

Systematic Design and Efficient Automated Implementation of Logic Locking

Introduction

- Globalizing the IC supply chain accelerates production and reduces costs by leveraging a worldwide network. However, it also increases vulnerabilities to various security threats, resulting in billions of dollars in losses.
- Logic Locking** emerged as a potent countermeasure to mitigate these threats. It integrates key-based logic into the target design to obscure it, ensuring proper functionality only by applying the correct secret key (Fig. 1).
- Over the years, logic locking has remained a persistent cat-and-mouse game – defenders counter specific attacks with new locking techniques only to expose vulnerabilities to others due to a lack of security guarantees in the newly introduced locking schemes.

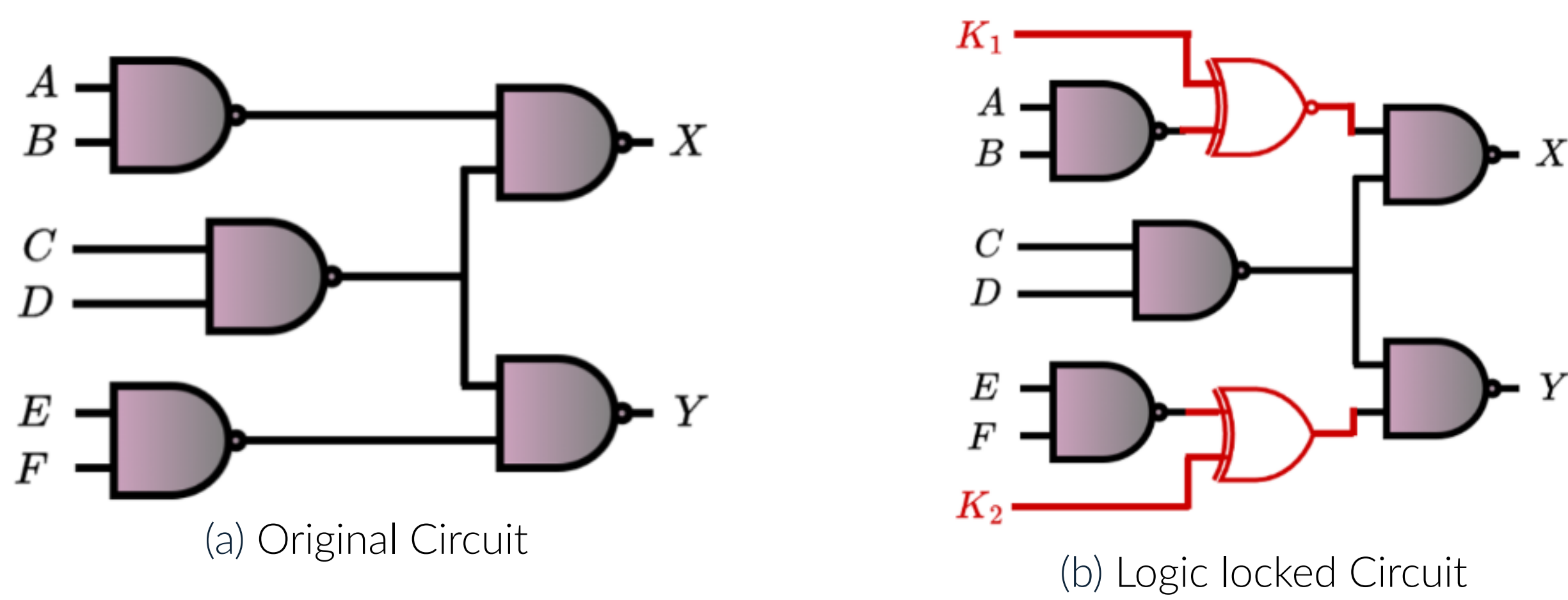


Figure 1. The original circuit is logic locked with key gates (highlighted in red) and operates correctly only when key inputs $K_1 = 1$ and $K_2 = 0$ are applied.

Research objectives

The thesis investigates the following objectives:

- Objective 1: How secure are advanced logic locking techniques?** We successfully attacked an assumed-to-be-secure cellular automata (CA)-based sequential logic locking scheme and a combinational logic locking technique CAS-Lock that claimed to mitigate all existing attacks.
- Objective 2: Is secure logic locking possible?** We introduce a couple of logic locking techniques based on cryptographic security, upon which no known attacks exist.
- Objective 3: Which circuit part should be locked to achieve maximal resilience against logic locking attacks while minimizing incurred overheads?** We introduce MIDAS, an end-to-end CAD framework that automates different genres of logic locking techniques.

Attacks on Logic Locking

- ORACALL attack [3] can compromise both variants of CA-based FSM obfuscation, extracting the secret key and other secret information, such as the CA state encoding for each FSM state.

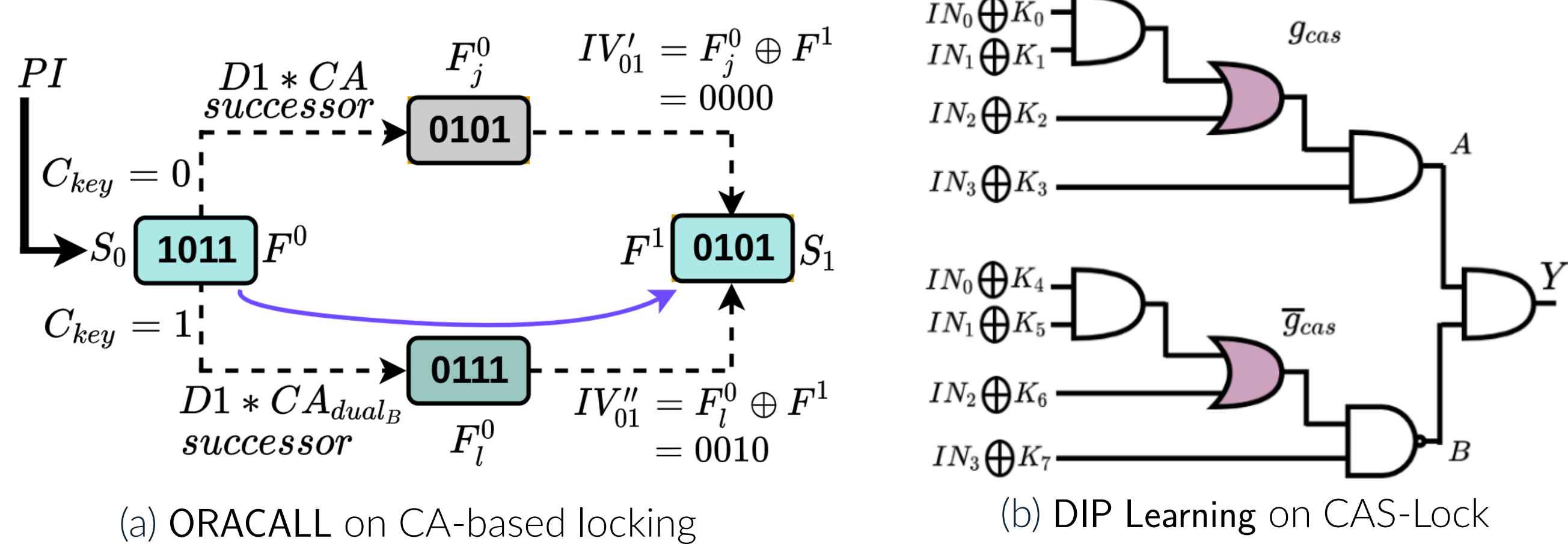


Figure 2. An example of the ORACALL and DIP Learning attacks on CA-based obfuscation and CAS-Lock, respectively.

- DIP Learning** [4] can compromise CAS-Lock [7] by extracting all secret information – including the precise AND/OR chain configuration, the types of key gates (XOR/XNOR) utilized, and the secret key – without relying on structural analysis.

Robust Logic Locking Techniques

We introduce **LoPher**, a novel locking technique that, for the first time, incorporates the cryptographic security of SPN-based block ciphers (e.g., PRESENT, GIFT) into logic locking. It demonstrates that S-Boxes can be configured to implement desired gates and switch boxes by fixing specific input bits.

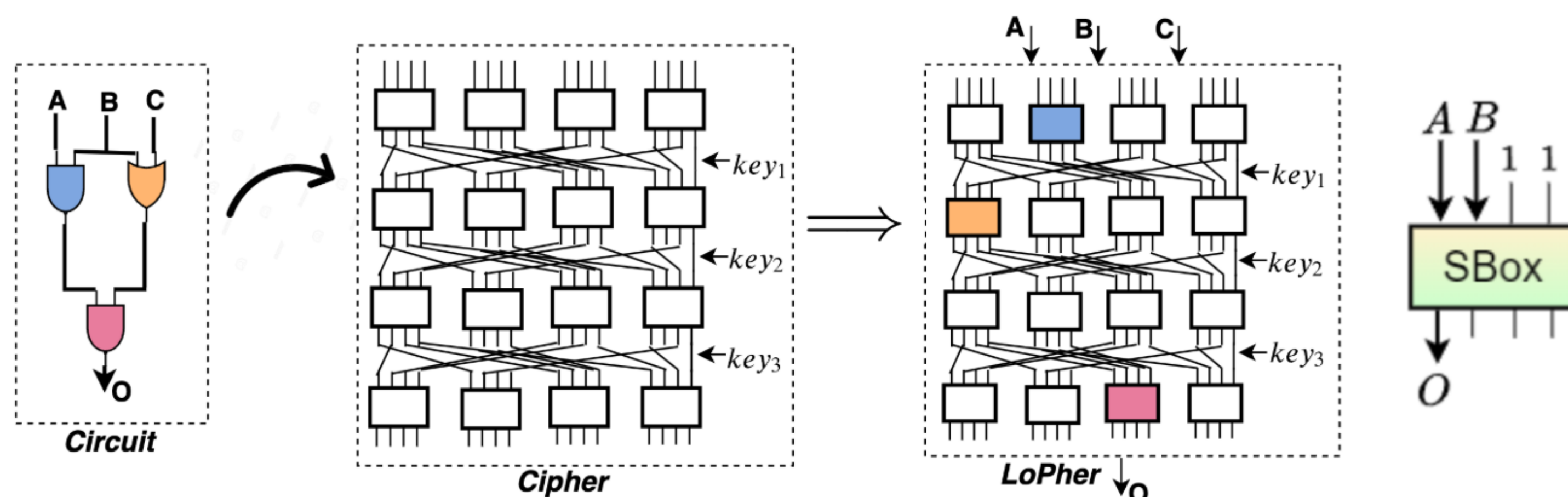


Figure 3. An overview of LoPher locking technique [2]. PRESENT cipher S-Box realizing as NAND gate.

By incorporating **non-linearity** [5] of cryptographic S-Boxes into the inherently linear (the root cause behind ORACALL) CA-based FSM obfuscation, we effectively thwart the attack while preserving the crucial CA properties used for FSM testing.

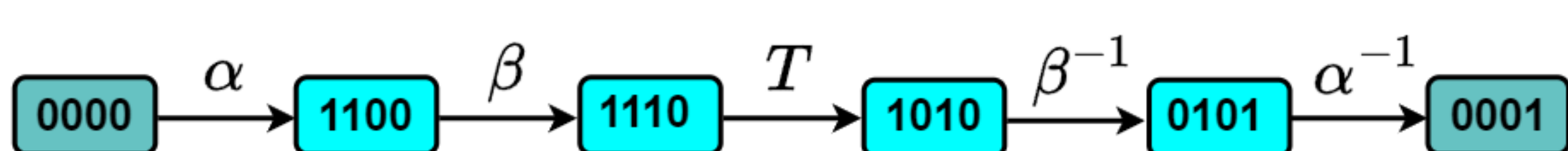


Figure 4. An example of state transition of a 4-cell CA mapping the state 0000 to 0001 in proposed countermeasure with α, β , being two S-Box mappings and characteristic matrix T .

The CAD Framework MIDAS for Automating Logic Locking

We propose an automated framework (Fig. 5) that identifies optimal circuit locations and integrates various logic locking techniques, including pre-SAT, SAT-Hard, and point function-based methods [6].

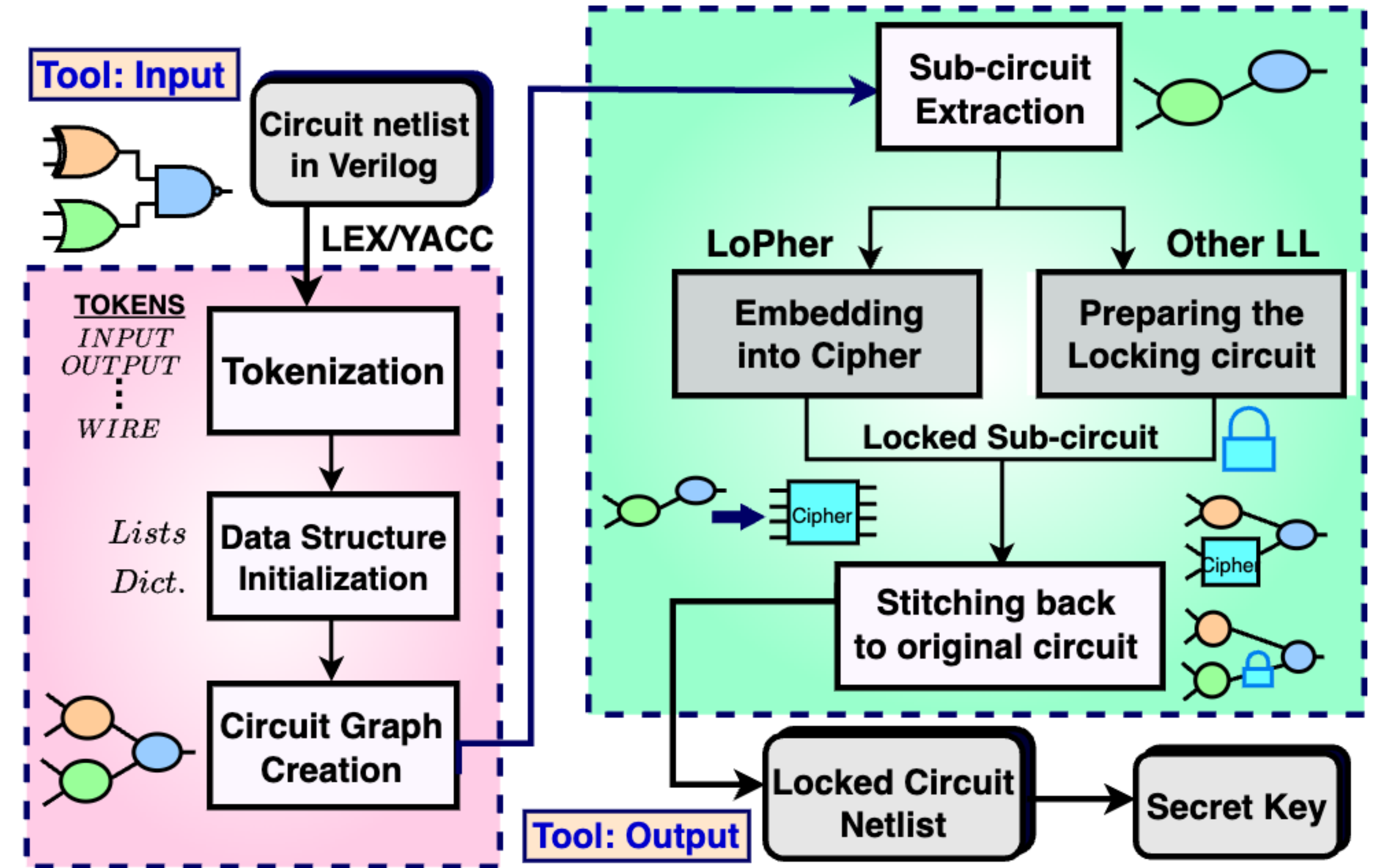


Figure 5. An overview of the proposed MIDAS framework. We utilized MIDAS to lock a RISC-V core.

Results and Discussion

Fig. 6 presents experimental results validating our attacks across various benchmarks. LoPher leverages the inherent security of its underlying block ciphers, making it robust against existing attacks.

ISCAS-85	# I/O	Chain Configuration	DIP L.	MCNC	#States	#Trans.	#Gates	#CA cells	40% thres.
c432	36/7	A-O-2A-O-2A-O-2A-O-2A-O-A	✓	bbara	10	60	178	4	54
c880	60/26	A-O-2A-O-2A-O-2A-O-2A-O-A	✓	donfile	24	96	446	5	121
c2670	233/140	2A-O-5A-O-2A-2O-2A	✓	planet	48	115	383	6	112
c3540	50/22	14A-O	✓	s820	25	232	568	5	229
c5315	178/123	2A-O-5A-O-2A-2O-2A	✓	sand	32	184	1004	6	180
c6288	32/32	3A-2O-3A-2O-3A-O-A	✓	scf	121	286	1264	7	276
c7552	207/108	3A-2O-3A-2O-3A-O-A	✓						

Figure 6. DIP Learning attack results on different CAS-Lock AND/OR chain configurations and ORACALL attack on CA-based obfuscation for various benchmarks.

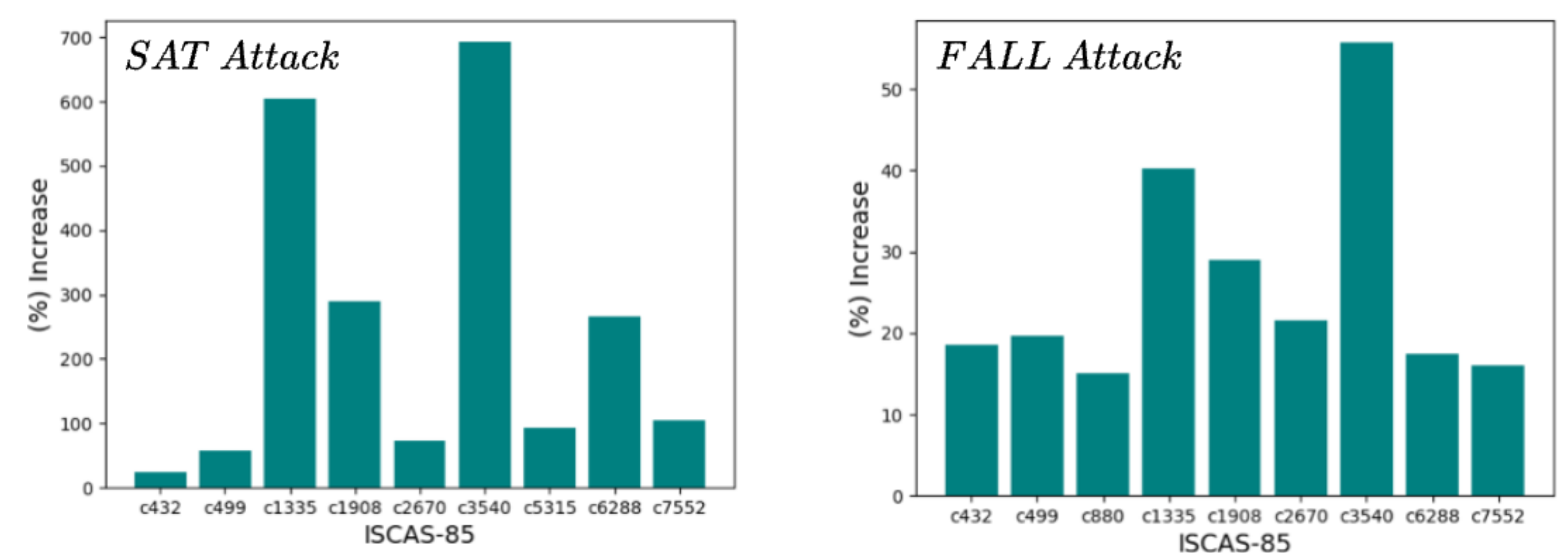


Figure 7. Timing comparison (second) when locked with MIDAS identified locations compared to random insertion.

The non-linearity of the proposed CA obfuscation that thwarts ORACALL is 2.

Conclusions

- ORACALL and DIP Learning effectively breach CA-based FSM obfuscation and CAS-Lock scheme, respectively, without relying on any form of structural analysis of the locked netlist.
- LoPher is the first approach to incorporating block cipher cryptographic security guarantees into logic locking. It also shows that S-Boxes can realize desired logic gates.
- MIDAS identifies critical nodes in a circuit's graph representation and can integrate multiple genres of logic locking techniques, with provision for integrating future locking techniques.

What does this research work add?

- Ad hoc logic locking methods will eventually be broken, leading to a persistent cycle of make and break, which shows the need for established security guarantees in devising logic locking.
- Placing the same logic locking technique at carefully selected locations in a target circuit can achieve greater robustness, a strategy exploited in our proposed framework.

Practical implications

- To the best of our knowledge, no known attacks have been successful on LoPher.
- MIDAS employs six locking techniques from various genres and is designed to incorporate future techniques without altering its core circuit analysis. We utilized MIDAS to lock a RISC-V core, demonstrating its effectiveness.

References

- A. Bogdanov and et al. PRESENT: An ultra-lightweight block cipher. In *CHES*, pages 450–466. Springer, 2007.
- A. Saha and et al. Lopher: Sat-hardened logic embedding on block ciphers. In *DAC*, pages 1–6. IEEE, 2020.
- A. Saha and et al. Oracall: An oracle-based attack on cellular automata guided logic locking. *IEEE TCAD*, 2021.
- A. Saha and et al. Dip learning on cas-lock: using distinguishing input patterns for attacking logic locking. In *DATE*, pages 688–693. IEEE, 2022.
- A. Saha and et al. Revisiting logic obfuscation using cellular automata. In *ASCAT*, pages 27–41. Springer, 2022.
- A. Saha and et al. MIDAS: an end-to-end CAD framework for automating combinational logic locking. *Cryptology ePrint Archive*, Paper 2025/433, 2025.
- B. Shakya and et al. Cas-lock: A security-corrupibility trade-off resilient logic locking scheme. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pages 175–202, 2020.