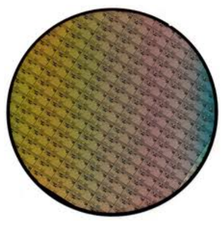
Hiding Instructions in a Processor

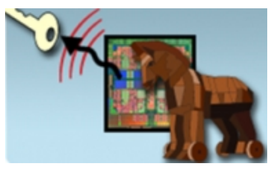
Luciano Brignone, Zhaokun Han, Austin Benedetti, Muhammad Yasin
PI: Jeyavijayan (JV) Rajendran
Department of Electrical and Computer Engineering, Texas A&M University
E-mail: {luciano.brignone, hzhk0618, austinben, myasin, jv.rajendran}@tamu.edu

IC Problem

ICs are prone to different types of attacks



IP/IC Piracy & overproduction



Hardware trojans

Counterfeiting

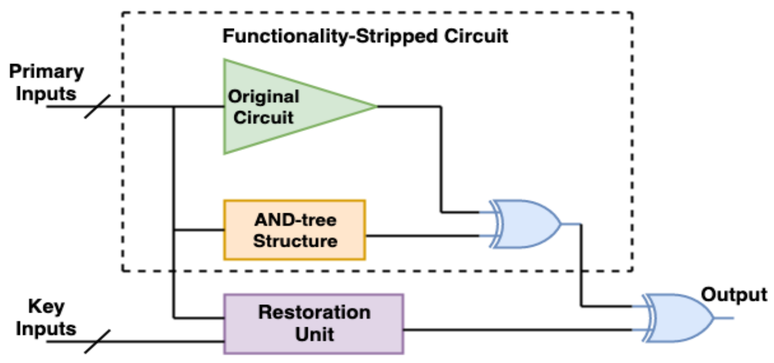
Threat Model

- The attacker is assumed to be in foundry having access to the chip layout and process details (reverse engineering)
- He/She has access to a functional chip (oracle)

Prior art and challenges

State of the art logic locking

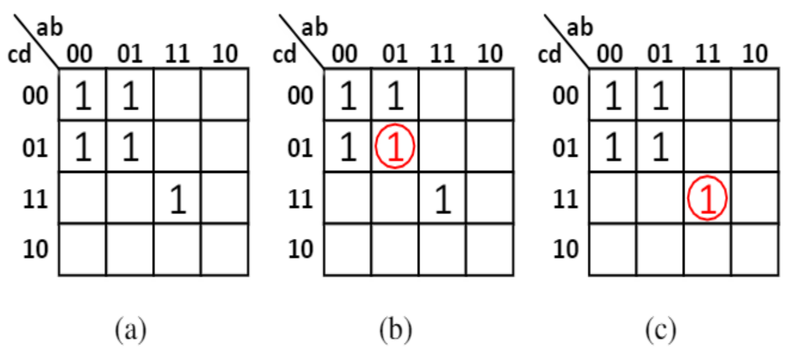
- Based on “strip and restore” concept
- Thwarts all I/O query attacks on logic locking
  - SAT, approximate, removal
- Easily integrates with IC design flow



SFLL circuit

Drawbacks of SFLL

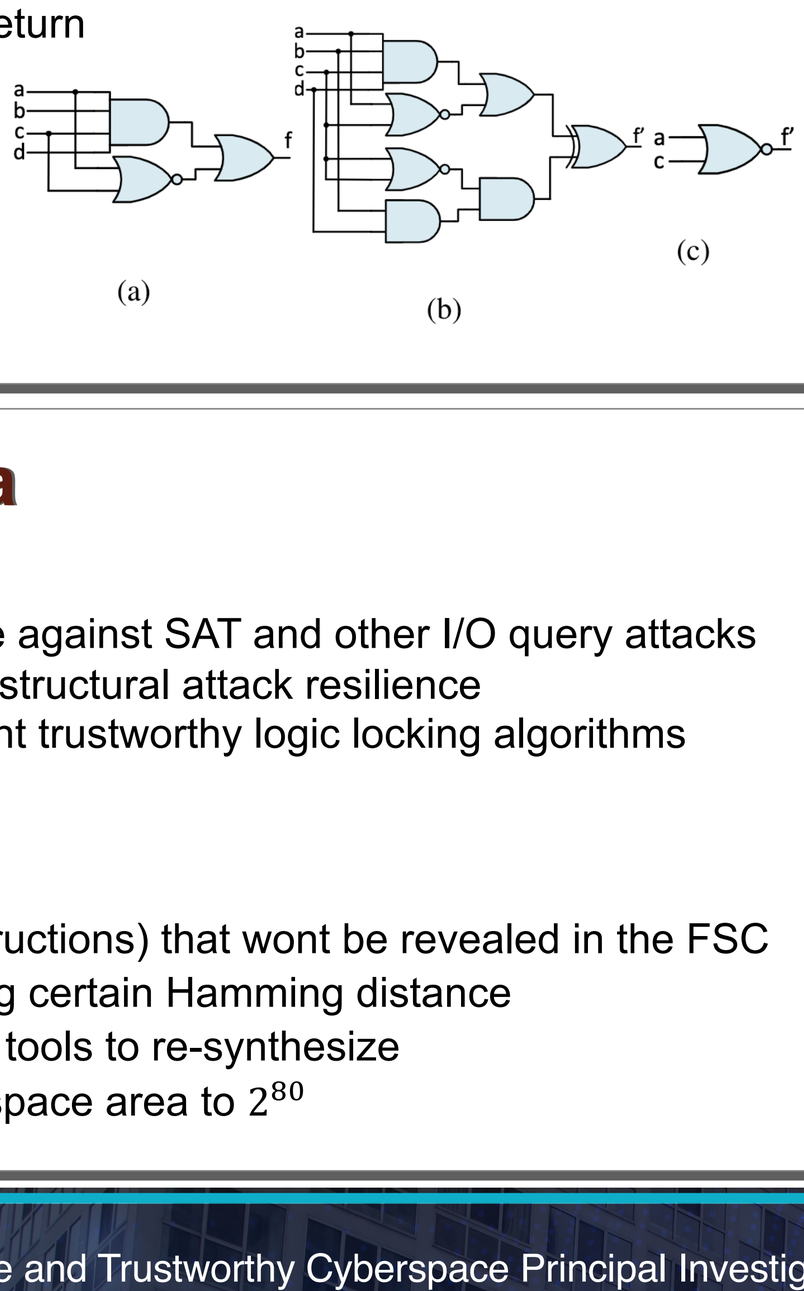
- Stripped Functionality Logic Locking (SFLL) technique offers no secure method to construct the functionality-stripped circuit (FSC)
  - FALL attack
  - AND-tree attack



a) Original function is
$f = \overline{ac} + abcd$

b) For input 0101 to protect, synthesis tools return
$f = (\overline{ac} + abcd) \oplus \bar{a}\bar{b}\bar{c}\bar{d}$

c) For input 1111, we get an FSC of
$f = \overline{ac}$



Proposed idea

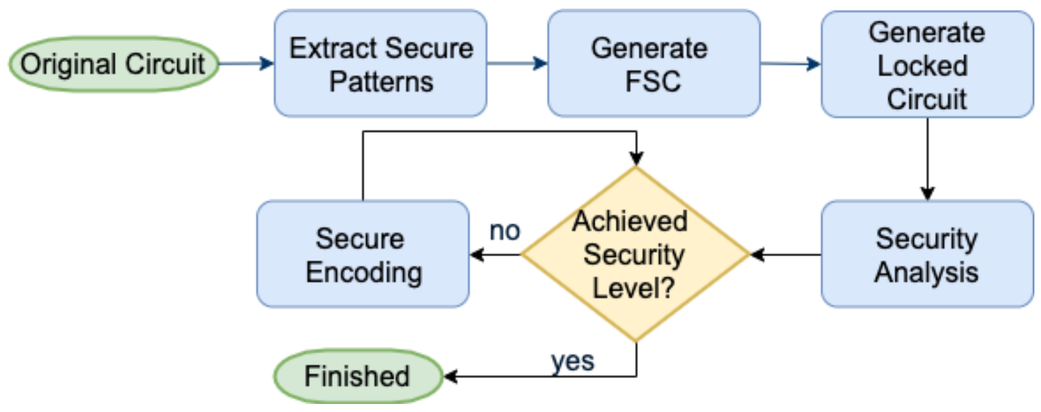
Desired goals

- Retain SFLL resilience against SAT and other I/O query attacks
- Develop our theory of structural attack resilience
- Develop and implement trustworthy logic locking algorithms

Idea in a nutshell

- Protect patterns (instructions) that wont be revealed in the FSC
- Achieved by satisfying certain Hamming distance
- Force logic synthesis tools to re-synthesize
- Increase the search space area to 2^{80}

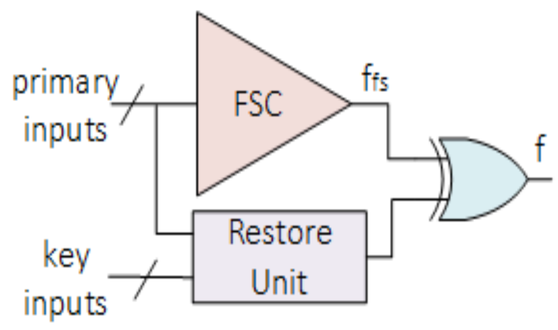
Proposed approach



Flowchart if secure logic locking algorithm

Locked circuit has:

- FSC generated using our secure logic locking algo.
- Restore unit composed of a comparator
- Key is stored in tamper-proof memory

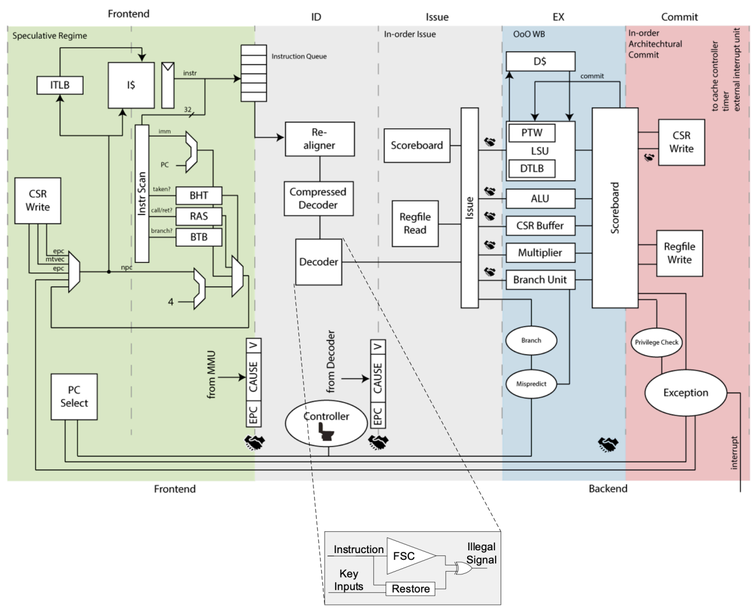


Circuit on our approach

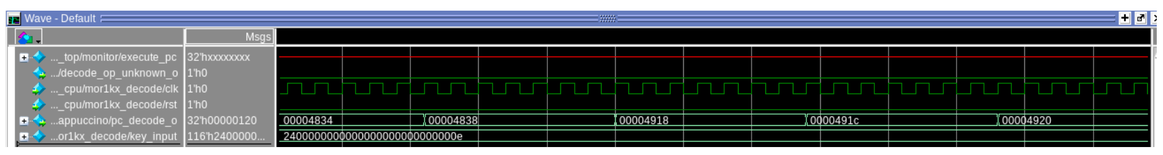
Results on mor1kx processor

Attack	SFLL	Our Approach
SAT	✔	✔
AND-tree Removal	✘	✔
FALL	✘	✔

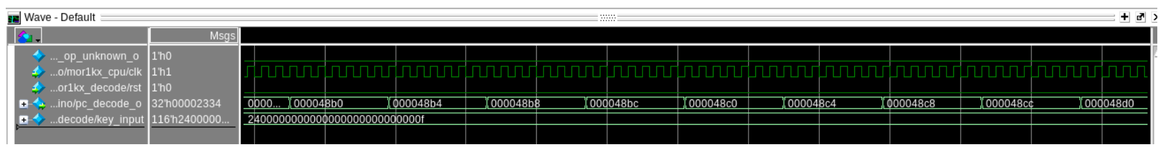
Security analysis



Locked processor



Simulation with incorrect key



Simulation with correct key

References

[1] M. Yasin et al., ”Provably Secure Logic Locking: From Theory to Practice”, CCS’17
[2] P. Subramanyan et al., “Evaluating the Security of Logic Encryption Algorithms”, HOST’15
[3] J.A. Roy et al. ,“EPIC: Ending Piracy of Integrated Circuits” DATE’08.
[4] M. Yasin et al., "On Improving the Security of Logic Locking”, TCAD’15
[5] M. Yasin et al., “Removal Attacks on Logic Locking and Camouflaging Techniques”, TETC’17
[6] K. Shamsi et al, “AppSAT: Approximately Deobfuscating Integrated Circuits”, HOST’17
[7] J. Rajendran at al, “Security Analysis of Logic Obfuscation”, DAC’12