


AutoCRAFT: Layout Automation for Custom Circuits in Advanced FinFET Technologies

Hao Chen[1], Walker J. Turner[2], Sanquan Song[2], Keren Zhu[1], George F. Kokai[2] , Brian Zimmer[2], C. Thomas Gray[2], Brucek Khailany[2], David Z. Pan[1], and Haoxing Ren[2]

[1]ECE Department, The University of Texas at Austin

[2]NVIDIA Corporation

This work is supported in part by NVIDIA, the NSF under Grant No. 1704758, and the DARPA ERI IDEA program

Outline

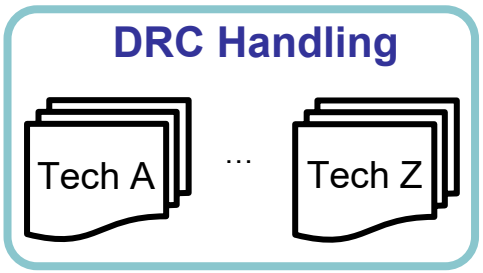
- Introduction
- Background
- AutoCRAFT Framework
  - Preprocessor
  - SMT-based Custom Placement
  - Hybrid Routing Flow
- Experimental Results
- Conclusions and Future Directions

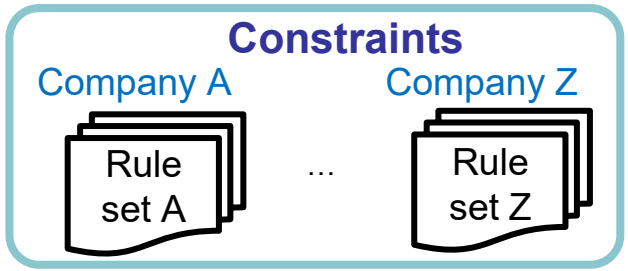
High Demand of Custom Circuits

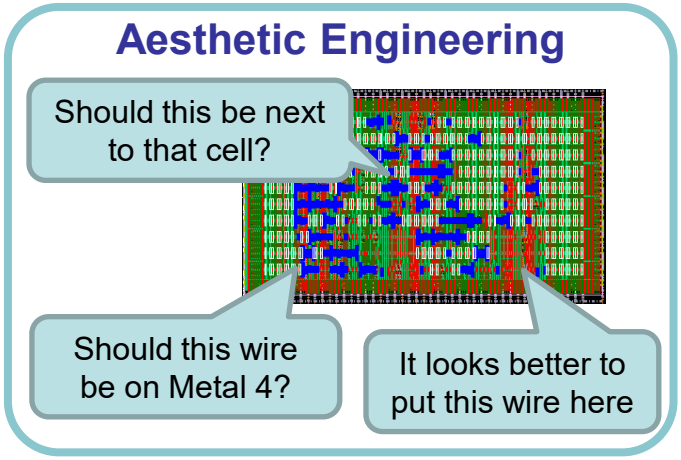
- Analog/mixed-signal (AMS) circuits
  - Internet of Things (IoT), autonomous and electric vehicles, communication and 5G networks…
  - Every sensor-related application needs AMS circuits
- Custom digital circuits
  - High speed/performance designs
  - Ex. Serializer, de-serializer

Challenges of Custom Layout Automation

- Exploding base layers DRC for transistor-level designs
- Complex geometrical/electrical constraints to satisfy performance
- FinFET-induced layout style
- No common evaluation metric for different circuit classes
- Layout design conventions







Increased Attention in Recent Research

- Strict design rules in advanced nodes (e.g., N7, N5) significantly limits the free-form layout process in traditional planar nodes
  - Reduced solution space!!
  - Bridging the gap between human layout expert and automated tools
- ML-assisted approaches show promising applications
  - Context-dependent constraint extraction [*Chen+, DAC'21*], [*Gao+, ASP-DAC'21*]
  - Routing guidance generation [*Zhu+, ICCAD'19*]
  - Parasitic prediction [*Ren+, DAC'21*], [*Shook+, DAC'21*]

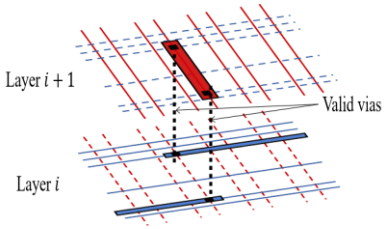
Region-Based FinFET Layout Style

Placement Grids

- Enforce regular row patterns for fin alignment
- The reduced solution space provides advantages to automated tools

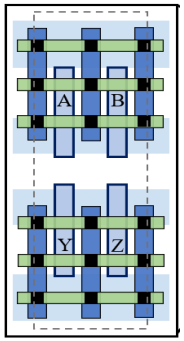
Routing Grids

- Non-uniform track configuration for each layer
- Specify valid via selections on grid points



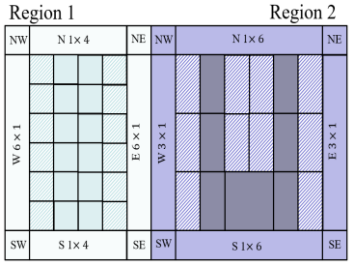
Layout Primitives

- Standard-cell-like basic building blocks formed by low-level devices
- Eliminate numerous complex FEOL design rules

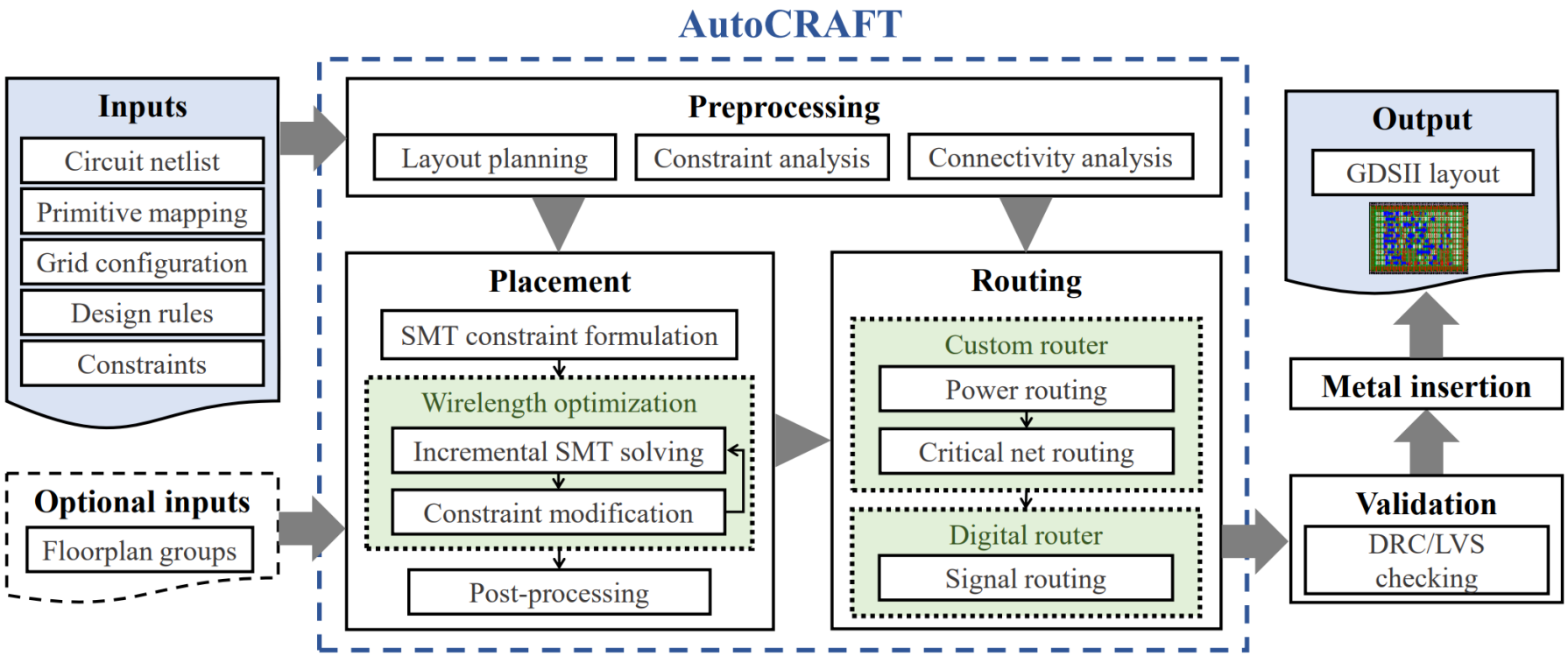


Regions

- Group primitives with similar properties together
- Allow different placement grids setting between regions
- Allow easy base layers DRC
- Enable reusable and stand-alone IPs

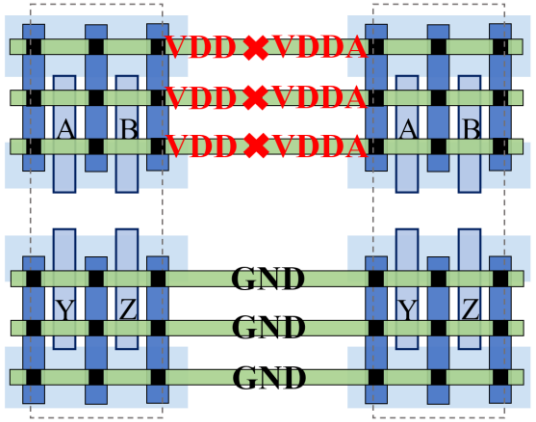


AutoCRAFT Framework

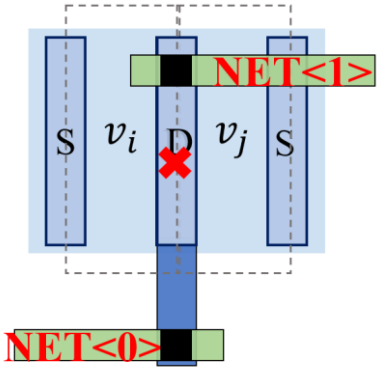


Preprocessing

- **Layout planning**
  - Decide the top-level floorplan according to user-defined aspect and utilization ratios
  - Decide the potential dimensions for each region
- **Constraint analysis**
  - Examine the given P&R constraints to detect conflicts at an early stage
- **Connectivity analysis**
  - Extract supplementary constraints for both power and signal connections
  - Add power abutment constraints to separate primitives with different power configurations to disjoint rows
  - Add extension placement constraints to avoid short violations between primitives with boundary pins



Power connection violations



Signal connection violations

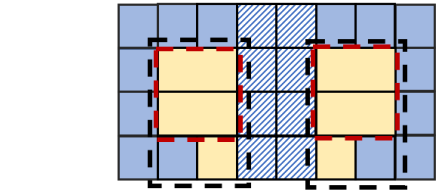
Custom Placement

- **Satisfiability modulo theories (SMT) formulations**
  - Comprehensive constraint handling
  - Flexible to extend for future constraint support
  - Window-based pin density modeling for routability
  - Incremental SMT solving for wirelength optimization
- **SMT variables initialization and scaling**
  - Pure Boolean and bit-vectors (BV) expressions for faster reasoning
  - Variables scaled by the placement grids
    - Directly satisfy the row-based placement requirement
    - Leftover spaces can be filled by dedicated dummy primitives (e.g., de-coupling capacitors)
    - Magnitudes of runtime speedup

Placement Constraints

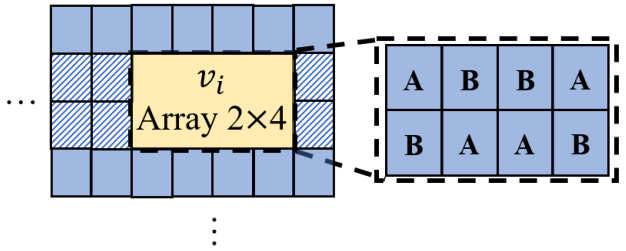
- **Basic constraints for a legal solution**
  - Region constraints (sizing, non-overlapping)
  - Primitive constraints (bounding, non-overlapping)
  - Power abutment constraints
- **Special geometrical constraints for performance**

Hierarchical (self-)symmetry

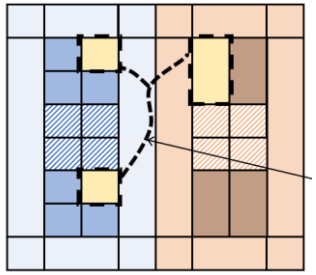


Sym. level 1 (H) Sym. level 2 (V)

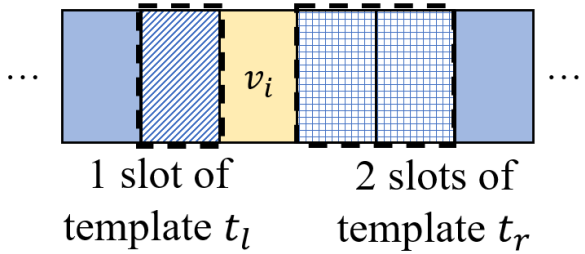
Hierarchical array



Cluster



Extension



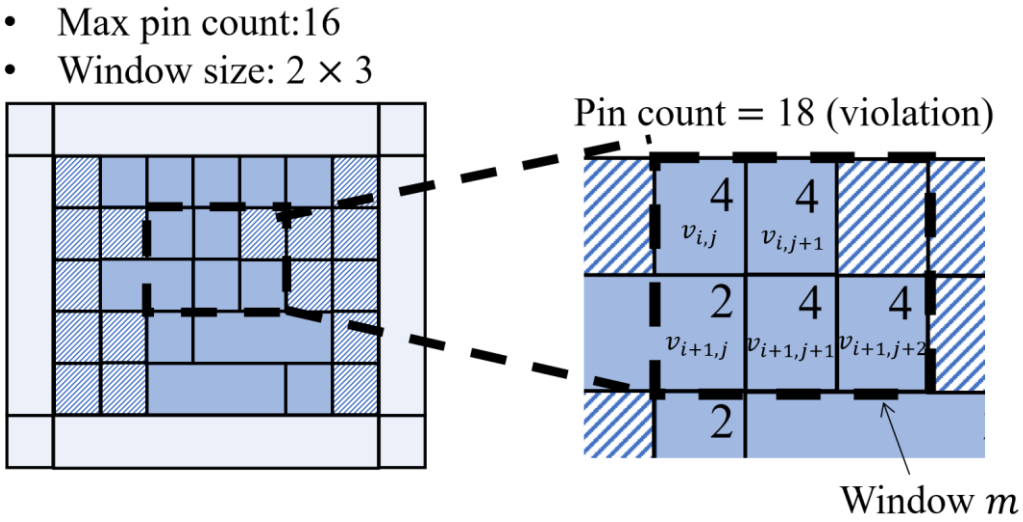
…and more

Placement Routability Consideration

- **Window-based pin density checking**
  - Local pin density is the dominant factor for routability in custom designs
    - Low utilization ratio
    - Complex primitive shapes, and dense power connection structures limit pin-access resources
  - Flexible window dimension
  - Several window configurations can exist and simultaneously apply to the SMT instance

Placement Routability Consideration

- Construct a window set that covers the entire design
- Apply pseudo-Boolean (PB) constraints for maximum pin count restriction



For window m and primitive $v_{i,j}$:

$$overlap(m, v_{i,j}) \rightarrow b^m_{v_{i,j}}$$

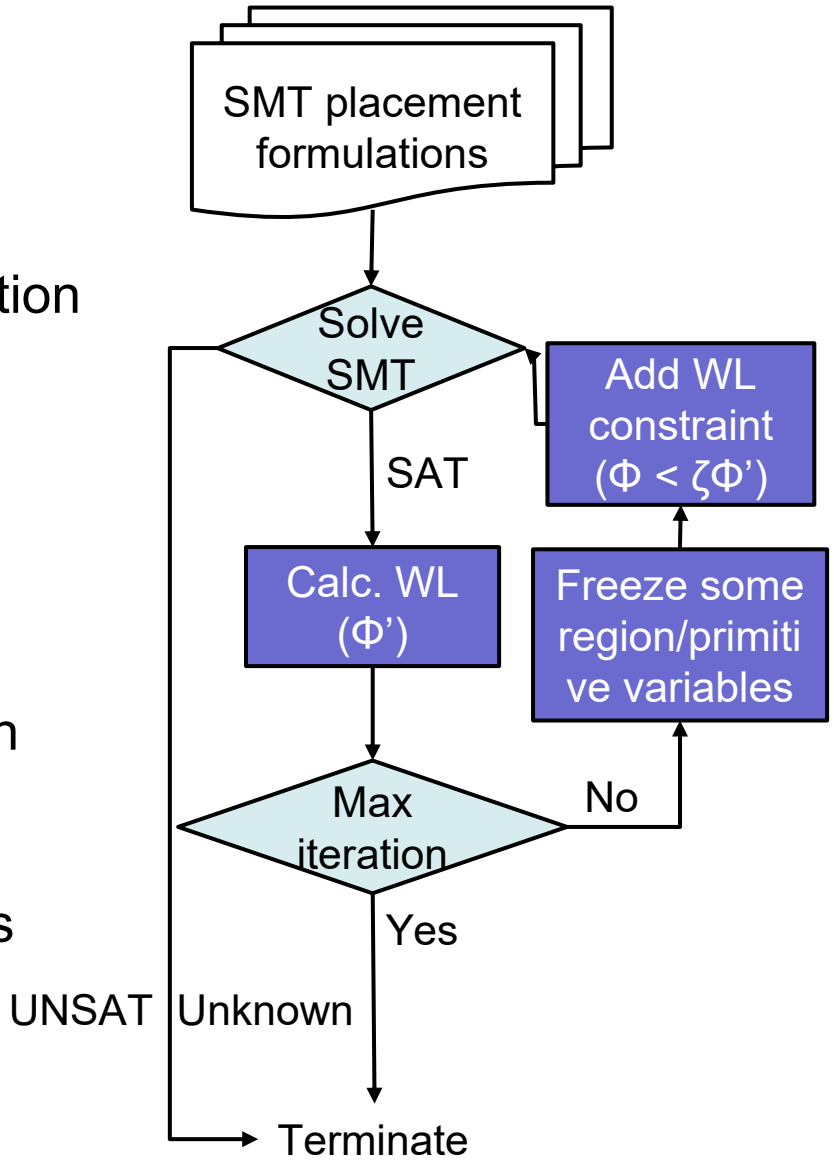
PB constraint:

$$\cdots + 4b^m_{v_{i,j}} + 4b^m_{v_{i,j+1}} + 2b^m_{v_{i+1,j}} + 4b^m_{v_{i+1,j+1}} + 4b^m_{v_{i+1,j+2}} + \cdots \leq 16$$

Detecting the overlaps between windows and cells!

Placement Wirelength Optimization

- **Incremental SMT solving**
  - Improved scalability
  - Efficiently target a near optimal solution
- **Trade-off between solution quality and runtime speedup**
  - Gradually add assumptions to region and primitive variables to preserve wirelength optimization opportunities

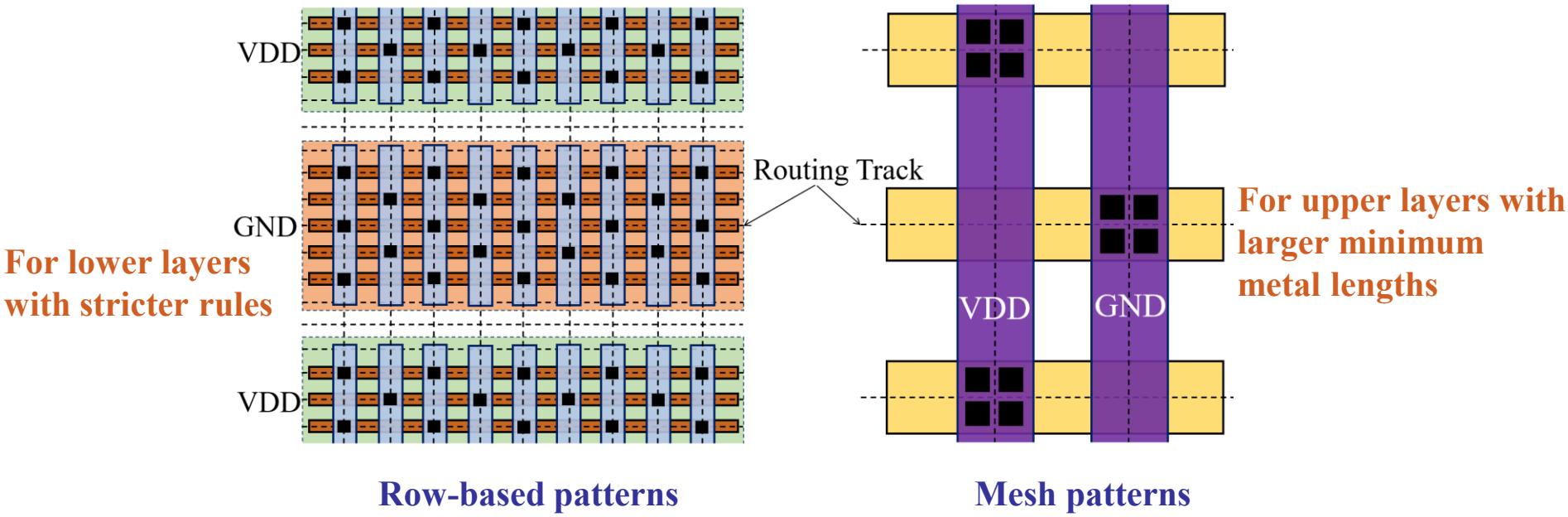


Hybrid Routing Flow

- **Power routing**
  - Specialized patterns to lower overall resistance for better performance
- **Critical net routing**
  - Custom routing algorithms to handle nets with constraints specified
- **Signal routing**
  - For non-critical and insensitive nets
    - Ex. Connections between digital functional blocks, low-speed/DC signals

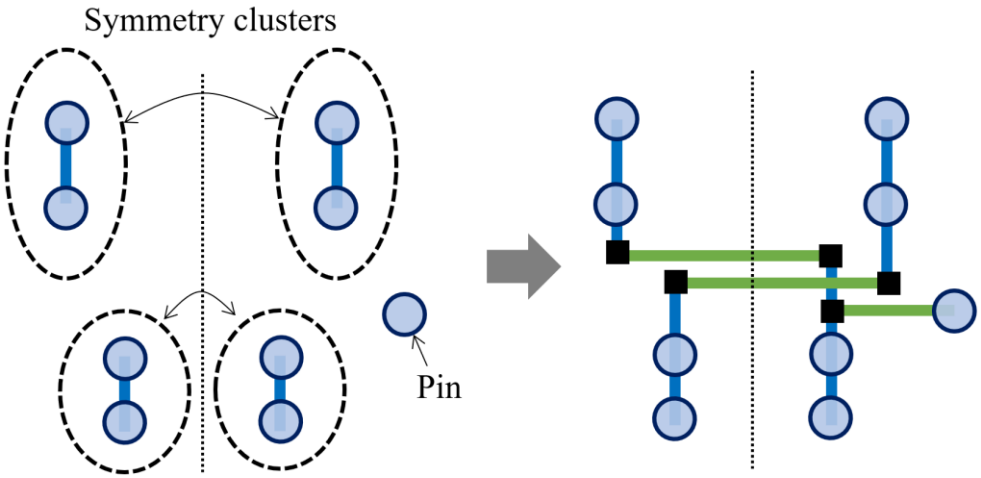
Power Routing

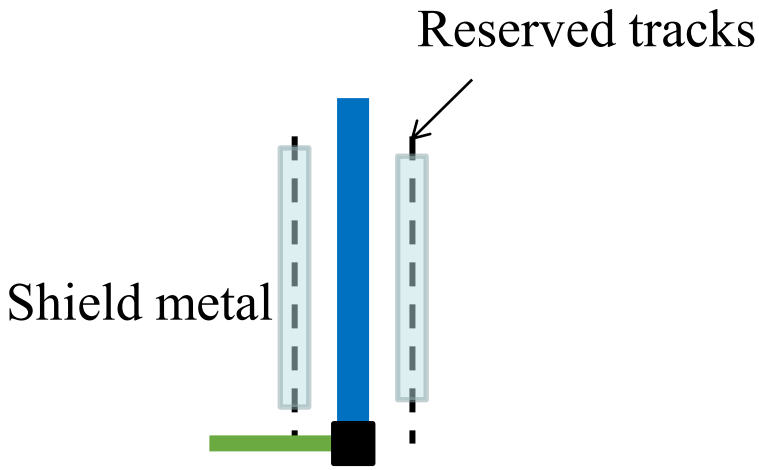
- **Routing on reserved power routing tracks**
- **Different routing strategies selections**
  - Row-based patterns, mesh patterns
  - Select a suitable routing style for each layer



Critical Net Routing

- **Symmetry routing variants**
  - Mirror, cross, partial symmetry
  - Pin clustering for maximum degree of symmetry
- **Net shielding**
  - Reserve neighboring tracks for shielding metals





Signal Routing

- **Handle the remaining connections of non-critical and insensitive nets**
  - Ex. Low-speed, DC signals
- **Automatically generate non-default routing rules**
  - To enforce routing on the grid configurations
  - To use the desired via templates
- **Digital router**
  - Can be commercial or academic tools
  - Ex. ICC2, TritonRoute…etc.

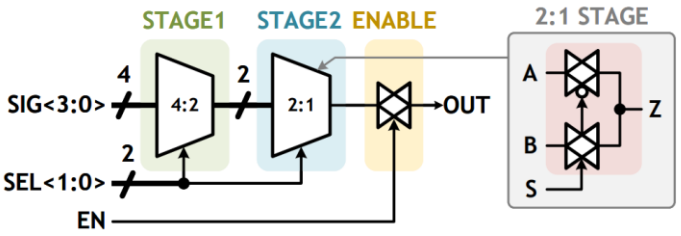
Experimental Results

Setup
- **C++ with Boost, Z3**
- **CPU: Intel Xeon Gold 6136 @ 3.0GHz**
- **RAM: 754GB**

Benchmark circuits
- **MUX-A: 4:1 analog multiplexer**
- **BUF-D: 16:1 digital multiplexing buffer**
- **VCO: 4-stage voltage-controlled oscillator**
- **DES: 1:16 25Gb/s data de-serializer**

Experimental Results

- **4:1 analog multiplexer (MUX-A)**
  - 1 region, 15 primitives, 26 nets
  - Per-stage grouping constraints to minimize parasitics for all internal nodes
  - Hierarchical symmetry constraints for both placement and routing
  - Runtime (Place: 9.9s, Route: 25.4s, Total: 35.3s)

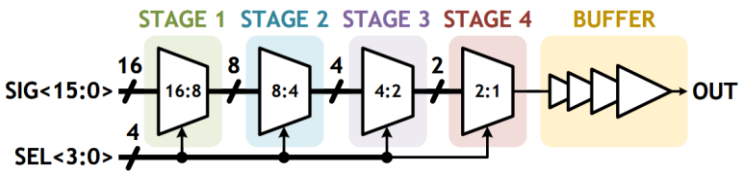


Post-layout parasitics

Stage Output	Node Cap. (fF)		Input Signal	Total Trace Res. (Ohm)	
	Manual	AutoCRAFT		Manual	AutoCRAFT
Stage 1A	1.98	1.64	SIG<0>	587	322
Stage 1B	2.11	1.68	SIG<1>	492	313
Stage 2	1.96	1.45	SIG<2>	451	295
			SIG<3>	541	303
Avg	2.02	1.59	Avg	518	308
Std Dev	0.07	0.10	Std Dev	51.1	10.2

Experimental Results

- **16:1 digital multiplexing buffer (BUF-D)**
  - 1 region, 42 primitives, 66 nets
  - Hierarchical symmetry constraints on all multiplexing circuits in each stage
  - Runtime (Place: 18.8s, Route: 45.3s, Total: 64.1s)

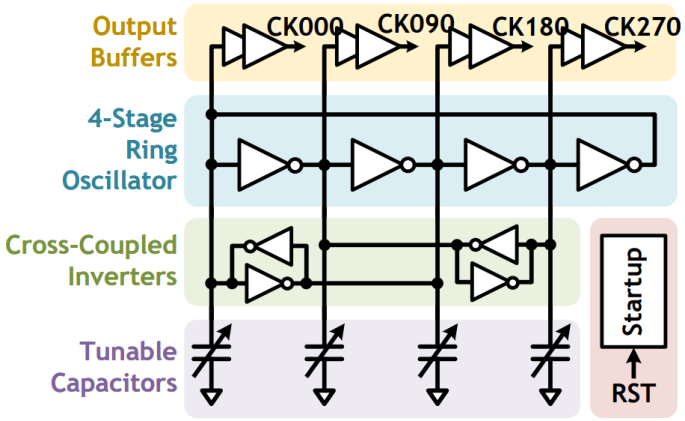


Post-layout insertion delay and rise/fall times

Stage		Insertion Delay (ps)		Rise/Fall Time (ps)	
		Manual	AutoCRAFT	Manual	AutoCRAFT
1	AVG SD	12.3 0.88	**9.5** **0.07**	12.1 / 10.9 0.55 / 0.55	**10.0 / 9.1** **0.06 / 0.06**
2	AVG SD	12.0 0.91	**10.5** **0.17**	12.2 / 12.5 0.77 / 0.59	**10.4 / 10.6** **0.17 / 0.17**
3	AVG SD	12.4 0.50	**11.8** **0.05**	13.4 / 11.6 0.98 / 0.41	**12.0 / 10.9** **0.02 / 0.04**
4	AVG SD	**9.4** 1.00	10.1 **0.15**	10.2 / 10.4 -	**9.2 / 9.2** -
BUFF	AVG	35.8	**35.2**	**5.8 / 5.8**	6.1 / 6.2
Total	AVG SD	82.0 0.92	**77.2** **0.18**	- -	- -

Experimental Results

- **High-speed 4-stage voltage-controlled oscillator (VCO)**
  - 2 regions, 110 primitives, 71 nets
  - Placement geometrical constraints: symmetry, array, cluster, extension, …
  - Pin-to-pin resistance matching routing constraints
  - Runtime (Place: 109.3s, Route: 18.1s, Total: 128.2s)

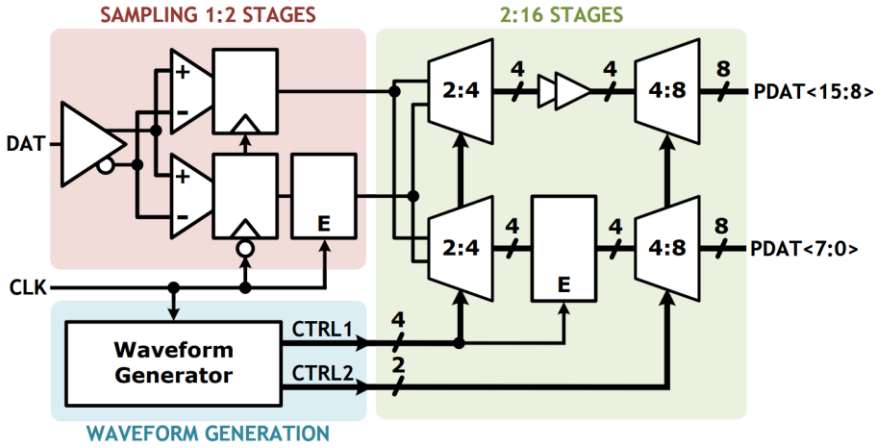


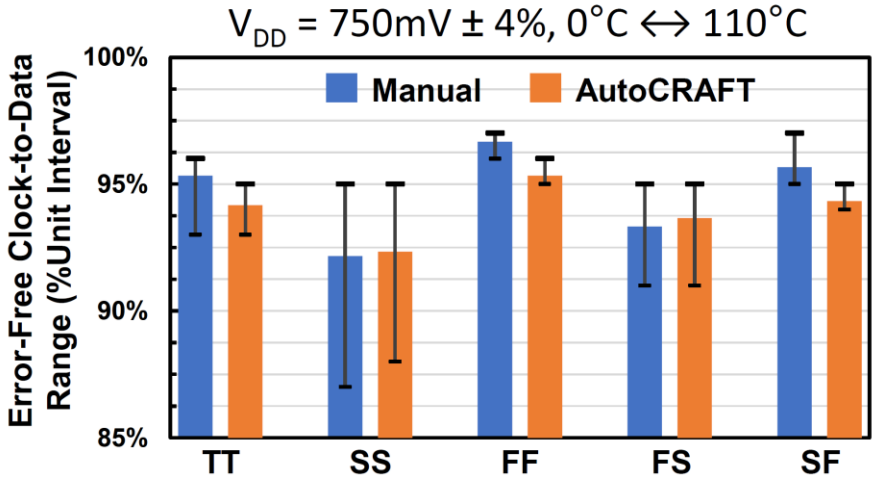
Post-layout oscillation frequency and power

Supply (mV)	Frequency (GHz)		Power (μW)	
	Manual	AutoCRAFT	Manual	AutoCRAFT
650	3.02	3.08	304.4	300.2
700	3.28	3.34	398.8	392.7
750	3.49	3.55	507.5	499.6
800	3.67	3.73	632.4	621.6
850	3.83	3.88	774.6	758.5
900	3.96	4.00	936.0	914.4

Experimental Results

- **1:16 25Gb/s data de-serializer (DES)**
  - 1 region, 92 primitives, 97 nets
  - Placement geometrical constraints: cluster, array, group ordering, extension,…
  - Resistance and capacitance minimizing routing constraints
  - Runtime (Place: 73.0s, Route: 143.2s, Total: 216.4s)





Error-free clock-to-data operating ranges across various PVT corners

Conclusions and Future Directions

- **AutoCRAFT custom layout system**
  - Custom SMT-based placer
    - Comprehensive constraint handling
    - Routability consideration
    - Efficient wirelength optimization
  - Hybrid routing flow
    - Custom routing algorithms for critical nets
    - Integration with digital routers for non-critical signals
- **Future directions**
  - Machine learning/Reinforcement learning techniques for performance-driven layout implementation

Thank you!