

# An Introduction to The OpenROAD Project

---

**Andrew Kahng / Tom Spyrou  
UC San Diego / Precision Innovations  
March 30, 2021**

# OpenROAD: Team

## PIs(\*) + Other Faculty



ANDREW KAHNG\*  
UC San Diego



LAWRENCE SAUL  
UC San Diego



MATTEO  
COLTELLA\*  
Arm



PAUL PENZES\*  
Qualcomm



SHERIEF REDA\*  
Brown University



DENNIS  
SYLVESTER\*  
University of Michigan

## Team



TUTU AJAYI  
Ph.D. Student  
U. Michigan



TUCK-BOON  
CHAN  
Qualcomm R&D



VIDYA  
CHHABRIA  
Ph.D. Student  
U. Minnesota



SORIN DOBRE  
Qualcomm R&D



WENBO DUAN  
M.S. Student  
U. Michigan



COLIN  
HOLEHOUSE  
Arm R&D



MINSOO KIM  
Ph.D. Student  
UC San Diego



DAVID BLAAUW  
University of Michigan



RONALD  
DRESLINSKI  
University of Michigan



SACHIN  
SAPATNEKAR\*  
University of Minnesota



LAWRENCE CLARK\*  
Arizona State University



SEUNGWON  
KIM  
Postdoc  
UC San Diego



JIAJIA LI  
Qualcomm R&D



GERALDO  
PRAPISTA  
Ph.D. Student  
U. Minnesota



MATT  
RADEJCIC  
Qualcomm program  
mgmt



AUSTIN  
ROVINSKI  
Ph.D. Student  
U. Michigan



MEHDI  
SALIGANE  
Postdoc  
U. Michigan



VAISHNAV  
SRINIVAS  
Qualcomm R&D

## Consultants



JAMES  
CHERRY

Parallax Software /  
UC San Diego



DIMITRIS  
FOTAKIS

Nefelus Inc. / UC San  
Diego



MOHAMED  
SHALAN

The American  
University in Cairo



TOM SPYROU  
Precision  
Innovations / UC San  
Diego



MATT LIBERTY  
Liberty Software LLC  
/ UC San Diego



DON  
MACMILLEN  
MacMillen Software  
/ University of  
Washington



SANJIV  
MATHUR  
Precision  
Innovations



DAVID  
URQUHART  
Arm program mgmt



RAVI  
VARADARAJAN  
Ph.D. Student  
UC San Diego



VINAY  
VASHISHTHA  
Postdoc  
Arizona State U.



MINGYU WOO  
Ph.D. Student  
UC San Diego

Also: Vitor Bandeira, Ahmad ElRouby, Stephano Goncalves, Eder Monteiro, Isadora Oliveira and  
Osama Abdel Raheem

OpenROAD

DARPA

# Our Project: theopenroadproject.org

- **Open source**
- **No-human-in-loop RTL to GDSII**
  - Limited “knobs”, restricted field of use
  - Must replace intelligent humans
    - partitioning, floorplanning, ...
- **First target: digital IC flow “RTL to GDS”**
- **This requires:**
  - **State of the art EDA architecture**
    - **Unified *openroad* executable**
    - **Shared hierarchical EDA database**
    - **Integrated engines**
  - **Easy to use TCL interface for optional customized execution**
  - **Python for DB access and collection of Machine Learning features**

OpenROAD

Home People News and Events Publications Files Outreach

Subscribe

DEMOCRATIZING HARDWARE DESIGN

The OpenROAD project attacks the barriers of Cost, Expertise and Uncertainty (i.e., Risk) that block the feasibility of hardware design in advanced technologies.

READ MORE

Open Source Tools

User Guide

Community

SHOW ON GITHUB

GETTING STARTED

JOIN THE DISCUSSION

OpenROAD 23 Dec

MANY THANKS to @Google and Tim Ansell @mithro for another very generous gift to @UCSD -- it will be a great boost to OpenROAD research as we continue toward the goal of open-source RTL-to-GDS !!!

Load More...

About OpenROAD

Problem: Hardware design requires too much effort, cost and time.

Challenge: \$\$\$ costs and “expertise gap” block system designers’ access to advanced technology.

Objective: We want to enable no-humans, 24-hour design and catalyze open source EDA.

\*\*Important Flow and Platform Information:

- New: OpenROAD RTL-to-GDS v1.0

Foundations and Realization of Open, Accessible Design

Prof. Kahng and the OpenROAD team are aiming to develop open-source tools that achieve autonomous, 24-hour layout implementation.

More Details:

- PowerPoint and video presentation from 2019 ERI Summit
- PowerPoint presentation from 2018 ERI Summit

Latest News and Events

OpenROAD releases ASAP7 7nm Predictive PDK on GitHub !

December 15, 2020

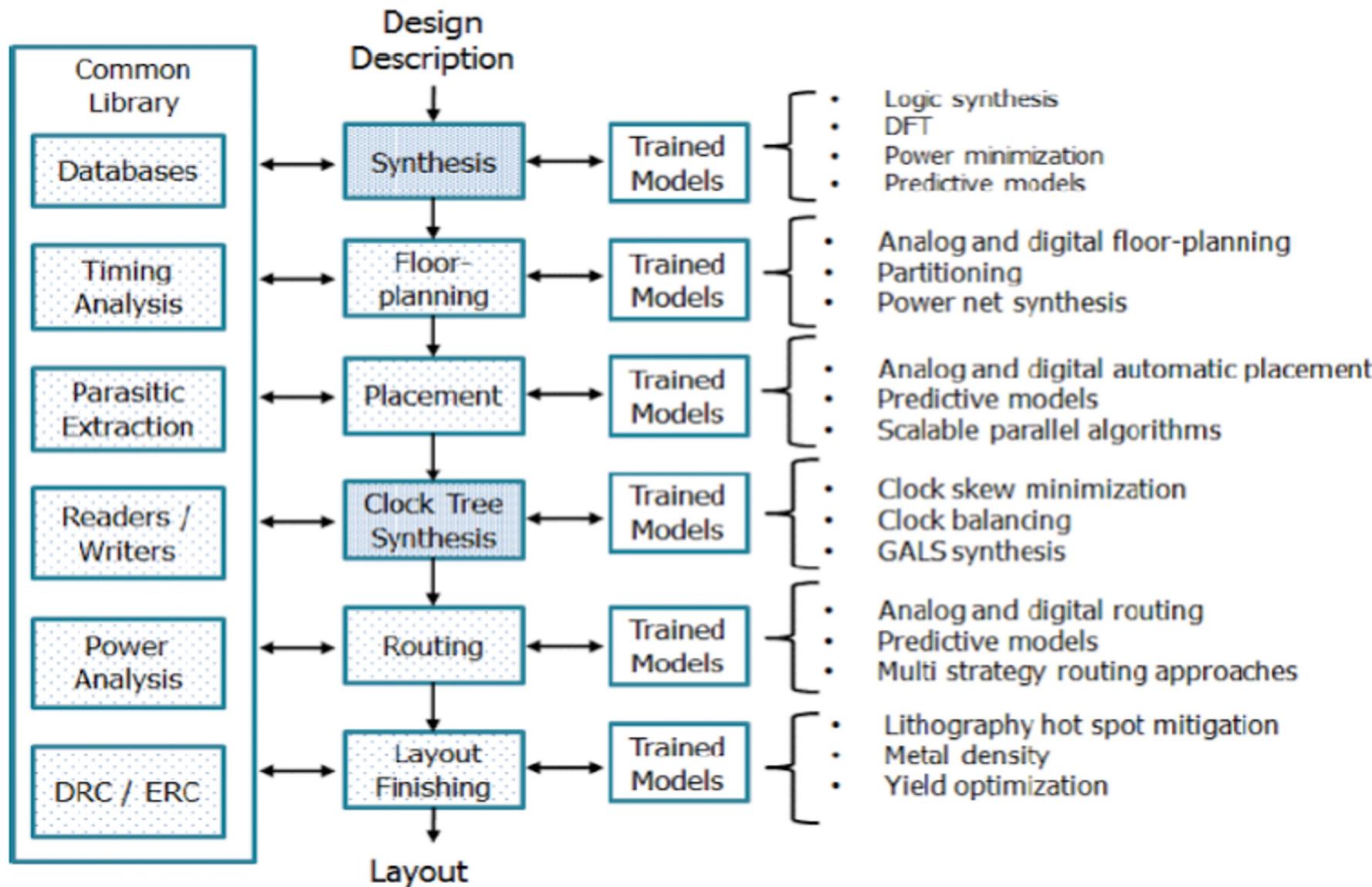
A post-route timing evaluation flow with OpenRCX is available !

December 14, 2020

Thanks — TritonRoute-WXL Open-Sourcing !

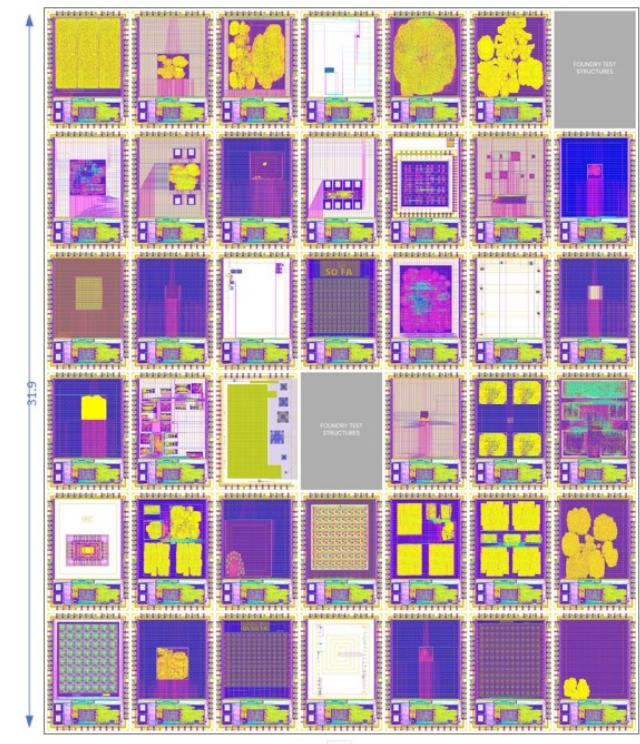
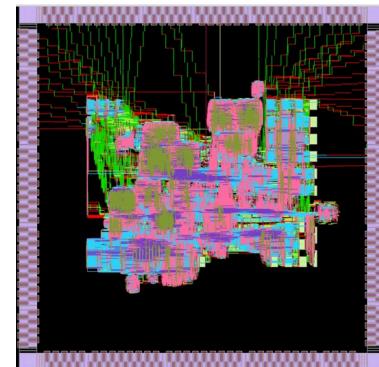
November 2, 2020

# OpenROAD = Digital SOC Layout Generation



# Recent accomplishments

- **Completion of modified, extended Phase 1**
  - Tapeout-clean GDS in GF12LP and TSMC65LP
  - Release of ASAP7 Open Source advanced-node PDK, libraries
  - Support of Open Source SkyWater SKY130 PDK
- **New functionality**
  - OpenRCX parasitic extraction
  - Phase 2 focus on PPA, end users
- **Improved architecture and integration**
  - Truly integrated Open Source EDA tool
- **PPA and machine learning**
  - CI, logging, metrics, insight from large runsets
- **More usage, GitHub issues, traction**
  - 40+ tapeouts in 130nm Google-SkyWater SKY130 shuttle
- **We are looking for early users for real designs**



Partially populated mask shot

efabless

# “Gallery” of TSMC65LP, GF12LP Proofpoints

- **Signoff-level timing/DRC-clean results**

- 65nm (TSMC65LP) and 12nm (GF12LP)
- Blocks: ibex, jpeg, coyote, swerv\_wrapper
- SOC: BP-1 (GF12LP)

- **Signoff criteria**

- **Signoff tools used for validation**

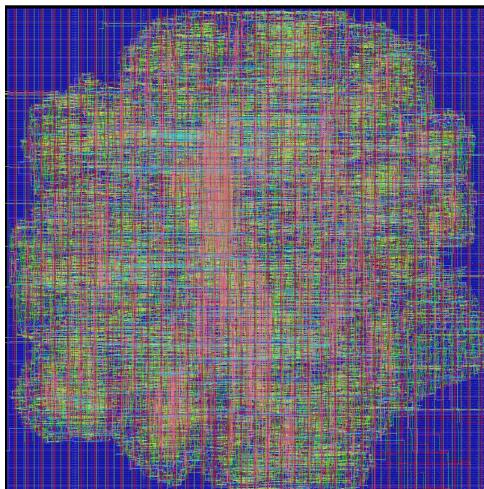
- StarRC/Quantus, PrimeTime, Calibre

- **GF12LP**

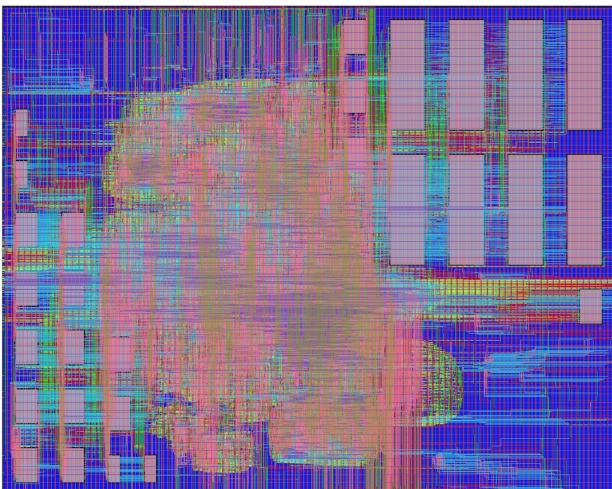
- Worst corner: SSPG, 0.72V, 125C, SigCmax
- Best corner: FFPG, 0.88V, -40C, SigCmin
- PDK / Misc.: Macro sc9mcpp84\_12lp;  
BEOL stack 13M\_3Mx\_2Cx\_4Kx\_2Hx\_2Gx\_LB

- **TSMC65LP**

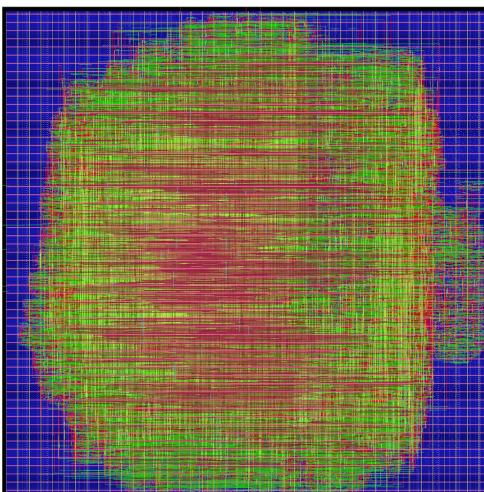
- Worst corner: SS, 1.08V, 125, RCmax, Ccmax
- Best corner: FF, 1.32V, -40C, RCmin, Ccmin
- PDK / Misc.: Macro sc12\_cln65lp;  
BEOL stack 1p9m\_6x2z



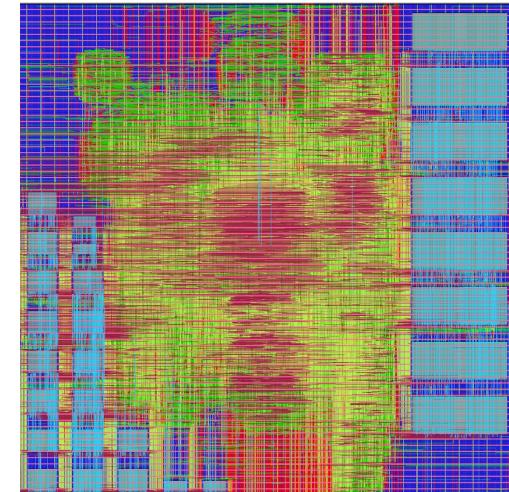
GF12 jpeg



GF12 swerv\_wrapper



TSMC65LP jpeg



TSMC65LP swerv\_wrapper

**DRC, LVS, Antenna, Hold, ERCs all clean !**

# Numerous PPA-Directed Projects underway

- Phase 2a focus: PPA improvement for RTL-to-GDS

- **Synthesis (Yosys+ABC)**

- Improve timing results out of synthesis
  - Enable buffering, up/down sizing

- **Placement (RePIAce)**

- Re-enable timing-driven global placement

- **CTS (TritonCTS)**

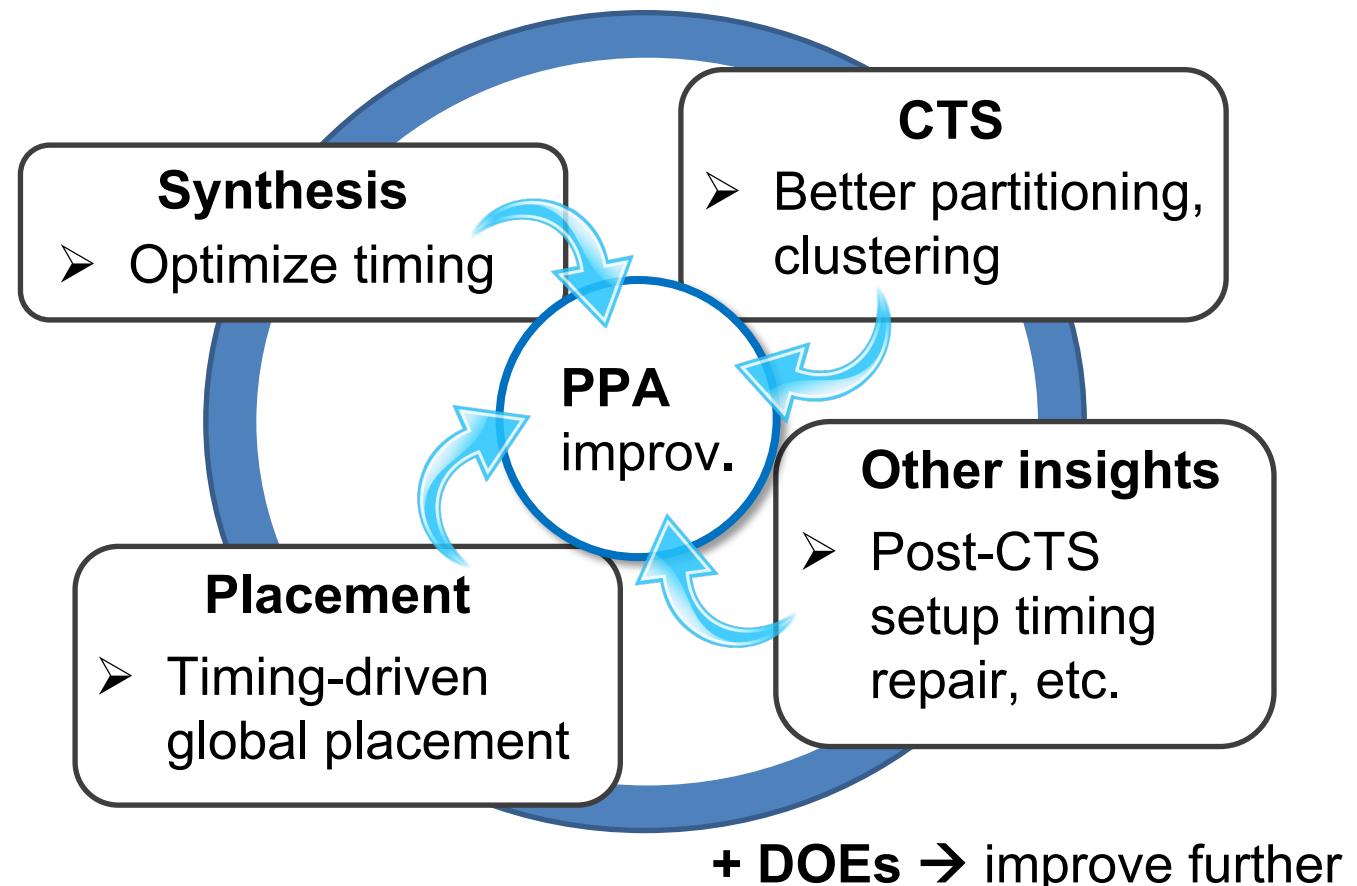
- Re-enable, update partitioning, clustering

- **Massive SynthDOE, PostSynthDOE**

- Dial in and ratchet up PPA from big data

- **And more...**

- **Resizer** post-CTS setup timing repair
- Tighten up P&R – e.g., less cell padding



# Improvements since Oct 2020 tape-in of Black Parrot CPU

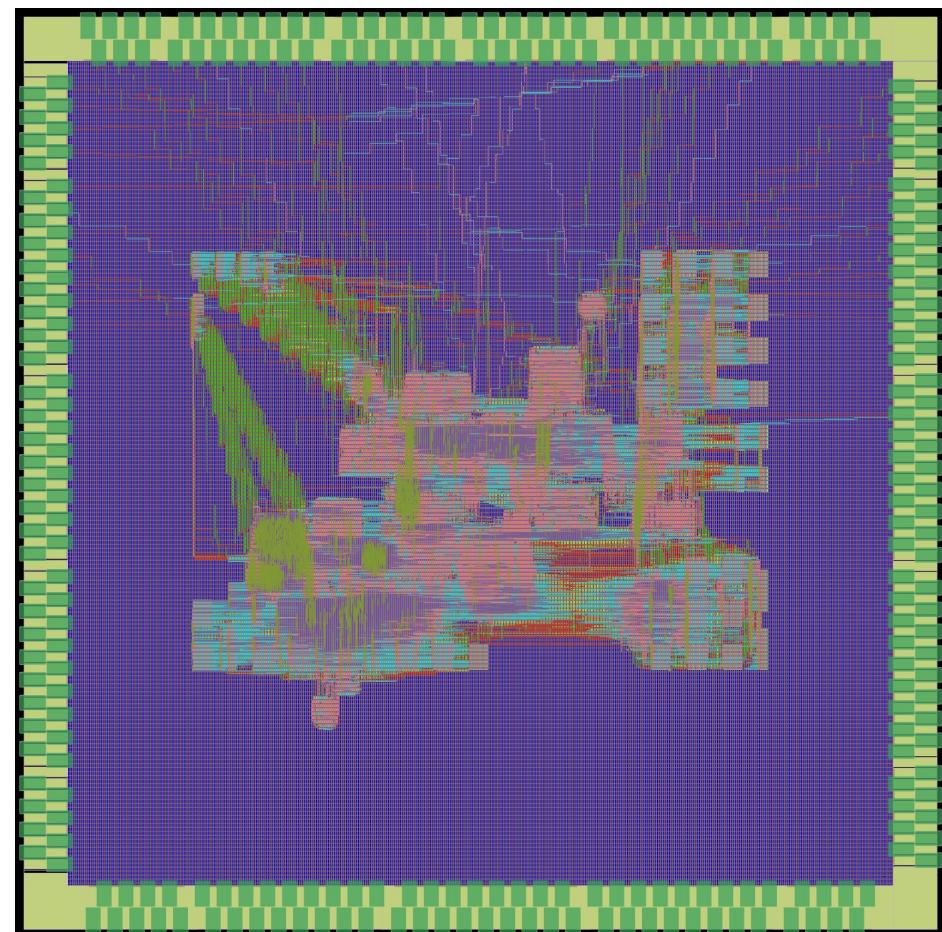
- **Current vs. last October's 'golden' BP-1**

- OpenROAD changes for PPA improvement
  - Synthesis, placement, CTS, setup timing repair, ...
- Push target clock period
  - 8 ns → 6 ns
- Increase placement density
  - Reduced global padding (bloating) of cell instances

- **Comparison table (from OpenSTA report)**

	target CP (ps)	WNS (ps)	TNS (ps)	fmax (MHz)	max skew (ps)	total WL (um)	#Insts	total power (W)
'golden' Oct20	8000	-894	-438729	112	813	9908654	795111	0.376
Current	6000	-580	-248060	152	583	8670446	730001	0.367

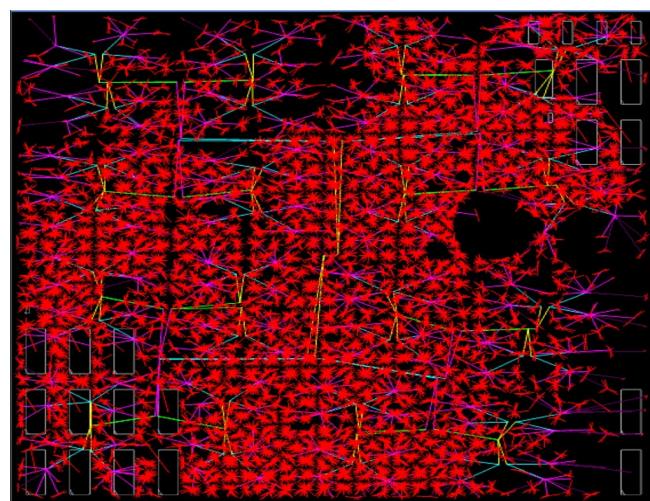
**Improvement: 43% 36% 28% 12% 8% 2%**  
(relative to the given CP)



post-route layout: GF12 bp-1

# Example PPA Vector: Clock Tree Synthesis

- Buffer characterization: accurate, on-the-fly
- **Improved partition assignment:** ~50% latency reduction in GF12LP regressions
- **Improved sink clustering:** less tree depth
- **Logger support and GUI views**



GUI visualizations for analysis and debug (NG45, BP-1 “block”)

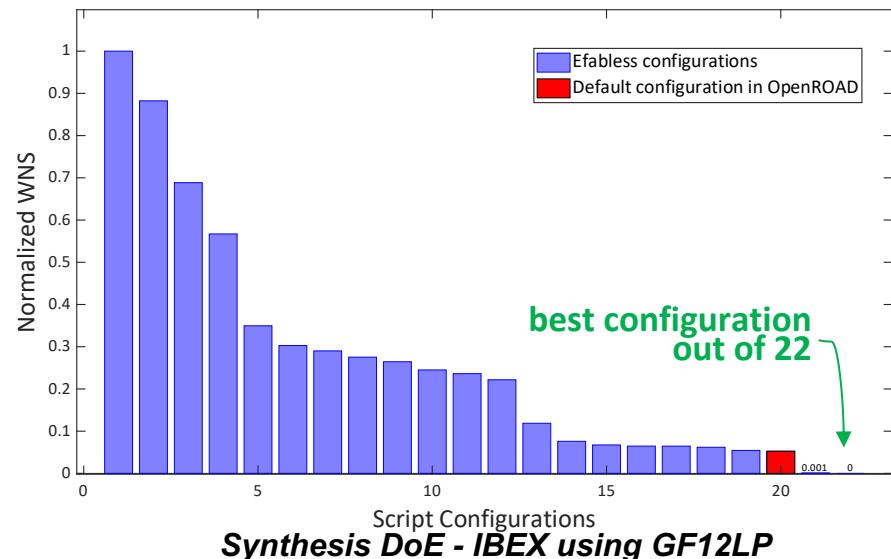
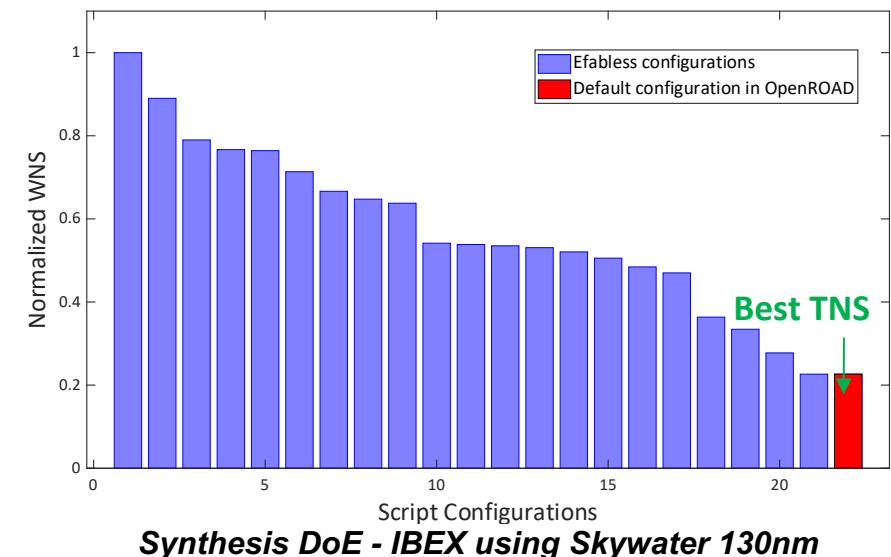
Baseline							
	Testcase	#hold_fix_bufs	total Area	WNS (ns)	TNS (ns)	skew (ns)	latency (ns)
1	aes	0	12675.9	-0.316	-49.281	0.055	0.183
2	coyote	164682	184158.6	-1.609	-174.517	1.543	2.103
3	bp_single	238842	332199.9	-0.808	-193.051	2.017	3.378
4	swerv_wrapper	7529	83344.9	-0.277	-20.815	0.354	0.805
Total		411053	612379.3	-3.01	-437.664	<b>3.969</b>	<b>6.469</b>

Partitioning Improvements							
	Testcase	#hold_fix_bufs	total Area	WNS (ns)	TNS (ns)	skew (ns)	latency (ns)
1	aes	0	12676.5	-0.313	-49.135	0.072	0.197
2	coyote	124929	171957.9	-0.944	-26.821	0.282	0.79
3	bp_single	184140	326041.2	-0.828	-99.773	0.402	1.734
4	swerv_wrapper	7334	83238.5	0	0	0.157	0.589
Total		316403	593914.1	-2.085	-175.728	<b>0.913</b>	<b>3.31</b>

Sink Clustering							
	Testcase	#hold_fix_bufs	total Area	WNS (ns)	TNS (ns)	skew (ns)	latency (ns)
1	aes	0	12642.9	-0.211	-32.906	0.059	0.182
2	coyote	124718	168671.8	-1.019	-53.768	0.179	0.789
3	bp_single	182852	321056.1	-0.824	-101.129	0.267	1.62
4	swerv_wrapper	7728	83110.2	-0.561	-138.837	0.105	0.609
Total		315298	585481	-2.615	-326.64	<b>0.61</b>	<b>3.2</b>

# Synthesis Automatic PPA Exploration Runs

- 22 synthesis optimization recipes
- 7 main configurations
  - 4 dedicated to timing
  - 3 dedicated to area reduction
- Buffering/sizing options include
  - max fanout
  - max transition constraints
  - upsizing/downsizing
- Default OpenROAD recipe shown in red
  - Contribution from new logic synthesis team
- Example learning: need design-dependent recipe and/or “cocktail” of recipes

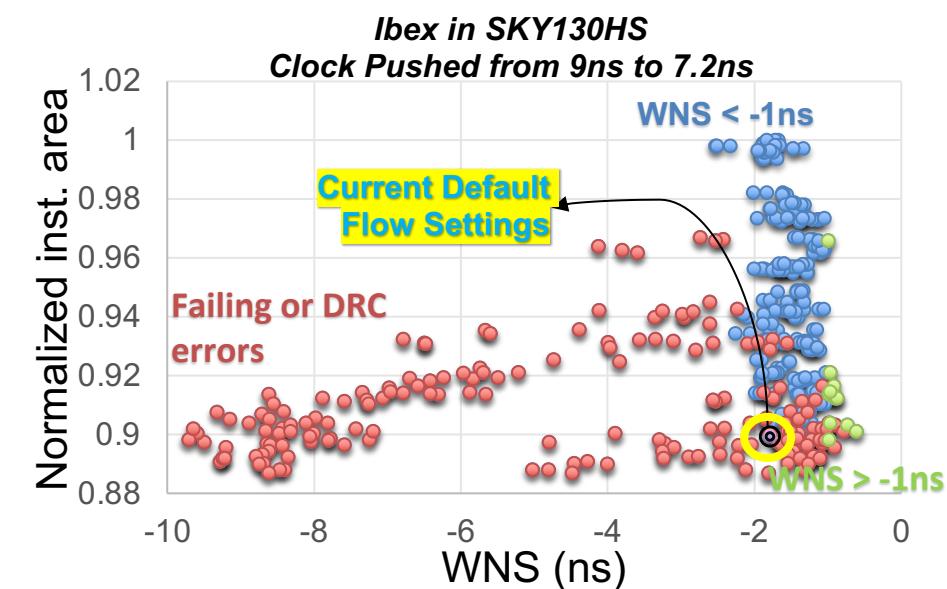
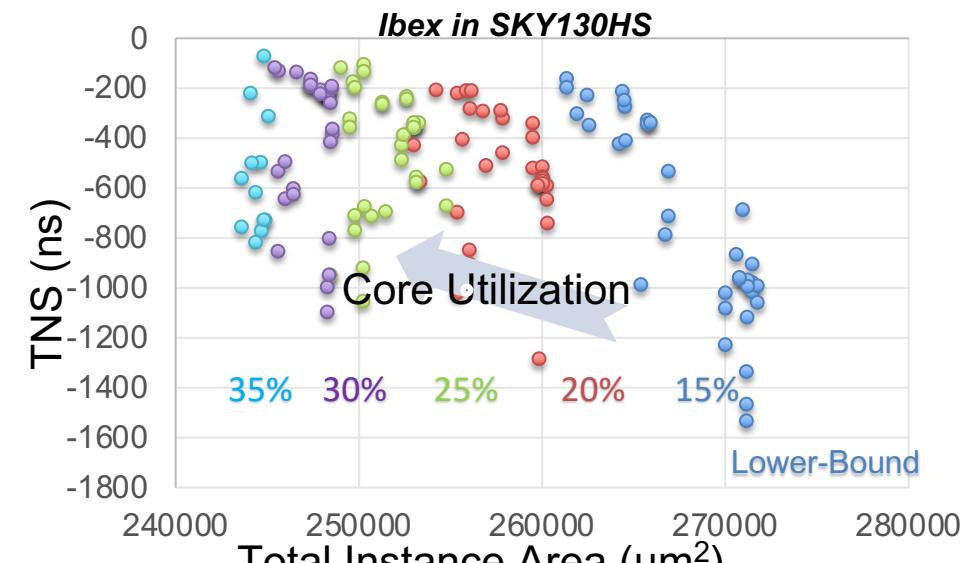


# Post-Synthesis Automatic PPA Exploration

- Analyze and improve QOR trajectories
- Uncover novel flow recipes, unsuspected issues
- Enable ML within OpenROAD
- Testcase “ratchet” example: SKY130HS ibex
  - QOR gains found for frequency, wirelength
  - Insight: routing fails with excessive post-CTS TNS

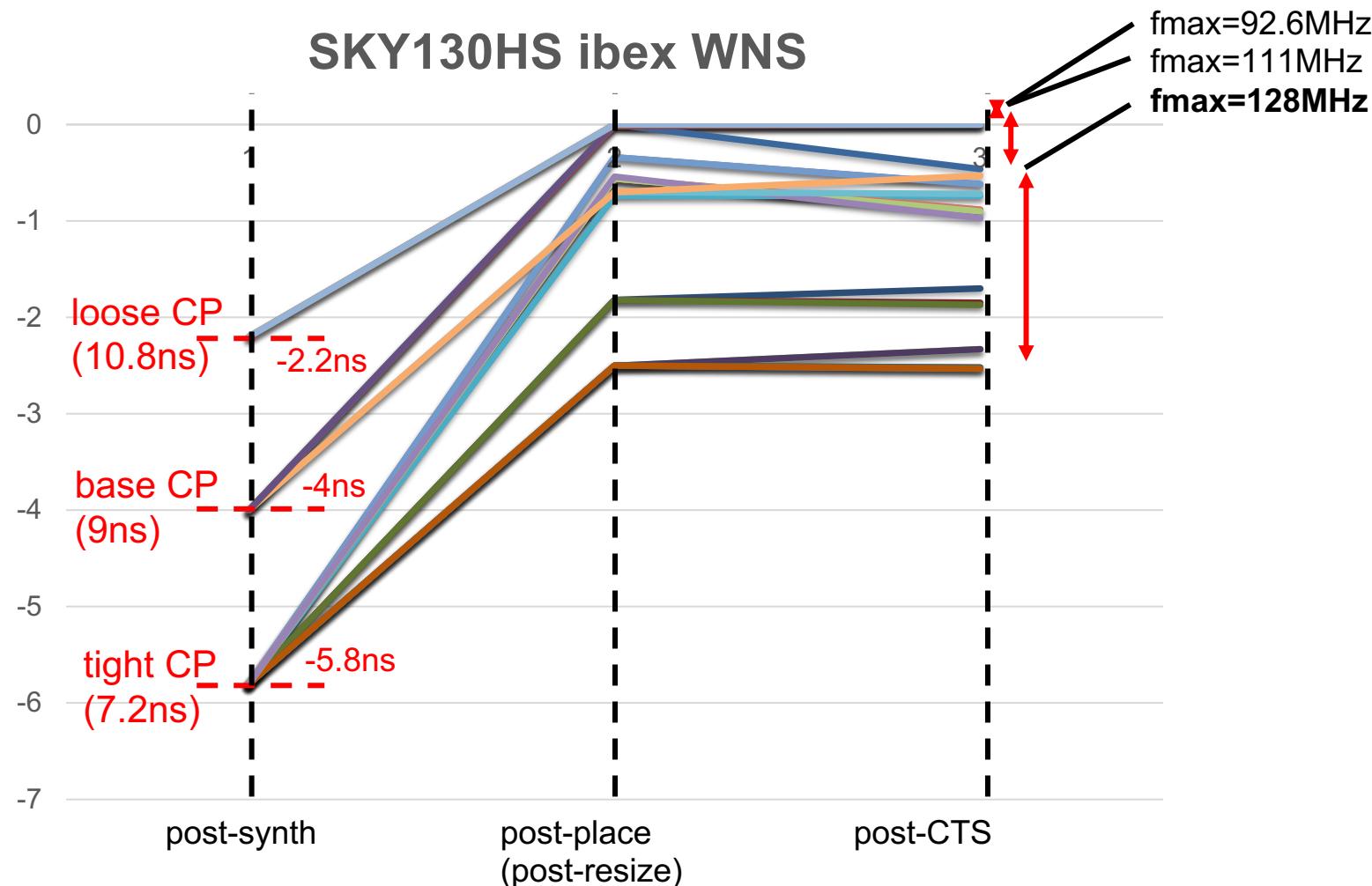
	Pre-DOE	Post-DOE
CLK Period (ns)	9	7.9
Aspect Ratio	0.7	1
Starting Utilization	35	35
GP Padding	4	1
DP Padding	2	1
Post-PlaceOpt Density	0.88	0.62
Inst. Area ( $\mu\text{m}^2$ )	244370	244841
DR WL ( $\mu\text{m}$ )	1102453	1050255

Can ratchet up our baseline ...



# Synthesis P&R Flow convergence study

- **Shown:** Trajectories of timing quality through SP&R in OpenROAD when **placer** parameters swept
- Pushing tool harder can achieve better QOR but with wider range of outcomes
  - Similar to commercial tools
- **Ongoing:** Learning how to manage populations of these trajectories within given “footprint” of (threads x hours) compute resource



# AI METRICS Standardization → Tool Learning

- Standardized metrics collection in OpenROAD flow
  - Design metrics (#buffers, total WL)
  - Run metrics (cpu time, peak memory usage)
  - Become features for training Machines
- Tools use unified logger with consistent namespaces, INFO/WARN/... nomenclature
- Essential for continuous PPA improvement, learning-enabled automation
- **Many purposes**
  - Dashboards, summary of nightly regression runs
  - QoR evaluation of incremental functional changes
  - Validations before PR merge to master
  - Distributed experiment data collection and analysis

## OpenROAD Metrics Naming

- **Design Stage**
  - Synthesis, Floorplan, Global Placement, Detailed Placement, CTS, Global Route, Detailed Route
- **Metric Category**
  - Area, Congestion, Timing, Power, CPU, Memory
- **Metric**
  - TNS, WNS, instances, switching\_power, cpu\_time, ...
- **Metric Modifiers**
  - worst, total, reg\_to\_reg, ...

## Examples

```
floorplan::area::instances::stdcell::count
globalplace::timing::wns::worst::reg_to_reg
cts::timing::latency::max
cts::timing::skew::max
```

# METRICS

run_flow_design	aes	coyote	swerv_wrapper	gcd	bp		
run_flow_platform	tsmc65lp	tsmc65lp	tsmc65lp	tsmc65lp	tsmc65lp		
run_flow_hostname	pdn.ucsd.edu	pdn.ucsd.edu	pdn.ucsd.edu	pdn.ucsd.edu	pdn.ucsd.edu		
synth_area_stdcell_count	20303	232048	97520	306	123143		
synth_area_stdcell_area	90611.5	1.70011e+06	1.15439e+06	1800	1.28939e+06		
constraints_clocks_count	1	1	2	1	1		

Metrics collected at different flow stages, across designs / platforms

floorplan_area_IO_count	388	784	1416	54	1198	54	N/A	5
floorplan_area_inst_util	14%	17%	28%	27%	29%	8%	5%	1
globalplace_area_density_target	0.40	0.45	0.40	0.50	0.40	0.30	0.20	0
globalplace_area_wirelength_est	1107513	13940754	9318033	10275	11166213	8601	10252337	5
placeopt_area_buffer_input	258	281	532	35	599	35	N/A	3
placeopt_area_buffer_output	129	502	882	18	598	18	N/A	1
placeopt_area_resize_inst	10777	92022	46268	161	58629	72	125857	1
placeopt_timing_tns_total	-0	-8911.92	-8102.38	0	-18495.6	0	0	0
placeopt_timing_wns_worst	0	-9.91	-2.66	0	-14.74	0	-837.35	0
placeopt_area_inst_area	121321 u^2	2258409 u^2	1369196 u^2	1965 u^2	1574557 u^2	560 u^2	483386 u^2	4
placeopt_area_inst_util	20%	24%	35%	31%	37%	9%	7%	6
detailedplace_area_displacement_total	27407	551656	168713	711.6	283850	511.8	204008	3
detailedplace_area_displacement_average	1.2	1.6	1.5	1.6	2	1	0.4	0
detailedplace_area_displacement_max	19.3	52.8	81.2	16.6	54.2	12.8	34	1

## Logger snippets: GlobalPlace, GlobalRoute

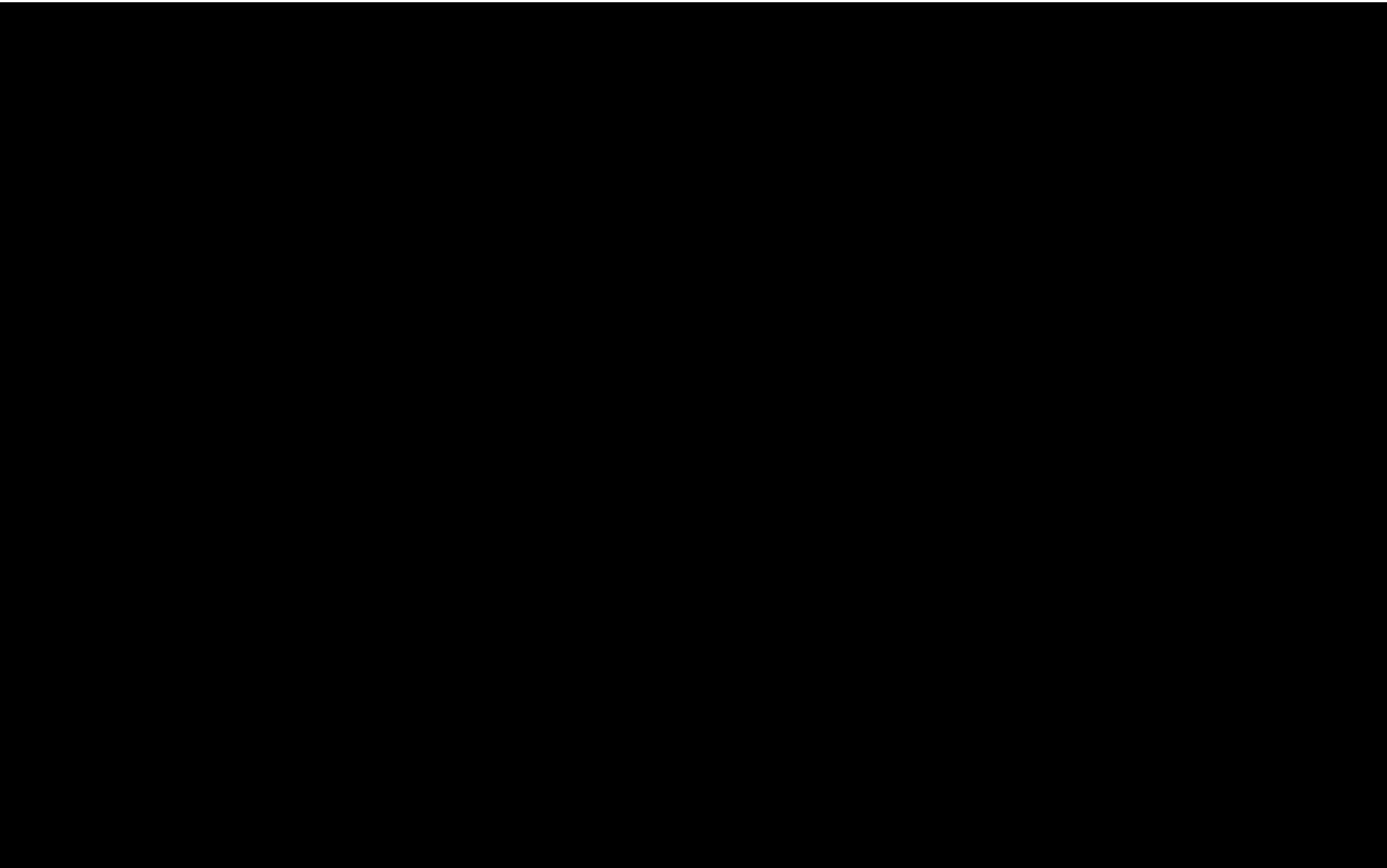
```
[INFO GPL-0003] SiteSize: 168 1152
[INFO GPL-0004] CoreAreaLxLy: 353976 353664
[INFO GPL-0005] CoreAreaUxUy: 5645976 5645952
[INFO GPL-0006] NumInstances: 460524
[INFO GPL-0007] NumPlaceInstances: 310188
[INFO GPL-0008] NumFixedInstances: 123576
[INFO GPL-0009] NumDummyInstances: 26760
[INFO GPL-0010] NumNets: 313960
```

```
[INFO GRT-0152] Layer 1 usage percentage: 0.00%.
[INFO GRT-0152] Layer 2 usage percentage: 10.41%.
[INFO GRT-0152] Layer 3 usage percentage: 10.24%.
[INFO GRT-0152] Layer 4 usage percentage: 8.61%.
[INFO GRT-0152] Layer 5 usage percentage: 7.88%.
[INFO GRT-0152] Layer 6 usage percentage: 4.99%.
[INFO GRT-0152] Layer 7 usage percentage: 3.52%.
```

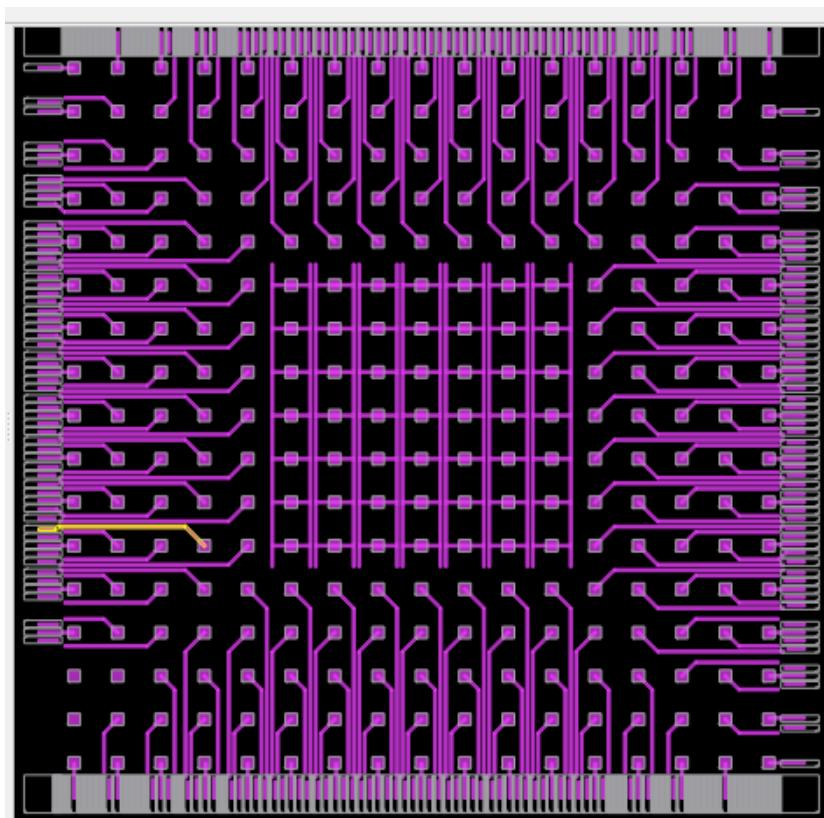
```
[INFO GRT-0156] Total Usage : 10010920.
[INFO GRT-0157] Total Capacity: 162323689.
[INFO GRT-0158] Max H Overflow: 0.
[INFO GRT-0159] Max V Overflow: 0.
```

# OpenROAD has a GUI for developers and users

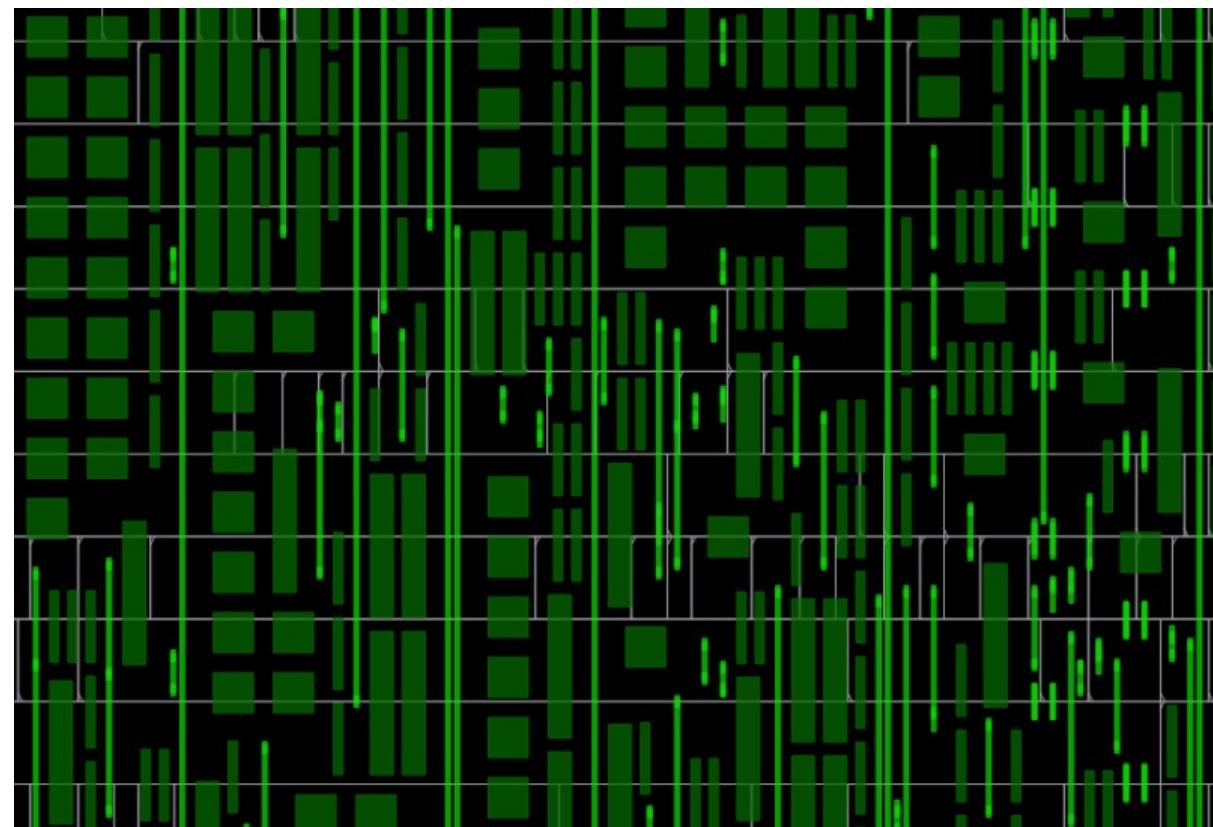
---



# GUI Visualizations: RDL, BEOL Fill

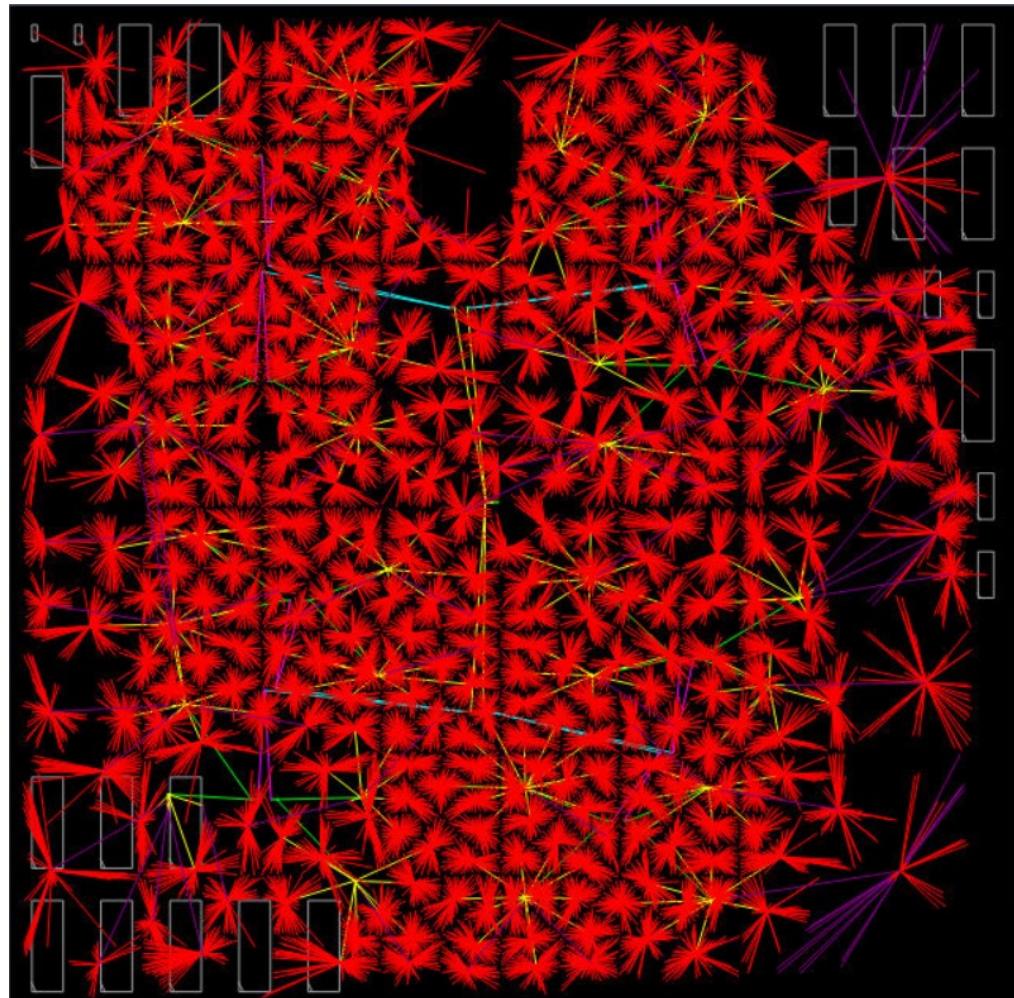


**RDL Support  
(45-degree geometries)**

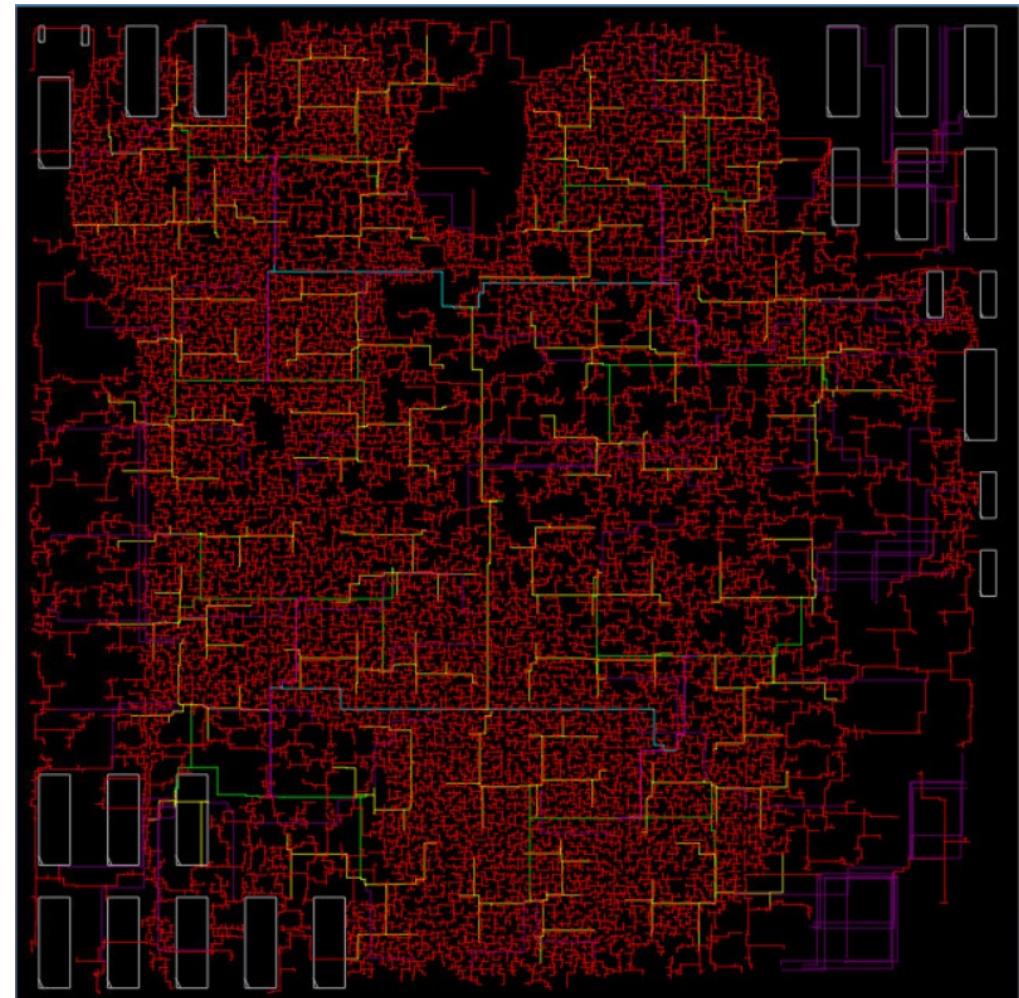


**OpenROAD-generated Metal Fill**

# GUI Visualizations: Clock Tree

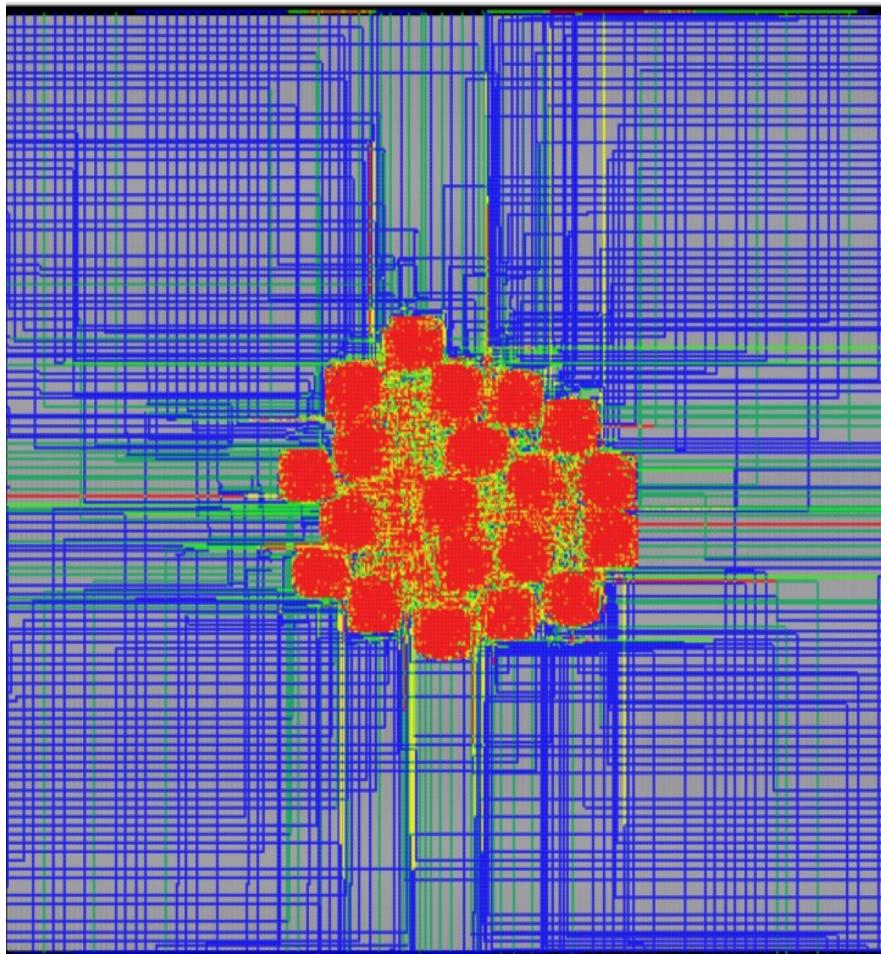


Placed Clock Tree (GF12LP BP-1)

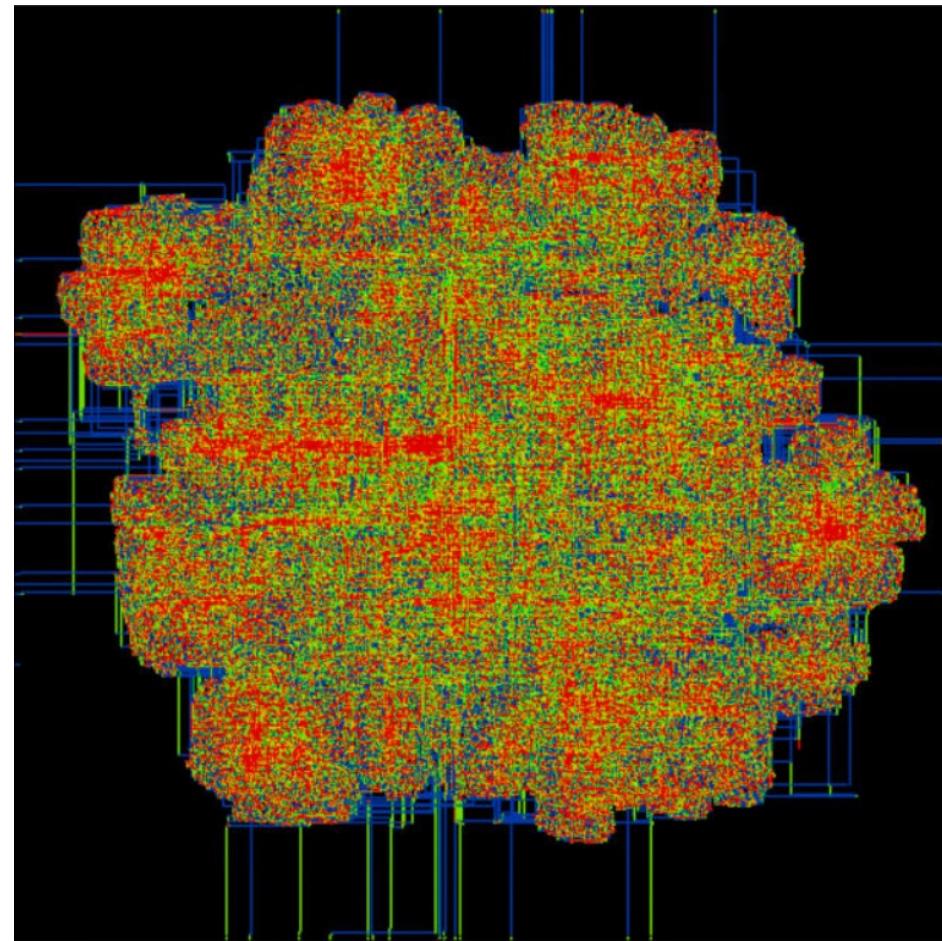


Routed Clock Tree (GF12LP BP-1)

# GUI Visualizations: Congestion Display

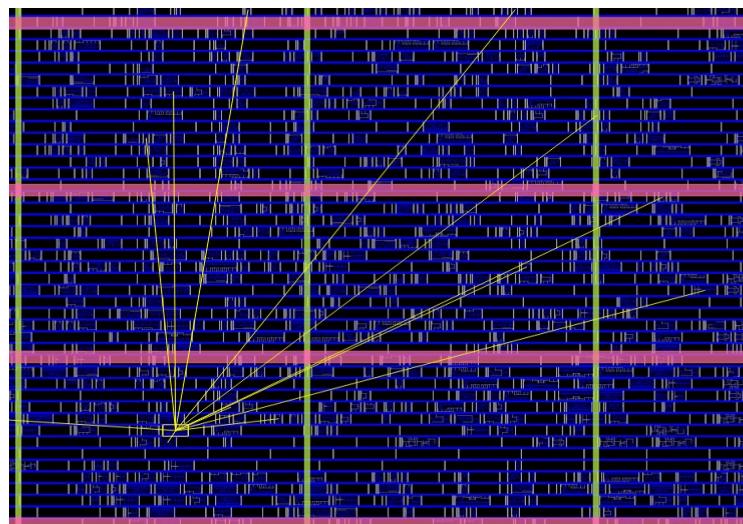


GF12LP AES

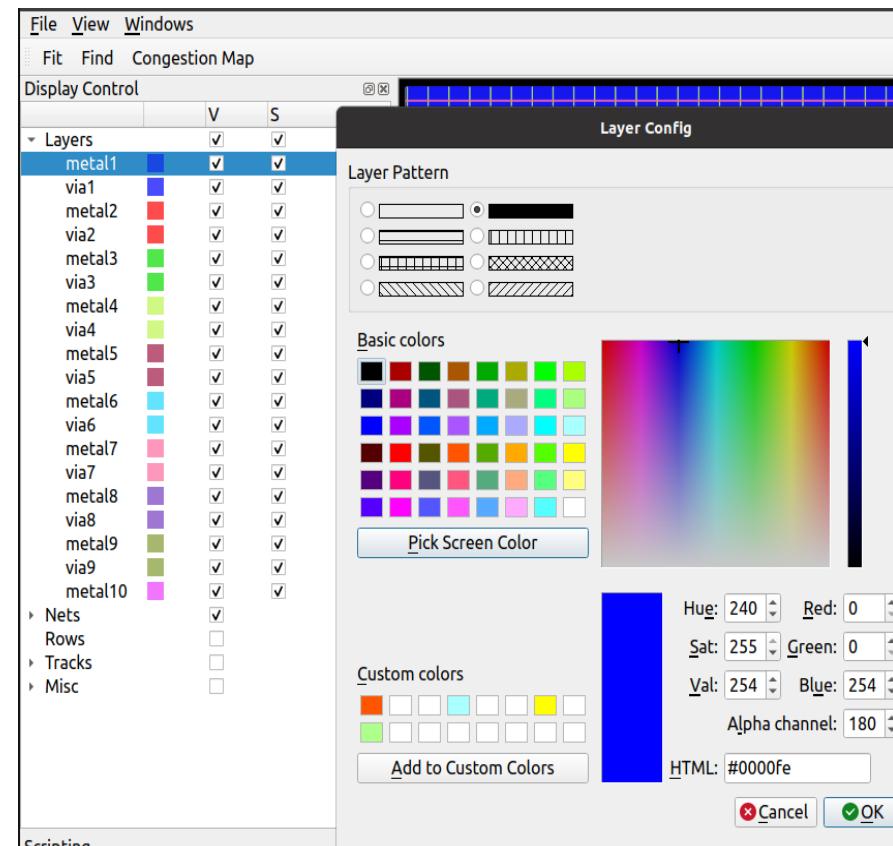


GF12LP JPEG

# GUI Visualizations: (and more)

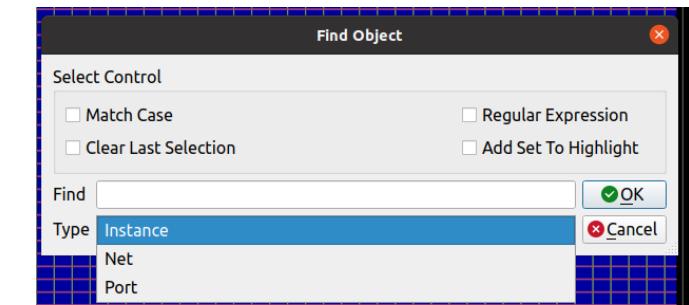


Flyline connectivity

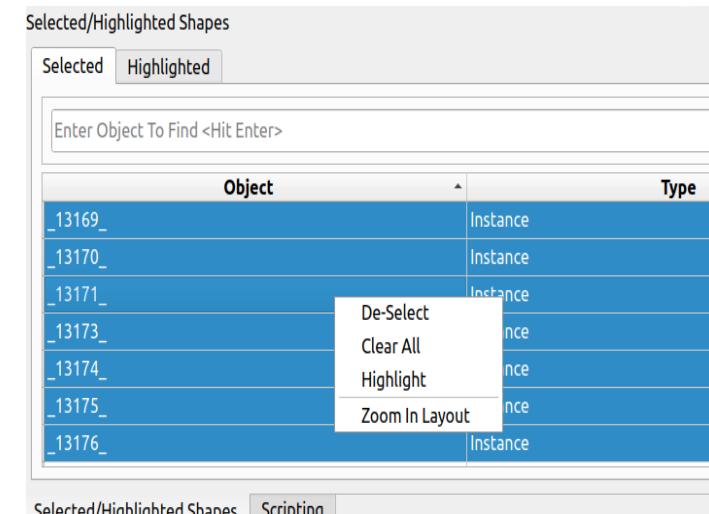


Layer pattern selection

Robust and easy to extend GUI architecture,  
timing GUI in progress



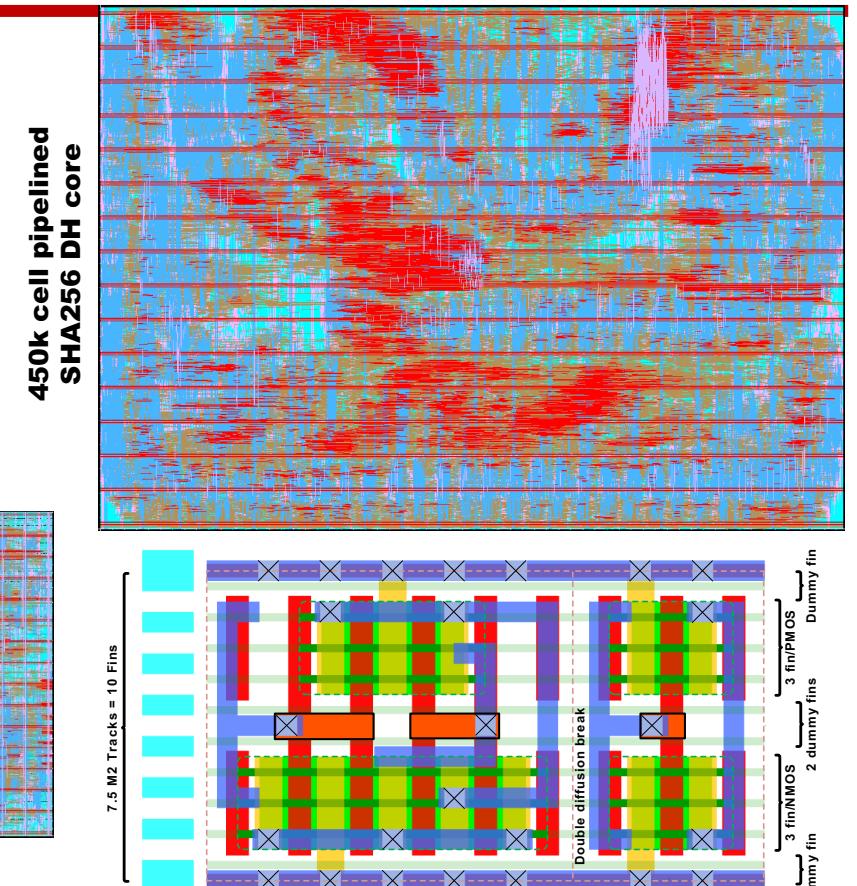
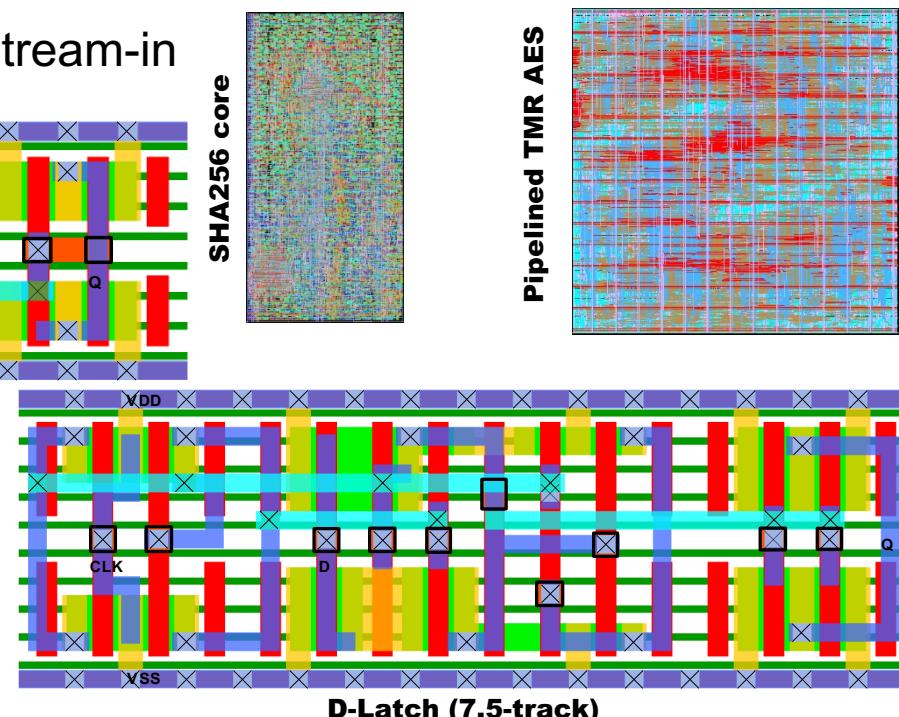
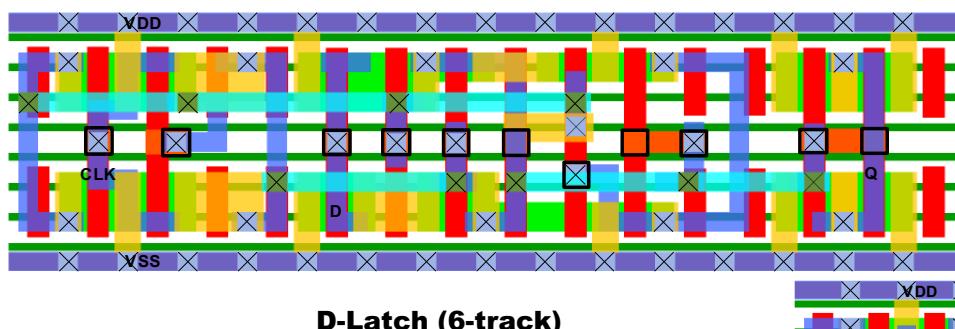
Object select/highlight dialog



Selected objects  
properties

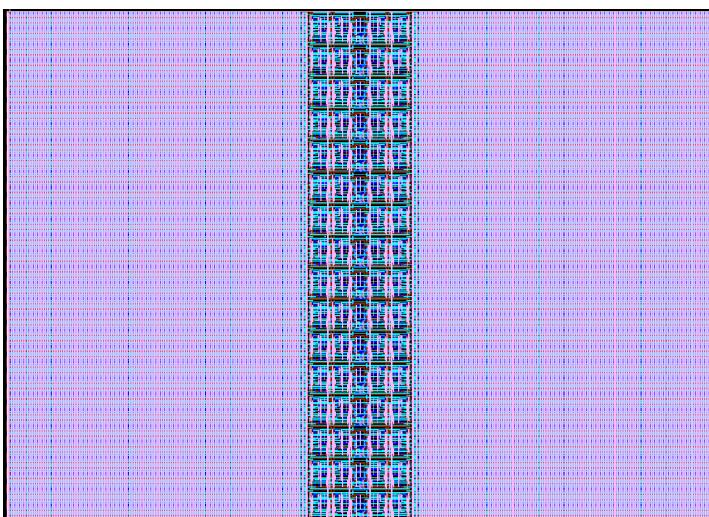
# ASAP7 Release + Milestones

- ASAP7 7nm FinFET predictive PDK and 7.5T libraries released
  - <https://github.com/The-OpenROAD-Project/asap7>
- 7.5-track library has 212 cells × four Vt's
  - Includes cell CDL, GDS (RVT only), LEF, LIB (NLDM and CCS), Verilog, and parasitic extracted CDL views
- 6-track library is nearing completion
  - Integrated clock-gaters require revision
  - Clean through synthesis, APR, stream-in

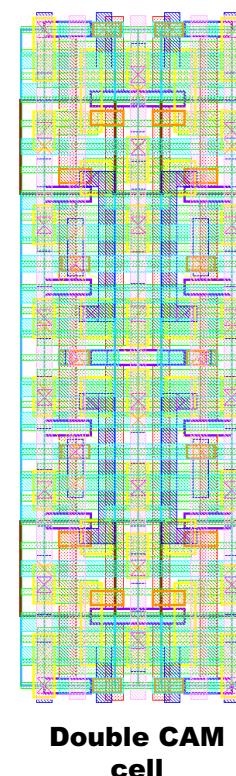


# ASAP7: Memories

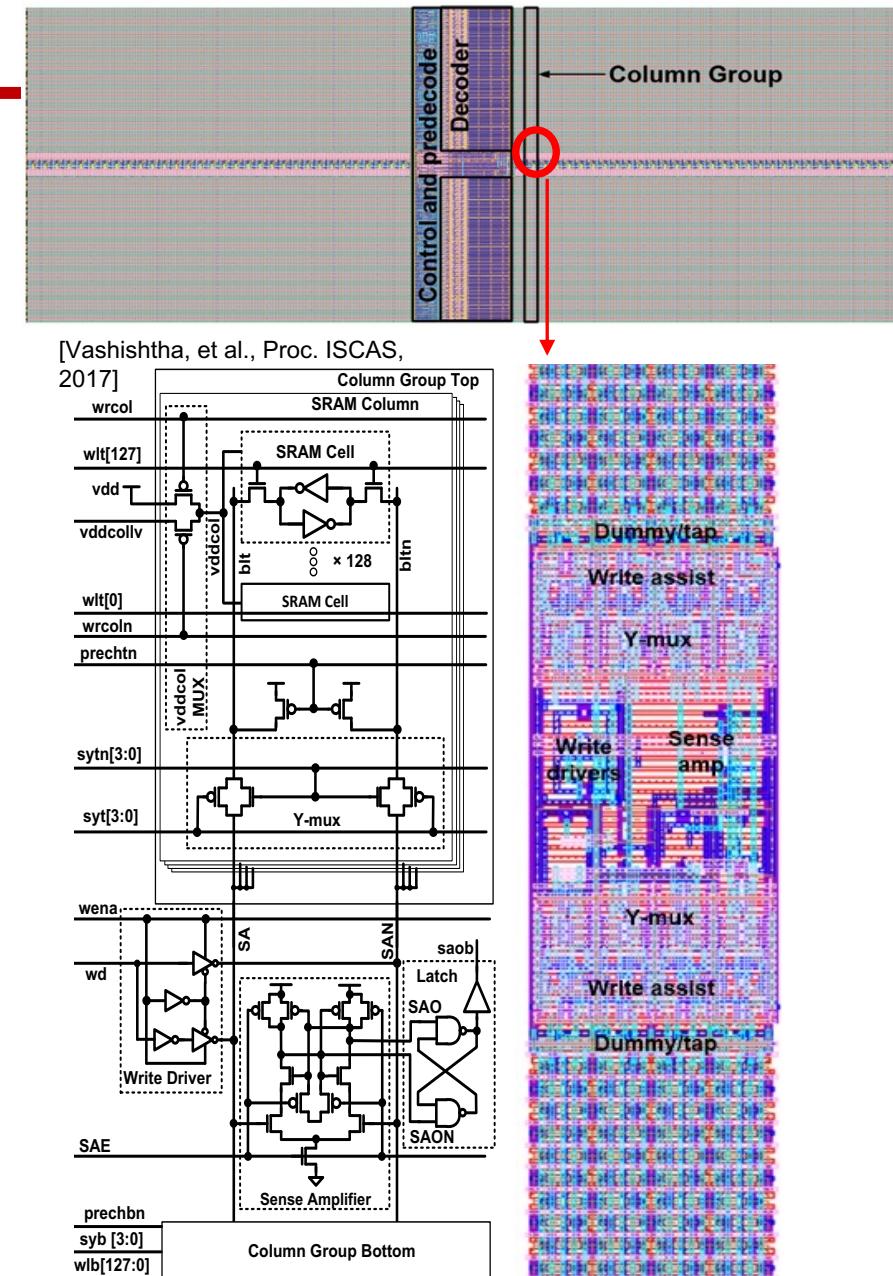
- ASAP7 SRAM: Base circuits and layouts finished
- Ongoing changes
  - Write assist improvements
  - Compatibility with characterization tools (designs are time-borrowing latch based)
- Timing characterization flow with Cadence Liberate MX in progress
- Register files, ROM, CAM, TCAM also in progress



2kB 8-T cell register file (16 bits)

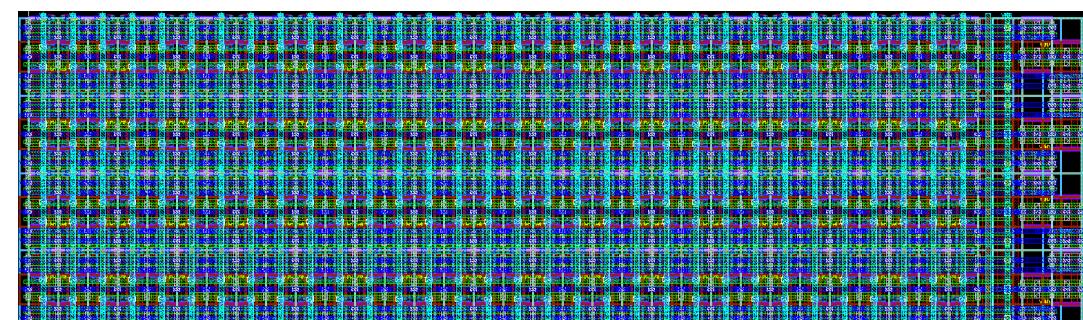
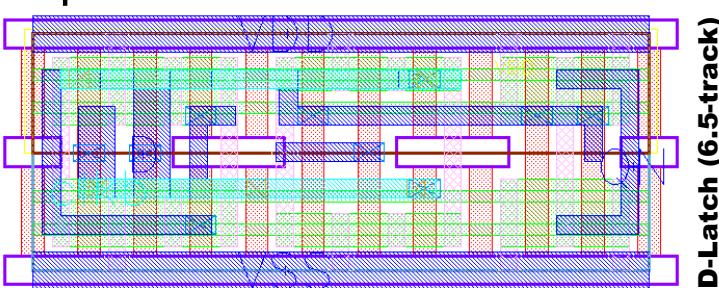
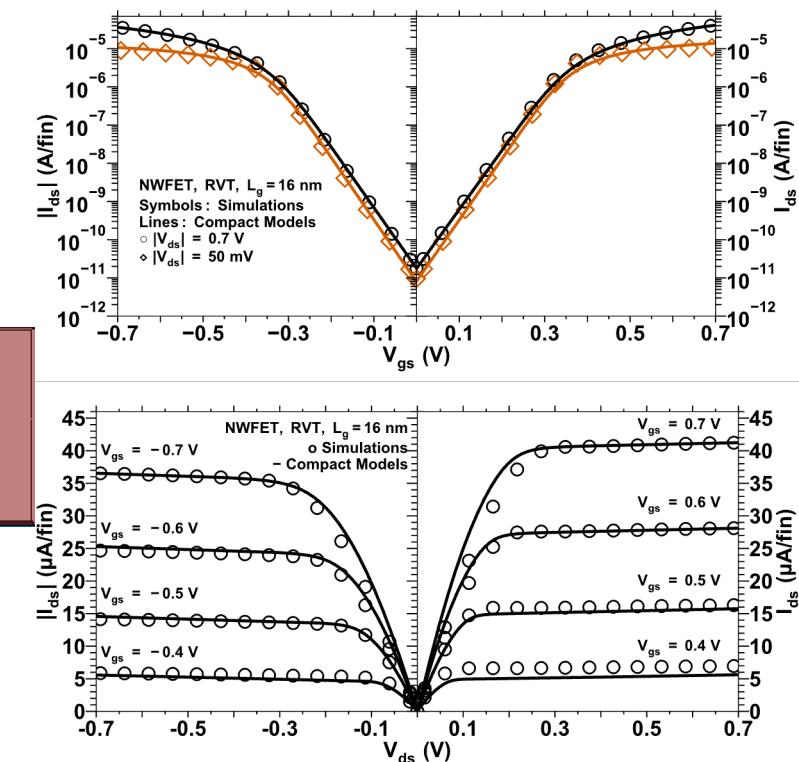
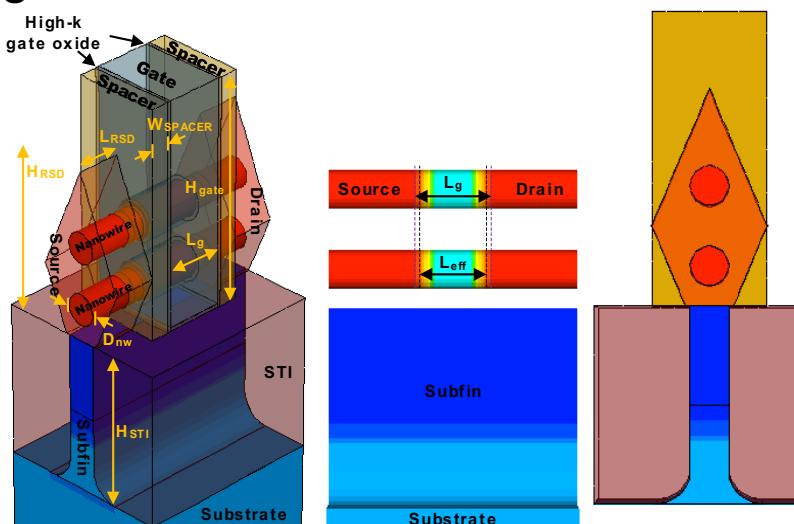


Double CAM cell



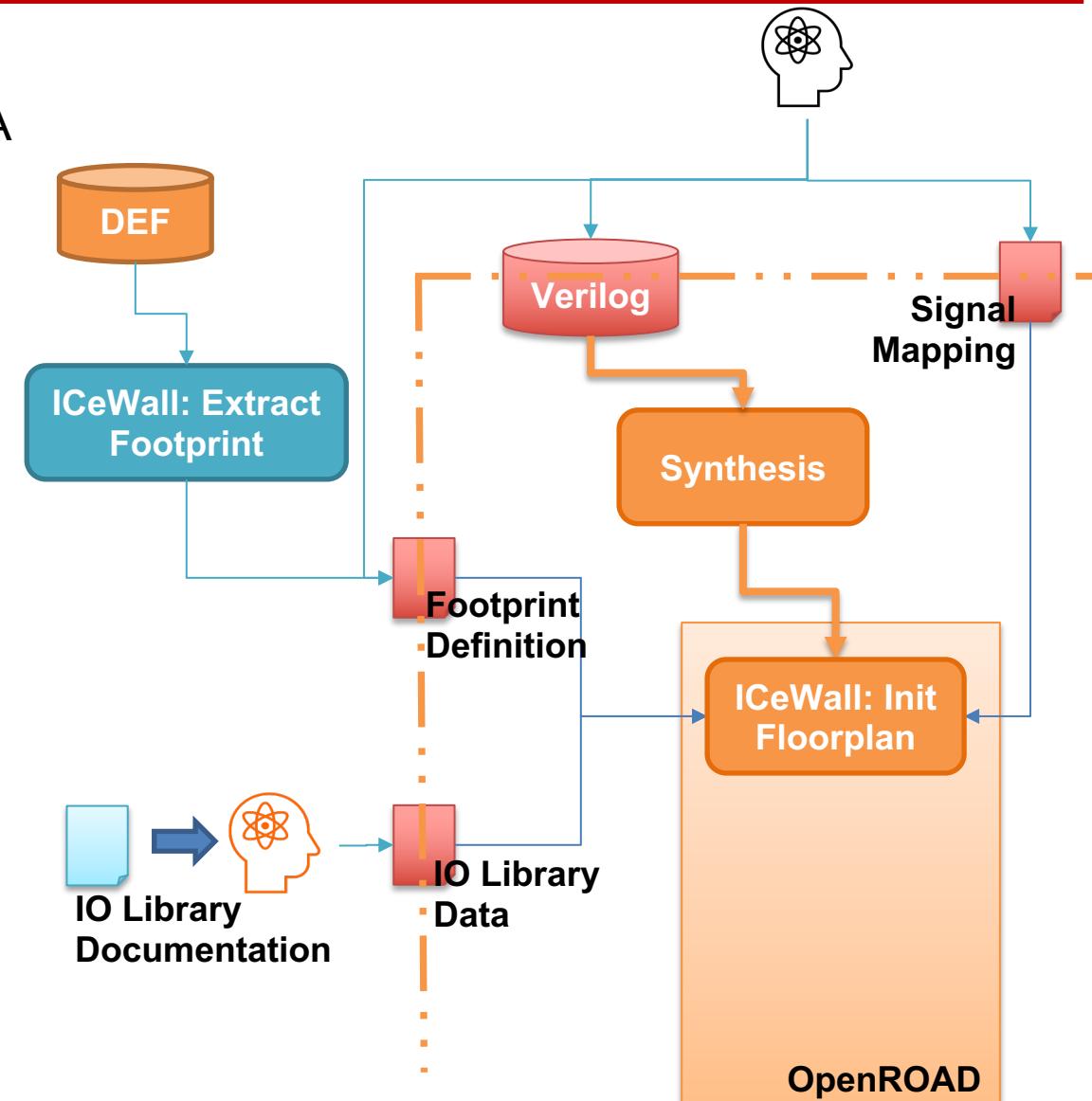
# ASAP: Future = ASAP5

- ASAP5: horizontal nanowire transistors
  - 3-D TCAD based compact models
- Greater density based on recent foundry PDK enhancements
  - Single diffusion breaks
  - Contact over-active gate
  - Denser cross-overs
- 6.5 track cell library 85% complete
  - APR checkout not yet begun
- Calibre decks
  - Parasitic RC extraction complete
  - LVS complete
  - DRCs 80% - pending APR checks

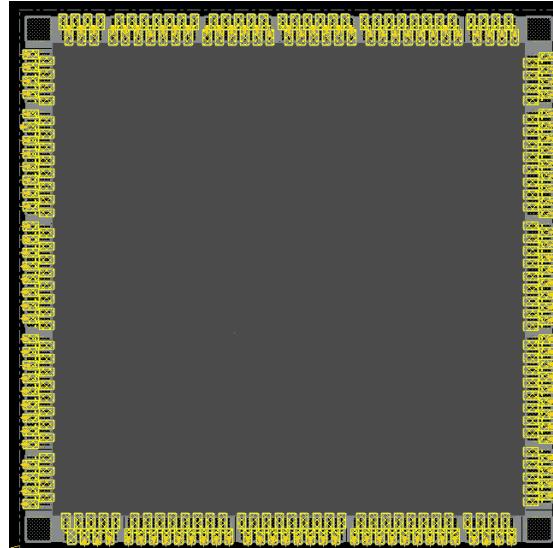


# SOC Integration and Planning: ICeWall Padding Generation

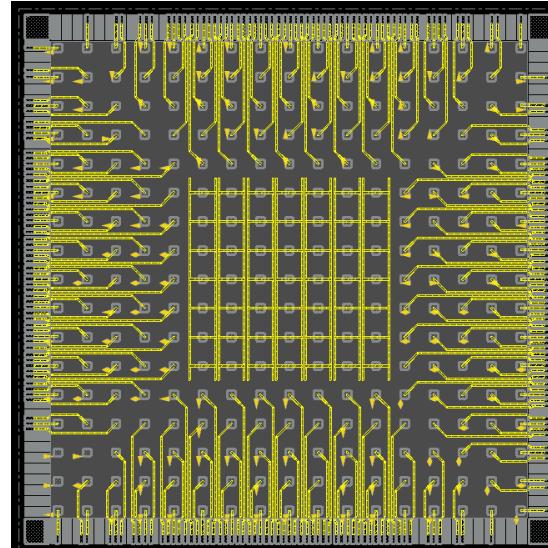
- Starts with:
  - Verilog netlist with signal IO pads for simulation and STA
  - Power/ground IO cells may be present
  - IO cell data (signal, P/G, fillers, ...) from library documentation
- **Footprint file** defines where each padcell is to be placed in the padring – supports reuse of pre-existing padframes
- **Signal mapping file** defines which signal in the Verilog is to be associated with which padcell in the padring
  - + Auto-assignment capability in ICeWall
- **Decouples footprint and signal mapping** for padframe reuse



# ICeWall Pad ring : Present and Future

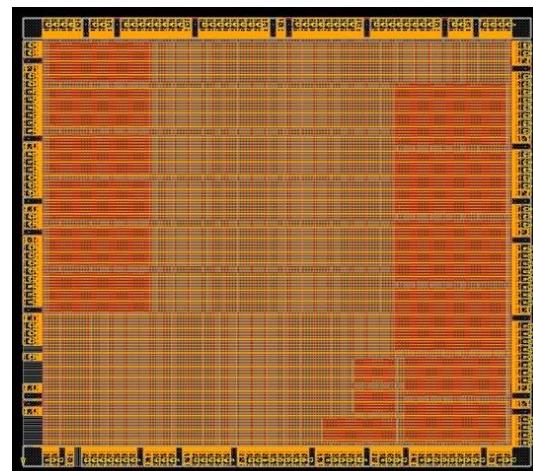


GF12LP BP-1,  
staggered pads



GF12LP BP-1,  
as a flipchip

SKY130 coyote,  
+ pads



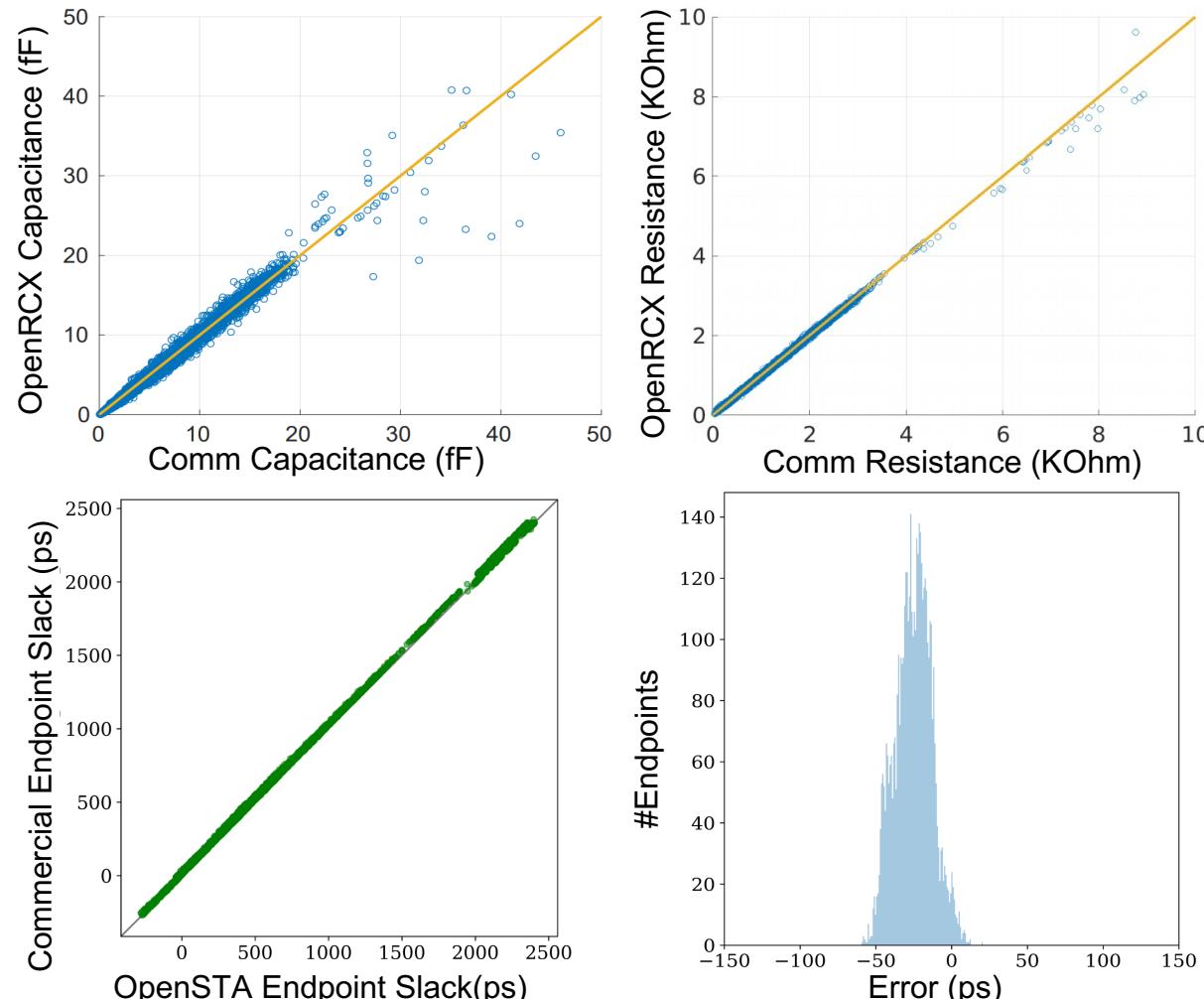
## Next steps

- Determining the number of required P/G pads to be provided as callback functions to allow  to encapsulate specs from library documentation
- Definition of padring segments for analog signals, PHYs, different IO voltages, etc.
- Definition of control cells that are required on a per-IO cell basis

# OpenRCX + OpenSTA Calibration

- OpenRCX brought up and calibrated in:
  - GF12LP
  - CMP28
  - NanGate45
  - TSMC65LP
  - SKY130
- **RC correlation** analysis between OpenRCX and CommRCX
  - Tech: **GF12LP**
  - Design: **jpeg\_encoder (~442K insts)**, **OpenROAD SP&R, 0 DRCs**
  - **Above** the 45-degree line is pessimistic
- **Endpoint slack correlation**
  - OpenRCX + OpenSTA (**x-axis**) vs. CommRCX + CommSTA (**y-axis**)
  - **Above** the 45-degree line is pessimistic

Github: <https://github.com/The-OpenROAD-Project/OpenROAD/tree/master/src/OpenRCX>



# An Academic/Industrial partnership

---

- OpenROAD is a partnership between EDA academic research and industry veterans
- You saw from Andrew's introduction that we have team members from
  - Several universities performing core research
  - Large industrial semiconductor companies providing guidance and priorities
  - Industry consults with extensive EDA experience performing key development
- This unique project and blend of expertise is focused on
  - Breaking new ground in terms of automation of RTL to GDSII
  - Creating a robust industrial quality piece of software
    - Basis for industry relevant research
    - Usable for important target users like the Defense Industrial Base
  - Documenting in open source form how robust EDA tools are put together

# Some Links to Explore

---

- Website: <https://theopenroadproject.org/>
- Docs: <https://openroad.readthedocs.io/en/latest/>
- OpenROAD on GitHub: <https://github.com/The-OpenROAD-Project>
- Email: [abk-openroad@eng.ucsd.edu](mailto:abk-openroad@eng.ucsd.edu) and [aspyrou@eng.ucsd.edu](mailto:aspyrou@eng.ucsd.edu)
- We look forward to telling you more about OpenROAD!



# THANK YOU!

---