. 2.2. The use of a dummy DFF as a JTL is introduced in Sect. 2.3.

2.1 Dummy Josephson Junction

A dummy JJ never switches into the superconducting state and always behaves as a resistor. JJs are fabricated as a sequence of Nb–AlO_X–Nb layers where the AlO_X layer is the insulator. The critical current density of a JJ depends upon the thickness of the AlO_X tunneling barrier [36]. Changing the thickness of the insulating layer and the quality of the superconductive material affects the switching characteristics of a JJ. Two approaches to fabricate a dummy JJ are considered. These approaches increase the fabrication cost by requiring two additional mask steps.

2.1.1 Method 1: Vary Insulator Thickness of JJ

The critical current density and thickness of the insulation layer depend upon the SFQ fabrication technology and the physical design rules. The thickness of AlO_X is currently about 1 nm in a standard JJ technology [36]. By increasing the thickness of AlO_X beyond the ∼38 nm coherence length of the Nb layer, a dummy JJ can be fabricated that always behaves as a resistor [37]. The magnitude of the resistance depends upon the insulator thickness.

While an attacker can differentiate between a true JJ and a dummy JJ by slicing the die and imaging a side view, this strategy does not scale due to the large number of JJs in a typical SFQ circuit. Hence, slicing an IC to decipher the function of every JJ is extremely challenging. Alternatively, a top view image of a dummy JJ is identical to a standard JJ. Hence, it is extremely difficult and costly to distinguish between two JJs using image-based RE.

To tune the McCumber damping parameter β of a JJ [4, 5], most of the JJs in current fabrication processes are shunted with a resistance [5]. A cross section of a JJ with a shunt resistor is shown in Fig. 2. The structure is composed of two Nb layers, a stack of Nb-Al-Al_2O_3-Nb for the JJ, a Mo layer for the shunt resistors, and Nb_2O_5 and SiO_2 for the isolation layers. A thicker insulator film yields a dummy JJ that isolates the superconductive current. The minimum thickness of Al-Al_2O_3 in a dummy JJ is 40 nm .

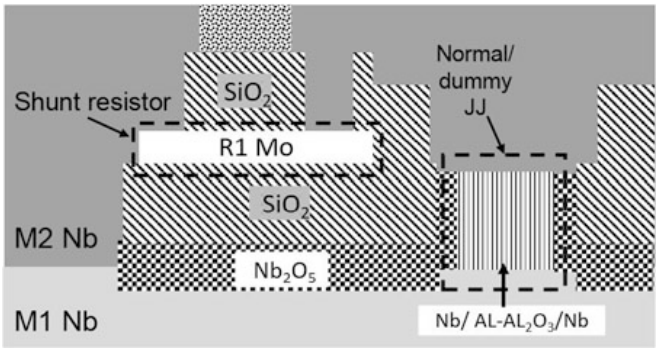

Fig. 2 Cross section of a normal and dummy JJ with a shunt resistor between the M1 and M2 layers [36]

Dummy JJs are shunted with a resistor to appear identical to a normal JJ. A small shunt resistor with a dummy JJ can degrade the operation of a camouflaged cell. Decreasing the thickness of the Mo layer increases the shunt resistance and prevents deterioration of the camouflaged cell. Two approaches exist to tune the thickness of a JJ, either by addition or by elimination.

- **A double deposition process** can be used to fabricate JJs to achieve the proper critical current density for a normal JJ and to maintain the resistive behavior for a dummy JJ. The initial deposition process determines the critical current of a normal JJ. A thicker deposition layer can be used for a dummy JJ based on the coherence length. As compared to normal junctions, a dummy JJ increases the fabrication time by adding several steps. As compared to other methods to fabricate a dummy JJ, a double deposition process offers benefits that include an accurate thickness for the normal and dummy JJs and a shorter fabrication time.
- **Ion beam etching** A thick AlO_X layer is deposited for the dummy JJs. This step is followed by an ion beam etch to fabricate a normal JJ. The etching time, surface roughness, and insulator depth determine the switching characteristics of the JJ.

2.1.2 Method 2: Damage the Nb Layer

In this method, the top Nb layer of a JJ is bombarded with an ion beam to damage and degrade the properties of the Nb surface of the JJ. The ion beam smooths the surface where the damage depends upon the energy, temperature, and angle of the beam. The properties of the ions used to bombard the surface are enhanced depending upon the thickness and material of the film affected by the ion beam. Several materials produce different effects on the Nb layer. To remove an undesirable surface, an Ar or He ion beam can be used [38, 39]. Dummy JJs, fabricated using an ion beam, have a Nb layer thickness identical to a normal JJ. Furthermore, bombarding the top Nb layer with carbon ions alters the superconductive properties (e.g., eliminates the superconductive current due to the large impurity concentration within the niobium). Applying these methods, a normal JJ and dummy JJ can be separately fabricated with different and specific parameters.

A dummy JJ can therefore be included within a fabricated SFQ circuit without being recognized to thwart RE.

2.2 *Camouflaged SFQ AND/OR Logic Cell*

Dummy JJs are used to camouflage AND/OR SFQ gates to appear as either a two-input AND or OR gate. A camouflaged AND/OR gate exploits the structural similarity of AND and OR gates to ensure low overhead. A schematic of a camouflaged AND/OR gate with dummy JJs is shown in Fig. 3. For an AND gate, J9 and J11 are dummy JJs. Similarly, J13 is a dummy JJ within an OR gate. For analysis purposes, a dummy JJ is modeled as a large resistor in parallel with a small shunt resistor (a normal shunt resistor is 2 to 5 ohms). The high operating speed of the camouflaged cell is retained by reducing the thickness of the shunt resistor in the dummy JJ. The resistance of a shunted dummy JJ is 24 ohms. A simulation of a camouflaged AND and OR gate is shown, respectively, in Figs. 4a and 4b. A cell layout of the camouflaged AND/OR is shown in Fig. 5. The layout is based on 4.5 kA/cm^2 Hypres SFQ design rules [40].

The output delay, power, and area depend upon the number of dummy JJs. Due to the small shunt resistor in a dummy JJ, the current passing through the JJ after each input pulse is significant. Hence, the power and output delay of a camouflaged gate can be reduced by lowering the shunt resistance.

The energy dissipated by the dummy JJs in a camouflaged AND/OR is shown in Fig. 6. Averaging the energy over one clock cycle produces a power overhead of 100 pW for an AND gate with two dummy JJs, and 30 pW for an OR gate with one dummy JJ at an operating frequency of 10 GHz. The dissipated energy is approximately 2 to 5% higher than a standard SFQ OR and AND gate. The delay of the camouflaged AND/OR gates is 11 ps as compared to a delay of 10 ps for a standard AND and 7 ps for a standard OR gate. As compared to standard AND and OR gates, the area overhead of the camouflaged AND and OR gates is, respectively, approximately 15% and 10%. Since the dissipated energy is higher for camouflaged gates as compared to standard SFQ gates, one might wonder whether a side-channel

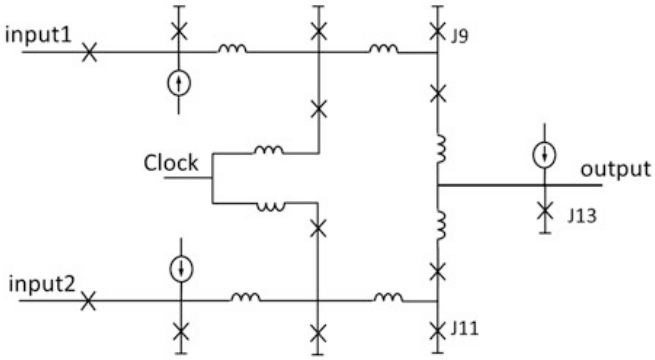

Fig. 3 Camouflaged SFQ AND-OR cell, J9 and J11 are dummy JJs for the AND gate, and J13 is a dummy JJ for the OR gate

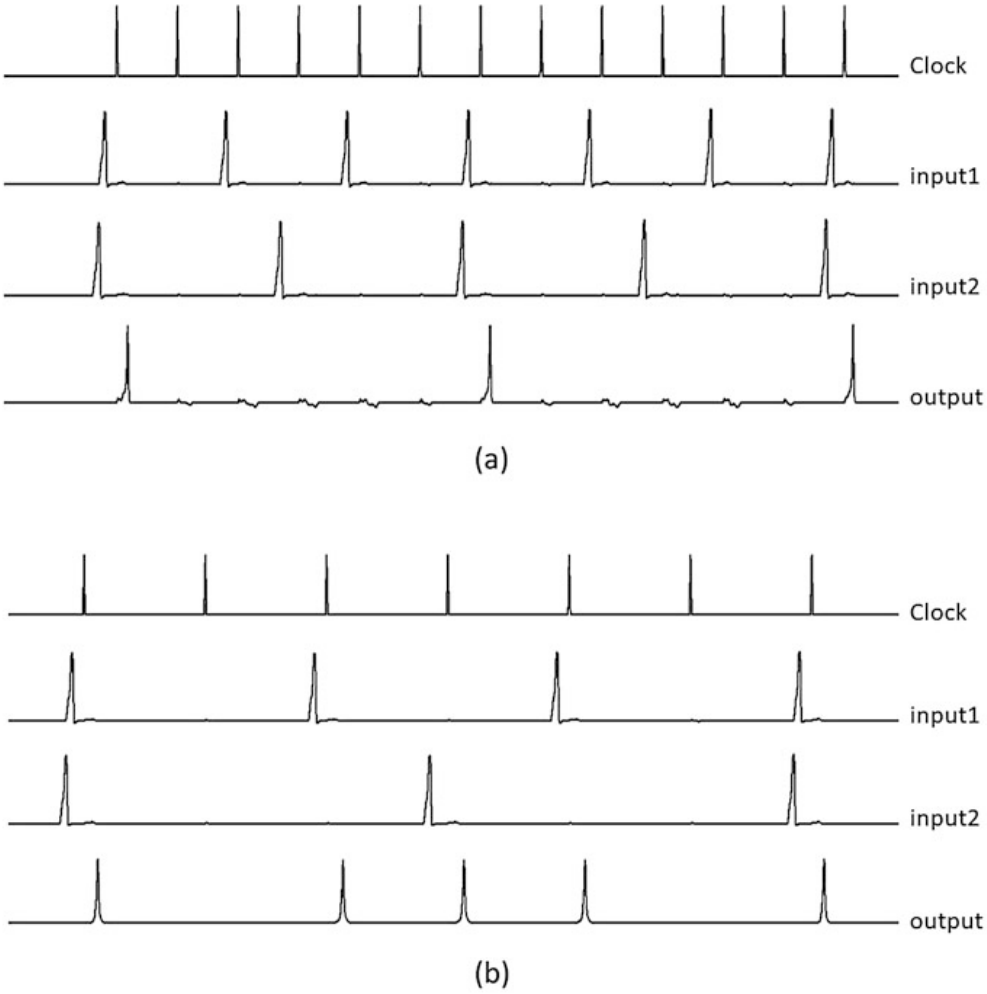

Fig. 4 Operation of SFQ cells, (**a**) AND gate (J9 and J11 are dummy JJs), and (**b**) OR gate (J13 is a dummy JJ)

attack [22] can distinguish between the two logic topologies. The energy dissipation due to JJ switching is quite low (i.e., $\sim 10^{-19}$ J), making an accurate measurement highly challenging. Hence, power side-channel attacks appear to be infeasible.

2.3 Camouflaged SFQ D Flip Flop

A Josephson transmission line (JTL) propagates a fluxon (ϕ_o) through a number of stages. A JTL improves the performance of an SFQ circuit by amplifying the SFQ pulse between logic stages. A camouflaged SFQ D flip flop (DFF) can function as

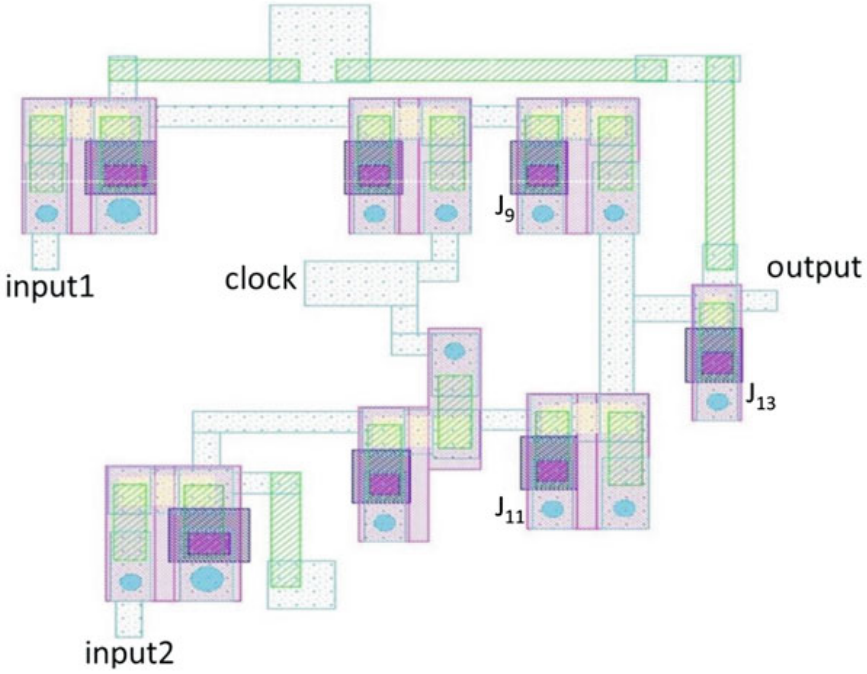

Fig. 5 Layout of a camouflaged SFQ AND-OR cell. J9 and J11 are dummy JJs for the AND gate, and J13 is a dummy JJ for the OR gate

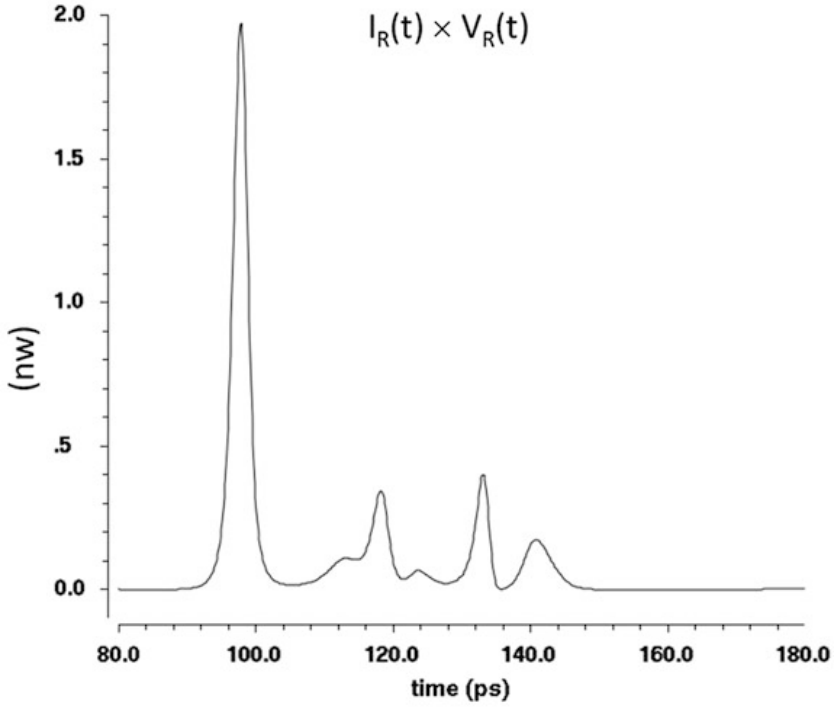

Fig. 6 Power dissipation of a dummy JJ within the AND-OR cell

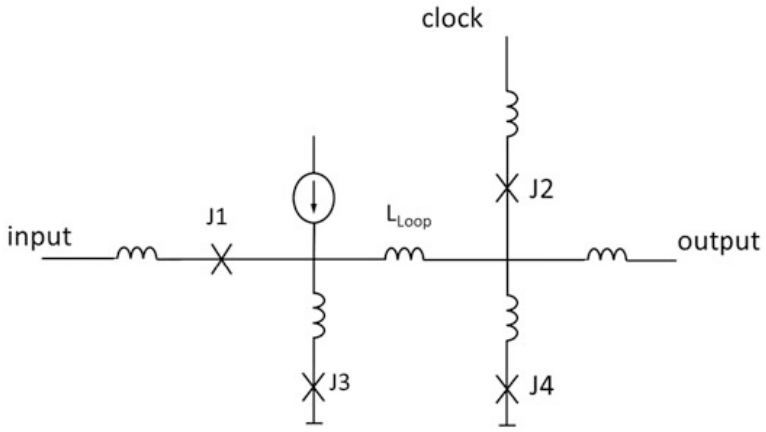

Fig. 7 Camouflaged SFQ DFF J4 is a dummy JJ

a JTL or as a standard D flip flop. A schematic of a camouflaged DFF is shown in Fig. 7. Note that the camouflaged DFF appears the same as a standard DFF.

In the camouflaged DFF shown in Fig. 7, J4 is a dummy JJ and behaves as a resistor. By adjusting the thickness of the insulating layer, the resistance is increased to lower the output delay and power dissipation. In a standard DFF, L_{Loop} is large, storing the information (i.e., bit value) when the content is read. To achieve the same physical layout, the length of the inductor in the camouflaged DFF is maintained the same as a regular DFF. Consequently, the large kinetic inductance in a camouflaged DFF produces a large output delay when functioning as a JTL. To circumvent this effect, the inductance is reduced to decrease the delay. This smaller inductance can be achieved by increasing the thickness of the kinetic inductance, thereby decreasing the overall inductance. Since a JTL is asynchronous and does not require a clock, the clock signal is eliminated by reducing the critical current of J2 by increasing the thickness of the insulator layer. Simulation results of a camouflaged DFF functioning as a JTL is shown in Fig. 8. By changing the thickness of the Nb, Mo, and insulator layers to the standard thickness, the functionality of a standard DFF can be achieved.

The output delay of a camouflaged DFF is approximately 11 ps which is roughly twice the delay of a standard JTL. This difference is attributed to the large inductance and two different input pulses—the clock and data signals. The throughput of the circuit is halved when the camouflaged DFF is part of the critical path. The current through the dummy JJ varies depending upon the clock frequency and the input signal of the camouflaged DFF. By assuming a frequency of 10 GHz for the input and clock pulses, the total power dissipated by the dummy JJ is approximately 100 pW. The energy dissipation of the camouflaged DFF is approximately the same as a standard DFF due to the different critical current of the JJs. The energy dissipated by a camouflaged DFF is approximately 2% more than

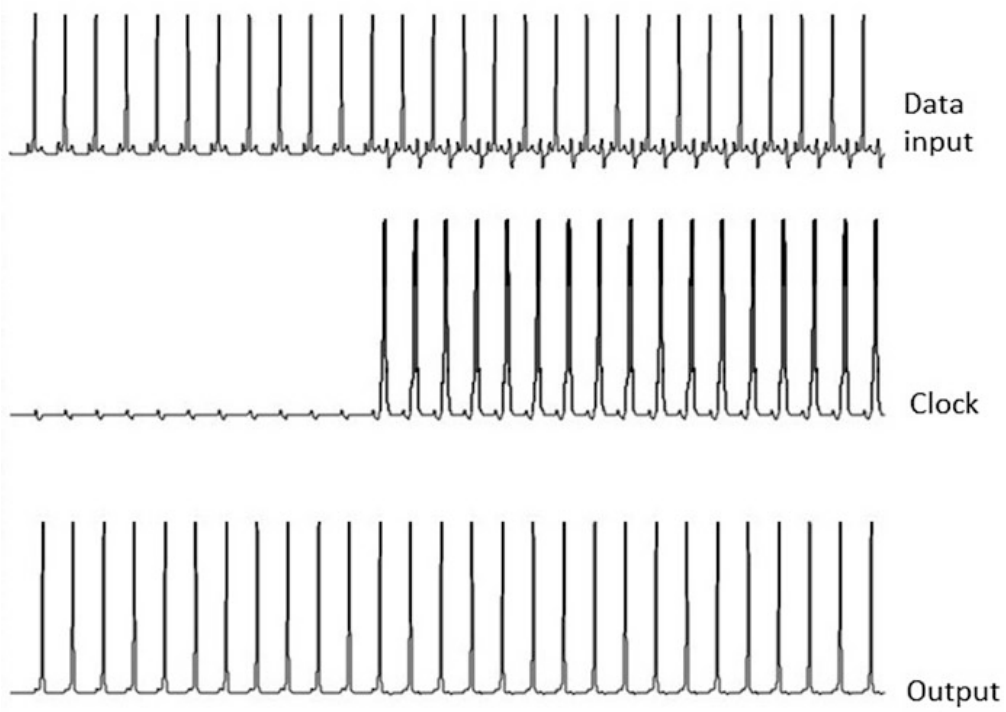

Fig. 8 Camouflaged DFF operating as a JTL. The JTL passes the input pulses regardless of the clock

a standard JTL. The top view of the camouflaged DFF is identical with a different thickness for J2 and dummy J4. A camouflaged DFF therefore exhibits the same area as a standard DFF. Furthermore, the area of a camouflaged DFF is approximately twice as large as a standard JTL.

2.4 Hardware Cost

A tradeoff between hardware security and physical area exists in camouflaged SFQ systems. ISCAS'85 benchmark circuits are used here to characterize the area and power overhead of camouflaged gates when applied to large scale SFQ circuits. Standard SFQ gates are replaced by camouflaged gates. Dummy DFFs are inserted at the inputs. The area and power overhead of the camouflaged gates as compared to standard SFQ gates is listed in Table 1. The area overhead and power overhead for these benchmarks are, respectively, approximately 40% and 30% if all of the gates are replaced with camouflaged gates, as shown in Fig. 9. The overhead is greater if additional camouflaged gates are used, providing enhanced security.

Table 1 Overhead of camouflaged SFQ cells as compared to standard SFQ cells. A camouflaged DFF behaves as a standard JTL or DFF. The camouflaged AND/OR gate behaves as a standard AND or OR cell

	Camouflaged gate					
	DFF/JTL			AND/OR		
Standard gates	Power	Delay	Area	Power	Delay	Area
DFF	0%	0%	0%	N/A		
JTL	2%	100%	100%	N/A		
AND	N/A			2 to 5%	50%	15%
OR	N/A			2 to 5%	50%	10%

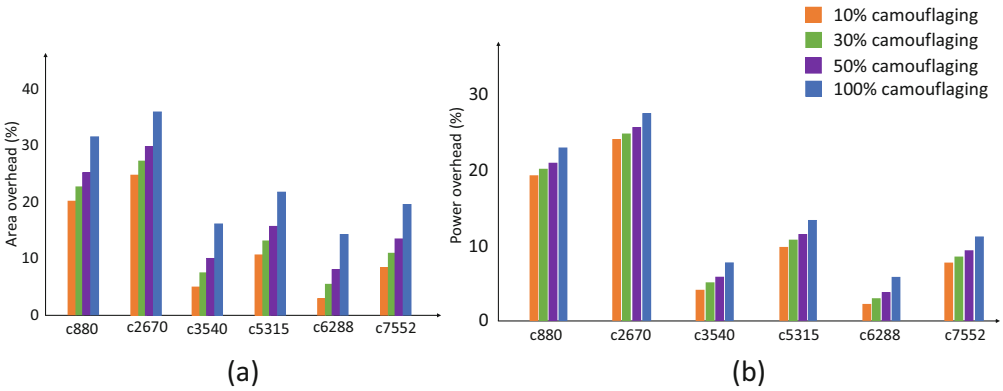

Fig. 9 Overhead of camouflaged gates, (**a**) area, and (**b**) power

3 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) [41]. An example of a locked circuit with key gates is shown in Fig. 10. The key inputs are K_1 and K_2 which connect to the key gates. The correct output is only produced if a correct value of the keys are applied [26]. An incorrect key used with a logic locked design causes incorrect or random operation.

Since the correct key is known only by 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 becomes infeasible. Furthermore, this technique prevents an external attacker from analyzing the structural behavior of the design even if another copy of the secured circuit is obtained.

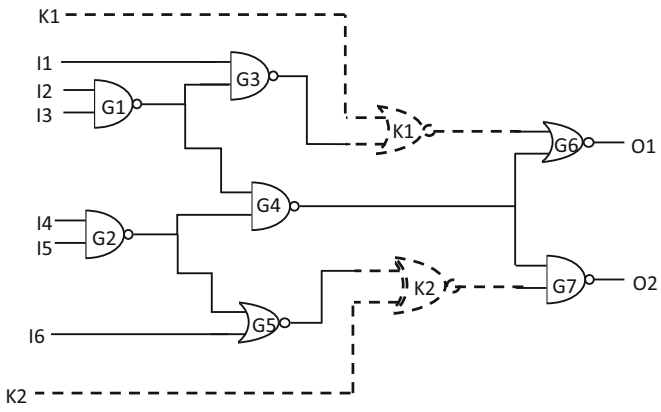

Fig. 10 Circuit utilizing logic locking with two key gates, K_1 and K_2

3.1 Threat Model of Attacks on Logic Locking

The primary objective of an attack on a logic locked circuit is to determine the correct value of the secret keys to decipher a functional netlist. If the keys are determined and the design is deciphered, the optimum location to insert a stealthy hardware Trojan can also be determined.

Different input patterns can be applied to both the circuit and 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 details of the circuit design are required. The netlist can be obtained from reverse engineering a GDSII layout file, masks, or an activated functional IC. With the increasing complexity of circuits and a large number of key inputs, these attacks become highly infeasible.

The importance of hardware security in SFQ circuits is emphasized by one of the primary prospective applications of these circuits—large scale data centers typically operating with sensitive information. Countermeasures to attacks applicable to SFQ circuits are discussed in the following sections.

4 Logic Locking in SFQ Circuits

Logic locking complicates the attacks, thereby improving the security of the SFQ circuits. Existing CMOS logic locking techniques rely on introducing additional gates, look up tables, and external inputs into the design [41]. Logic locking can be similarly applied to SFQ circuits without additional modifications. The necessary gates, however –typically XOR/XNOR and multiplexers– are expensive in terms of pysical area. LUTs also require significant area. Additionally, the pinout limitations of modern superconductive ICs severely limit the size of the secret key, compromising security.

A methodology for logic locking in SFQ circuits is proposed here [21]. 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 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. The range of key currents shrinks with an increase in the number of locked cells, enhancing the security of the system. In the following section, this proposed logic locking technique is evaluated in terms of the security of SFQ circuits.

4.1 Modified OR Gate for Logic Locking

A modified OR gate is shown in Fig. 11. Mutual inductances are used to apply additional secret key current to unlock the correct functionality of an OR gate. In this circuit, the mutual inductance between L1 and L_{M1} and between L2 and L_{M2} is used to apply the key currents. The dependence of the additional secret key current on the input current and coupling coefficient K_i is

$$I_{L_i} \propto K_i I_{in}. \tag{2}$$

The inductive coupling coefficient K_n changes the current through the inductance of the internal gate. 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

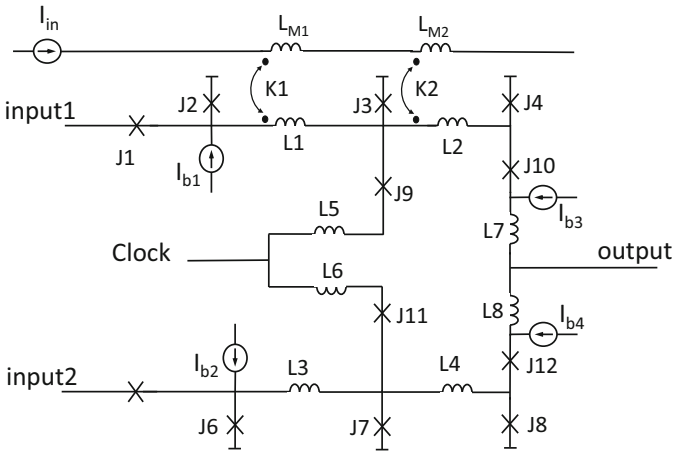

Fig. 11 Locked OR gate with mutual inductances to apply a secret key current. L1 = L3 = 15.1 pH, L2 = L4 = 3.8 pH, L5 to L8 = 5.68 pH, $L_{M1} = L_{M2} = 1$ pH, $I_{in} = 250\,\mu$A, $I_{b1,b2} = 176\,\mu$A, $I_{b3,b4} = 172\,\mu$A, $I_{C_i} = 176\,\mu$A, and $I_{c2,c6} = 250\,\mu$A (for a 10 kA/cm^2 technology)

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 affects the current within the state storage loop as well as switching junction J3. The 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}.

To unlock this OR gate, an attacker needs to determine the correct value of the key current. The correct output is only produced when the key current with a correct value is provided. Incorrect key currents coupled to 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. 12a. The circuit incorrectly produces an output after the second and third pulse of input 1 (see Fig. 12a). 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. 12b. The correct key current is described by

$$-1 \leq K_i \leq 1; \qquad 0 \leq I_{in} \leq I_{in_{max}}, \tag{3}$$

where $I_{in_{max}}$ is the maximum input current that can be supplied to the circuit. With a greater number of locked OR cells, the key current exponentially increases.

5 Security Characteristics of Logic Locking

An analysis of the security characteristics of the proposed technique is presented in Sect. 5.1. The area of the proposed logic locking technique based on a modified OR gate is quantified in Sect. 5.2.

5.1 Analysis of Security Characteristics

To increase the security of the proposed technique, a 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. 13. Each range of K_2 is a specific additional key current. The range of additional current is listed in Table 2. The key current margins are described as margins of K_2. To unlock the circuit, the correct value of K_1, range of K_2, and range of key current need to be determined. With $K_1 = 0.3$, the range of coupling coefficient K_2 is $0.25 < K_2 < 0.35$. For smaller K_1, the circuit exhibits a large range of K_2, resulting in

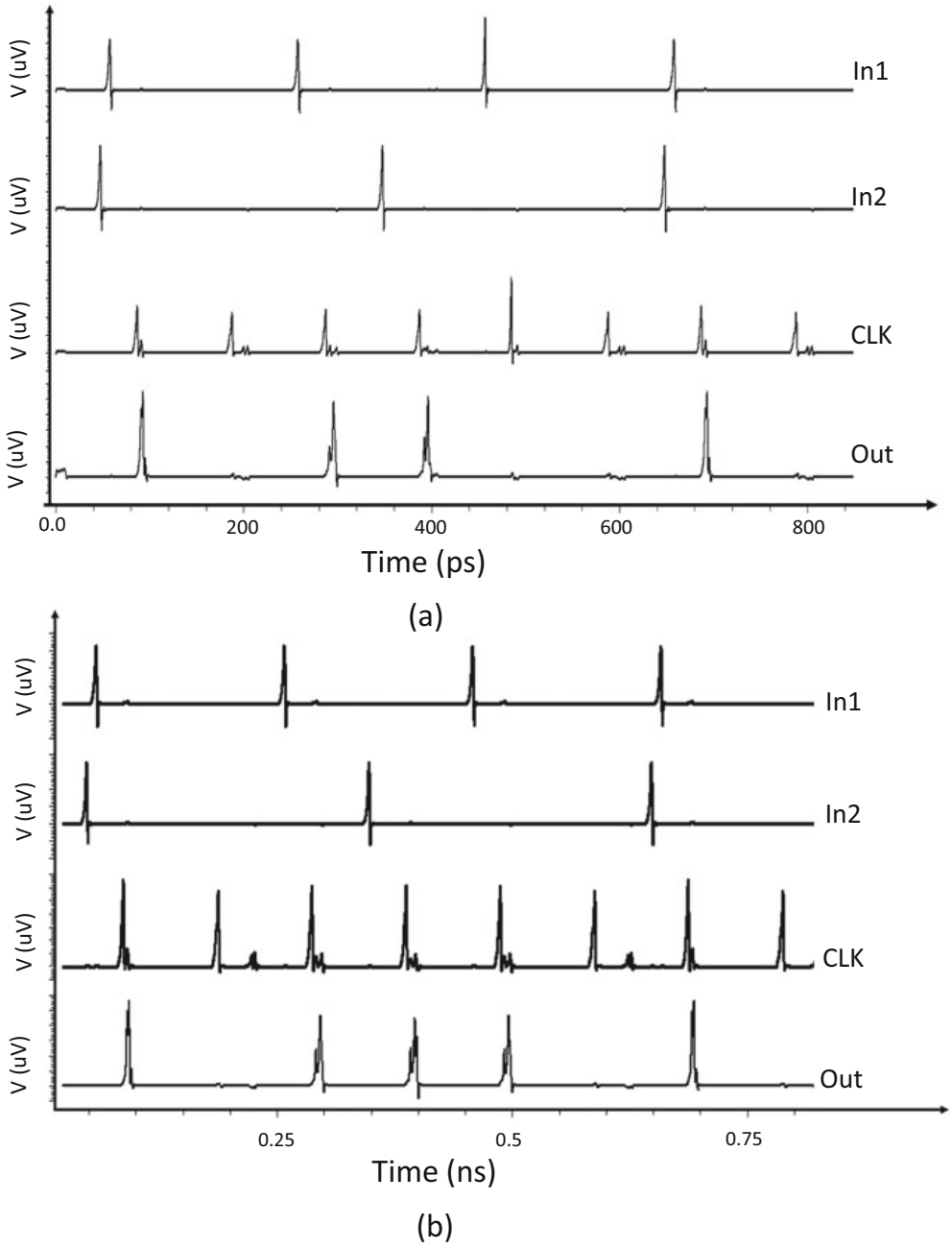

Fig. 12 OR gate operation, (**a**) incorrect current key currents with K1 = 0.5 and K2 = 0.5, and (**b**) correct current key currents with K1 = 0.3 and K2 = 0.3

lower security as compared to a higher K_1. A narrower range of K_n increases the effort required by the attacker to determine the secret key current.

Manufacturing process variations is a challenging issue in all large scale ICs. 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

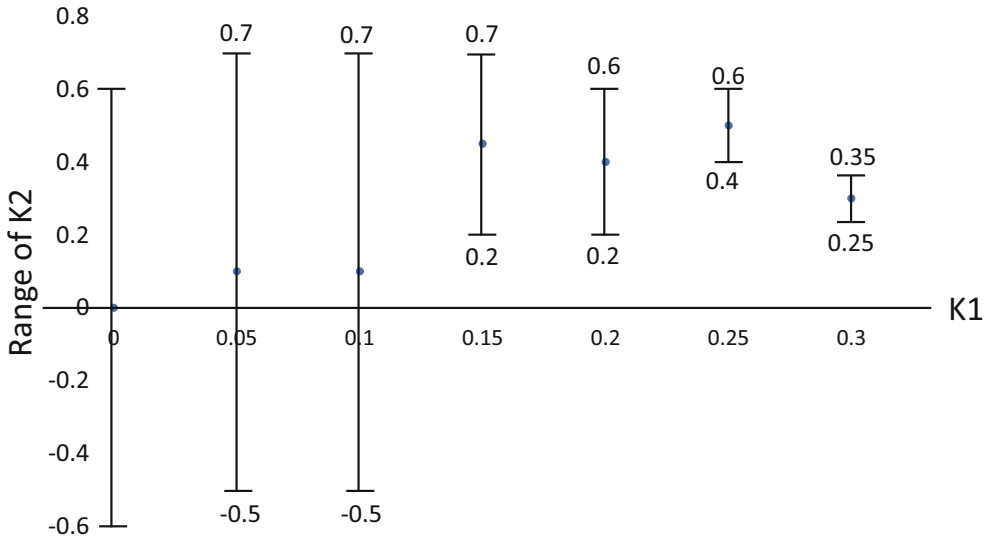

Fig. 13 Security characteristics of an OR gate for different ranges of coupling coefficients

Table 2 Range of key currents for different range of coupling coefficients

Coupling coefficient	Coupling coefficient	Current through L1	Current through L2
$K1 = 0$	$K2 = 0$	$9\,\mu A$	$-33\,\mu A$
$K1 = 0$	$-0.6 < K2 < 0.6$	$5\,\mu A$ to $13\,\mu A$	$-73\,\mu A$ to $5\,\mu A$
$K1 = 0.1$	$-0.5 < K2 < 0.7$	$12\,\mu A$ to $19\,\mu A$	$-66\,\mu A$ to $12\,\mu A$
$K1 = 0.2$	$0.2 < K2 < 0.6$	$21\,\mu A$ to $24\,\mu A$	$-17\,\mu A$ to $7\,\mu A$
$K1 = 0.3$	$0.25 < K2 < 0.35$	$27\,\mu A$ to $28\,\mu A$	$-13\,\mu A$ to $-7\,\mu A$

should be wider than any expected bias variations caused by manufacturing and the bias distribution network [42]. Process variations can improve the overall security of a logic locked system, further protecting the circuit. Unlike the intended user of an IC, the correct operation of an IC is hidden from the attacker, inhibiting a brute force attack.

Multiple locked gates can be connected to the same source of key current. These gates utilize a different magnitude and direction 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 the case of greater manufacturing variations.

5.2 Area Overhead

An important tradeoff exists between the level of security and dedicated physical area required by the proposed logic locking technique. The area overhead of the logic locked OR gate described here is approximately 20%. ISCAS'85 benchmark

Table 3 Characteristics of ISCAS'85 benchmark circuits [43] with locked OR gates

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	1193	89	0.15%	0.3%
c3540	1669	160	0.2%	0.4%
c5315	2406	241	0.2%	0.4%
c6288	2406	2128	1.77%	3.6%
c7552	3512	298	0.17%	0.3%

circuits are used here to characterize the area overhead of the proposed technique when applied to large scale circuits. In the benchmark circuits listed in Table 3, the OR and NOR (OR + NOT) gates are replaced with locked OR gates to produce a narrow range for the correct key current. The number of OR gates within each benchmark circuit is listed in Table 3. Only a few locked OR gates are necessary to have a considerable impact on the security of the system. The area overhead of these benchmark circuits is also listed in Table 3, assuming 10% and 20% of the OR gates are replaced by locked OR gates. In the c6288 benchmark circuit which includes 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 required 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.

6 Attack Models on Logic Locking

One of the better known methods for attacking logic locking in CMOS circuits is a Boolean satisfiability-based attack (SAT attack) [44]. The objective of this attack method is to reduce the key space, and thus the computational time of a brute force attack. Similar attacks can be applied to SFQ circuits once a locked gate is characterized. Two attacks can target logic locking in SFQ circuits; resetting the locked OR cell and overproduction of complex locked circuits [45]. Similar to CMOS, logic locking is unable to secure SFQ circuits against these two attacks. Resetting the locked OR cell under attack is described in Sect. 6.1. The overproduction model is described in Sect. 6.2.

6.1 Attack Model #1: Increasing Correct Key Space of Locked SFQ Circuits

In the reset attack model, the range of correct key values for the locked OR cell is significantly increased by applying a specific input combination. In this section,

the attack model is applied on a locked OR cell to evaluate the region of the correct key space. The threat model and attack scenario are described, respectively, in Sects. 6.1.1 and 6.1.2. Related countermeasures to prevent a reset attack are discussed in Sect. 6.1.3.

6.1.1 Threat Model

Based on Krckhoffs' principle [46], in the logic locking threat model, an attacker has access to the simulation files of the circuit, but has no knowledge of the key values. The attacker therefore knows the mutual inductances, L_1, L_2, L_{M1}, and L_{M2}, for the secret key current and can simulate the locked OR cell using arbitrary key values. The coupling coefficients are evaluated to determine the regions of operation of the locked OR cell. By sweeping the coupling coefficients from -1 to 1 at the maximum input current, the operation of the single locked OR cell (see Fig. 11) can be characterized, as shown in Fig. 14. The "C" and "I" regions represent, respectively, correct and incorrect operation. The "R" region represents

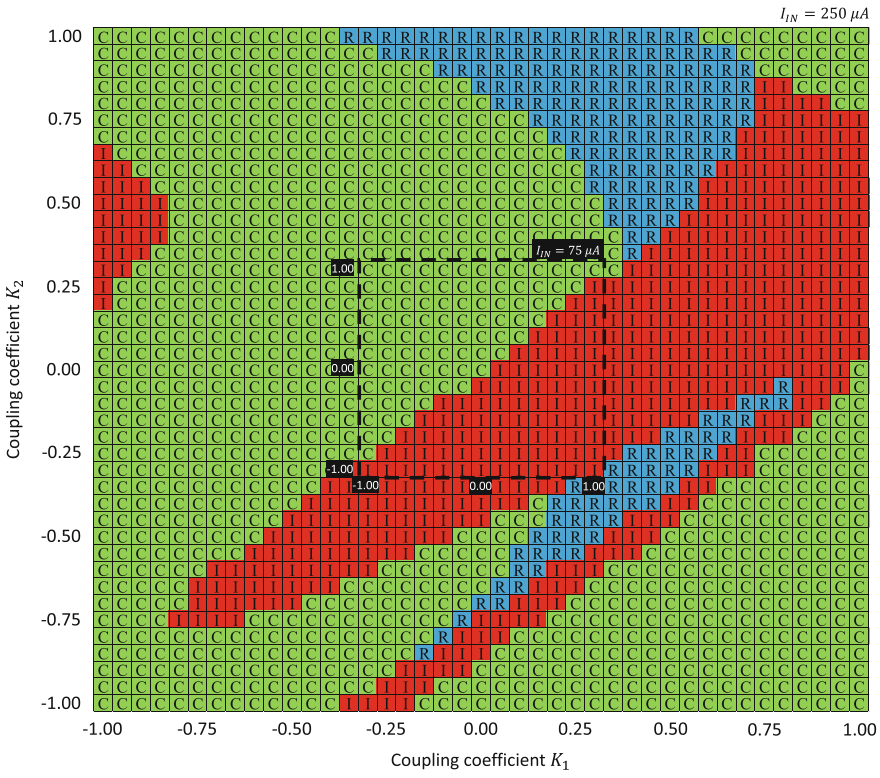

Fig. 14 Characterization of the locked OR cell when L_1 and L_2 are coupled, respectively, at input current values $I_{in} = 250\,\mu A$ and $75\,\mu A$. The "C" region corresponds to correct operation, the "I" region corresponds to incorrect operation, and the "R" region corresponds to incorrect operation which can be corrected by resetting the cell with specific inputs

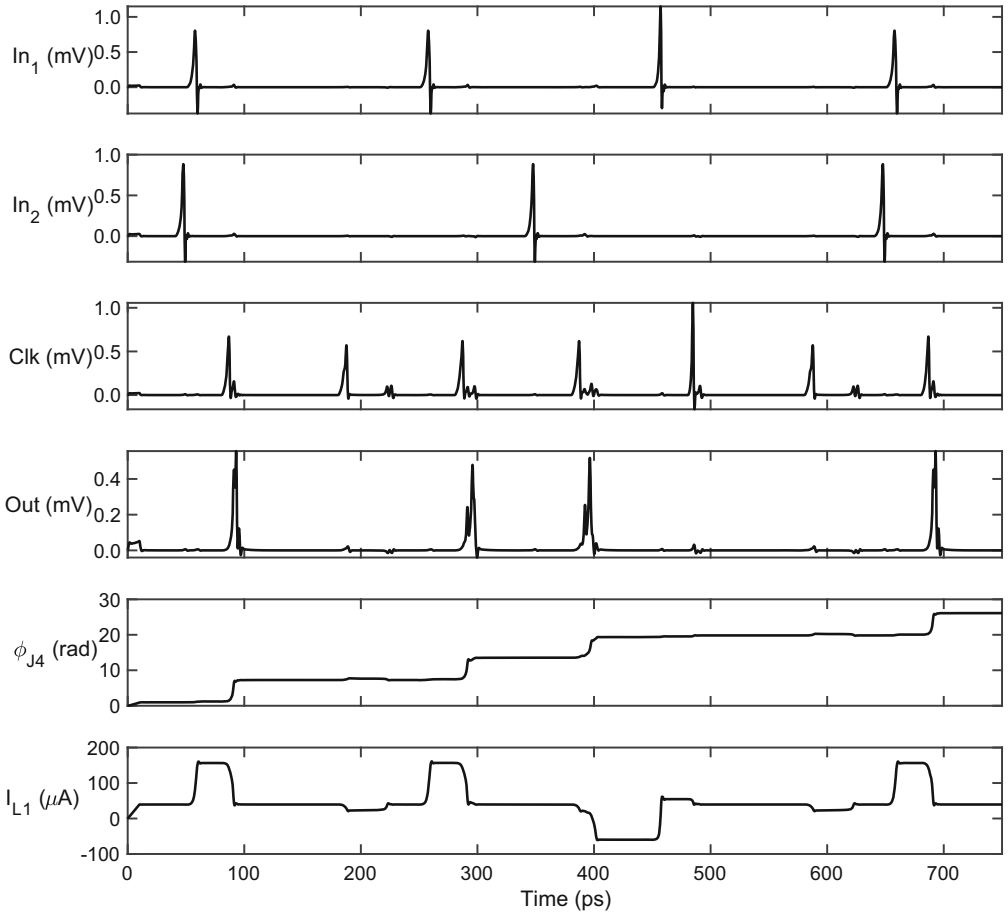

Fig. 15 Operation of the locked OR cell when $K_1 = K_2 = 0.5$ and $I_{in} = 250\,\mu A$

incorrect operation which can be transformed into correct operation as a result of the proposed attack.

For certain coupling coefficients, the locked OR cell produces two output pulses or slightly delayed output pulses with a clock pulse. Both of these cases are classified as logic "1" since, from a system-wide perspective, these cases do not produce errors.

Note that characterizing a locked OR cell at lower input currents is not necessary. Decreasing the input current in (2) exhibits the same effect as decreasing the coupling coefficient within a locked OR cell. For smaller input currents, the pattern remains the same. The key range, however, is constrained to a smaller region (see Fig. 14). For $I_{in} = 75\ \mu$A, the OR cell can be characterized within the dashed area at the center of Fig. 14.

For key values $K_1 = K_2 = 0.5$ and $I_{in} = 250\ \mu$A, a locked OR cell exhibits incorrect operation. Operation of the OR cell is shown in Fig. 15. In particular, the error occurs at approximately 500 ps, where the output is intended to be logic "1." The cell operates incorrectly due to a 2π phase shift of J_4 at around 400 ps. The 2π

phase shift occurs with the input combination of $In_1 = 0$ and $In_2 = 1$. Negative current therefore flows through inductor L_1. During the following clock pulse, the input combination of $In_1 = 1$ and $In_2 = 0$ restores the current through L_1 without producing a pulse at the output. Note that these explanations are common to all of the regions designated as "R" in Fig. 14. The input combination of $In_1 = 0$ and $In_2 = 1$ therefore destabilizes the current in the storage loop, making the output of the next clock cycle incorrect. The opposite input combination (i.e., $In_1 = 1$ and $In_2 = 0$) removes this effect and resets the OR cell back into normal operation. It is therefore possible to transform the "R" region into a region of correct operation by increasing the allowable space for the correct keys. The attack scenario is explained in the following subsection.

6.1.2 Attack Scenario

The proposed attack aims to *reset* the OR cell by applying the input combination $In_1 = 1$ and $In_2 = 0$ during each clock pulse. The "R" region depicted in Fig. 14 can be converted into the "C" region (i.e., correct operation of the OR cell). The correct key space therefore increases. A security parameter M is defined to quantify the ratio of the area of the correct operation region to the incorrect operation regions at $I_{in_{max}}$,

$$M \equiv \frac{\text{Correct key space}}{\text{Incorrect key space}}. \tag{4}$$

In the original locked OR cell [21], M is 1.59. By resetting the OR cell, the security parameter is increased to 2.69, corresponding to an 18.8% increase in the correct key space. The disadvantage of using this attack is lower throughput. For every input, a corresponding reset signal needs to be sent. The overall throughput therefore decreases by a factor of two.

In CMOS logic locking, threat models such as SAT attacks aim to decrease the incorrect key space, making a brute force search more efficient [47]. The reset threat model aims to increase the correct key space. From the perspective of the ratio of correct and incorrect key spaces, both of these threat models have a similar effect. The reset threat model can therefore be used in conjunction with other attacks to boost overall efficiency.

Although this attack focuses on a locked OR cell, a similar attack strategy for other types of locked circuits such as AND and XOR can also be applied. In the first step, the internal operation of a cell is analyzed to determine the specific input combination(s) that reset the cell into normal operation.

6.1.3 Possible Countermeasures

Limiting the key space prevents and thwarts the reset attack model. Selected coupling coefficients outside of the reset region (i.e., the "R" region in Fig. 14) limits

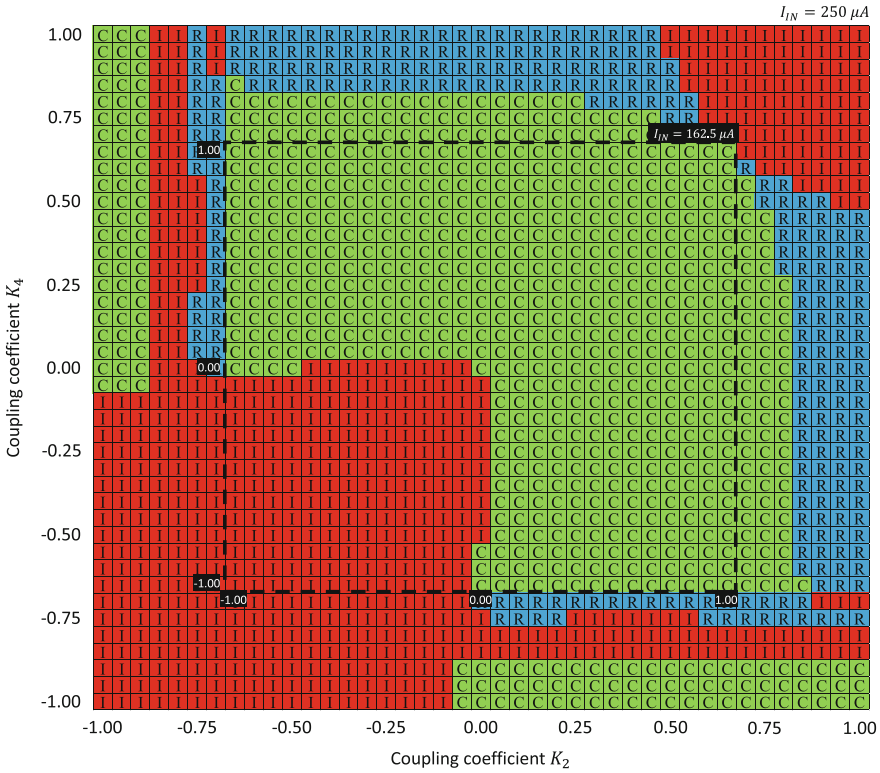

Fig. 16 Characterization of a locked OR cell when L_2 and L_4 are coupled at, respectively, input current $I_{in} = 250\ \mu A$ and 162.5 μA. The region notation is the same as shown in Fig. 14

the key space of the locked gate. However, choosing smaller coupling coefficients may not be an effective countermeasure due to the additional current flowing through the coupled inductor, as noted in (2). For example, in Fig. 14, the correct keys $K_1 = K_2 = 1$ at $I_{in} = 250\ \mu$A can be transformed into the "R" region shown in Fig. 14 by decreasing the input current by two times (i.e., $I_{in} = 125\ \mu$A), which is equivalent to $K_1 = K_2 = 0.5$ at $I_{in} = 250\ \mu$A.

Another solution is to set a limit on the input current, which can be achieved by changing the line width and coupled inductors. By limiting the input current, the potential region is smaller. For example, for $I_{in_{max}} = 75\ \mu$A, the reset region ("R" region) is eliminated see Fig. 14 (note the dashed rectangular area at the center of the figure). In this case, $M = 1.04$.

The key space of the input current is significantly reduced by the limited input current. A different configuration of coupled inductors is therefore used; L_2 and L_4 as coupled inductors in the locked OR gate. This locked OR cell is characterized as shown in Fig. 16, where the correct key space can be increased by 32.9% by resetting the cell. In this case, $I_{in_{max}} = 162.5\ \mu$A is sufficient to remove the possibility of resetting a cell, increasing parameter M to 2.88. A tradeoff therefore exists between M and $I_{in_{max}}$.

The bias margins of the SFQ OR cell are ±30% [48]. For the countermeasure of limiting the input current to be more robust against the reset attack model, the effect of process variations should be considered. The margins of a locked OR cell with coupled inductors L_1 and L_2 are evaluated. The margins of bias currents I_{b1}, I_{b2}, I_{b3}, and I_{b4} are, respectively, (−45%, +83%), (−36%, +80%), (−82%, +36%), and (−78%, +34%). The locked OR cell operates correctly under correct key values, and the reset attack model is not possible under any combination of coupling coefficients. The bias margins of the locked OR cell are therefore not degraded as compared to a standard SFQ cell. The margins of the coupled inductances, L_1, L_2, L_{M1}, and L_{M2}, are −68% and +61%, much larger than the inductance variations in the MIT Lincoln Laboratory SFQ5ee fabrication process [49–51]. A margin analysis therefore supports robust operation of the locked cell when applying logic locking in SFQ systems.

6.2 *Attack Model #2: Overproduction of Locked SFQ Circuits*

The overproduction of complex locked circuits can be enabled by characterizing a locked cell and by sweeping the key values in polynomial time (i.e., the number of steps is linear rather than exponential). The threat model and attack scenario for overproduction based attacks are presented, respectively, in Sects. 6.2.1 and 6.2.2. Related countermeasures to thwart the proposed attack are described in Sect. 6.2.3.

6.2.1 Threat Model

In this threat model, an attacker is assumed to be located at the foundry, where the masks and layout of the fabricated device are available. The objective of the attacker is to produce a greater number of devices than requested by the company that developed the original IC design. Additionally, the attacker is assumed to know the type of logic locking technique (in this case, mutual inductance coupling). Since the device layout is known, correct operation of the circuit can be predicted with reverse engineering. In the case where the layout is protected from reverse engineering by camouflaged gates [22], a more powerful attack can be assumed given access to the simulation files.

In the overproduction threat model, the coupling coefficients are fixed to a specific (correct) value since the layout has previously been sent to the foundry, and no further modifications are possible once the device is fabricated. Although the coupling coefficients are fixed, the actual coefficients are difficult for an attacker to determine. The only key that remains under control is the input current. Due to pinout limitations in SFQ circuits, only one input current for the entire device is assumed. As a result, to successfully overproduce the device, the attacker should determine the correct range of input current.

A possible method to setup an attack is to apply a particular input combination, sweep the input current from 0 to $I_{in_{max}}$, and monitor whether the final output is correct. Although the attacker has access to the physical device, intermediate signals within the device cannot be monitored (i.e., only the output signals can be monitored). However, due to false positive outputs, the operation can be mistakenly classified as correct if a certain input combination is not applied to the locked cell. In the example of a locked OR cell, the key values within the reset region (the "R" region), shown in Figs. 14 and 16, can be considered as correct if the combination $In_1 = 0$ and $In_2 = 1$ is not evaluated. The attacker needs to check all of the input combinations for all of the input current values to ensure that the device operates correctly in all cases. For k number of inputs,

$$\sum_{i=1}^{2^k} \frac{I_{in_i}}{\Delta I_{in}} \tag{5}$$

measurements are required, where I_{in_i} is the range of input current that is swept during the ith iteration, and ΔI_{in} is the step size. (5) increases exponentially with a greater number of inputs (which is impractical in complex circuit designs). For example, assuming that $I_{in_i} = I_{in_{max}} = 250\ \mu A$ (i.e., worst case scenario) and $\Delta I_{in} = 25\ \mu A$, for $k = \{1, 2, 3\}$ number of inputs $\{20, 40, 80\}$, measurements are required, which correspond to a 200% increase with one additional input signal.

6.2.2 Attack Scenario

In SFQ systems, the dependence of the current state of the output on a previous logic state is treated as a memory effect [52]. Alternatively, the operation performed during a clock cycle affects the operation during the subsequent clock cycle. Since the locked OR cell exhibits the memory effect with incorrect key values, an attacker can potentially apply an input sequence rather than just a single set of inputs to exploit this memory effect. An attacker needs to characterize a locked cell by setting $I_{in} = I_{in_{max}}$ and sweeping the coupling coefficients K_i from -1 to 1 with certain input sequences (similar to Fig. 14 without the reset region). This process can be described at the simulation level by either inferring the circuit configuration and parameters by reverse engineering the layout or directly accessing the simulation files (without the key values). The objective is to determine an input sequence that could reveal all possible incorrect key values of a locked OR cell. The input sequence should consist of all possible input combinations (e.g., for the 2 bit input, the combinations are {0,0}, {0,1}, {1,0}, and {1,1}) to trigger all of the internal states. Additionally, certain input combinations should be inserted at least two times to ensure the preceding input combination is different. This process is performed to trigger different variations of the memory effect. By evaluating different input combinations, an attacker can determine a particular input sequence to generate the same characterization plot as shown in Fig. 14. Note that this original

characterization plot is generated with multiple input sequences rather than a single input sequence. For the locked OR cell with coupled L_1 and L_2 (see Fig. 11), the input sequence of

$$\{In_1, In_2\} = \{1, 1\} \rightarrow \{0, 0\} \rightarrow \{1, 0\} \rightarrow \{0, 1\} \rightarrow \{1, 0\} \rightarrow \{0, 0\} \tag{6}$$

satisfies this condition. This input sequence is also shown in Fig. 15. Note that an overproduction attack is not only limited to this input sequence. The sequence in (6) is arbitrarily chosen. Other input sequences can also be used. The overproduction attack is also applicable for other locked cells (e.g., AND and XOR). Identifying the input sequence(s) that reveal(s) the incorrect operation of the circuit is required.

Once the input sequences are determined, those sequences should be applied to all locked cells within the device. By applying the input sequences for the first locked cell and monitoring the output, the range of input current can be determined that enable correct operation. The input current therefore converges to a certain range that corresponds to the correct key space. For a particular connection of cells, the locked cell may not receive a certain input combination. In this case, the attacker should characterize the cell under a limited range of input combinations and proceed with the attack. For example, if the locked OR cell (shown in Fig. 11) cannot receive the input combination $In_1 = 0$ and $In_2 = 1$, the "R" region shown in a new characterization should be in the "C" region shown in Fig. 14.

For the number of locked cells N in a circuit,

$$\sum_{i=1}^{N} \frac{I_{in_i}}{\Delta I_{in}} \tag{7}$$

measurements are needed with the proposed overproduction attack. Equation (7) increases linearly with the number of locked cells and is independent of the number of inputs.

As a case study, a 4-to-2 priority encoder is treated as a circuit under attack. This priority encoder converts multiple input bits into a smaller number of output bits. The truth table of a 4-to-2 priority encoder is listed in Table 4 where V stands for the valid bit and X represents the don't care states. A 4-to-2 encoder is often used in interrupt controllers within processors to provide high priority interrupt requests [53]. This circuit also includes a considerable number of OR gates, which can be locked. The 4-to-2 priority encoder is therefore a useful topology to evaluate proposed attacks.

A 4-to-2 priority encoder is converted into SFQ by inserting delay elements, splitters, and logic gates [54]. The circuit is shown in Fig. 17. The coupling coefficients for four locked OR cells are specified in this figure. Note that the coupling coefficients of the locked OR cells are unknown to the attacker. Correct operation of this circuit is verified at $I_{in} = 250\,\mu\text{A}$, as shown in Fig. 18. The output signals are available four clock cycles after the inputs are applied.

Table 4 Truth table of a 4-to-2 priority encoder

In_3	In_2	In_1	In_0	Out_1	Out_0	V
0	0	0	0	X	X	0
0	0	0	1	0	0	1
0	0	1	X	0	1	1
0	1	X	X	1	0	1
1	X	X	X	1	1	1

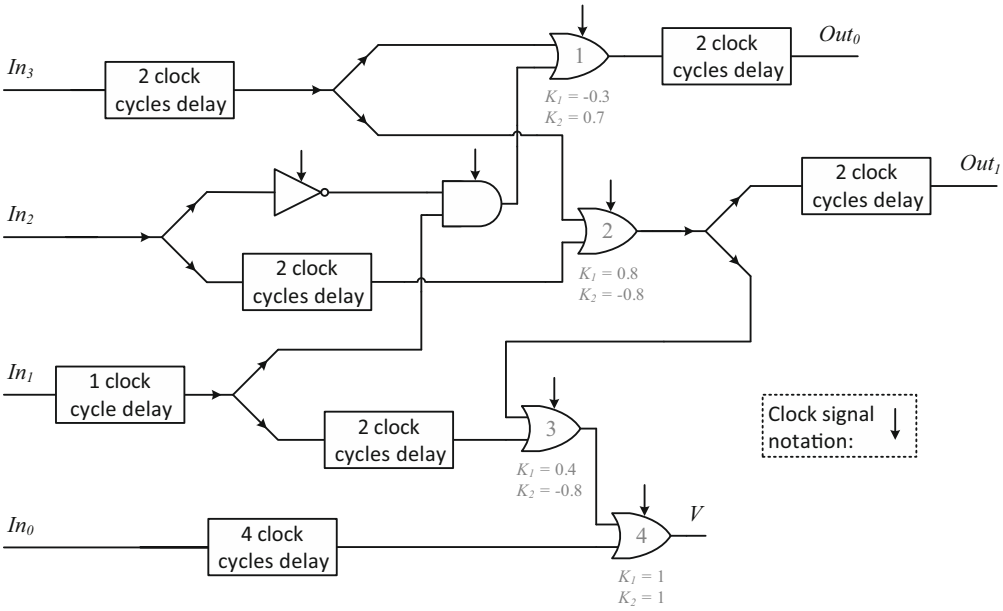

Fig. 17 4-to-2 priority encoder in SFQ

The maximum input current is assumed to be 250 μA ($I_{in_{max}} = 250\,\mu A$). An overproduction attack is realized in four steps, where the number of steps is the same as the number of locked OR cells, see (7). The applied input sequences and expected (correct) output sequences are listed in Table 5. Each step corresponds to unlocking one of the OR cells labeled in Fig. 17. In Table 5, the input sequences are generated to produce (6) for each locked OR cell. The correct output sequence therefore always becomes an OR version of (6).

By generating the input sequences listed in Table 5 and monitoring the output signals, the attacker should record the range of input currents that produces correct operation. The results for each step are depicted in Fig. 19. To reduce the computational time, the range of correct input currents determined in one of the steps should be within a range of input currents in the following step (see Table 5).

In the last step, as shown in Fig. 19, the range of correct I_{in} is determined to be between 225 and 250 μA. In this range, the 4-to-2 priority encoder operates correctly under any input combinations and sequences. An overproduction attack is therefore successful on logic locked SFQ circuits.

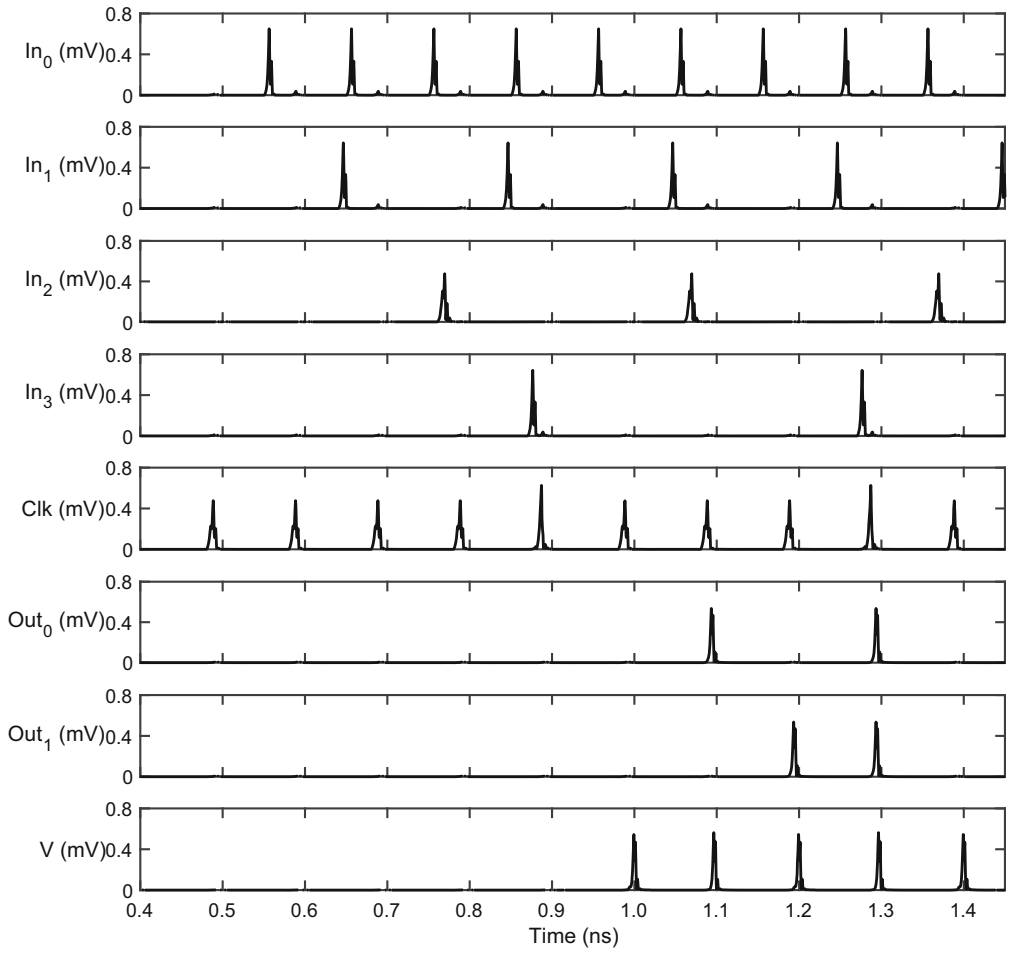

Fig. 18 Operation of SFQ 4-to-2 priority encoder with $I_{in} = 250\,\mu A$

6.2.3 Possible Countermeasures

A primary countermeasure against the overproduction threat model is to restrict the available knowledge of the attacker. In particular, by using camouflaged cells, deducing the correct operation of a device by reverse engineering the layout becomes significantly more difficult even for an experienced attacker. Since the correct operation is unknown in a camouflaged SFQ cell, the attacker cannot generate the desired input sequence for the locked cells. For an attacker that has access to the layout and simulation files except for the keys, a possible countermeasure is to increase the number of input current sources; two current sources, I_{in_1} and I_{in_2}, can be used rather than a single input current. The area overhead is, however, greater if an additional input current source is used to further increase the security of the circuit. A tradeoff therefore exists among the security of the system, area overhead, and number of input pins.

Table 5 Input and output sequences for each step in an overproduction attack

Step number	Input sequences	Correct output sequences	Range of input current sweep
1	$In_1 = \{1, 0, 0, 1, 0, 0\}$, $In_2 = \{0, 0, 1, 0, 0, 1\}$, $In_3 = \{1, 0, 1, 0, 1, 0\}$	$Out_0 = \{1, 0, 1, 1, 1, 0\}$	0 to 250 μA
2	$In_2 = \{1, 0, 0, 1, 0, 0\}$, $In_3 = \{1, 0, 1, 0, 1, 0\}$	$Out_1 = \{1, 0, 1, 1, 1, 0\}$	0 to 250 μA
3	$In_0 = \{0, 0, 0, 0, 0, 0\}$, $In_1 = \{1, 0, 0, 1, 0, 0\}$, $In_2 = \{0, 0, 0, 0, 0, 0\}$, $In_3 = \{1, 0, 1, 0, 1, 0\}$	$V = \{1, 0, 1, 1, 1, 0\}$	180 to 250 μA
4	$In_0 = \{1, 0, 0, 1, 0, 0\}$, $In_1 = \{1, 0, 1, 0, 1, 0\}$, $In_2 = \{0, 0, 0, 0, 0, 0\}$, $In_3 = \{0, 0, 0, 0, 0, 0\}$	$V = \{1, 0, 1, 1, 1, 0\}$	210 to 250 μA

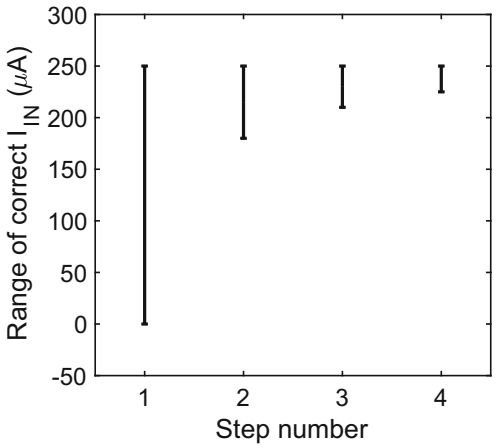

Fig. 19 Range of input current (I_{in}) resulting in correct operation of a 4-to-2 priority encoder

7 Conclusions

An important application of SFQ systems –large scale data centers operating with sensitive information– emphasizes the importance of hardware security in SFQ systems. Hardware security approaches for SFQ circuits –SFQ camouflaging and logic locking– are proposed herein. IC camouflaging and logic locking are well known techniques widely used to secure CMOS circuits. Although these techniques can be applied to SFQ circuits without modifications, standard approaches require a significant number of gates and additional input pins. IC camouflaging in SFQ circuits obstructs the reverse engineering process by inserting dummy JJs and camouflaged cells into a layout. A dummy JJ exhibits an identical top view image of a layout as a standard JJ. The rest of the layout and synthesis process remains unchanged. A large camouflaged SFQ circuit consists of camouflaged and regular gates with indistinguishable layouts. A novel method to provide a secret key for

logic locking is also proposed. Standard SFQ gates can be modified to depend on a secret key current to maintain correct functionality. Mutual inductors can also be used to couple an additional positive or negative current to the locked gate from the key current. The efficacy of the proposed techniques is characterized by the number of camouflaged and logic locked gates. The area and power overhead of IC camouflaging and logic locking techniques are characterized with ISCAS'85 benchmark circuits. Tradeoffs among security, area, and power for these different approaches are evaluated. IC camouflaging increases the effort necessary for hardware-based reverse engineering attacks. Logic locking prevents an external attacker from analyzing the structural behavior of a design even if a copy of the secured circuit is obtained. Two new attacks models on logic locking, reset and overproduction, are evaluated. A 4-to-2 priority encoder is characterized to evaluate different attacks on logic locked circuits. Hardware security for superconductive computing systems can provide robust and trustworthy VLSI complexity SFQ circuits.

References

1. J. Bardeen, L.N. Cooper, R. Schrieffer, Theory of superconductivity. Phys. Rev. **108**(5), 1175–1204 (1957)
2. B.D. Josephson, Possible new effects in superconductive tunneling. Phys. Lett. **1**(7), 251–253 (1962)
3. T. Orlando, K. Delin, *Foundations of Applied Superconductivity* (Addison-Wesley, Boston, 1991)
4. T.V. Duzer, C.W. Turner, *Principles of Superconductive Devices and Circuits*, 2nd edn. (Pearson, London, 1981)
5. G. Krylov, E.G. Friedman, *Single Flux Quantum Integrated Circuit Design* (Springer, Berlin, 2022)
6. K.K. Likharev, V.K. Semenov, RSFQ logic/memory family: a new Josephson-Junction technology for sub-terahertz-clock-frequency digital systems. IEEE Trans. Appl. Supercond. **1**(1), 3–28 (1991)
7. H. Suhl, B.T. Matthias, L.R. Walker, Bardeen-cooper-schrieffer theory of superconductivity in the case of overlapping bands. Phys. Rev. Lett. **3**(12), 552–554 (1959)
8. T. Jabbari, G. Krylov, J. Kawa, E.G. Friedman, Splitter trees in single flux quantum circuits. IEEE Trans. Appl. Supercond. **31**(5) (2021)
9. T. Jabbari, J. Kawa, E.G. Friedman, H-tree clock synthesis in RSFQ circuits, in *Proceedings of the IEEE Baltic Electronics Conference* (2020)
10. T. Jabbari, E.G. Friedman, Global interconnects in VLSI complexity single flux quantum systems, in *Proceedings of the Workshop on System-Level Interconnect: Problems and Pathfinding Workshop* (2020), pp. 1–7
11. K. Gaj, Q.P. Herr, V. Adler, A. Krasniewski, E.G. Friedman, M. J. Feldman, Tools for the computer-aided design of multigigahertz superconducting digital circuits. IEEE Trans. Appl. Supercond. **9**(1), 18–38 (1999)
12. C.J. Fourie, Digital superconducting electronics design tools – status and roadmap. IEEE Trans. Appl. Supercond. **28**(5), 1–12 (2018)
13. G. Krylov, J. Kawa, E.G. Friedman, Design automation of superconductive digital circuits a review. IEEE Nanotechnol. Mag. **15**(6), 54–67 (2021)
14. V.K. Semenov, Y.A. Polyakov, S.K. Tolpygo, New AC-powered SFQ digital circuits. IEEE Trans. Appl. Supercond. **25**(3), 1–7 (2015)

15. T. Jabbari, G. Krylov, S. Whiteley, E. Mlinar, J. Kawa, E.G. Friedman, Interconnect routing for large-scale RSFQ circuits. IEEE Trans. Appl. Supercond. **29**(5) (2019)
16. T. Jabbari, E.G. Friedman, Flux mitigation in wide superconductive striplines. IEEE Trans. Appl. Supercond. **32**(3), 1–6 (2022)
17. T. Jabbari, G. Krylov, S. Whiteley, J. Kawa, E.G. Friedman, Repeater insertion in SFQ interconnect. IEEE Trans. Appl. Supercond. **30**(8) (2020)
18. T. Jabbari, E.G. Friedman, Surface inductance of superconductive Striplines. IEEE Transactions on Circuits and Systems II: Express Briefs **69**(6), 2952–2956 (2022)
19. T.V. Filippova, A. Sahua, A.F. Kirichenkoa, I.V. Vernika, M. Dorojevetsb, C.L. Ayalab, O.A. Mukhanov, 20 GHz operation of an asynchronous wave-pipelined RSFQ arithmetic-logic unit. Phys. Procedia **36**, 59–65 (2012)
20. J.Y. Kim, J.H. Kang, High frequency operation of a rapid single flux quantum arithmetic and logic unit. J. Korean Phys. Soc. **48**(5), 1004–1007 (2006)
21. T. Jabbari, G. Krylov, E.G. Friedman, Logic locking in single flux quantum circuits. IEEE Trans. Appl. Supercond. **31**(5) (2021)
22. H. Kumar, T. Jabbari, G. Krylov, K. Basu, E.G. Friedman, R. Karri, Toward increasing the difficulty of reverse engineering of RSFQ circuits. IEEE Trans. Appl. Supercond. **30**(3), 1–13 (2020)
23. J. Hurtarte, E. Wolsheimer, L. Tafoya, *Understanding Fabless IC Technology*. (Elsevier, Amsterdam, 2007), pp. 25–32
24. M.M. Tehranipoor, U. Guin, D. Forte, *Counterfeit Integrated Circuits*. (Springer, Berlin, 2015), pp. 15–36
25. T. Huffmire, B. Brotherton, T. Sherwood, R. Kastner, T. Levin, T.D. Nguyen, C. Irvine, Managing security in FPGA-based embedded systems. IEEE Des. Test Comput. **25**(6) (2008)
26. J.A. Roy, F. Koushanfar, I.L. Markov, EPIC: ending piracy of integrated circuits, in *Proceedings of the IEEE/ACM Design, Automation and Test Conference in Europe* (2008), pp. 1069–1074
27. S. Köse, L. Wang, R.F. DeMara, On-chip sensor circle distribution technique for real-time hardware trojan detection, in *Government Microcircuit Applications and Critical Technology Conference* (2017), pp. 1–4
28. M. Yasin, J.J. Rajendran, O. Sinanoglu, R. Karri, On improving the security of logic locking. IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. **35**(9), 1411–1424 (2016)
29. J.A. Roy, F. Koushanfar, I.L. Markov, Ending piracy of integrated circuits. Computer **43**(10), 30–38 (2010)
30. P. Prinetto, G. Roascio, Hardware security, vulnerabilities, and attacks: a comprehensive taxonomy, in *Proceedings of the Italian Conference on Cybersecurity* (2010), pp. 177–189
31. W. Yu, O.A. Uzun, S. Köse, Leveraging on-chip voltage regulators as a countermeasure against side-channel attacks, in *Proceedings of the Design Automation Conference* (2015), pp. 1–6
32. W. Yu, S. Köse, False key-controlled aggressive voltage scaling: a countermeasure against LPA attacks. IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. **36**(12), 2149–2153 (2017)
33. S. Seçkiner, S. Köse, Preprocessing of the physical leakage information to combine side-channel distinguishers. IEEE Trans. Very Large Scale Integr. Syst. **29**(12), 2052–2063 (2021)
34. A.W. Khan, T. Wanchoo, G. Mumcu, S. Kose, Implications of distributed on-chip power delivery on EM side-channel attacks, in *Proceedings of the IEEE International Conference on Computer Design* (2017), pp. 329–336
35. R. Torrance, D. James, The state-of-the-art in semiconductor reverse engineering, in *Proceedings of the ACM/EDAC/IEEE Design Automation Conference* (2011), pp. 333–338
36. J. Kunert, O. Brandel, S. Linzen, O. Wetzstein, H. Toepfer, T. Ortlepp, H. Meyer, Recent developments in superconductor digital electronics technology at FLUXONICS foundry. IEEE Trans. Appl. Supercond. **23**(5), 1,101,707–1,101,707 (2013)
37. J.F. Annett, *Superconductivity, Superfluids and Condensates* (Oxford University Press, Oxford, 2004)

38. F. Frost, R. Fechner, B. Ziberi, J. Völlner, D. Flamm, A. Schindler, Large area smoothing of surfaces by ion bombardment: fundamentals and applications. J. Phys. Condens. Matter **21**(22), 224026 (2009)
39. T. Kanayama, H. Tanoue, T. Tsurushima, Niobium silicide formation induced by Ar-ion bombardment. Appl. Phys. Lett. **35**(3), 222–224 (1979)
40. HYPRES Design Rules HYPRES, Inc (2015). https://www.hypres.com/wp-content/uploads/2010/11/DesignRules-6.pdf
41. T. Thangam, G. Gayathri, T. Madhubala, A novel logic locking technique for hardware security, in *Proceedings of the IEEE International Conference on Electrical, Instrumentation and Communication Engineering* (2017), pp. 1–7
42. G. Krylov, E.G. Friedman, Design methodology for distributed large-scale ERSFQ bias networks. IEEE Trans. Very Large Scale Integr. Syst. **28**(11), 2438–2447 (2020)
43. F. Brglez, H. Fujiwara, A neutral netlist of 10 combinational benchmark circuits, in *Proceedings of the IEEE International Symposium on Circuits and Systems* (1985), pp. 685–698
44. P. Subramanyan, S. Ray, S. Malik, Evaluating the security of logic encryption algorithms, in *Proceedings of the IEEE International Symposium on Hardware Oriented Security and Trust* (2015), pp. 137–143
45. Y. Mustafa, T. Jabbari, S. Köse, Emerging attacks on logic locking in SFQ circuits and related countermeasures. IEEE Trans. Appl. Supercond. **32**(3), 1–8 (2022)
46. F.A.P. Petitcolas, *Kerckhoffs' Principle* (Springer US, New York City, 2011). https://doi.org/10.1007/978-1-4419-5906-5_487
47. R. Hofstede, M. Jonker, A. Sperotto, A. Pras, Flow-based web application brute-force attack and compromise detection. J. Netw. Syst. Manag. **25**, 735–758 (2017)
48. S.V. Polonsky, V.K. Semenov, P.I. Bunyk, A.F. Kirichenko, A.Y. Kidiyarova-Shevchenko, O.A. Mukhanov, P.N. Shevchenko, D.F. Schneider, D.Y. Zinoviev, K.K. Likharev, New RSFQ circuits (Josephson Junction digital devices). IEEE Trans. Appl. Supercond. **3**(1), 2566–2577 (1993)
49. S.K. Tolpygo, V. Bolkhovsky, T.J. Weir, C.J. Galbraith, L.M. Johnson, M.A. Gouker, V.K. Semenov, Inductance of circuit structures for MIT LL superconductor electronics fabrication process with 8 niobium layers. IEEE Trans. Appl. Supercond. **25**(3), 1–5 (2015)
50. S.K. Tolpygo, V. Bolkhovsky, T.J. Weir, A. Wynn, D.E. Oates, L.M. Johnson, M.A. Gouker, Advanced fabrication processes for superconducting very large-scale integrated circuits. IEEE Trans. Appl. Supercond. **26**(3), 1–10 (2016)
51. S.K. Tolpygo, E.B. Golden, T.J. Weir, V. Bolkhovsky, Inductance of superconductor integrated circuit features with sizes down to 120 nm. Supercond. Sci. Technol. **34**(8), 1–24 (2021)
52. T. Ortlepp, M.H. Volkmann, Y. Yamanashi, Memory effect in balanced Josephson comparators. Phys. C **500**, 20–24 (2014)
53. Intel, *8259A Programmable Interrupt Controller* (8259A/8259A-2). (Intel Corporation, Santa Clara, 1988)
54. M.D. Ciletti, M.M. Mano, *Digital Design* (Prentice-Hall, Hoboken, 2007)