ASSURER: A PPA-friendly Security Closure Framework for Physical Design

Guangxin Guo, Hailong You, Zhengguang Tang, Benzheng Li, Cong Li[*], and Xiaojue Zhang

Department of Microelectronics,

Xidian University

ASP-DAC 2023

Outline

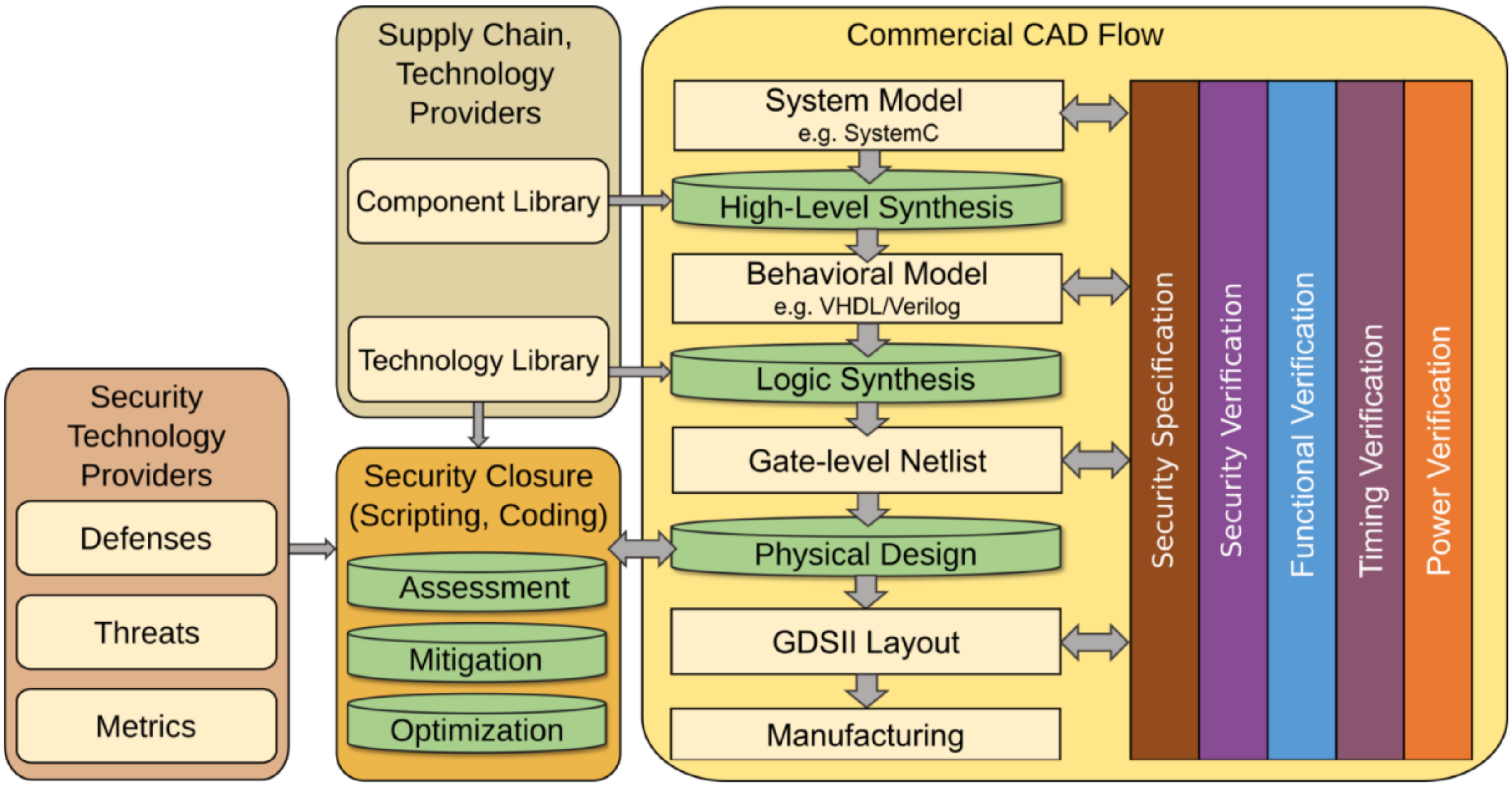
- Problem Background
- Problem Formulation
- Proposed Framework
- Experimental Results and Conclusions

Outline

- Problem Background
- Problem Formulation
- Proposed Framework
- Experimental Results and Conclusions

Problem Background

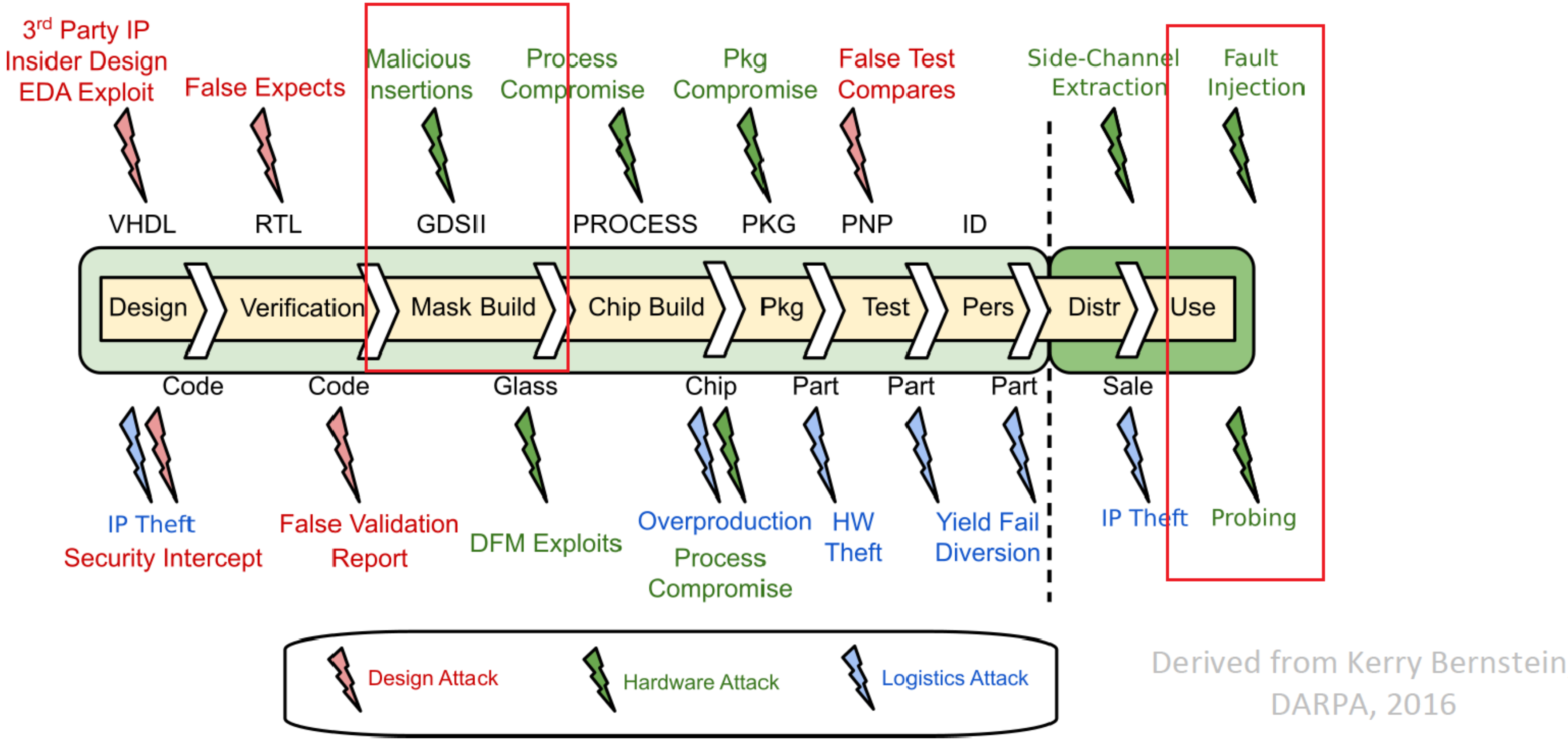
- Security Closure



Reference: https://wp.nyu.edu/ispd_22_contest/details/

Problem Background

- Attack targeting at physical design



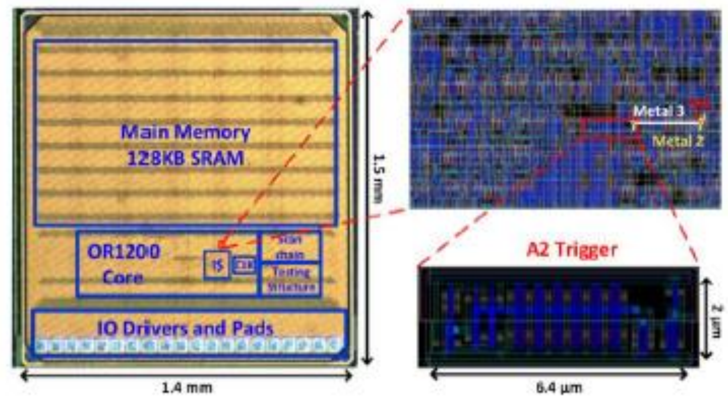
Objective:

1. harden layouts against post-design Trojan insertion
2. against electro-optical or contact-based probing, fault injection attacks targeting the frontside

Reference: https://wp.nyu.edu/ispd_22_contest/details/

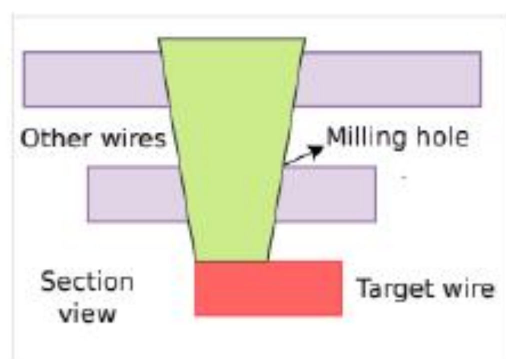
Problem Background

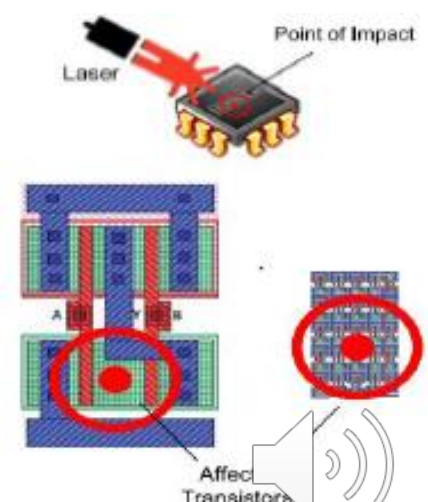
- Hardware Trojan attack
  - targeting at the physical level
  - seeking to leak information
  - reduce the IC's performance
  - disrupt an IC's working altogether
  - always on



Yang et al., SP 2016

- Probing and Fault injection
  - extract data from frontside
  - contact-based micro-probing, electromagnetic field probing, or electro-optical device probing.





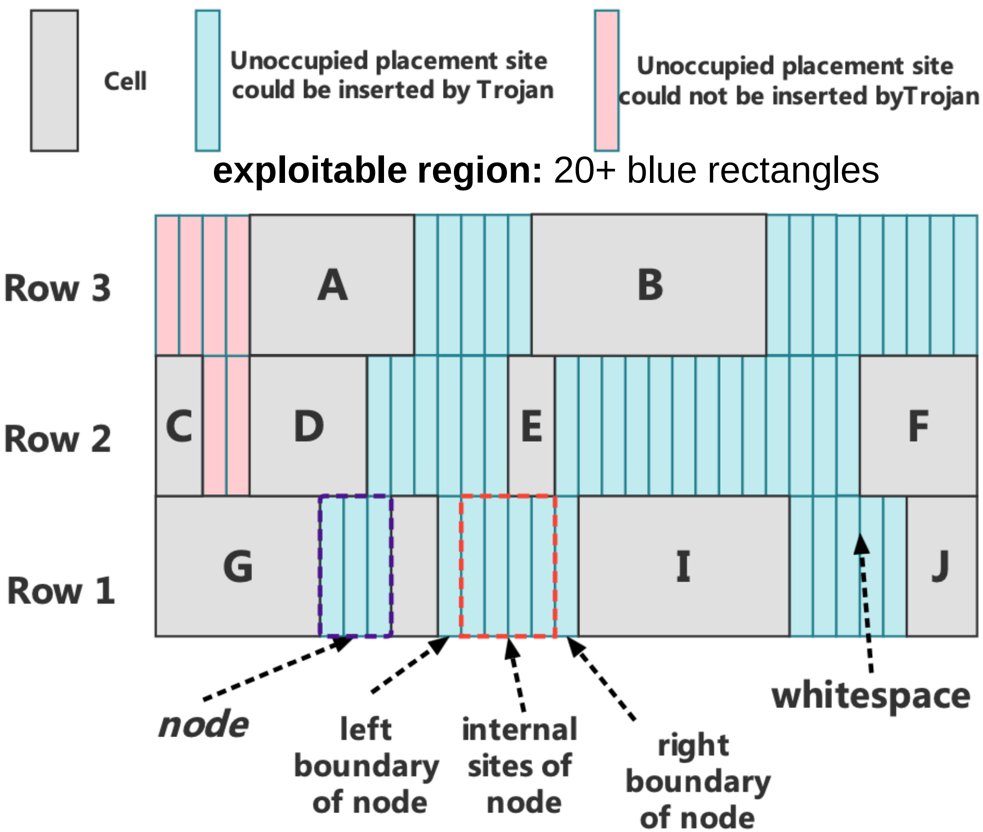
Reference: https://wp.nyu.edu/ispd_22_contest/details/

Outline

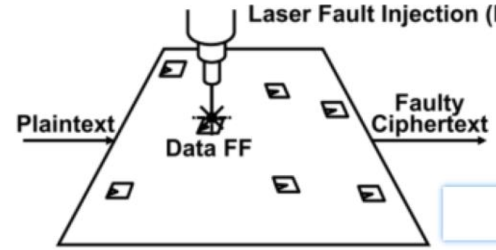
- Problem Background
- Problem Formulation
- Proposed Framework
- Experimental Results and Conclusions

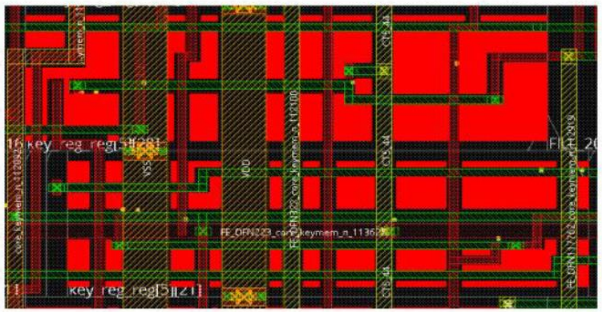
Problem Formulation

- ISPD 2022 Challenge



Trojan Insertion attack





Security cells and nets

exposed at frontside

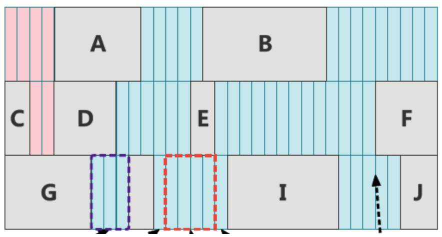
Probing attack

Both can be solved by P&R

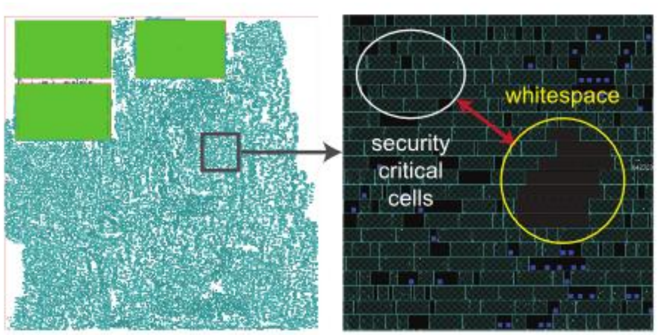
Previous works

Prevent Trojan insertion

1. Fill functional cell greedily



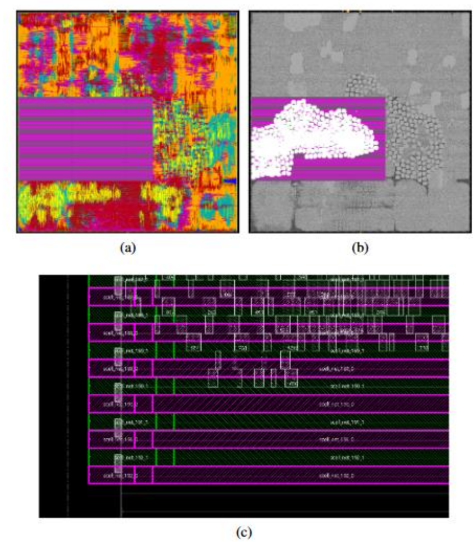
2. Increase cell density locally



100% completed   whitespace moved away

Prevent probing attack

1. Routing security nets in the low metal layer
2. Widen high layer nets width
3. Add self defined cells and nets

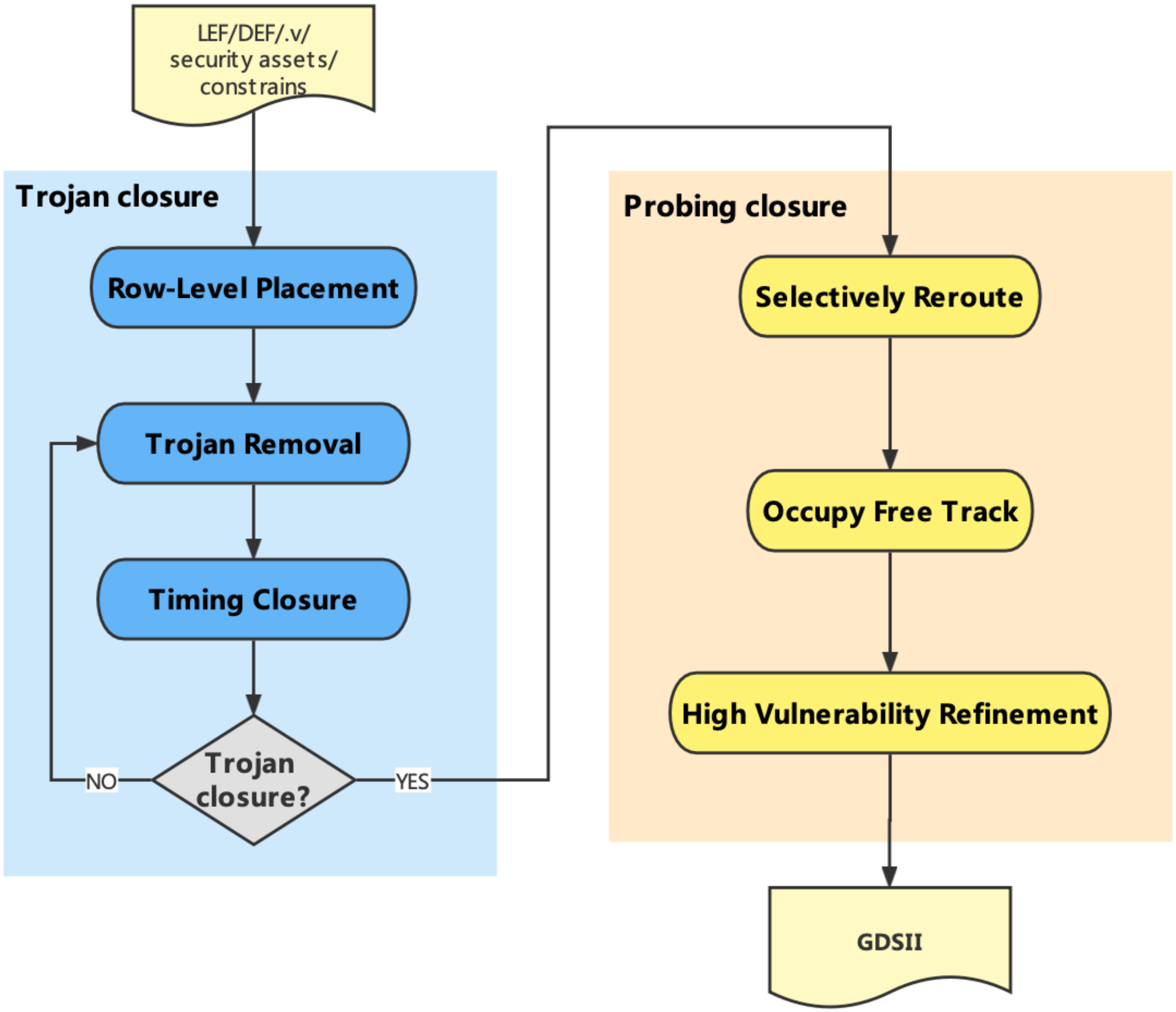


Outline

- Problem Background
- Problem Formulation
- Proposed Framework
- Experimental Results and Conclusions

Proposed Framework

- ASSURER Framework

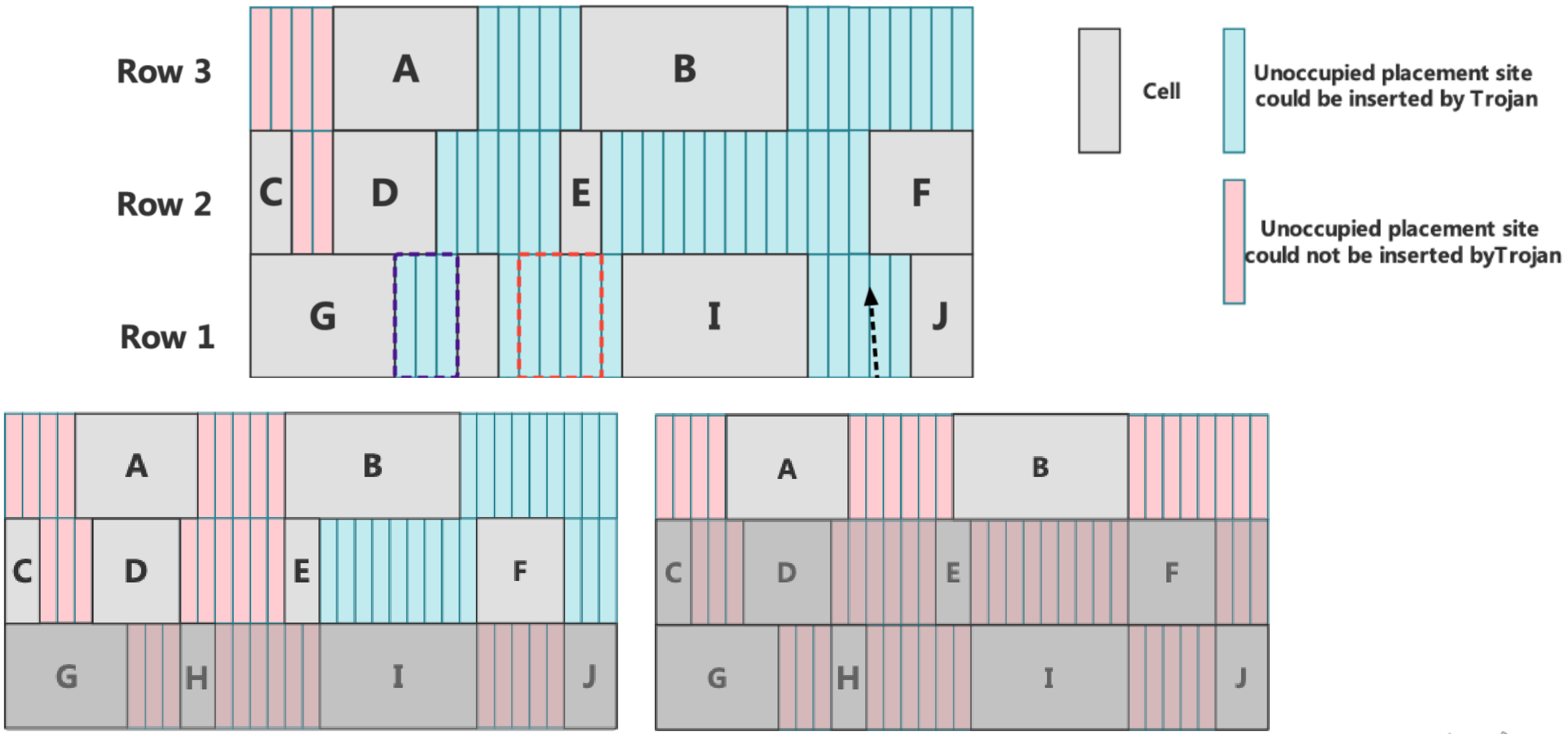


Trojan Defense Framework

- Row-level Placement Refinement

$$R(p,s) = -\lambda \times |s| + \alpha \times C_{left}(p) + \beta \times C_{right}(p) + \gamma \times C_{inter}(p)$$

$$\lambda = \begin{cases} 0.1, & s < 0 \\ 0.2, & s > 0 \end{cases}$$



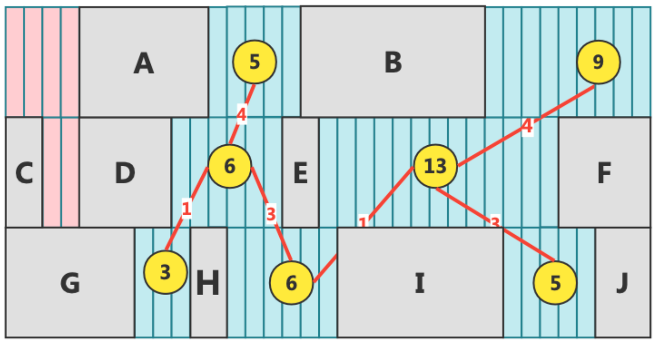
(a) After placing row 2

(b) After placing row 3

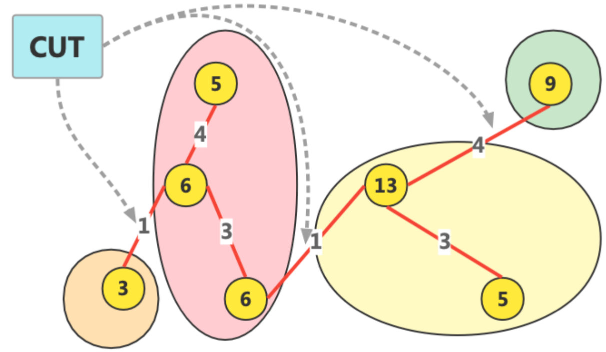
Trojan Defense Framework

- Trojan Removal

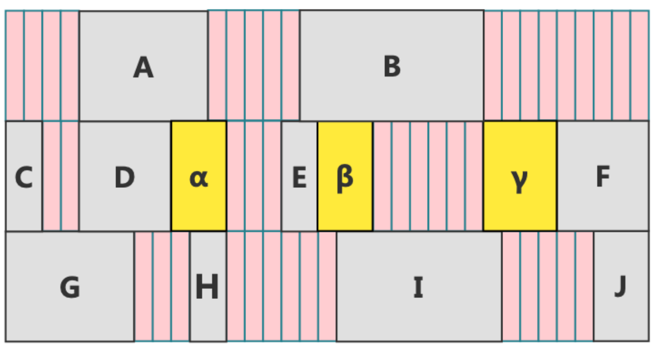
Stage1: Partition



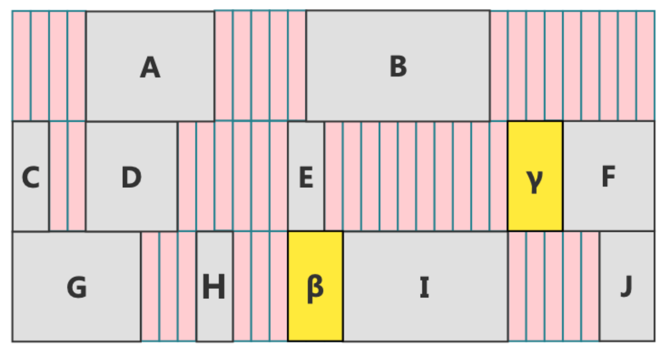
(a) Build graph from layout.



(b) Cluster node and cut edge.



(c) Cutting sites without refinement. Rectangles in yellow are *cutting sites*, whose sites number do not less than three for cell library consideration.



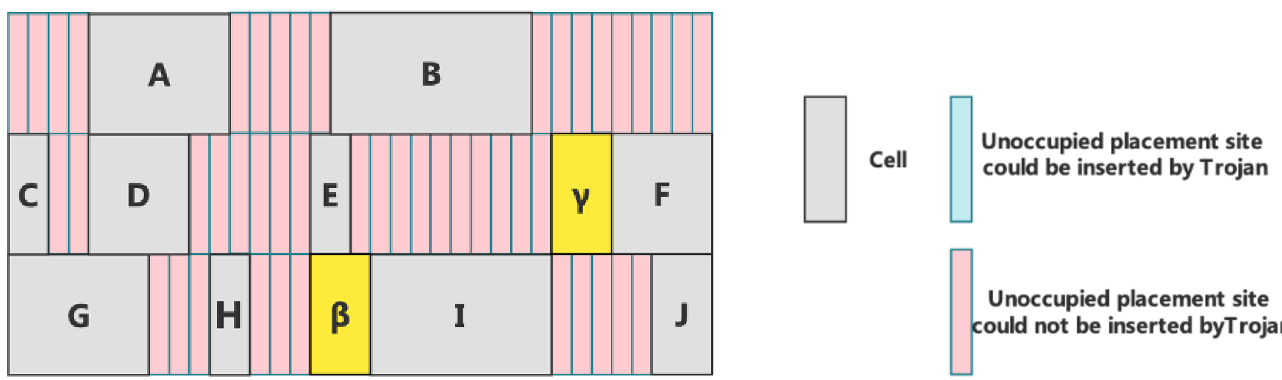
(d) Cutting sites with refinement.

Trojan Defense Framework

- Trojan Removal

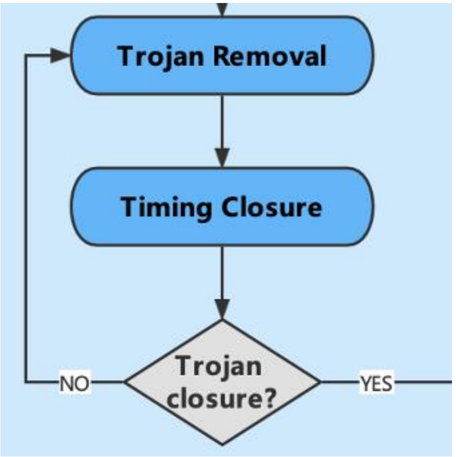
Stage2: Standard cell refinement

  - chain movement
  - increasing cell drive strength
  - deleting redundancy inserted cells

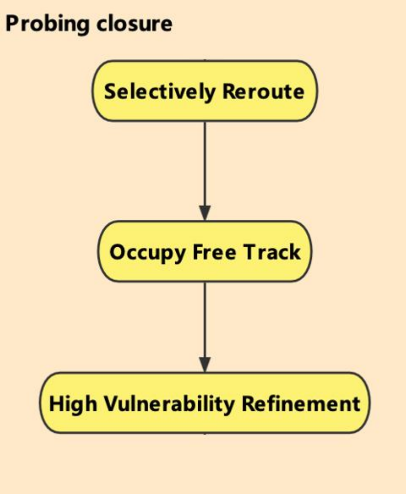


Trojan Defense Framework

- Timing Closure
  - Connecting buffers to the net with maximum time slack
  - Timing optimization based on the Cadence Innovus



Probing Defense Framework

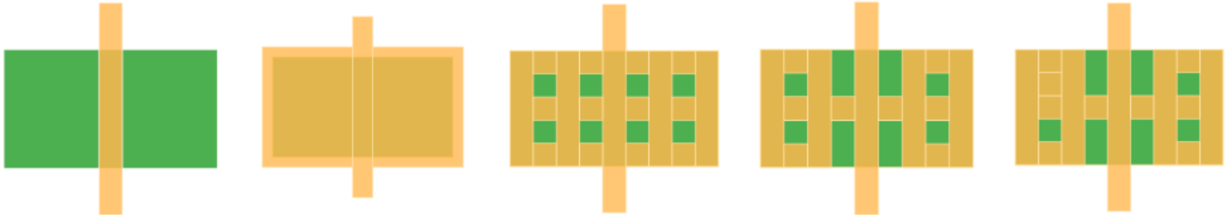


- Selectively Reroute
  - Objective
    - Routing security nets in the lower metals.
    - Routing non-security nets to cover security nets
  - Steps
    1) delete the routes of security nets and export routing of residual nets.
    2) delete all routing.
    3) set routing constraints, e.g., set the top routing layer of the security nets, and set routing blockage at the top layer at the specific rectangles.
    4) route the security nets considering the constraints.
    5) import the routing, which is exported in the previous step and deal with the conflicts.

Probing Defense Framework

- Occupying Free Track

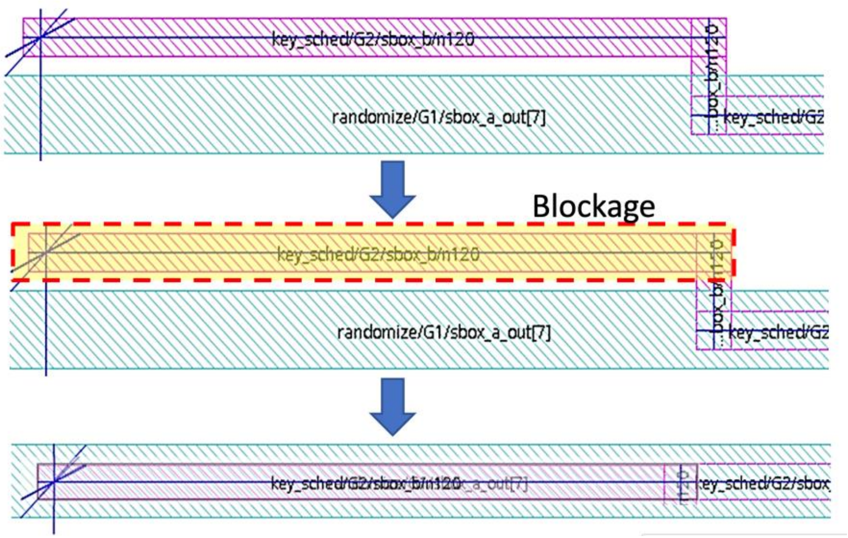
- Steps
  1) Get the present routing result
  2) Patch routing segment on track if layer num is even, else middle.
  3) DFS find free tracks above security nets and cells.
  4) Connect added segment to non-security nets
  5) DRC-informed hole-patching algorithm

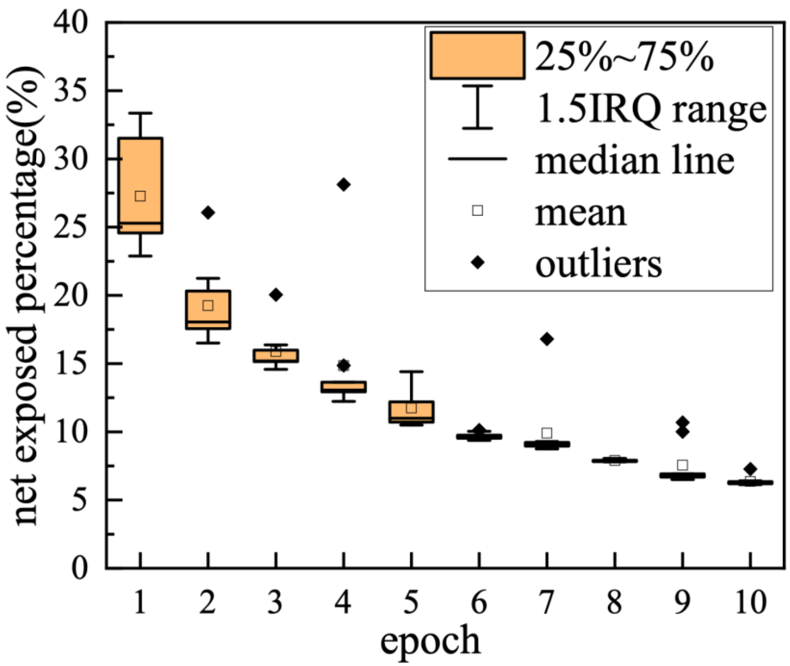


(a) (b) (c) (d) (e)

Probing Defense Framework

- High Vulnerability Refinement
  - Targeting at exposed area of high vulnerable
  - Move nets with a few epochs

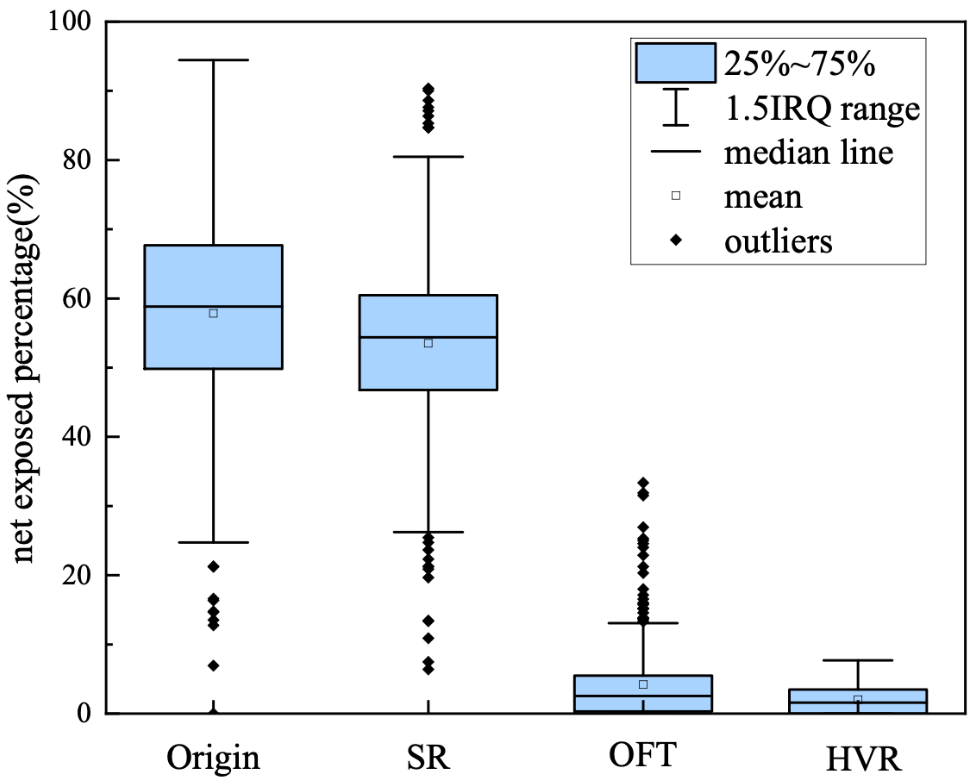




- furtherly reduce the maximal and total exposed area

Probing Defense Framework

- Selectively Reroute(SR)
- Occupying Free Track(OFT)
- High Vulnerability Refinement(HVR)



Outline

- Problem Background
- Problem Formulation
- Proposed Framework
- Experimental Results and Conclusions

Experimental Results

- Trojan Closure Result

ISPD 2022 contest benchmarks

1) Core utilization (CU) is the utilization percentage of placement sites.

2) Area (AR) is the area of the layout.

3) Cell number (CN) is the cell number across the layout

4) Leakage power (LP)

Table 1: Experimental results of Trojan closure on ISPD2022 Contest benchmarks [4]

Case name	Initial				Best in ISPD'22*				Ours				Ours-Shrink			
	CU	AR	CN	LP	CU	AR	CN	LP	CU	AR	CN	LP	CU	AR	CN	LP
AES_1	75	51113	16509	0.77	78	51113	17302	0.81	76	51113	16887	0.79	93	40814	16571	0.74
Camellia	51	19698	6710	0.15	59	19698	8158	0.18	53	19698	6979	0.15	94	11072	6730	0.15
CAST	51	30494	12682	0.26	72	30494	15903	0.41	57	30494	13784	0.30	92	17954	13105	0.26
MISTY	52	24168	9517	0.20	70	24168	11479	0.31	64	24168	10931	0.27	92	14346	9850	0.20
openMSP430_1	50	19395	4690	0.11	61	19395	6372	0.16	56	19395	5390	0.14	98	10377	4625	0.11
PRESENT	51	4301	868	0.02	60	4301	1144	0.03	55	4301	994	0.02	99	2410	869	0.02
SEED	51	30494	12682	0.26	72	30494	15777	0.41	57	30494	13294	0.30	92	17954	13093	0.27
TDEA	81	5443	2269	0.05	81	5443	2279	0.05	81	5443	2269	0.05	95	4456	2263	0.05
Ratio	1.00	1.00	1.00	1.00	1.22	1.00	1.21	1.34	1.09	1.00	1.08	1.14	1.69	0.63	[illegible]	1.00

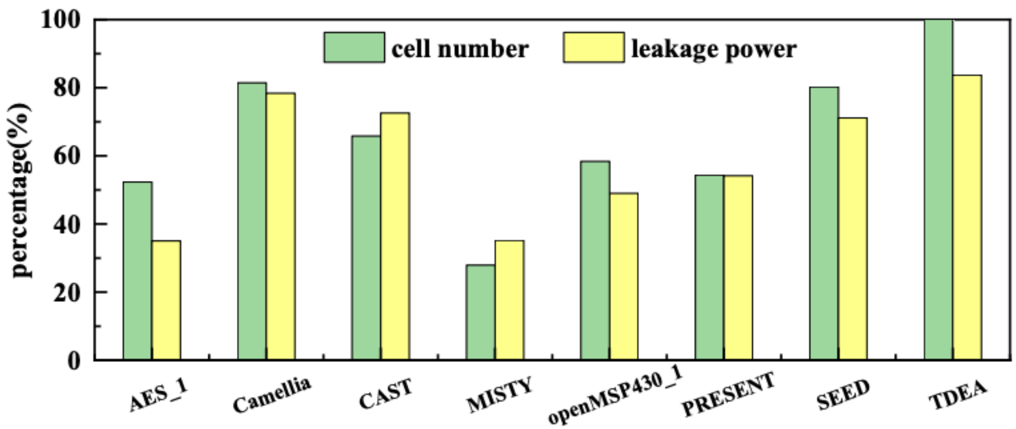
* The scripts and executable program are got from the first place in ISPD'22.

Experimental Results

- Trojan Closure Result

Table 2: Total power(mW) after Trojan closure

Design	Initial	ISPD'22	Ours	Ours-shrink
AES_1	66.67	68.81	68.61	64.48
Camellia	1.69	2.15	1.89	1.73
CAST	4.60	7.16	5.69	4.83
MISTY	3.30	4.66	4.16	3.42
openMSP430_1	0.38	0.50	0.44	0.42
PRESENT	0.38	0.44	0.41	0.33
SEED	4.60	7.20	5.39	4.85
TDEA	1.48	1.53	1.48	1.49
Ratio	1.00	1.30	1.14	1.02



Reduced percentage compared to ISPD'22 first place

Experimental Results

- Probing Prevention Result

Experimental results of probing prevention on ISPD2022 Contest benchmarks

Design	Initial						After probing hardened						**Score**
	c_{total}	c_{max}	c_{avg}	n_{total}	n_{max}	n_{avg}	c_{total}	c_{max}	c_{avg}	n_{total}	n_{max}	n_{avg}	
AES_1	505.86	60.71	26.81	3524.54	100.00	48.44	0.43	6.51	0.04	7.80	2.95	0.14	2.39%
Camellia	481.93	81.40	40.22	525.72	94.31	57.73	1.77	7.10	0.10	18.79	7.69	2.01	4.11%
CAST	913.82	86.19	35.38	1878.62	100.00	54.59	3.38	5.99	0.11	51.13	9.96	1.22	3.80%
MISTY	516.86	76.28	44.65	13.65	94.64	74.88	0.81	2.23	1.28	0.00	0.00	0.00	0.77%
openMSP430_1	1505.88	85.03	44.92	1693.41	100.00	65.36	0.67	3.57	0.03	19.10	6.91	0.48	2.21%
PRESENT	469.82	82.40	58.64	110.91	99.51	72.15	0.06	0.73	0.01	0.19	1.74	0.09	0.50%
SEED	2112.25	86.19	37.26	4212.88	100.00	55.68	2.72	6.56	0.05	49.40	8.76	0.45	3.13%
TDEA	407.76	75.59	53.67	180.98	100.00	71.70	0.48	1.82	0.06	2.33	8.31	1.63	2.34%
Ratio	100%	100%	100%	100%	100%	100%	0.16%	5.55%	0.49%	1.28%	5.85%	1.24%	2.40%

- Reduce the vulnerability score by 97.6%, from 100% to 2.4%.
- Cell total exposed area and net total exposed area can be substantially reduced: 0.16% and 1.28%

Conclusion

- Present ASSURER for security closure considering PPA
  - Using Reward-directed placement to prevent Trojan
  - Casting Trojan removal into graph partition problem
  - Probing attack prevention flow based on ECO routing
    - Selectively reroute security nets.
    - Occupy free track above security assets
    - Iterative high vulnerability refinement
- Compared with the first place of ISPD 2022 contest:
  - Reduce 53% additional total power
  - Reduce 65% additional cell insertion
  - Probing vulnerability can be reduced by 97.6% on average

ASSURER: A PPA-friendly Security Closure Framework for Physical Design

Guangxin Guo, Hailong You, Zhengguang Tang, Benzheng Li, Cong Li*, and Xiaojue Zhang

Thanks for your listening