Distributed Logic Encryption: Essential Security Requirements and Low-Overhead Implementation

Raheel Afsharmazayejani, Hossein Sayadi, and Amin Rezaei

Computer Architecture, Reliability, and Security Laboratory (CARS-Lab), California State University Long Beach

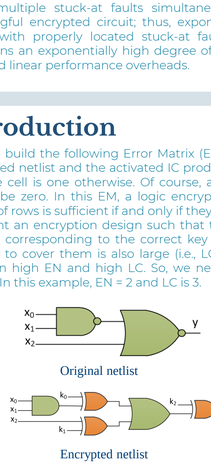
Abstract

Due to outsource manufacturing, the semiconductor industry must deal with various hardware threats such as piracy and overproduction. To prevent illegal electronic products from functioning, the circuit can be encrypted using a protected key only known to the designer. However, an attacker can still decipher the secret key utilizing a functioning circuit bought from the market, and the encrypted layout leaked from an untrusted foundry. In this paper, after introducing essential conformity and mutuality features for secure logic encryption, we propose DLE, a novel Distributed Logic Encryption design that resists against all known oracle guided and structural attacks including the newly proposed fault-aided SAT-based attack that iteratively injects a single stuck-at fault to thwart the locking effect. DLE forces the attacker to insert multiple stuck-at faults simultaneously in critical points to achieve a smaller but meaningful encrypted circuit; thus, exponentially reducing the chance to hit all the critical points with properly located stuck-at fault injections. Our experiments confirm that DLE maintains an exponentially high degree of security under diverse attacks with the polynomial area and linear performance overheads.

Introduction

We can build the following Error Matrix (EM) for any encrypted circuit. A cell is zero when the encrypted netlist and the activated IC produce the same output under each key and input pair, and the cell is one otherwise. Of course, all the associated values for the correct key column should be zero. In this EM, a logic encryption attack is formulated as a covering problem: A subset of rows is sufficient if and only if they cover all the columns with ones.

We want an encryption design such that the number of ones in each column other than the column corresponding to the correct key is large (i.e., EN is high,) while the number of rows needed to cover them is also large (i.e., LC is also high). However, there is a clear contention between high EN and high LC. So, we need to think about how to achieve the best of both worlds. In this example, EN = 2 and LC is 3.



	K_0	K_1	K_2	K_3	K_4	K_5	K_6	K_7
X_0	0	1	1	1	1	0	0	0
X_1	0	1	1	1	1	0	0	0
X_2	0	1	1	1	1	0	0	0
X_3	0	1	0	0	1	0	1	1
X_4	1	1	0	1	0	0	1	0
X_5	1	1	0	1	0	0	1	0
X_6	1	1	0	1	0	0	1	0
X_7	1	1	1	0	0	0	0	1

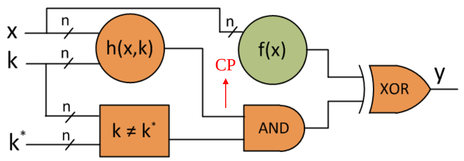
Logic Complexity (LC): The minimum number of DIPs that are required to be checked under the SAT attack to reveal the correct key. The higher the LC of an encryption approach, the more secure it is against the SAT attack [1].

Error Number (EN): The error of a key is the number of input patterns in which there is an inconsistency between the output of the original netlist and the encrypted one. Accordingly, the EN of the whole netlist is the minimum error among all the wrong keys. The higher the EN of an encryption approach, the more secure it is against approximate SAT attack [2].

Structural Complexity (SC): The size of a class of encrypted netlists in which any two netlists are structurally indistinguishable and given any encrypted netlist, structurally separating the original netlist from the locked circuit is exponentially hard with respect to the key size. The higher the SC of an encryption approach, the more secure it is against Functional Analysis Attack (FAA) [3].

Bilateral Logic Encryption (BLE)

Bilateral Logic Encryption (BLE) [4] is proposed to first extract a sub-circuit with high corruptibility and high affectability. A sub-circuit has high corruptibility if its output corruption is sensitizable to the output of the original circuit under many input patterns. A sub-circuit has high affectability if the number of its inputs is comparable to the number of the original circuit inputs. Second, the chosen sub-circuit is locked with a SAT-hard and approximate SAT-hard function h(x,k) with high EN and high LC. Third, to hide the structure of the proposed h-function, a signal routing obfuscation with high SC is utilized on top of the locked circuit. Finally, the sub-circuit is concatenated with the original circuit.

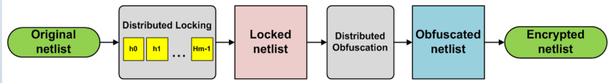


BLE has been shown to be secure against SAT, AppSAT, and FAA. However, utilizing fault-injection, it is possible to identify a Critical Point (CP) in the encrypted netlist of BLE that helps the SAT solver return the correct key in a sub-exponential time..

Fault-aided SAT-based Attack (Fa-SAT)

Fa-SAT [5] iteratively inserts a single stuck-at fault at each signal of the encrypted circuit before feeding the faulty encrypted circuit to the SAT attack [1] framework with a timeout. In Fa-SAT, inserting the fault at an incorrect point can result in reporting a wrong key. Therefore, additional functional verification (implemented by random sampling) is required to check the correctness of the reported key. Since the BLE scheme has a single CP, if a stuck-at-1 fault is inserted in the output of the h-function, Fa-SAT will be able to nullify the security effect of this function and report the correct key.

Distributed Logic Encryption (DLE)

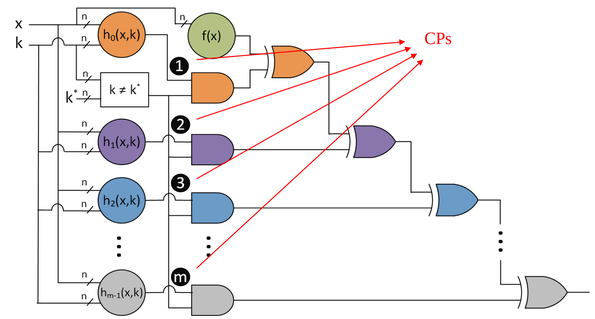


We propose Distributed Logic Encryption (DLE) consisting of joint low-overhead locking and obfuscation with the following requirements: The locked netlist must be secure against SAT [1], AppSAT [2], and Fa-SAT [5], and the obfuscated netlist must be secure against structural attacks like FAA [3]. To thwart Fa-SAT, one naive way is to lock the circuit multiple times each time with a new key. However, in this case, the key size will become very large. Thus, we suggest locking the circuit m times using the same key and a single key comparison component as depicted in this figure. Another naive way to adopt the proposed locking scheme is to repeat a single h-function multiple times. If so, each pair of the h-functions neutralize each other; thus, if Fa-SAT inserts a stuck-at-1 fault on one of the CPs, it still will be able to break the scheme.

Based on our detailed analysis, in order to propose a secure logic locking scheme based on the distributed locking method, we need to choose m distinct h-functions with the following essential security requirements:

Conformity feature:
If we use m h-functions, the distributed locking must have exponentially high LC and exponentially high EN with respect to the input size n.
Tip: In this case, the locking scheme will be secure against SAT and AppSAT.

Mutuality feature:
If we use any group of the $m - 1$ h-functions, an exponential number of columns with respect to the input size n must have zero error in both the EM and the flipped EM of the distributed locking.
Tip: In this case, the locking scheme will be secure against Fa-SAT.



The distributed locking can be divided into four main zones, including h-functions, CPs, original circuit, and final XOR connections. We introduce key controlled OR and AND gates to obfuscate some inter-zone and intra-zone connections with minimal modification to the circuit structure.

Real signal obfuscation: Inside each zone at least one random signal is chosen; if the signal is the fan-in of an AND/NAND gate, a key bit controlled OR gate is added; if it is the fan-in of an OR/NOR gate, a key bit controlled AND gate is inserted.

Dummy signal obfuscation: Inside each zone at least one random signal is chosen; then a random target gate outside the zone is selected; if the target gate is AND/NAND, a key bit controlled OR gate is added; if it is an OR/NOR gate, a key bit controlled AND gate is inserted.

If the target is an AND/ANND gate, to neutralize the dummy signal, the correct value of the corresponding obfuscation key bit should be "1" while for the real signal, the correct value should be "0". The chosen values are opposite if the target is an OR/NOR gate.

Tip: In this case, the obfuscation scheme will be secure against FAA.

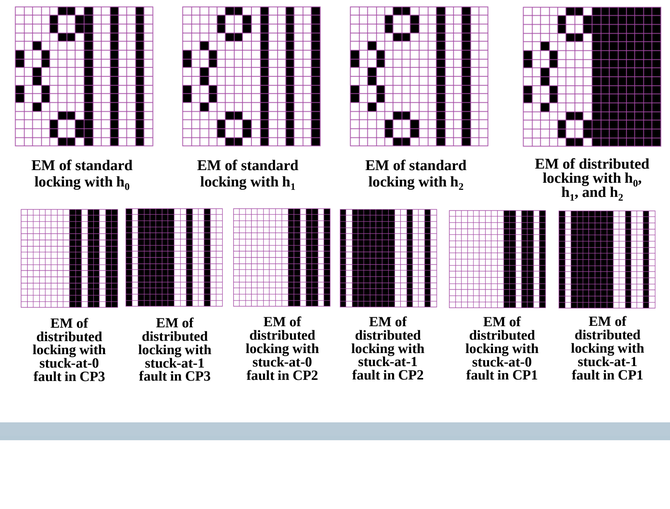
Implementation

To implement the proposed distributed locking scheme with conformity and mutuality features, we suggest the following formula: In this case, the scheme will have $2^{n/2}$ LC and $2^{n/2}$ EN. In addition, there exist at least $2^{n/2}$ wrong keys with zero error upon insertion of a stuck-at-0 or a stuck-at-1 in each CP.

$$\forall\, n = 2k, \qquad \forall\, m = 2k+1, 3 \le m < n,$$
$$\forall\, i \in \{0, 1, \dots, m-1\},$$
$$h_i(x,k) = \bigvee \bigwedge \forall\, \hat{k}_w \in \left\{0, 1, \dots, \frac{n}{2} - 1\right\}, (x_{2j} \oplus k_{2j}) \oplus (x_{2j+1} \oplus k_{2j+1})$$
$$\bigwedge \forall\, \hat{k}_w \in White(i), k \neq \hat{k}_w$$
$$\bigvee \forall\, \hat{k}_w \in Black(i), k = \hat{k}_b$$
$$Black(i) = \{k \mid \forall c \in \left\{0, 1, \dots, \frac{\frac{2^n}{2} - i - 1}{2}\right\}, k = \hat{k}_{i + \frac{2^n}{2} + c \times m}\}$$
$$White(i) = \{k \mid \forall d \in \left\{\frac{2^n}{2}, \dots, 2^n\right\}, k = \hat{k}_d \notin Black(i)\}$$

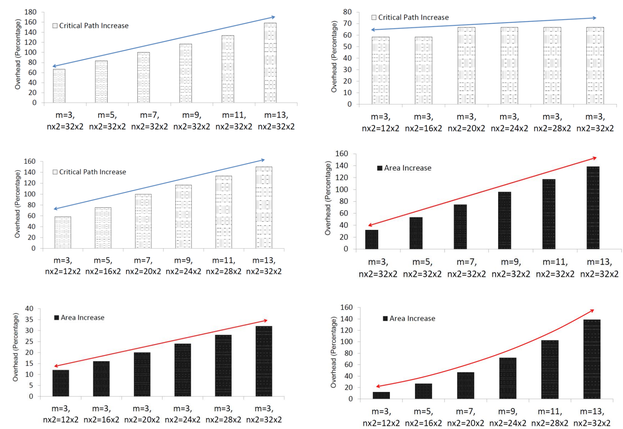
As an example, we consider $n = 4$ and $m = 3$ and define three distinct h-functions: $h_0(x, k)$, $h_1(x, k)$, and $h_2(x, k)$ based on the proposed formula. The EM of this distributed scheme without stuck-at faults is shown on the right-hand side.

Then, we implement the proposed distributed obfuscation with an obfuscation key size equal to the locking key size. In this case, the scheme will have 2^n SC. In other words, by structural analysis of the key bit connections, the zones of DLE are indistinguishable since the key bit controlled AND and OR gates have the same look for real inter-zone and dummy intra-zone signals



Experimental Results

Bench	#Inputs	#Keys	#Gates	SAT attack [1]	AppSAT attack [2]	Fa-SAT attack [5]
apex2	39	38_{X2}	610	No result (24 hours)	Wrong key (262 it.)	Finished w/ no result (432.072 s)
c17	5	4_{X2}	6	Correct Key (4 it., 0.016 s)	No attack	No attack
c432	36	36_{X2}	160	No result (24 hours)	Wrong key (262 it.)	Finished w/ no result (65.552 s)
c499	41	40_{X2}	202	No result (24 hours)	Wrong key (262 it.)	Finished w/ no result (116.352 s)
c880	60	60_{X2}	383	No result (24 hours)	Wrong key (262 it.)	Finished w/ no result (172.35 s)
c1355	41	40_{X2}	546	No result (24 hours)	Wrong key (262 it.)	Finished w/ no result (280.098 s)
c1908	33	32_{X2}	880	No result (24 hours)	Wrong key (262 it.)	Finished w/ no result (212.72 s)
dalu	75	74_{X2}	2298	No result (24 hours)	Wrong key (262 it.)	Finished w/ no result (1089.067 s)
ex5	8	8_{X2}	1055	Correct key (16 it., 0.18 s)	No attack	No attack
i4	192	192_{X2}	338	No result (24 hours)	Wrong key (262 it.)	Wrong key (74.244 s)
i7	199	198_{X2}	1315	No result (24 hours)	Wrong key (262 it.)	Finished w/ no result (5406.73 s)
seq	41	40_{X2}	3519	No result (24 hours)	Wrong key (262 it.)	Finished w/ no result (345.668 s)



Conclusion

In this work, we took a novel perspective on hardware intellectual property protection by proposing a low-overhead and highly secure distributed logic encryption. If both conformity and mutuality features are covered and the SC parameter is defined, none of the existing oracle guided and structural attacks [1, 2, 3, 5] are successful in revealing the correct key and/or the original circuit. The experiments confirmed that by adopting DLE, we can exponentially secure a digital design with only a linear performance overhead and a polynomial area overhead.

References

[1] P. Subramanyan, S. Ray, and S. Malik. 2015. Evaluating the security of logic encryption algorithms. In *International Symposium on Hardware Oriented Security and Trust (HOST)*, 2015.
[2] K. Shamsi, M. Li, T. Meade, Z. Zhao, D. Z. Pan, and Y. Jin, "AppSAT: Approximately deobfuscating integrated circuits," In *International Symposium on Hardware Oriented Security and Trust (HOST)*, 2017.
[3] D. Sirone and P. Subramanyan, "Functional analysis attacks on logic locking," In *IEEE Transactions on Information Forensics and Security*, 2020.
[4] A. Rezaei, Y. Shen, and H. Zhou, "Rescuing logic encryption in post-SAT era by locking & obfuscation," In *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2020.
[5] N. Limaye, S. Patnaik, and O. Sinanoglu. "Fa-SAT: Fault-aided SAT-based attack on compound logic locking techniques," In *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2021.