Towards Provably-Secure Analog and Mixed-Signal Locking Against Overproduction

N. G. Jayasankaran, *Student Member, IEEE,* A. Sanabria Bórbon, *Student Member, IEEE,* E. Sánchez-Sinencio, *Life Fellow, IEEE,* J. Hu, *Fellow, IEEE,* and J. Rajendran, *Member, IEEE*

Abstract—Similar to digital circuits, analog and mixed-signal (AMS) circuits are also susceptible to supply-chain attacks, such as piracy, overproduction, and Trojan insertion. However, unlike digital circuits, the supply-chain security of AMS circuits is less explored. In this work, we propose to perform "logic-locking" on the digital section of the AMS circuits. The idea is to make the analog design intentionally suffer from the effects of process variations, which impede the operation of the circuit. Only on applying the correct key, the effect of process variations are mitigated, and the analog circuit performs as desired. To this end, we render certain components in the analog circuit configurable. We propose an analysis to dictate which components need to be configurable to maximize the effect of an incorrect key. We conduct our analysis on the bandpass filter (BPF), low-noise amplifier (LNA), and low-dropout voltage regulator (LDO) for both correct and incorrect keys to the locked optimizer. We also show experimental results for our technique on a BPF. We also analyze the effect of aging on our locking technique to ensure the reliability of the circuit with the correct key.

Index Terms—Analog security, process variations, SFLL, aging effect, logic locking, optimization

1 Introduction

1.1 Motivation

The increasing cost of manufacturing of integrated circuits (IC) has forced many companies to go fabless over the years. With the outsourcing of IC fabrication in a globalized/distributed design flow, including multiple (potentially untrusted) entities, the semiconductor industry is facing a number of challenging security threats. This fragility in the face of poor state-of-the-art intellectual property (IP) protection has resulted in hardware security vulnerabilities, such as IP piracy, overbuilding, reverse engineering, and hardware Trojans [1]. To address these issues most effectively at the hardware level [2], logic locking inserts additional logic into a circuit, locking the original design with a secret key. For a given input, a locked design produces correct output only upon applying the correct key; otherwise, an incorrect output is produced. In addition to the original inputs, a locked circuit has *key inputs*. An on-chip tamper-proof memory drives these *key inputs*. [3], [4]. In the case of digital designs, the additional logic may consist of XOR gates [5], [6] or look-up tables (LUTs) [7]. The locked netlist passes through the untrusted design phases. Without the secret key (i) the design details cannot be recovered by reverse-engineering the IC, and (ii) the over-produced IC gives incorrect outputs. A locked IC has to be activated by loading the secret key onto the chip's memory.

- *N.G. Jayasankaran, A. Sanabria Bórbon, E. Sánchez Sinencio, and J. Rajendran are with the Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX, 77843. E-mail: {gjn, adca.sanabria, s-sanchez, jv.rajendran}@tamu.edu*
- *J. Hu is with the Department of Electrical and Computer Engineering and the Department of Computer Science and Engineering, Texas A&M University, College Station, TX, 77843. E-mail: jianghu@tamu.edu*

While logic locking techniques exist for digital circuits, there is a great dearth of techniques for AMS IP protection. Moreover, analog ICs are not simple although they have less number of transistors. Even with only hundreds of transistors, analog IC design requires highly experienced designers and long time, as analog behaviors are quite complicated. Hence, it involves more capital in designing analog ICs [8]. Also, as explained in [9], analog ICs rank one in the top five counterfeited parts and cost several million dollars loss. Hence, we focused on developing a provable defense technique to secure AMS circuits. Hence, in this work, we focused on developing a provable defense technique to secure AMS circuits. Out of the different digital logic locking techniques available, the SFLL technique [10] provides provable security against SAT and removal attacks. Adding to this, it gives the freedom to the designer to choose the input patterns to protect. As this locking technique has provable security against different attacks and can successfully lock designs that have more than 100K gates, it is suitable to lock AMS design that consist of digital section (the optimizer circuit) with approximately 50K gates.

1.2 Problem statement

While logic locking schemes are well-defined for digital designs, there is no formal approach for analog designs. In this work, we develop a logic locking scheme for AMS designs. Here, only on applying the correct key, the locked AMS design produces the desired response. Otherwise, for an incorrect key, the response deviates from the desired value. For example, in the case of BPF, it exhibits the desired frequency response for the correct key and an incorrect frequency response for an incorrect key.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TETC.2020.3025561, IEEE Transactions on Emerging Topics in Computing

1.3 Prior work on analog locking

A locking technique using memristors is proposed in [11]. It uses a memristor-based voltage divider to bias the bulk terminal of transistors in a differential amplifier. Only the correct key can configure the voltage divider to provide the correct body-bias voltage. This scheme conceptually works well, but its practical applicability is quite restrictive due to its dependence on memristor, and hence, it does not apply to conventional CMOS-based AMS designs. The work [12], proposes a split manufacturing technique for RF circuits. This technique protects the circuit from an attacker in the foundry.

The work [13] demonstrates a satisfiability modulo theories (SMT)-based combinational locking. This defense mechanism ensures that each chip has a unique key. Hence, any attack to make the chip usable by finding the key is applicable only to that chip. Though this technique has increased the effort of the attack by using SMT-based combinational locking, one disadvantage is applying an incorrect key may sometimes produce close to the desired response. However, in our work, we ensure that the circuit suffers a deterministic error for an incorrect key. Another similar work [14] obfuscates the analog circuit performance using parameter-biasing obfuscation technique. Applying the correct key sets the required transistor width in the current mirror, which in turn provides the suitable bias current for the analog circuit operation.

Similar to our work, in [15], the locked digital circuit mandates the correct key input to set one or more specifications of the analog circuit correctly. This technique is demonstrated on a $\Sigma\Delta$ analog to digital converter (ADC). In [16], both the analog and the digital sections of the AMS circuits are locked. Here, the analog section is locked using the parameter-biasing obfuscation [14], and the digital section is locked using SFLL [10]. Though this technique increases the security by locking both analog and digital sections, it did not convey how to scale up for larger key sizes to thwart brute-force attack. In [17], only on providing the unique key inputs, the trained neural network generates the necessary bias to the analog circuit. This technique cannot protect the analog design against overproduction, cloning, and lock removal attacks. Our technique, however, protects the design against overproduction and lock removal attacks.

1.4 Attacks on analog locking

The work in [18] evaluates the resilience offered by the existing analog protection schemes [13], [14], and [11] using SMT. SMT is a decision problem similar to the Boolean satisfiability (SAT) problem. Unlike the SAT, which can only handle Boolean variables, SMT can handle non-Boolean variables. The bias current or voltage range and the locked analog netlist are the inputs to the SMT formulation. This formulation provides the correct key required to unlock the circuit. As the existing analog locks [11], [13], [14] are broken by [18], one needs to develop a new defense technique to protect the AMS circuits. Hence, in this work we develop one such technique.

1.5 Attacks on digital logic locking

Recent attacks such as SFLL-hd – Unlocked [19] and FALL attack [20] break SFLL-HD^0 and SFLL-HD^h. In combinational logic-locked circuits, the output is a Boolean variable for a given input-key combination. However, in analog circuits, the output is a non-Boolean variable, such as bias current, bias voltage, and frequency response. As these attacks [19], [20] can handle only Boolean variables, it cannot break analog logic locking. Removal attack [21] identifies the protection logic and removes them, thereby extracting the original functionality of the locked circuit. However, launching this attack removes the logic-locked optimizer, which sets the correct value of the passive components in the analog circuit-under-protection. Therefore, this attack does not apply to our proposed work.

1.6 Challenges in AMS locking

A simple and obvious approach to lock an AMS design is to insert extra transistors, controlled by key inputs in the analog circuit. These key-transistors can be inserted at random locations in the circuit. On applying the correct key, the analog circuit provides the correct output. However, such a simple approach suffers from the following issues:

- As this includes a minimal number of key-transistors, the attacker determines the correct key by brute-forcing.
- Analog circuits have a smaller number of devices (only a few hundreds). Hence it is relatively simple to reverse engineer than digital circuits, which have millions of transistors on a single chip.
- From the reverse-engineered netlist, the attacker can find the key-transistors by tracking the key inputs and remove them, thereby obtaining the original circuit [22].
- Unlike digital circuits, which have thousands of gates in the circuit-to-be-protected, analog circuits have a few hundreds of components. Thus, one needs to select the best set of components to lock so as to trade-off between overhead and corruptibility.

1.7 Proposed approach

Piracy vs. overproduction. In analog designs, most of the commonly-used circuits follow standard layout techniques, such as common-centroid and interdigitization, which makes it easy to reverse engineer [23]. Also, an attacker can always recreate a design from scratch, given the circuit specification; an attacker can obtain such information from the publicly-available datasheet. These challenges make it difficult to prevent piracy attacks, where the attacker can modify the existing design, produce the mask for the modified design, and manufacture new chips. Hence, we try to prevent overproduction, where the foundry uses the same masks and produces excess chips. Our technique renders the overproduced chips non-functional, even if the attacker has access to the complete specification of the target chip.

Our technique for protecting the analog circuit is to logic-lock the digital section of the AMS circuit, as illustrated in Fig. 1. This digital section minimizes the effect of process variation by choosing the correct value of passive components in the analog circuit via the tuning knob settings. Analog circuits are susceptible to process variations; for

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TETC.2020.3025561, IEEE Transactions on Emerging Topics in Computing

instance, a filter can suffer up to 20% of variation due to the component's tolerances [24]. Many approaches have been proposed to minimize the effect of process variations [25]. In one of these approaches, the passive components of the analog circuits, such as resistors and capacitors, are set to their optimal values using tuning knobs [26]. The digital components determine the optimal values for these tuning knobs. By performing *judicious* logic locking on the digital components of such circuits, only on applying the correct key, the effect of process variations are nullified as the digital circuit works correctly, and thus making the analog circuit to perform as desired. On applying an incorrect key, the digital circuit produces incorrect output, thereby setting the tuning knobs of the analog circuit to non-optimal values. This improper tuning deteriorates the performance of the analog circuit.

Our approach provides the following benefits:

1) By setting the default tuning knobs where the harmful process variation effect is high, even if the attacker removes the locked digital circuit, the resultant analog circuit has degraded functionality. This degradation is due to the presence of the harmful effect of process variations.
2) The tuning knobs are selected such that even a small amount of change in their values significantly impact the behavior of analog circuits.
3) Since we cannot protect all the input patterns of the digital circuits, we protect only those input patterns that significantly impact the values of the tuning knobs, thereby the output of the analog circuit.
4) Furthermore, we judiciously perform all these steps to minimize area, power, and delay overheads.

1.8 Contributions

The paper has the following contributions:

- The first technique that can protect AMS designs against overproduction using digital logic locking techniques, including the attacks demonstrated in [19], [20], [21], [27].
- A sensitivity analysis that can maximize the impact of protection, thereby reducing the overhead. The number of tuning knobs is increased using the same analysis for a higher deterministic error experienced by the attacker for an incorrect key.
- The AMS lock technique proposed in this work applies to a wide variety of analog circuits. It is demonstrated on three different circuits: BPF, LNA, and LDO, including experimental results from a BPF chip.
- The effect of aging on the locked AMS circuits is analyzed. The simulation result proves that the locked circuits are reliable even if the transistors, age over time.
- The effect of SFLL-flex lock on the AMS circuits is analyzed. Results prove that we could achieve higher security with lesser area using SFLL-flex compared to SFLL-HD^0 or SFLL-HD^h.

The paper is organized as follows. In Section 2, we explain the background and previous work related to logic locking and process variations impact on AMS circuits. In Section 3, we explain the locking strategy with the BPF circuit as a motivating example. Section 4 shows the experimental and simulation results of the proposed technique. Section 5 concludes the paper.

2 Background

2.1 Logic locking

In this work, we lock the digital section of the AMS circuit using stripped-functionality logic locking (SFLL) [10]. The functionality-stripped circuit (FSC) replaces the original circuit-to-be-protected. The FSC is generated by inserting or replacing a few of the logic gates in the original circuit. The FSC's output is corrupted for those input patterns which are protected by the defender. These patterns are called protected input patterns (PIPs). The output is inverted for the PIP corresponding to the correct key. The restore unit then inverts the inverted output only for the correct key, thereby restoring the correct output. For an incorrect key, SFLL produces an inverted output for the PIP. Both the key and the protected input patterns are the designer's secrets.

There are three variants of SFLL, namely, SFLL-HD^0, SFLL-HD^h, and SFLL-flex [10]. Depending on the variant of SFLL, the restore unit implements one of the techniques given below. The corruption injected by the inversion logic is restored, when

1) the Hamming distance (HD) between the external key (k) and the input pattern equals 0 in SFLL-HD^0.
2) the HD between k and the input pattern equals h in SFLL-HD^h.
3) the PIPs or "cubes" are stored in a content-addressable memory along with their corresponding flip vectors, as illustrated in Fig. 2. The flip vector associated with each protected cube holds information regarding which outputs are to be flipped (restored) for that cube. When the input to the FSC is equal to one of the protected cubes in the content-addressable memory, the corresponding flip-vector is retrieved and XORed with the outputs to restore the original functionality.

The choice of the input patterns to protect is not restricted by the value of HD in SFLL-flex. Hence, the defender can protect any IP-critical input patterns. In the con-

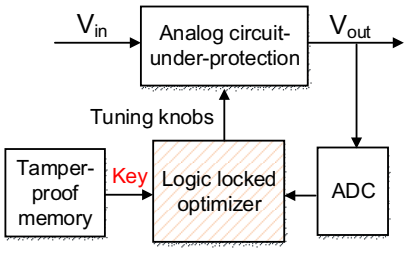


Fig. 1: Logic locking of the AMS circuit.

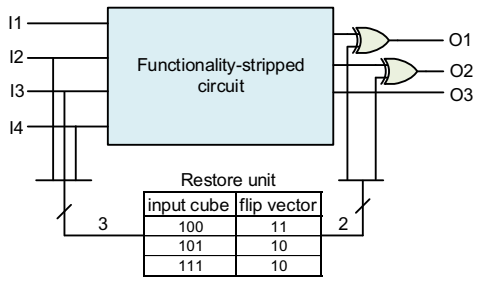


Fig. 2: Stripped-functionality locking locking (SFLL)-flex [10].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TETC.2020.3025561, IEEE Transactions on Emerging Topics in Computing

text of this research, one needs to select what input patterns of the optimizer, that compensate the process variations effects. This way, an incorrect key does not eliminate the effect of process variations.

2.2 Analog ICs and process variations

The performance of analog circuits is degraded in the presence of process variations. During the design phase, the sizes of transistors and passive components are chosen to meet the required specifications. Being aware of the process variations, the designer performs Monte Carlo and/or corner simulations to tune these sizes to improve the robustness of the design. However, due to the limitations in the fabrication process, it is not possible to fabricate the precise sizes of the transistors and passive components. Adding to this, the intra-die process variations in the manufactured chip have an impact on the circuit's performance. Hence, to nullify the effect of process variations, researchers have proposed different techniques, such as body voltage tuning and built-in self-test (BIST) optimization techniques. Body-bias voltage tuning is an efficient way to address process variation in terms of power, performance, and area [28].

Likewise, in the BIST technique [26] at start-up, the tuning knobs are in their default settings. The input voltage is applied to the analog circuit, and the corresponding output response is digitized by ADC. This digitized data is sent to the optimizer. The optimizer calculates the deviation in the actual output response from the ideal characteristics. If the magnitude of this difference is high, it indicates that the circuit's response is far from the ideal characteristics and, if low, indicates it is closer to the ideal characteristics. The optimizer chooses a different tuning knob setting such that the cost function calculated is lesser than the cost function value corresponding to the previous settings. This process is iterated until the output response for the chosen tuning knob gives zero cost function value. This tuning process helps in compensating the process variation impact on the fabricated components. The optimizer uses the simulated annealing algorithm to determine the tuning knob settings. The performance of this tuning depends on various factors, such as temperature step size, the maximum number of iterations, and initial temperature.

The body-bias tuning can compensate for process variations only in the transistors but not other passive components, such as resistors and capacitors. Therefore, they cannot be deployed to analog circuits that require tuning of their passive components. Also, as there is no secure analog locking scheme [18], that can lock the supply voltage used by the body-bias tuning technique. Hence, we chose the optimization technique [26] (i) to compensate for process variations in passive components and bias currents and (ii) the digital optimizer can be locked using a provably secure digital locking technique [10]. Our technique is power-, performance-, and area-efficient and can be used in wearable devices and IoT architectures. The wearable IoT ECG sensors [29] consists of AMS circuits showing the possibility of implementing our technique in wearable devices. Also, any IoT SoC has multiple analog modules such as audio units, radio units, sensors, and power management units [30]. Here, a single optimizer can be used to tune all the modules during power-up, justifying the area overhead of the optimization unit. Hence, to show the proof-of-concept of our technique, we illustrate an optimizer controlling one analog circuit. However, in practical scenarios, a single optimizer can control more number of analog blocks.

3 Locking Approach for AMS Circuit

3.1 Choosing analog circuits to demonstrate our locking approach

We have chosen BPF, LNA, and LDO circuits to protect as they are widely used in many domains, such as IoT, communication, and signal processing. The filter circuits, such as BPF, lowpass filter, and highpass filter, are essential in many communication and signal processing systems [31]. The LNA is highly prevalent in RF communication [32], [33], and the LDO voltage regulator is a common entity in the power management unit [34]. Also, the analog circuits considered in this work are used in the individual blocks of the IoT SoC architectures [30]. In this section, we first describe the BPF circuit, which we use as a motivating example to explain our idea. We then describe our locking architecture, and the methodology used in selecting the input patterns to protect. Finally, we explain how the proposed mechanism applies to other analog circuits, such as LNA and LDO.

3.2 Motivational example: Bandpass filter

Consider the Tow-Thomas filter illustrated in Fig. 3 with a transfer function defined by Equation (1).

$$H_{BP}(s) = \frac{s/(R_1C)}{s^2 + s/(R_1C) + 1/(R_2^2C^2)} \tag{1}$$

Assuming ideal amplifiers, i.e., amplifiers with infinite gain and bandwidth, we set $R_1 = R_3$ and $R_2 = R_4$. The filter characteristics, such as center frequency $f_o = 1/2\pi R_2 C$ and quality factor $Q = R_1/R_2$ are defined by the passive components in the circuit, i.e., resistors and capacitors. Each of the R (C) is replaced by arrays of Rs (Cs) to enable tuning that helps in addressing the impact of process variation. The value of R (C) is tuned to get the optimal circuit performance in the presence of process variation. Such tuning helps to compensate for any changes in resistor and capacitor values due to process variations.

The required tuning resolution controls the size of each passive component in the array. This resolution is defined as the minimum increase or decrease in the value of the passive component between two consecutive tuning knob settings. The resolution of the tuning is determined based on the performance of the optimization technique, the area

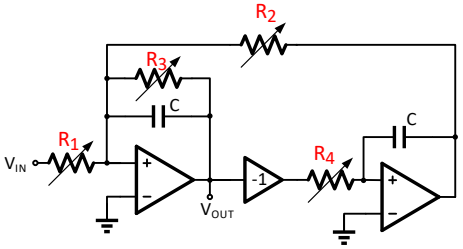


Fig. 3: Tow-Thomas BPF circuit. The resistors R_1, R_2, R_3, and R_4 are tunable resistors.

incurred by this technique, and the power consumption. For example, a very low resolution in tuning (corresponding to bigger component sizes) results in a sub-optimal operating point after optimization. However, for a high resolution in tuning (corresponding to smaller component sizes) results in an optimal operating point. Nevertheless, this incurs a large area and higher power consumption. Hence, there is a trade-off between the optimization quality, area, and power consumption in choosing the resolution. In BPF, the tuning knob controls the resistor values, which in turn controls the output response of the BPF. Each resistor in the array has a resistance of 865.38Ω.

The passive components, such as Rs and Cs, are made tunable to compensate for process variations. Hence, the number of these components required is controlled by the maximum variation in the circuit parameters. This variation in the circuit parameters is due to the impact of process variations. In BPF, the variations in the circuit parameters such as center frequency and bandwidth are estimated using Monte Carlo simulations. From these simulations, the defender determines the minimum and maximum value of the filter parameters. Using this range, he/she calculates the minimum and maximum values of the Rs and Cs required. This information gives the range of the value of the passive components that has to be implemented in the array.

3.3 Locking architecture

The use of SFLL techniques to lock the AMS circuits does not correspond to a plug-and-play concept. Rather, it involves multiple steps and analyses for successfully locking the digital optimizer, which controls the performance of the analog circuit. The locking architecture consists of the following steps given below:

1) **Choosing the tunable components using sensitivity analysis.** To ensure that the attacker suffers maximum degradation in the performance on applying an incorrect key, we perform a sensitivity analysis to determine those passive components on which the output response of the analog circuit is highly dependent.
2) **Replace the chosen component with an array of components.** To make the chosen component tunable, replace it with an array of components. The required value of the component is chosen using the tuning knobs.
3) **Determine all possible input patterns to the optimizer.** Simulate the analog circuit-under-protection for each tuning knob setting. The input and output values are determined at the frequency points of interest, depending on the circuit-under-protection.
4) **Determine the cost function corresponding to all the input patterns.** The defender calculates the value of the cost function for each possible input pattern. This value determines if the output response of the circuit-under-protection follows the ideal characteristics of the circuit.
5) **Choosing the input patterns to protect.** Select the minimum cost the attacker should encounter for an incorrect key. Choose all the input patterns that produce the cost equal to or below the selected cost. These patterns are the PIPs.
6) **Locking the optimizer.** Using the PIPs selected in the previous step, lock the optimizer using SFLL-HD^0, SFLL-HD^h, or SFLL-flex.

Hence, the analog circuit-under-protection, along with the locked optimizer, is the functionality-stripped AMS circuit. Only on applying the correct key, the original functionality of the circuit is recovered. The following sections explain the locking architecture in detail.

The AMS design in Fig. 1 consists of the BPF circuit-to-be-protected along with the ADC and the logic-locked optimizer. The voltage input and the two tuning knobs from the optimizer are the inputs to the BPF. Each tuning knob setting corresponds to a unique value of the resistor in the BPF circuit, which in turn impacts its frequency response. During the start-up, the tuning knobs are in their default settings. For the given input voltage, the output response of the BPF is digitized by ADC and sent to the logic-locked optimizer. The secret key required for the proper operation of the optimizer is loaded from a tamper-proof memory. The optimizer calculates the cost difference in the measured and the desired output response of the BPF. If the magnitude of this difference is high, it indicates that the BPF response has deviated from the desired response and if low, indicates it is more close to the desired response. The deviation in the output response from the desired response is calculated based on the following equation.

$$\begin{aligned} CF =&(G_{f_2} - (\sqrt{2} \times G_{f_1})) + (G_{f_2} - G_{f_3}) + \\ &(G_{f_3} - (\sqrt{2} \times G_{f_4})) + (G_{f_1} - G_{f_4}) \end{aligned} \tag{2}$$

Here, CF is the cost function, f_1 and f_4 are the lower and upper cut-off frequencies, and f_2 and f_3 are the center frequencies ($f_2 = f_3$). G_{f_1}, G_{f_2}, G_{f_3}, and G_{f_4} are gain at f_1, f_2, f_3, and f_4, respectively. The G_{f_2} should be equal to $\sqrt{2} \times G_{f_1}$ and the G_{f_3} should be equal to $\sqrt{2} \times G_{f_4}$ in the ideal characteristics. Likewise, G_{f_2} and G_{f_3} should be equal, and G_{f_1} and G_{f_4} should be equal in an ideal characteristic. The gain is calculated by the ratio of the output voltage to the input voltage. The transient analysis is performed at these four points. There is a total of 18, 10-bit data measured from the input signal and the output response of the BPF. It corresponds to the 180-bit data from the analog circuit. This data is concatenated with the 40-bit control input that configures the optimizer. This concatenated data constitutes the 220-bit data that is fed to the optimizer. As our optimizer is implemented using a combinational logic, we provide all the 18, 10-bit inputs driven by the ADC along with the 40-bit optimizer configuration input at the same time. These inputs together constitute the 220-bit input to the locked optimizer.

This simple architecture suffers from two challenges:

Issue 1: Not all resistors and capacitors in the BPF are made tunable. Making every component tunable increases the area overhead of the analog circuit. Also, the optimizer now has to tune all the components to address the process variations. Consequently, the area and delay overheads of the optimizer are increased. Thus, one needs to judiciously choose the parameters to tune.

Issue 2: The logic-locking techniques can protect only a limited number of input patterns. In SFLL-fault [35], the input patterns to be protected are chosen based on the VLSI testability metrics such as controllability and observability.

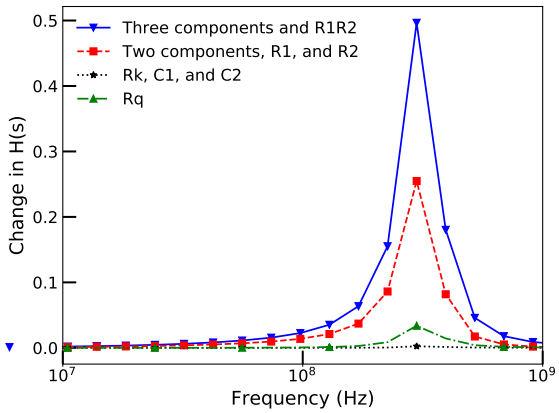


Fig. 4: Change in H(s) with respect to the change in component values.

This ensures that for an incorrect key, the maximum number of output bits are corrupted. However, in our work, we want to protect a specific set of input patterns, which sets the tuning knobs to optimal values. Hence, we use the SFLL-flex, where the user has the freedom to choose specific input patterns to protect. Also, as increasing the number of patterns protected increases the area overhead, we need to be diligent in choosing the patterns protected.

3.4 Sensitivity analysis to solve Issue 1

To ensure that the attacker suffers maximum degradation in the performance on applying an incorrect key, we perform a sensitivity analysis to determine those passive components on which the output response of the analog circuit is highly dependent. The sensitivity is a measure of the variation in a performance metric, such as f_o and Q, due to the change in certain circuit parameters [36]. The normalized sensitivity of the chosen metric p_i with respect to the change in parameter x_j is represented by Equation (3). This equation helps in determining the sensitivity of the circuit's response to each of the component considered.

$$S_{x_j}^{p_i} = \frac{x_j}{p_i}\frac{\partial p_i}{\partial x_j} \tag{3}$$

There is no difference in selecting the tuning knobs (tunable components) in secured and unsecured implementations. In both these implementations, sensitivity analysis is used to determine the components over which the output response of the circuit-under-protection is highly dependent. This choice ensures that for an incorrect key, the incorrect value of the parameters is chosen. Hence, there is maximum deviation in the output response. The following steps are carried out to perform the sensitivity analysis:

1) The sensitivity of the circuit's response with respect to the circuit parameters is plotted.
2) The component for which the circuit response has the highest sensitivity is chosen. An array of components replaces this component.
3) The optimal value of this component is chosen from the array by the tuning knobs controlled by the optimizer.

For example, in BPF, the center frequency and the quality factor are the circuit metrics on which the sensitivity analysis is performed. The selected tuning knobs are the resistors R_1 and R_2. The optimal value of the component varies from chip to chip due to process variations to achieve the same performance metric. Hence, the defender makes the single component tunable by replacing it with an array of components. Once the chip is manufactured, the component that has the optimal value is chosen, such that the process variation impact is compensated. The defender uses the tunable components only to compensate for the process variation and does not use it to have flexibility in the performance metrics.

The effect of an incorrect key on the locked circuit can be increased by tuning more than two passive components. Let the sensitivity of the metric p_i is calculated based on the chosen passive components. Let j be the total number of components considered for sensitivity analysis. The change in one of the passive components does not affect the other. Hence, the change in p_i, such as transfer function and transconductance, due to change in one of the passive components does not depend upon the change in p_i due to another passive component. Therefore, the total change in p_i due to the changes in more than one passive component is the sum of the partial derivative p_i with respect to the passive components considered, i.e, $dp_i = dp_i(x_1, x_2, \cdots, x_j) = \frac{\partial p_i}{\partial x_1}d_{x1} + \frac{\partial p_i}{\partial x_2}dx2 + \cdots + \frac{\partial p_i}{\partial x_j}dx_j$. Thus,

$$\frac{dp_i}{p_i} = S_{x_1}^{p_i}\left(\frac{dx_1}{x_1}\right) + S_{x_2}^{p_i}\left(\frac{dx_2}{x_2}\right) + \cdots + S_{x_j}^{p_i}\left(\frac{dx_j}{x_j}\right) \tag{4}$$

The BPF circuit considered in this example has six passive components, as illustrated in Fig. 3. The change in the output response of BPF, $H(s)$ with respect to the change in the passive components are measured using Equation 4. The change in $H(s)$ is plotted for the following cases: (i) all possible combinations of three passive components are changed, (ii) all possible combinations of two passive components changed, and (iii) one of the passive components is changed. Fig. 4 shows the change in $H(s)$ of the BPF due to the change in one or more components. As illustrated in the figure, change in $H(s)$ is highest when all possible combinations of three passive components change and also when one of the combinations of two passive components (R_1 and R_2) change.

3.5 Choosing input patterns to solve Issue 2

Based on the minimum deviation in the output response the attacker has to encounter for an incorrect key, the designer protects all those input patterns that correspond to a deviation less than the chosen deviation. This deviation is quantified by the error in the cost function. If the error is close to zero, the output response is close to the ideal characteristics. Otherwise, it deviates from the ideal characteristics. The input patterns to the locked-optimizer are the digitized voltage values of the output response of the analog circuit-under-protection via the ADC. The defender can simulate the analog circuit, for example, the BPF, for each tuning knob settings, to determine the output response for each of these settings. The in-phase and quadrature-phase values of the input and output voltages are measured via transient analysis at the lower cut-off, upper cut-off, and two center frequencies. The voltage values calculated via simulations may differ from the chip results due to the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TETC.2020.3025561, IEEE Transactions on Emerging Topics in Computing

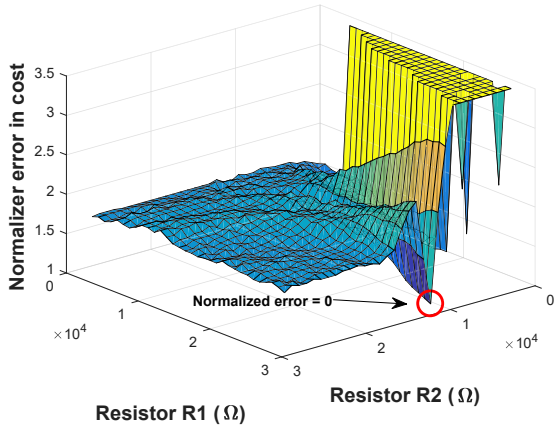


Fig. 5: Normalized error on the output of BPF for different tuning knobs. The normalized error at $(\mathbf{27.68K\Omega}, \mathbf{10.38K\Omega}) = \mathbf{0}$. The data is collected from a IBM-180nm process BPF chip described in Section 4.

PVT impact. These values can differ by a small percentage, thereby changing the LSBs of the measured voltages. Hence, the designer considers the MSBs of the voltage values in the locking, and the LSBs are ignored. Using these values, the designer can calculate the cost for each tuning knob setting. He/She will then choose those input patterns which corresponds to the least cost as protected input patterns.

In the case of SFLL-HD^0, since only one input pattern can be protected, a designer can obviously select the input pattern of the optimizer that results in the minimum cost function. For instance, in the case of BPF, we need to protect the input pattern corresponding to resistor settings $R_1 = 27.68K\Omega$ and $R_2 = 10.38K\Omega$. This setting ensures minimum cost, as illustrated in Fig. 5. In other words, this is the input pattern, for which the error between the desired response and the actual response of the BPF is minimum. In the case of SFLL-HD^h, a designer can increase the number of PIPs by increasing the value of h. However, this decreases the security level of an n-bit design to $2^{n-k} \times \binom{k}{h}$. Hence, one can increase the value of h only to an extent. Here, we select those input patterns such that they are at an $HD = h$ away from the one that produces the minimum cost function.

However, as SFLL-flex is devoid of the HD restriction, the defender can choose any input patterns he/she want to protect. Also, as the FALL attack [20] and SFLL-hd – Unlocked [19] have broken SFLL-HD0 and HDh without the use of an oracle, the optimizer is locked using SFLL-flex as it is resilient against these attacks. The following section explains the locking process using the SFLL-flex technique. The optimizer controls two tuning knobs. Each knob has a 5-bit input. They tune the resistors R_1 and R_2 of the BPF circuit. Though the optimizer's input size is 220 bits, as the total number of tuning knob settings equals 1024, the effective number of input patterns to consider reduces to 1024 from 2^{220}. The optimizer is designed to choose the tuning knob settings, which gives the minimum cost. Therefore, the defender can choose PIPs, whose corresponding costs are low.

3.6 Extending to other AMS circuits

This locking technique is illustrated over two other analog circuits–LNA and LDO.

3.6.1 LNA

A common-gate topology-based LNA was tested as a study case [33]. The specifications to optimize are the gain (S_{21}) and the input matching (S_{11}) for the given resonance frequency. Based on sensitivity analysis, the tuning knobs are determined to be the biasing current and the capacitance of the load tank. The minimum bias current is given by $10\mu A$, and the maximum bias current is $55\mu A$. The minimum increment in the bias current is $5\mu A$. The metrics S_{11} and S_{21} are estimated by applying two frequency tones at $f_R \pm \Delta_f$ and connecting the proper matching at the input and output. Then, the signal's amplitude is measured at the input and output of the LNA.

3.6.2 LDO voltage regulator

A capless LDO with a PMOS pass transistor and a single-stage error amplifier is tested [37]. The performance metrics to optimize are the power supply rejection (PSR) and the phase margin. The selected tuning knobs are the biasing current of the error amplifier and the compensation capacitor. The minimum bias current is given by $10\mu A$, and the maximum bias current is $60\mu A$. The minimum increment in the bias current is $10\mu A$.

3.6.3 Extending to large scale AMS circuits

Though the analog section of the example AMS circuits (BPF, LNA, and LDO) has only a few tens of transistors, the digital section (optimizer) consists of around 50K gates. These AMS circuits are generally a part of bigger analog circuits, such as receivers and phase-locked loops [38], [39]. Hence, for an incorrect key, the degraded performance of the AMS circuit has an impact on the overall performance of the receiver and phase-locked loops.

4 Results

4.1 Experimental setup

We demonstrate our analog locking technique on three different AMS circuits: BPF, LNA, and LDO. The specifications for each of these circuits are as follows. The BPF has a center frequency $f_c = 74MHz$ and $BW = 13MHz$. The input size of the optimizer is 220 bits. This input is fed with the digitized output (frequency response) of the BPF. The specifications of the LNA circuit are $S_{21} > 20dB$ and $S_{11} < -20dB$ at a resonance frequency $f_R = 6GHz$, with input size to the optimizer equal to 154 bits. Similarly,

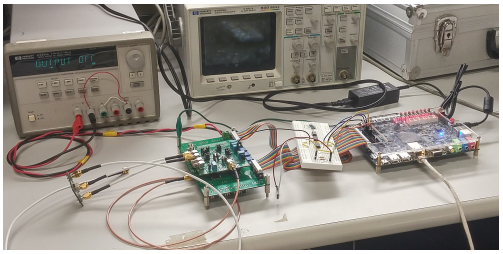


Fig. 6: Measurement setup of the setting the tuning knobs of the BPF circuit.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TETC.2020.3025561, IEEE Transactions on Emerging Topics in Computing

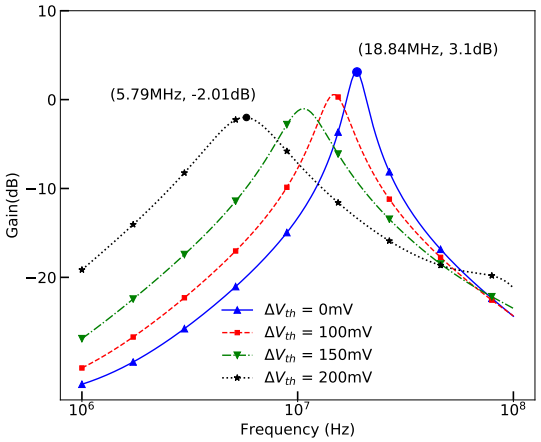


Fig. 7: The change in output response of BPF due to aging. The BPF is designed for a center frequency of 18.84MHz and gain of 3.1dB. As the circuit gets aged, the center frequency and the gain reduce to 5.79MHz and −2.01dB, respectively.

the LDO's specifications are $PSR \leqslant -50dB$ and a phase margin larger than 45° with optimizer input size equal to 234 bits. The experiments are executed on 40, 10-core Intel Xeon processors running at 2.8GHz with 256 GB of RAM. The designs are synthesized using Synopsys Design Compiler tool using Nangate 45nm open cell library [40].

Measurement setup: Fig. 6 shows the printed circuit board containing the BPF chip fabricated using the IBM-180nm process. The output response of the BPF for each tuning knob setting is collected from this chip. The optimizer is implemented on an FPGA, and a dual voltage source is used as supply. **Measurement setup for aging analysis.** For analyzing the impact of aging on BPF, V_{th} is increased for the PMOS and NMOS transistors to model the NBTI and HCI effects, respectively, caused due to aging.

The in-phase and the quadrature-phase values of the input and output voltages are measured at the center, lower, and upper cut-off frequencies. The experiment is repeated for all possible tuning knob settings, and the voltage values are logged for each simulation run. The fresh and the aged analog circuits are simulated by applying the correct key to the optimizer. Fig. 7 shows the degradation in the center frequency and gain at the center frequency as the circuit ages. The center frequency and the gain of BPF are designed to be $18.84MHz$ and $3.1dB$, respectively. Also, as illustrated in the figure, as the circuit ages, the output response deviates marginally from the original response. The binary equivalent of the voltage values is analyzed to determine the bit positions to lock, as mentioned in Section 3.5. As indicated by Fig. 7, the output responses vary marginally, i.e., the MSBs are constant, but the LSBs vary between the measurements before and after the aging of the BPF. Hence, a designer needs to consider the MSB positions for locking and the LSBs as don't cares.

4.2 Effect of tuning knobs on BPF's output

For a locking technique on the BPF circuit to be effective, any deviation in the tuning knob values from its ideal set of values should degrade the BPF's response. The normalized error value can quantify this effect. Fig. 5 shows the normalized error value for different tuning knob values. As one can see, when the (R1,R2) values are (27.68KΩ, 10.38KΩ), the normalized error value is zero, and for all the other cases, there is a non-zero error. Thus, only on setting the tuning knobs to the correct values, the desired response is obtained. Any deviation from these correct values indicates an incorrect response from the BPF circuit.

4.3 Effect of logic locking on tuning knob

The following section illustrates the impact of different SFLL locking on the AMS circuits. **SFLL-HD^0 and SFLL-HD^h.** Table 1 lists the effect of SFLL-HD^0 and SFLL-HD^h logic locking techniques on the optimizer circuits of BPF, LNA, and LDO. In the case of SFLL-HD^0, when the key size equals the input size of the optimizer circuit, $k = n = 220$, the HD $h = 0$, and only one input pattern can be protected. Based on the normalized error in Fig. 5, we choose to protect the pattern that results in the minimum error. Hence, the input pattern resulting in the optimal tuning knob values, i.e., $(27.68K\Omega, 10.38K\Omega)$ is protected; in this case, the normalized error is 0%. For an incorrect key, the optimizer sets the tuning knob that results in a normalized error value of at least 8.11%. Though the security level (s) achieved by this approach is 220, the normalized error value is only 8.11%.

One approach to increase the error value is to protect more number of inputs patterns. This increase in the error can be achieved by reducing the key size. For instance, for SFLL-HD^0 and a key size of $k = 112$, the number of input patterns protected is 3.25×10^{32}, increasing the normalized error to 44.59%. Similarly, by choosing a key size of $k = 87$, a normalized error value is increased to 72.97%. However, one cannot reduce the key size below 80 bits, because this reduces the s and hence, the search space to less than 2^{80}, making it vulnerable to SAT and brute-force attacks. Another approach to increase the number of protected input patterns and hence, the normalized error value is to use SFLL-HD^h, whose results in Table 1.

In case of LNA, SFLL-HD^0 and SFLL-HD^h achieve the normalized error rate of 3100% and 0%, respectively. For LDO, SFLL-HD^0 and SFLL-HD^h obtain the normalized error rate of 39.58% and 0.7%, respectively. As one can see, for the same key size, SFLL-HD^h protects more input patterns compared to SFLL-HD^0. For instance, in case of BPF, for a key size of 220, SFLL-HD^0 protects only one input pattern, whereas SFLL-HD^h, for $h = 37$, protects 1.37×10^{42}. However, the normalized error rate is still the same (i.e., 8.11%) or even lesser (i.e., for LNA it is 0%). This is because SFLL-HD^h requires all the protected input patterns to have the same HD from the key with key size equal to the input size. The probability of all the patterns protected having the same h is very small. This indicates that SFLL-HD^0 results in a higher error than SFLL-HD^h.

SFLL-flex. Table 2 lists the effect of SFLL-flex on the AMS circuit. The input to the optimizer is 220 bits. The output of the optimizer is a set of two tuning knobs, where each of them is 5 bits. The input to the optimizer corresponds to the BPF frequency response measurements for the particular tuning knob settings. As the effective size of the tuning knobs are 10 bits, there are 1024 possible settings for the tuning knobs. Hence, there are only 1024 unique BPF responses. Though there is a possibility of 2^{220} input patterns

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TETC.2020.3025561, IEEE Transactions on Emerging Topics in Computing

TABLE 1: Effect of SFLL-HD^0 and SFLL-HD^h logic-locking techniques on the optimizer circuits of BPF, LNA, and LDO. In the case of BPF, the data is collected from the IBM-180nm process BPF chip described in Section 4. The correct values of the tuning knobs for BPF are $(27.68\text{K}\Omega, 10.38\text{K}\Omega)$. We used simulation results for LNA and LDO. % error listed is the minimum error on applying any incorrect key.

Analog Circuit	# of inputs	Key size	Security level (s)	# of patterns protected	% error (min.)	Key size	Hamming distance (h)	Security level (s)	# of patterns protected	% error (min.)
Bandpass filter	220	220	220	1	8.11	220	1	212	220	8.11
	220	112	112	3.25×10^{32}	44.59	220	20	126	1.19×10^{28}	8.11
	220	87	87	1.09×10^{40}	72.97	220	37	80	1.37×10^{42}	8.11
Low noise amplifier	154	154	154	1	0	154	1	146	154	0
	154	84	84	1.18×10^{21}	3100	154	9	107	1.06×10^{14}	0
	154	81	81	9.44×10^{21}	3100	154	17	80	1.73×10^{22}	0
Low-dropout regulator	234	234	234	1	0.7	234	1	226	234	0.7
	234	135	135	6.34×10^{29}	12.59	234	20	138	4.32×10^{28}	0.7
	234	109	109	4.25×10^{37}	39.58	234	41	81	9.89×10^{45}	0.7

TABLE 2: Impact of aging on the locked AMS circuits using SFLL-flex. The error in the output response for a correct key, for all the cases is 0. The maximum possible error is 255.

Analog Circuit	Input size	Key size	Security level	Area %	% Error before aging	% Error after aging
Bandpass filter	220	80	219	0.14	50	50
	220	220	210	171.30	50	0
	220	220	210	107.53	39.21	0
	220	220	211	70.10	27.45	0
Low noise amplifier	154	84	153	0.13	50	50
	154	154	144	92.46	50	0
	154	154	144	72.01	39.21	0
	154	154	145	46.72	28.23	0
Low-dropout votlage regulator	240	122	240	0.14	50	50
	240	240	230	116.97	50	0
	240	240	231	60.72	39.60	0
	240	240	232	27.03	27.84	0

to the optimizer, as the input size is 220 bits, the input depends only on the tuning knob settings. Hence, there are only 1024 unique input patterns to the optimizer. As the SFLL-flex has the flexibility to choose the input patterns to protect, we can protect all the 1024 patterns, as shown in Table 2. However, this also incurs a huge area. As it is not necessary to protect the input patterns for which the cost is maximum (BPF's frequency response deviated from the original response), we could ignore these patterns from protecting. Based on the minimum error the attacker has to encounter for an incorrect key, the corresponding input patterns can be protected, thereby reducing the area overhead. Also, the security level (s) achieved for all the cases is more than 80, ensuring resiliency against the SAT attack.

Based on the minimum deviation in the output response (which is the input pattern to the locked optimizer) the attacker has to encounter for an incorrect key, the defender protects all the input patterns that correspond to a deviation less than the chosen deviation. The error in the cost function quantifies this deviation, which is tabulated in Table 2. In a few cases, for the same input size, key size, and s achieved, the area overhead value differs. For example, in BPF, for two setups the input size and key size is equal to 220 and s is equal to 210. However, these setups have different area overheads (107.53% and 171.30%). This is because the overhead depends on the number of protected input patterns (c). Also, s is calculated using the SAT resilience formula for SFLL-flex given by $k - \lceil \log_2 c \rceil$. As s is dependent on $\log_2 c$, its value does not change for a marginal increase in c. Adding to this, the impact of aging on the AMS circuit is studied. The deviation in the output response (error in cost function) when an incorrect key is applied to an aged AMS circuit (analog circuit-under-protection and the logic-locked optimizer) is added to Table 2. Fig. 8.(a) shows the impact of an incorrect key on the optimizer locked using SFLL-flex when the PIPs whose cost function value is (i) less than 70, (ii) less than 100, and (iii) less than 255. The cost function depicts the deviation of the BPF response from the expected one. Hence, as the cost function increases, the deviation of the BPF output from the expected response also increases, as illustrated in Fig. 8. Initially, SFLL-HD^0 and SFLL-HD^h were used to lock the optimizer as it does not require an expensive restore unit, where all the protected input patterns are stored. However, by judiciously choosing the input pattern to protect and the key size, both security and lesser area overhead using SFLL-flex is achieved. Also, as shown in Fig. 8, there is no degradation in the security even after the circuit being aged.

4.4 Security analysis

The following section shows the resiliency offered by the locked optimizer.

Resiliency against SAT. The resiliency against SAT attack offered by SFLL-HD^0 and SFLL-HD^h are k and $k - log_2\binom{k}{h}$, respectively [10]. From Fig. 9, we can infer that security level s achieved for the BPF is the maximum when $h = 0$ or $h = 220$ and the minimum when $h = 110$. To ensure that the locked circuit is SAT attack resilient, we need to choose h and k such that the security level is greater than 80. Hence, the allowable h values can be $0 \leqslant h \leqslant 37$ or $183 \leqslant h \leqslant 220$, and the corresponding number of input patterns which can be protected are $1 <$ # of patterns protected $< 1.37 \times 10^{42}$. Similarly, for LNA, the allowable value of h is $0 \leqslant h \leqslant 17$ or $137 \leqslant h \leqslant 154$ and the number of input patterns which are protected ranges $(1, 1.73 \times 10^{22})$. For the LDO, $0 \leqslant h \leqslant 41$ or $193 \leqslant h \leqslant 234$ and the input patterns protected are in the range $(1, 9.89 \times 10^{45})$.

From Fig. 10, the security increases with the increase in key size, whereas the number of input patterns protected reduces with the increase in key size. A key size, $k \geqslant 80$ ensures resilience against the SAT attack. Hence, the number of input patterns that can be protected ranges $(1, 1.39 \times 10^{42})$ for BPF. Likewise, the number of input patterns protected for LNA ranges $(1, 1.89 \times 10^{22})$ and that of LDO is $(1, 2.28 \times 10^{46})$. The time required for the attack, as shown in Fig. 11, increases exponentially with the input size. For the input size of 14, the attack takes close to 1.5 hours to identify the key. This trend indicates that our

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TETC.2020.3025561, IEEE Transactions on Emerging Topics in Computing

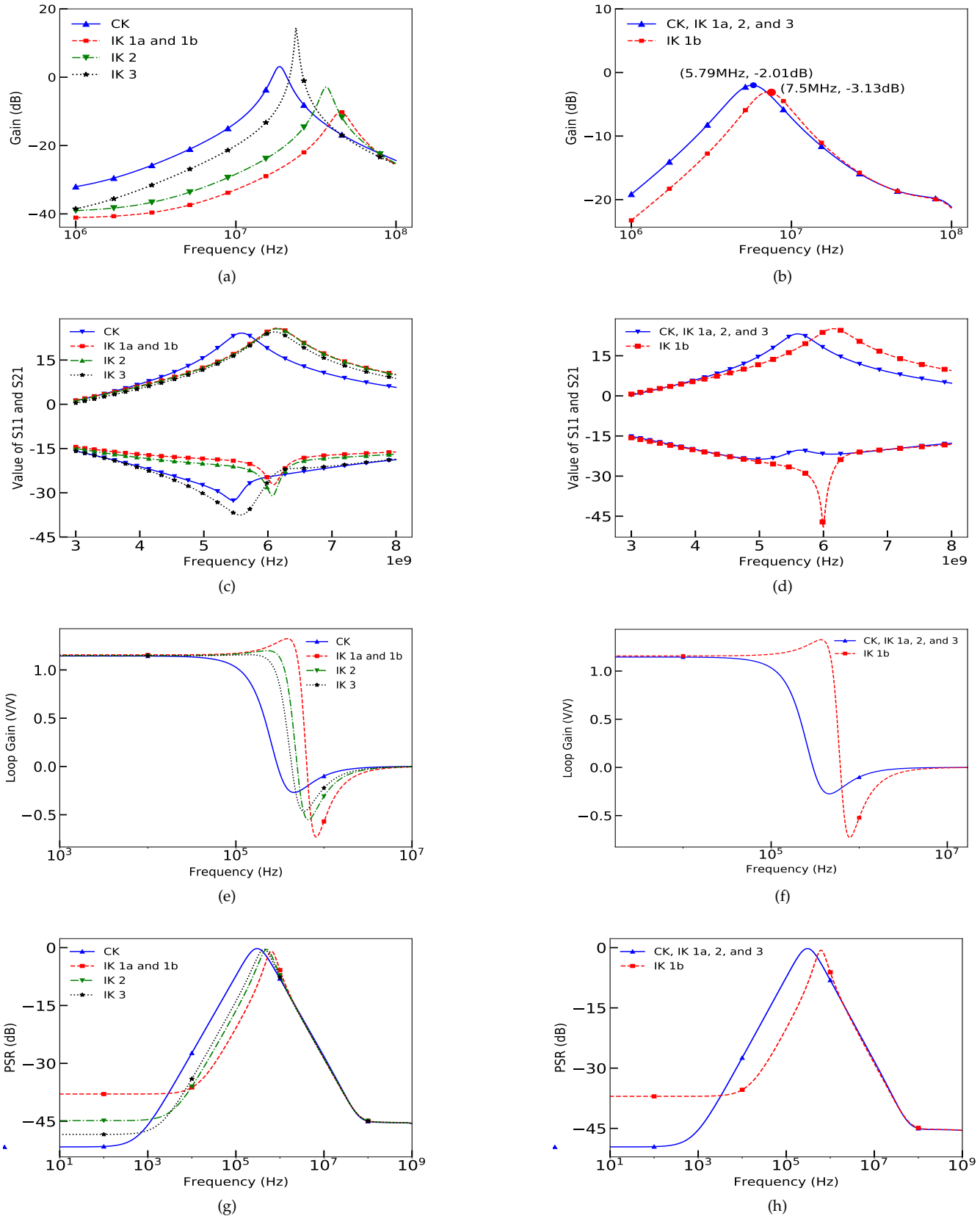


Fig. 8: The impact of aged analog circuit on SFLL locked-optimizer. (a) Impact on the frequency response of new BPF circuit. (b) Impact on the frequency response of aged BPF circuit. (c) Impact on the S11 and S21 parameters of the new LNA circuit. (d) Impact on the S11 and S21 parameters of the aged LNA circuit. (e) Impact on the loop gain of the new LDO circuit. (f) Impact on the loop gain of the aged LDO circuit. (g) Impact on the PSR of the new LDO circuit. (h) Impact on the PSR of the aged LDO circuit. Correct key (CK), incorrect key (IK), and power supply rejection (PSR).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TETC.2020.3025561, IEEE Transactions on Emerging Topics in Computing

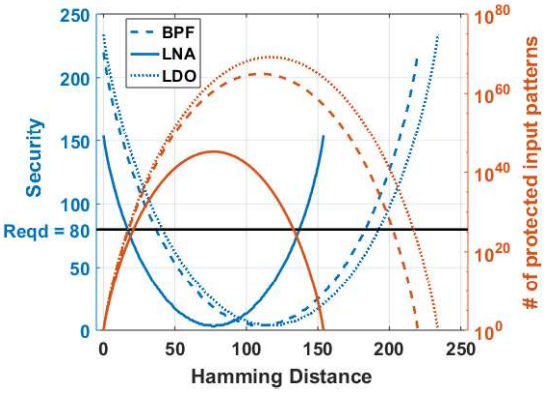


Fig. 9: The impact of HD on SAT attack resiliency and the number of input patterns protected in SFLL-HD^h. The right-hand side y-axis is in log scale.

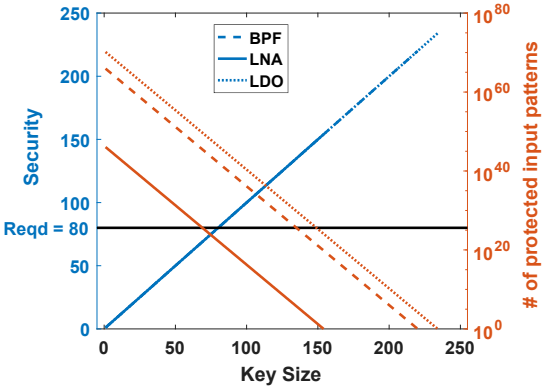


Fig. 10: Key size vs. SAT attack resiliency and the number of input patterns protected for BPF, LNA, and LDO. The right-hand side y-axis is in log scale. The Hamming distance $h = 0$. The security level achieved by BPF, LNA, and LDO are the same and hence, are superimposed.

technique is secure against the SAT attack. Similarly, it is also secure against AppSAT [41], as we protect only a linear number of input patterns.

Resiliency against removal attack [22]. An attacker cannot remove the locked optimizer circuit and make the analog circuit functional because the tuning knobs are not set to the optimal values due to process variations, thus preventing removal attacks. If he removes the locked optimizer unit,

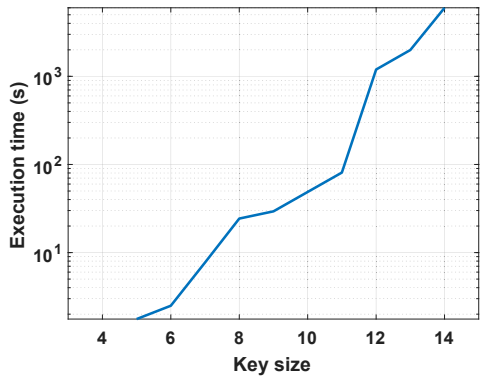


Fig. 11: Execution time of the SAT attack for BPF. The time required for the attack to find the key increases exponentially with respect to key size. Note that y-axis is in log scale.

the circuit parameters will be fixed to the default value. The probability of this value being equal to the desired value to address process variations is negligible. An attacker cannot set the tuning knob value to its optimal value through a focused-ion-beam (FIB) because even identifying the value of one chip cannot be used to set the value for another chip, as these values are different because of process variations. In other words, the amount of compensation varies from one chip to another chip.

Resiliency against bypass attacks [42]. Bypass attack finds the PIPs that give an incorrect output for an incorrect key. The attacker adds a bypass circuitry around the protection block to restore the output for those PIPs. However, the bypass attack cannot compute all the PIPs from the circuit protected using SFLL. This is because, in SFLL, a PIP produces the same incorrect output for most of the incorrect key values. Also, the output corresponding to the PIP may be restored correctly even for an incorrect key. Hence, the bypass attack does not consider the corresponding input pattern as PIP. Therefore, the construction of the bypass circuitry using the incomplete set of PIPs will be erroneous. Thus, it renders the attack unsuccessful.

4.5 Effect of incorrect keys

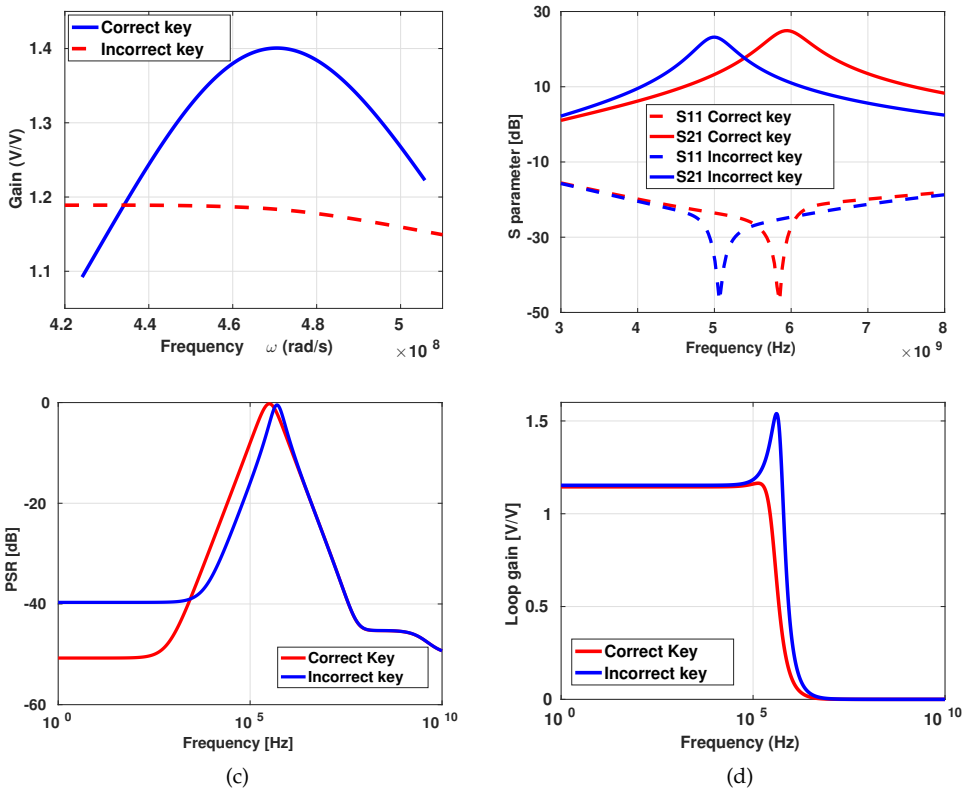
The response of the circuits for a correct and an incorrect key are compared for the most commonly used metrics for the analog circuits-under-protection [31], [33], [34]. The BIST structure should be enhanced to measure advanced metrics, e.g., LNA noise. This is because the transient analysis and the quadratic sampling are not sufficient to measure this metric. Fig. 12(a) shows the difference in the frequency response of the BPF. In this case, the correct key allows the optimizer, tune the circuit to the target ω_o and BW, while an incorrect key forces the optimizer to tune to a lower frequency and also reduces the Q and gain values. Fig. 12(b) compares the difference in the S-parameters of the LNA targeting $f_R = 6GHz$. One can observe an error of $1GHz$ in f_R and an error on S_{11} and S_{21} of at least 15 dB. Finally, the deviation on the LDO performance for the two cases was evaluated. Fig. 12(c) shows a degradation close to 10dB in the PSR. Fig. 12(d) shows a large peaking on the loop gain for the incorrect key, which indicates a low phase margin and potential instability.

From Fig. 12(b) it is evident that the deviation in gain S21 is 10dB. As mentioned in [33], the gain of the LNA must be large enough to minimize the noise contribution, specifically in the downconversion mixers. This 10dB reduction leads to higher input noise corrupting the signal. Also, the LNA is designed such that the gain has its peak value at the frequency band of interest. If the amplifier resonates at another frequency, it does not only mean that the signal of interest is not amplified enough, but also that the receiver is acquiring the signal from a different frequency band. In communications, this causes interference between different channels and the channel of interest.

4.6 Analog circuit's performance for a random tuning knob setting

The minimum percentage normalized mean-squared-error (NMSE) due to arbitrary tuning knob settings can be less

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TETC.2020.3025561, IEEE Transactions on Emerging Topics in Computing



(c) (d)

Fig. 12: Behavior of the analog circuits for correct and incorrect keys on using SFLL-HD^0. The key size used for BPF, LNA, and LDO are 87, 81, and 109, respectively. (a) Frequency response of BPF, (b) S-parameters of LNA, (c) Power supply rejection (PSR) of LDO, and (d) Loop gain of LDO.

than the NMSE due to process variation, thereby giving a performance close to the desired one. However, as the attacker does not have the resources to modify the layout, it is not possible to choose an arbitrary tuning knob setting. The following are the reasons why choosing the tuning knob at random does not always work:

1) Please note that the NMSE in the output response when the correct tuning knob is chosen is 0 (as the actual response is equal to the expected response). However, the probability of randomly choosing this optimal tuning knob setting is 1/1024, as there are a total of 1024 tuning knob settings.
2) As this technique is resilient only against the overproduction attack, the attacker does not have the resources for layout-level modifications such as removing the locked optimizer or changing the circuit parameters to control the gain of LNA.
3) Even if the attacker removes the locked optimizer, the optimal tuning knob setting will change chip to chip due to process variations. Hence, choosing the same setting on all chips will not compensate for process variation.

A statistical representation of the NMSE for more number of arbitrary tuning knob settings may not help infer the impact of random tuning knob selections due to the above reasons.

4.7 Impact of increasing the number of tuning knobs

In this section, we explain the effect of increasing the number of tuning knobs, i.e., the tunable parameters on the security of the locked AMS circuit. Along with the resistors

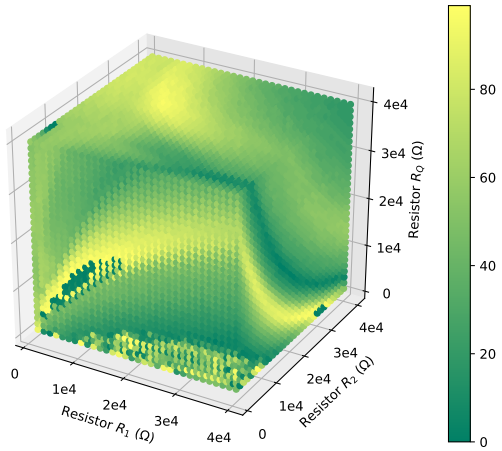


Fig. 13: Cost function (error) for all possible settings of three tuning knobs ($\mathbf{R_1}$, $\mathbf{R_2}$, and $\mathbf{R_Q}$).

R_1 and R_2, R_Q is also made tunable. Hence, there are three 5-bit tuning knobs. Fig. 13 shows the cost function value for all possible tuning knob settings. As illustrated in the figure, very few tuning knob settings give the desired circuit performance, i.e., minimal cost function value. The optimizer controlling the three tuning knobs is locked using SFLL-flex. Here, all possible input patterns (2^{15}) are protected. Fig. 14 shows the output response of the BPF circuit for the (in)correct keys supplied to the locked optimizers controlling two and three tuning knobs.

As shown in the figure, the deviation of the output response from the desired response, for an incorrect key,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TETC.2020.3025561, IEEE Transactions on Emerging Topics in Computing

increases as the number of tuning knob increases. As shown in Fig. 14, for the correct key, both the optimizers tune the resistors for providing the required $f_c = 18.84MHz$ and $Gain = 3.1dB$. For an incorrect key, the optimizer controlling two tuning knobs sets the BPF to function with $f_c = 34.47MHz$ and $Gain = 0.362dB$. However, the effect of an incorrect key on the optimizer controlling three tuning knobs has greater impact on the BPF's performance. It sets the tuning knobs such that $f_c = 23.71MHz$ and $Gain = 24.34dB$. Unlike two tuning knobs, where only f_c varies considerably and $Gain$ varies marginally, in three tuning knob settings, both the metrics vary considerably.

4.8 Effect of aging on the locked-optimizer

Analog circuits are subjected to aging effects such as negative bias temperature instability (NBTI) [43] and hot carrier injection (HCI) [44]. PMOS transistors are affected by NBTI. An increase in threshold voltage can model this effect in the PMOS transistors. The change in V_{th} due to NBTI is modeled by [43] as,

$$\Delta V_{th} \approx exp(\alpha_1 V_{GS}) t^{n_p} + V_{GS}^{\alpha_2}(C_R + n_R log_{10}(t)) \quad (5)$$

Here, α_1 and α_2 are voltage scaling factors, which are 0.26 and 2.4, respectively. n_p and n_R are time exponents. They are process-dependent parameters similar to C_R. V_{GS} is the gate to source voltage. The impact of HCI on the NMOS transistors degrades the drain current, which is depicted as an increase in the threshold voltage. The change in V_{th} due to HCI is modeled by [43] as,

$$\Delta V_{th} \approx \frac{1}{\sqrt{L}} exp(\alpha_3 V_{GS}) exp(\alpha_4 V_{DS}) t^{n_{HC}} \quad (6)$$

Here, α_3 and α_4 are process-dependent voltage scaling factors, $n_{HC} \approx 0.5$ is a time exponent, L is the transistor length, and V_{DS} is the drain to source voltage.

Consider the locked analog circuit used in applications such as transceivers and communication protocols. As only the analog circuit is powered-on for a longer duration, the impact of aging is comparatively higher on this circuit than the other circuits which are powered-off after the tuning knobs are selected. Hence, the analog circuit is simulated with an increased V_{th} in PMOS and NMOS transistors to model the aging process. This increase in V_{th} causes a marginal difference in the analog circuit response, which in turn varies the input voltage to the locked optimizer marginally. As the increase in V_{th} in PMOS and NMOS increases with the age of the circuit, the output response of the analog circuits changes with age. This trend is illustrated in Fig. 7, where the x-axis denotes the frequency of operation, and the y-axis is the frequency response of the second-order BPF. It is evident from the figure that as the analog circuit ages, its output response deviates more from the desired response. A change in the output response results in the change in the input patterns to the locked-optimizer. As the defender has protected only a handful of input patterns, the new input patterns from the aged analog circuit may not be protected. Hence, it is imperative to study the effects of aging on the locked-optimizer.

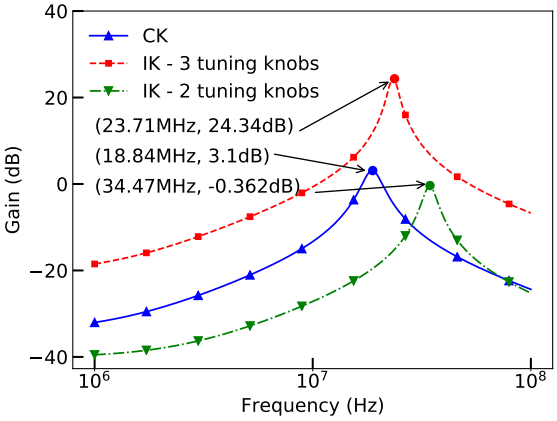


Fig. 14: The optimizer of the BPFs with two (R_1, R_2) and three tuning knobs (R_1, R_2, and R_Q) are locked using SFLL-flex. The output response for the correct key (CK) is same in both the cases as indicated in the figure. However, the output response for an incorrect key is deviated from the original response much more for the BPF with three tuning knobs rather than two.

In the following section, the impact of aging is studied on the logic-locked AMS circuits. The defender chooses the PIPs that result in a minimum error, i.e., the deviation of the output response from the desired response the attacker has to experience for an incorrect key. The optimizer is locked using SFLL-flex to protect (i) the input patterns corresponding to all possible tuning knob settings which give an error of 50% (ii) the input patterns which give an error of less than 27%, and (iii) the input patterns which give an error of less than 39%. The PIPs corresponds to the analog circuit before aging. Figures 8 (a), (c), (e), and (g) show the output responses for the analog circuits when the locked-optimizer is supplied with correct and incorrect keys for the above three cases. As the number of PIPs increases, the error in output response for an incorrect key increases. The impact of aging on the frequency response, S11 and S21 parameters, and PSR and loop gain, are plotted for BPF, LNA, and LDO, respectively. For the analog circuits considered, the output response corresponding to the incorrect keys, IK 1a and IK 1b output responses are for the optimizer where the input patterns corresponding to all the are tuning knobs are protected. For the incorrect key IK 1a, all the bits of the input pattern are considered by the locking mechanism, thereby achieving a key size of 220, 154, and 240 bits for BPF, LNA, and LDO, respectively. Whereas for the response due to incorrect key IK 1b, only the MSBs and the constants bits of the voltage inputs are considered by the locking mechanisms, and the rest of the bit positions in the input patterns are considered as don't cares. This reduces the key size to 80, 84, and 122 for BPF, LNA, and LDO, respectively. Similarly, the output responses corresponding to incorrect keys 2 and 3 belong to the optimizer, which protects the input patterns whose error in output response is less than 39% and less than 27%, respectively. The optimizer considers all the bits similar to the response for incorrect key IK 1a case, thereby achieving a larger key size equaling the input size.

The analog circuit is now replaced with the aged circuit to study the effect of aging on the SFLL locking. Figures 8 (b), (d), (f), and (h) shows the output responses for the analog circuits, when the optimizer locked using SFLL-flex,

is supplied with correct and incorrect key IK 1b based on the input patterns they are protecting. As few of the bit positions in the input patterns to the optimizer (from the analog circuit) have changed due to aging (especially LSBs), the locked optimizer considering all the bit positions in the input patterns fails to secure the circuit. This trend is illustrated in the figures 8 (b), (d), (f), and (h). However, the optimizer which considers only the MSBs (and/or the bit positions whose values are constant) with lower security level (but more than 80) protects the analog circuits as shown by the output response for incorrect key IK 1b.

4.9 Discussion

Why can we not protect all the digital components in the AMS circuit? A simple solution is to protect all the digital components of the AMS circuit. However, this seemingly straightforward approach is not simple and may not meet the desiderata for analog circuits, for reasons below. The desiderata for protecting AMS circuit via logic locking of digital components:

- An attacker should not be able to identify the locked digital part, remove it, and make the resultant analog circuit functional[1].
- Logic locking the entire digital circuit may not necessarily yield incorrect responses from the analog part. Hence, the digital circuit needs to be locked such that the analog component becomes non-functional when an incorrect key is applied.
- State-of-the-art logic locking techniques can protect only a linear number of input patterns in key size [10]. Hence, one needs to select which input patterns to protect, such that incorrect keys will have the highest impact on the functionality of the analog circuit.
- Locking the entire circuit incurs high area, power, and delay overhead. Hence, one has to be judicious in selecting which components to protect.

Power, delay, and area overheads. In this implementation, the power overhead is not a concern since the optimization and security platform is consuming power only at the start time for a short period. Once the optimization finds a solution and sets the tuning knobs, the digital core is turned off. Concerning the delay overhead, there is a delay between the circuit turn-on time and the time at which the regular operation starts. This delay is the time taken by the optimizer to choose the optimal tuning knob settings. Area overhead of SFLL-HD^0 for BPF, LNA, and LDO is 8.79%, 2.61%, and 3.08%, respectively. Similarly, for SFLL-HD^h, the overhead is 8.78%, 5.84%, and 4.91%, respectively, for the h values listed in Table 1. In case of SFLL-flex, the overhead is 0.14%, 0.13%, and 0.14%, for BPF, LNA, and LDO, respectively, proving to be the most area-efficient variant of SFLL. **Effect of temperature and environmental noise.** The on-chip optimizer can measure the performance of the AMS circuit with respect to the circuit components along with operating conditions, such as temperature and noise on the power supply. Thus, the optimizer can recalibrate and set the tuning knobs to obtain the desired response—but only when the correct key is in place. Thus, our technique can ensure the effect of locking, even in the presence of temperature variation and environmental noise.

Can we individually attack the analog and digital sections of the AMS circuit? As we are targeting resilience only against overproduction attacks, the threat model assumes that the attacker can only overproduce the chip but cannot physically modify the existing layout. Hence, he/she cannot remove the optimizer circuit and independently target the analog circuit. Even if the attacker simulates the analog circuit for the different tuning knob settings and determines the correct settings, he/she cannot change the tuning knob as they are not controllable by the attacker.

What guarantees that an incorrect key does not produce the correct circuit performances? As the optimizer is locked using SFLL-flex, the guarantees that an incorrect key does not produce the correct circuit response is given by the security metrics of SFLL-flex. This is because the circuit response depends on the unique settings of one of the 1024 tuning knob settings, which in turn is the output of the locked optimizer. As the input patterns which have the highest impact on the correct tuning knob settings are protected, unless the optimizer is provided with the correct key, the analog circuit does not produce the desired output response. Only the correct key selects the optimal tuning knobs, whereas all the incorrect key selects sub-optimal tuning knobs that produce sub-optimal performance. Hence, as given in [10], the output is corrupted for those input patterns which are protected. This is given by the SAT and removal attack resiliencies, which are $k - \lceil \log_2 c \rceil$ and $c \times 2^{n-k}$, respectively. Here, n, k, and c are the input size, key size, and the number of protected input patterns, respectively.

Is it possible to determine the correct key to unlock the optimizer using the quantified Boolean formulation? An attacker can try finding the key using the input and output relationship of the analog circuit. In the oracle, an attacker can control and observe the input and output ports, respectively, of the analog circuit. He/she can send in the desired input and observe the corresponding output as the key input is loaded with the correct key. He/she can formulate the operation of the analog circuit as QBF equations and try to find the optimizer key that satisfies this relationship. However, an attacker cannot find this key to unlock the optimizer because of the nature of digital locking techniques. Furthermore, an attacker does not know the complete input and output relationship of the optimizer to build his/her own optimizer. This is due to the following reasons: (i) the underlying digital locking techniques are secure in revealing the complete functionality of the optimizer and (ii) every locked chip only reveals one or few input-output relationships of the optimizer, and this is not enough to build the complete optimizer. This happens because the effects of process variations are different for different chip and thus applying different inputs to the optimizer.

R1.6 Due to the large size of the passive components, the number of components in an array is limited. Does this limit the key search space? As this technique is resilient only against overproduction, the key size is not dependent on the number of passive devices in the array. Instead, it is dependent on the size of the input to the locked-optimizer. Depending on the process variation impact, the correct

1. Here, we consider an analog design as functional when it produces the expected response.

value of the passive device is chosen by the optimizer via the tuning knobs. Hence, it is necessary to unlock the optimizer, which is protected by SFLL. In this locking technique, the key size should be less than or equal to the input size. For example, in BPF, as the input size to the optimizer is 220 bits, the key size can be a maximum of 220 bits. Hence, this key size and hence, the key search space is not limited by the size or number of passive devices in the array.

5 Conclusion

In this paper, we propose the first technique to thwart the overproduction of AMS circuits, by securely locking the digital part, which is controlling the tuning knobs judiciously. Our analysis indicates that by properly selecting two tuning knobs, we can secure several performance metrics of different analog circuits—a BPF, an LNA, and an LDO. On applying an incorrect key, our approach achieves at least 27.45% error and at most 50% error in the circuit's response when the optimizer is locked using SFLL-flex. Our technique is agnostic to logic locking techniques: we have used SFLL-flex [10], as it can prevent SAT [27], AppSAT [41], removal [22], sensitization [6], and bypass attacks [42]. Our approach is provably-secure, as it leverages the properties of SFLL. More importantly, it is well integrated with the analog component, without sacrificing the security properties of SFLL.

From the simulation results, one can conclude that the SFLL-flex technique secures the analog circuit irrespective of aging. We can also increase the error in the output response of the analog circuit experienced by the attacker for an incorrect key by increasing the number of tunable parameters in the circuit. Our future work entails: (i) Exploring the effect of other logic locking techniques; (ii) Embedding secret keys as part of analog designs, not just digital; and (iii) Exploring techniques to prevent piracy and not just overproduction.

6 Acknowledgement

This work is supported by National Science Foundation CNS-1618824, CNS-1828840, STARSS-1618797 and SATC CAREER-1822848, Semiconductor Research Corporation 2016-T3S-2688 and 2016-T3S-2689, and Intel.

References

[1] M. Rostami, F. Koushanfar, and R. Karri, "A Primer on Hardware Security: Models, Methods, and Metrics," *Proceedings of the IEEE*, vol. 102, no. 8, pp. 1283–1295, 2014.

[2] T. S. Perry, "Why Hardware Engineers Have to Think Like Cybercriminals, and Why Engineers Are Easy to Fool," 2017, Last accessed on 01/13/2020. [Online]. Available: https://bit.ly/2RfoBkS

[3] P. Tuyls, G. Schrijen, B. Škorić, J. van Geloven, N. Verhaegh, and R. Wolters, "Read-Proof Hardware from Protective Coatings," *Cryptographic Hardware and Embedded Systems*, pp. 369–383, 2006.

[4] M. Integrated, "DeepCover Security Manager for Low-Voltage Operation with 1KB Secure Memory and Programmable Tamper Hierarchy," https://www.maximintegrated.com/en/products/embedded-security/security-managers/DS3660.html, 2010, Last accessed on 01/13/2020.

[5] J. A. Roy, F. Koushanfar, and I. L. Markov, "Ending Piracy of Integrated Circuits," *IEEE Computer*, vol. 43, no. 10, pp. 30–38, 2010.

[6] J. Rajendran, Y. Pino, O. Sinanoglu, and R. Karri, "Security Analysis of Logic Obfuscation," *IEEE/ACM Design Automation Conference*, pp. 83–89, 2012.

[7] A. Baumgarten, A. Tyagi, and J. Zambreno, "Preventing IC Piracy Using Reconfigurable Logic Barriers," *IEEE Design & Test of Computers*, vol. 27, no. 1, pp. 66–75, 2010.

[8] R. A. Rutenbar, "Design Automation for Analog: The Next Generation of Tool Challenges," *IEEE/ACM International Conference on Computer Aided Design*, pp. 458–460, 2006.

[9] "Top 5 Most Counterfeited Parts Represent a $169 Billion Potential Challenge for Global Semiconductor Market," https://technology.ihs.com/405654/top-, 2012, Last accessed on 01/12/2020.

[10] M. Yasin, A. Sengupta, M. T. Nabeel, M. Ashraf, J. J. Rajendran, and O. Sinanoglu, "Provably-Secure Logic Locking: From Theory To Practice," *ACM SIGSAC Conference on Computer & Communications Security*, pp. 1601–1618, 2017.

[11] D. H. K. Hoe, J. Rajendran, and R. Karri, "Towards Secure Analog Designs: A Secure Sense Amplifier Using Memristors," *IEEE Computer Society Annual Symposium on VLSI*, pp. 516–521, 2014.

[12] Y. Bi, J. Yuan, and Y. Jin, "Beyond the Interconnections: Split Manufacturing in RF Designs," *MDPI Electronics*, vol. 4, no. 3, pp. 541–564, 2015.

[13] J. Wang, C. Shi, A. Sanabria-Borbon, E. Sanchez-Sinencio, and J. Hu, "Thwarting Analog IC Piracy via Combinational Locking," *IEEE International Test Conference*, pp. 1–10, 2017.

[14] V. V. Rao and I. Savidis, "Parameter Biasing Obfuscation for Analog IP Protection," *IEEE Latin American Test Symposium*, pp. 1–6, 2017.

[15] J. Leonhard, M.-M. Louërat, H. Aboushady, O. Sinanoglu, and H.-G. Stratigopoulos, "Mixed-Signal Hardware Security Using MixLock: Demonstration in an Audio Application," *IEEE International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design*, 2019.

[16] K. Juretus, V. Venugopal Rao, and I. Savidis, "Securing Analog Mixed-Signal Integrated Circuits Through Shared Dependencies," *Great Lakes Symposium on VLSI*, pp. 483–488, 2019.

[17] G. Volanis, Y. Lu, S. G. R. Nimmalapudi, A. Antonopoulos, A. Marshall, and Y. Makris, "Analog Performance Locking through Neural Network-Based Biasing," *VLSI Test Symposium*, pp. 1–6, 2019.

[18] N. G. Jayasankaran, A. S. Borbon, A. Abuellil, E. Sanchez-Sinencio, J. Hu, and J. Rajendran, "Breaking Analog Locking Techniques via Satisfiability Modulo Theories," *IEEE International Test Conference*, 2019.

[19] F. Yang, M. Tang, and O. Sinanoglu, "Stripped Functionality Logic Locking With Hamming Distance-Based Restore Unit (SFLL-hd) — Unlocked," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 10, 2019.

[20] D. Sirone and P. Subramanyan, "Functional Analysis Attacks on Logic Locking," *IEEE/ACM Design Automation and Test in Europe*, pp. 936–939, 2019.

[21] M. Yasin, B. Mazumdar, O. Sinanoglu, and J. Rajendran, "Removal Attacks on Logic Locking and Camouflaging Techniques," *IEEE Transactions on Emerging Topics in Computing*, pp. 1–1, 2017.

[22] ——, "Security analysis of Anti-SAT," *IEEE Asia and South Pacific Design Automation Conference*, pp. 342–347, 2017.

[23] A. Hastings, *The Art of Analog Layout*. Pearson, 2005.

[24] C. Toumazou, G. Moschytz, and B. Gilbert, *Trade-offs in Analog Circuit Design*. Kluwer Academic Publishers, 2002.

[25] T. McConaghy, K. Breen, J. Dyck, and A. Gupta, *Variation-Aware Design of Custom Integrated Circuits: A Hands-on Field Guide*, 2013.

[26] J. Wang, C. Shi, E. Sanchez-Sinencio, and J. Hu, "Built-In Self Optimization for Variation Resilience of Analog Filters," *IEEE Computer Society Annual Symposium on VLSI*, pp. 656–661, 2015.

[27] P. Subramanyan, S. Ray, and S. Malik, "Evaluating the Security of Logic Encryption Algorithms," *IEEE International Symposium on Hardware Oriented Security and Trust*, pp. 137–143, 2015.

[28] P. Mroszczyk, J. Goodacre, and V. F. Pavlidis, "Energy Efficient Flash ADC With PVT Variability Compensation Through Advanced Body Biasing," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 66, no. 11, pp. 1775–1779, 2019.

[29] Y. Zhao, Z. Shang, and Y. Lian, "A 13.34W Event-driven Patient-specific ANN Cardiac Arrhythmia Classifier for Wearable ECG Sensors," *IEEE Transactions on Biomedical Circuits and Systems*, pp. 1–1, 2019.

[30] R. Lowman, "A Holistic Approach to IoT Chip Design," https://bit.ly/36q37Yi, 2015, Last accessed on 01/24/2020.

[31] M. V. V. Rolf Schaumann, Haiqiao Xiao, *Design of Analog Filters*. Oxford University Press, 2009.

[32] G. Volanis, D. Maliuk, Y. Lu, K. S. Subramani, A. Antonopoulos, and Y. Makris, "On-Die Learning-based Self-Calibration of Analog/RF ICs," *IEEE VLSI Test Symposium*, pp. 1–6, 2016.

[33] B. Razavi, *RF Microelectronics*. Pearson Education, 2011.

[34] T. Instruments, "Understanding Low Drop Out (LDO) Regulators," https://bit.ly/37bvvyE, 2006, Last accessed on 01/24/2020.

[35] A. Sengupta, M. Nabeel, M. Yasin, and O. Sinanoglu, "ATPG-based Cost-Effective, Secure Logic Locking," *VLSI Test Symposium*, pp. 1–6, 2018.

[36] I. Guerra-Gómez, E. Tlelo-Cuautle, and L. G. De La Fraga, "Richardson Extrapolation-based Sensitivity Analysis in the Multi-objective Optimization of Analog Circuits," *Applied Mathematics and Computation*, vol. 222, pp. 167–176, 2013.

[37] J. Torres, M. El-Nozahi, A. Amer, S. Gopalraju, R. Abdullah, K. Entesari, and E. Sanchez-Sinencio, "Low Drop-Out Voltage Regulators: Capacitor-less Architecture Comparison," *IEEE Circuits and Systems Magazine*, vol. 14, no. 2, pp. 6–26, 2014.

[38] Y. Feng, G. Takemura, S. Kawaguchi, N. Itoh, and P. R. Kinget, "Digitally Assisted IIP2 Calibration for CMOS Direct-Conversion Receivers," *IEEE Journal of Solid-State Circuits*, vol. 46, no. 10, pp. 2253–2267, 2011.

[39] B. Xiang, Y. Fan, J. Ayers, J. Shen, and D. Zhang, "A 0.5V-to-0.9V 0.2GHz-to-5GHz Ultra-Low-Power Digitally-Assisted Analog Ring PLL with Less Than 200ns Lock Time in 22nm FinFET CMOS Technology," *IEEE Custom Integrated Circuits Conference*, pp. 1–4, 2020.

[40] N. Inc, "NanGate FreePDK45 Open Cell Library," http://www.nangate.com/?page_id=2325, 2011, Last accessed on 01/13/2020.

[41] K. Shamsi, M. Li, T. Meade, Z. Zhao, D. Z. Pan, and Y. Jin, "AppSAT: Approximately Deobfuscating Integrated Circuits," *IEEE International Symposium on Hardware Oriented Security and Trust*, pp. 95–100, 2017.

[42] X. Xu, B. Shakya, M. M. Tehranipoor, and D. Forte, "Novel Bypass Attack and BDD-based Tradeoff Analysis Against All Known Logic Locking Attacks," *Cryptographic Hardware and Embedded Systems*, 2017.

[43] E. Maricau and G. Gielen, "Transistor Aging-induced Degradation of Analog Circuits: Impact Analysis and Design Guidelines," *IEEE Proceedings of ESSCIRC*, pp. 243–246, 2011.

[44] D. Sengupta and S. S. Sapatnekar, "Estimating Circuit Aging Due to BTI and HCI Using Ring-Oscillator-Based Sensors," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 36, no. 10, pp. 1688–1701, 2017.

Nithyashankari G. Jayasankaran (S'18) is pursuing her Ph.D. in Texas A&M University under the guidance of Prof. Jiang Hu and Prof. JV Rajendran. Her research interest is in the hardware security domain, emphasizing on analog and mixed-signal circuits security. Before joining Texas A&M, she was working on SoC emulation, FPGA design, and board testing for base transceiver station based applications.

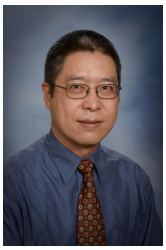
Adriana C. Sanabria-Borbón (S'07, M17) Adriana C. Sanabria-Borbón received the engineering degree (summa cum laude) in electronics engineering from Santo Tomas University, Bogotá, Colombia, in 2012. In 2014 she received the M.Sc. degree from Instituto Nacional de Astrofísica Óptica y Electrónica (INAOE), Puebla, Mexico. Since 2015 is pursuing a Ph.D. degree in electrical engineering at the analog and mixed-signal center (AMSC), Texas A&M University, College Station, Texas.

Her research interests include analog integrated circuits, configurable active filters, design optimization, and hardware security. She held summer internships at Intel Labs in 2017 and 2018, where she worked in the area of power management. She has been the recipient of the TI J. Kilby fellowship on 2018 and 2019.

Edgar Sánchez-Sinencio (F'92-LF'10) was born in Mexico City, Mexico. He received the degree in communications and electronic engineering (Professional degree) from the National Polytechnic Institute of Mexico, Mexico City, in 1966, the M.S.E.E. degree from Stanford University, Stanford, CA, USA, in 1970, and the Ph.D. degree from the University of Illinois at Champaign-Urbana, Champaign, IL, USA, in 1973.

He is currently the University Distinguished Professor, Texas Instruments Jack Kilby Chair Professor, and the Director with the Analog and Mixed-Signal Center, Texas A&M University, College Station, TX, USA. He has graduated 61 M.Sc. and 56 Ph.D. students. He is a co-author of six books on different topics, such as RF circuits, low-voltage low-power analog circuits, and neural networks. His current interests include the area of ultra-low power analog circuits, RF circuits, harvesting techniques, power management, and medical electronics circuit design.

He is a former Editor-in-Chief for the IEEE Transactions on Circuits and Systems II and a former IEEE Circuits and Systems Society's Vice President-Publications. He was the recipient of a Honoris Causa Doctorate by the National Institute for Astrophysics, Optics and Electronics, Mexico, in November 1995. This degree was the first honorary degree awarded for microelectronic circuit-design contributions. He was a co-recipient of the 1995 Guillemin-Cauer Award for his work on cellular networks, the recipient of the Texas Senate Proclamation for Outstanding Accomplishments in 1996, a co-recipient of the 1997 Darlington Award for his work on high-frequency filters, and the recipient of the IEEE Circuits and Systems Society Golden Jubilee Medal in 1999 and the prestigious IEEE Circuits and Systems Society 2008 Charles A. Desoer Technical Achievement Award. He was the IEEE Circuits and Systems Society's Representative to the IEEE Solid-State Circuits Society during 2000-2002. He was a Member of the IEEE Solid-State Circuits Society Fellow Award Committee from 2002 to 2004. He is a former (2012-2013) Distinguished Lecturer of the IEEE Circuit and Systems Society and a Guest Editor of the Analog section of the IEEE JSSC special issue of December 2016. He is a Co-Guest Editor of the Special Issue on Circuits and Systems for the Internet of Things-From Sensing to Sense making published in September 2017. He is a Fellow of the Institution of Engineering and Technology, which is the largest multidisciplinary professional engineering institution in the world (http://ece.tamu.edu/∼sanchez/).



Jiang Hu (F'16) received B. S. degree in optical engineering from Zhejiang University, China, in 1990, M.S. degree in physics in 1997, and his Ph.D. degree in electrical engineering from the University of Minnesota in 2001. He was with IBM Microelectronics from January 2001 to June 2002. Currently, he is a professor in the Department of Electrical and Computer Engineering at Texas A&M University. His research interests include EDA, machine learning applications, hardware-algorithm co-design and hardware security. He published over 200 journal articles and conference papers, and received six U.S. patents. He received a best paper award at the ACM/IEEE Design Automation Conference in 2001, an IBM Invention Achievement Award in 2003, a best paper award at the IEEE/ACM International Conference on Computer-Aided Design in 2011 and a best paper award at the IEEE International Conference on Vehicular Electronics and Safety in 2018. He coauthored a paper on microprocessor power management that received HiPEAC Paper Award in 2014. He received an Outstanding Professor Award in the Department of ECE at Texas AM University in 2016. He has served as General Chair and Technical Program Chair for the ACM International Symposium on Physical Design and as associate editor for IEEE Transactions on CAD and ACM Transactions on Design Automation of Electronic Systems. He received the Humboldt Research Fellowship in 2012. He was named an IEEE Fellow in 2016.

Jeyavijayan (JV) Rajendran (S'09, M'15) is an Assistant Professor in the Department of Electrical and Computer Engineering at the Texas A&M University. Previously, he was an Assistant Professor at UT Dallas between 2015 and 2017. He obtained his Ph.D. degree from New York University in August 2015. His research interests include hardware security and computer security. His research has won the NSF CAREER Award in 2017, the ACM SIGDA Outstanding Young Faculty Award in 2019, the ACM SIGDA Outstanding Ph.D. Dissertation Award in 2017, and the Alexander Hessel Award for the Best Ph.D. Dissertation in the Electrical and Computer Engineering Department at NYU in 2016, along with several best student paper awards. He organizes the annual Embedded Security Challenge, a red-team/blue-team hardware security competition and has co-founded Hack@DAC, a student security competition co-located with DAC, and FOSTER. He is a member of IEEE and ACM.