ISPD2022 – DESIGN CONTEST

SECURITY CLOSURE OF PHYSICAL LAYOUTS

Samuel Pagliarini, Tiago D. Perez, Felipe Almeida, and Mohammad Eslami
Dpt. of Computer Systems - School of IT
Tallinn University of Technology

ISPD

- ISPD itself is a "mature" conference
  - First workshop took place in 1987
  - Conference status since 2001

- ISPD also organizes a *CAD contest*
  - Long history in the circuit design community, 18th edition

ISPD 2013: Discrete Gate Sizing Contest
ISPD 2012: Discrete Gate Sizing Contest
ISPD 2011: Routability-Driven Placement
ISPD 2010: High Performance Clock Network Synthesis
ISPD 2009: Clock Network Synthesis
ISPD 2008: Global Routing
ISPD 2007: Global Routing
ISPD 2006: Placement
ISPD 2005: Placement

Router links: choose a link
Placer links: choose a link

ISPD 2021: Wafer-Scale Physics Modeling Contest
ISPD 2020: Wafer-Scale Deep Learning Accelerator Placement
ISPD 2019: Initial Detailed Routing
ISPD 2018: Initial Detailed Routing
ISPD 2017: Clock-Aware FPGA Placement
ISPD 2016: Routability-Driven FPGA Placement Contest
ISPD 2015: Blockage-Aware Detailed Routing-Driven Placement Contest
ISPD 2014: Detailed Routing-Driven Placement Contest

TAL TECH

Source: https://ispd.cc/ispd2022/index.php

Why participate?

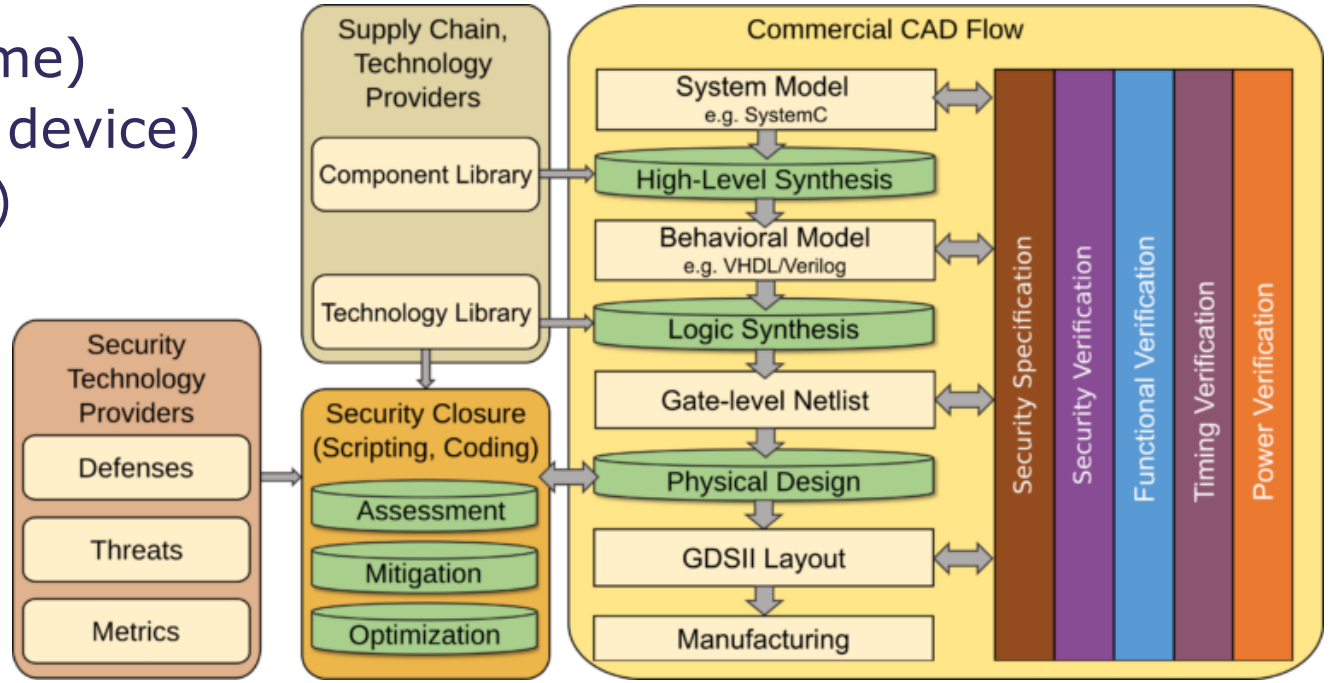
- First time it covered **hardware security** topics
- Theme: Security closure of Physical Layouts
  - "CAD tools traditionally optimize for PPA. However, considering that various and serious threats are emerging, future CAD flows should also incorporate techniques for secure IC design."
    - *Here is a layout, go and secure it!*
- Duration: Eight weeks to "fix" security problems
  - Alpha phase
  - Final phase

Source: https://wp.nyu.edu/ispd_22_contest/

Theme – Security Closure

Main theme: *leverage CAD tools features for not only improving PPA, but also enhancing the layout security*

- Hardening layouts at design time against threats that are executed post-design
  - Trojan horses (at fabrication time)
  - Fault injection (on a fabricated device)
  - Probing (of a fabricated device)
- Implement measures for security closure, i.e., to proactively harden layouts

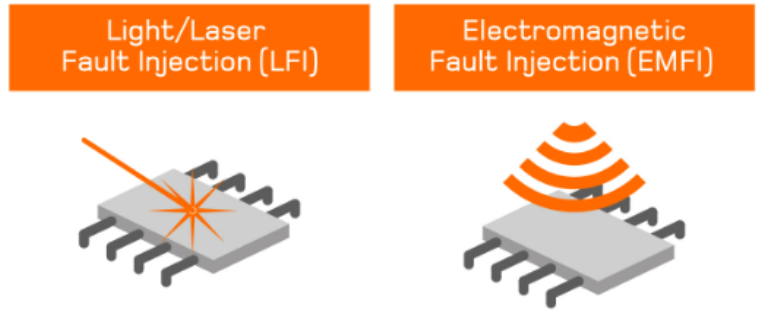


Source: https://wp.nyu.edu/ispd_22_contest/ and https://www.appluslaboratories.com/global/en/news/publications/new-fault-injection-attacks

Security Closure of Physical Layouts

The threats the contest focused on:

- Probing and fault injection: in-field electro-optical or contact-based probing, fault injection attacks targeting the the front side (from top to bottom)
  - **Cell** assets and **net** assets
  - The design must *protect itself*



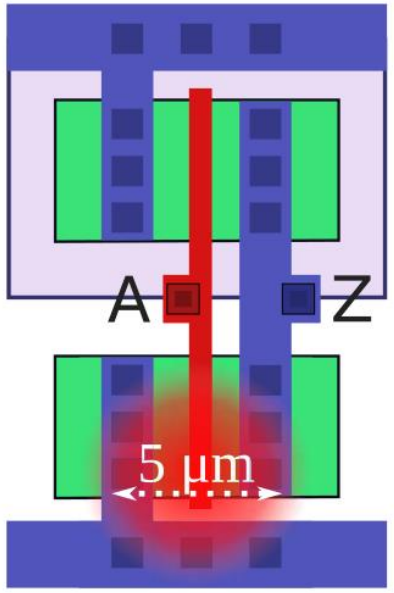
- Trojan Insertion: fabrication-time attack
  - **Exploitable region:** placement sites and routing resources

- *Control placement and routing in such a way that insertion of Trojan components (trigger and payload) becomes difficult*
- *Control placement and routing in such a way that probing/fault injection on particular devices or wires becomes difficult*

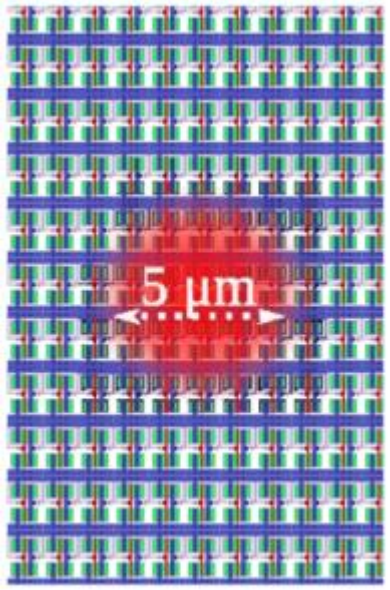
TAL TECH

Frontside Probing, Fault Injection

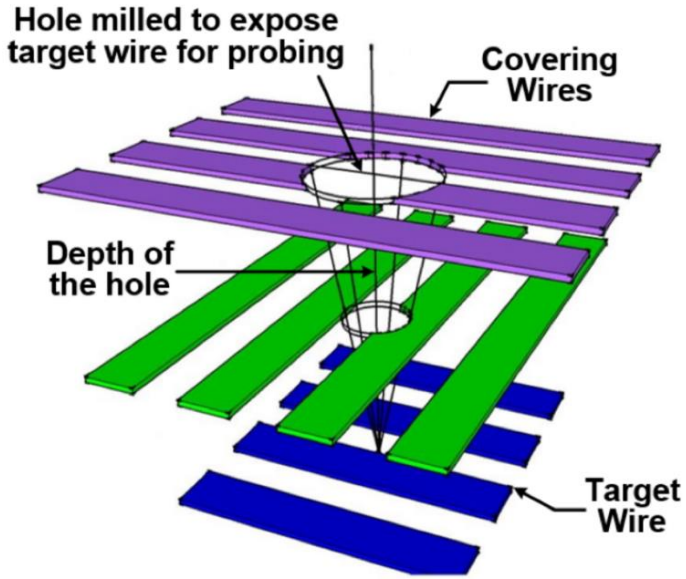
- Adversary capabilities



250nm technology



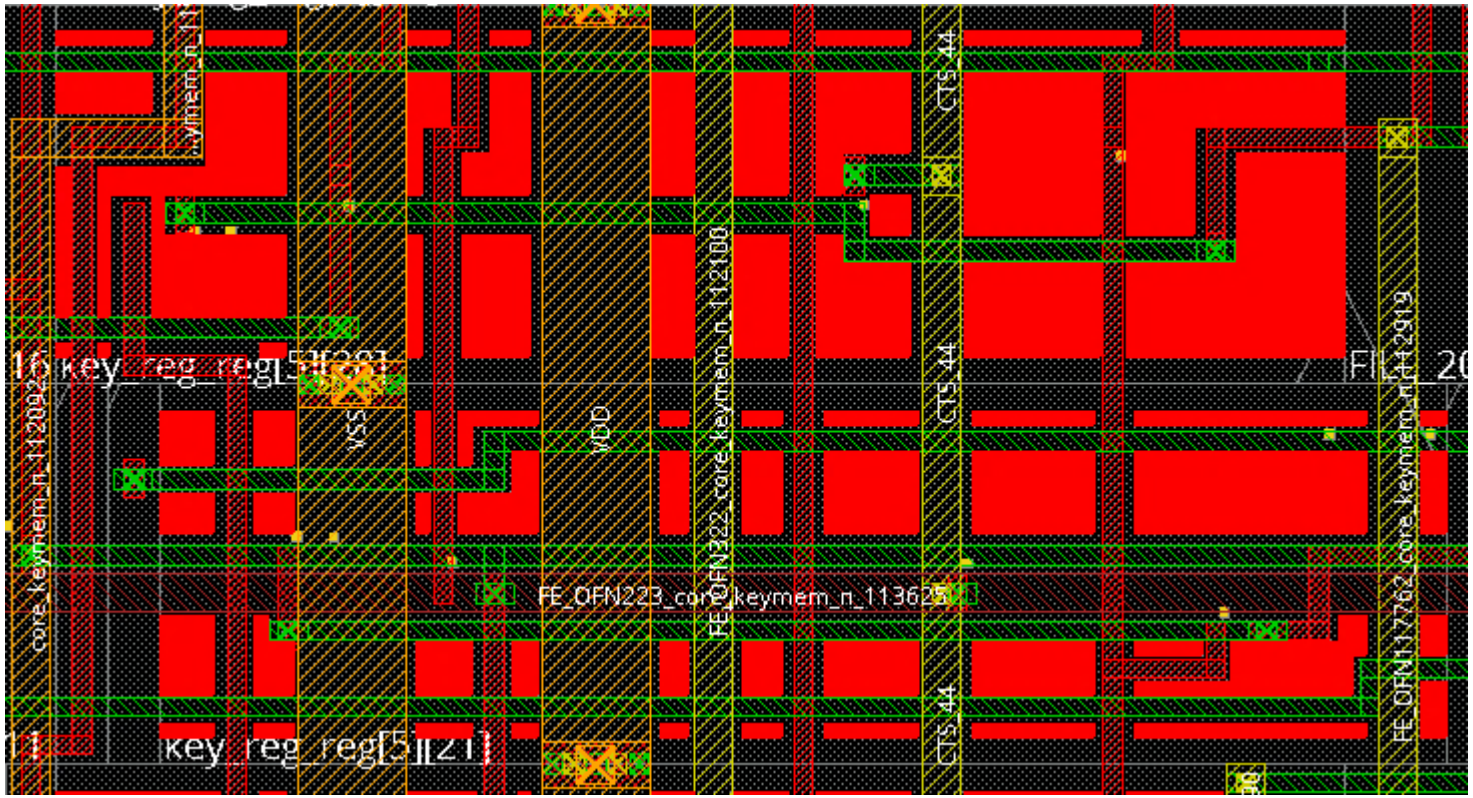
28nm technology



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

Frontside Probing, Fault Injection

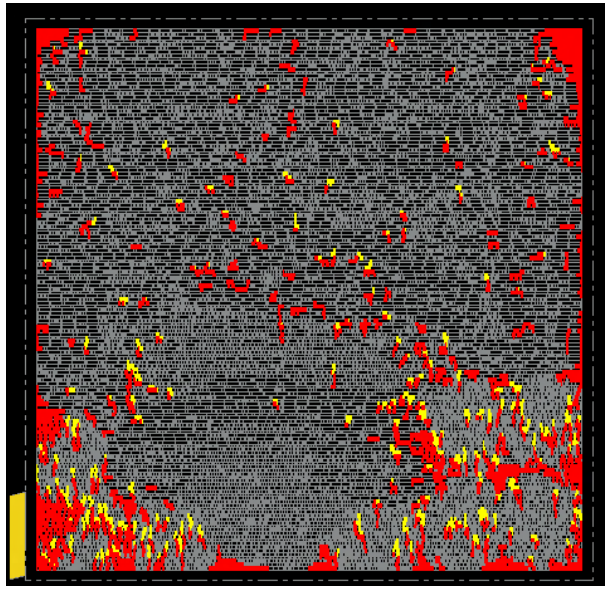
- A zoomed-in example for exposure of standard cells
  - The regions highlighted in red are exposed from the frontside, i.e., direct line of sight



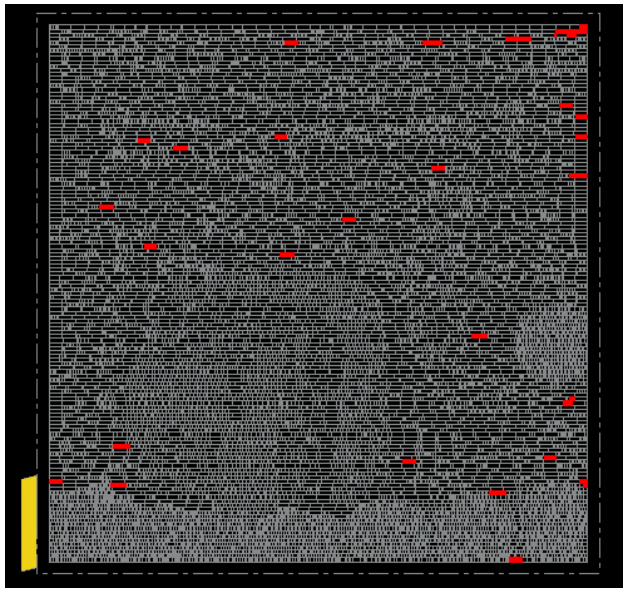
Source: https://wp.nyu.edu/ispd_22_contest/details

Hardware Trojan insertion

- For Trojan insertion, metrics are based on *exploitable regions*, i.e., sets of spatially continuous placement sites that are either a) **free** or b) occupied only by **filler cells**
- Routing resources are also considered, as Trojans would require some connectivity as well. In other words, exploitable regions are those where an attacker would be able to find or make some space and ruting resources to insert and connect their Trojans.



70% utilization

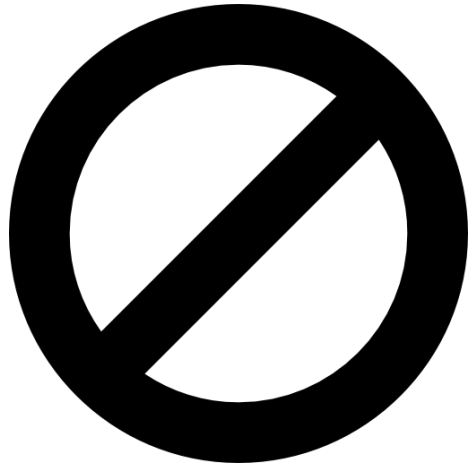


90% utilization

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

Additional restrictions

- Very little freedom to move **pins**
- Very little freedom to change the **power distribution network**
- Cannot change/improve **standard cell library**
- Cannot change **metal stack**
- Designs must remain *functionally equivalent*
  - We can resize, reroute, add/remove buffers…



- Trivial defenses are not considered effective
  - Filler cells, unconnected cells

- It is not allowed to introduce **dedicated sensor circuitry** or checkers

TAL TECH

Scoring system

- **Trojan insertion metrics** (ti)
  - Placement sites of exploitable regions (place_sts)
  - Routing resources of exploitable regions (route_rsrcs)
- **Frontside probing and fault injection** (fsp_fi)
  - Exposed area of standard cells assets (exp_cell)
  - Exposed area of net assets (exp_net)
- **Design cost:**
  - Power (pwr)
  - Performance (perf)
  - Area (area)
  - Routing (drc)

Final score= SEC x DES (normalized)

TAL TECH

Scoring system

Final score= SEC x DES

SEC = (ti + fsp_fi)/2

ti = 0.5*place_sts + 0.5*route_rsrcs

fsp_fi = 0.5*exp_cell + 0.5*exp_nets

DES = 0.25*pwr + 0.25*perf + 0.25*area + 0.25*drc

TAL TECH

Benchmarks assessed

- AES_v1
- AES_v2
- AES_v3
- PRESENT
- CAST
- MISTY
- Camellia

Alpha

- openMSP430
- SEED
- TDEA

Final

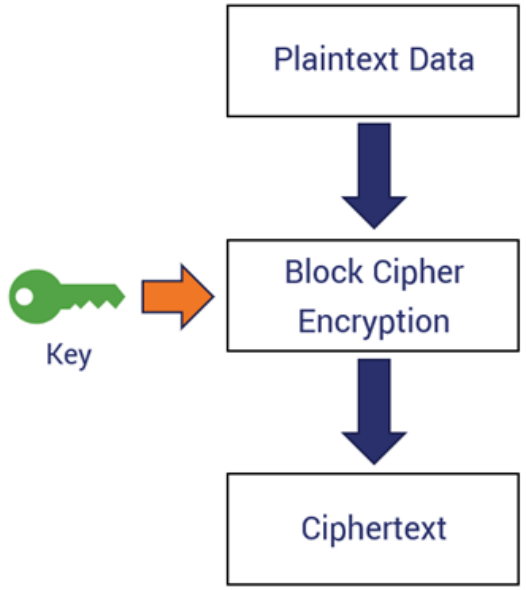
- But why these designs?

Benchmarks assessed

- **AES_v1**
- **AES_v2**
- **AES_v3**
- **PRESENT**
- **CAST**
- **MISTY**
- **Camellia**

- openMSP430
- **SEED**
- **TDEA**

- But why these designs?
  - Most are **ciphers**

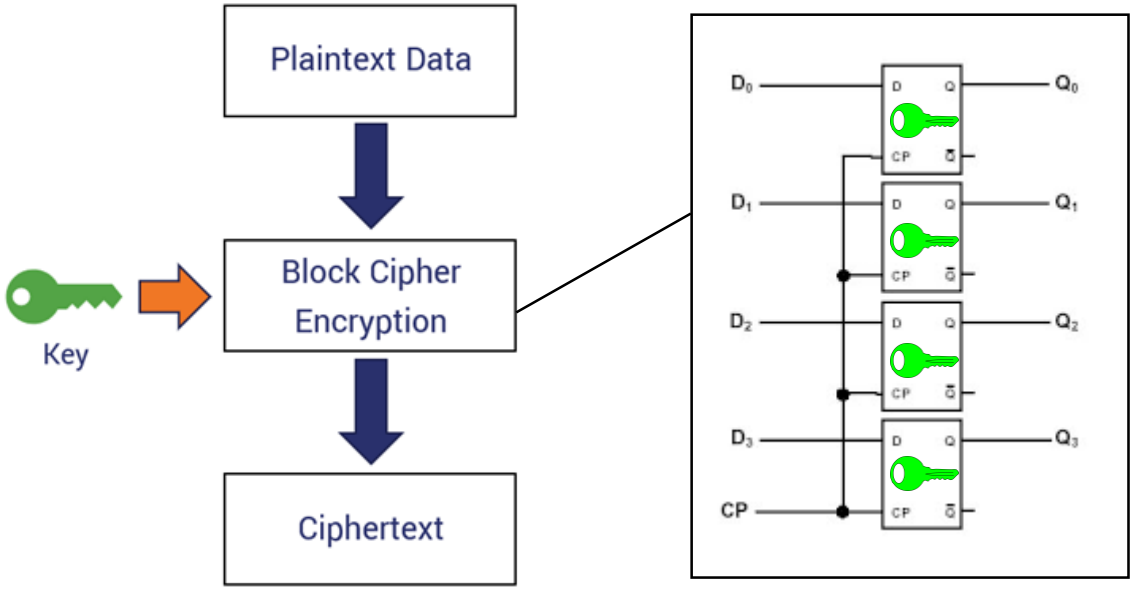


Benchmarks assessed

- **AES_v1**
- **AES_v2**
- **AES_v3**
- **PRESENT**
- **CAST**
- **MISTY**
- **Camellia**

- openMSP430
- **SEED**
- **TDEA**

- But why these designs?
  - Most are **ciphers**

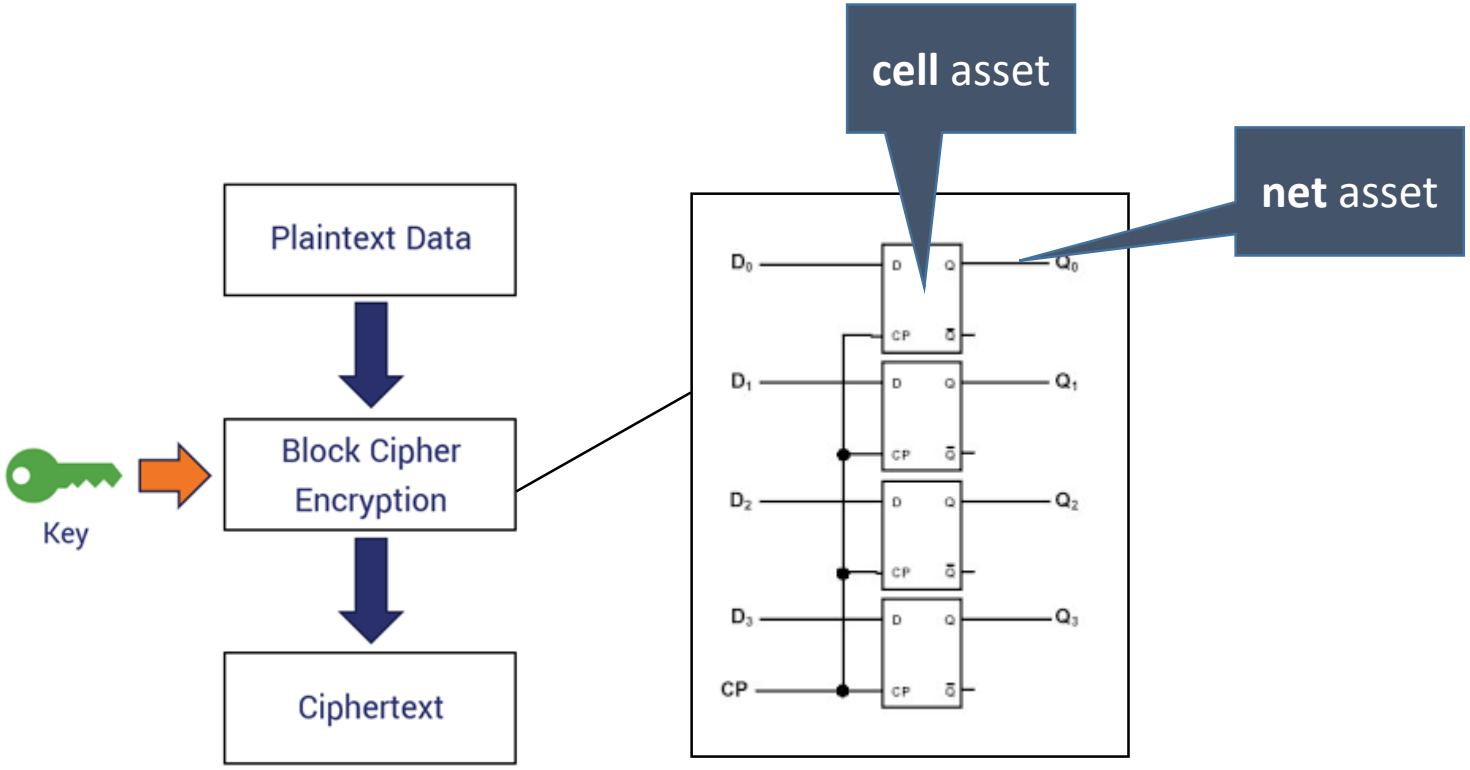


Benchmarks assessed

- **AES_v1**
- **AES_v2**
- **AES_v3**
- **PRESENT**
- **CAST**
- **MISTY**
- **Camellia**

- openMSP430
- **SEED**
- **TDEA**

- But why these designs?
  - Most are **ciphers**



Benchmarks assessed

- **AES_v1**
- **AES_v2**
- **AES_v3**
- PRESENT
- CAST
- MISTY
- Camellia

- openMSP430
- SEED
- TDEA

- But why these designs?
  - Most are ciphers
  - Some are **fast** (1GHz target frequency)

TAL TECH

Benchmarks assessed

- AES_v1 (3000 flops)
- AES_v2 (3000 flops)
- AES_v3 (3000 flops)
- **PRESENT (153 flops)**
- **CAST (300 flops)**
- **MISTY (300 flops)**
- Camellia (400 flops)

- openMSP430 (800 flops)
- **SEED (300 flops)**
- **TDEA (250 flops)**

- But why these designs?
  - Most are ciphers
  - Some are fast (1GHz target frequency)
  - Some are **very small**

TAL TECH

Benchmarks assessed

- **AES_v1 (10 metals)**
- **AES_v2 (10 metals)**
- **AES_v3 (10 metals)**
- PRESENT (6 metals)
- CAST (6 metals)
- MISTY (6 metals)
- Camellia (6 metals)

- openMSP430 (6 metals)
- SEED (6 metals)
- TDEA (6 metals)

- But why these designs?
  - Most are ciphers
  - Some are fast (1GHz target frequency)
  - Some are very small
  - Some are **hard to route** (10 metal stack)

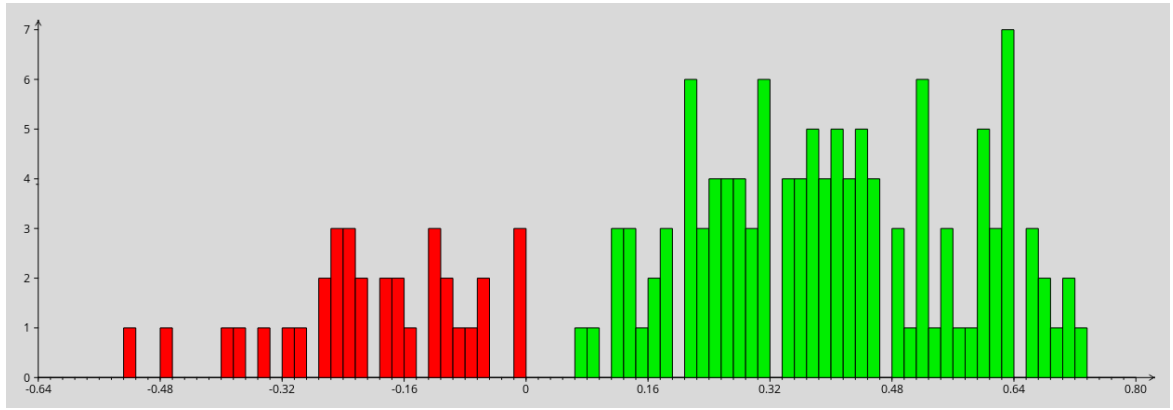
TAL TECH

Benchmarks assessed

- **AES_v1 (WNS=100ps)**
- **AES_v2 (WNS=200ps)**
- **AES_v3 (WNS=100ps)**
- PRESENT
- **CAST (WNS=500ps)**
- MISTY
- Camellia

- openMSP430
- SEED **(WNS=500ps)**
- TDEA

- But why these designs?
  - Most are ciphers
  - Some are fast (1GHz target frequency)
  - Some are very small
  - Some are hard to route (10 metal stack)
  - Some had **timing violations**

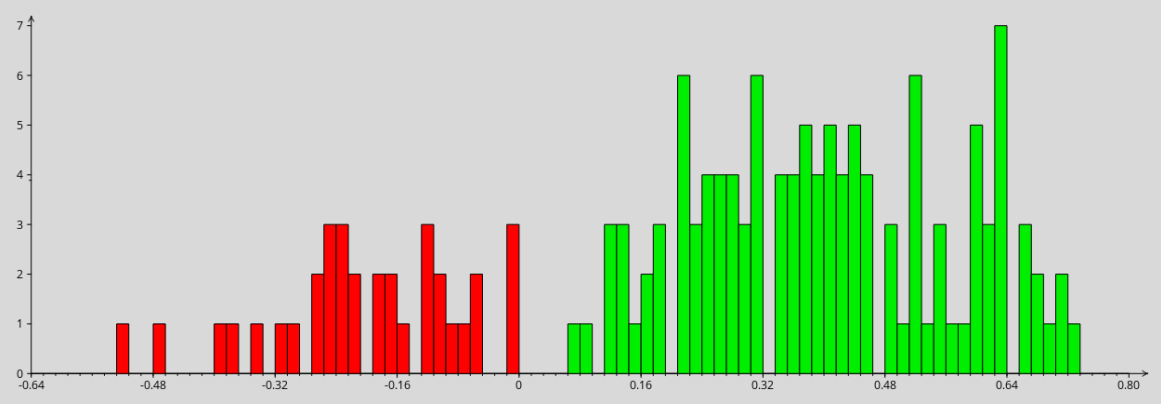


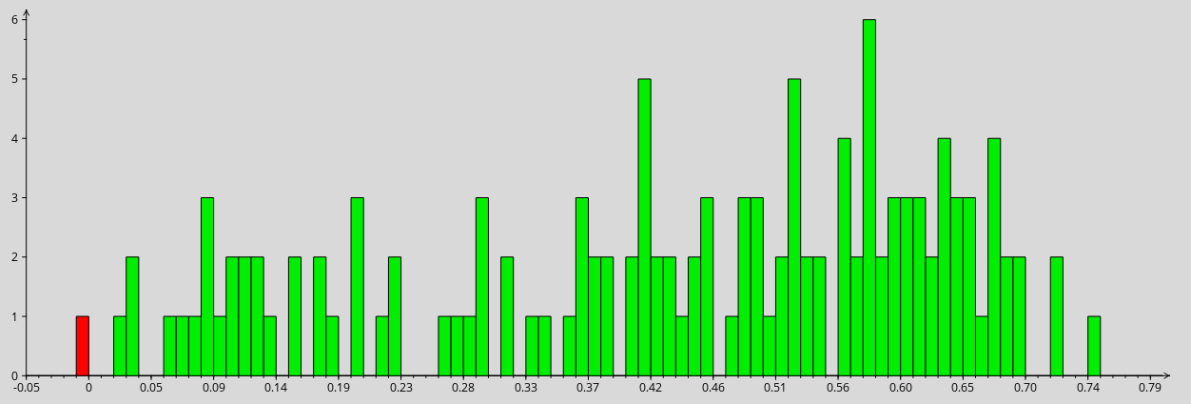
Details about CAST:
Clock = 100MHz
FEP = 33 paths
WNS = 0.519ns
TNS = 6.693ns

TAL TECH

Strategies

- Observation: the designs are not “good”
- Fix design problems first, security problems second

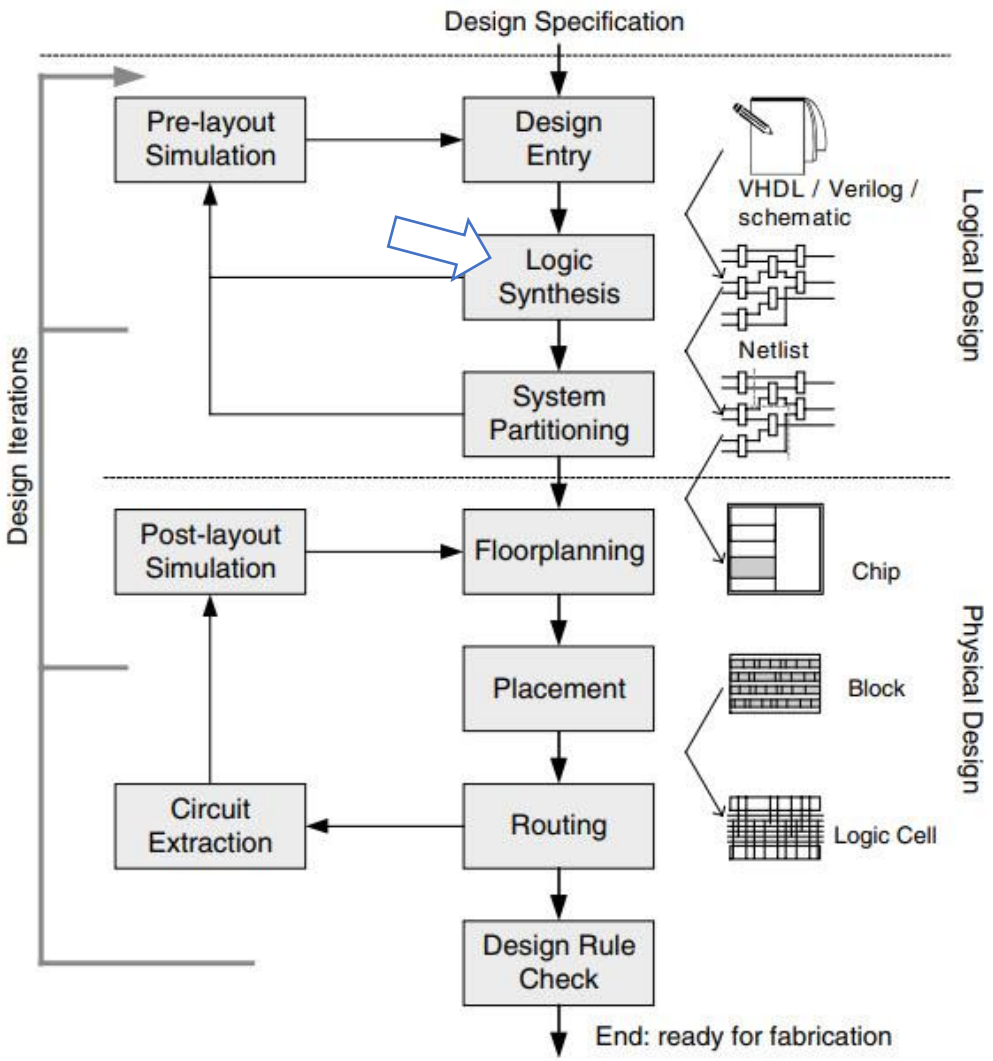




- We tried many things, not all ideas worked…
  - Logic synthesis
  - Physical synthesis
  - Security

TAL TECH

Logic synthesis strategies

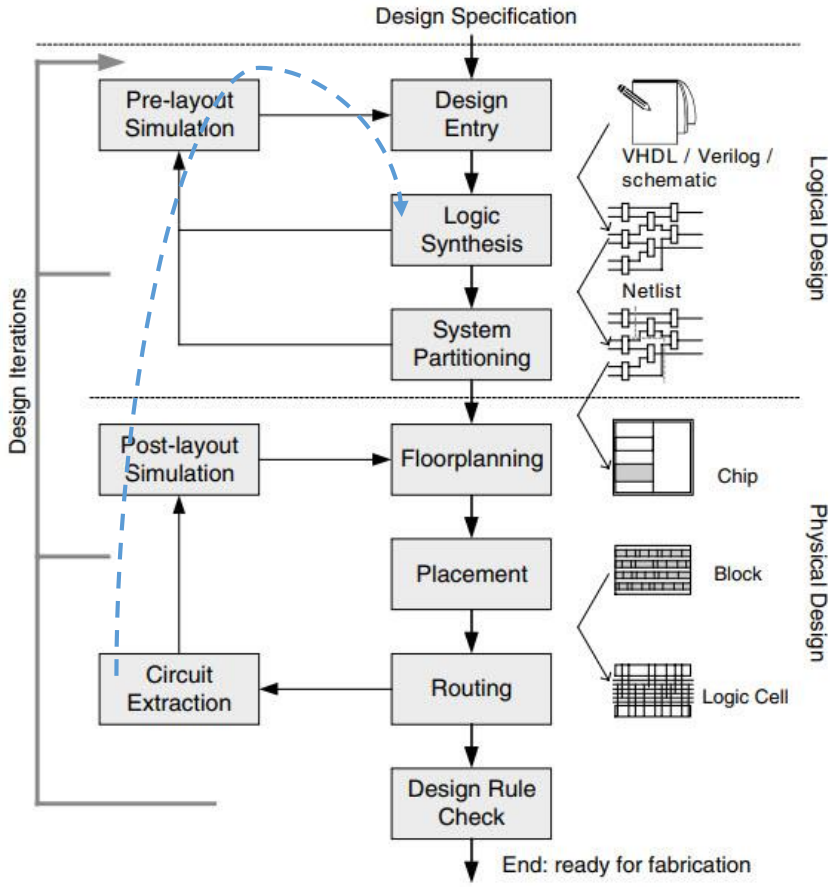



```
1  /*
2  ##################################################################
3  #  Generated by:      Cadence Innovus 16.15-s078_1
4  #  OS:                Linux x86_64(Host ID aduae260-lap)
5  #  Generated on:      Fri Jan 14 19:05:50 2022
6  #  Design:            top
7  #  Command:           saveNetlist -excludeLeafCell outputs/design_original.v
8  ##################################################################
9  */
10 //////////////////////////////////////////////////////////////
11 // Created by: Synopsys DC Ultra(TM) in wire load mode
12 // Version   : M-2016.12-SP2
13 // Date      : Fri Jan 14 18:14:29 2022
14 //////////////////////////////////////////////////////////////
15 module top (
16         data_out,
17         data_valid,
18         key_valid,
19         busy,
20         clk,
21         nreset,
22         data_rdy,
23         key_rdy,
24         EncDec,
25         data_in);
```

Source: https://www.electronicshub.org/introduction-to-asic-technology/

Logic synthesis strategy - resynthesis

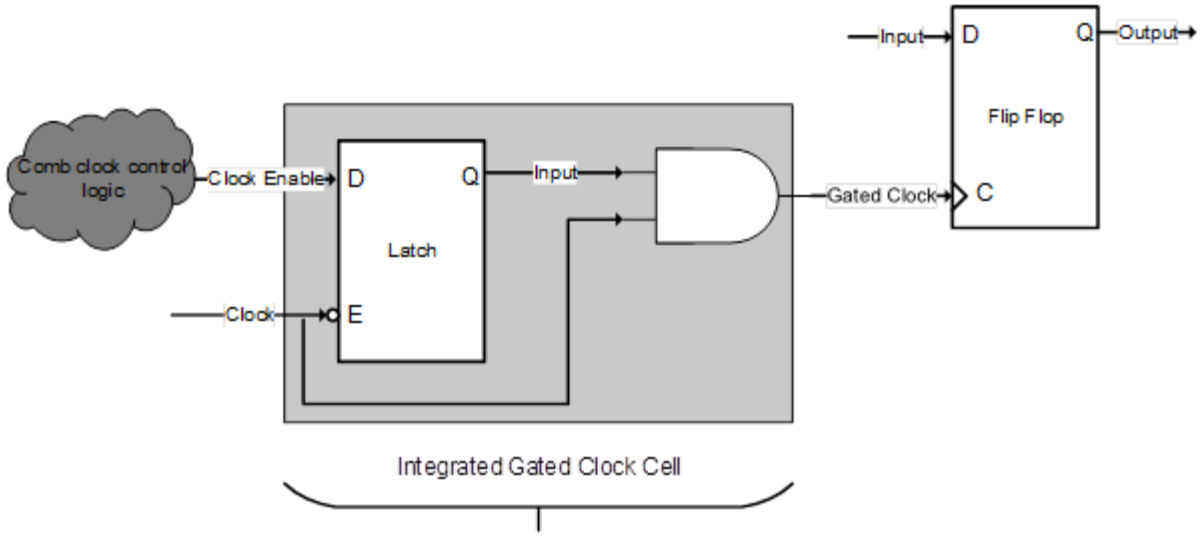
- Caveat: no RTL available
- Extract netlist from layout, use it as input to logic synthesis
- **Failed**: this is backwards. Design is already buffered up, clock tree is already present…
- **Failed**: cell assets and net assets had to be marked *dont_touch*

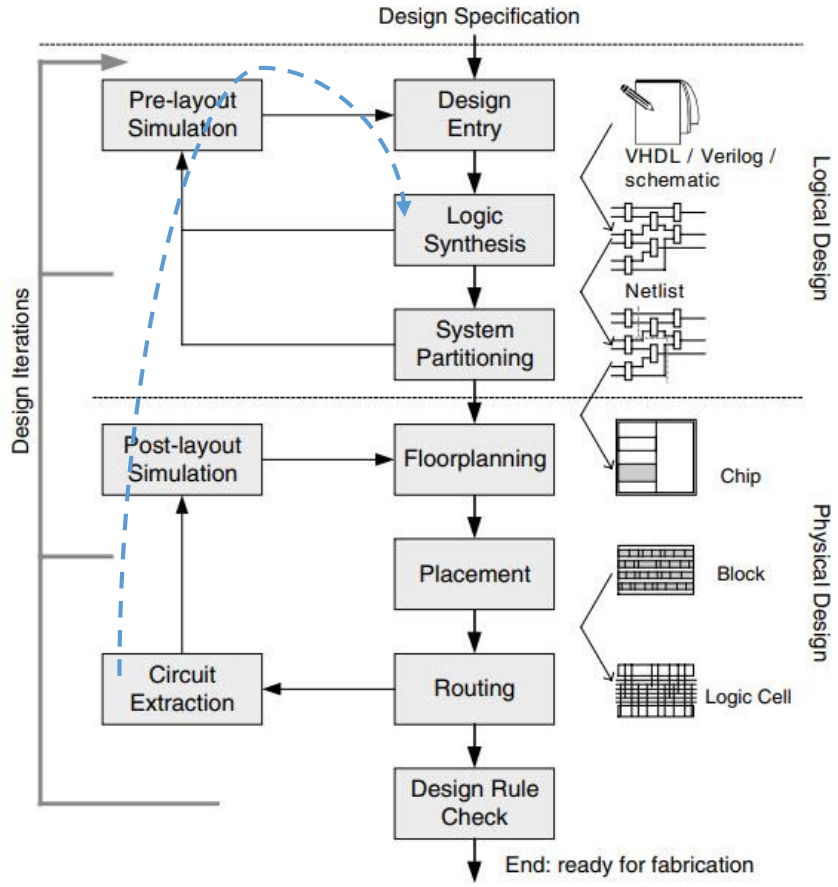


Source: https://www.electronicshub.org/introduction-to-asic-technology/

Logic synthesis strategy – clock gating

- Observation: netlists had no CG, but standard cell library has CG-specialized cell
- Cell assets and net assets had to be marked *dont_touch…* but maybe it's ok
- **Failed**: designs are considered non-equivalent

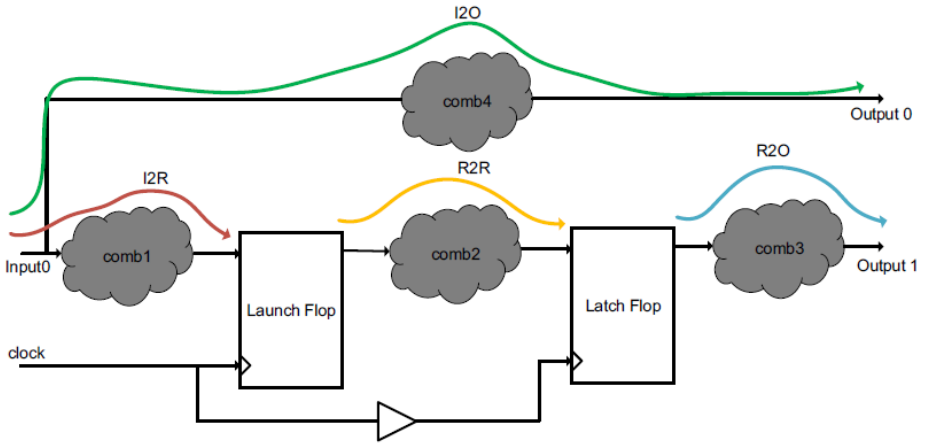


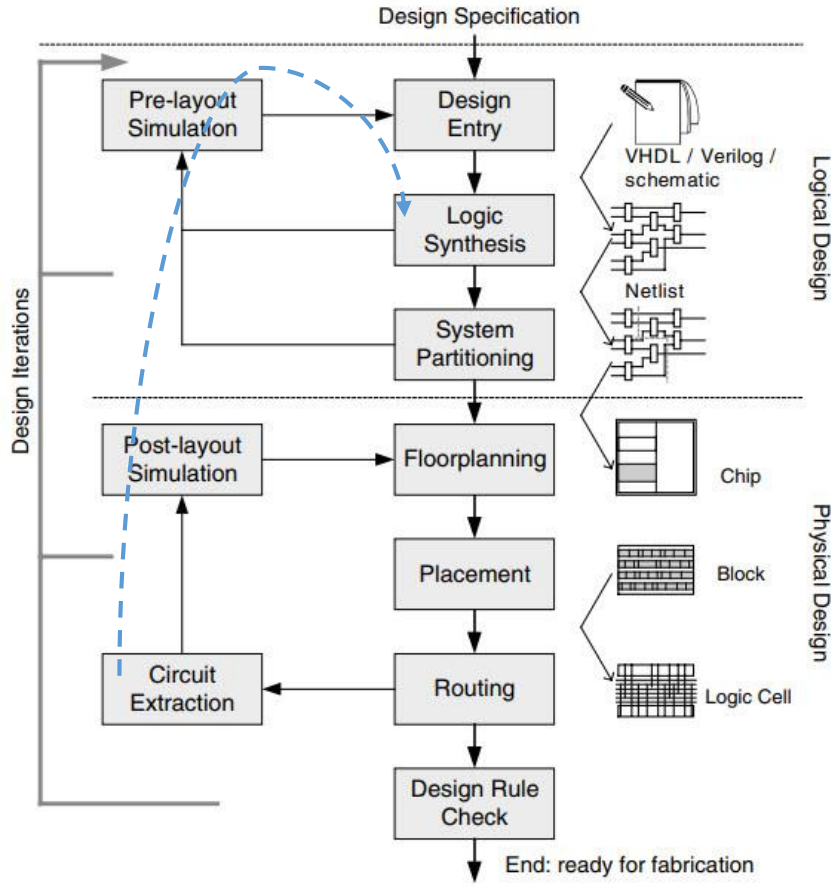


Source: https://anysilicon.com/the-ultimate-guide-to-clock-gating/

Logic synthesis strategy - retiming

- Observation: some designs (Camellia, Misty, TDEA) had comfortable reg-to-reg paths but tight reg-to-out timing
- **Failed**: this is backwards. Design is already buffered up, clock tree is already present…
- **Failed**: cell assets and net assets had to be marked *dont_touch*





Source: https://anysilicon.com/the-ultimate-guide-to-clock-gating/

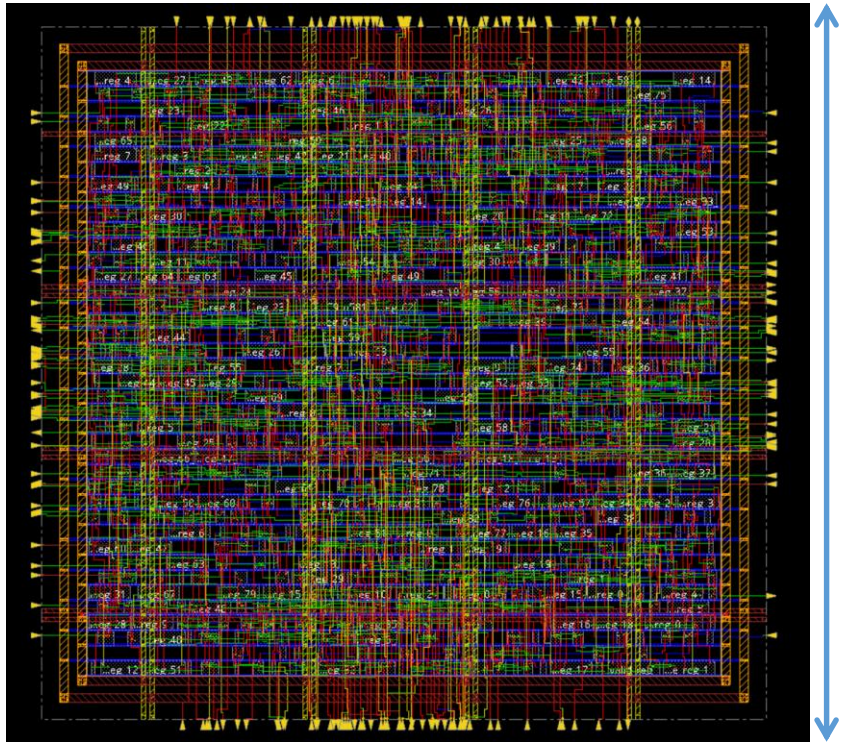
Physical synthesis strategies

- **First phase**: improving design
  - Shrinking block size
  - Fixing timing violations
  - Improving CTS and routing scripts
- **Second phase**: improving security metrics with generic tactics
  - Routing all net assets underneath other nets
  - CTS with huge metal width
  - Placing cell assets under power grid stripes
- **Third phase**: fine tuning security metrics
  - Leveraging ECO features to hide net assets
  - Leveraging ECO features to fill empty sites
  - Manually repositioning cells for diminishing exploitable areas
- **Final phase**: manual fixes to improve security
  - Fixing gaps by manually replacing cells
  - Adding buffers manually to fill gaps
  - Manual shield drawing for hiding net or cell assets

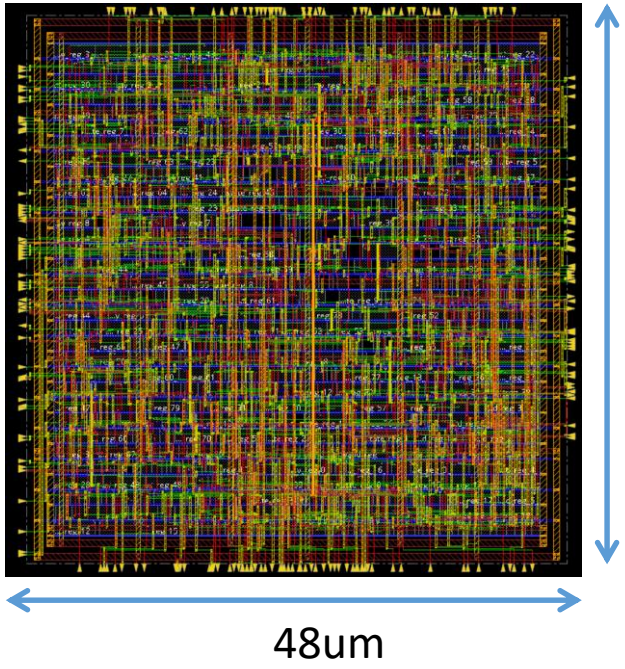
TAL TECH

First phase – Design Cost Improvements

Baseline



Our solution



PRESENT

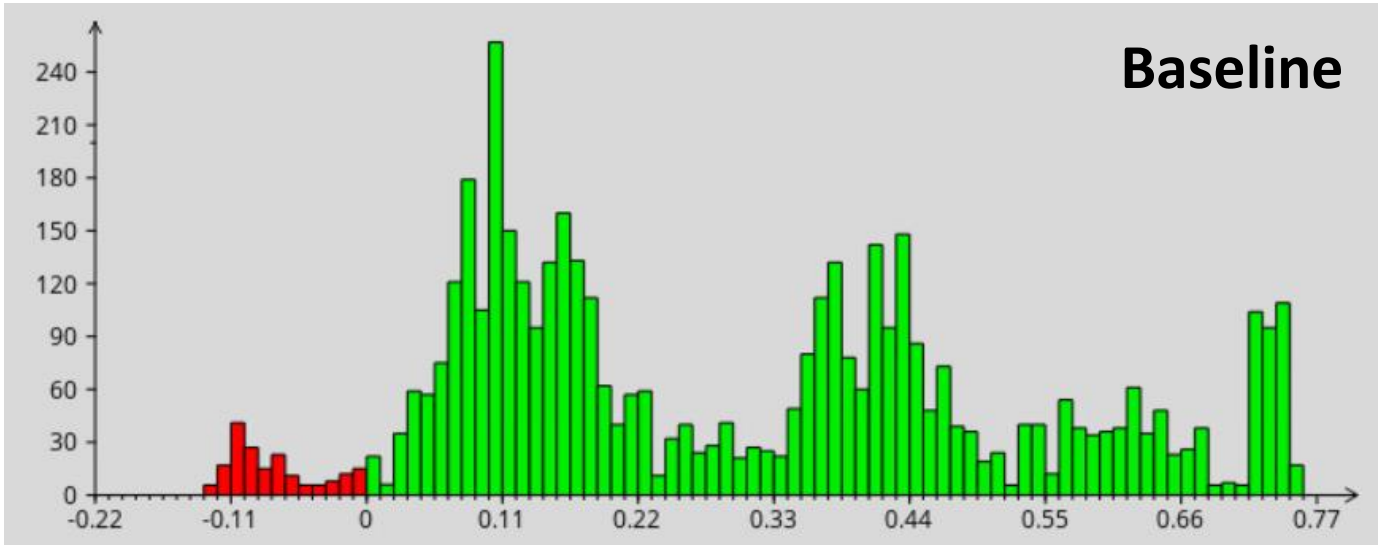
Area reduced from 4225um^2 to 2304um^2

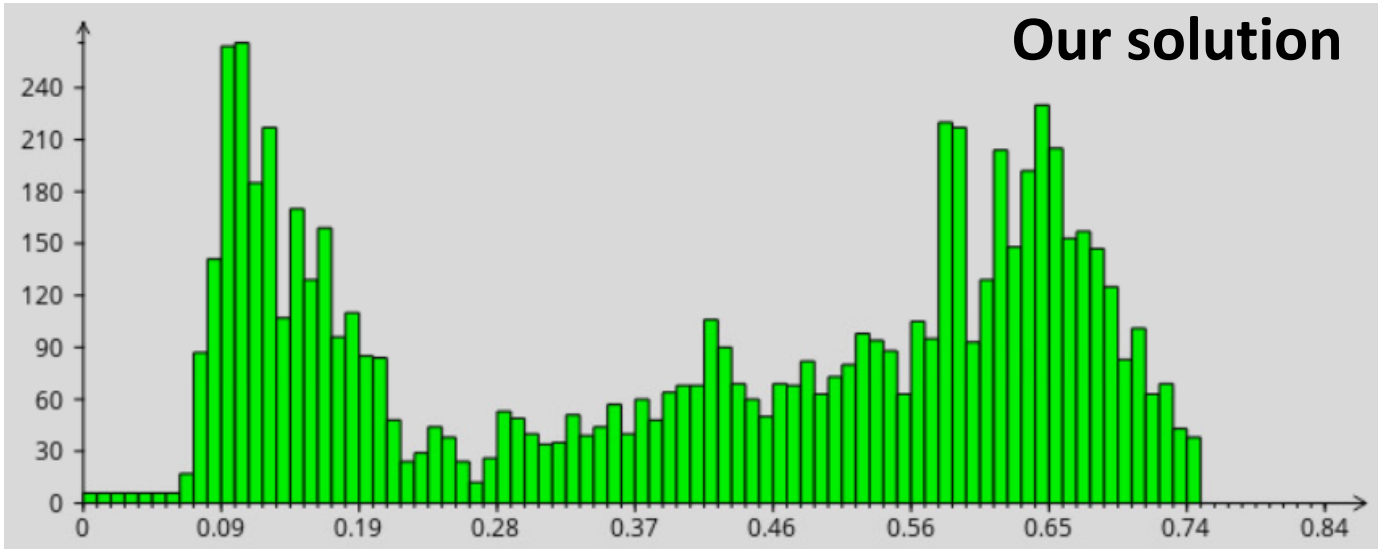
54% of original area!!!

Floorplan shrinkage –> improved CTS properties –> improved timing -> improved power
Compromise on routing density

TAL TECH

First phase – Design Cost Improvements





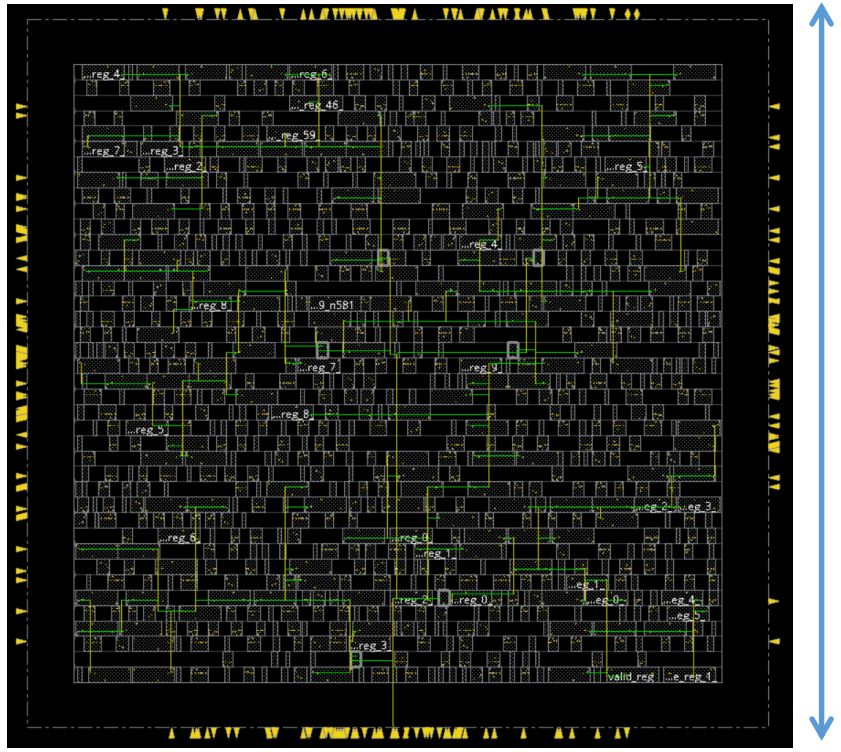
PRESENT

Physical Synthesis flow improved

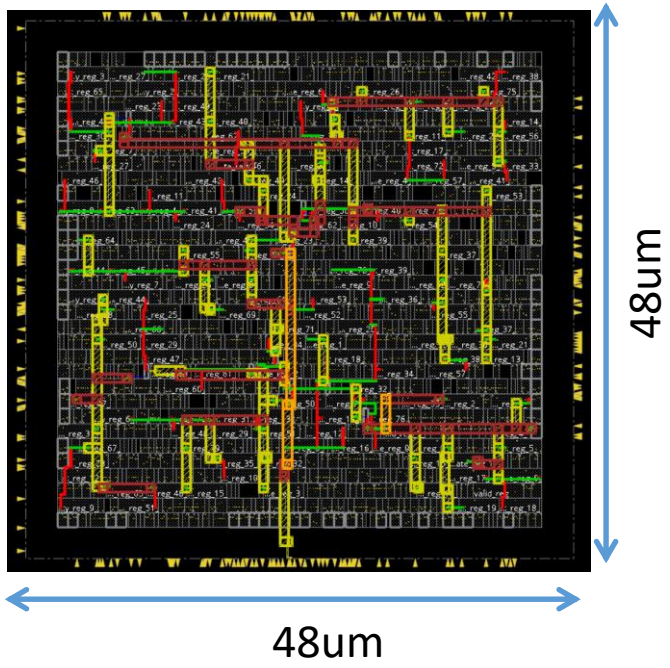
No timing violations!!

Second phase – CTS with Non Default Rules

Baseline



Our solution



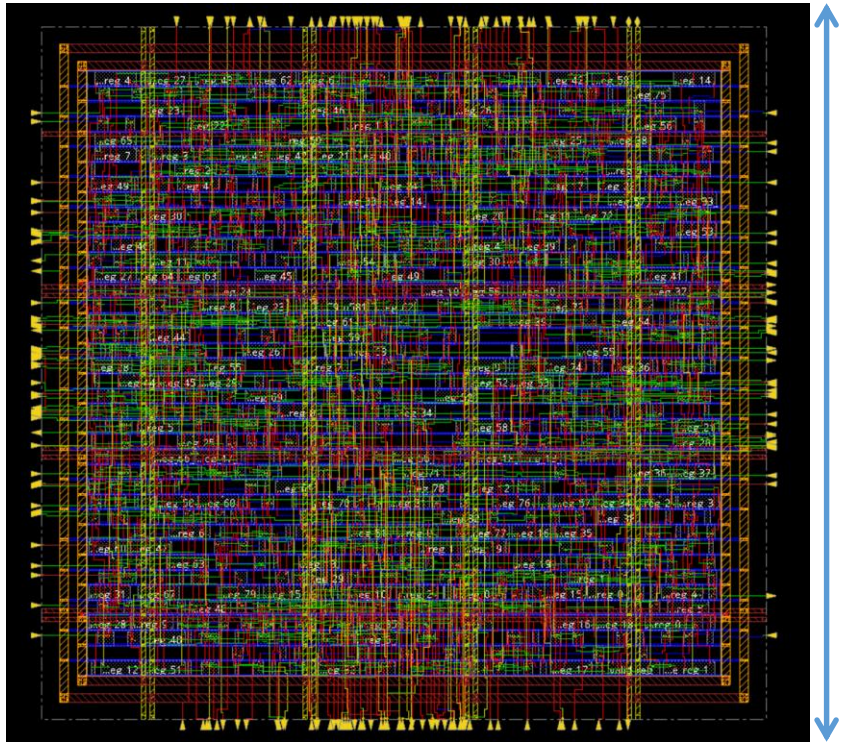
PRESENT

Large wires for improving CT quality

Chance to increase congestion, in consequence covering cells and nets assets.

Second phase – Routing with Non Default Rules

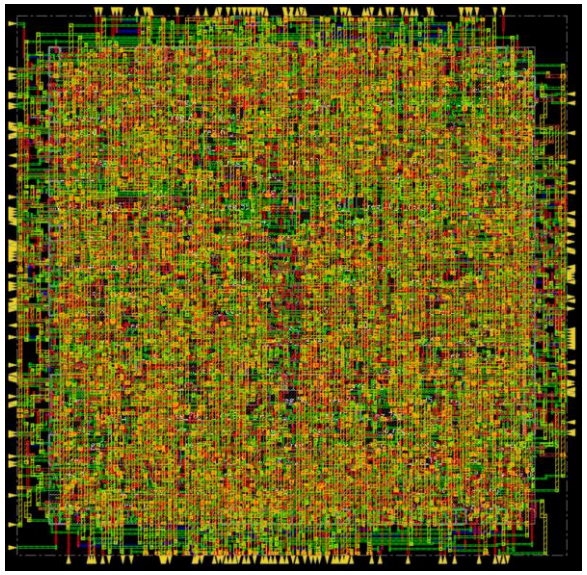
Baseline



65um

65um

Our solution



48um

48um

PRESENT

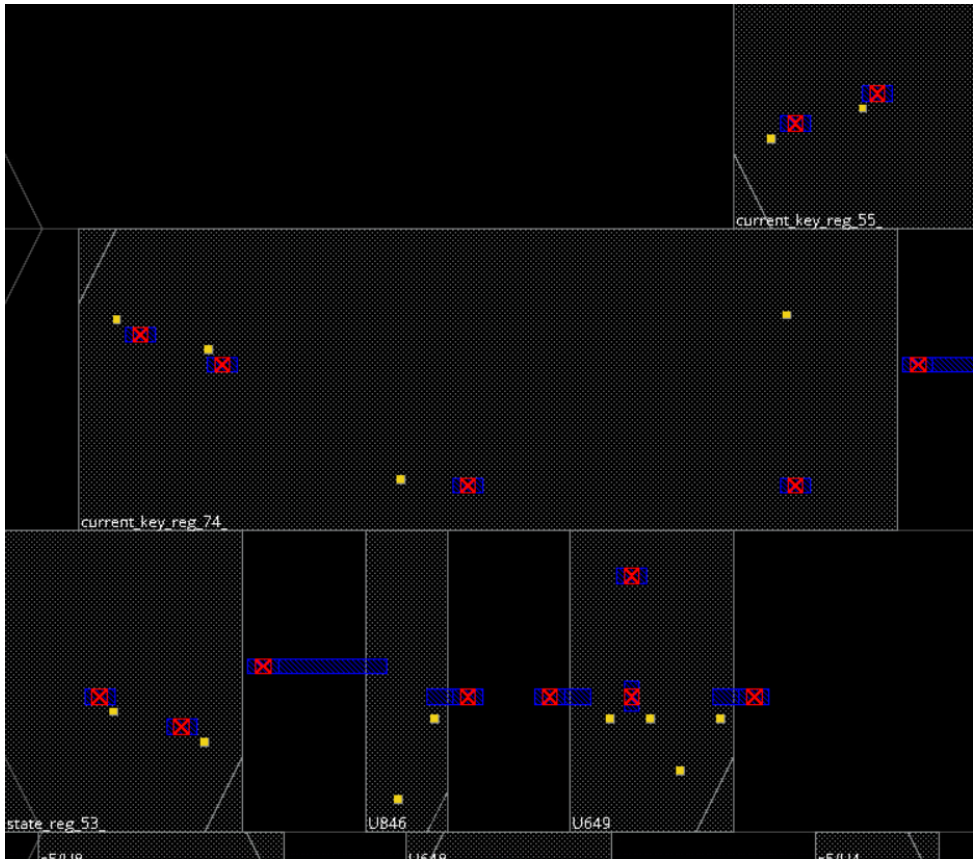
- Non-asset nets routed with large width
- Asset nets routed at lower metal possible

Increase coverage of cell assets, and, decrease the chance of exposing net assets

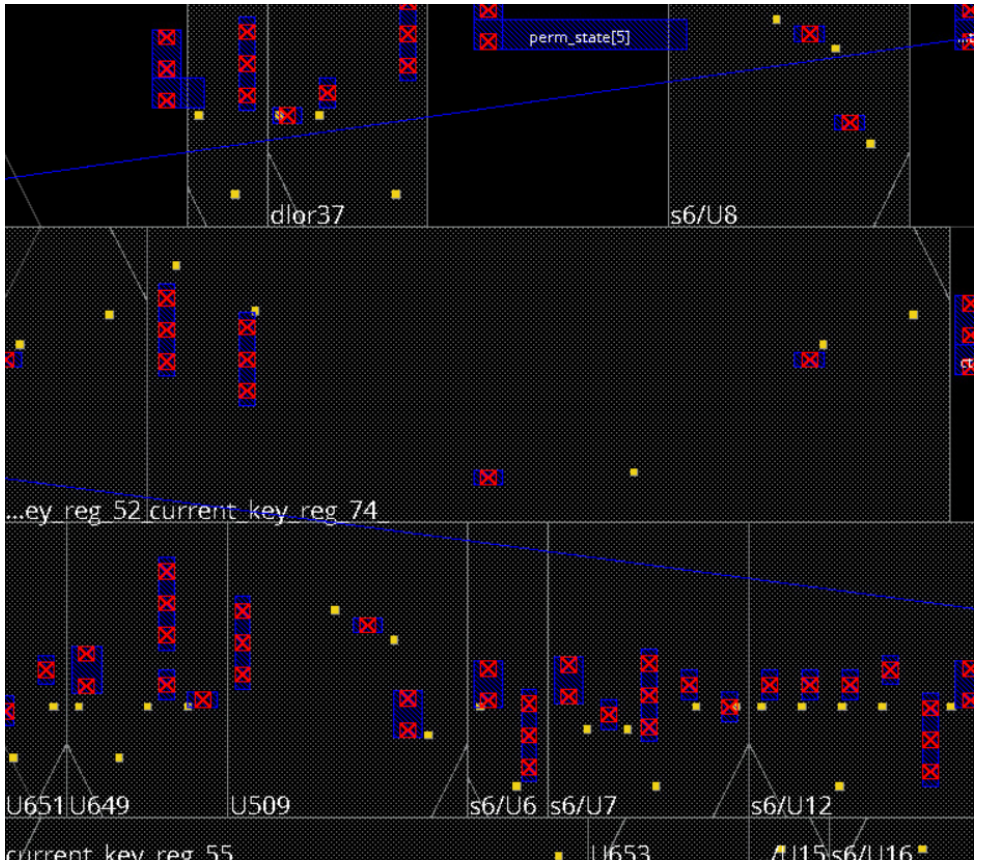
TAL TECH

Second phase – Connecting Pins with Multi Cut Vias

Before



After



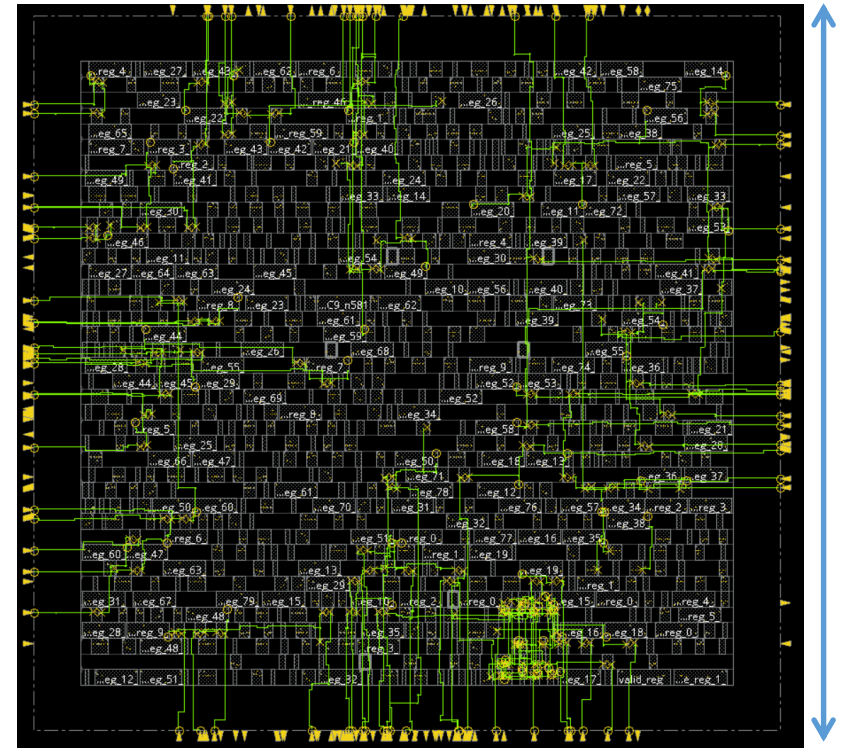
Utilizing multi cut vias for connecting M1 to M2

Multi cut vias means a bigger piece of metal routed on top of cell assets, which improves cell assets coverage

TAL TECH

Third phase – Manual placement of cells

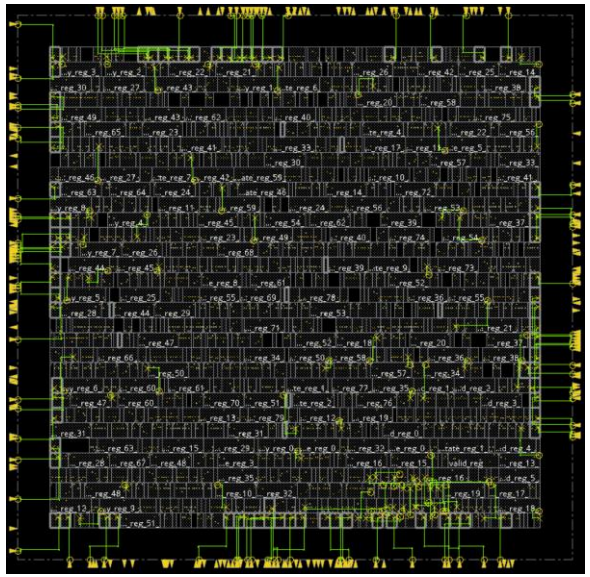
Baseline



65um

65um

Our solution



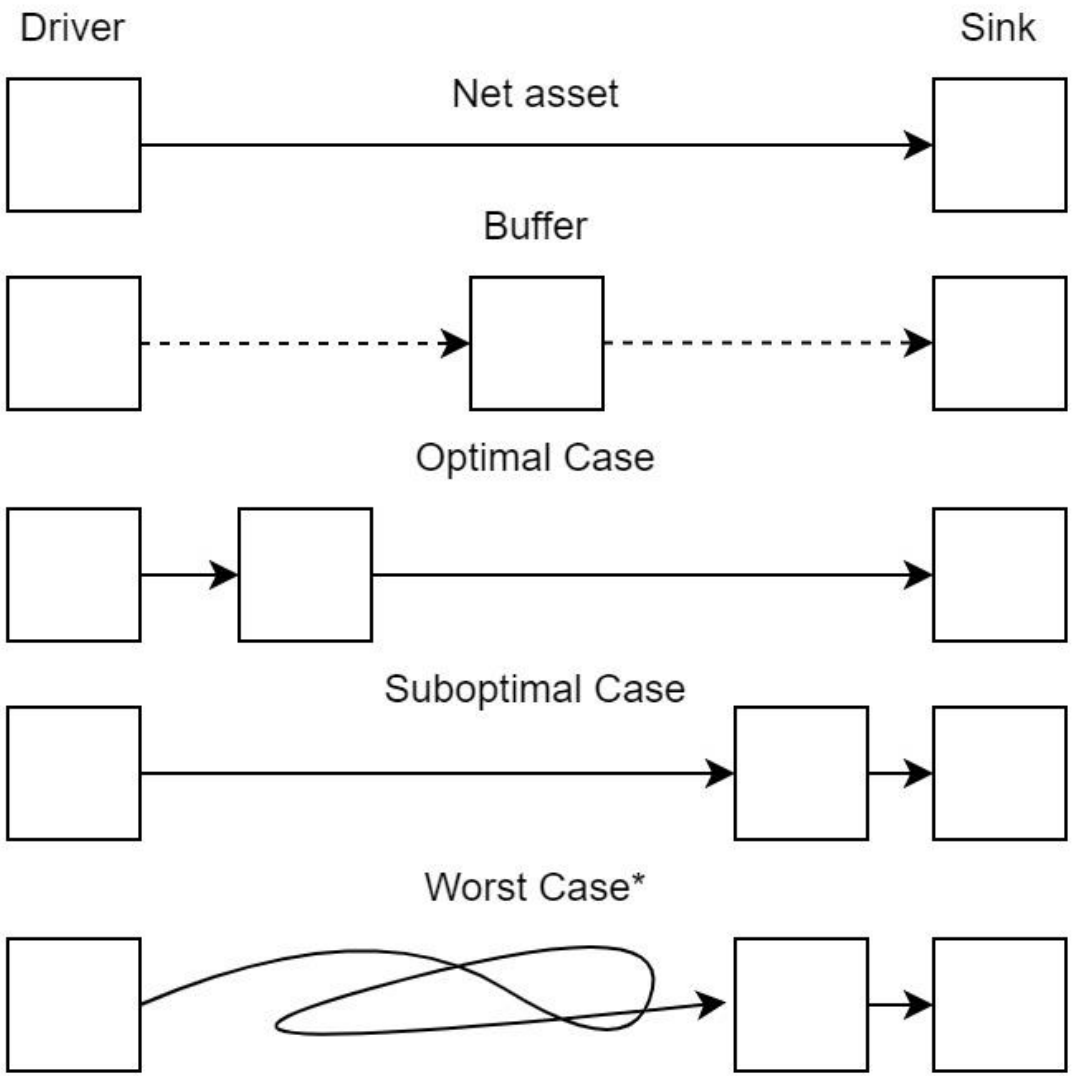
48um

48um

For designs with net assets with external connections, placing their sinks near the IO helps to shorten its wire length

Short wires = easier to hide

Third phase – Addition of Buffers

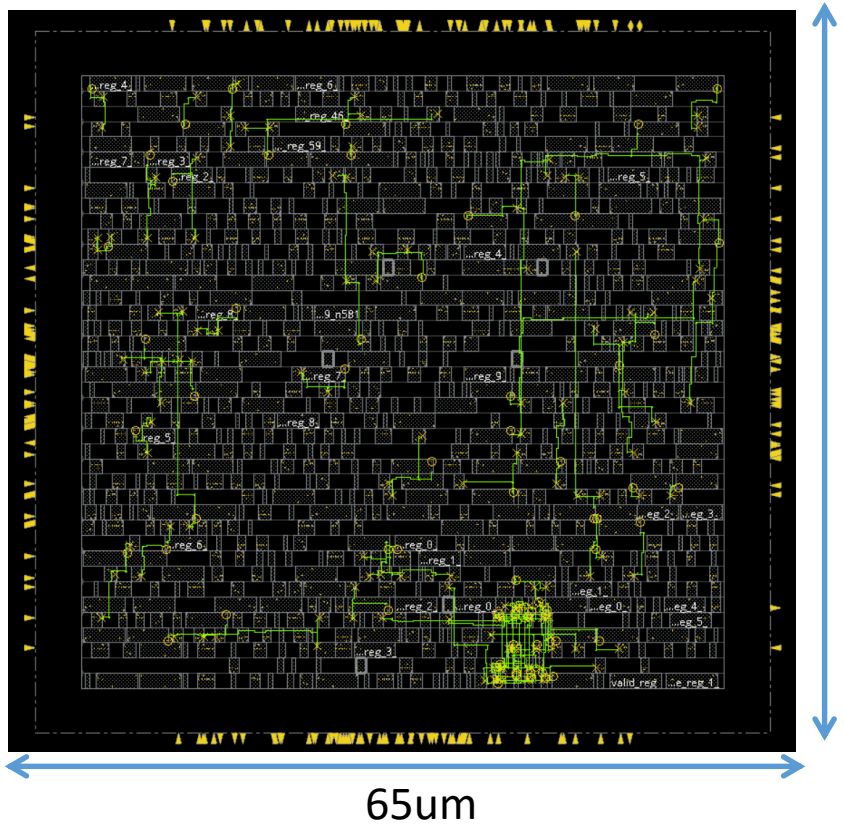


Addition of buffer in front of net assets drivers

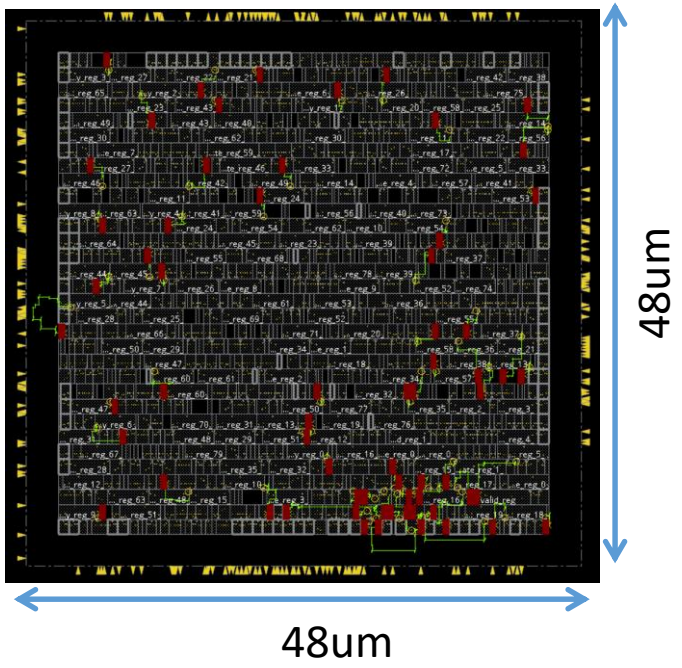
Chance to reduce net assets length, trade-off w/ power

Third phase – Addition of Buffers

Baseline



Our solution



Addition of buffer in front of net assets drivers

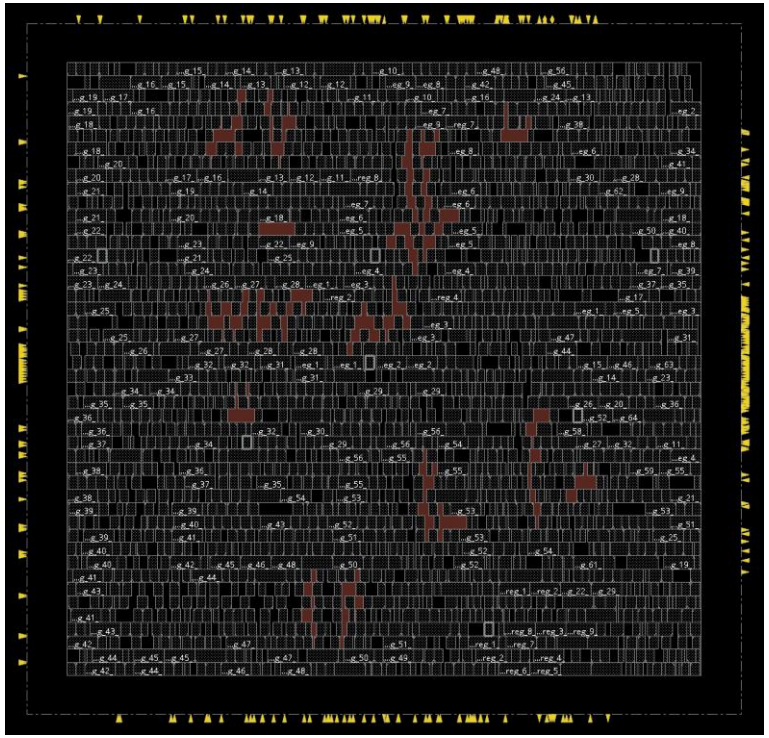
Chance to reduce net assets length, trade-off w/ power

Final phase – Manual Fixes

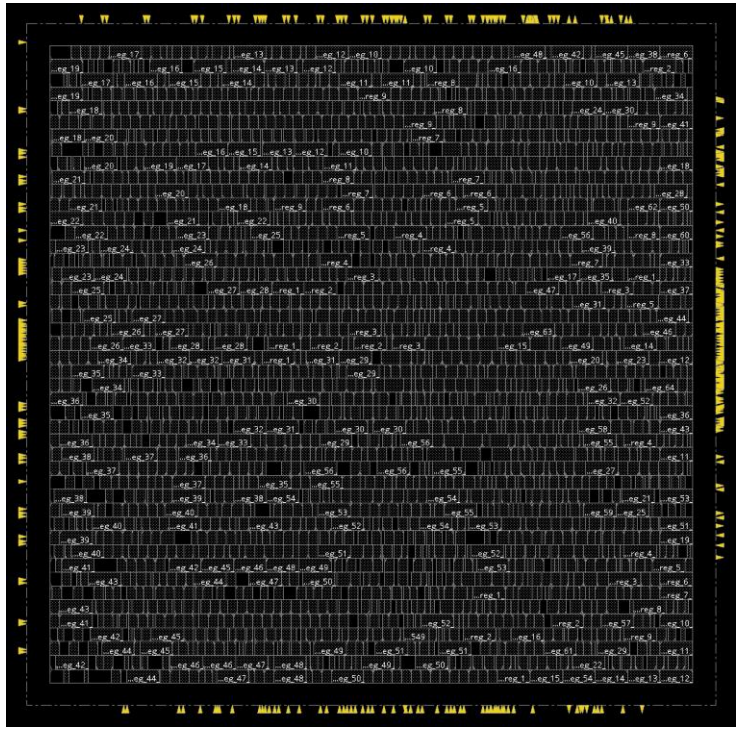
TDEA

Regions with >= 20 continuous sites are considered *exploitable regions* for Trojan insertion

Before



After

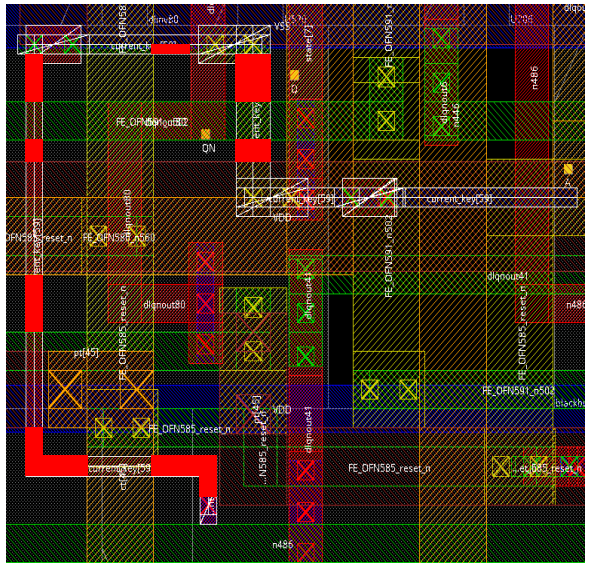


- Buffers were added for filling the gaps
- Cells moved (shifted to the right or left in most cases) to break the large gaps into smaller ones

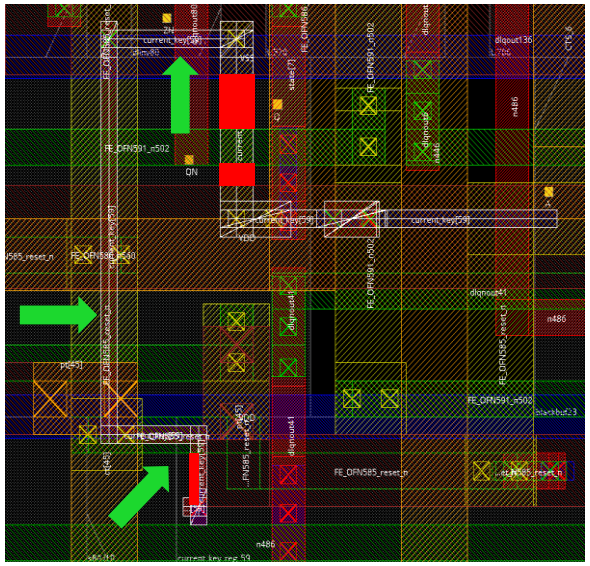
Zero exploitable regions in the final design

Final phase – Manual Fixes

Before



After



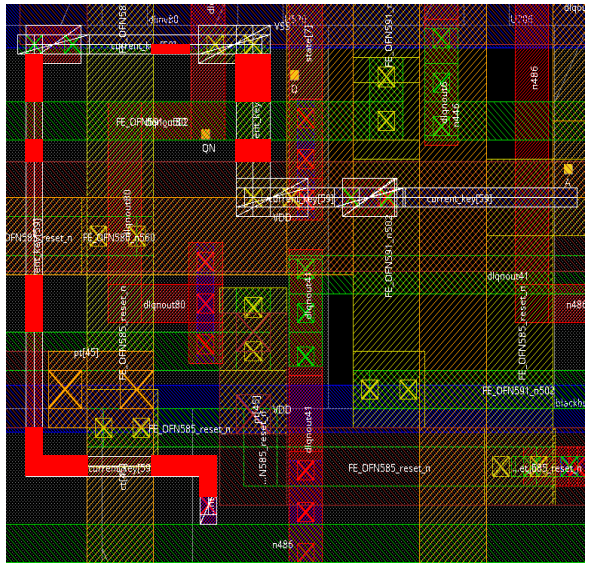
Example of net detour by rerouting the net

Detour the net assets so that they can be hidden under the upper layers.

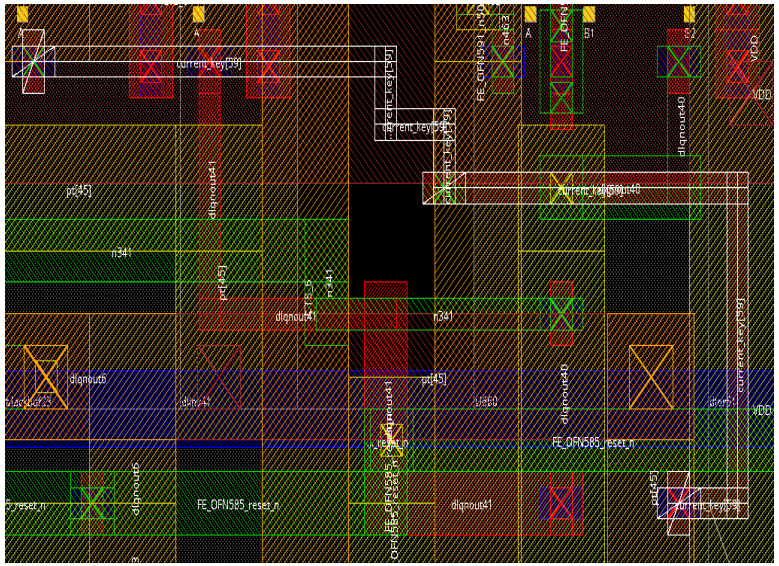
The total exposed area of the net asset is reduced

Final phase – Manual Fixes

Before



After



Example of net detour by changing the driver cell orientation. This changes the position of the cell pin, which forces the routing to be in a difference direction. Replacing the cell can have a similar effect

Detour the net assets so that they can be hidden under the upper layers.

Some of the net assets can be entirely covered!

End of part 1

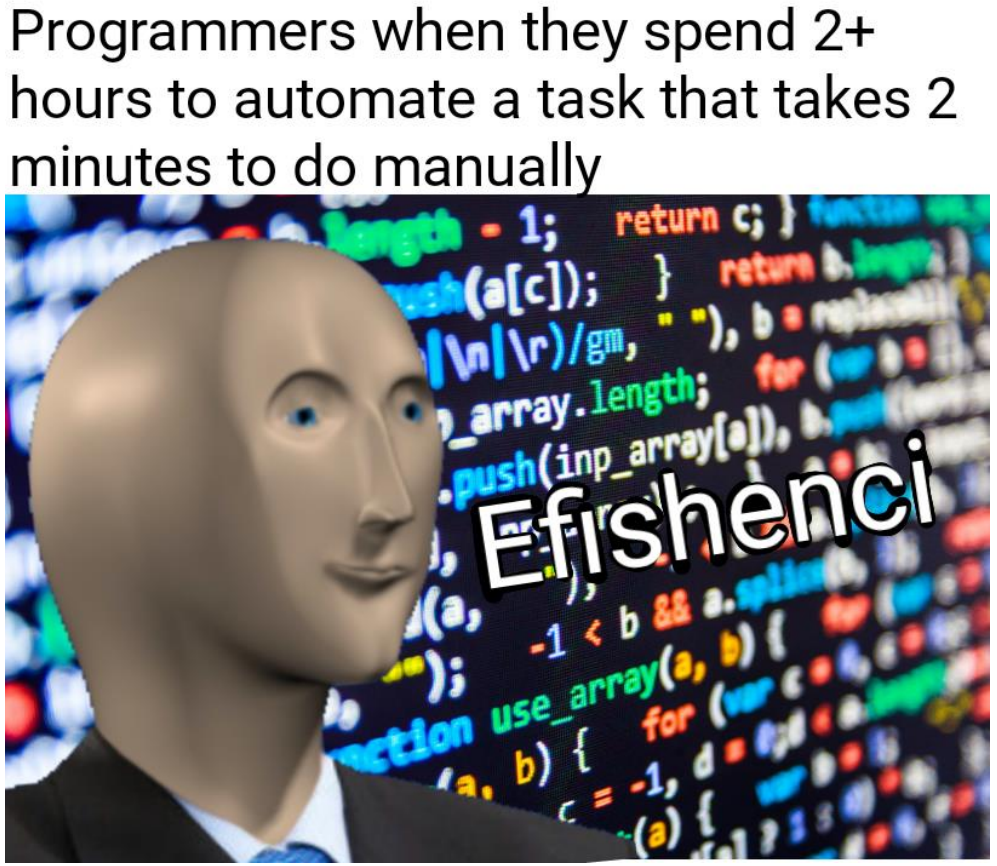
- Students take over from here!

Other techniques – Abandoned!

- Physical synthesis
  - Placing the cell assets under the **PG stripes** – power stripes were not thick enough

Other techniques – Abandoned!

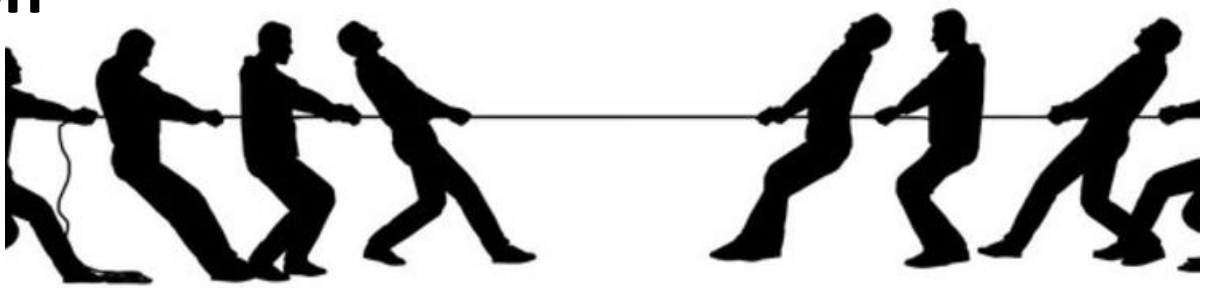
- Physical synthesis
  - **Automatically** adding buffers for net shrinkage or filling gaps – generates residual DRCs impossible to solve



Other techniques – Abandoned!

- Physical synthesis
  - Setting the **max length** allowed to a small number in order to force short nets and more congestion – designs became unrouteable for the high density that we wanted

Congestion



Density

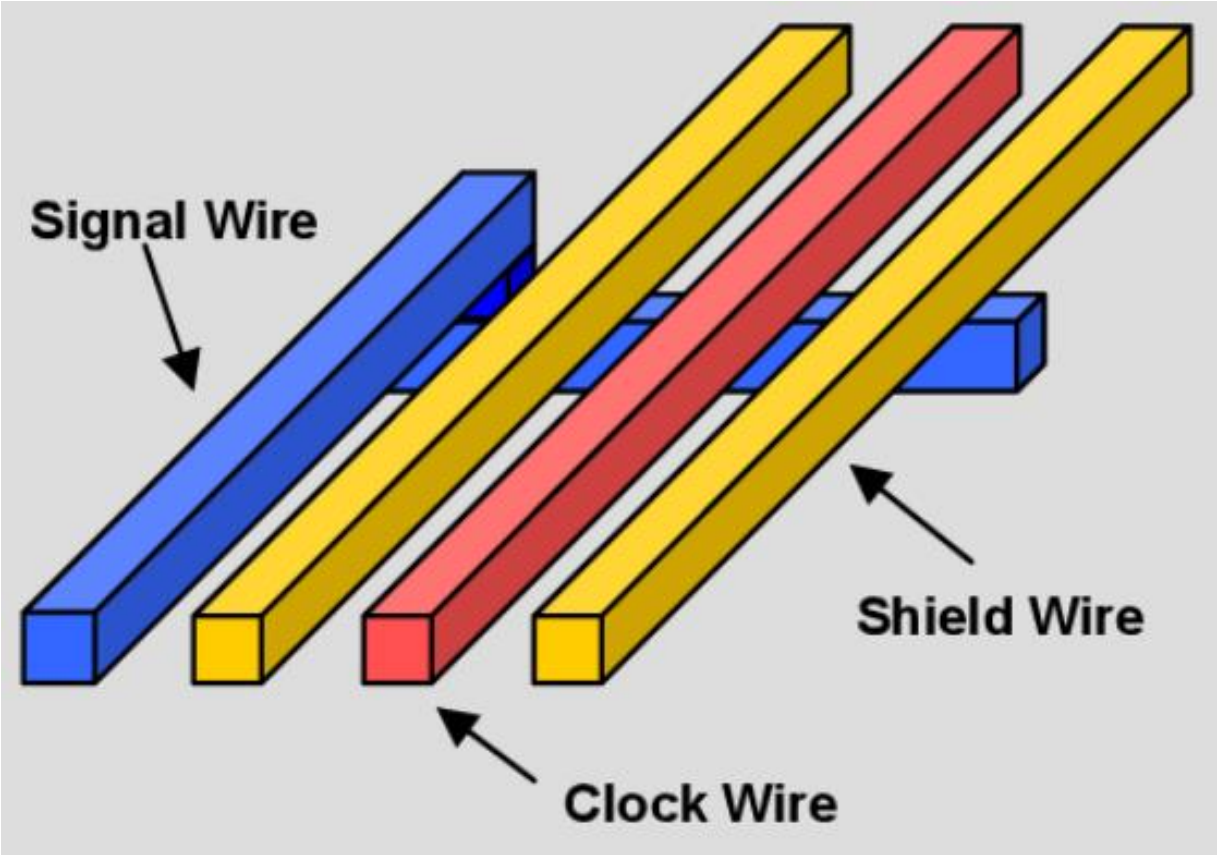
Other techniques – Abandoned!

- Physical synthesis
  - Disabling **FF optimization** while resizing all the FF to the minimum driving strength – it did not help reducing power consumption and created timing problems

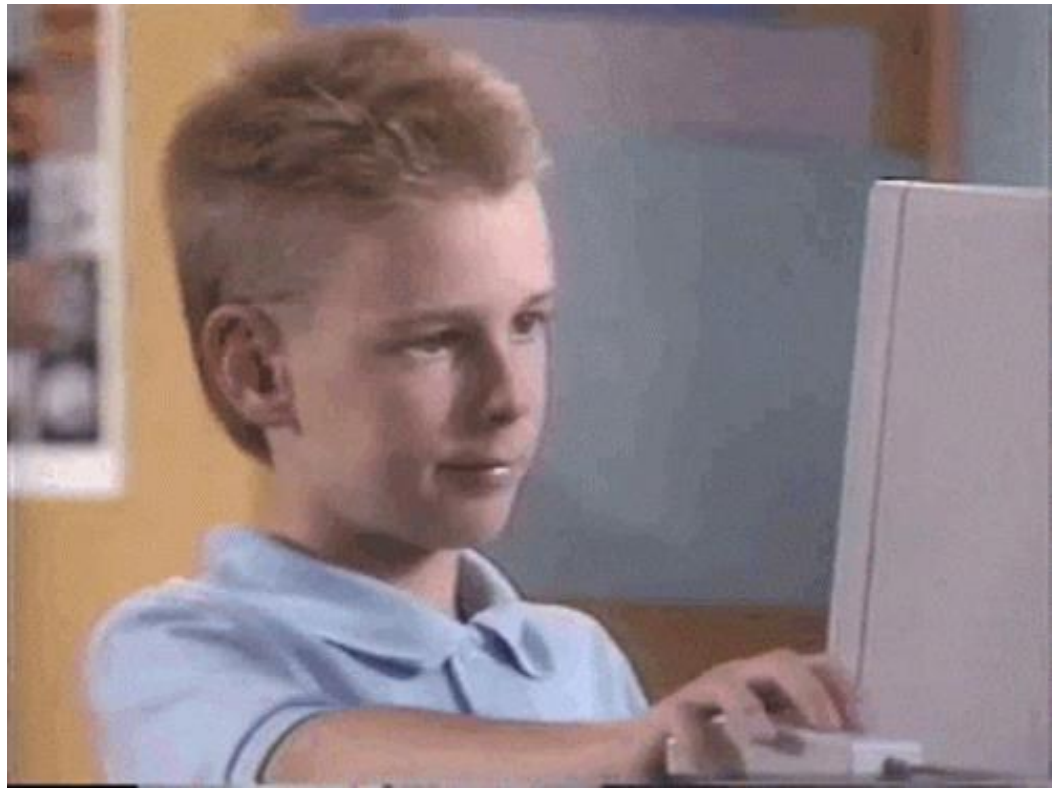
TAL TECH

Other techniques – Abandoned!

- Physical synthesis
  - **Shielding** net assets with external connections (I/O) with power nets – later in the contest the amount of metal for power nets was constrained



Champions of the alpha round – Team K!



	AES_1	AES_2	AES_3	Camellia	CAST	MISTY	PRESENT
*	1.000000	1.000000	1.000000	0.750000	1.000000	0.750000	0.750000
A	0.156474	0.942725	0.182120	0.180730	0.103810	0.067178	0.123786
B	--------	--------	--------	--------	--------	--------	--------
C	--------	--------	--------	--------	--------	--------	--------
D	--------	--------	--------	--------	--------	--------	--------
E	0.236819	0.222501	0.069017	0.070125	0.032580	0.033190	0.052571
F	--------	--------	--------	--------	--------	--------	--------
G	--------	--------	--------	--------	--------	--------	--------
H	--------	--------	--------	--------	--------	--------	--------
I	--------	--------	--------	--------	--------	--------	--------
J	0.235452	0.915177	2.606206	0.606139	0.365939	0.467680	3.714756
K	0.020040	0.085066	0.175733	0.027968	0.051747	0.046060	0.018165
L	0.632226	0.731865	0.587291	0.747774	0.735023	0.747292	0.734942
M	--------	--------	--------	--------	--------	--------	--------
N	0.051073	0.308361	0.801496	0.155475	0.085726	0.075256	0.064884
O	0.408307	0.532993	0.501970	0.229697	0.320193	0.143594	0.208114
P	--------	--------	--------	--------	--------	--------	--------
Q	0.846710	1.025901	1.018366	0.757782	1.007793	0.756043	0.753972

TAL TECH

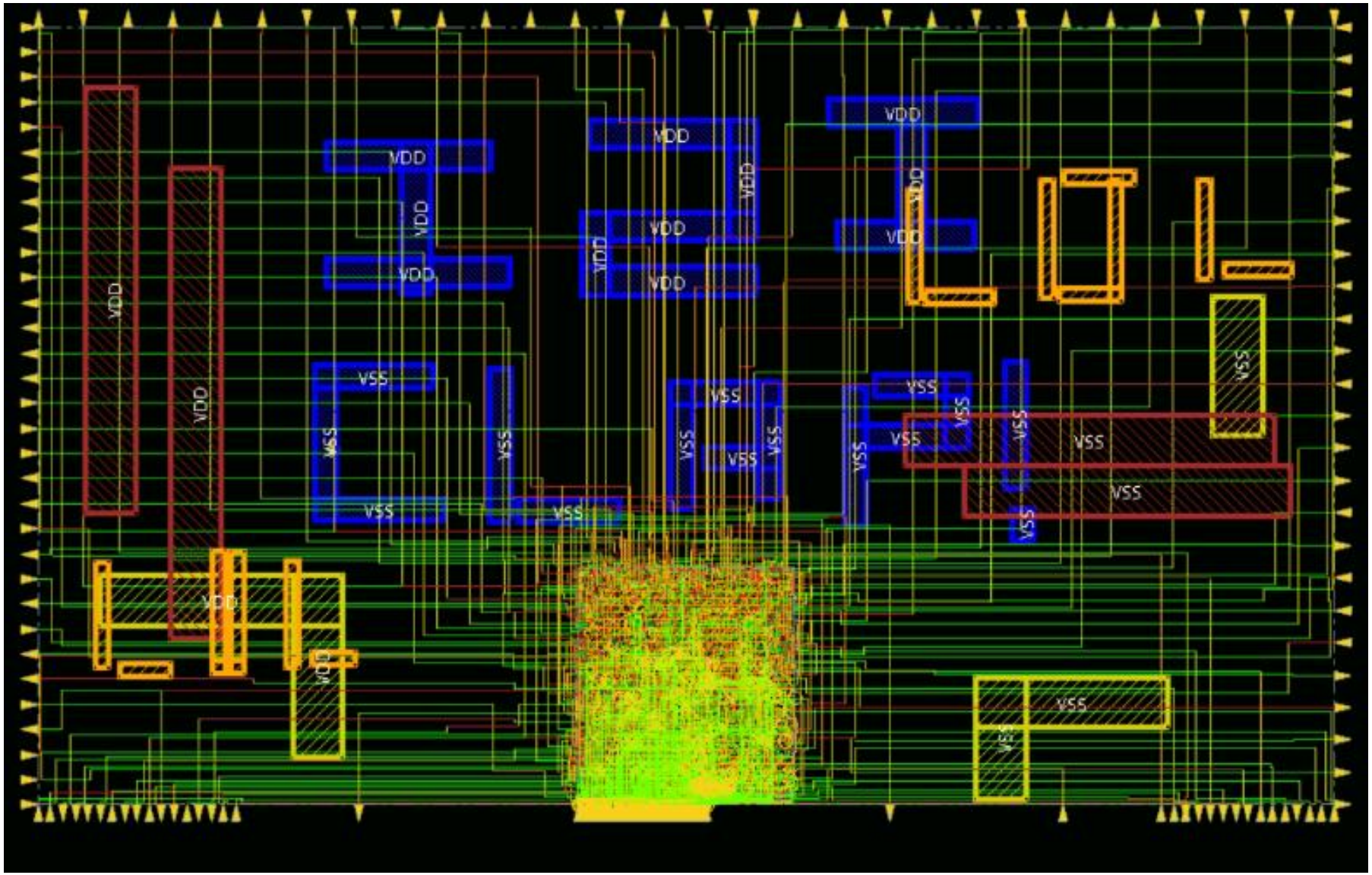
Dark Friday
The Zeros and Despair

Team/Benchmark	Baseline	J	N	O	E	L	A	Q	K
AES_1	1.000000	0.764884	0.299508	0.008447	0.041552	0.271596	0.000001	0.194594	0.052321
AES_2	1.000000	1.687749	0.514212	0.010548	0.101933	0.324694	0.430016	0.101044	0.104176
AES_3	1.000000	1.332768	0.497878	0.003901	0.056400	0.295023	0.000001	0.000001	0.061247
Camellia	0.750000	0.676397	0.260423	0.017719	0.093440	0.749726	0.000000	--------	0.122173
CAST	1.000000	1.687787	0.244816	0.016850	0.087135	0.751540	0.000001	--------	0.128377
MISTY	0.750000	3.178107	0.255207	0.002300	0.055931	0.749526	--------	--------	0.081914
openMSP430_1	0.750000	0.841673	0.322593	0.009447	0.105195	0.554981	0.554621	0.000001	0.170911
PRESENT	0.750000	0.629633	0.289205	0.001957	0.079465	0.749990	0.110556	0.000498	0.043454
SEED	1.000000	2.203857	0.316475	0.000001	0.086032	0.775886	0.000001	--------	0.174475
TDEA	0.750000	0.596819	0.350483	0.008851	0.126647	0.478162	0.107474	0.002950	0.111408

SEC x DES

Zero? How??

Despair layout



Example of a perfect score layout

Final round - results

Team/Benchmark	Baseline	J	N	O	E	L	A	Q	K
AES_1	1.000000	0.764884	0.025684	0.000000	0.000000	0.271596	0.000000	0.000001	0.000000
AES_2	1.000000	1.687749	0.054186	0.000000	0.000000	0.324694	0.000000	0.000001	0.000000
AES_3	1.000000	1.332768	0.000001	0.000000	0.000000	0.295023	0.000000	0.000001	0.000000
Camellia	0.750000	0.676397	0.000001	0.000000	0.000000	0.281597	0.000000	0.000001	0.000000
CAST	1.000000	1.687787	0.000001	0.000000	0.000000	0.300895	0.000000	0.000001	0.000000
MISTY	0.750000	3.178107	0.000001	0.000000	0.000000	0.254930	0.000000	0.000001	0.000000
openMSP430_1	0.750000	0.841673	0.000000	0.000000	0.000000	0.344685	0.000000	0.000001	0.000000
PRESENT	0.750000	0.629633	0.000001	0.000000	0.000000	0.319908	0.000000	0.000498	0.000000
SEED	1.000000	2.203857	0.000001	0.000000	0.000000	0.207375	0.000000	0.000001	0.000000
TDEA	0.750000	0.596819	0.003351	0.000000	0.000000	0.246417	0.000000	0.002950	0.000000

Four teams with all perfect scores!!!

TAL TECH

Final rankings

Contest Winners

1. XDSecurity
    - Xidian University: Zhengguang Tang, Guangxin Guo, Benzheng Li, Hailong You, Jiangyi Shi
    - Giga Design Automation: Xiaojue Zhang
2. NTUsplace.
    - National Taiwan University: Jhih-Wei Hsu, Kuan-Cheng Chen, Yu-Hsiang Lo, Yan-Syuan Chen, Yao-Wen Chang
3. CUEDA
    - The Chinese University of Hong Kong: Fangzhou Wang, Qijing Wang, Bangqi Fu, Shui Jiang, Xiaopeng Zhang, Tsung-Yi Ho, Evangeline F.Y. Young
3. TalTech
    - Tallinn University of Technology: Tiago Perez, Mohammad Eslami, Felipe Almeida, Samuel Pagliarini

TAL TECH

Conclusions

- Pros:
  - We created **several techniques** for securing a layout
  - Our findings for sure will be published soon
  - All team members learned from the experience

- Cons:
  - The contest was easy to game, the score formula was too easy to abuse
  - Many rules changed throughout the competition, and many others were not even considered for the final scores
  - …

TAL TECH

THANK YOU!