A TUTORIAL ON DESIGN OBFUSCATION:
FROM TRANSISTORS TO SYSTEMS

Prof. Samuel Pagliarini
Centre for HW Security - Dpt. of Computer Systems - School of IT
Tallinn University of Technology

OUTLINE

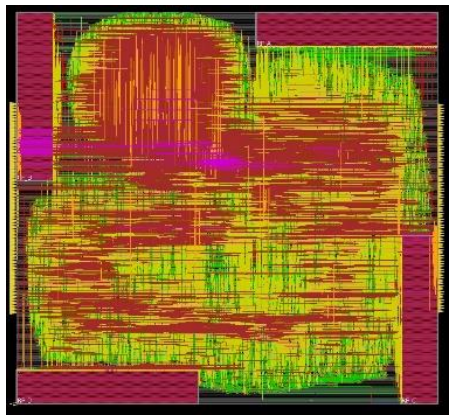
- **What is Design Obfuscation?**
- **Layout-based solutions**
  - Look alike cells
- **Locking techniques**
  - Logic locking
- **Macro approaches**
  - Split fabrication

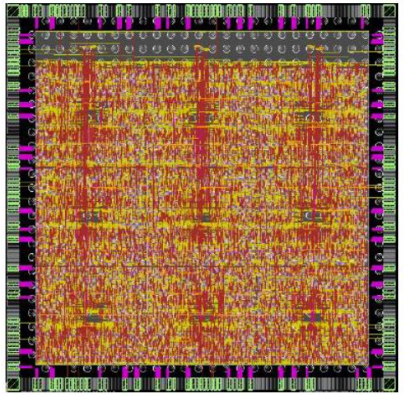
TAL TECH

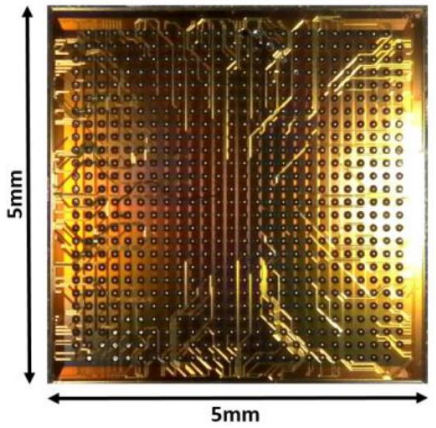
Disclaimers

- This talk focuses on solutions for obfuscating digital circuits
  - Solutions for analog do exist but are slightly less mature
- No prior knowledge on hardware security is required
  - But, there more you know about **IC design**, the more you will get out of this talk

- My own background
  - IC designer with dozens of tapeouts in 10+ different technologies
  - Worked on every node from 16nm to 650nm
  - Experience taping out at least a dozen different obfuscation techniques



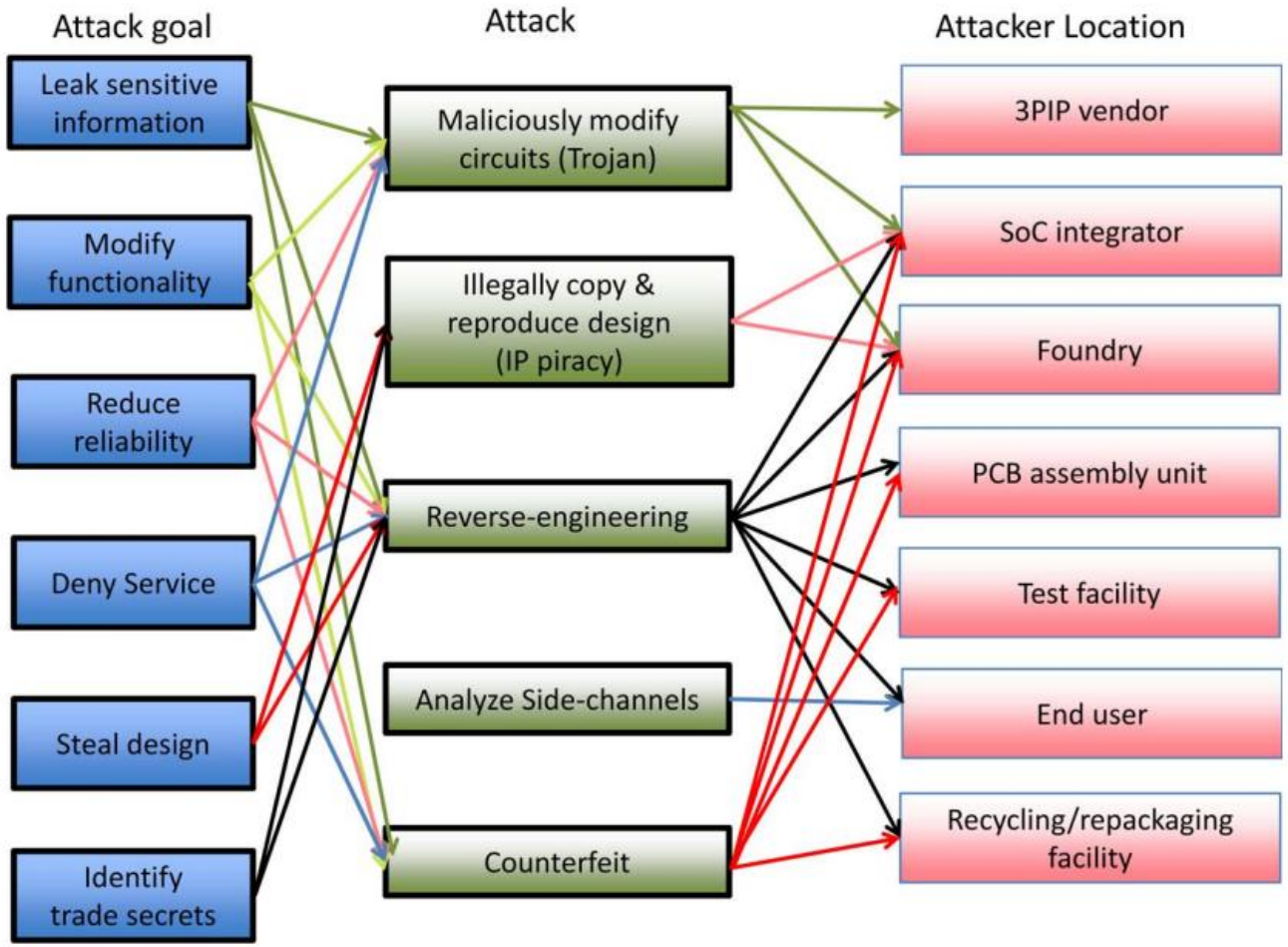




TAL TECH

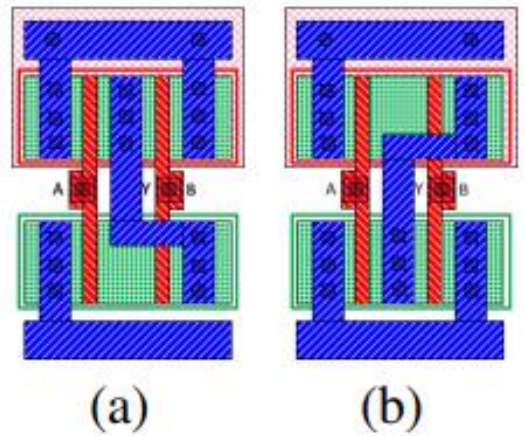
Obfuscation and a map of goals/attacks/attackers

- **To obfuscate is to create confusion**
- Obfuscation can stop or discourage several scenarios



Masoud Rostami, Farinaz Koushanfar, and Ramesh Karri: A Primer on Hardware Security: Models, Methods, and Metrics

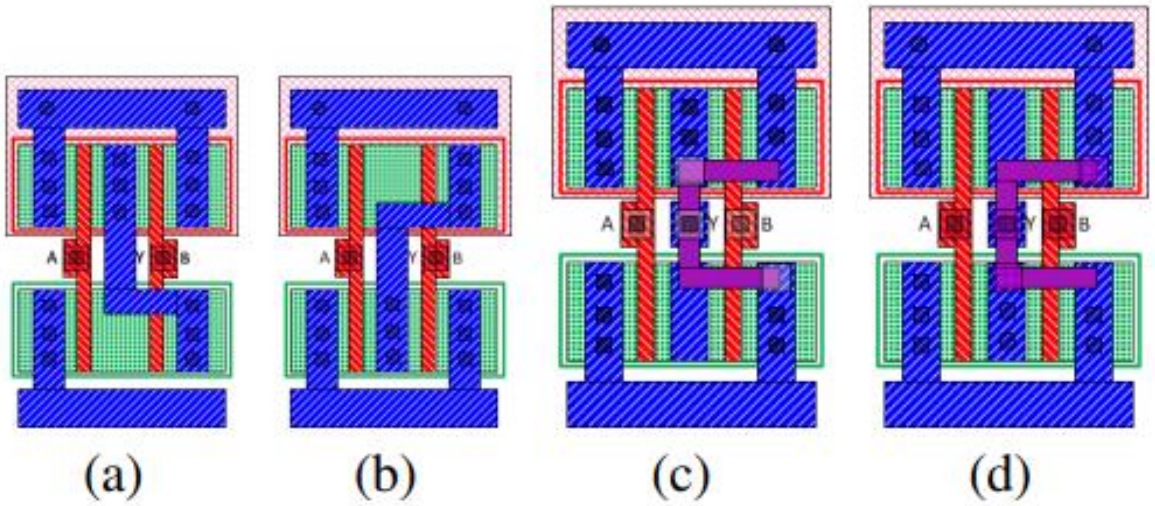
Layout-based solutions: look-alike cells



- (a) is a NAND
- (b) is a NOR
- From a top view, just looking at the blue lines (M1), one can tell a NAND from a NOR
- Concept: make standard cells look alike
- How: push the cell-defining characteristics to contacts (and vias), making the top level view patterns identical

J. Rajendran et al., "Security Analysis of Integrated Circuit Camouflaging," ACM SIGSAC'13.

Layout-based solutions: look-alike cells



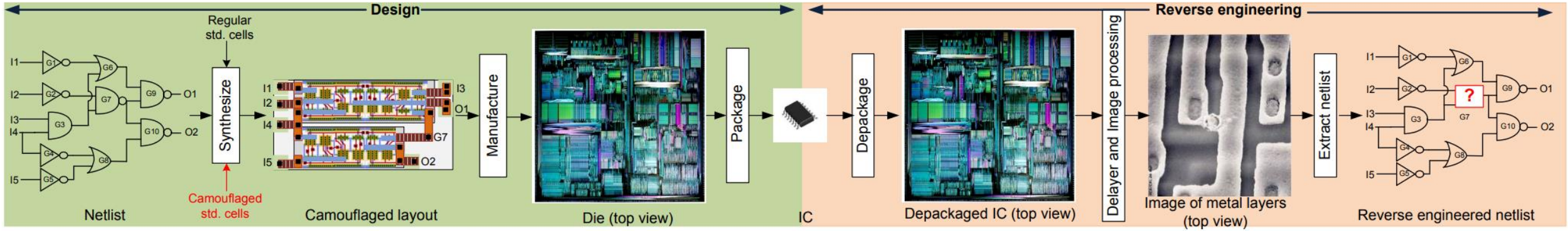
- (a) is a NAND
- (b) is a NOR
- From a top view, just looking at the blue lines (M1), one **CANNOT** tell NAND from NOR
- Concept: make standard cells look alike
- How: push the cell-defining characteristics to contacts (and vias), making the top level view patterns identical

- Notice the increase in size…
- Notice the use of M2 (purple)

J. Rajendran et al., "Security Analysis of Integrated Circuit Camouflaging," ACM SIGSAC'13.

Layout-based solutions: look-alike cells

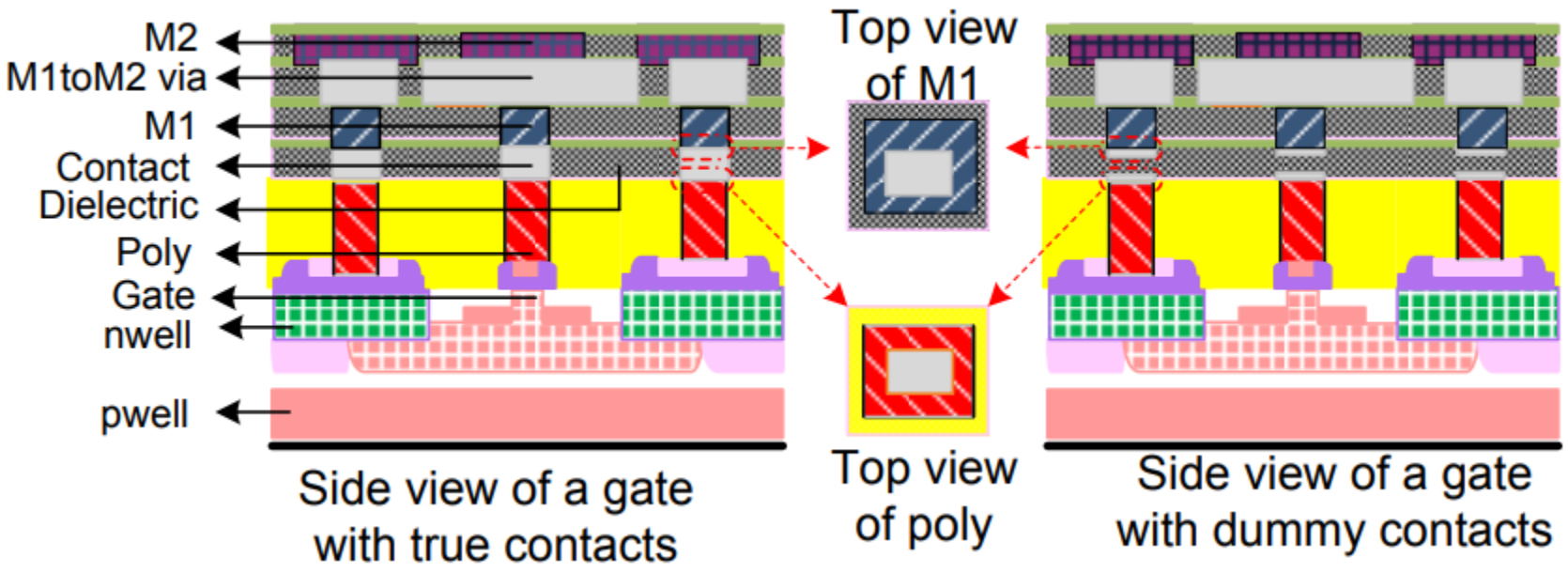
- Design flow and attack



J. Rajendran et al., "Security Analysis of Integrated Circuit Camouflaging," ACM SIGSAC'13.

Layout-based solutions: dummy via/contact

- Previous solution has some merits
- Adversary can distinguish NAND from NOR by looking at contacts
- Can we create dummy contacts/vias?



J. Rajendran et al., "Security Analysis of Integrated Circuit Camouflaging," ACM SIGSAC'13.

Layout-based solutions: dummy via/contact

- Does it work? Answer is most likely no
- Reason: dual damascene technology, invented in the late 90s, most likely to be used in all nodes from ~180nm to ~2nm

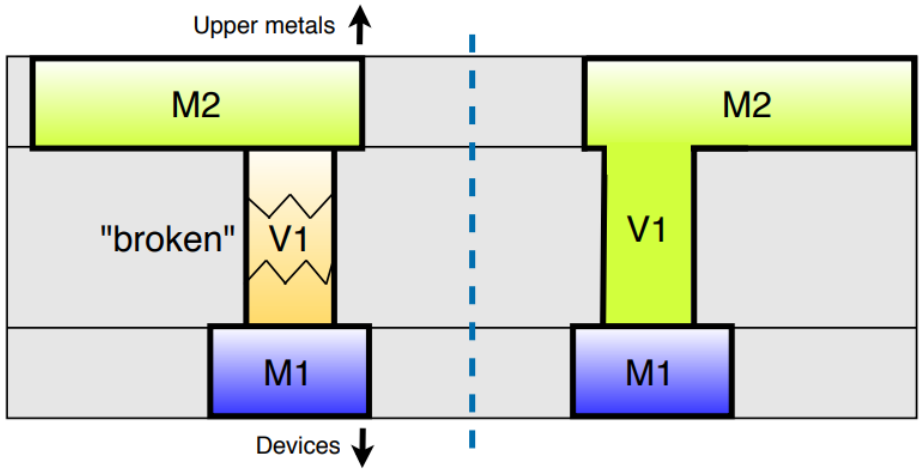
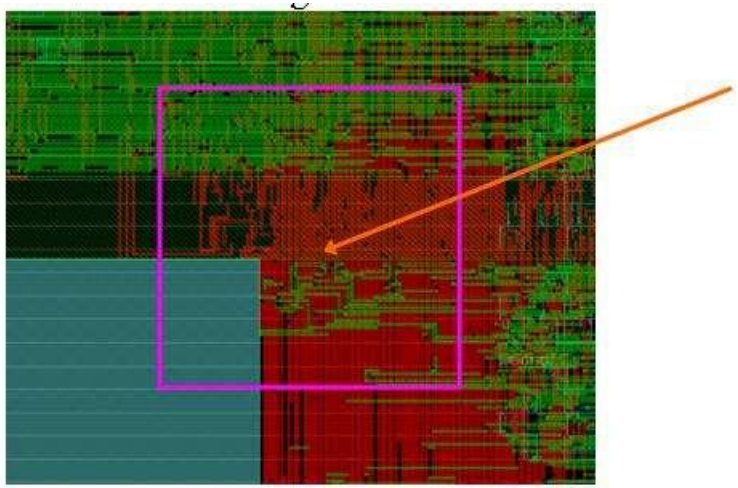
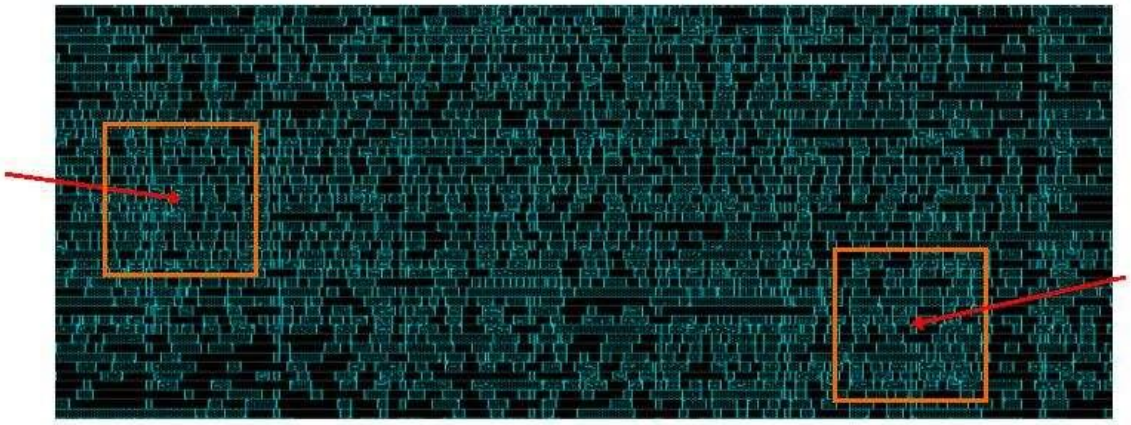


Fig. 1. Via and metal deposition processes. On the left, conventional deposition where vias and metals are formed one by one. On the right, the dual damascene process is highlighted where metals and vias are deposited at the same time.

Layout-based solutions: white space filling

- Layouts have empty spaces
  - 50% density is not unheard of in complex SoC
  - Function of floorplan/powerplan decisions
  - Function of pin count vs. routing resources

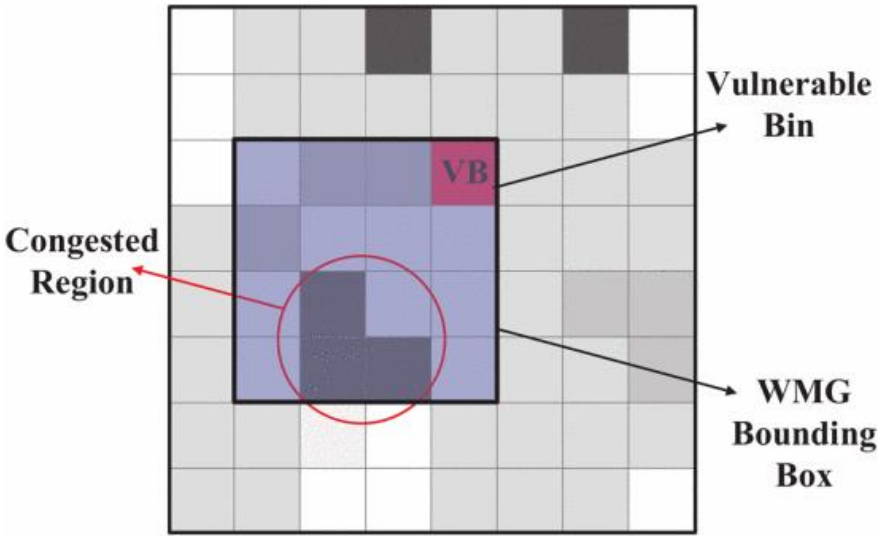


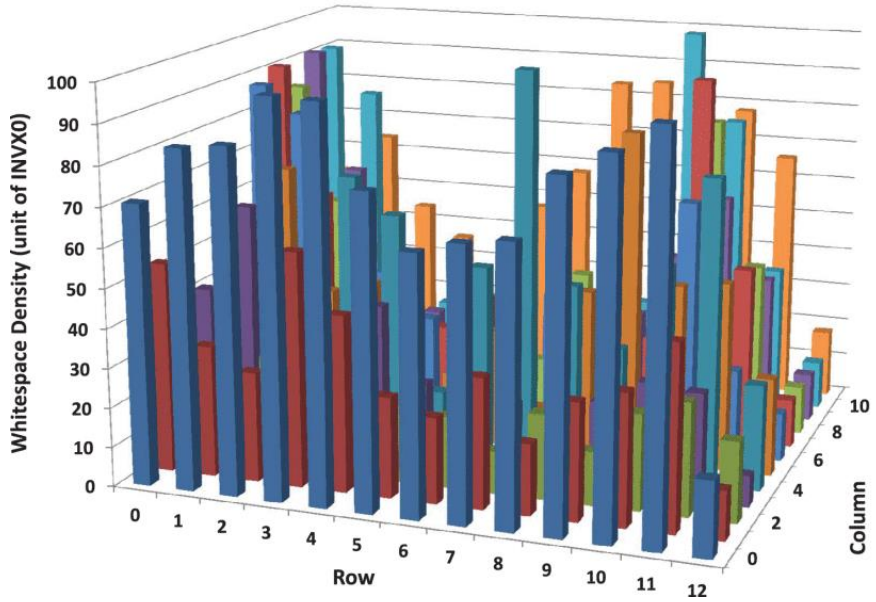


Source: https://www.edn.com/combating-congestion-in-high-performance-low-cost-systems-on-chip/

Layout-based solutions: white space filling

- Hypothesis: trojans can be inserted in the empty spaces of a layout
- "Obfuscation" could be used to prevent this

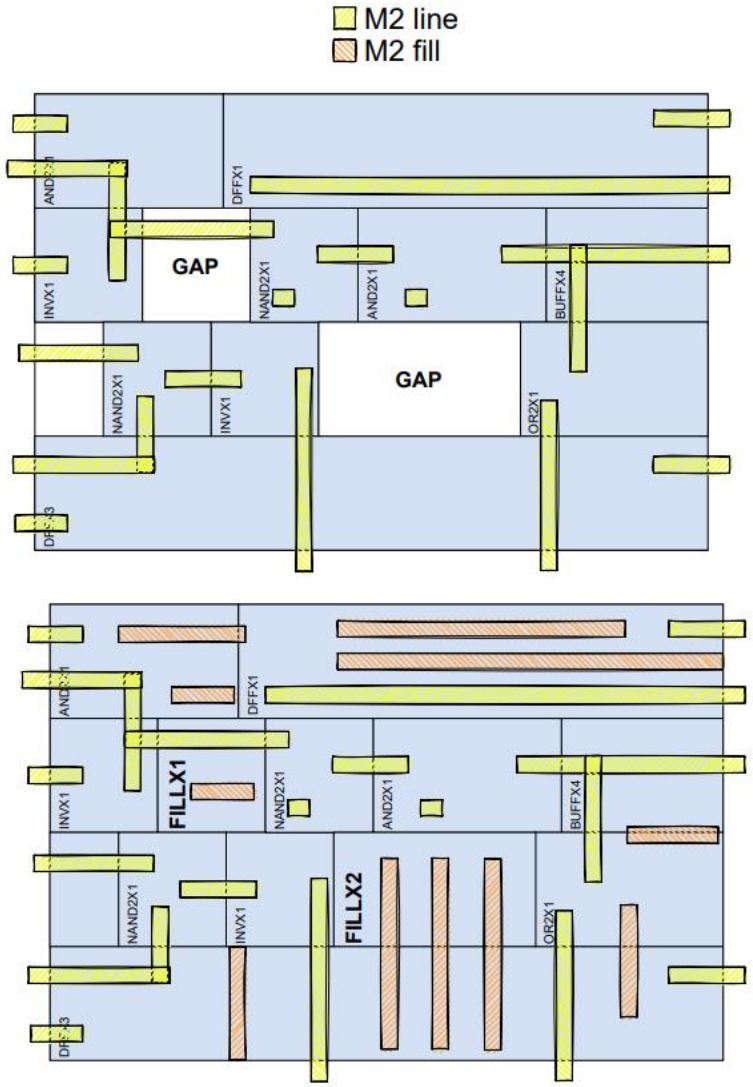




H. Hossein-Talaee and A. Jahanian, "Layout vulnerability reduction against trojan insertion using security-aware white space distribution," ISVLSI 2017

H. Salmani and M. M. Tehranipoor, "Vulnerability Analysis of a Circuit Layout to Hardware Trojan Insertion," in *IEEE Transactions on Information Forensics and Security*, 2016

Layout-based solutions: white space filling



- Many inconsistencies
- Increasing density?
  - At the expense of performance/area?
- Filling what is already filled?
  - Filler, decaps, ECO cells
  - Complex metal fill patterns

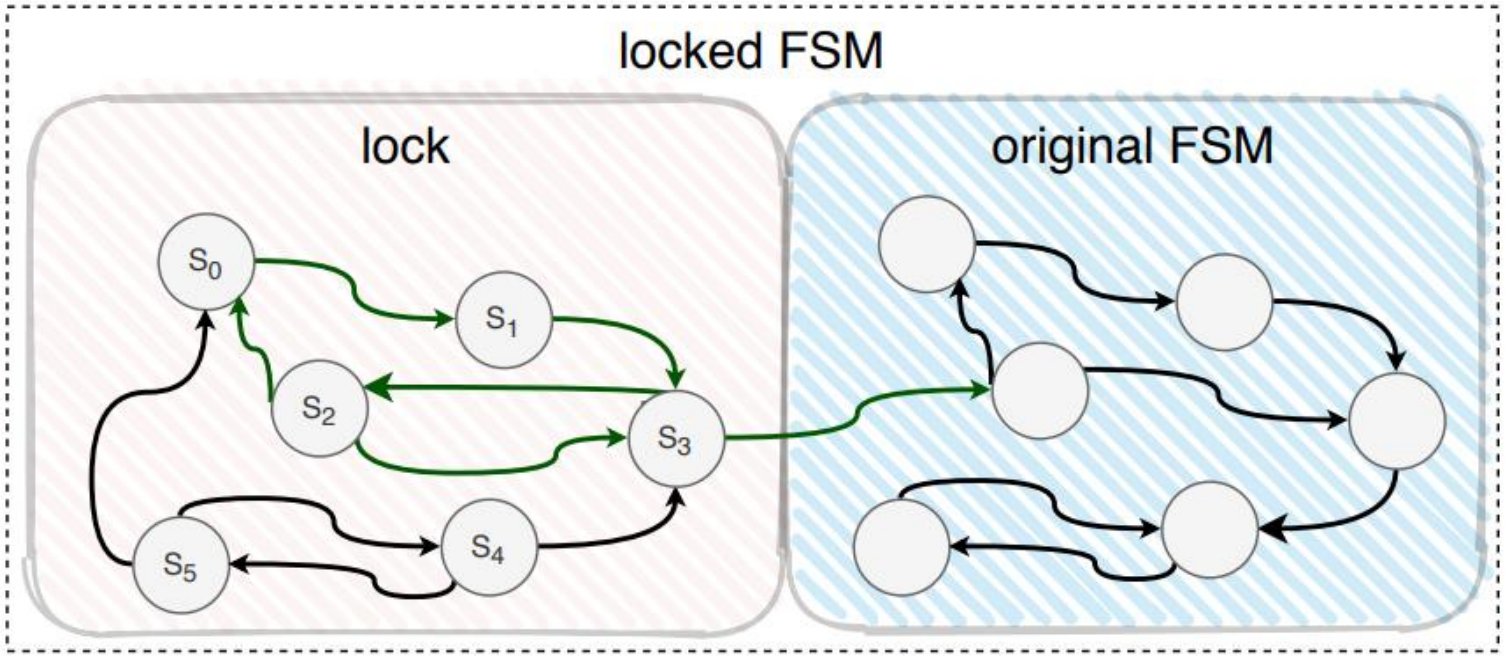
Overview

Class of technique	Effectiveness	Tooling support	Future adoption?
Layout-based	**Very limited ☹**	**Non-existent ☹**	**Unlikely ☹**
Locking			
Macro approaches			

TAL TECH

Locking-based solutions

- Concept: original FSM is protected by adding more states and transitions
  - These states/transitions depend on keyed inputs
  - Keys are a secret, not shared at fabrication time
  - Protects against IP theft, overproduction



Follow the green arrows: S0-S1-S3-S2-S3

Locking-based solutions

- **Logic locking** is the combinational counterpart of sequential locking
- Minimal example:

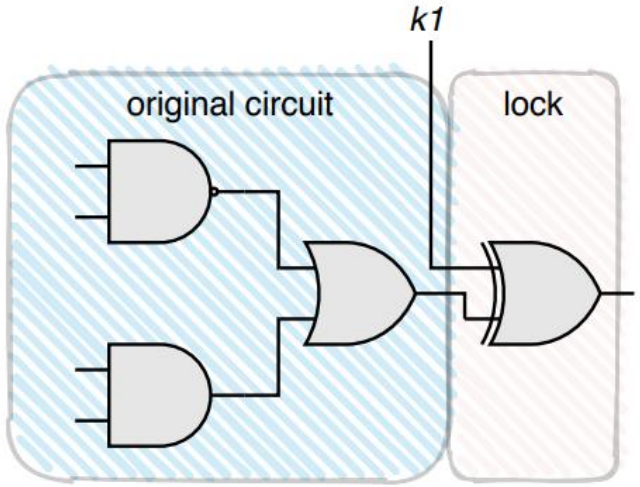


Fig. 3. Example of a locked circuit: the lock is the XOR gate controlled by key *k1*. The circuit behaves as expected when *k1=0*.

- One way to look at this technique is to consider the key gate as a bit flipper

Locking-based solutions

- **Logic locking** is the combinational counterpart of sequential locking
- Minimal example:

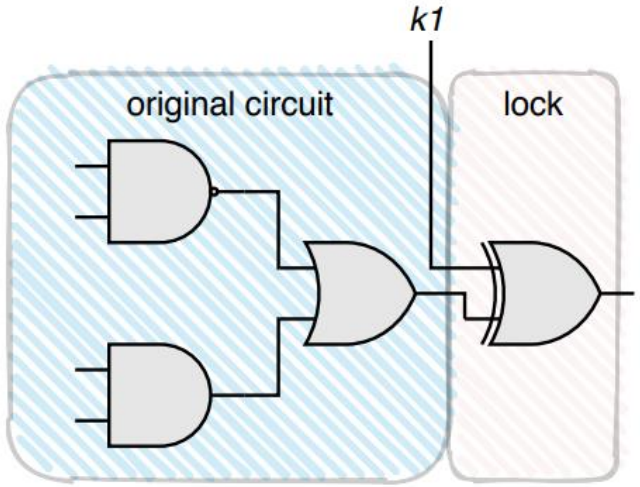
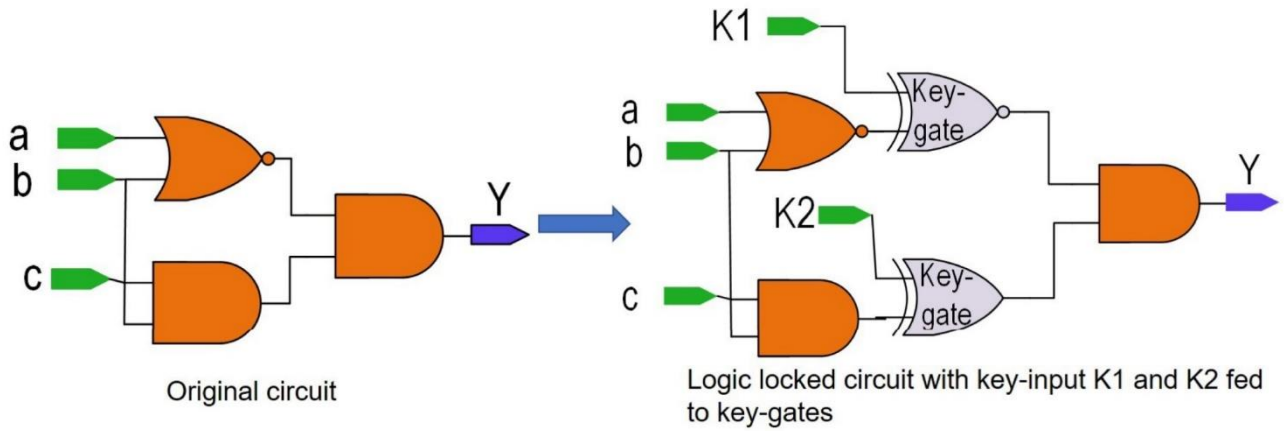


Fig. 3. Example of a locked circuit: the lock is the XOR gate controlled by key *k1*. The circuit behaves as expected when *k1=0*.

i	k1	*i* XOR *k1*
i	0	**i**
i	1	**!i**

- One way to look at this technique is to consider the key gate as a bit flipper
- Let's look at another example…

Locking-based solutions



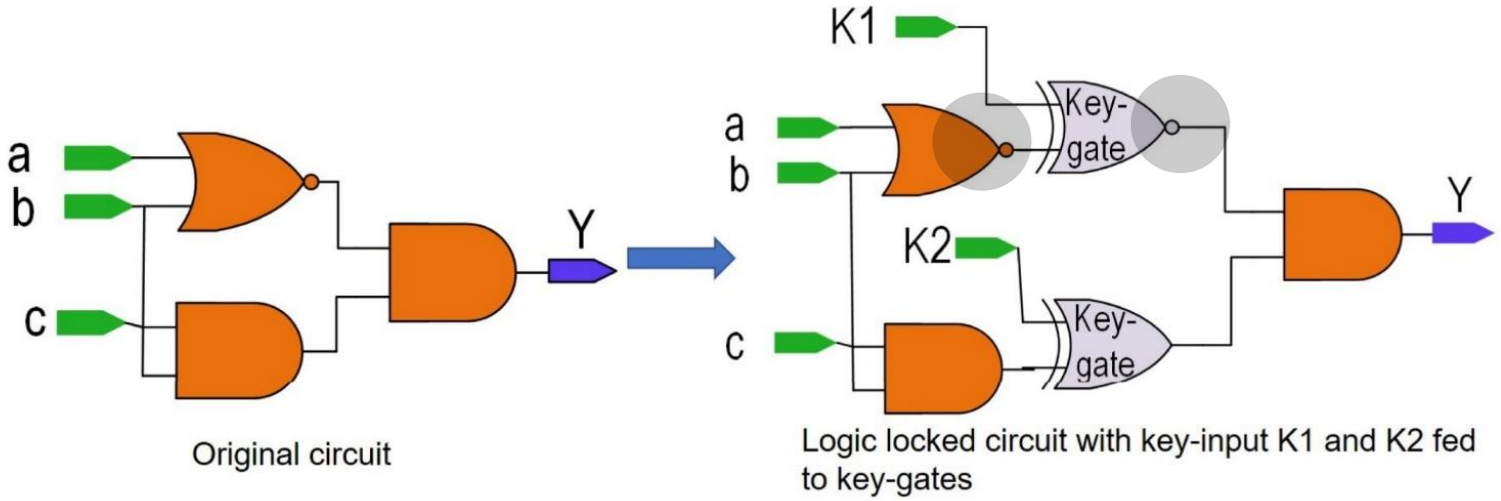
i	j	i xor j	i xnor j
0	0	**0**	**1**
0	1	**1**	**0**
1	0	**1**	**0**
1	1	**0**	**1**

- Can you determine the correct key bits just by looking at the netlist?
  - K1 = ?
  - K2 = ?

M. Tanjidur Rahman et al., "Defense-in-depth: A recipe for logic locking to prevail". Integration vol. 72, May 2020.

Locking-based solutions

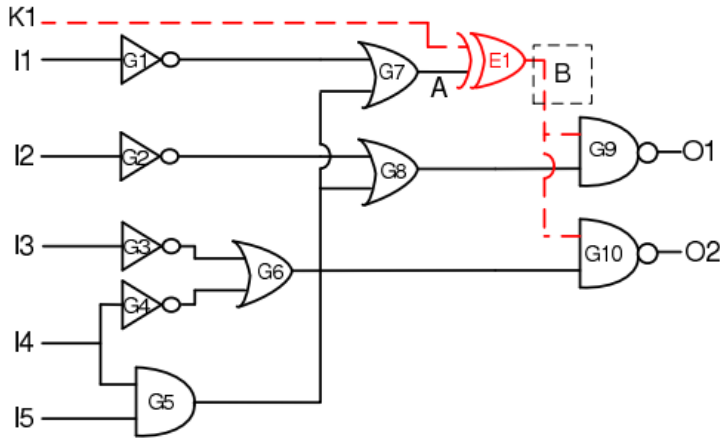
- Resynthesis solves the problem
  - XOR and XNORS merge with the design
  - Inverters are pushed forward/backward

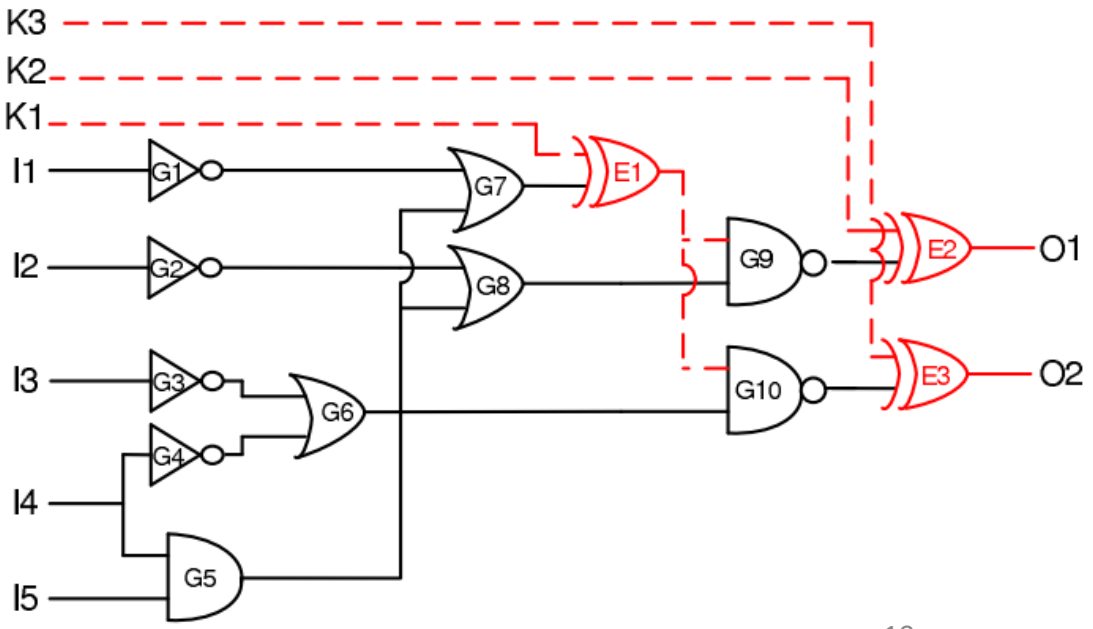


J. Rajendran et al., “Logic encryption: a fault analysis perspective,” DATE '12

Locking-based solutions

- How about location?
  - K1 = 1 (wrong key)
  - Corruption when input = 00000
  - No corruption when input = 01110
- How about runs of key gates?
  - Masking (K1=K2=K3=1)
- **Controllability/observability** plays a role
  - Solutions have been found

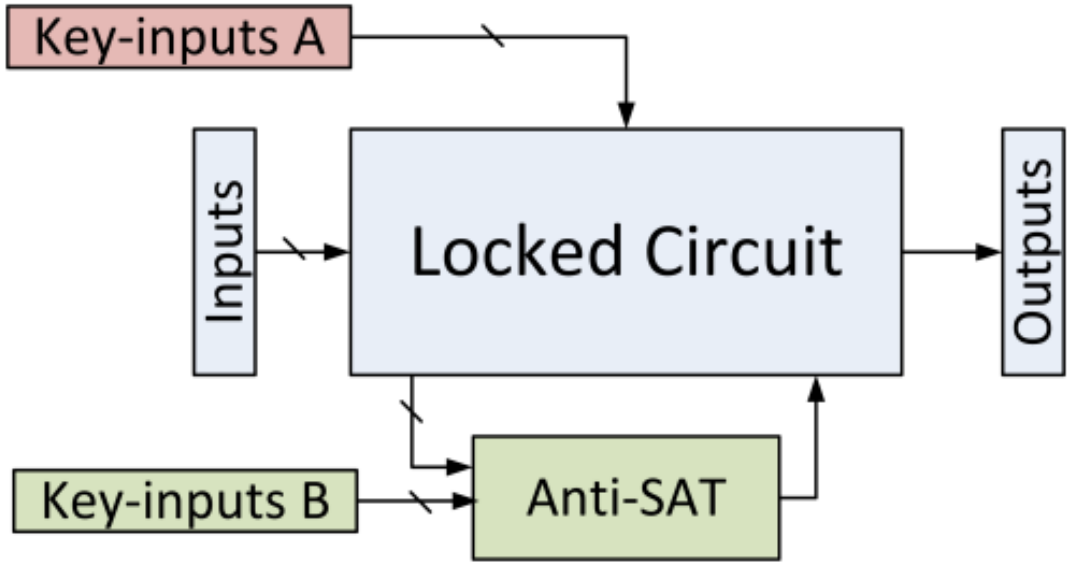




J. Rajendran et al., “Logic encryption: a fault analysis perspective,” DATE '12

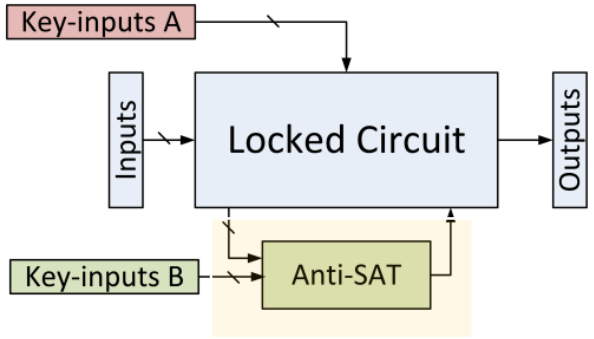
Locking-based solutions

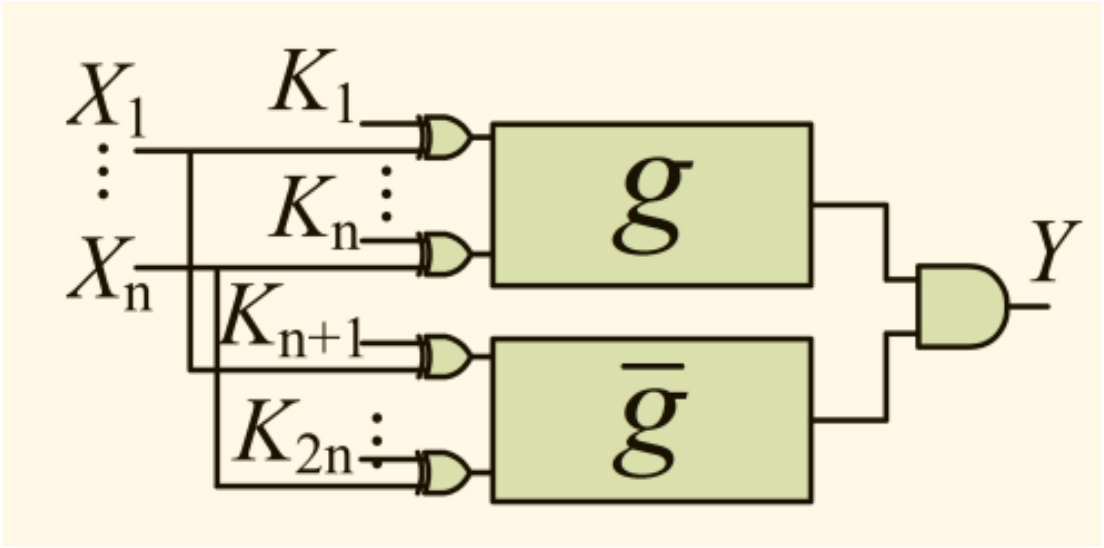
- SAT solver: (not covered) broke all known logic locking attacks in 2015
- Let's look at Anti-SAT



Y. Xie and A. Srivastava, "Anti-SAT: Mitigating SAT Attack on Logic Locking," in *IEEE TCAD*, Feb. 2019

Locking-based solutions





- *n* signals from the original circuit, *2n* key gates
  - Connection of internal signals to keys using individual XORs
- Complementary functions (or blocks) *g/gbar* that must produce Y=0 only when all key bits have their correct values
- Difficulty in mapping Inputs to Outputs = SAT resilience

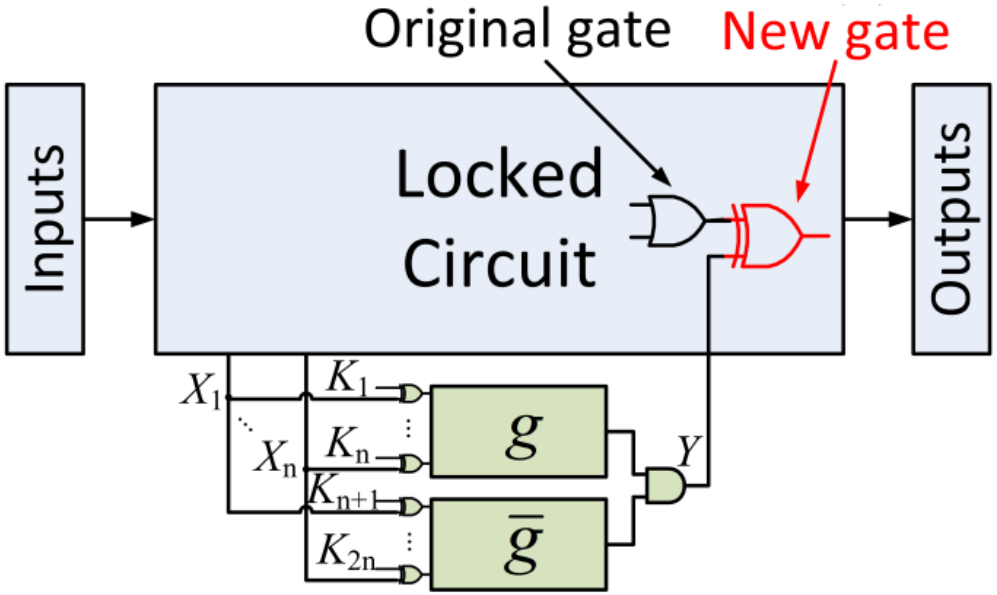
A	B	C	Z
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

TAL TECH

Y. Xie and A. Srivastava, "Anti-SAT: Mitigating SAT Attack on Logic Locking," in *IEEE TCAD*, Feb. 2019

Locking-based solutions

- Let's look at Anti-SAT

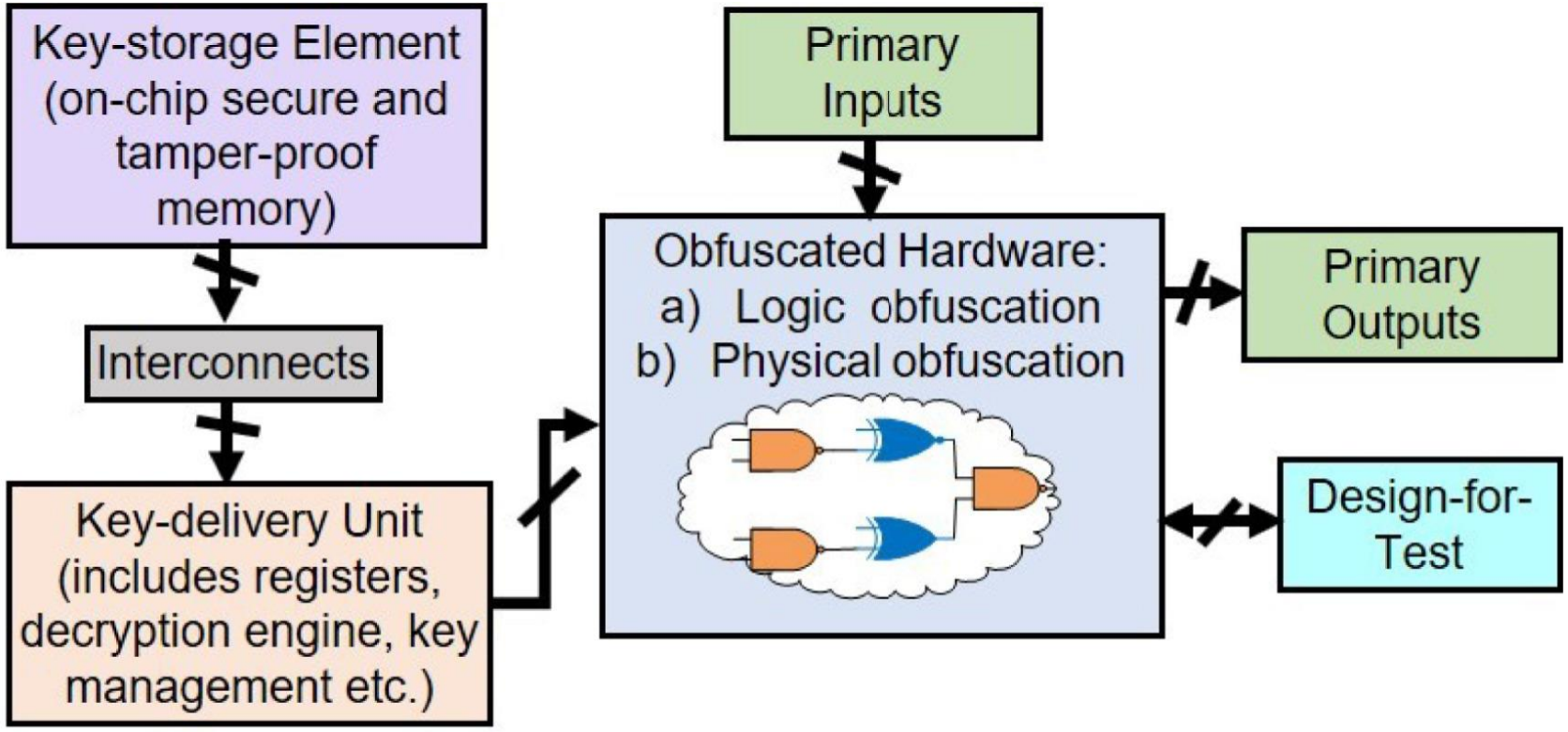


- Problem: removal attack
  - An adversary can identify and remove all yellow parts and replace by a logic 0

Y. Xie and A. Srivastava, "Anti-SAT: Mitigating SAT Attack on Logic Locking," in *IEEE TCAD*, Feb. 2019

Locking-based solutions: system view

- Logic locking: simple idea, complex implementation



M. Tanjidur Rahman et al., "Defense-in-depth: A recipe for logic locking to prevail". Integration vol. 72, May 2020.

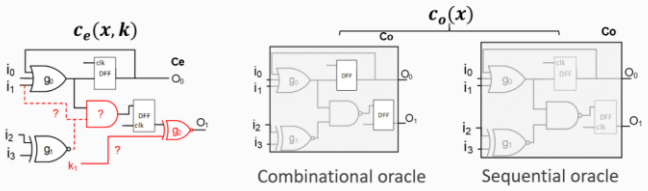
Overview

Class of technique	Effectiveness	Tooling support	Future adoption?
Layout-based	Very limited ☹	Non-existent ☹	Unlikely ☹
Locking	**Under debate, several iterations of the technique ☹☺😐**	**Various, open source ☺**	**Possible ☺**

NEOS: Netlist Encryption and Obfuscation Suite

By: Kaveh Shamsi and Yier Jin

Stage: RTL, Gate level



RANE: Reversal Assessment of Netlist Encryption

By: Shervin Roshanisefat and Avesta Sasan (George Mason University)

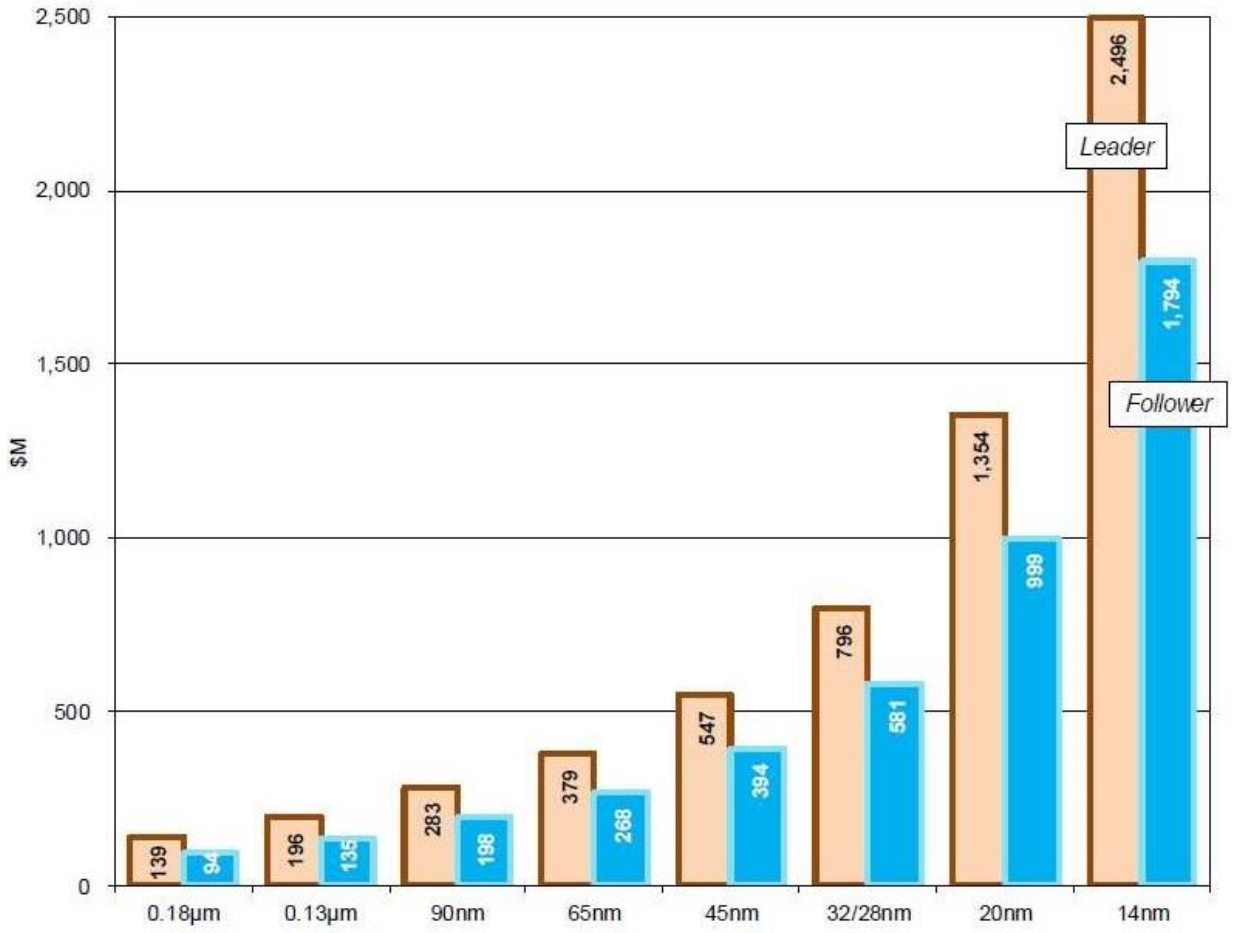
Stage: RTL, HDL

https://cadforassurance.org/

TAL TECH

Macro approaches

- Key concept: trusted fabrication is not feasible or affordable

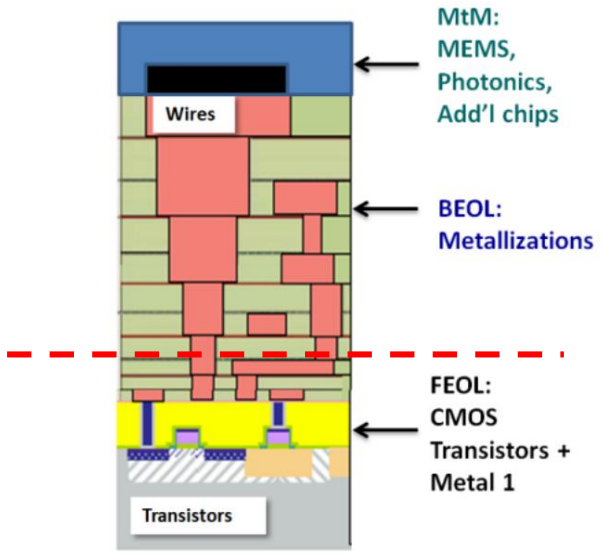


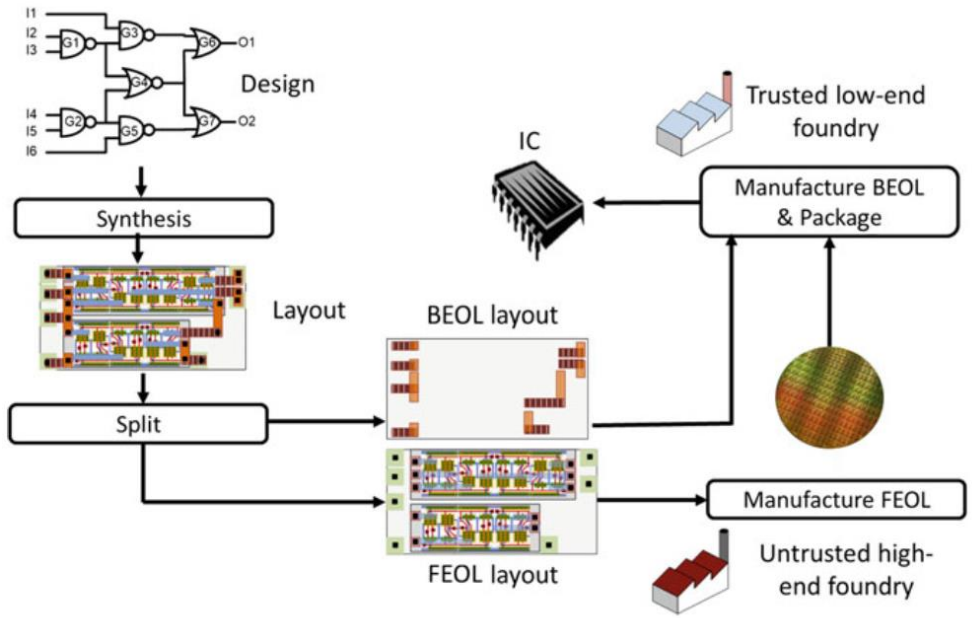
- Cost of a <14nm foundry: $5B
- Very few sites in the world with this capability
  - TSMC, Samsung, Intel
- Question is, how can I have access to the best transistor technology there is without revealing information about my design?
- (partial) trusted fabrication of integrated circuits
  Macro approaches

Source: EETimes, June 15, 2012.

Macro approaches: Split-Fabrication

- **Hybrid manufacturing solution**
  - Trusted fab
  - Untrusted fab
- Leverages the **high-performance** of untrusted fabrication (fast and power-efficient transistors)
- Prevents Trojan insertion
- Prevents IP theft
- Prevents overproduction

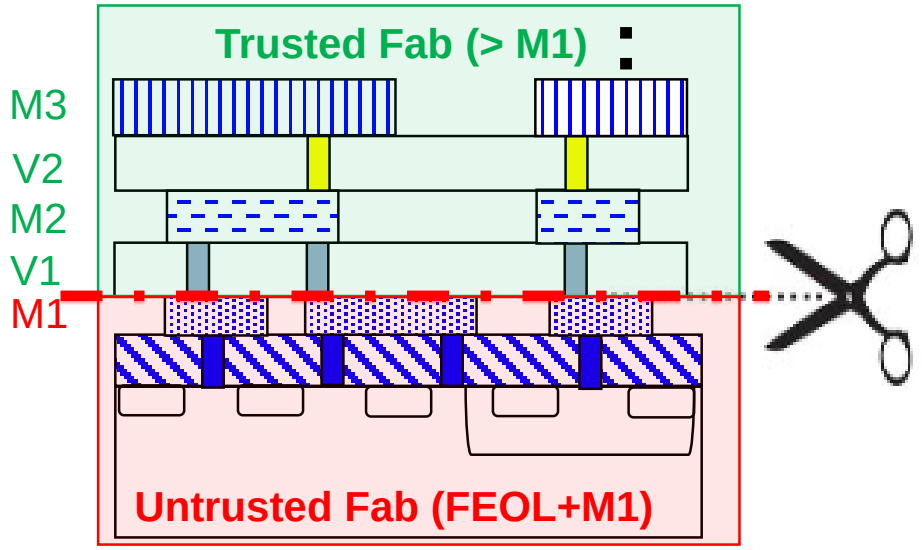




Macro approaches: Split-Fabrication

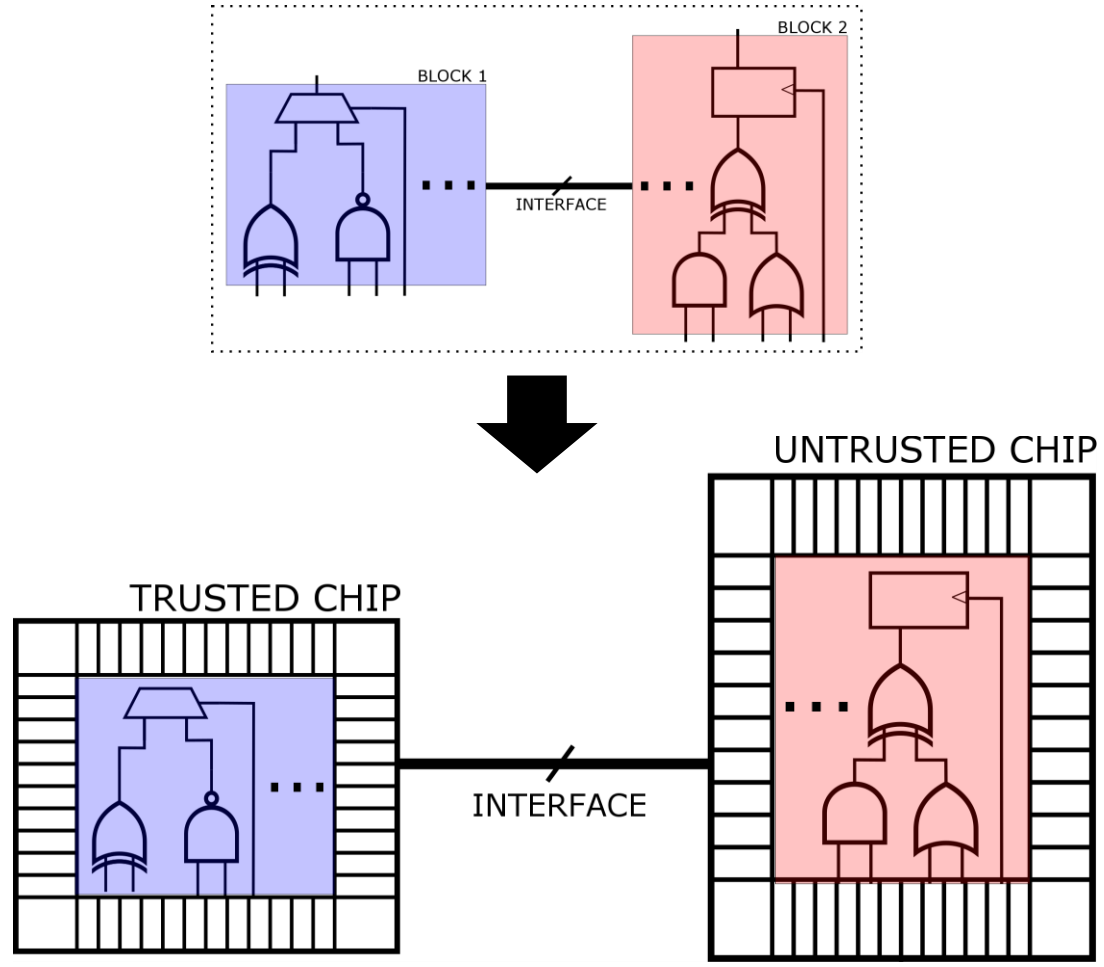
- Drawbacks of Split-Fab
  - **Hybrid PDK needed**
  - **Yield assignment?**
  - Alignment concerns?
  - **Finding foundries willing to play along ☹**

What is the alternative?



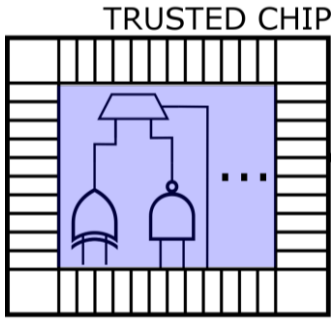
Macro approaches: Split-Chip

- Core concept: one design, two chips
- May have ‘zero’ performance loss if split thoughtfully

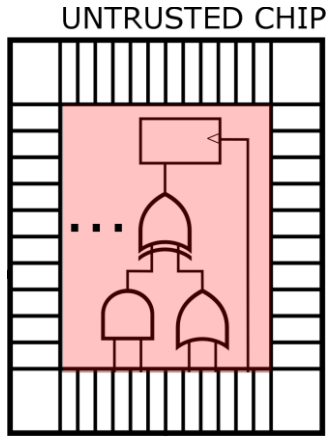


TAL TECH

Macro approaches: Split-Chip



- ASIC design (trusted foundry, onshore)
- **Legacy technology node**
- Control oriented

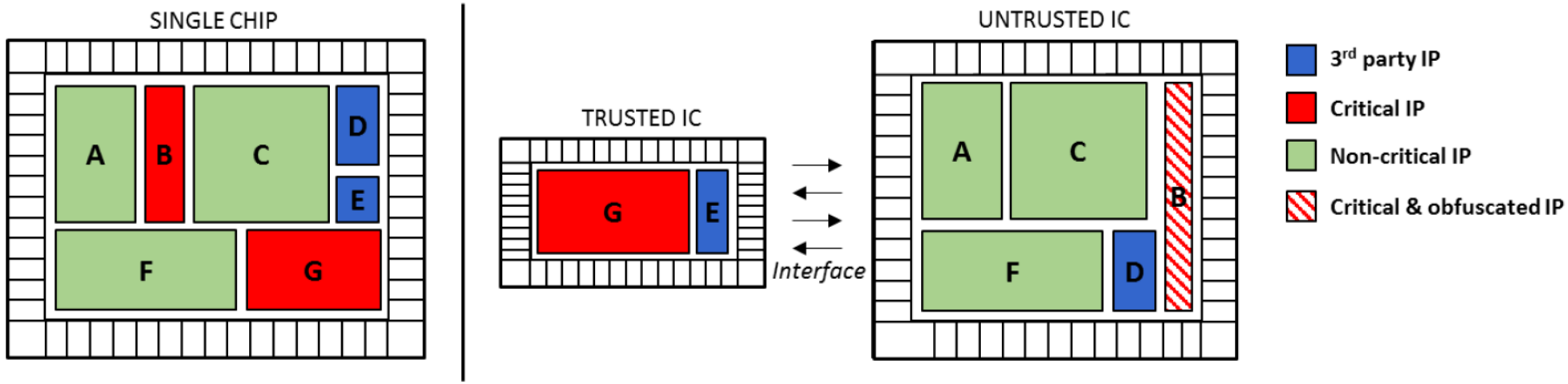


- ASIC design (untrusted foundry, offshore)
- High performance, high density
- Data oriented, efficient processing

TAL TECH

No silver bullet in obfuscation…

- What about block B?



Source: S. Pagliarini et al., "Split-Chip Design to Prevent IP Reverse Engineering," in IEEE Design & Test, 2021

Macro approaches

- There are fabricated designs that use split-fab technology
- There are fabricated designs that use split-chip technology

- Other macro or system-level approaches do exist
  - 3D integration, 2.5D, silicon on interposer, chiplets
  - **eFPGA**
  - Combining CMOS with other materials, emerging technologies

TAL TECH

Overview

Class of technique	Effectiveness	Tooling support	Future adoption?
Layout-based	Very limited ☹	Non-existent ☹	Unlikely ☹
Locking	Under debate, several iterations of the technique	Various, open source	Possible
Macro approaches	**Under debate**	**Non-existent ☹**	**Uncertain ☹**

Take-away message?

TAL TECH

Overview

Class of technique	Effectiveness	Tooling support	Future adoption?
Layout-based	Very limited ☹	Non-existent ☹	Unlikely ☹
Locking	Under debate, several iterations of the technique	Various, open source	Possible
Macro approaches	Under debate	Non-existent ☹	Uncertain ☹
Behavioral obfuscation	**Ask Levent!**	**Ask Levent!**	**Ask Levent!**

TAL TECH

QUESTIONS?