

Automation of Standard Cell Layout Generation and Design-Technology Co-optimization

Taewhan Kim, Seoul National University

TUTORIAL, ASP-DAC2025 Tokyo, January 20, 2025



Content

1. Auto-generation of standard cells
 - Design and technology co-optimization (DTCO)
 - Problems
 - Algorithms
 - Multi-row cells
 - Placement legalization
 2. Multi-bit flip-flop (MBFF) cells
 - Structure
 - DTCO flow with MBFF cells
 - DTCO techniques with MBFF cells
 3. Complementary FET (CFET) cells
 - FEOL and BEOL
 - Backside interconnect
 - CFET vs. Flip-FET (FFET)
 4. Conclusion
-

Content

1. Auto-generation of standard cells
 - Design and technology co-optimization (DTCO)
 - Problems
 - Algorithms
 - Multi-row cells
 - Placement legalization
 2. Multi-bit flip-flop (MBFF) cells
 - Structure
 - DTCO flow with MBFF cells
 - DTCO techniques with MBFF cells
 3. Complementary FET (CFET) cells
 - FEOL and BEOL
 - Backside interconnect
 - CFET vs. Flip-FET (FFET)
 4. Conclusion
-

DTCO/STCO

DTCO (design technology co-optimization)

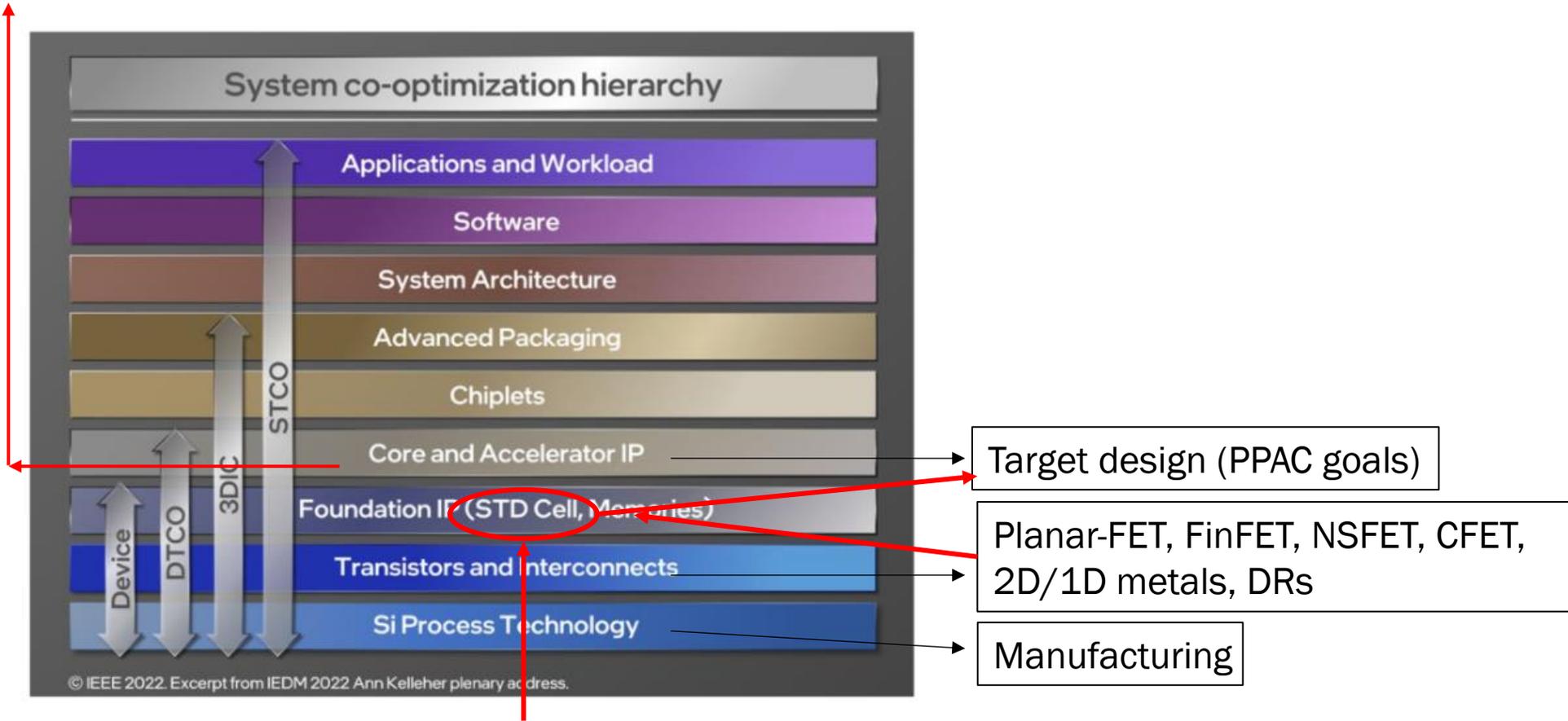
- Optimizing the process technology and chip design together to improve performance, power efficiency, transistor density, and cost.
- DTCO for a new technology node involves substantial architectural innovation rather than just delivering the exact same structure as the previous generation, just simply smaller.

STOC (system technology co-optimization)

- Optimizing the the packaging technology and chip design together.
 - STCO is essential to develop the advanced integration technologies for emerging systems.
 - STCO is required to comprehend not only integration technology, circuits, architectures and software but also their interactions with the power delivery, cooling and system costs.
-

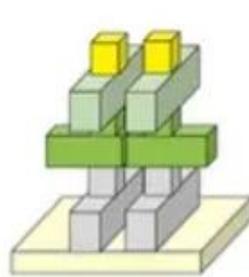
DTCO/STCO

RTL-to-GDSII

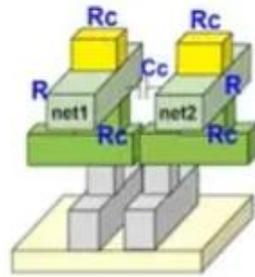


Cells: SR/MR, min-area, min-delay, min-power, max-yield, pin access

TSMC's DTCO

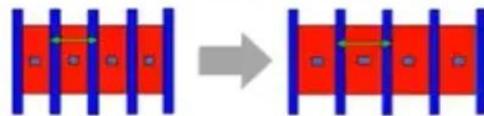


Min. Metal Pitch for Mobile Designs



Wide Metal Pitch with larger VIA for HPC Designs

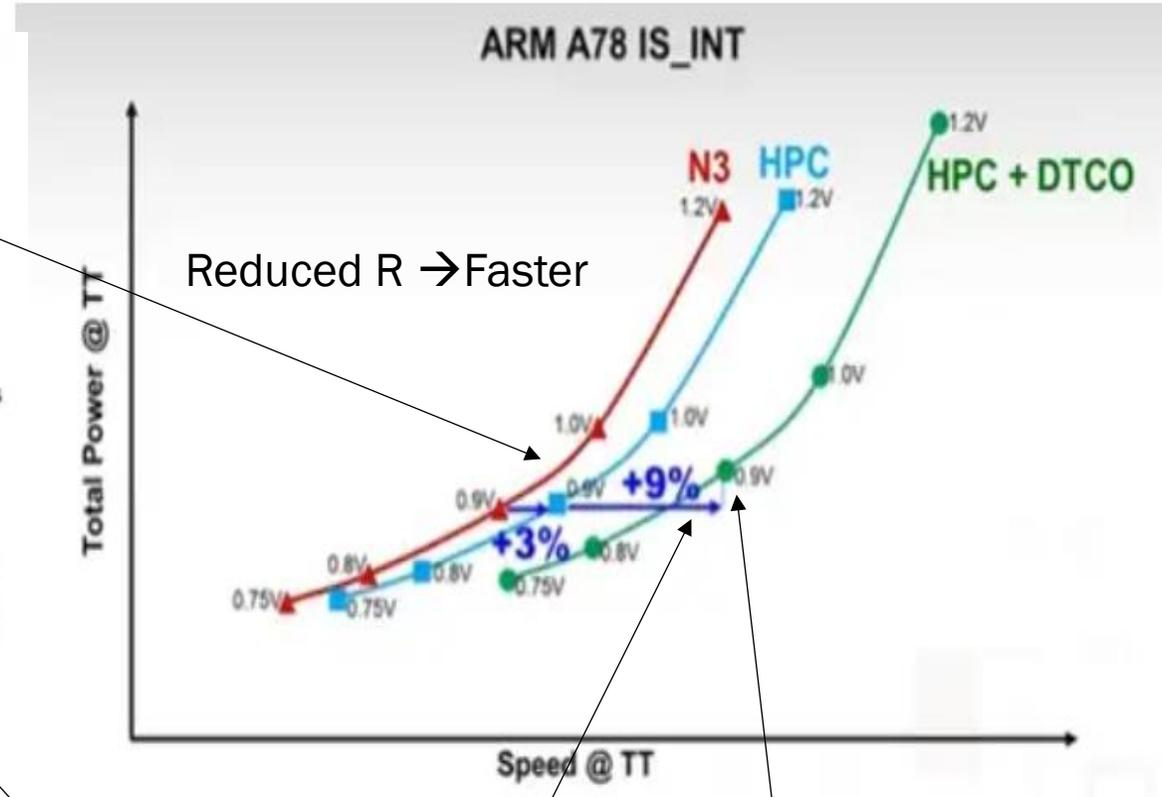
- **Larger Cell Height and Larger CPP**
 - Larger cell height to drive long wires
 - Larger CPP device for higher speed



Larger Cell CPP



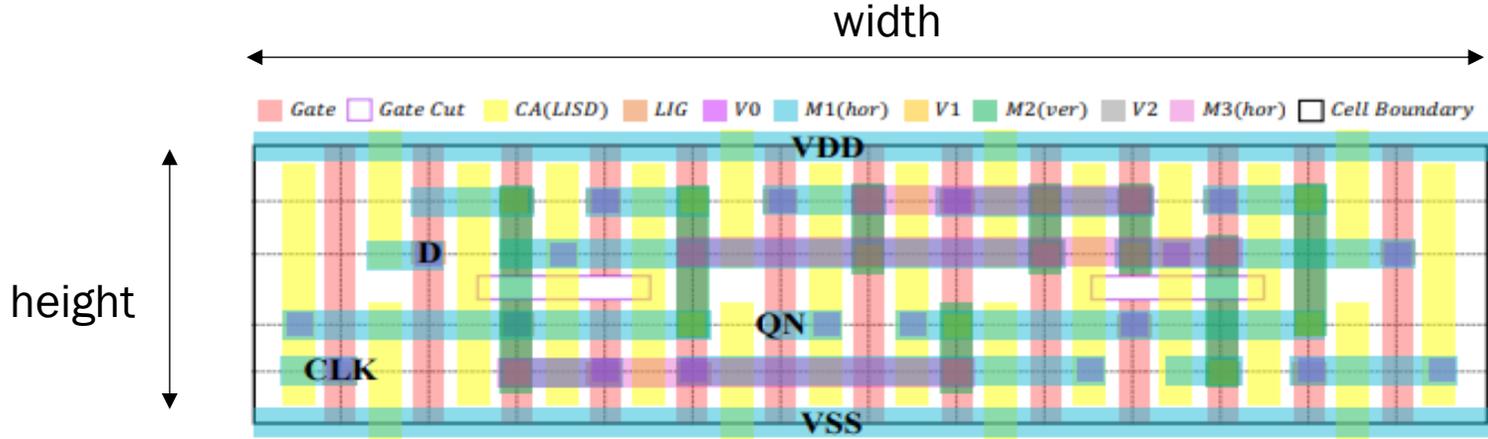
Larger Cell Height



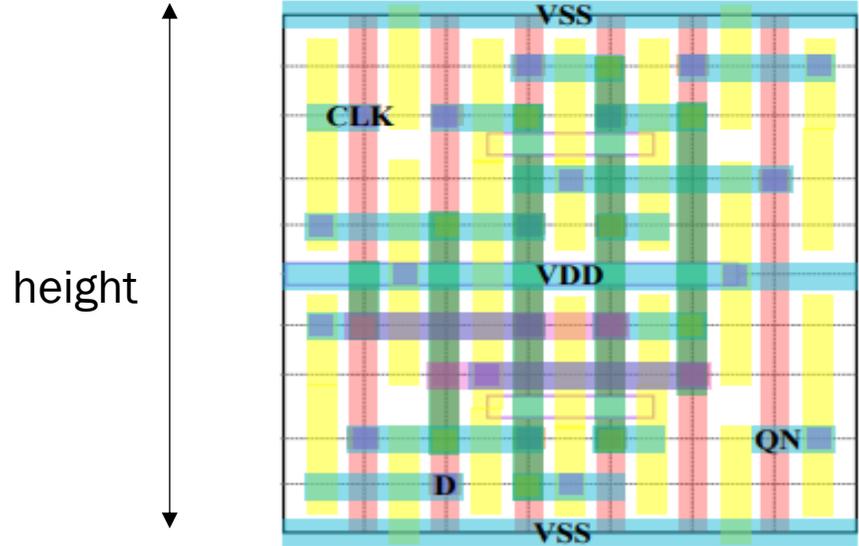
Reduced R → Faster

More fins → Faster, but more power

Standard Cell Example: DFFHQNx1 [Chung, ICCAD24]



(a) Single-row layout of 14 CPPs



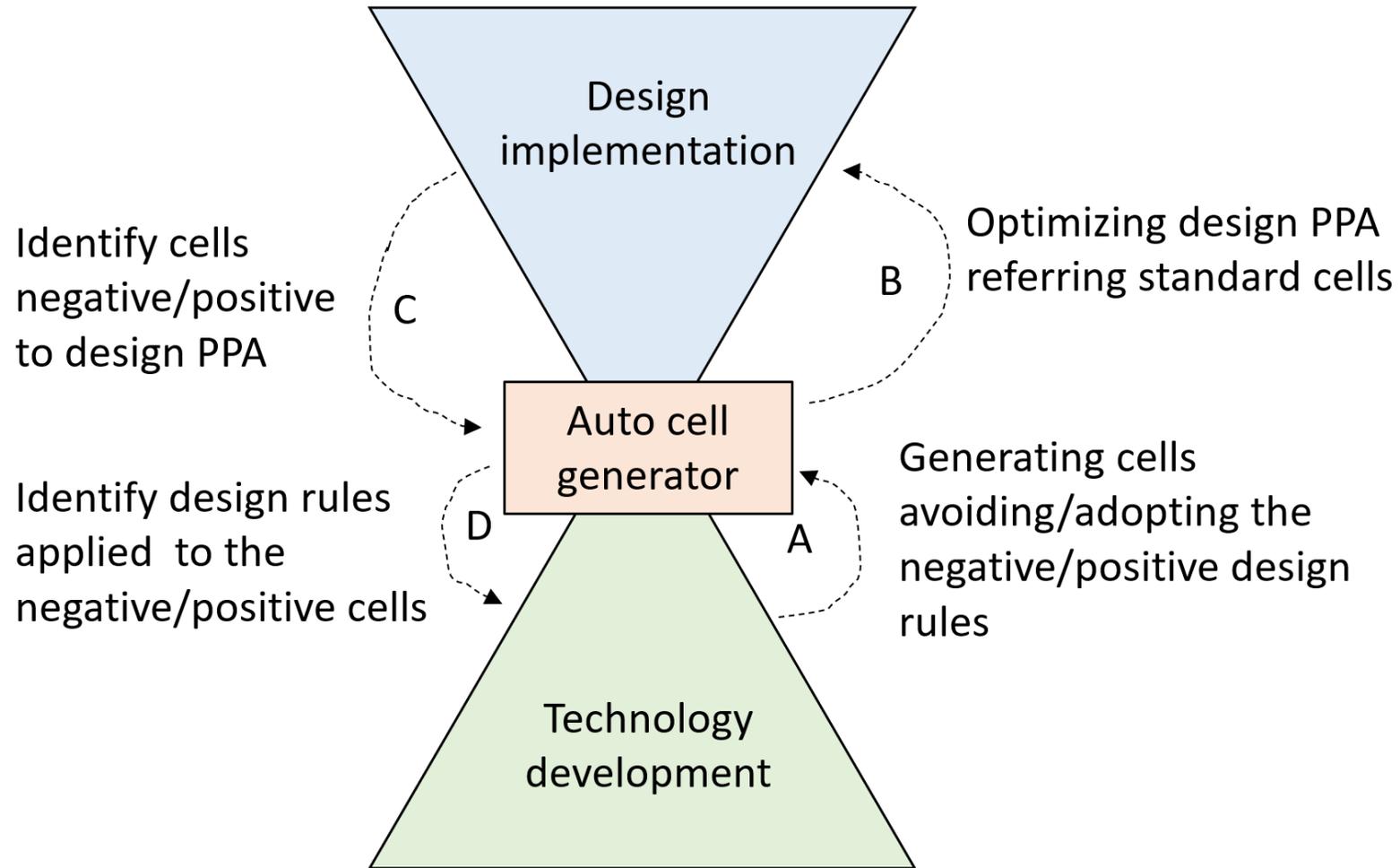
(b) Double-row (VSS-abut) layout of 7 CPPs produced by our generator.

	Single-row	Double-row
Area	14	14
Aspect ratio (#Row/#CPP)	$\frac{1}{14}$	$\frac{2}{7}$
RPA(D)	1	2
RPA(CLK)	1	1
RPA(QN)	1	1
#M3T	3	2
#M2T	8	5

(c) Comparison of two cell layouts.

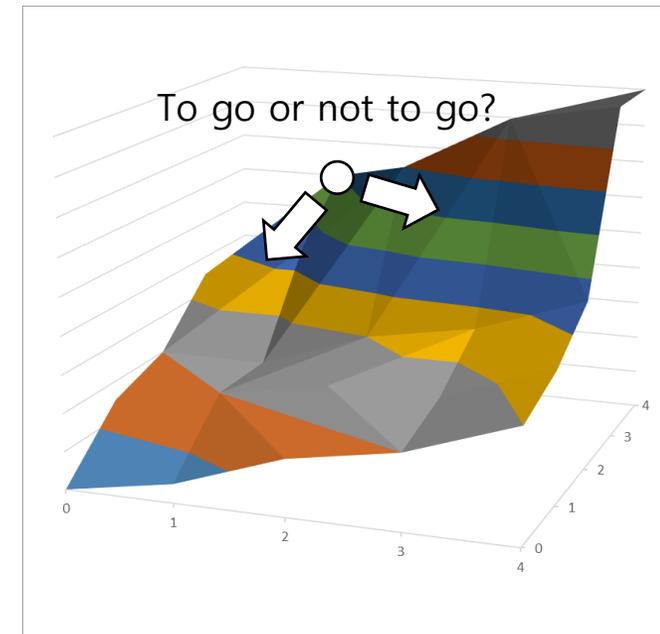
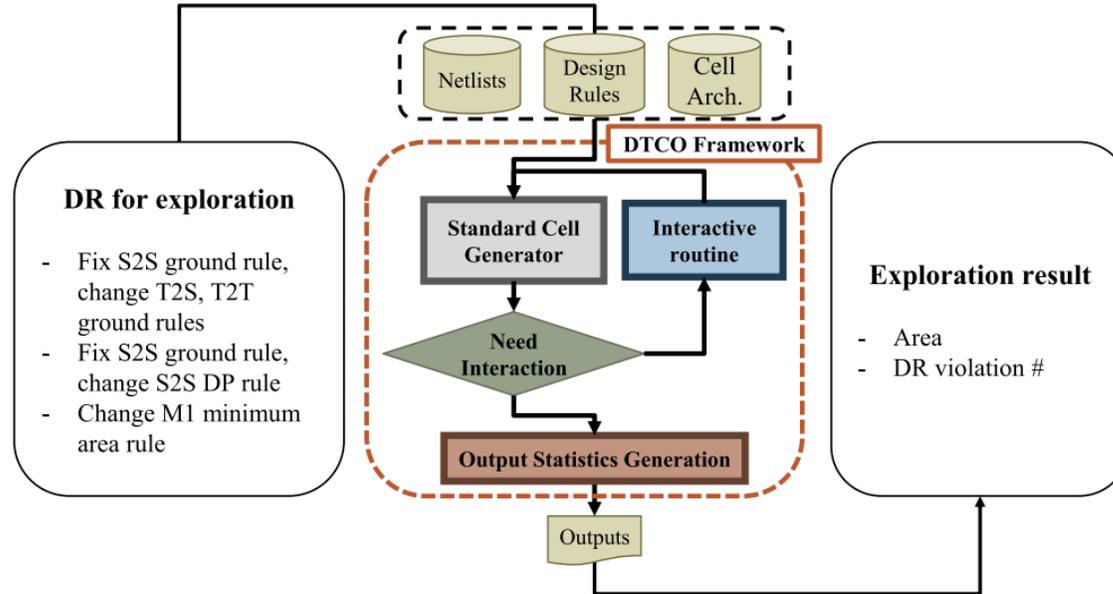
RPA: A probabilistic measure of the number of access points on a particular pin in a cell that can be "accessible".

DTCO Flow using Auto Cell Generator [Kim, MWSCAS23]

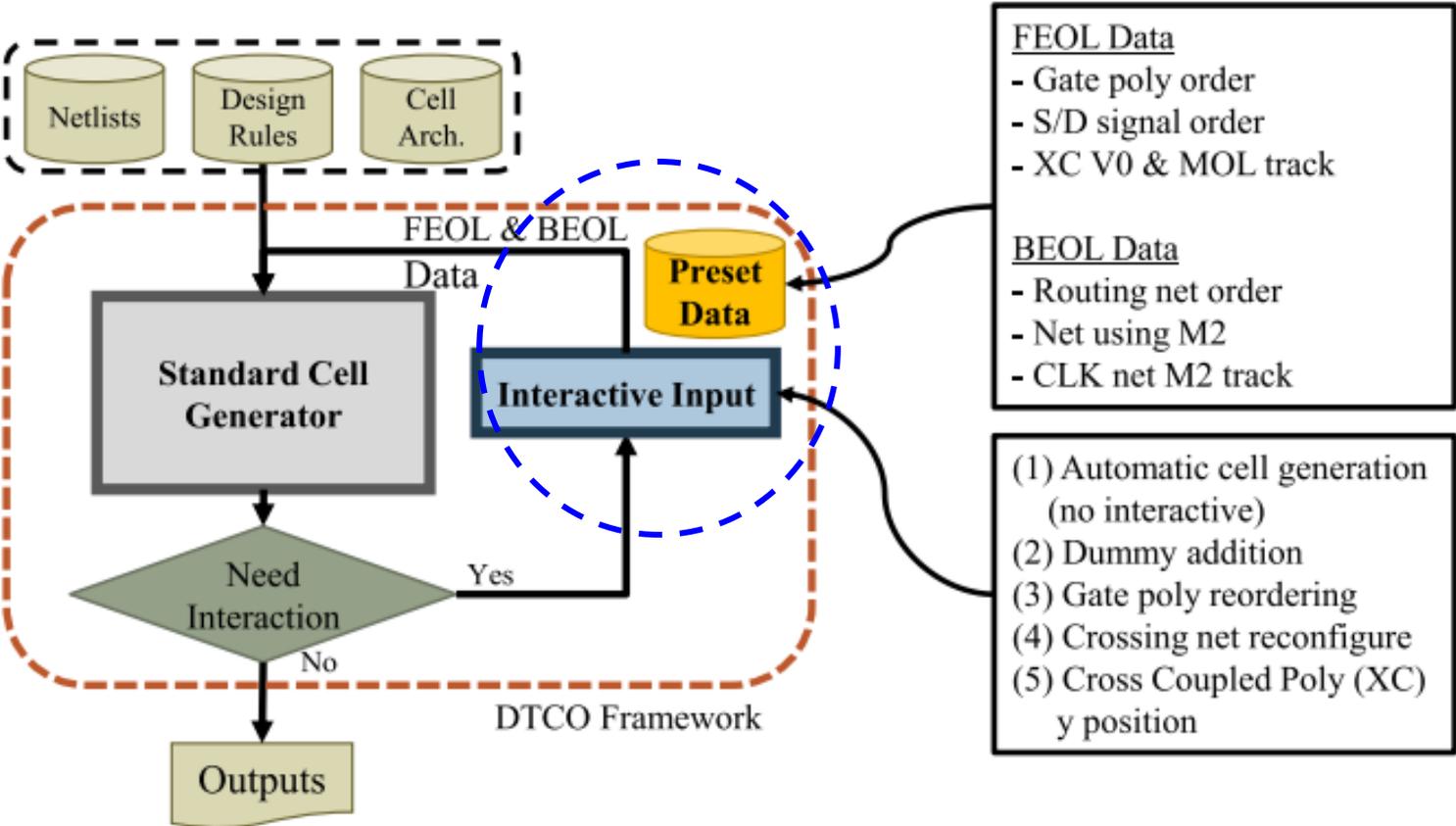


Design Technology Co-optimization Flow [Jo, TVLSI2019]

- Explore the effects of the design rule changes
 - Major metric: The number of design rule violations
 - Find the inflection points – meaningful information or decision point

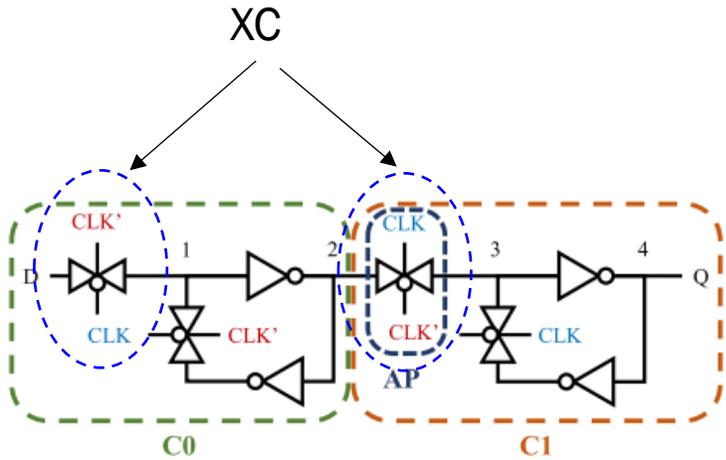


Semi-automation

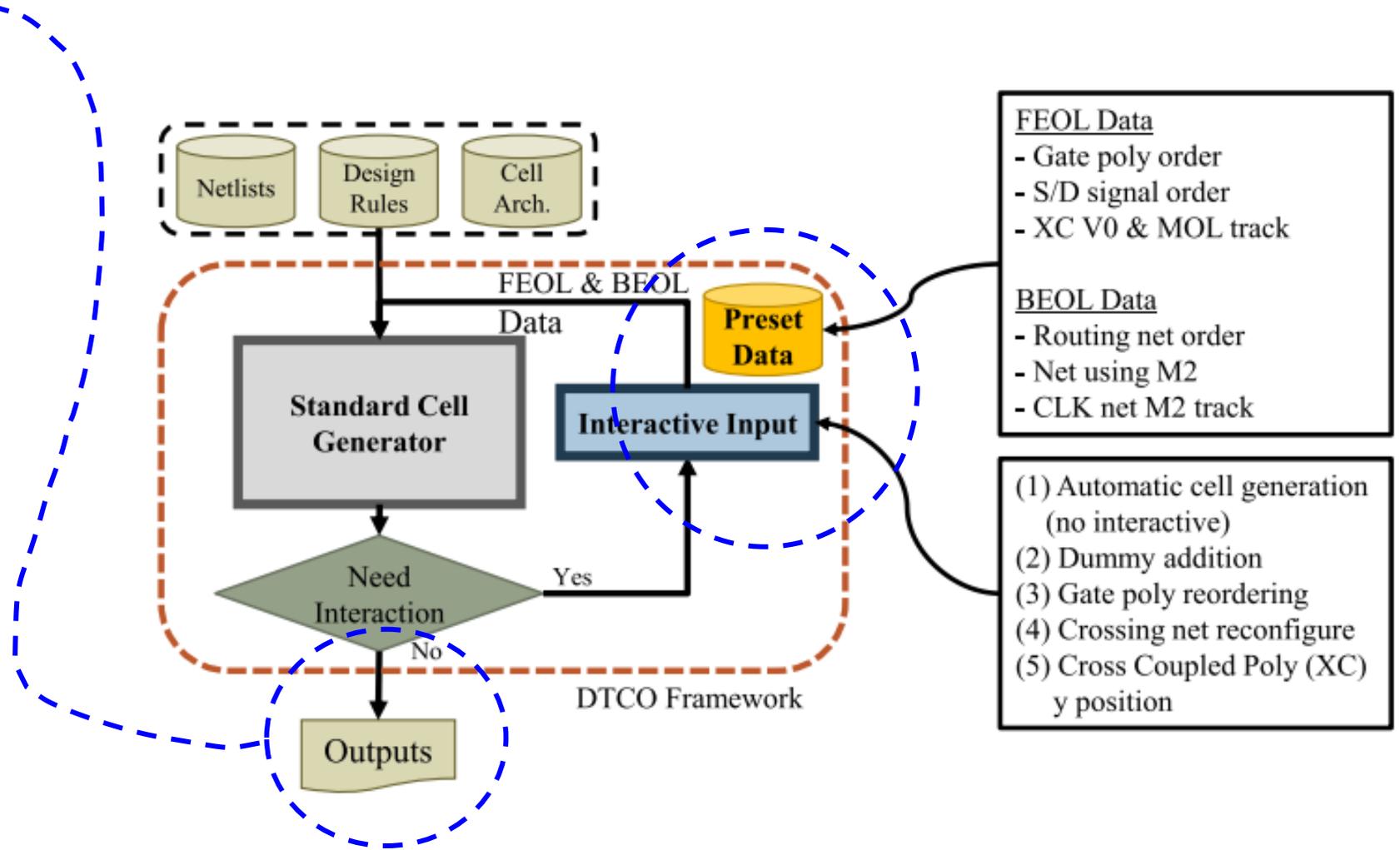
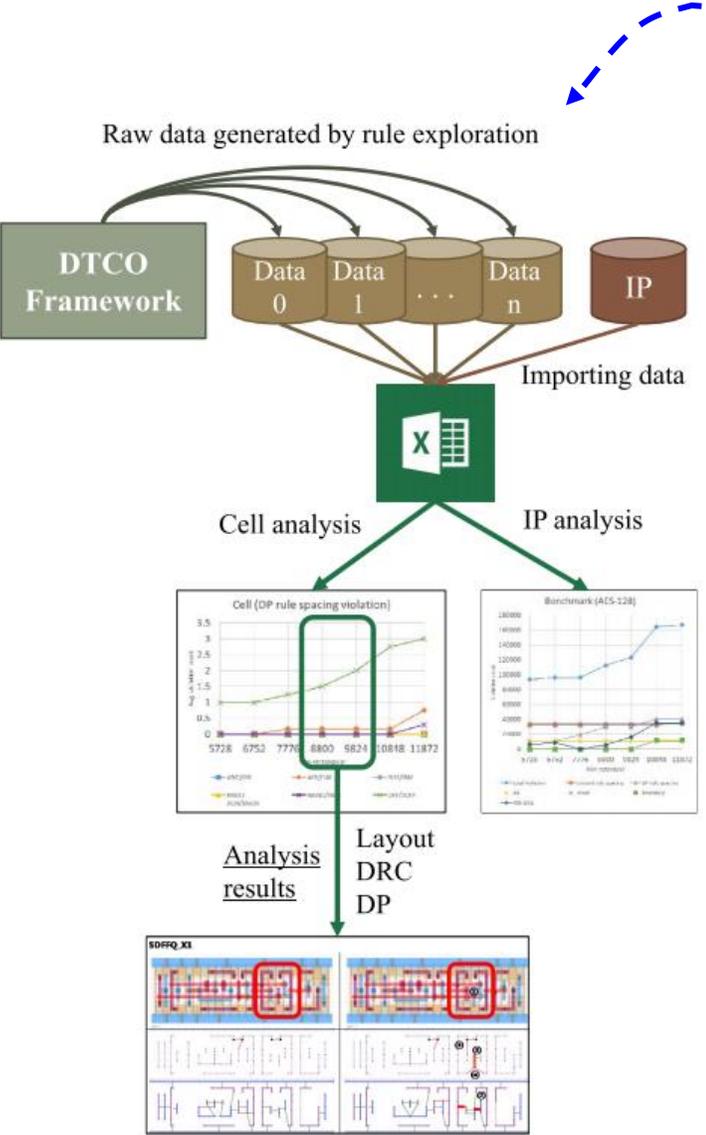


- FEOL Data**
- Gate poly order
 - S/D signal order
 - XC V0 & MOL track
- BEOL Data**
- Routing net order
 - Net using M2
 - CLK net M2 track

- (1) Automatic cell generation (no interactive)
- (2) Dummy addition
- (3) Gate poly reordering
- (4) Crossing net reconfigure
- (5) Cross Coupled Poly (XC) y position

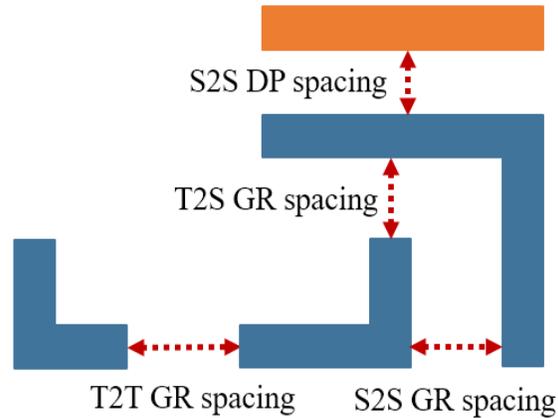


Output Analysis



DR Exploration Examples [Jo, TVLSI2019]

- Baseline design rules



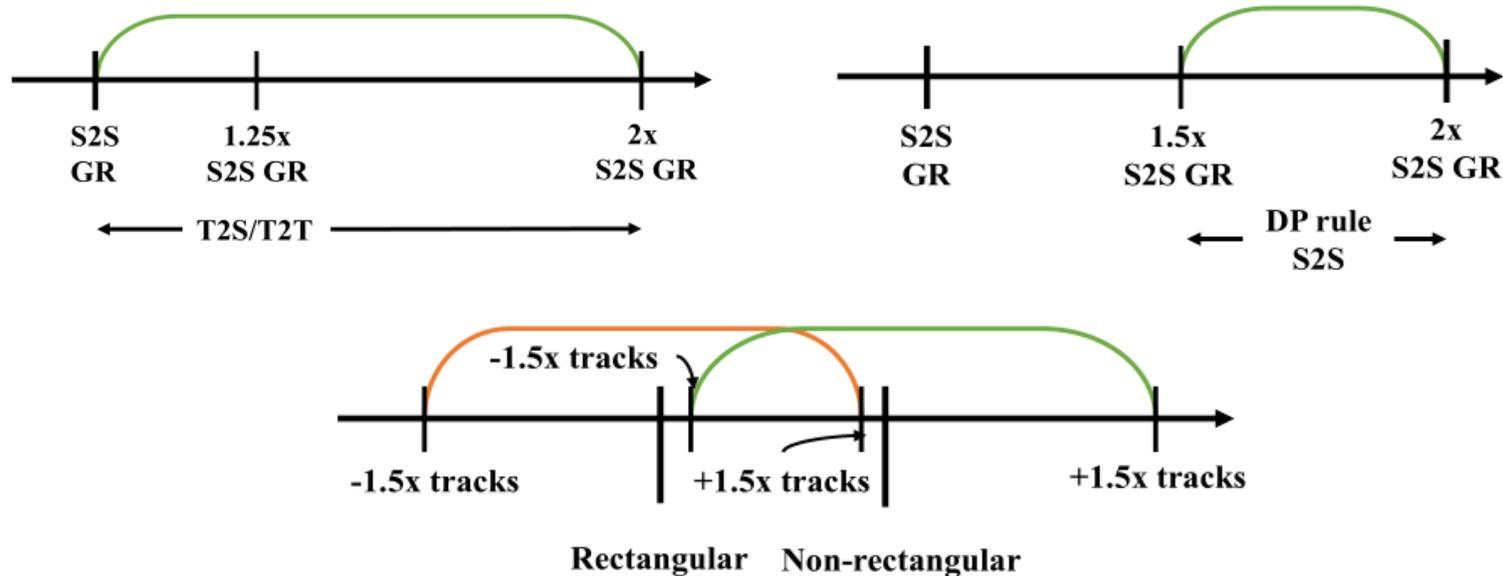
DR Name	#baseline
S2S GR	32nm
T2S/T2T GR	40nm
S2S DP rule	64nm
T2S/T2T DP rule	80nm
M1 minimum area(rect)	5040nm ²
M1 minimum area(non-rect)	8064nm ²

- IP blocks

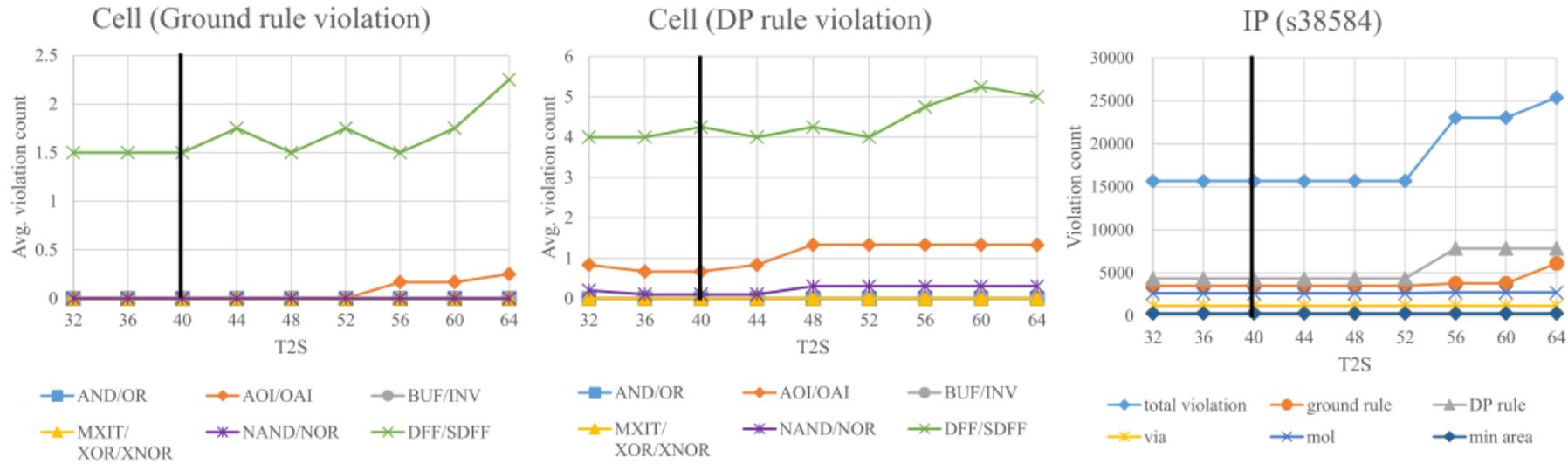
IP	Description	Cell #
s38584	ISCAS '89 benchmark	~7k
Nova	H.264/AVC Baseline Decoder	~109k
AES-128	AES-128 Encryption	~134k
openMSP430	16bit microcontroller core	~6k
USB	USB 2.0 Function Core	~9k

Items for Exploration

- Change T2S, T2T GR (ground rules) with fixed S2S GR
- Change S2S DP (Double Patterning) rule with fixed S2S GR
- Change M1 minimum area rule



Analysis: Change T2S, T2T GR with fixed S2S GR



Impact:

Affects GR violation and DP rule violation

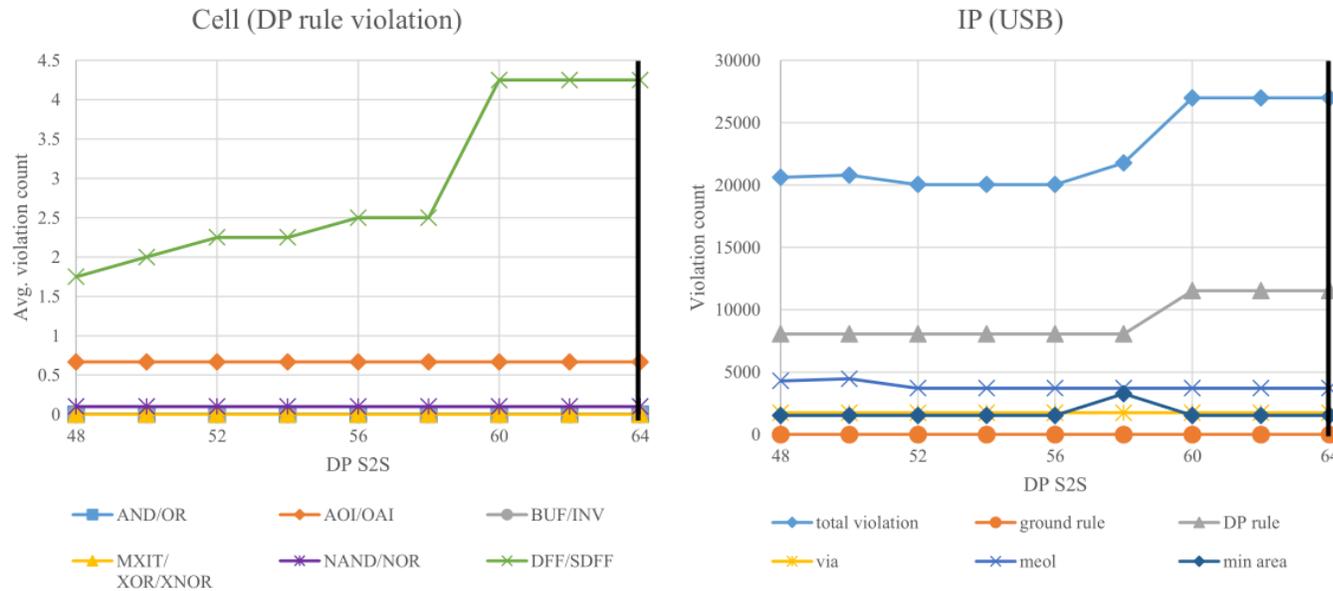
Violations start to increase from 56nm (larger than the baseline)

Reason of violation

Occurrence of coloring conflicts between adjacent metals

→ Possibility of increasing T2S/T2T GR from the baseline

Analysis: Change S2S DP rule with fixed S2S GR



Impact

Affects DP rule violation

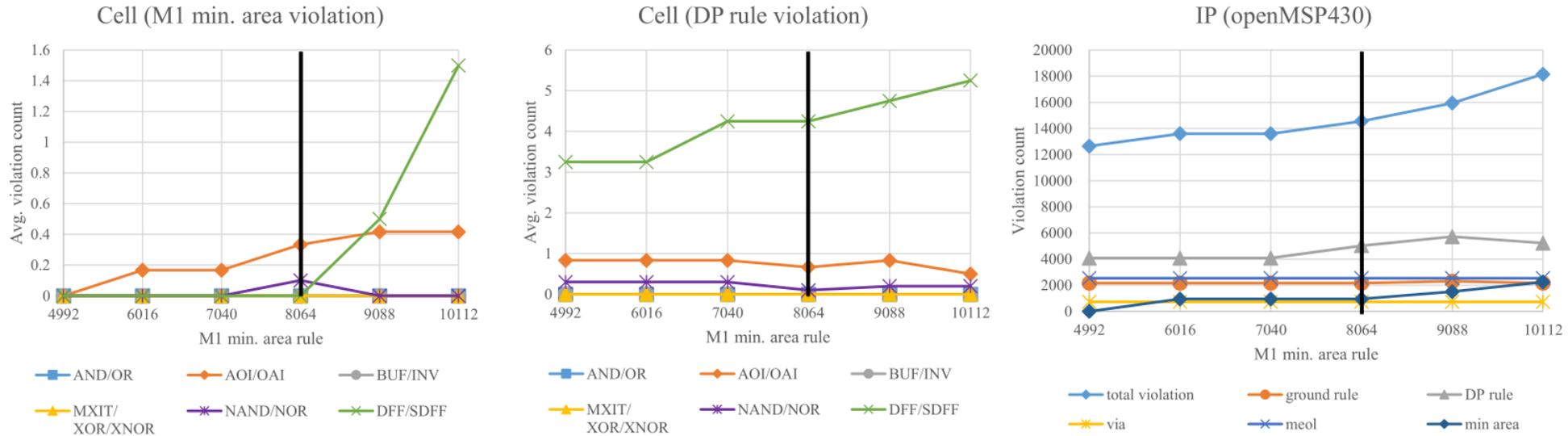
Violations start to increase from 58nm (smaller than the baseline)

Reason of violation

S2S DP rule barely affect layout itself but only coloring of metal pattern

→ Possibility of decreasing S2S DP rule from the baseline

Analysis: Change minimum M1 area rule



Impact

Affects min. area violation and DP rule violation

Violations start to increase immediately after the baseline

Reason of violation

Some M1 patterns cannot be enlarged due to adjacent metals

DP odd cycle can occur due to enlarged metal pattern

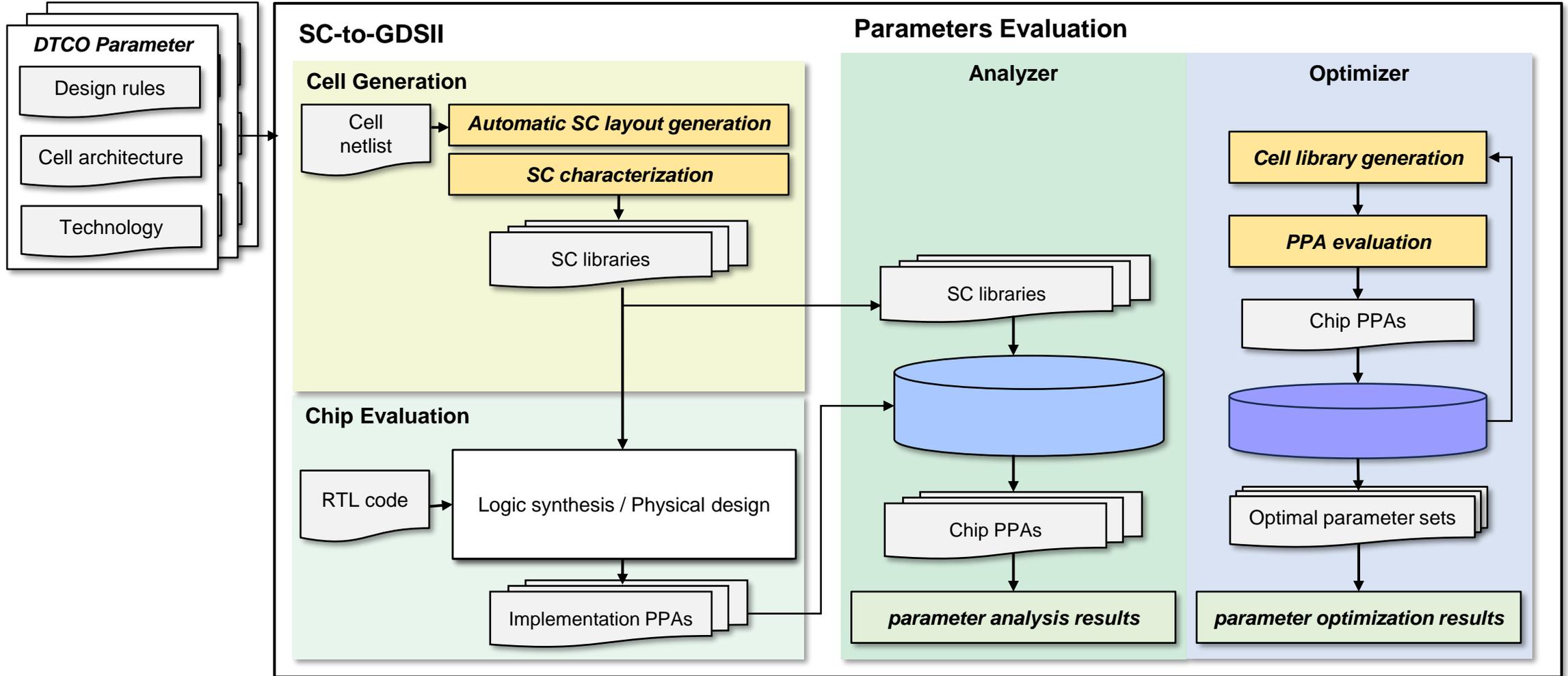
→ Possibility of additional optimization as the baseline is in the middle of transition

DR Exploration Summary

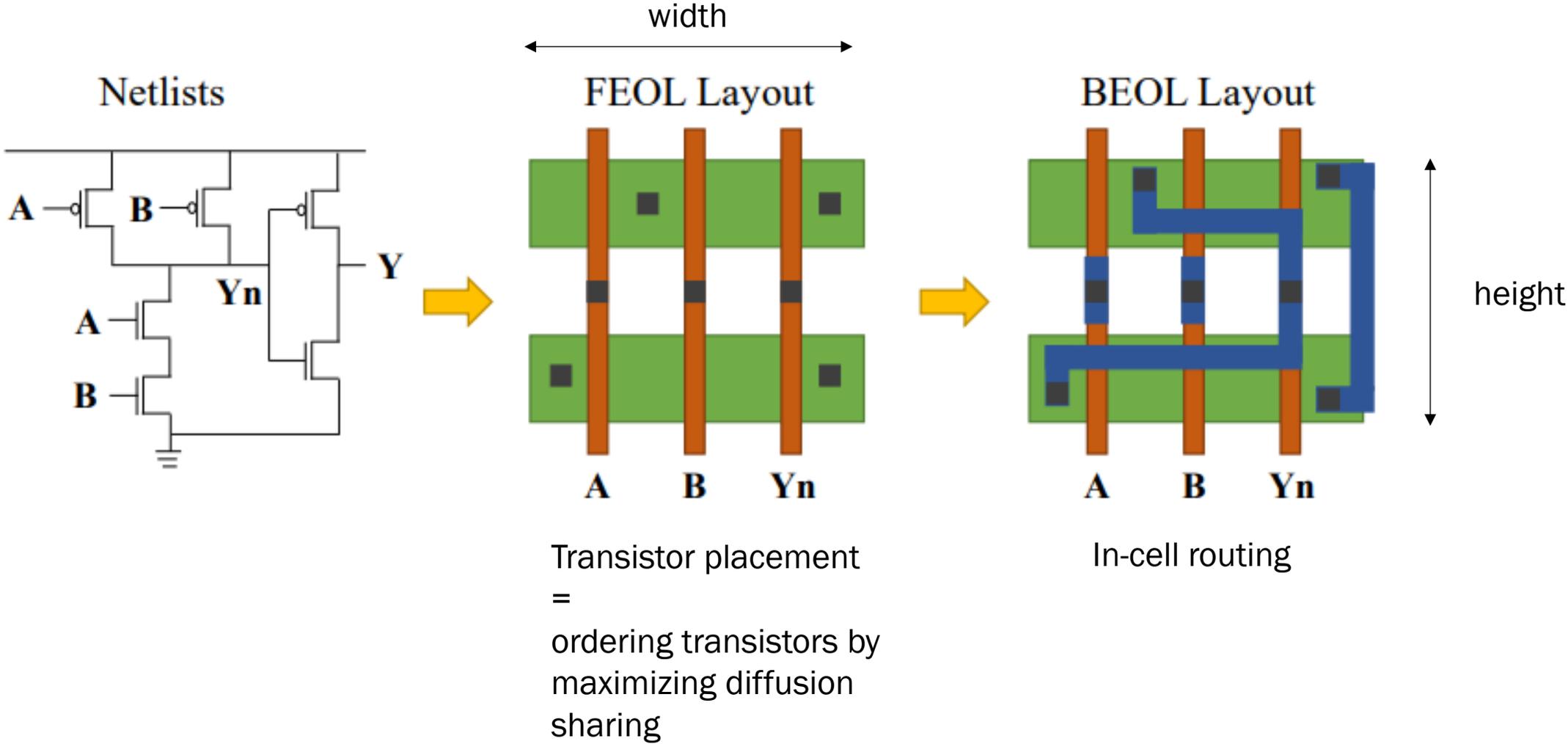
Design rule	Baseline	After exploration
S2S GR	32nm	-
T2S/T2T GR	40nm	Can increase (process-friendly)
S2S DP rule	64nm	Can decrease (design-friendly)
M1 min. area (rect)	5040nm ²	Another optimization
M1 min. area (non-rect)	8064nm ²	

DTCO Framework

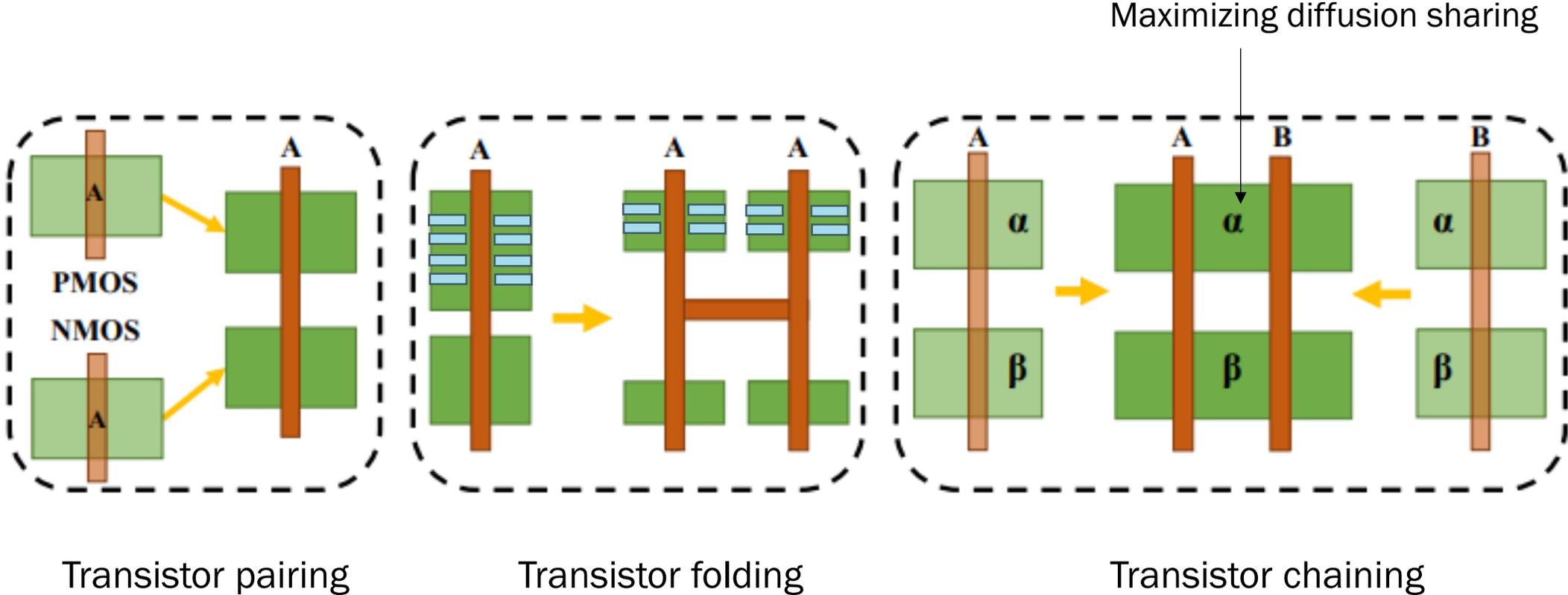
Complete Framework



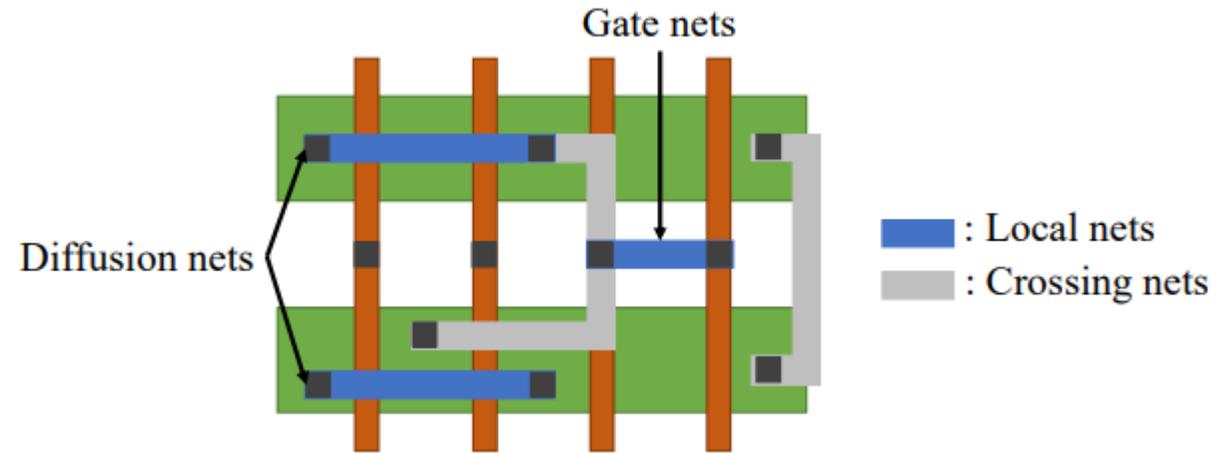
STD Cell Generation Problems



Transistor Placement

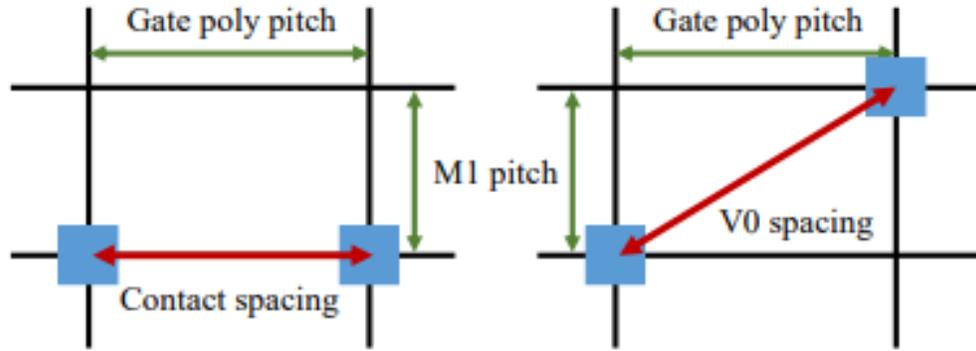


In-cell Routing

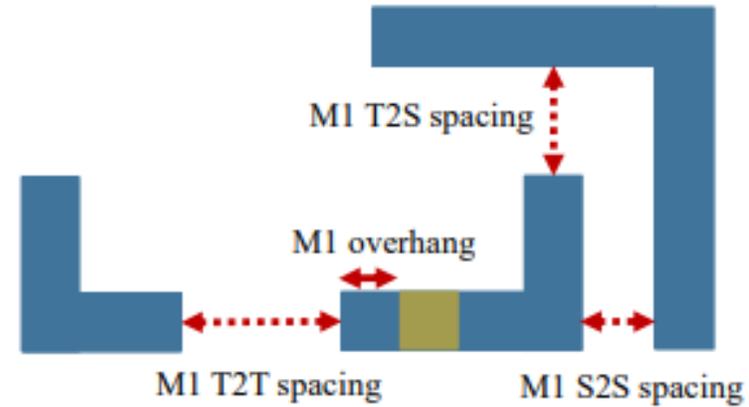


1. Net routing
2. Generate pin pattern generation – should consider pin accessibility
3. Maximize the use of M0, M1 metals
4. Minimize the use of M2 metals
5. Should consider design rule constraints

Design Rule Examples



(a) V0 spacing

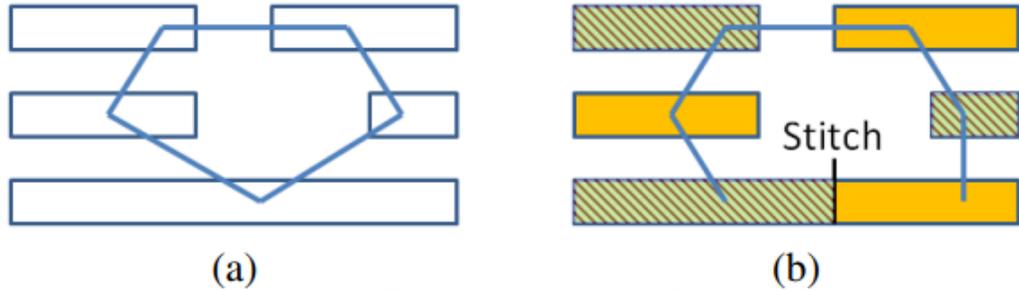


(b) M1 overhang and M1 spacing rules

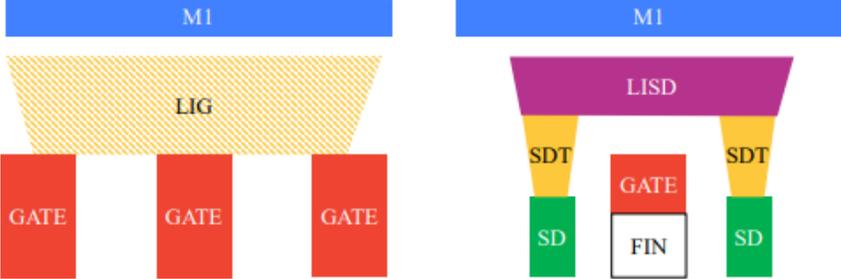
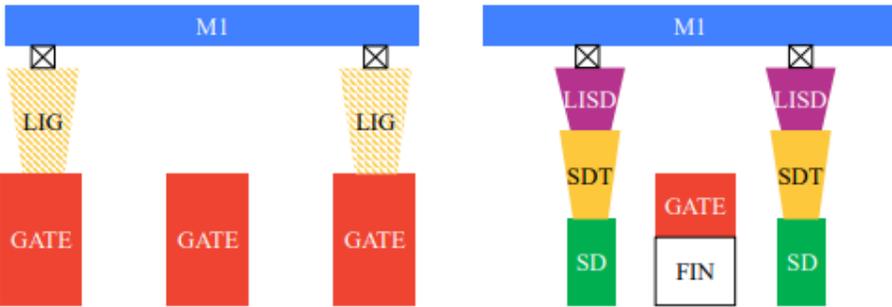
Major Design Rules

- M1 spacing : S2S/T2S/T2T tip relations
 - DP (double patterning) rule
 - V0 center-to-center spacing: the Euclidean distance between the centers of two V0 instances;
 - MOL spacing, indicating the Manhattan distance between two contacts on LISD and/or LIG;
 - M1 minimum area: the minimum feature size of M1 layer.
-
- ✓ The spacing violation occurs when the measured distance is shorter than the corresponding DR.
 - ✓ The area violation occurs when the measured area of a pattern is smaller than the DR corresponding to its shape (rectangular/nonrectangular).
-

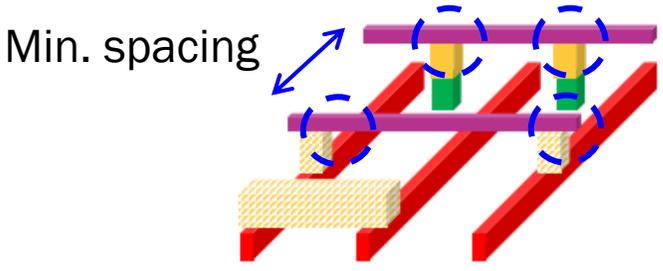
DP and MOL Rules



For legal masking



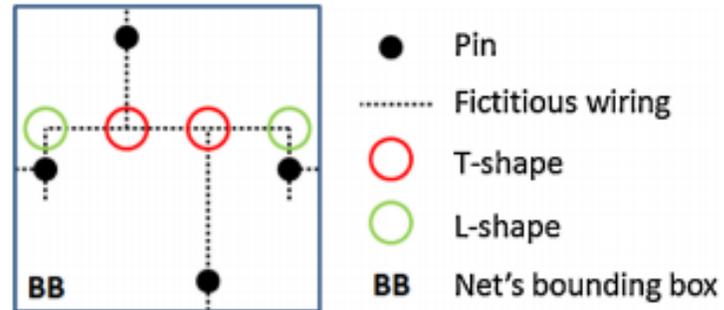
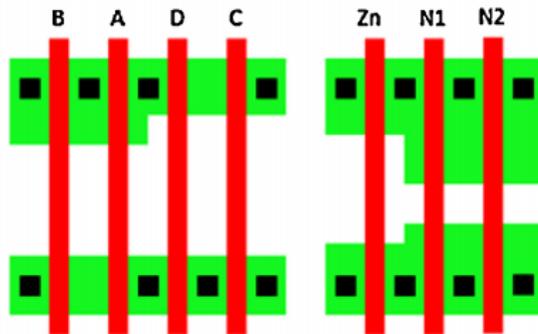
(a) (b)



(c)

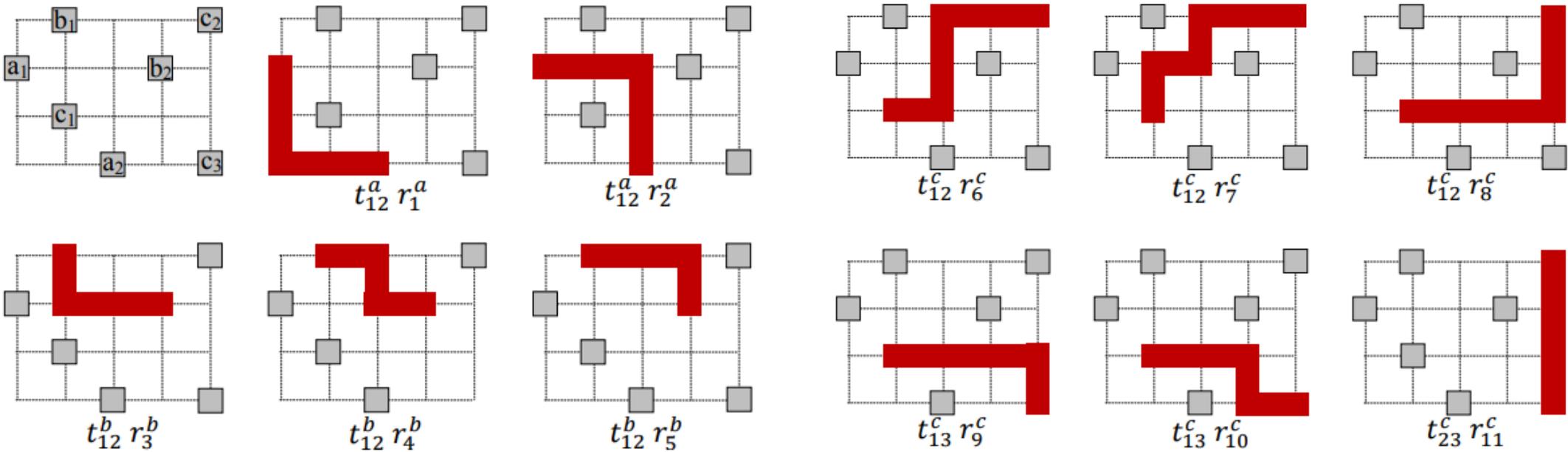
DRE (Design Rule Evaluator) [Gupta, TCAD12]

- Design rule evaluator
 - 'Virtual' cell layouts
 - BEOL layout is not generated
 - Only routing estimation based on single-trunk Steiner tree routing
 - Weak correlation between estimation and result



SAT based Router [Ryzhenko, DAC12]

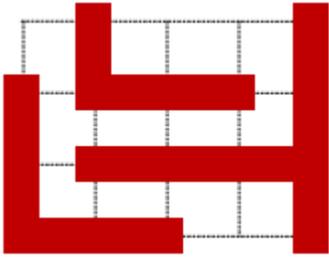
All possible two-pin routes



	t_{12}^a (m)		t_{12}^b (m)			t_{12}^c			t_{13}^c		t_{23}^c
	r_1	r_2	r_3	r_4	r_5	r_6	r_7	r_8	r_9	r_{10}	r_{11}
r_1	■										
r_2		■	x	x		x	x	x	x	x	
r_3		x	■			x	x				
r_4		x		■		x	x				
r_5					■	x	x				
r_6		x	x	x	x	■					
r_7		x	x	x	x		■				
r_8		x						■			
r_9		x							■		
r_{10}		x								■	
r_{11}											■

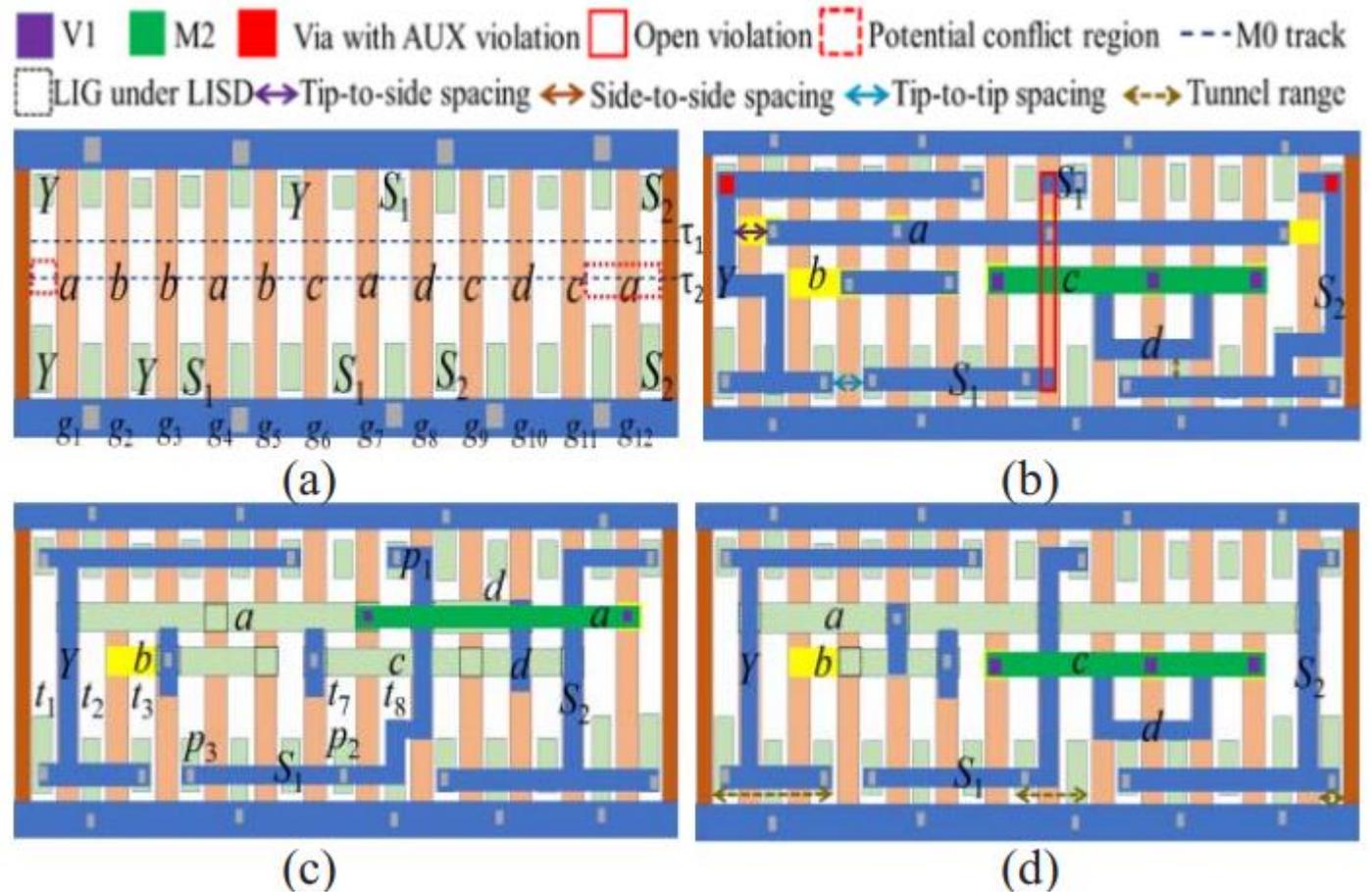
Conflict among the routes

SAT formulation

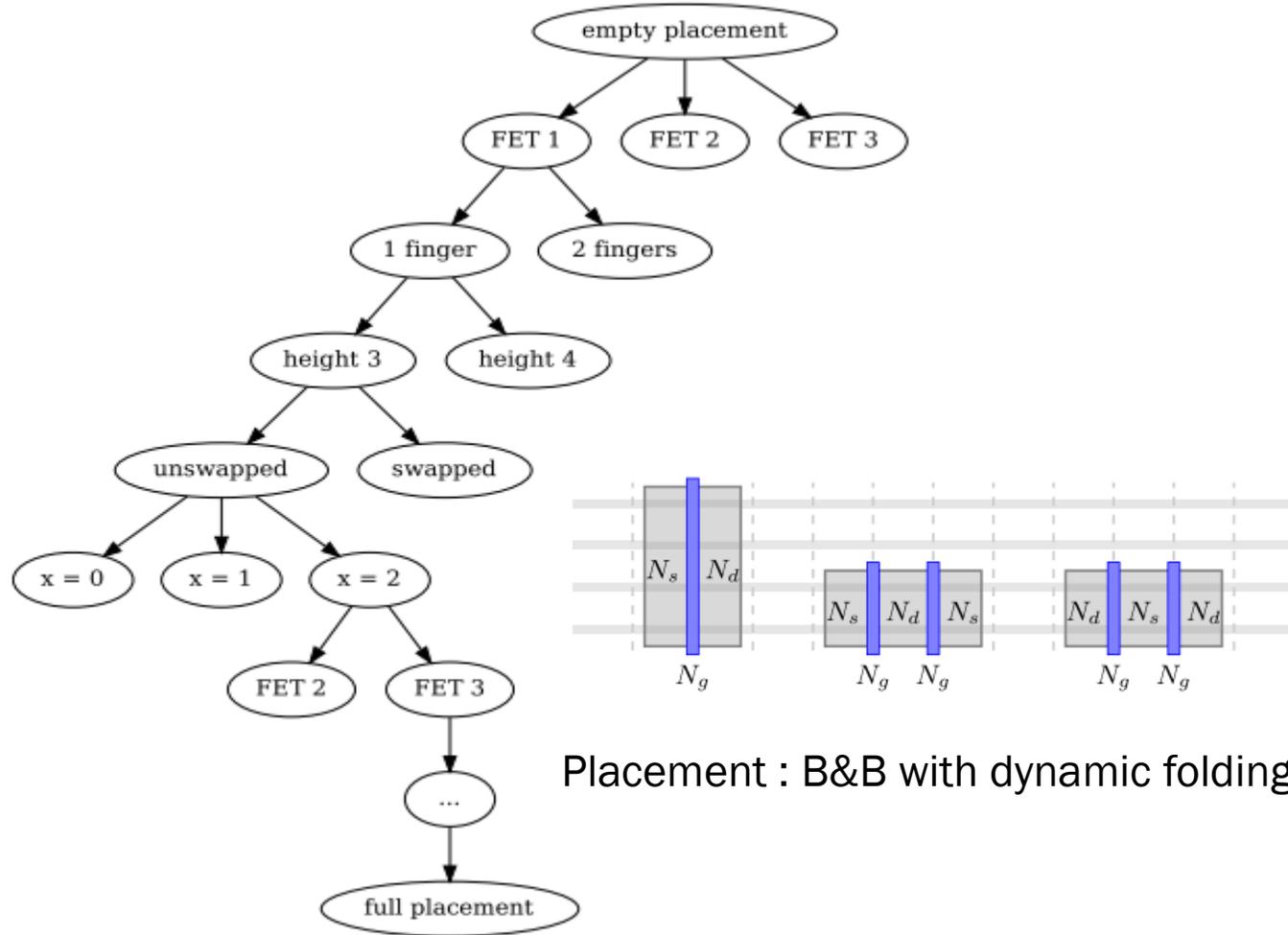


NCTUcell [Li, DAC19]

- Transistor placement
 - Dynamic programming
 - Minimizing a weighed sum of cell area, (estimated) wirelength and routing congestion
 - Transistor folding is not considered.
- In-cell routing
 - Maximize the use of metals (LISD and LIG) in M0 layer
 - ILP formulation



BonnCell [Cleeff, TCAD20]



Metal segment

net

$$\begin{aligned} \text{min} \quad & \sum_{e \in E} c_e x_e \\ \text{s.t.} \quad & x_e = \sum_{k \in \mathcal{N}} x_e^k \quad \forall e \in E \\ & x_e \in \{0, 1\} \quad \forall e \in E \\ & x_e^k \in \{0, 1\} \quad \forall e \in E, k \in \mathcal{N} \\ & b^{ks}(v) = \begin{cases} 1 & \text{if } v = r_k \\ -1 & \text{if } v = s \\ 0 & \text{otherwise} \end{cases} \\ & \quad \forall v \in V, k \in \mathcal{N}, s \in S_k \\ & 0 \leq f_{ij}^{ks} \leq \bar{x}_{ij}^k \quad \forall (i, j) \in A, k \in \mathcal{N}, s \in S_k \\ & \bar{x}_{ij}^k + \bar{x}_{ji}^k \leq x_{\{i, j\}}^k \quad \forall \{i, j\} \in E, k \in \mathcal{N} \\ & x_v^k \in \{0, 1\} \quad \forall v \in V, k \in \mathcal{N} \\ & x_e^k \leq x_v^k \quad \forall v \in e \in E, k \in \mathcal{N} \\ & \sum_{k \in \mathcal{N}} x_v^k \leq 1 \quad \forall v \in V \end{aligned}$$

Routing : MILP

SP&R [Cheng, TCAD20] (1/2)

- Simultaneous transistor placement and in-cell routing
 - SMT formulation -- slow
 - Transistor folding is not considered.

Placement (Cell Size): $\max\{x_t + w_t \mid t \in T\}$

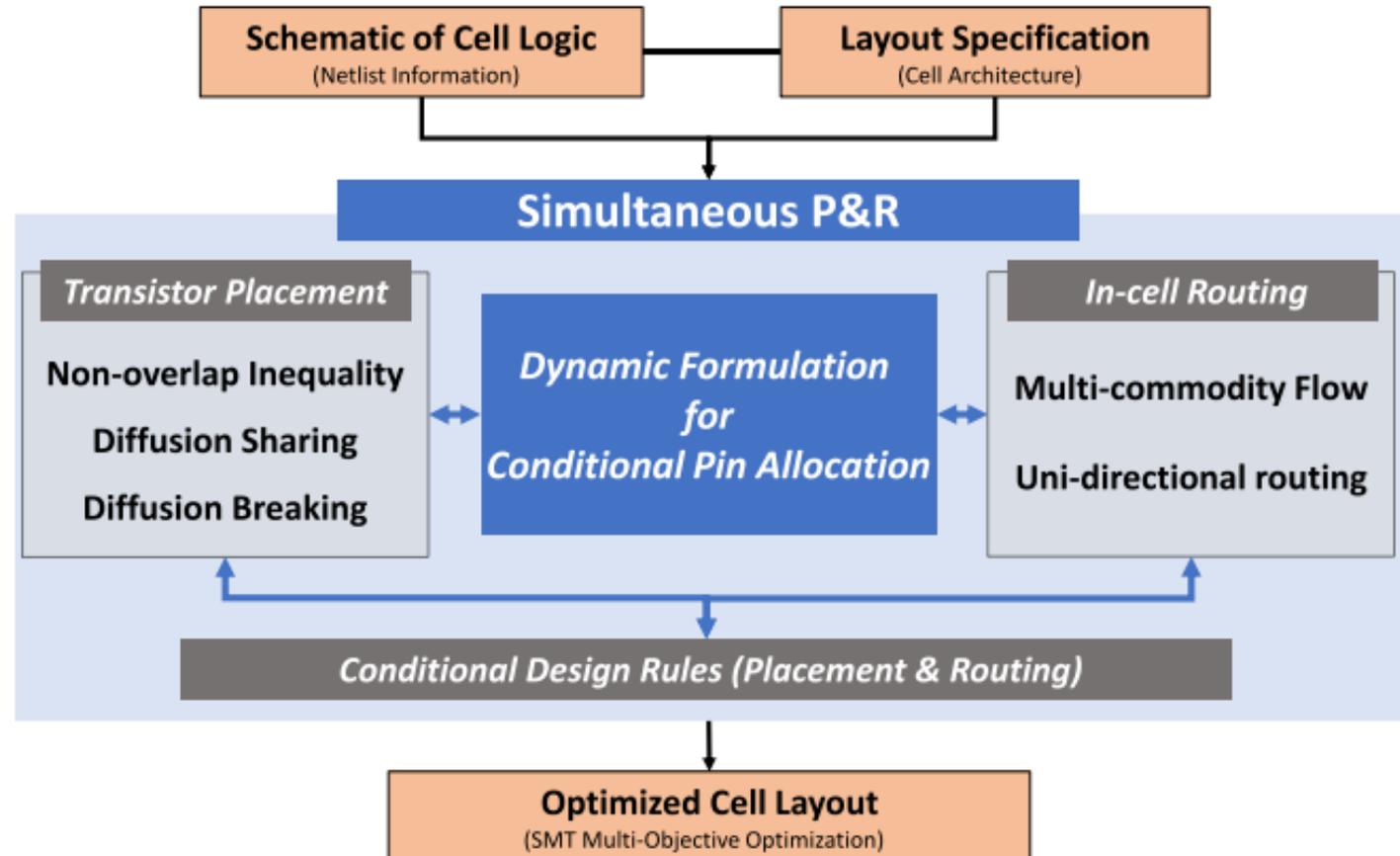
Routability (#M2 Track): $\sum_{k=1}^l \bigvee_{e_{v,u} \in E_k} m_{v,u}$

$l = \# \text{Horizontal Tracks}$

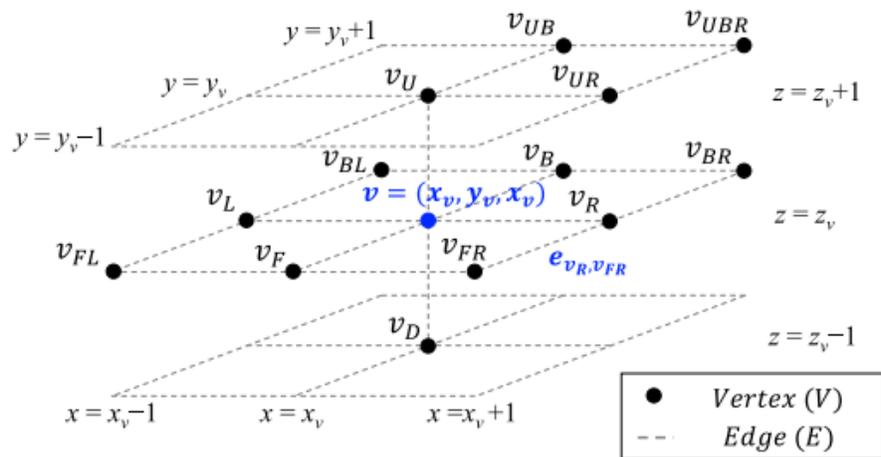
$E_k = \text{Set of M2 Layer Edges in } k\text{th Track}$

Routing (Total ML): $\sum_{e_{v,u} \in E} (w_{v,u} \times m_{v,u})$

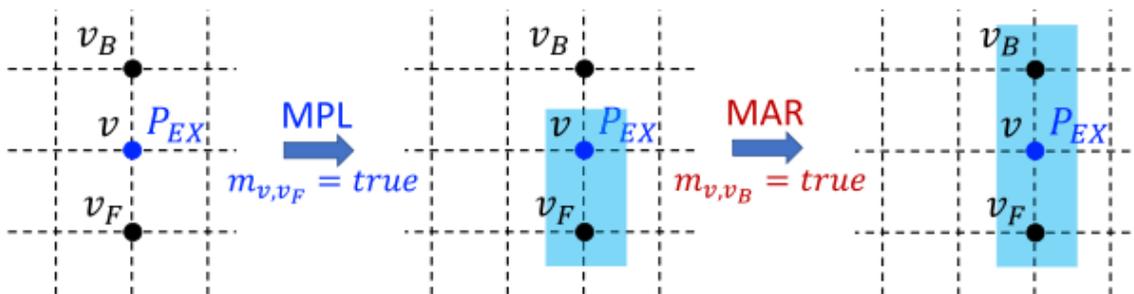
LexMin: (a) Cell Size, (b) #M2 Track, (c) Total ML.



SP&R [Cheng, TCAD20] (2/2)

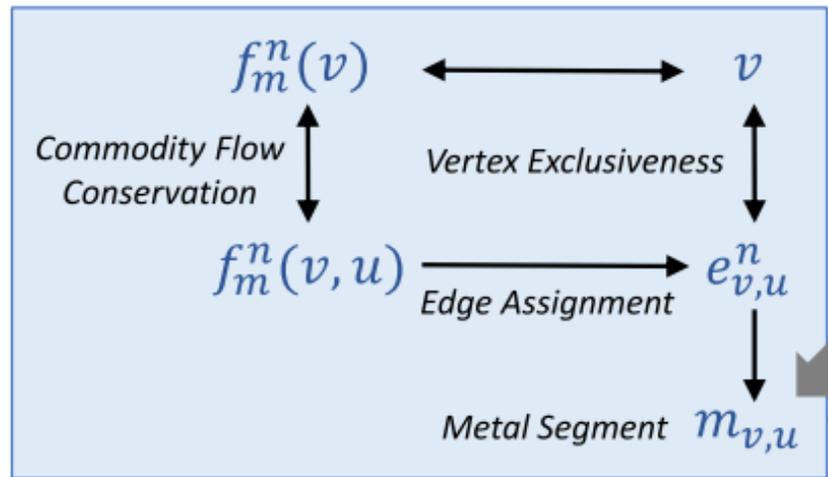


Example of 3-D grid-based routing graph, $G(V, E)$.



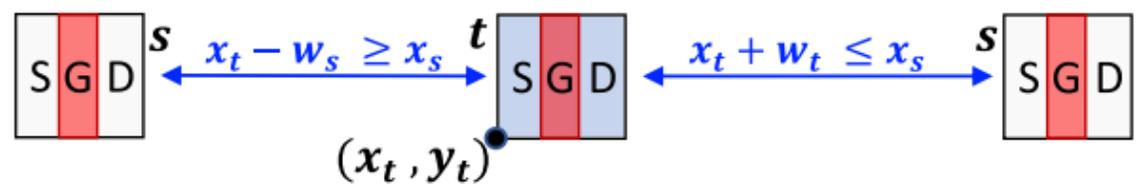
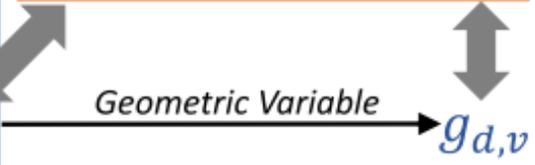
Example of MPL.

Flow Formulation



Conditional Design Rules

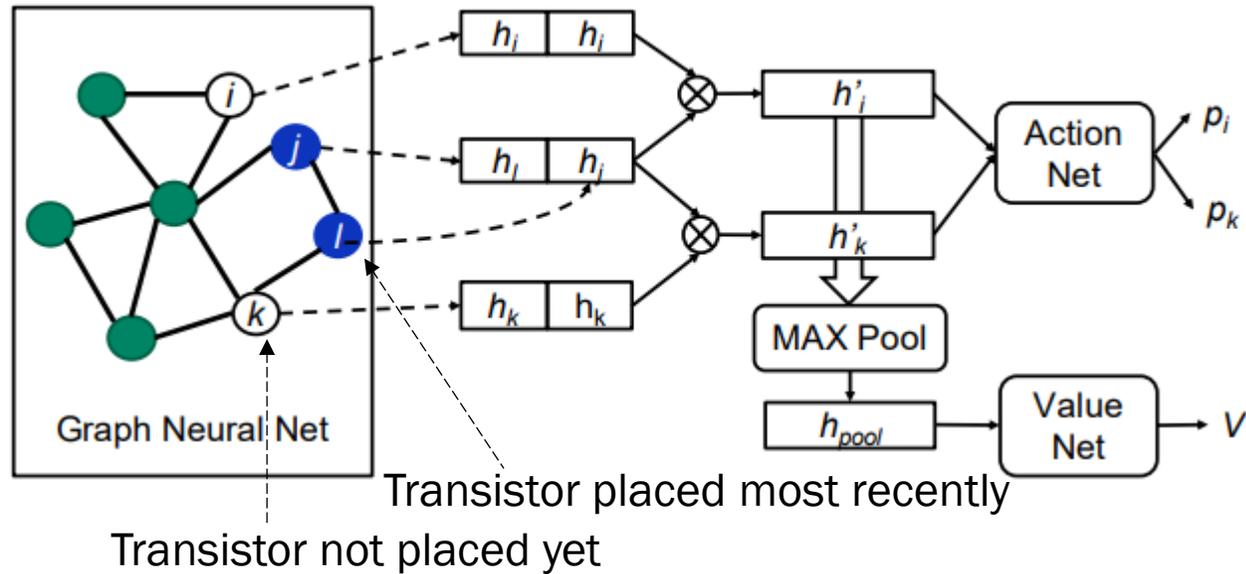
1. End-of-Line Space (EOL)
2. Minimum Area (MAR)
3. Via Rule (VR)
4. Parallel Run Length (PRL)
5. Step Heights (SHR)
6. Minimum Pin Length (MPL)
7. Crosstalk Mitigation (CM)



Relative positions between two FETs.

NVCell [Ren, Invite-DAC21]

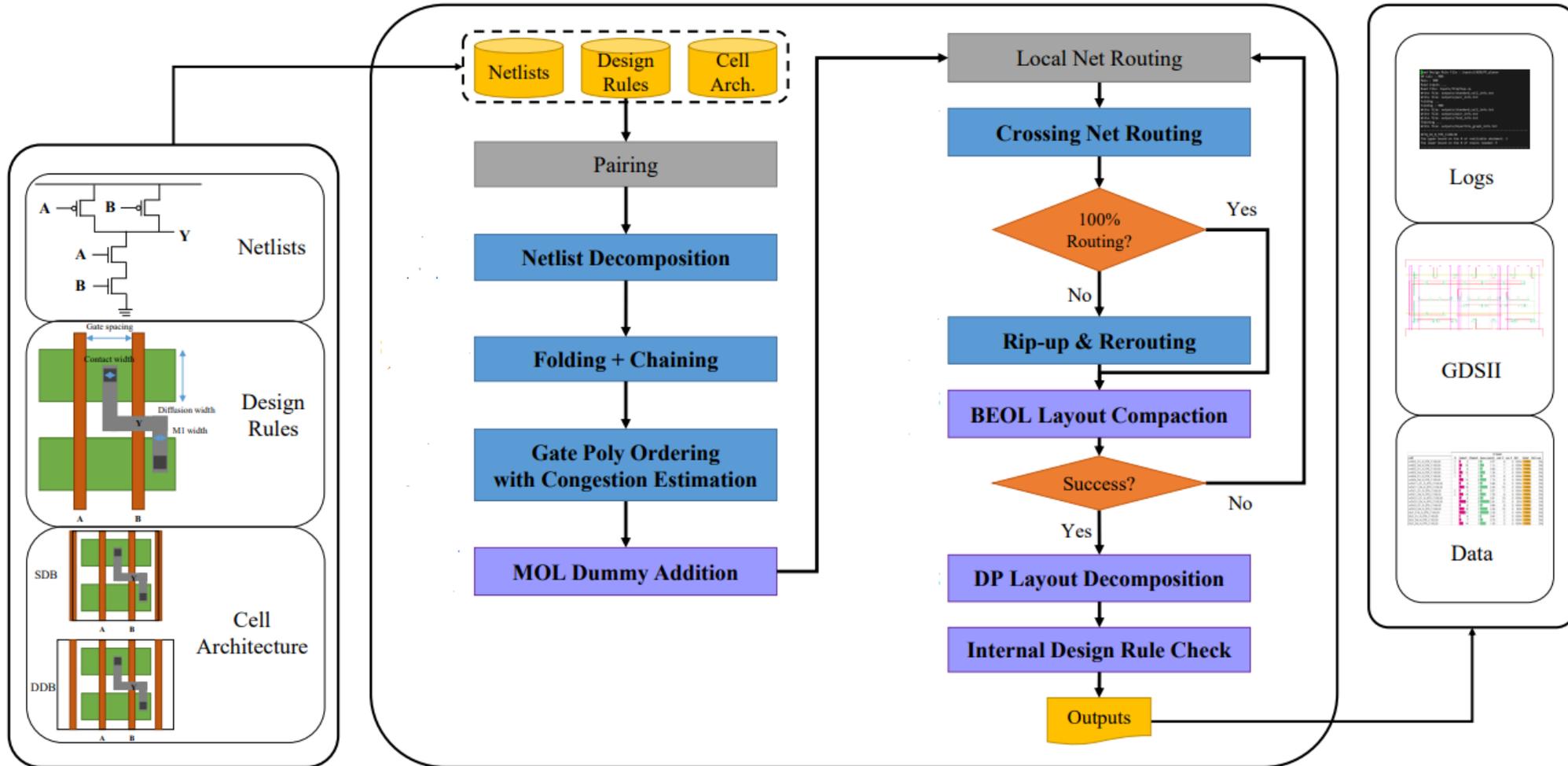
Placement : Simulated Annealing with RL for cost function to move



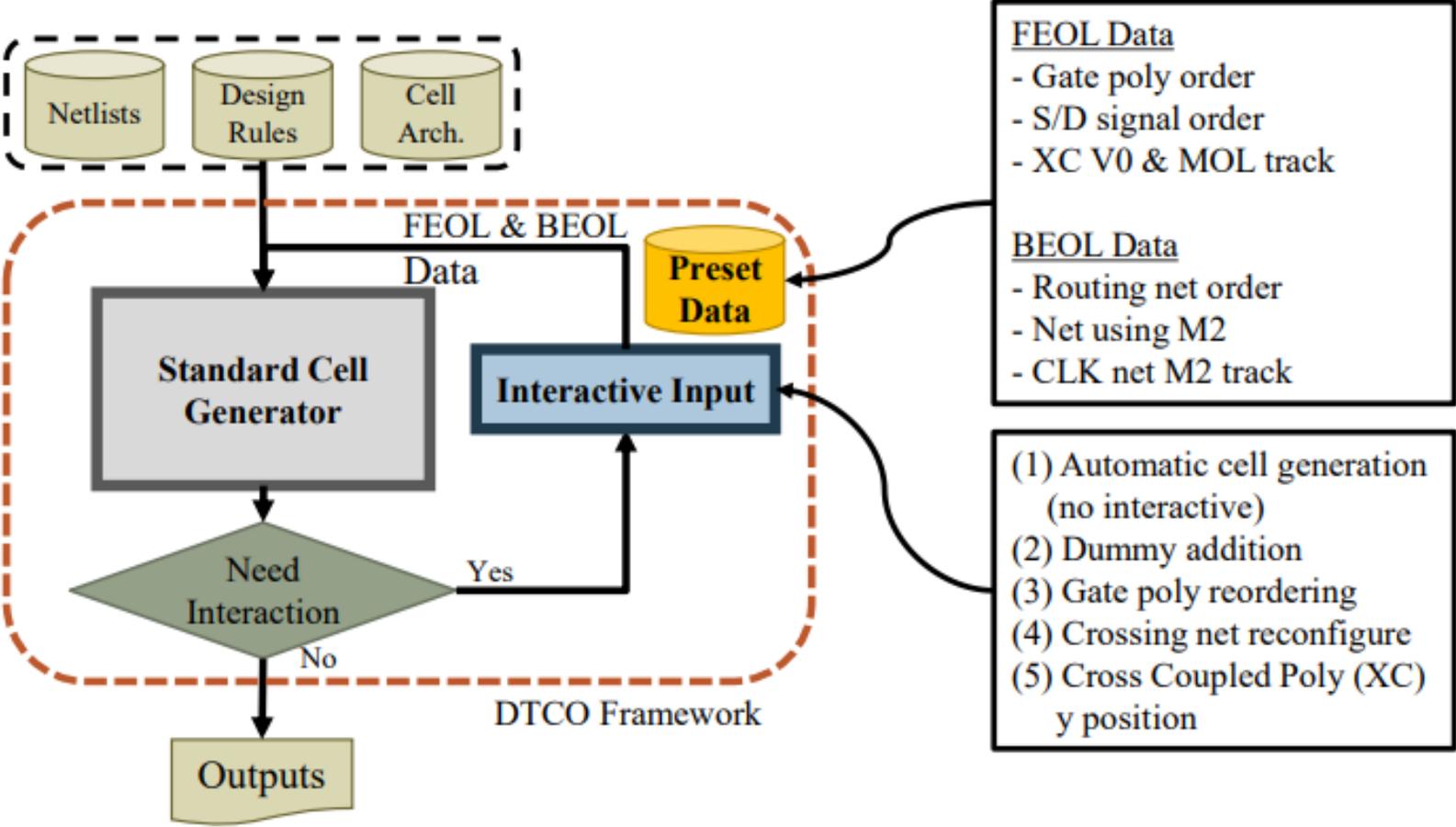
Routing : Genetic Algorithm with each routing solution applying RL for DRC fixing

DP rule only

Example: Auto. Cell Generation [SNU, TVLSI19]

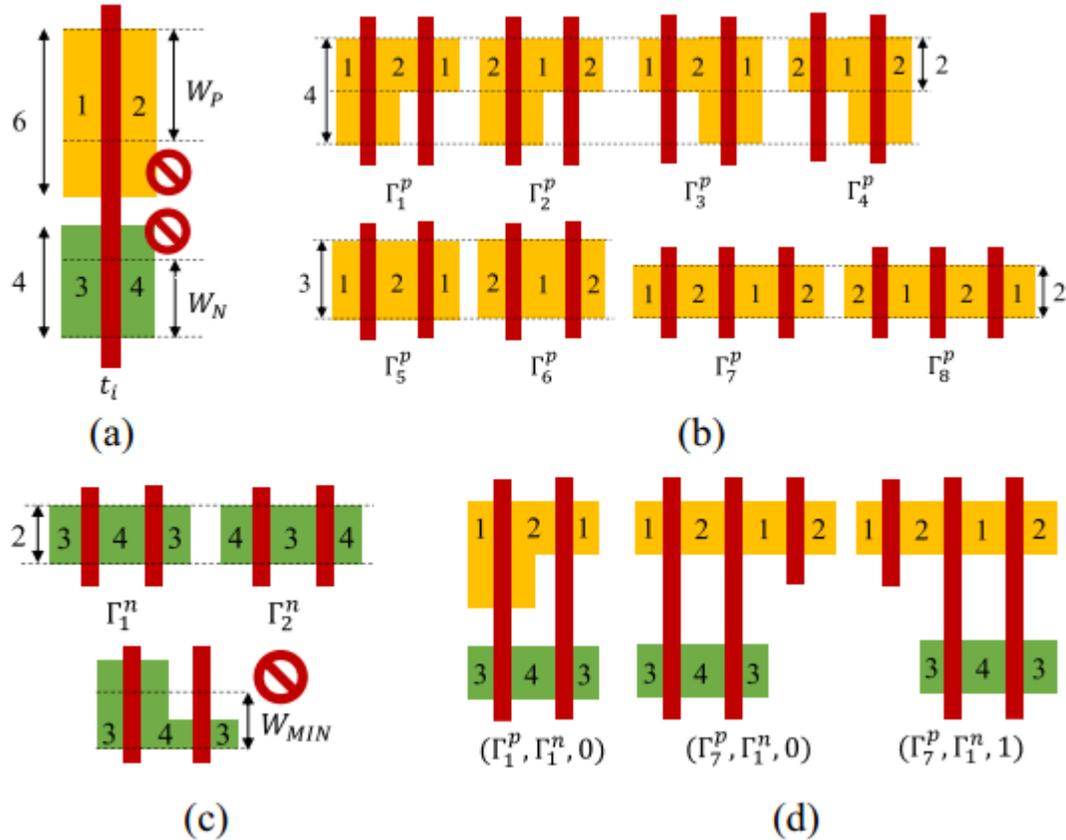
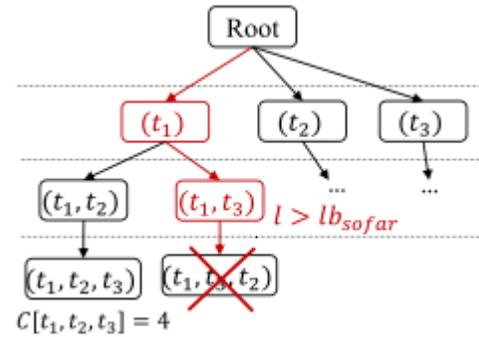


Standard Cell Exploration [SNU, TVLSI19]

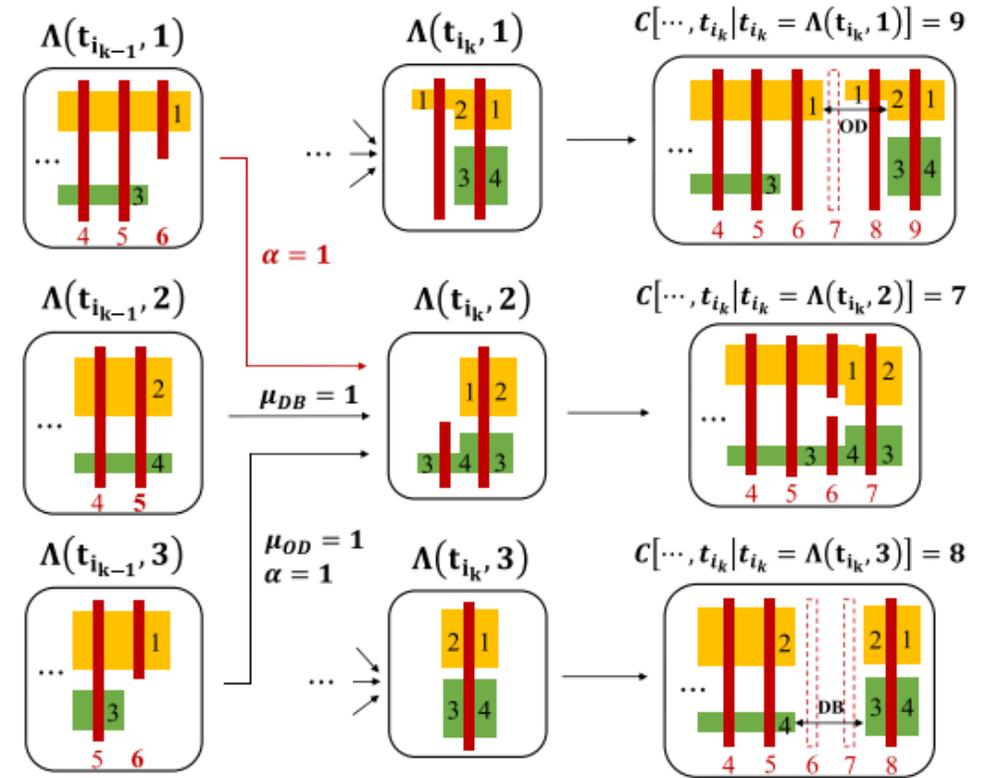


Csyn-fp [Baek, TCAD23]

Folding + Placement : DP (optimal)



Generating all folding shapes

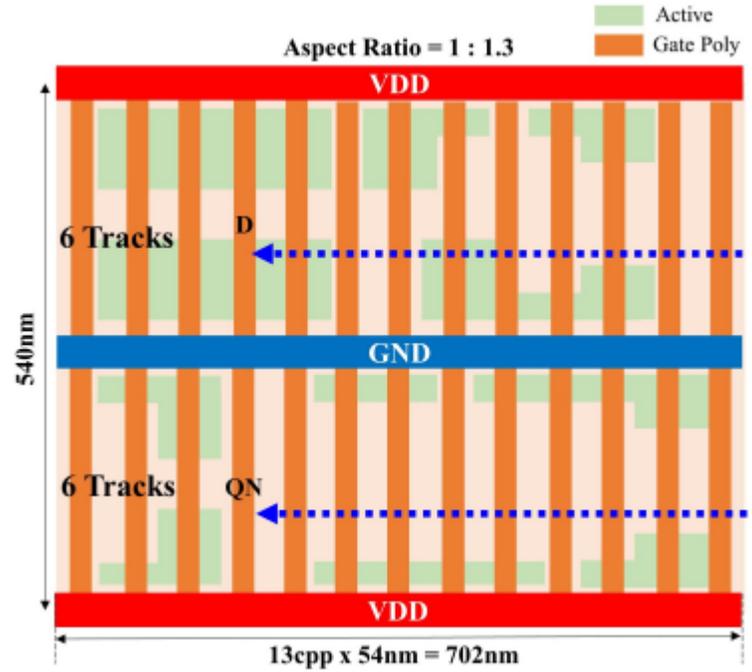
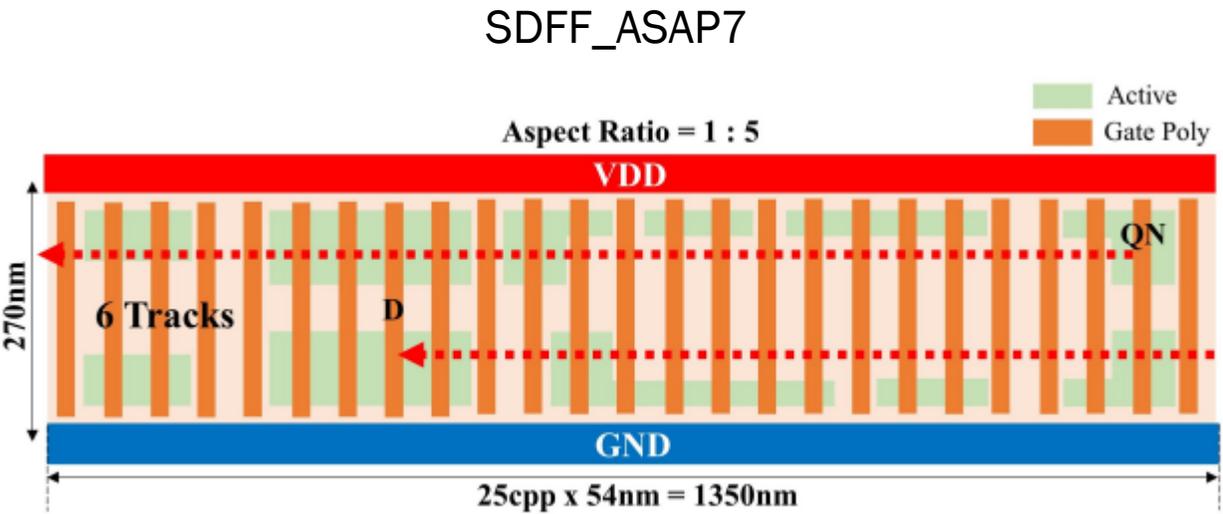


Placement based on DP

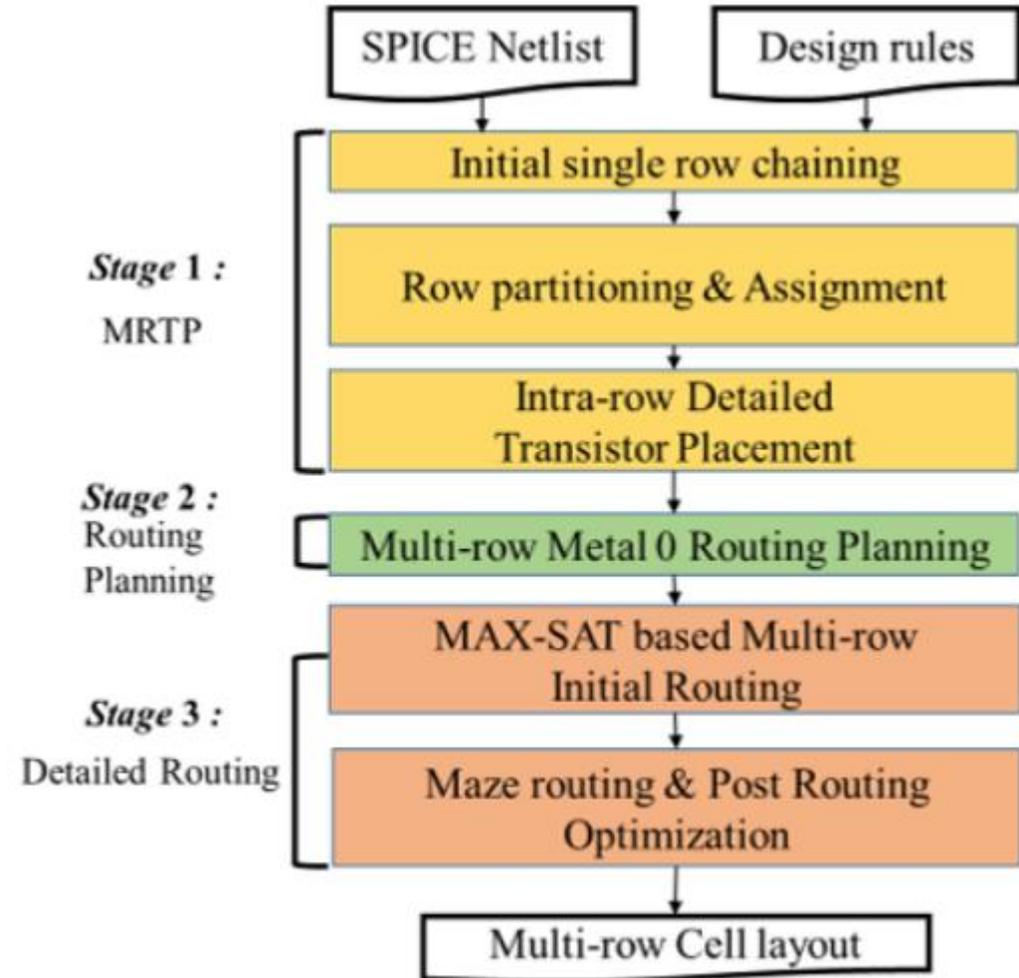
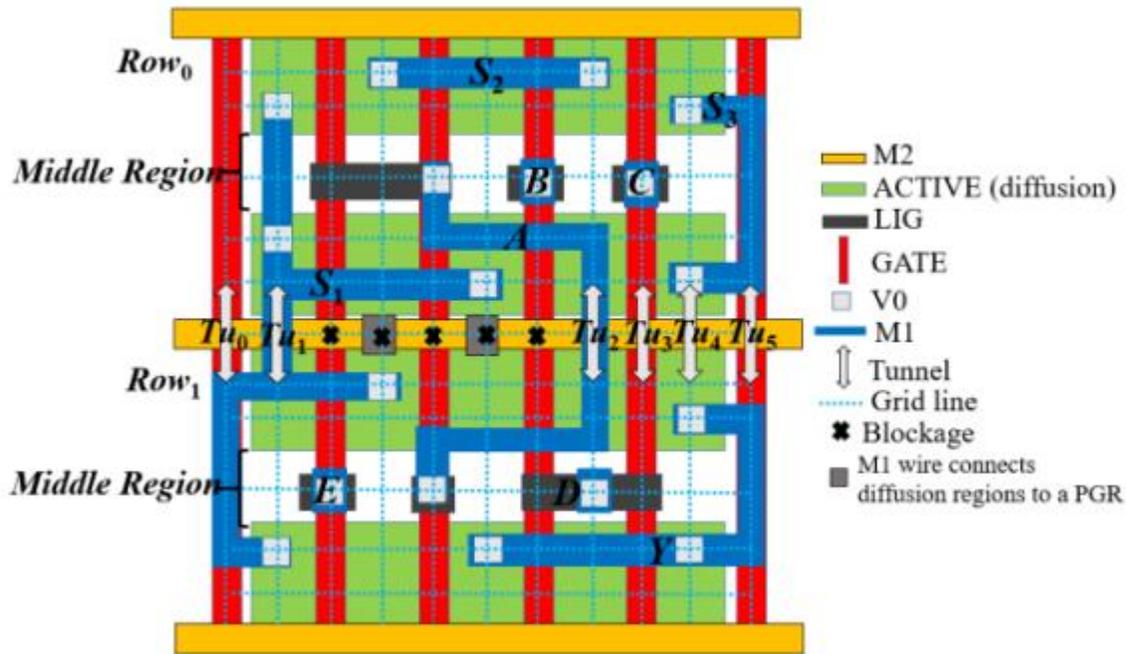
Content

1. Auto-generation of standard cells
 - Design and technology co-optimization (DTCO)
 - Problems
 - Algorithms
 - Multi-row cells
 - Placement legalization
 2. Multi-bit flip-flop (MBFF) cells
 - Structure
 - DTCO flow with MBFF cells
 - DTCO techniques with MBFF cells
 3. Complementary FET (CFET) cells
 - FEOL and BEOL
 - Backside interconnect
 - CFET vs. Flip-FET (FFET)
 4. Conclusion
-

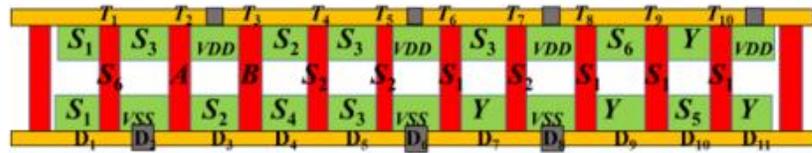
Single-row Height vs. Multi-row Height [Kim, GLSVLSI23]



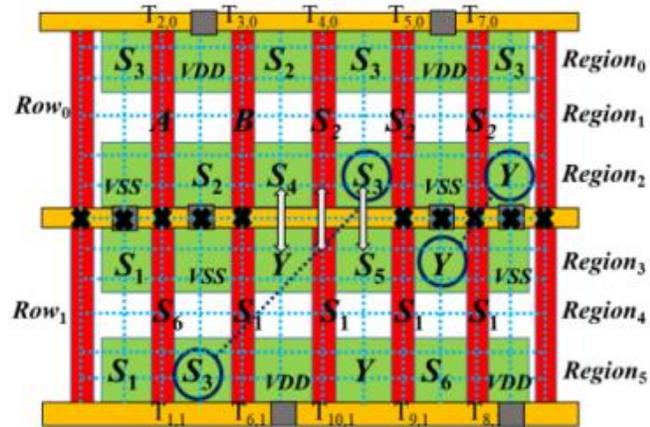
Multi-row-Cell [Li, ICCAD20] (1/2)



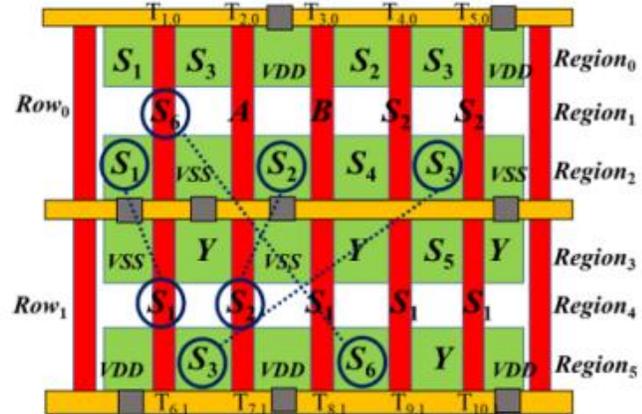
Multi-row-Cell [Li, ICCAD20] (2/2)



(a)

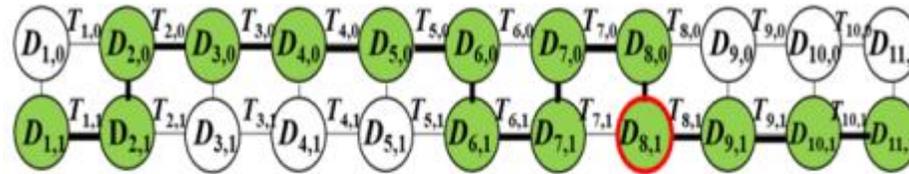


(b)

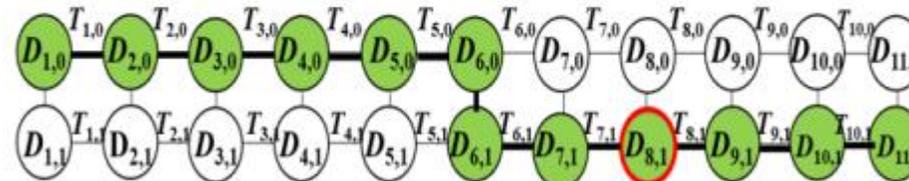


(c)

- Want to obtain a multi-row transistor placement by producing an optimal a single-row transistor placement and then folding it into two or several rows
- A* search algorithm with cost function considering area and net length estimates
- Max-SAT for in-cell routing

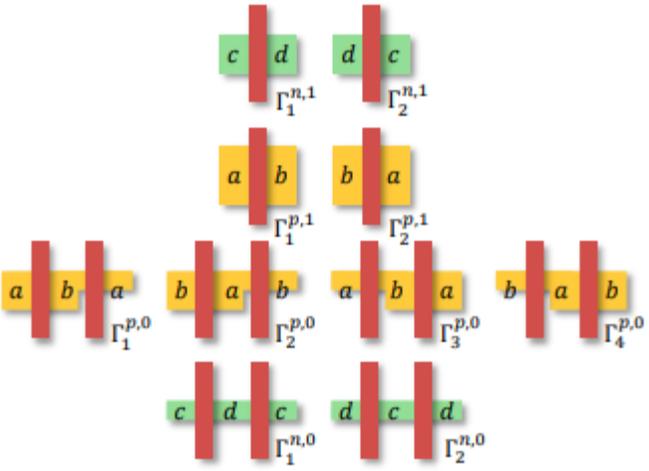
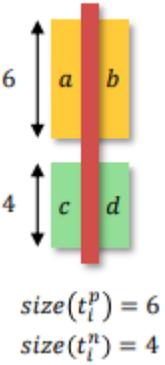


(d)

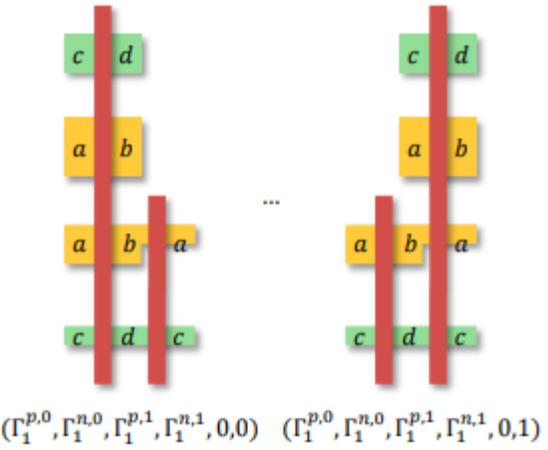


(e)

Tr-level MCell [Chung, ICCAD24] (1/3)

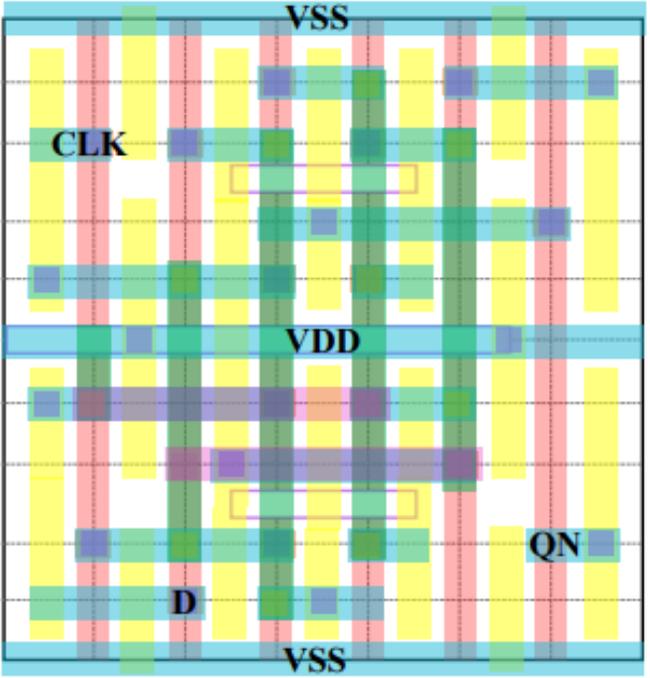
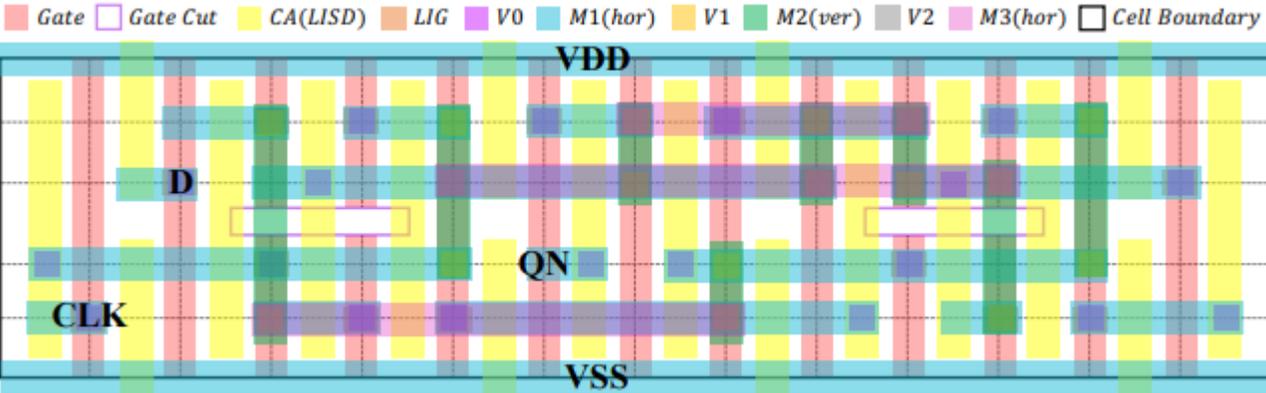


all folding shapes of each of n/p-MOS transistors



all folding shapes of each of n/p-MOS transistor pairs

Tr.-level Multi-row-Cell [Chung, ICCAD24] (2/3)

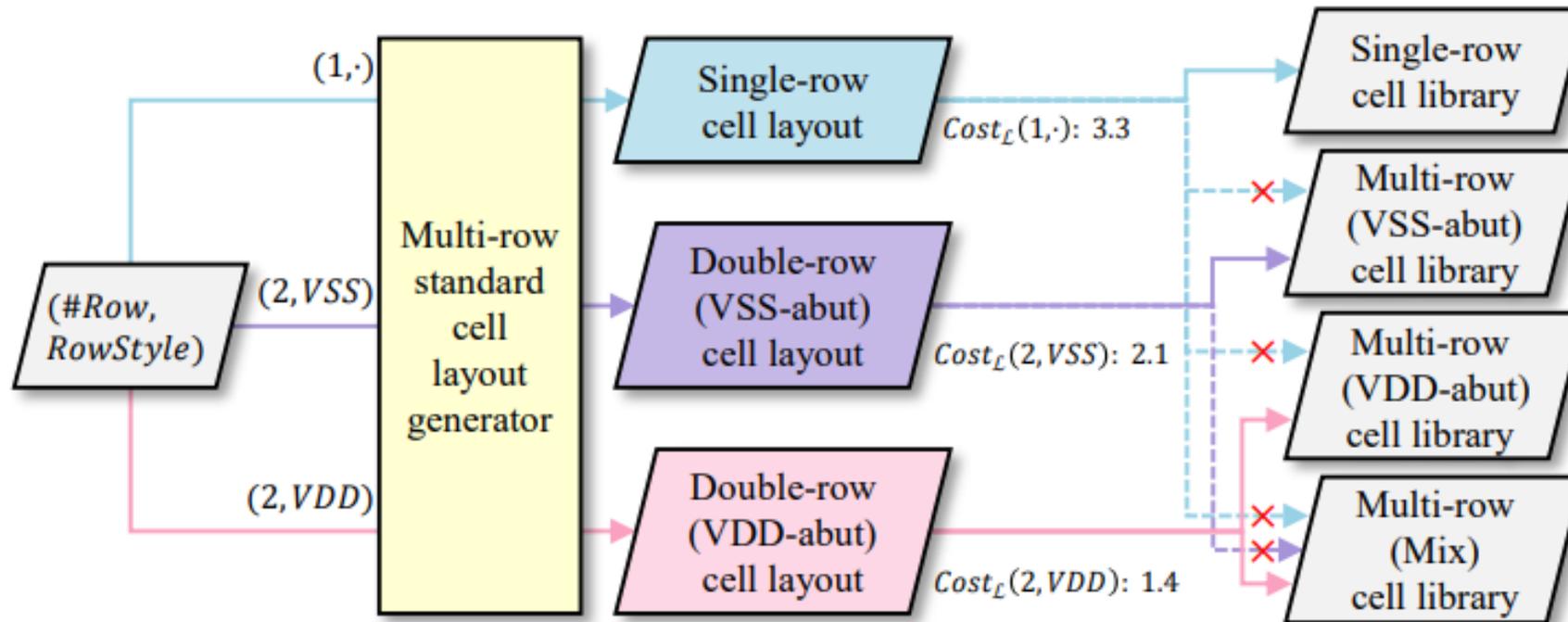


Layouts of DFFHQ Nx1

	Single-row	Double-row
Area	14	14
Aspect ratio (#Row/#CPP)	$\frac{1}{14}$	$\frac{2}{7}$
RPA(D)	1	2
RPA(CLK)	1	1
RPA(QN)	1	1
#M3T	3	2
#M2T	8	5

multi-row standard cell library generation [Chung, ICCAD24] (3/3)

$$Cost_{\mathcal{L}} = \underset{\text{Cell area}}{Area} + \underset{\text{Aspect ratio}}{|AR - 1|} - \sum_{p_i \in P} \overset{\text{Remaining pin access}}{\min(RPA(p_i), \beta)} + \underset{\text{total number of M3 and M2 tracks used}}{\#MT}$$



Placement Legalization [Kim, GLSVLSI23]

Circuits	Util.	Displacement		HPWL	
		SC-only	SC+DC	SC-only	SC+DC
des_perf_1	0.98	227,550	681,815	1,281,320	2,022,800
fft_1	0.84	62,258	192,497	283,184	532,551
fft_2	0.50	33,613	38,008	277,016	286,495
mat_mult_1	0.80	199,207	242,767	1,504,430	1,572,490
fft_a	0.25	28,568	24,057	661,846	687,357
fft_b	0.28	19,714	21,118	694,253	698,159
edit_dt_a	0.46	89,777	94,949	4,224,540	4,273,450
ratio (avg.)		1	1.62	1	1.26

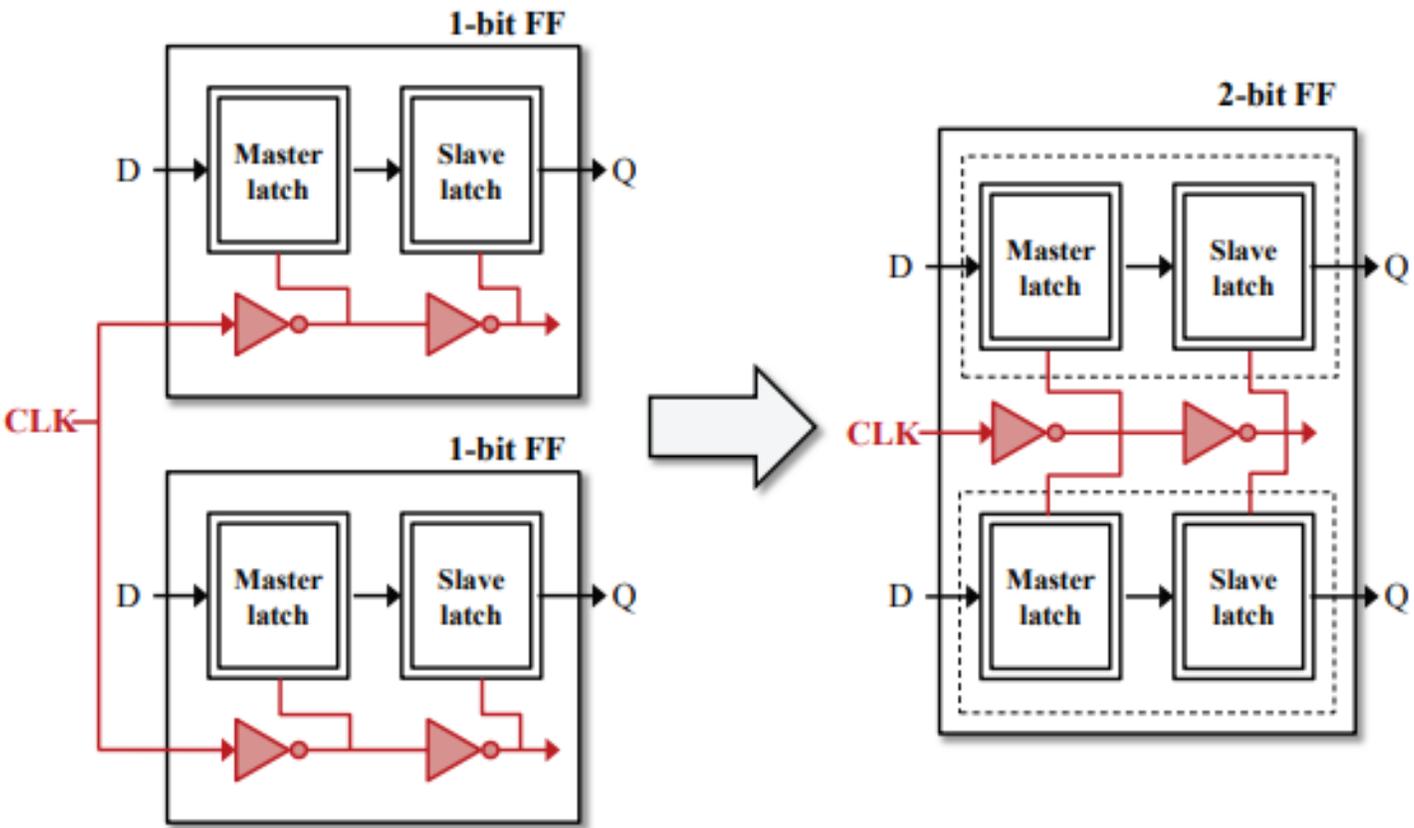
SC-only: Placement legalization for designs with single-row height cells

SC+DC : Placement legalization for designs with both of single-row height cells (85%~95%) and double-row height cells (5%~15%)

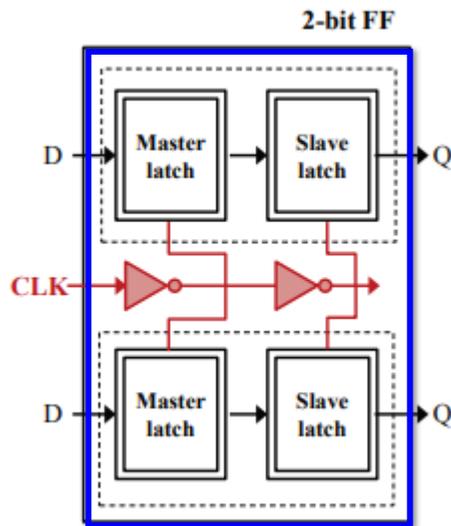
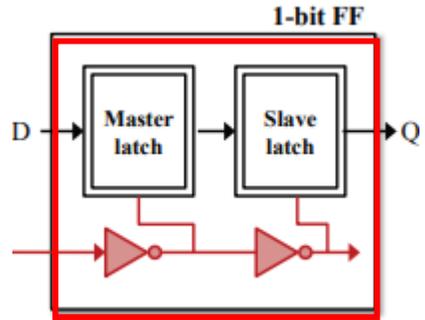
Content

1. Auto-generation of standard cells
 - Design and technology co-optimization (DTCO)
 - Problems
 - Algorithms
 - Multi-row cells
 - Placement legalization
 2. Multi-bit flip-flop (MBFF) cells
 - Structure
 - DTCO flow with MBFF cells
 - DTCO techniques with MBFF cells
 3. Complementary FET (CFET) cells
 - Planar FET vs. CFET
 - Algorithms
 - Flip-FET (FFET) cells
 4. Conclusion
-

Multi-bit Flip-flop: Structure



Multi-bit Flip-flop: Delay and Power [Yang, SOCC23]

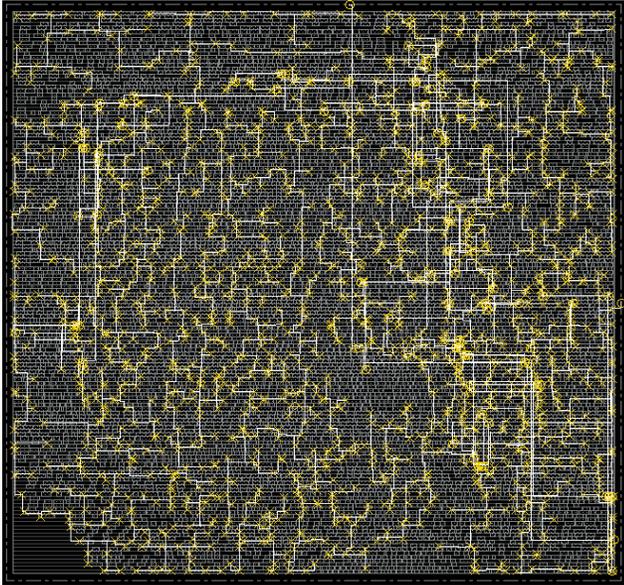


Cell-level Delay / Power	n -bit Flip-flop			
	1-bit FF ($n = 1$)	2-bit FF ($n = 2$)	3-bit FF ($n = 3$)	4-bit FF ($n = 4$)
CK-to-Q delay (ps)	84.6	85.7	85.6	85.8
Dynamic power (μW) ($PWR[n]$) [*]	1.200	1.400	1.800	2.300
Dynamic power per bit (μW) ($PWR[n] / n$)	1.200	0.700	0.600	0.575

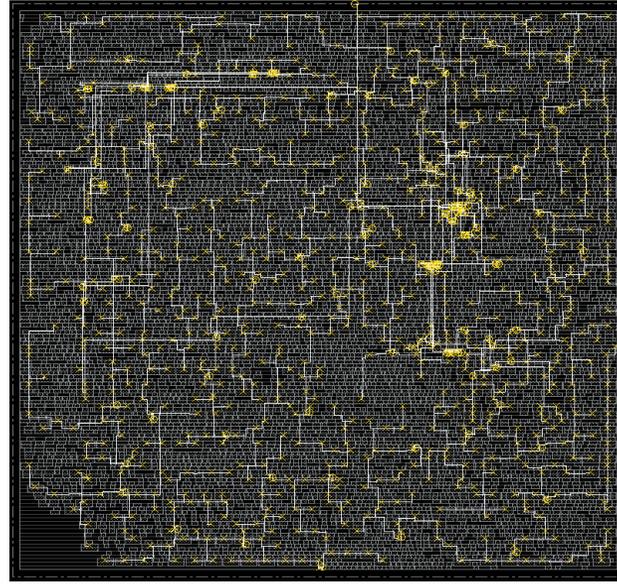
^{*} $PWR[n]$: The amount of dynamic power consumed by an n -bit flip-flop

Multi-bit Flip-flop: Impact on Clock Network

- Reduce the number of clock sinks
 - Reduce the number of clock buffers and wirelength of clock tree
 - Generate simpler clock tree structure



(a) Clock tree using 1-bit FF



(b) Clock tree using 2-bit MBFF

- 40%+ of total power consumed by clock network
 - Reduce clock network power and improve routability

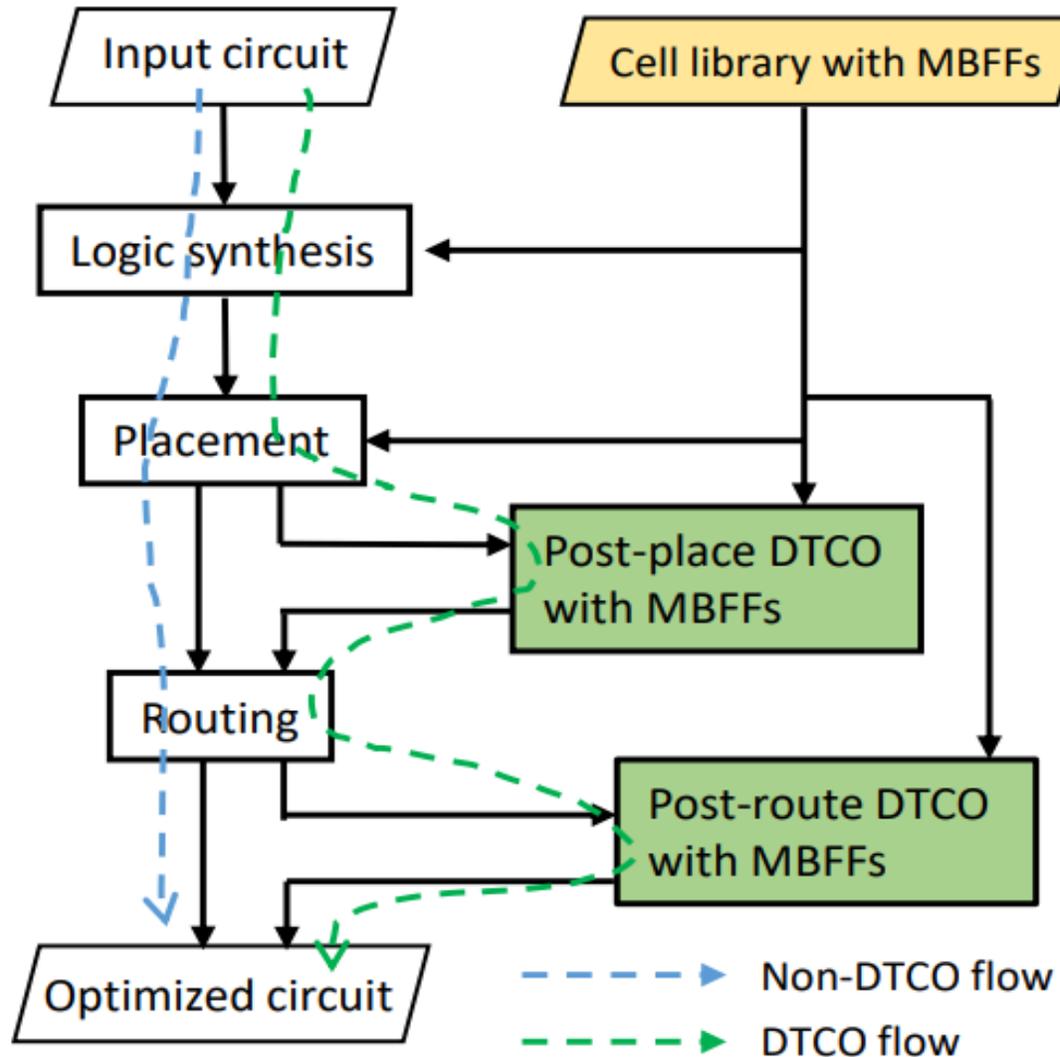
Multi-bit Flip-flop: Impact on Timing and Routing [Kim, ICCAD22]

- MBFF grouping causes inferior circuit timing as well as more routing failures

Circuit	Using no MBFF				Using MBFFs			
	#DRVs	WNS	TNS	Power	#DRVs	WNS	TNS	Power
MEM_CTRL	35	-27.3	-464.0	2862.2	-29%	-44%	-100%	43%
USB_FUNCT	66	-6.5	-13.5	16743.0	-58%	-541%	-3559%	56%
AES_CIPHER	87	0.0	0.0	2944.3	-38%	N/A	N/A	35%
WB_CONMAX	518	-177.6	-3369.8	6715.7	-7%	51%	5%	6%
ETHERNET	427	-63.1	-1210.0	32624.5	30%	-188%	-179%	65%
DES3	44	-9.6	-45.4	52382.5	-93%	-251%	-6130%	51%
NOVA	1974	-39.9	-761.4	13700.6	-21%	-1034%	-1386%	53%
Average	450	-46.3	-837.7	18281.8	-31%	-335%	-1891%	44%

Circuit	MBFFs		Banking ratio
	4-bit	2-bit	
MEM_CTRL	186	9	68%
USB_FUNCT	427	9	99%
AES_CIPHER	128	7	99%
WB_CONMAX	44	5	23%
ETHERNET	2296	42	88%
DES3	2170	44	100%
NOVA	6051	112	84%

DTCO Flow with MBFF Cells [Kim, MWSCAS24]



Objectives

- Minimizing wirelength
- Optimizing Timing
- Trading power against timing

Content

1. Auto-generation of standard cells

- Design and technology co-optimization (DTCO)
- Problems
- Algorithms
- Multi-row cells
- Placement legalization

2. Multi-bit flip-flop (MBFF) cells

- Structure
- DTCO flow with MBFF cells
- DTCO techniques with MBFF cells
 - Post-placement DTCO techniques
 - Post-route DTCO techniques
 - Debanking technique

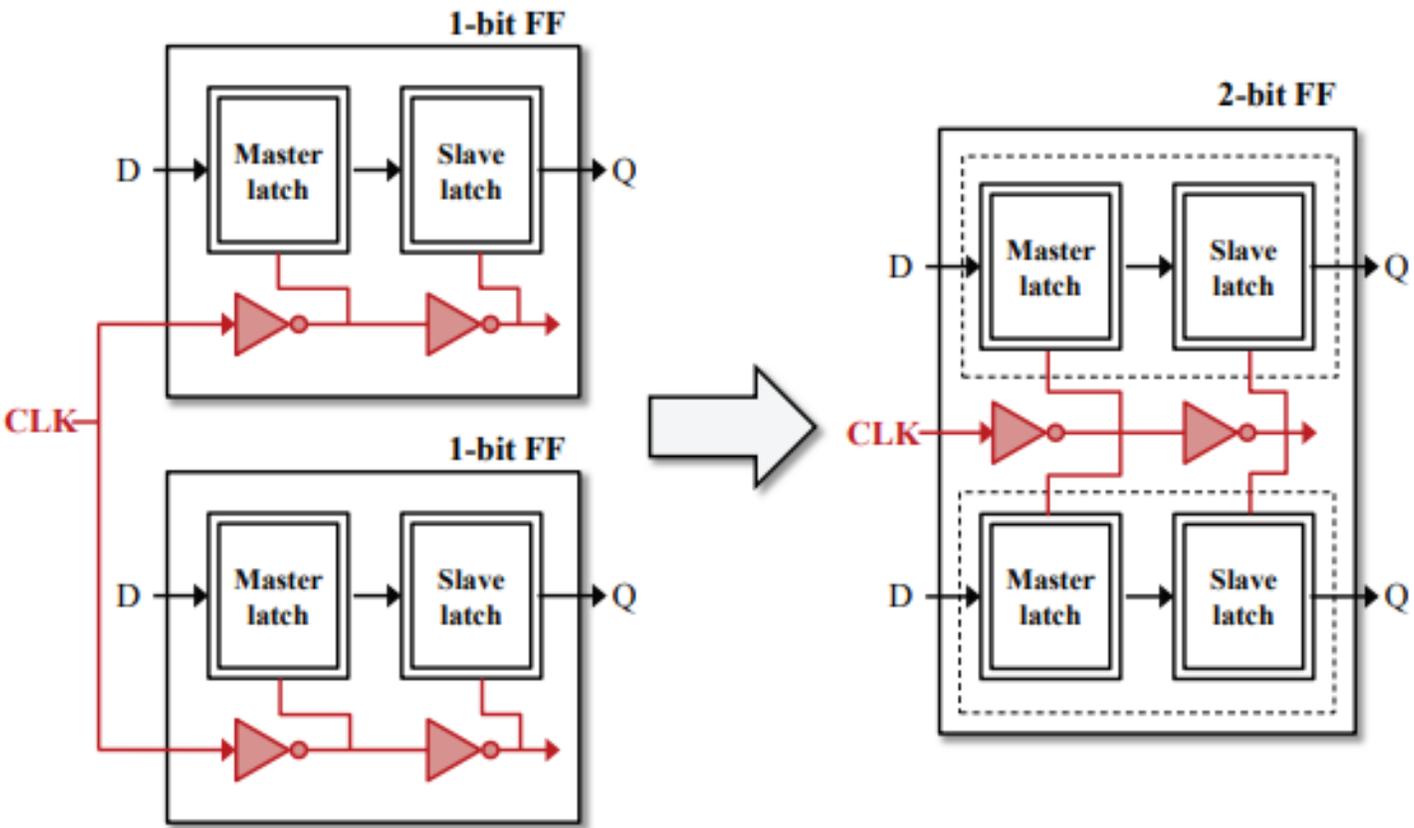
3. Complementary FET (CFET) cells

- FEOL and BEOL
- Backside interconnect
- CFET vs. Flip-FET (FFET)

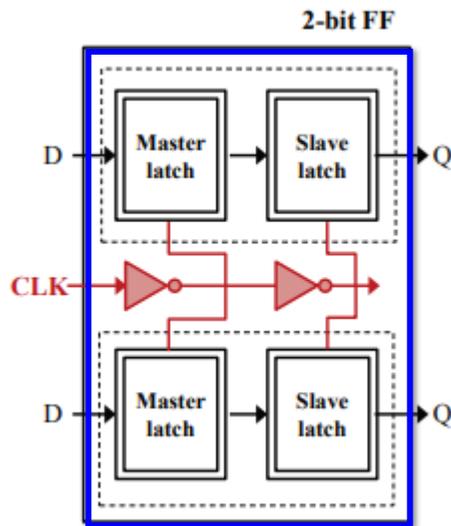
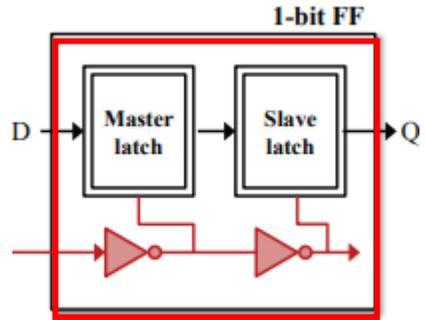
4. Conclusion



Multi-bit Flip-flop: Structure



Multi-bit Flip-flop: Delay and Power [Yang, SOCC23]

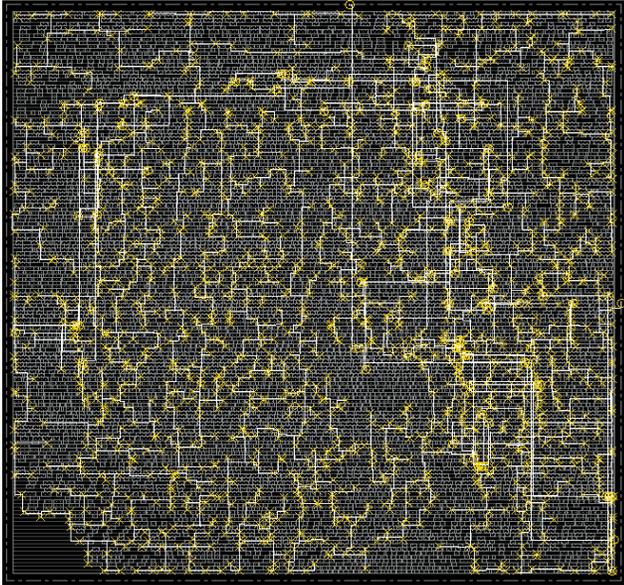


Cell-level Delay / Power	<i>n</i> -bit Flip-flop			
	1-bit FF (<i>n</i> = 1)	2-bit FF (<i>n</i> = 2)	3-bit FF (<i>n</i> = 3)	4-bit FF (<i>n</i> = 4)
CK-to-Q delay (<i>ps</i>)	84.6	85.7	85.6	85.8
Dynamic power (μW) ($PWR[n]$) [*]	1.200	1.400	1.800	2.300
Dynamic power per bit (μW) ($PWR[n] / n$)	1.200	0.700	0.600	0.575

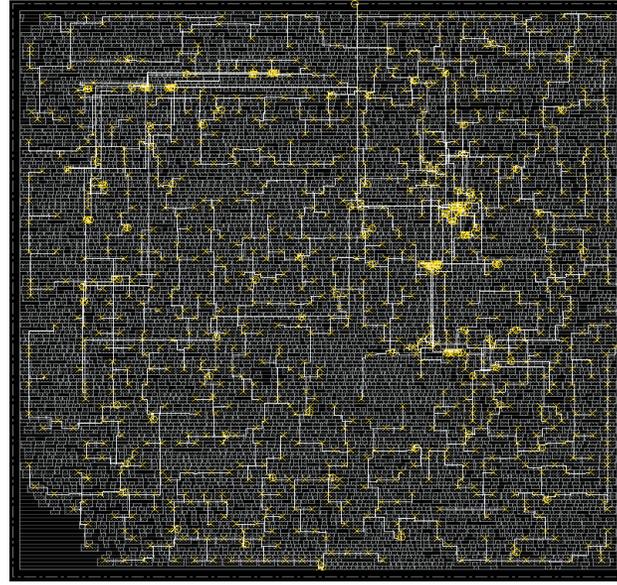
^{*} $PWR[n]$: The amount of dynamic power consumed by an *n*-bit flip-flop

Multi-bit Flip-flop: Impact on Clock Network

- Reduce the number of clock sinks
 - Reduce the number of clock buffers and wirelength of clock tree
 - Generate simpler clock tree structure



(a) Clock tree using 1-bit FF



(b) Clock tree using 2-bit MBFF

- 40%+ of total power consumed by clock network
 - Reduce clock network power and improve routability

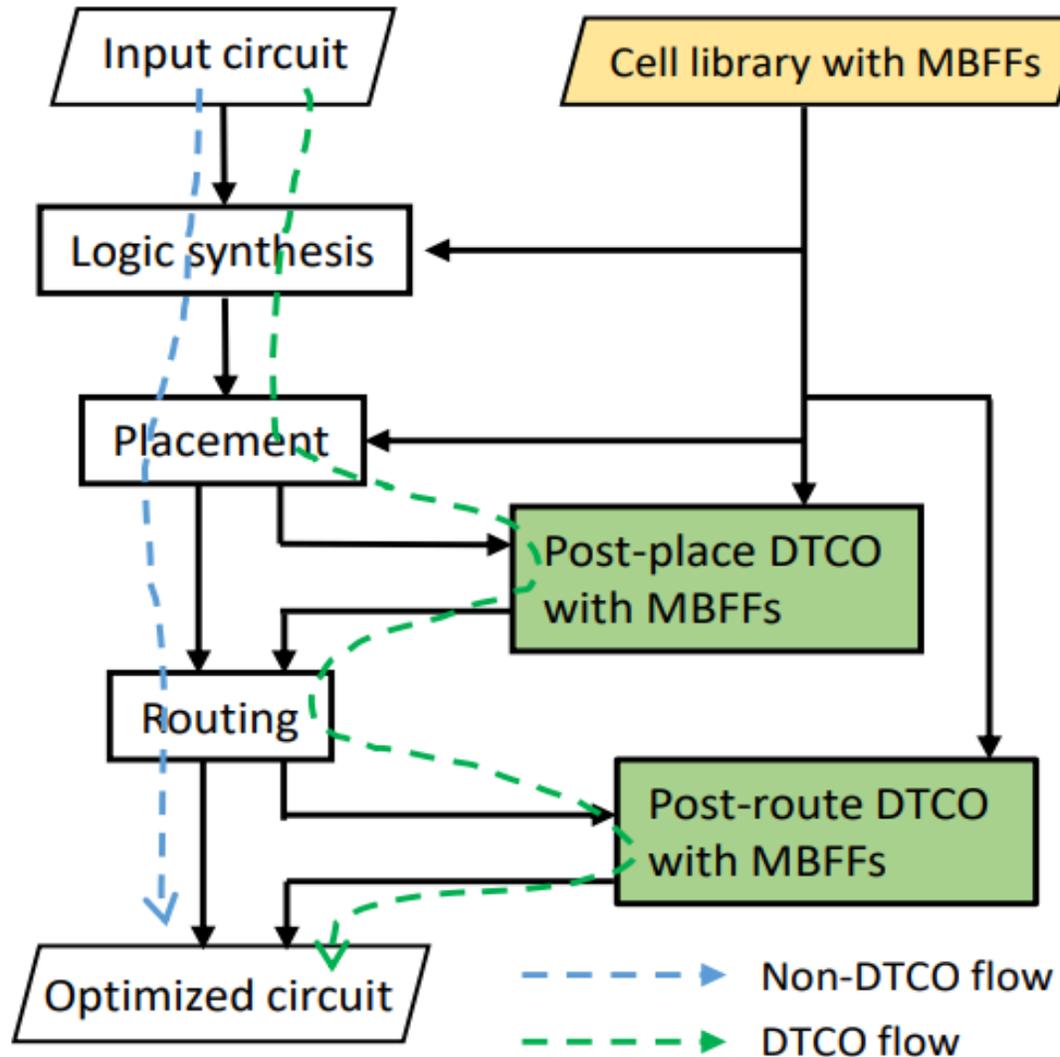
Multi-bit Flip-flop: Impact on Timing and Routing [Kim, ICCAD22]

- MBFF grouping causes inferior circuit timing as well as more routing failures

Circuit	Using no MBFF				Using MBFFs			
	#DRVs	WNS	TNS	Power	#DRVs	WNS	TNS	Power
MEM_CTRL	35	-27.3	-464.0	2862.2	-29%	-44%	-100%	43%
USB_FUNCT	66	-6.5	-13.5	16743.0	-58%	-541%	-3559%	56%
AES_CIPHER	87	0.0	0.0	2944.3	-38%	N/A	N/A	35%
WB_CONMAX	518	-177.6	-3369.8	6715.7	-7%	51%	5%	6%
ETHERNET	427	-63.1	-1210.0	32624.5	30%	-188%	-179%	65%
DES3	44	-9.6	-45.4	52382.5	-93%	-251%	-6130%	51%
NOVA	1974	-39.9	-761.4	13700.6	-21%	-1034%	-1386%	53%
Average	450	-46.3	-837.7	18281.8	-31%	-335%	-1891%	44%

Circuit	MBFFs		Banking ratio
	4-bit	2-bit	
MEM_CTRL	186	9	68%
USB_FUNCT	427	9	99%
AES_CIPHER	128	7	99%
WB_CONMAX	44	5	23%
ETHERNET	2296	42	88%
DES3	2170	44	100%
NOVA	6051	112	84%

DTCO Flow with MBFF Cells [Kim, MWSCAS24]



Objectives

- Minimizing wirelength
- Optimizing Timing
- Trading power against timing

Content

1. Auto-generation of standard cells

- Design and technology co-optimization (DTCO)
- Problems
- Algorithms
- Multi-row cells
- Placement legalization

2. Multi-bit flip-flop (MBFF) cells

- Structure
- DTCO flow with MBFF cells
- DTCO techniques with MBFF cells
 - Post-placement DTCO techniques
 - Post-route DTCO techniques
 - Debanking technique

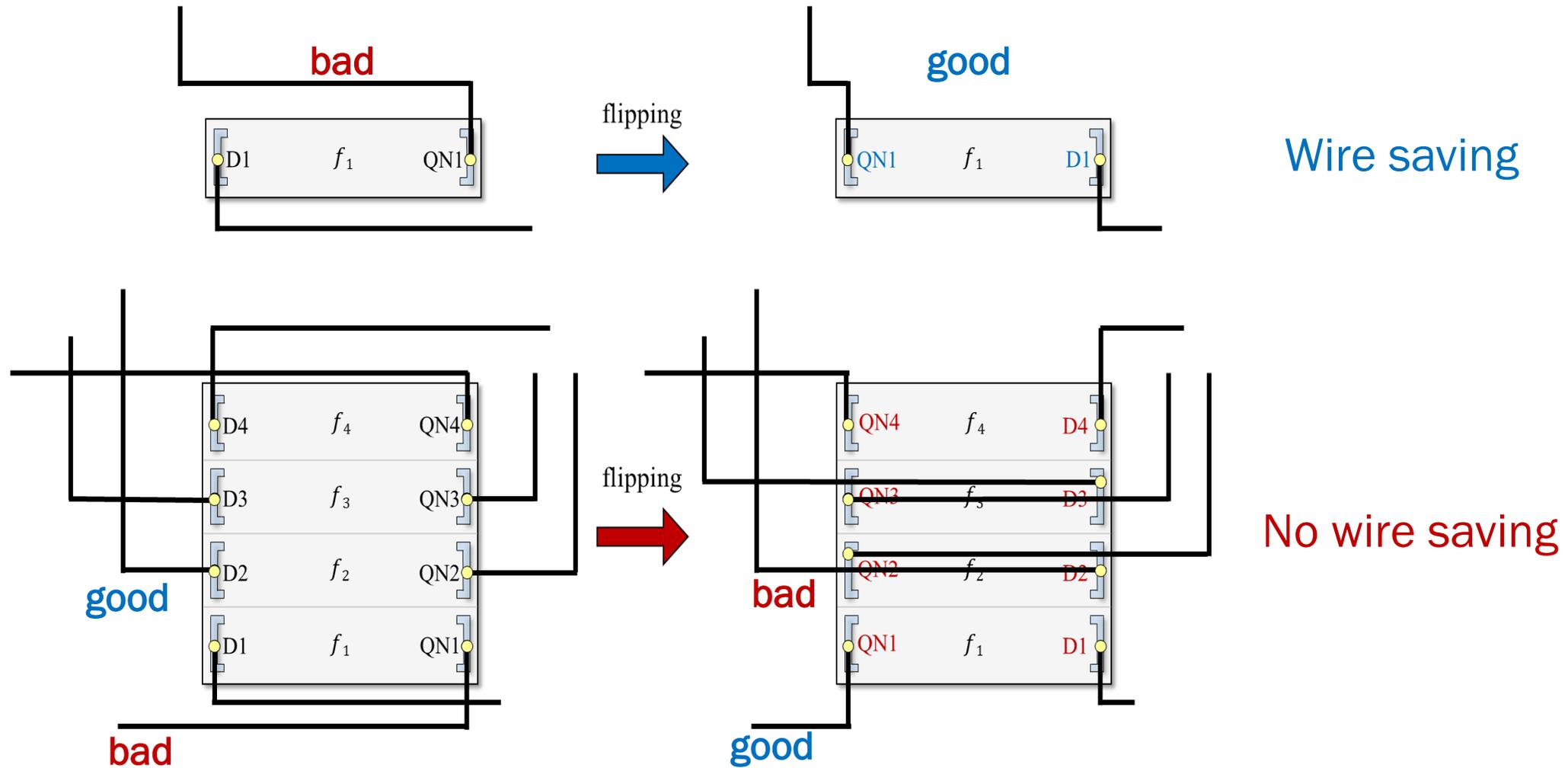
3. Complementary FET (CFET) cells

- FEOL and BEOL
- Backside interconnect
- CFET vs. Flip-FET (FFET)

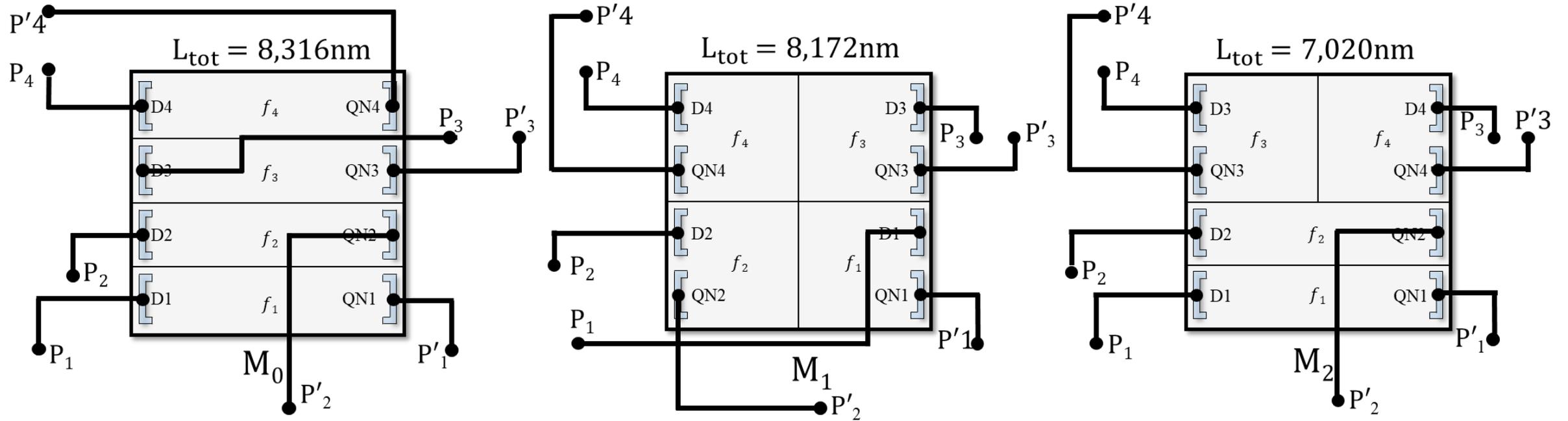
4. Conclusion



Impact of Cell Flipping on WL at Post-Place

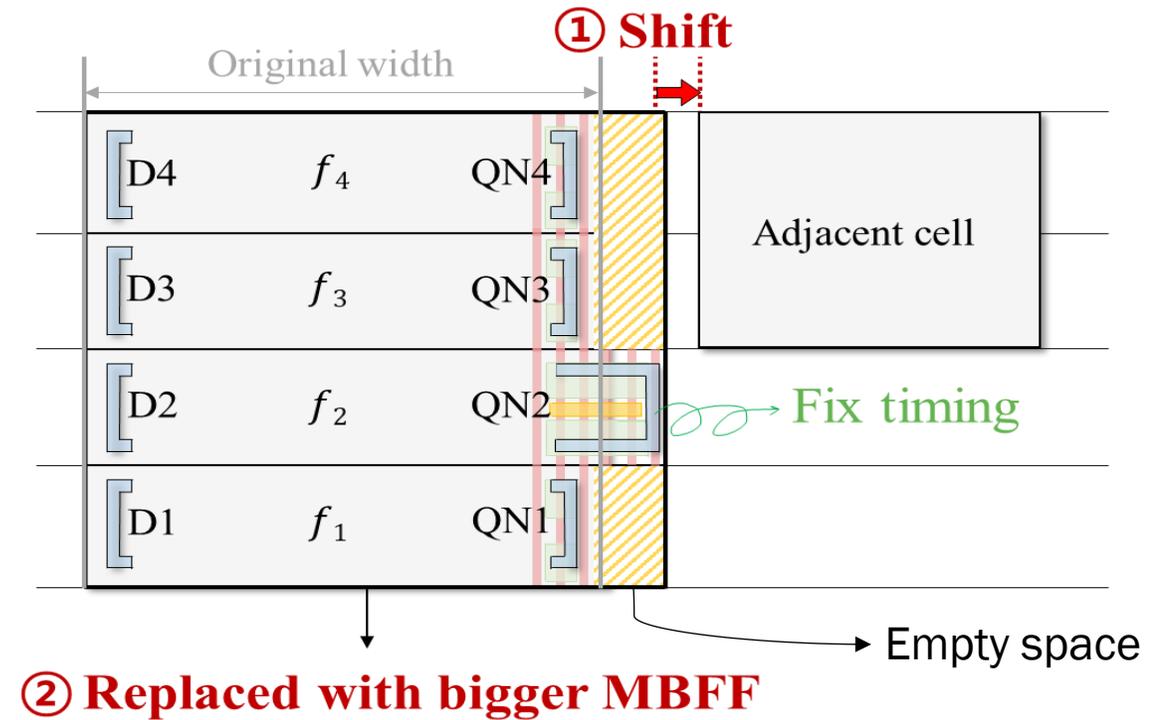
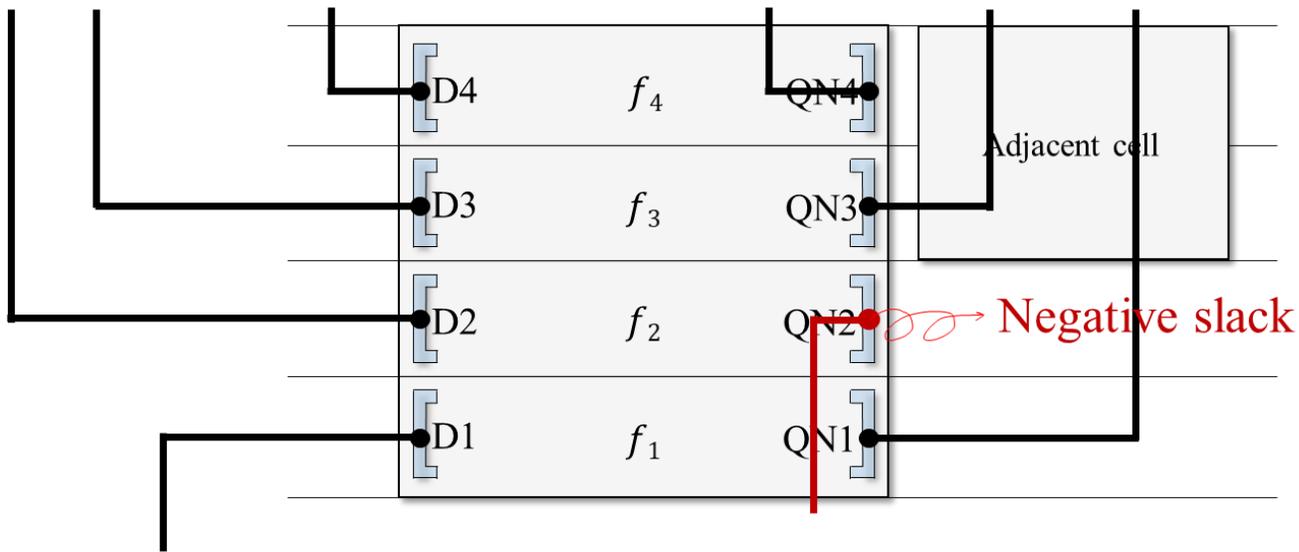


MBFF Binding Problem for Minimizing WL [Jeong, DAC24]

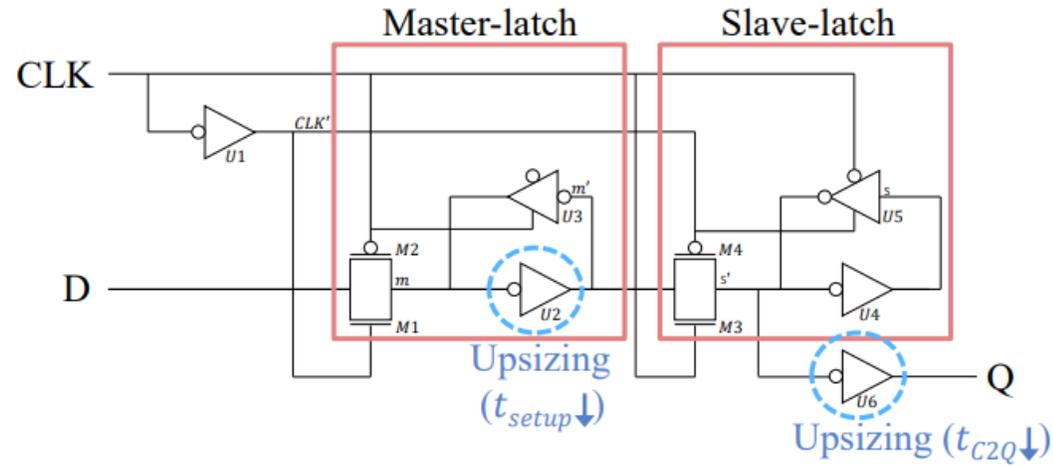


Impact of Resizing MBFF on Timing at Post-Route

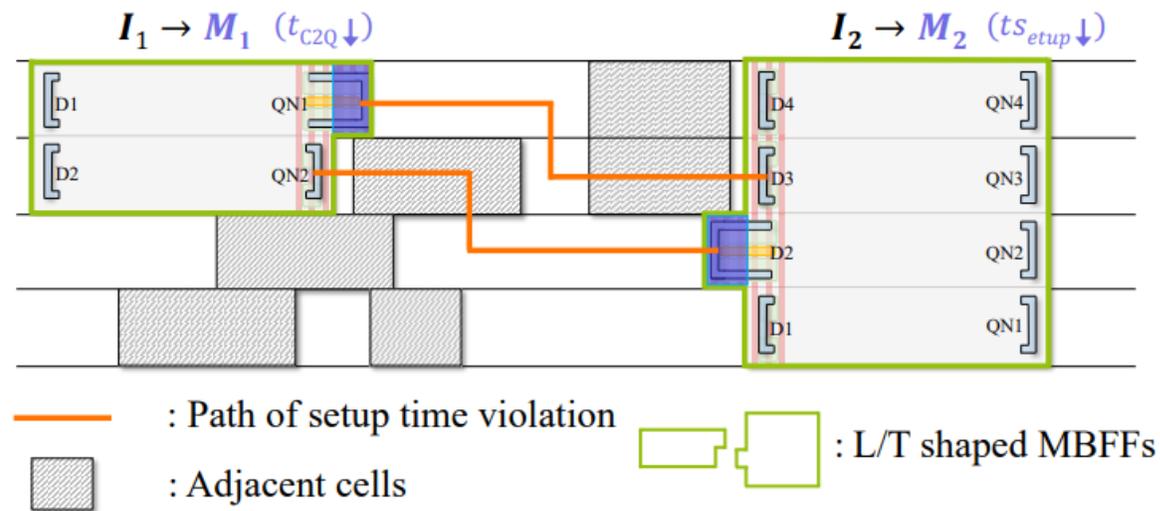
- Degrading design quality by
 - ① Cell shifting
 - ② Multi-bit flip-flop with empty space



Solution 1: Upsize Internal FF(s) Selectively (Jeong, DAC24)

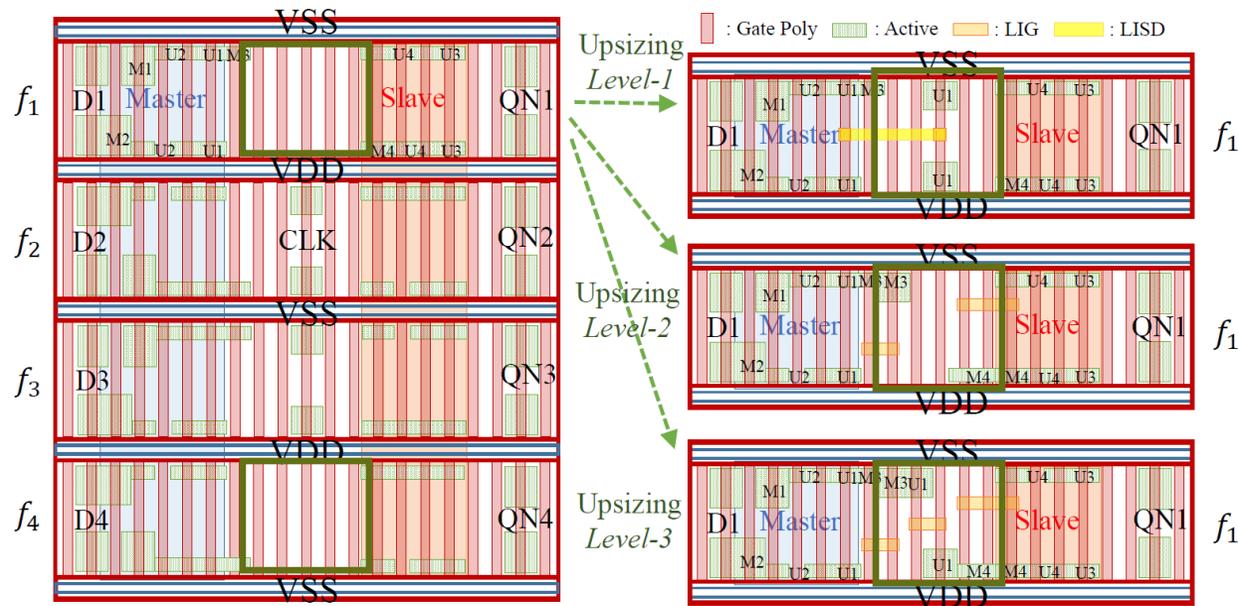


(a) Upsizing transistors in a flip-flop



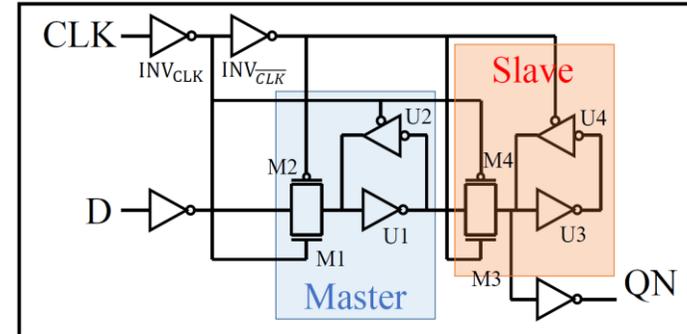
(b) Optimizing timing by replacing MBFFs

Solution 2: Utilize Empty Space for MBFF Resizing [Kim, ICCAD22]



(a) An MBFF layout stacked 4 DFFHQx1 in ASAP7

(b) Upsizing transistor to the empty space



Flip-flop	Unsizing	Transistor upsizing		
		Level-1	Level-2	Level-3
f_1	136.1ps (1)	129.1ps	130.9ps	122.2ps (0.899)
f_2	137.7ps	138.3ps	135.5ps	136.3ps
f_3	136.5ps	136.8ps	134.0ps	134.5ps
f_4	137.9ps (1)	130.8ps	132.1ps	123.5ps (0.896)

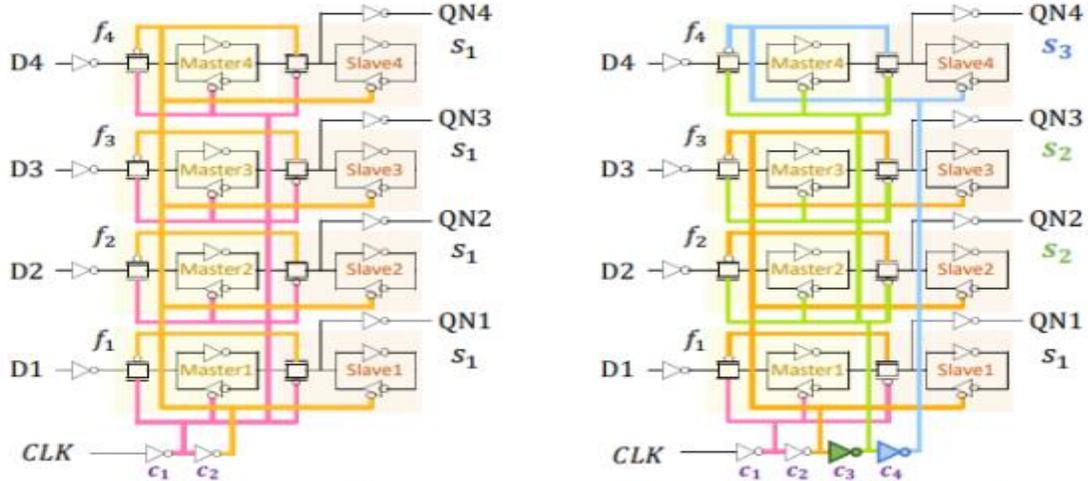
(c) Timing (setup time + clock-to-Q delay) on each flip-flop

Level-1 : Upsizing with 2fins in U1

Level-2 : Upsizing with 2fins in M3 and M4

Level-3 : Upsizing with 2fins in U1 and 2fins in M3 and M4

Solution 3: Utilize Empty Space for Multi-skewed MBFF [Kim, ICCAD23]



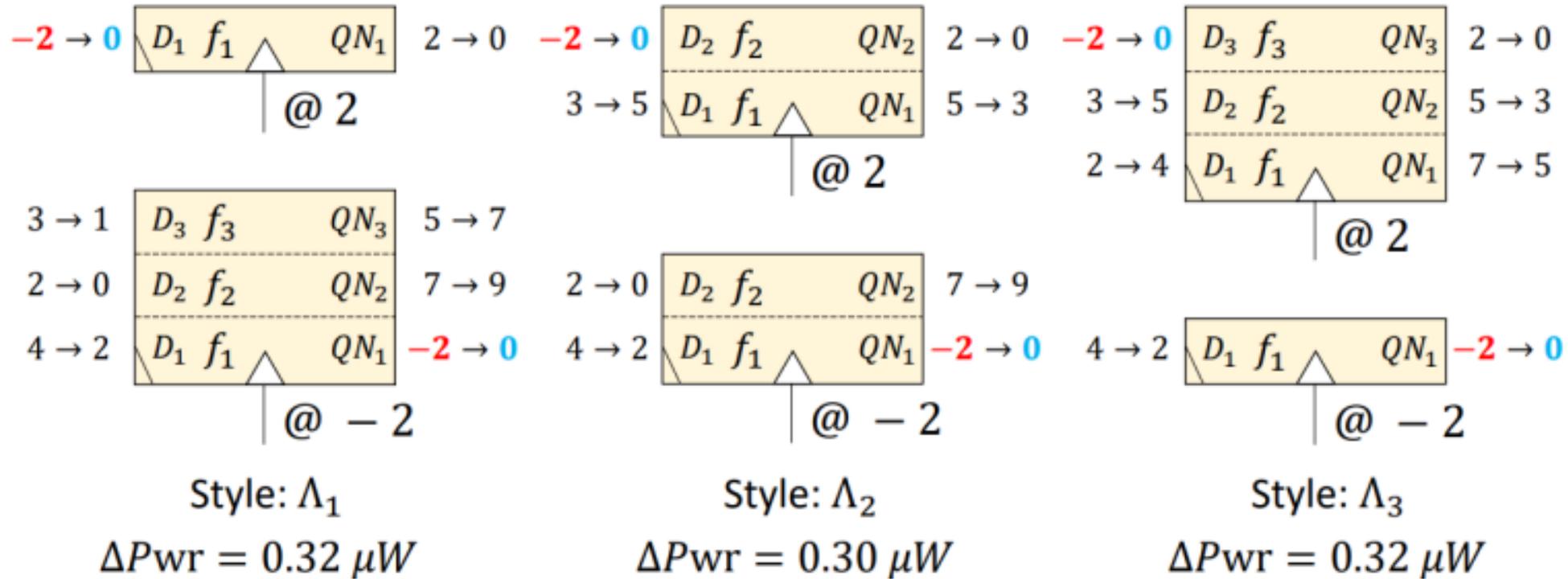
(a) MBFF with a single clock skew (b) MBFF with multiple clock skews



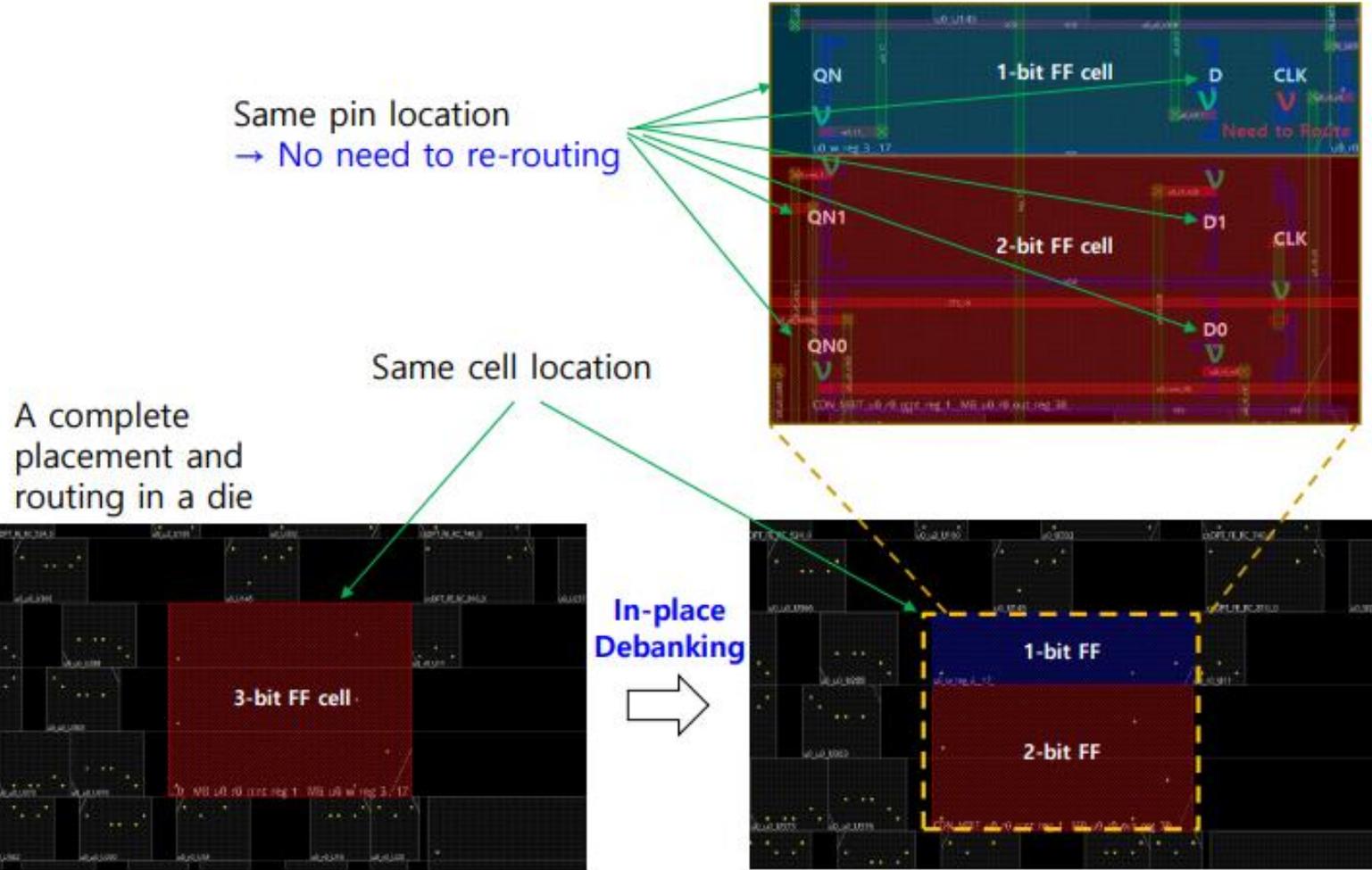
(c) Layout of MBFF in (a) (d) Layout of MBFF in (b)

Legend: Gate poly (orange), LISD (yellow), Inverter (dashed box)

Impact of Debanking on Power and Timing [Kim, ICCAD23]



In-place Debanking [Yang, SOCC23]



Clock Gating vs. MBFFs

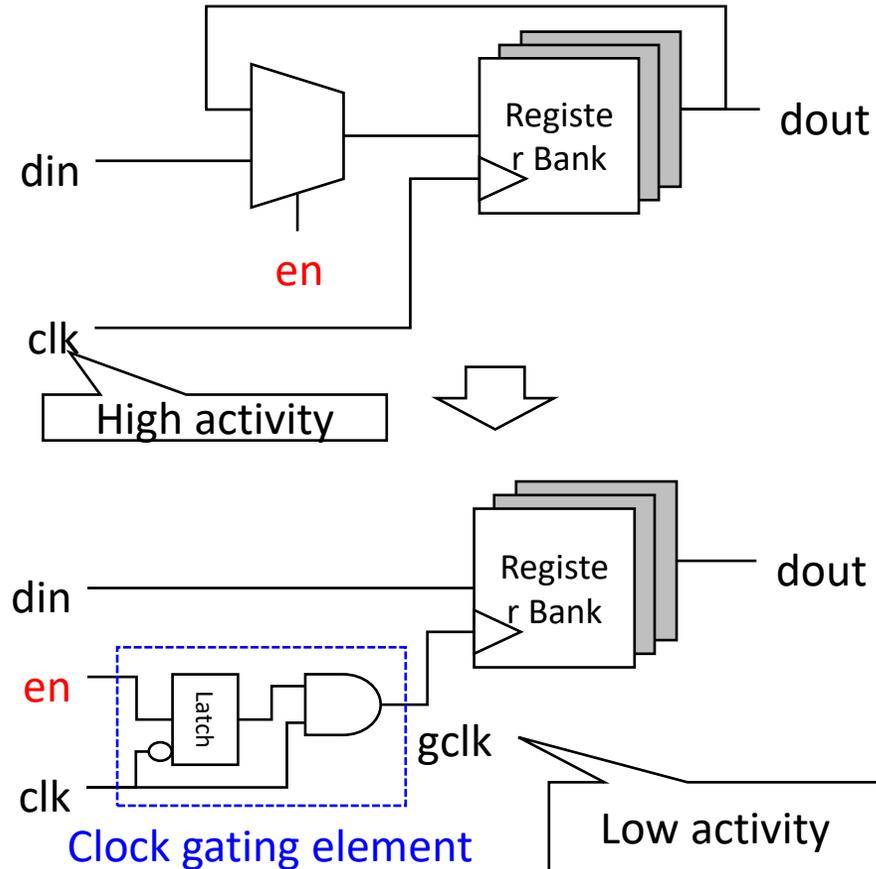
- A clock gating groups flip-flops to be driven by a common clock signal, so that it disables clocks when all the grouped flip-flops do not make output signal toggling. → power saving depends on toggling rates and toggling behavior similarity.
 - An MBFF cell is formed by grouping flip-flops, so that the internal clock inverters are shared by the flip-flops inside of the cell. → power saving depends on MBFF size.
-
- 

Idle Logic-driven Clock Gating

```
module TEST (clk, rstn, en, din,
dout);
input clk, rstn, en;
input [7:0] din;
output [7:0] dout;
reg [7:0] dout;

always @(posedge clk or
negedge rstn) begin
    if (~rstn)
        dout <= 8'h0;
    else
        if (en)
            dout <= din;
end

endmodule
```

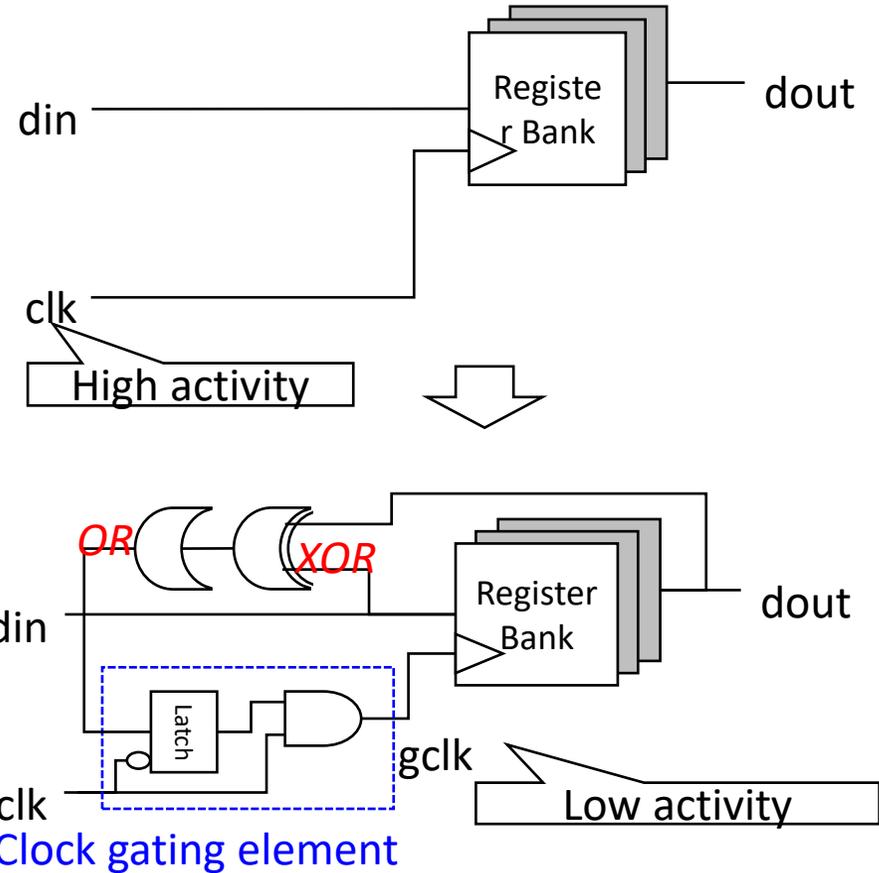


Data Toggling driven Clock Gating

```
module TEST (clk, rstn, din, dout);
input clk, rstn;
input [7:0] din;
output [7:0] dout;
reg [7:0] dout;

always @(posedge clk or negedge
rstn) begin
    if (~rstn)
        dout <= 8'h0;
    else
        dout <= din;
end

endmodule
```



Grouping Flip-flops

Option 1: Clock gating → MBFF grouping

Option 2: MBFF grouping → Clock gating

Option 3: Clock gating for idle logic driven → MBFF grouping
→ Clock gating for data toggling driven

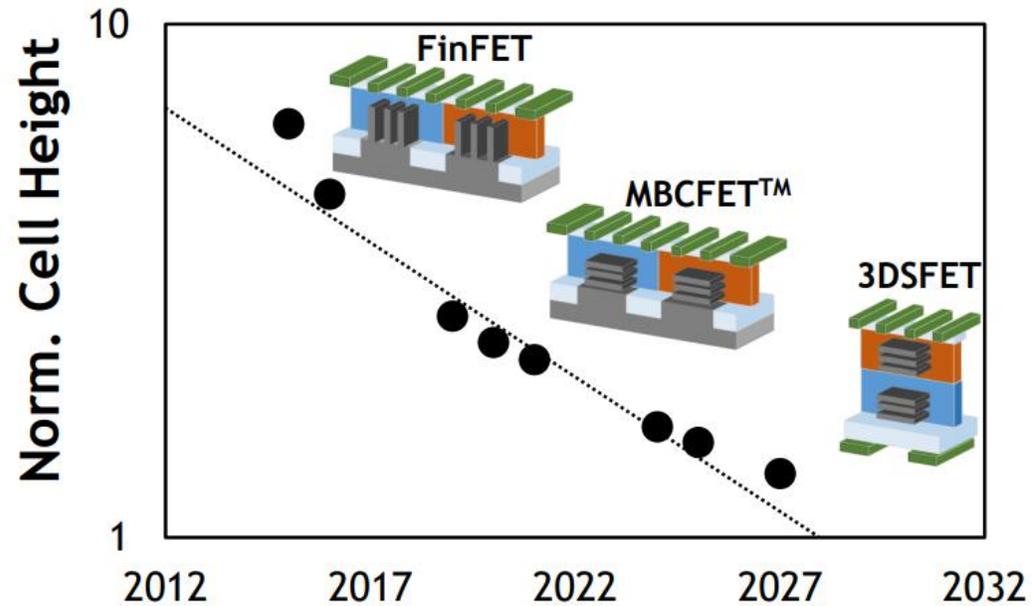
Content

1. Auto-generation of standard cells
 - Design and technology co-optimization (DTCO)
 - Problems
 - Algorithms
 - Multi-row cells
 - Placement legalization
 2. Multi-bit flip-flop (MBFF) cells
 - Structure
 - DTCO flow with MBFF cells
 - DTCO techniques with MBFF cells
 3. Complementary FET (CFET) cells
 - FEOL and BEOL
 - Backside interconnect
 - CFET vs. Flip-FET (FFET)
 4. Conclusion
-

FEOL

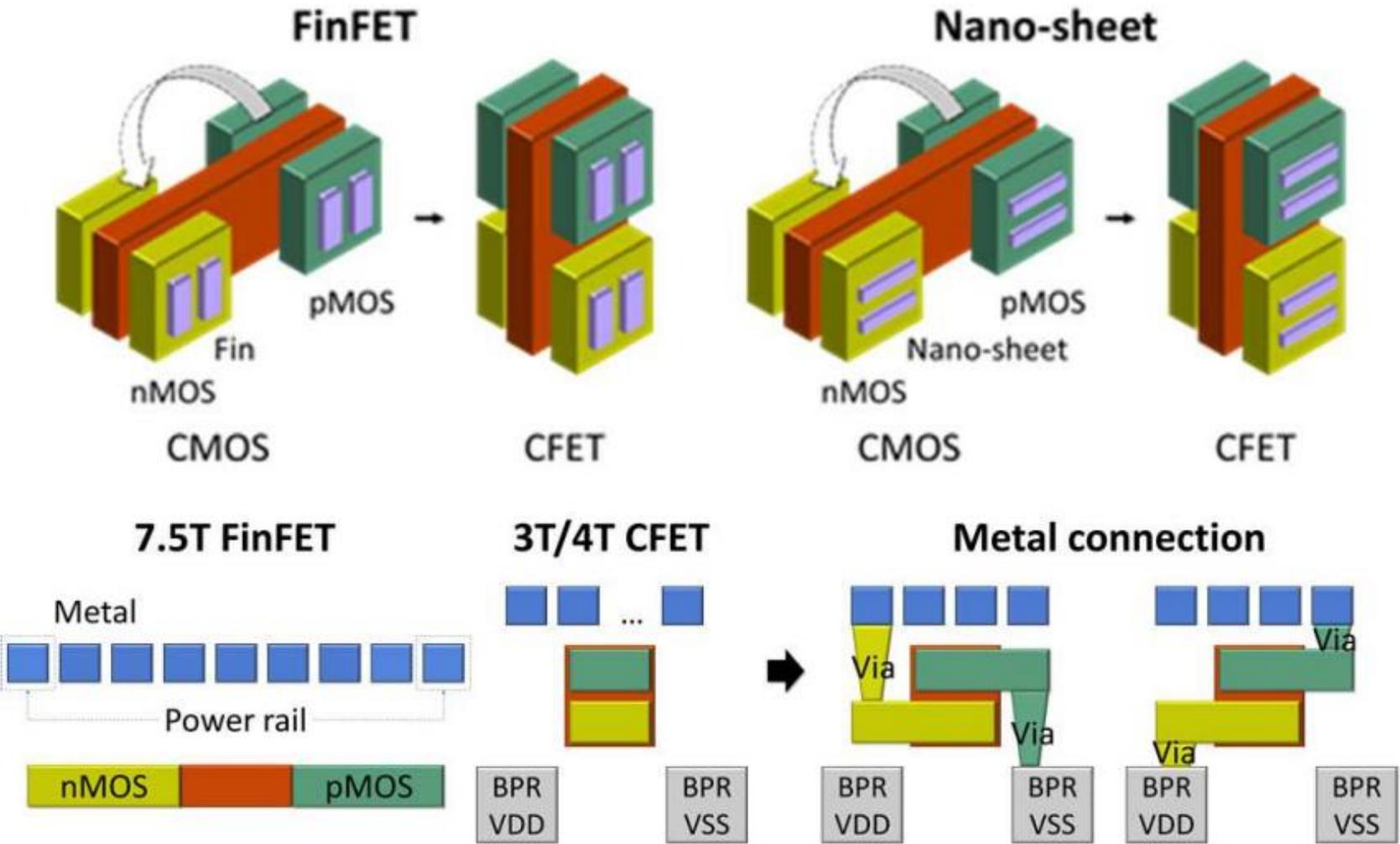
In terms of FEOL, scaling is maintained through device innovation.

– Planar → FinFET → MBCFET™ (Multi-Bridge Channel FET) → 3DSFET

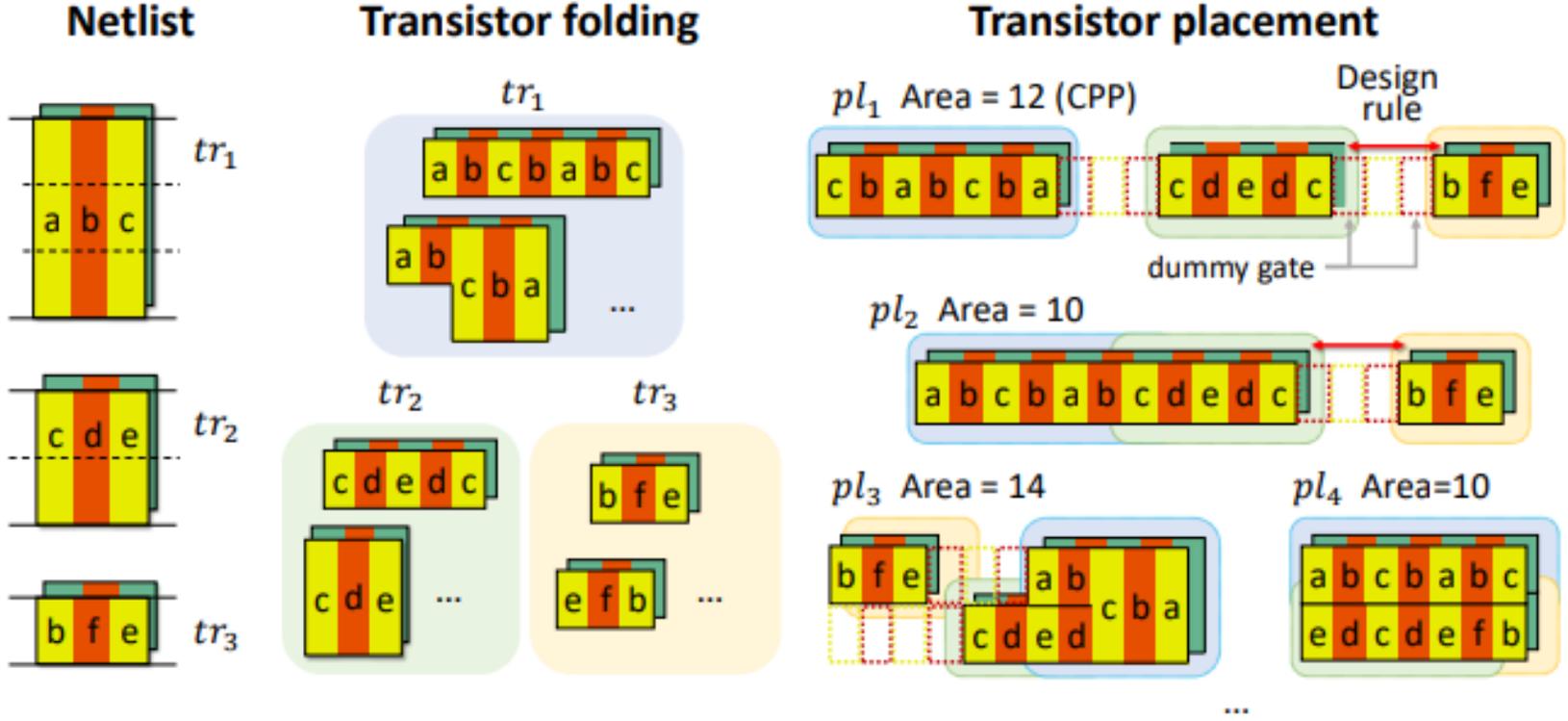


- CFET follows the concept of the 3D stacked Fin-CMOS
- Pro: scalability beyond GAA
- Cons: Insufficient routing resources (minimum 3T)

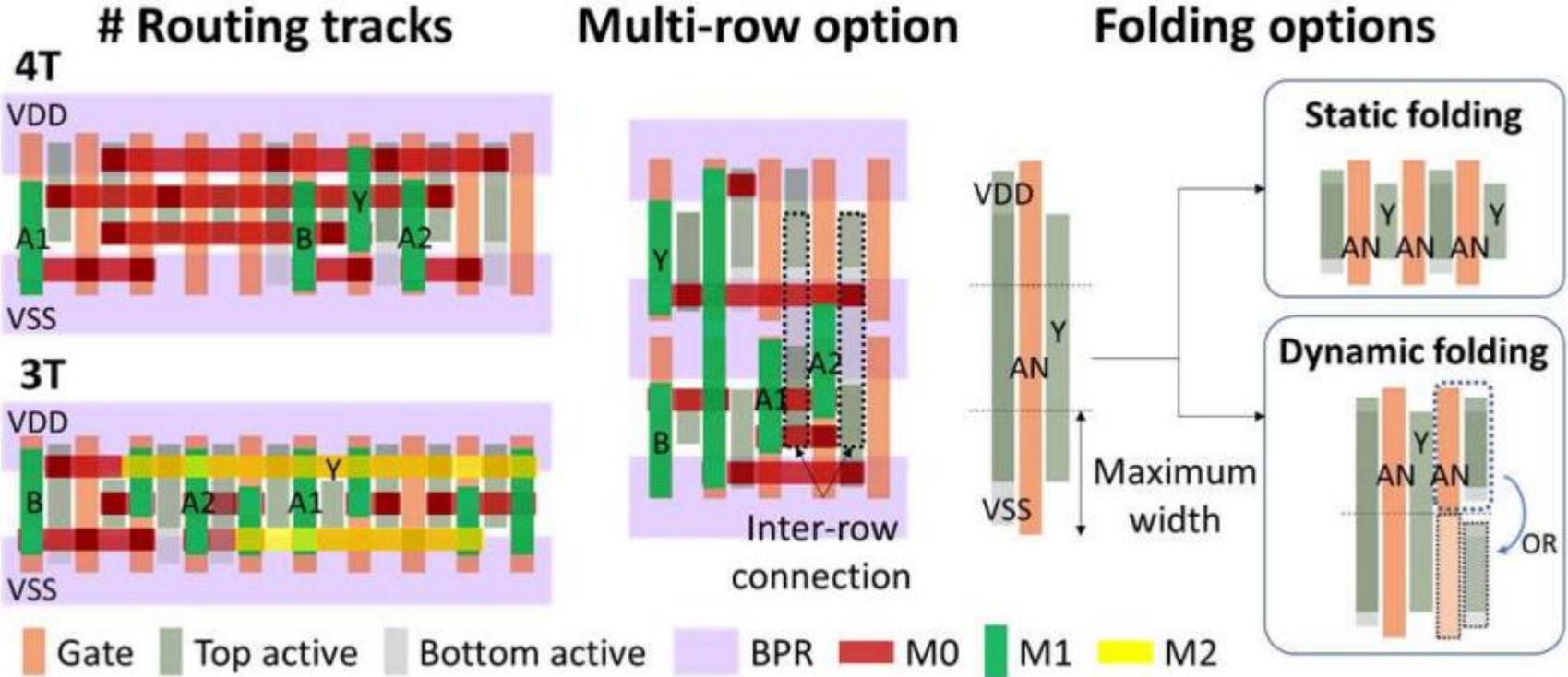
CFET Cell Structure



Transistor Folding and Placement



Static vs. Dynamic Folding



CFET Cell Generation Algorithms

50x speed-up, 5% same cell area,
38% metal length reduction

	Cheng, TVLSI21 (UCSD)	Cheng, J. Exp. Solid- State Device21 (UCSD)	Kim, DAC24 (SNU)
Single-row height	0	0	0
Multi-row height		0	0
Transistor placement (TP)	0	0	0
Static folding (SF)	0	0	
Dynamic folding (DF)			0
In-cell routing (IR)	0	0	0
Optimality (TP+SF)	0	0	
Optimality (TP+DF)			0
Optimality (IR)	0	0	0
Optimality (TP+SF+IR)	0	0	
Optimality (TP+DF+IR)	x	x	x

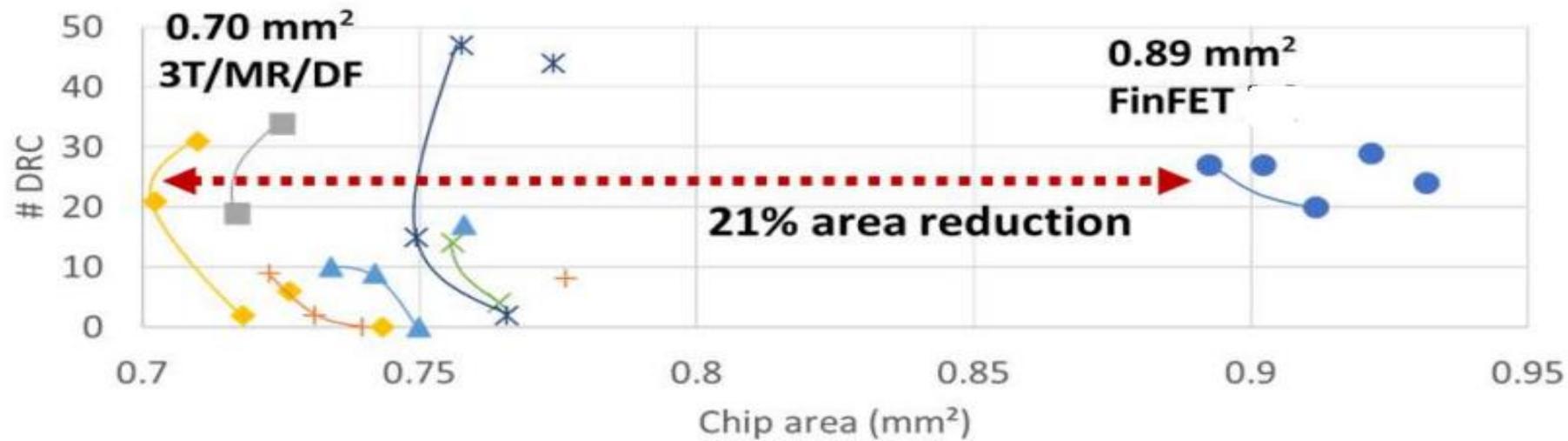
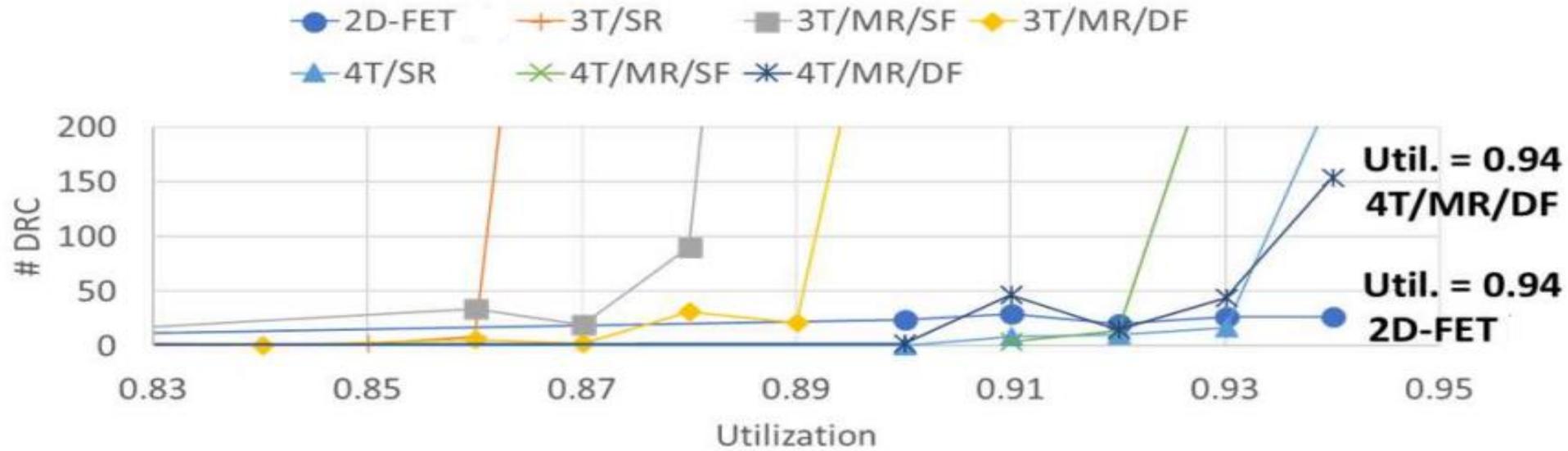
7% performance improvement



PPA Comparison

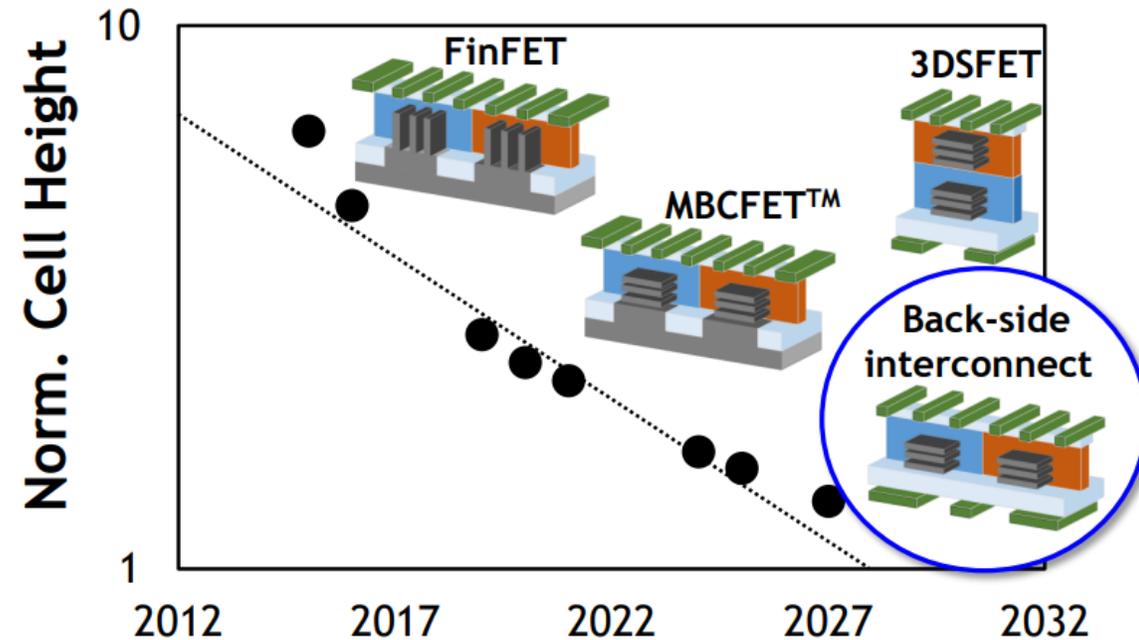
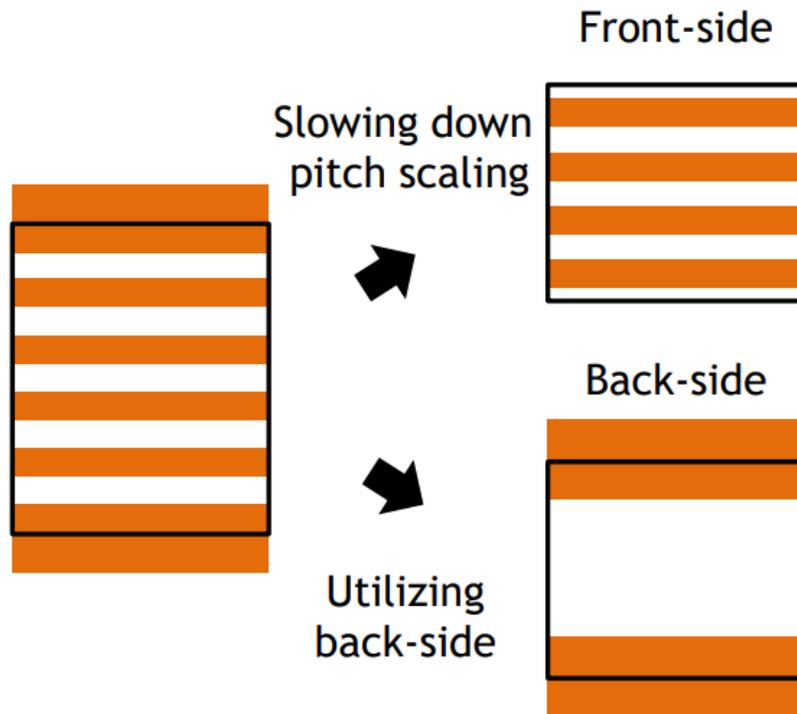
Cell netlist			Cell width (#CPP)				#Cell row		Metal length [†]				Usage of M2 tracks				Runtime (sec.)						# Synthesized cells			
			Multi-row		Single-row				Multi-row		Single-row		Multi-row		Single-row		Multi-row			Single-row			Multi-row		Single-row	
Name	#FET	#Net	[3]	Ours	[2]	Ours	[3]	Ours	[3]	Ours	[2]	Ours	[3]	Ours	[2]	Ours	[3]	Ours (1 cell)	Ours (all)	[2]	Ours (1 cell)	Ours (all)	[3]	Ours	[2]	Ours
AND2x2	6	7	6	6	6	6	1	1	85	53	60	56	1	0	0	0	8.65	0.29	0.29	8.12	0.45	0.45	1	1	1	1
AND3x1	8	9	7	6	6	6	1	1	108	59	68	63	2	0	0	0	50.44	0.91	0.91	30.09	1.40	1.40	1	1	1	1
AND3x2	8	9	8	7	7	7	1	1	106	65	76	69	1	0	0	0	50.52	1.44	1.44	28.35	1.64	1.64	1	1	1	1
AOI21x1	6	8	6	5	9	9	2	2	162	75	142	122	2	0	0	0	3575.33	0.79	1.39	119.69	6.25	70.36	1	2	1	10
AOI22x1	8	10	7	6	11	11	2	2	226	97	240	203	2	0	1	1	6808.18	2.63	14.43	363.16	20.47	42.71	1	5	1	2
BUFx2	4	5	5	5	5	5	1	1	60	37	40	39	1	0	0	0	4.22	0.10	0.10	4.92	0.21	0.21	1	1	1	1
BUFx3	4	5	6	6	6	6	1	1	70	51	53	56	1	0	0	0	5.49	0.18	0.18	11.2	0.58	0.58	1	1	1	1
BUFx4	4	5	7	7	7	7	1	1	77	57	59	62	1	0	0	0	8.75	0.24	0.24	7.91	0.77	0.77	1	1	1	1
BUFx8	4	5	12	12	12	12	1	1	123	103	105	107	1	0	0	0	48.64	0.95	0.95	43.65	4.36	4.36	1	1	1	1
DFFHQN	24	17	9	9	16	15	2	2	206	256	182	203	0	2	0	1	21071.8	915.39	1026.77	6831.77	410.27	1359.70	1	8	1	10
FA	24	17	8	7	14	14	2	3	288	164	379	276	4	0	2	3	24394.2	1281.33	1281.33	6653.07	484.55	1568.44	1	1	1	9
INVx1	2	4	3	3	3	3	1	1	44	20	23	22	2	0	0	0	1.48	0.02	0.11	0.49	0.06	0.29	1	4	1	4
INVx2	2	4	4	4	4	4	1	1	52	26	29	27	1	0	0	0	1.94	0.05	0.27	1.03	0.12	0.85	1	4	1	4
INVx4	2	4	6	6	6	6	1	1	70	46	48	47	1	0	0	0	457	0.16	0.72	3.46	0.60	2.79	1	4	1	4
INVx8	2	4	10	10	10	10	1	1	108	86	92	111	1	0	0	0	15.08	0.56	2.56	19.14	3.40	10.09	1	4	1	3
NAND2x1	4	6	6	6	6	6	1	1	103	108	74	70	2	2	0	0	12.76	0.52	1.33	15.88	1.33	11.13	1	3	1	8
NAND2x2	4	6	11	10	10	10	1	1	99	184	131	125	2	2	0	0	41.26	2.27	8.36	33.83	5.28	57.20	1	6	1	10
NAND3x1	6	8	14	7	11	11	1	2	283	175	146	147	2	2	0	0	887.56	5.16	34.34	124.11	12.19	101.95	1	7	1	8
NAND3x2	6	8	26	11	21	21	1	2	441	150	283	303	2	0	0	0	909.03	30.12	55.62	2869.53	81.86	609.56	1	3	1	8
NOR2x1	4	6	6	6	6	6	1	1	103	108	74	70	2	2	0	0	16.96	0.57	1.46	12.89	1.29	12.51	1	3	1	9
NOR2x2	4	6	11	10	10	10	1	1	199	197	131	119	2	2	0	0	44.37	2.73	5.58	27.94	4.99	52.44	1	3	1	10
NOR3x1	6	8	14	7	11	11	1	2	283	178	156	163	2	2	0	0	914.35	9.52	46.44	52.33	11.07	108.84	1	7	1	10
NOR3x2	6	8	26	11	21	21	1	2	510	176	286	320	2	0	0	1	1027.98	34.81	54.86	1897.53	82.25	373.63	1	2	1	5
OAI21x1	6	8	6	5	9	9	2	2	150	85	149	131	2	0	0	0	2122.94	1.07	2.95	52.52	7.93	62.86	1	3	1	8
OAI22x1	8	10	7	6	11	11	2	2	229	111	255	174	2	0	1	0	7043.85	3.96	25.65	612.6	19.34	167.27	1	6	1	8
OR2x2	6	8	6	6	6	6	1	1	85	53	60	56	1	0	0	0	14.22	0.52	0.52	12.99	0.50	0.50	1	1	1	1
OR3x1	8	9	6	6	6	6	1	1	72	59	68	63	2	0	0	0	73.58	1.96	1.96	76.77	1.54	1.54	1	1	1	1
OR3x2	8	9	7	7	7	7	1	1	106	65	76	69	2	0	0	0	95.94	2.42	2.42	89.22	1.65	1.65	1	1	1	1
XNOR2x1	10	9	7	7	11	11	2	2	268	116	223	180	2	0	1	1	5766.47	58.11	99.17	977	47.47	87.45	1	10	1	3
XOR2x1	10	9	7	7	11	11	2	2	256	121	213	175	2	1	1	1	2122.94	63.66	95.70	134.86	50.37	94.69	1	8	1	4
Avg.	6.80	7.70	8.80	7.03	9.30	9.27	1.27	1.43	165.73	102.70	130.70	120.93	1.67	0.50	0.20	0.27	2586.53	45.29	56.81	703.87	42.14	160.26	1.00	3.43	1.00	4.90
Norm.			1.00 [‡]	0.95	1.00	1.00	1.00	1.13	1.00	0.62	1.00	0.93	1.00	0.30	1.00	1.33	1.00	0.02	0.02	1.00	0.06	0.23	1.00	3.43	1.00	4.90

DTCO Results



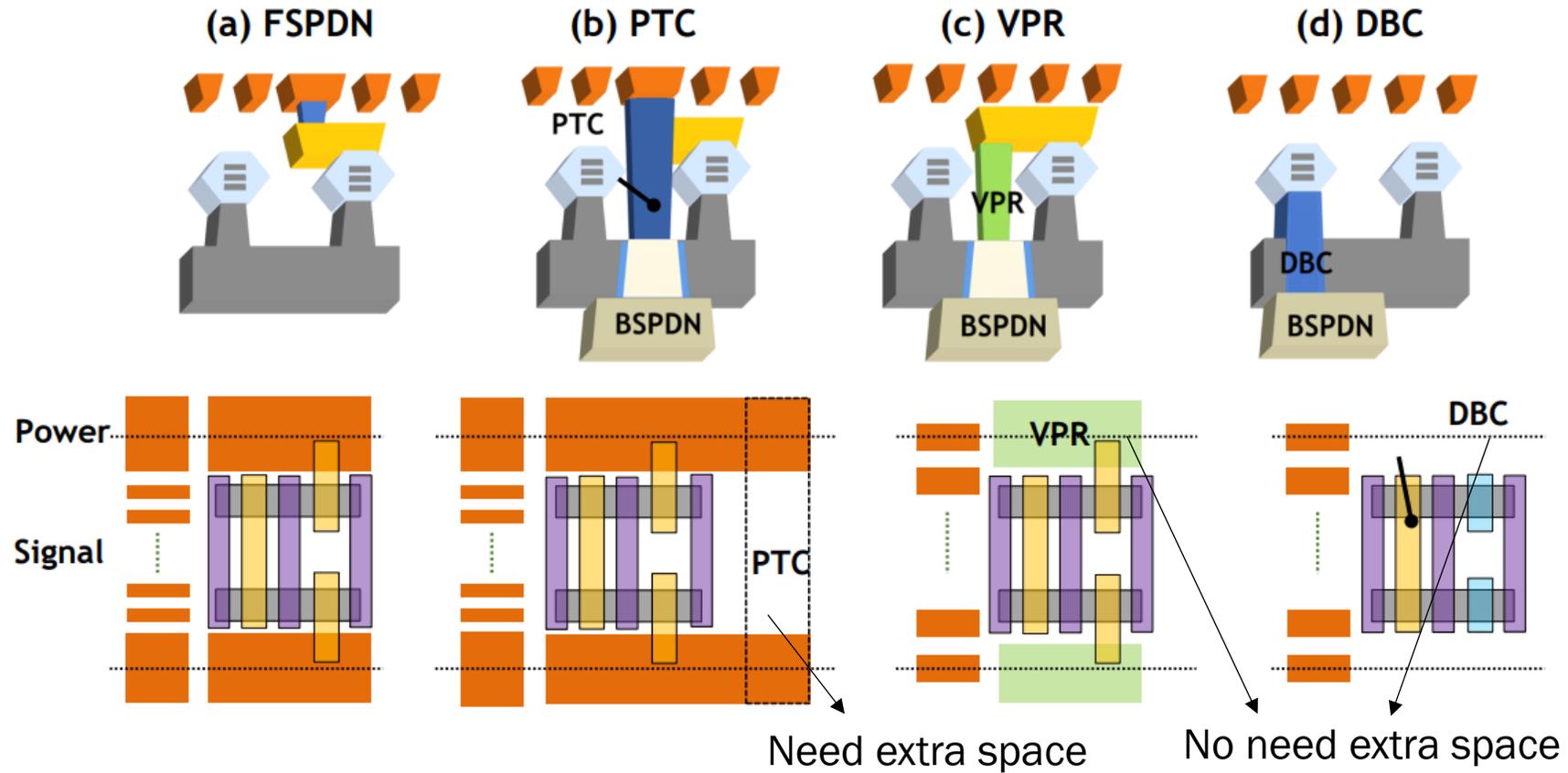
BEOL

In terms of BEOL, scaling is continued through innovation that leverages the backside instead of pitch reduction.



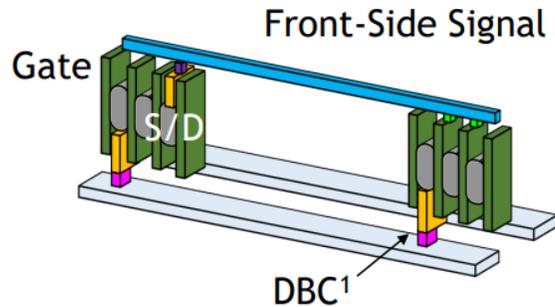
BSPDN [Kim, VLSI Symp24]

Power Tapping Cell, Via Power Rail, Direct Back-side Contact schemes

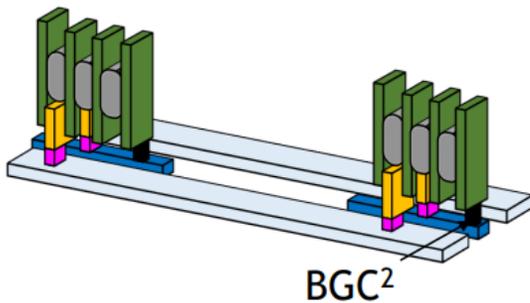


Backside Interconnect [Kim, VLSI Symp24] (1/2)

BSPDN



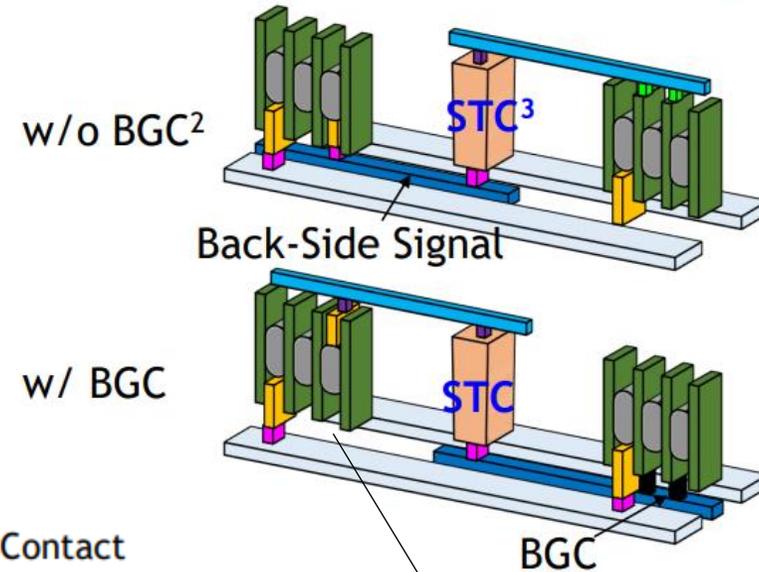
Intra-cell BSS routing



Intra-cell BSS routing requires BGC to transfer signals between gate and S/D on the back-side.

Inter-cell BSS routing requires STC to transfer signals between the back and front side. → Enabling clock net on the back side

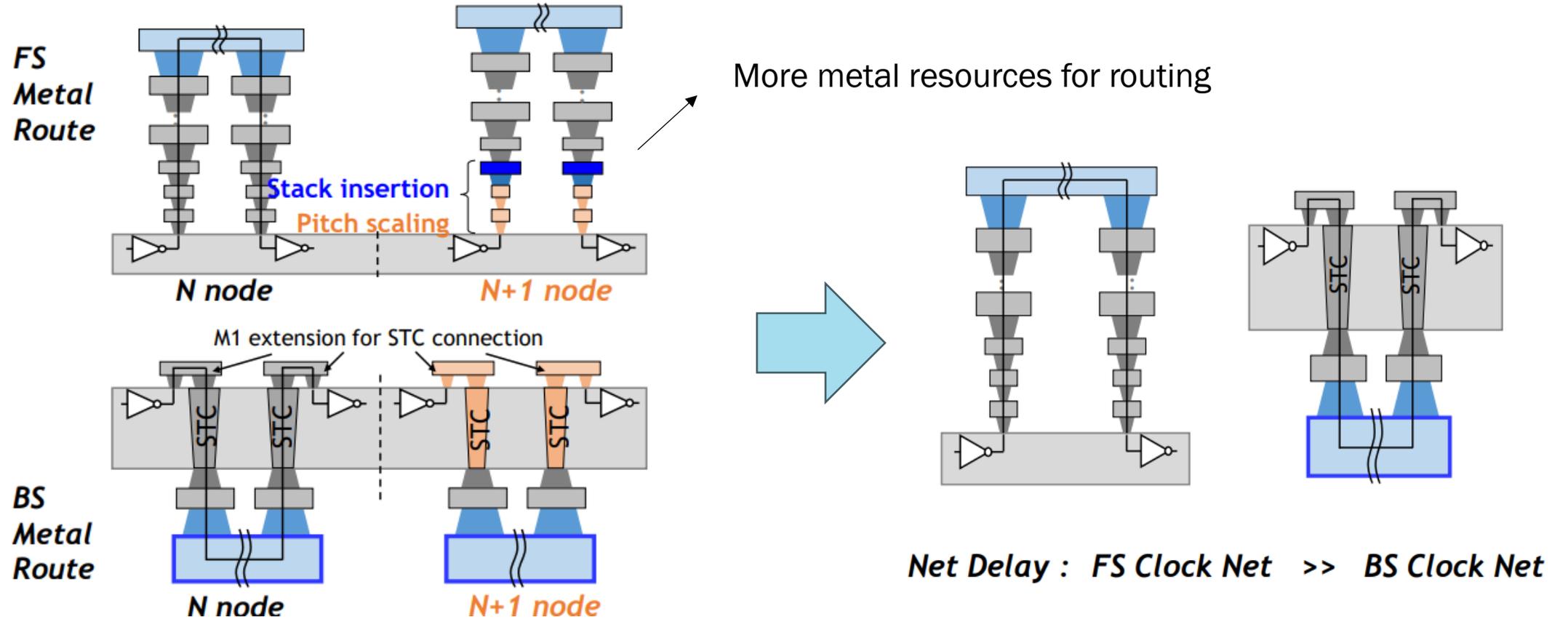
Inter-cell BSS routing



- ¹DBC : Direct Back-side Contact
- ²BGC : Back-side Gate Contact
- ³STC : Signal Transfer Cell

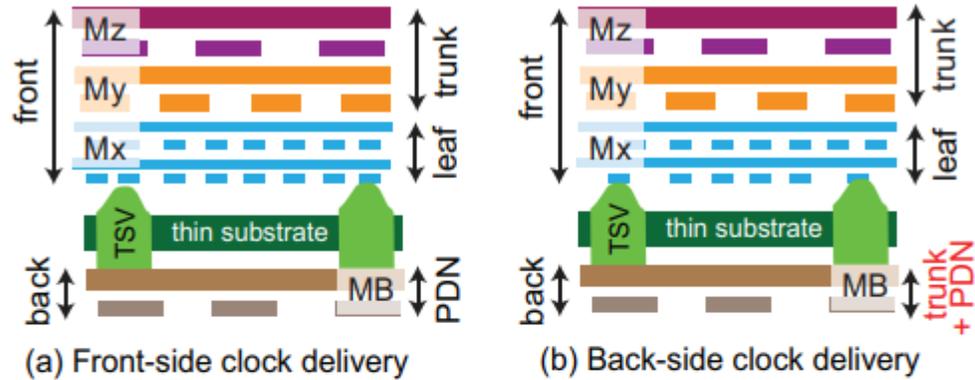
No need STC if DBC is also supported.

Backside Interconnect [Kim, VLSI Symp24] (2/2)



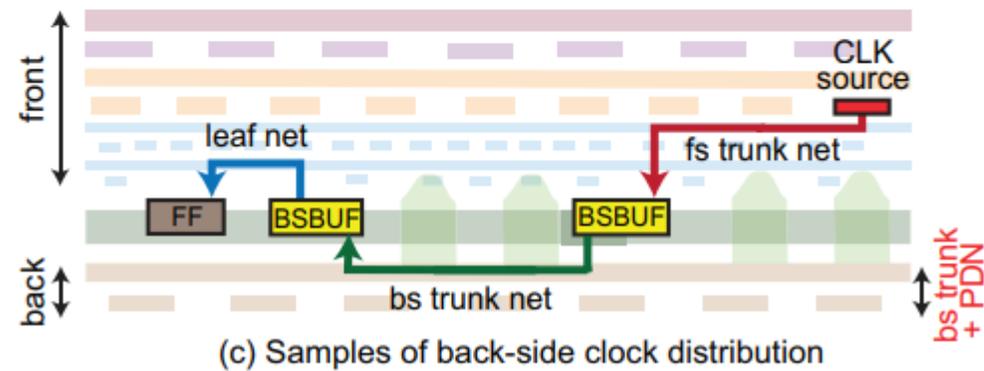
BS metal routing shows faster speed between repeaters.

Backside Clock Routing [Lim, VLSI Symp24]

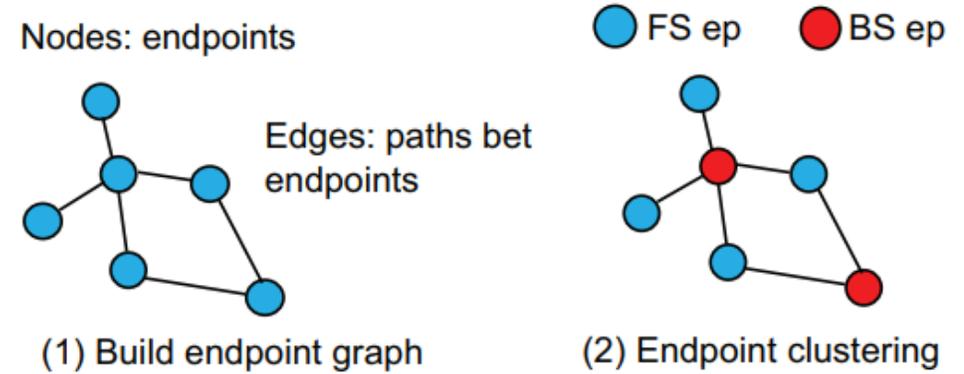


(a) Front-side clock delivery

(b) Back-side clock delivery

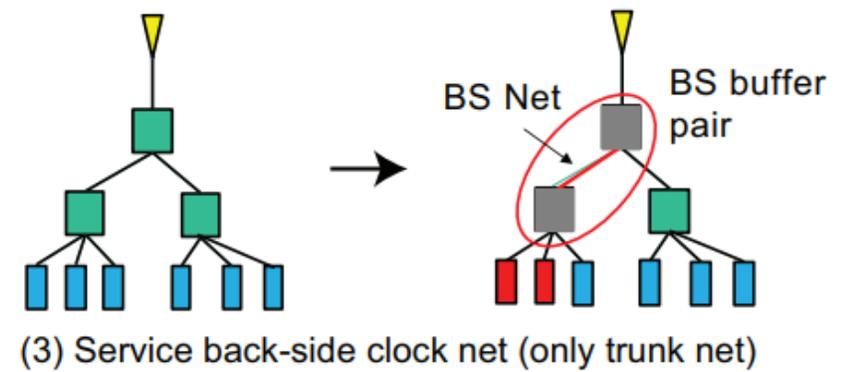


(c) Samples of back-side clock distribution



(1) Build endpoint graph

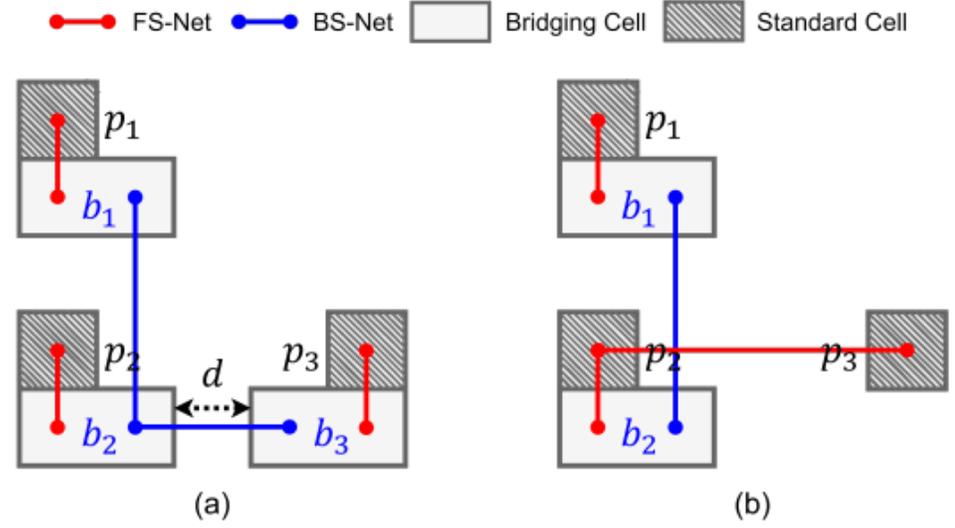
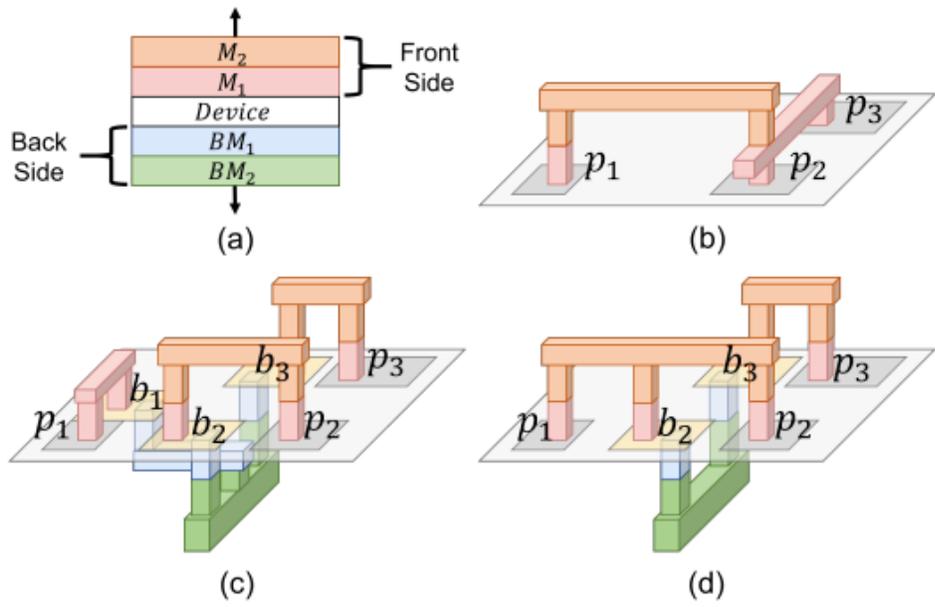
(2) Endpoint clustering



(3) Service back-side clock net (only trunk net)

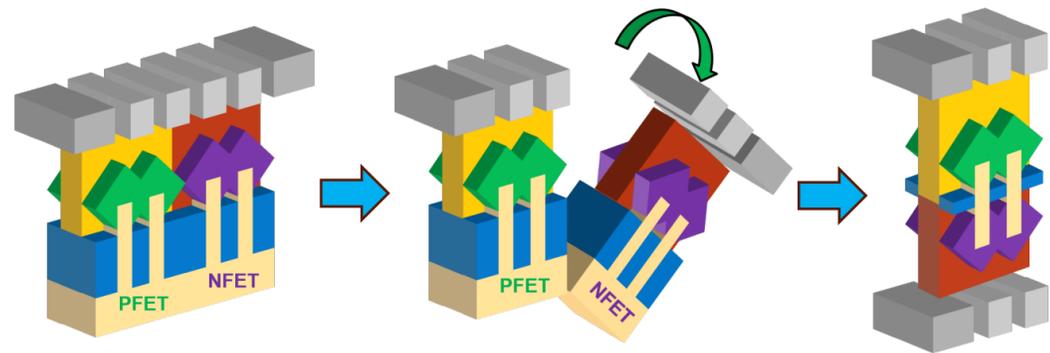
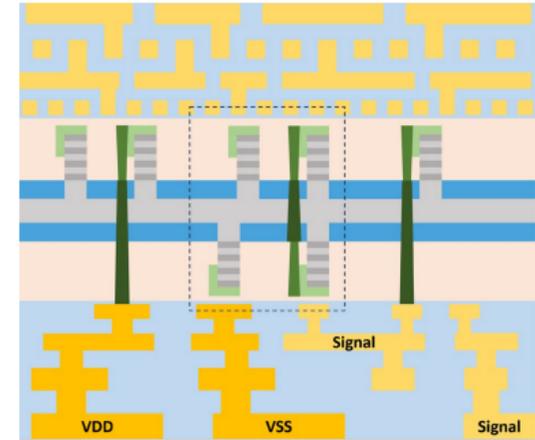
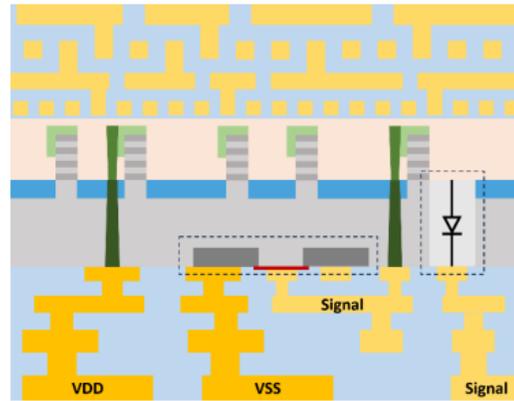
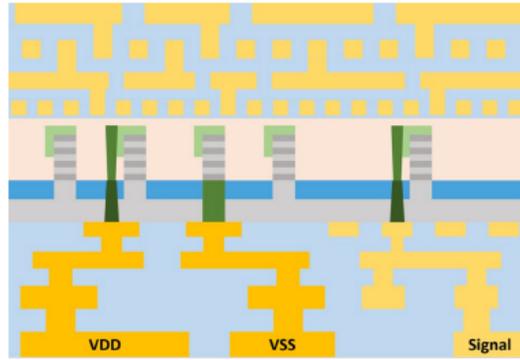
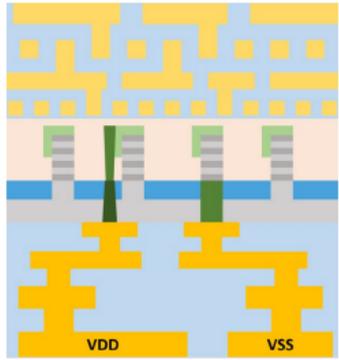
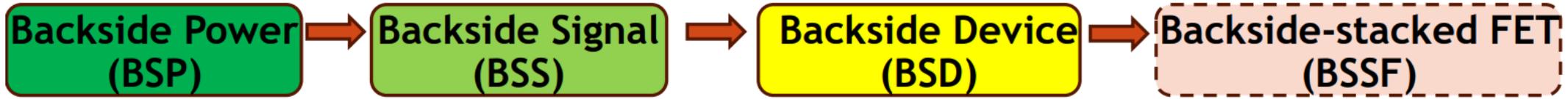
GNN based EP clustering

Backside ECO-routing [Tsai, ICCAD24]

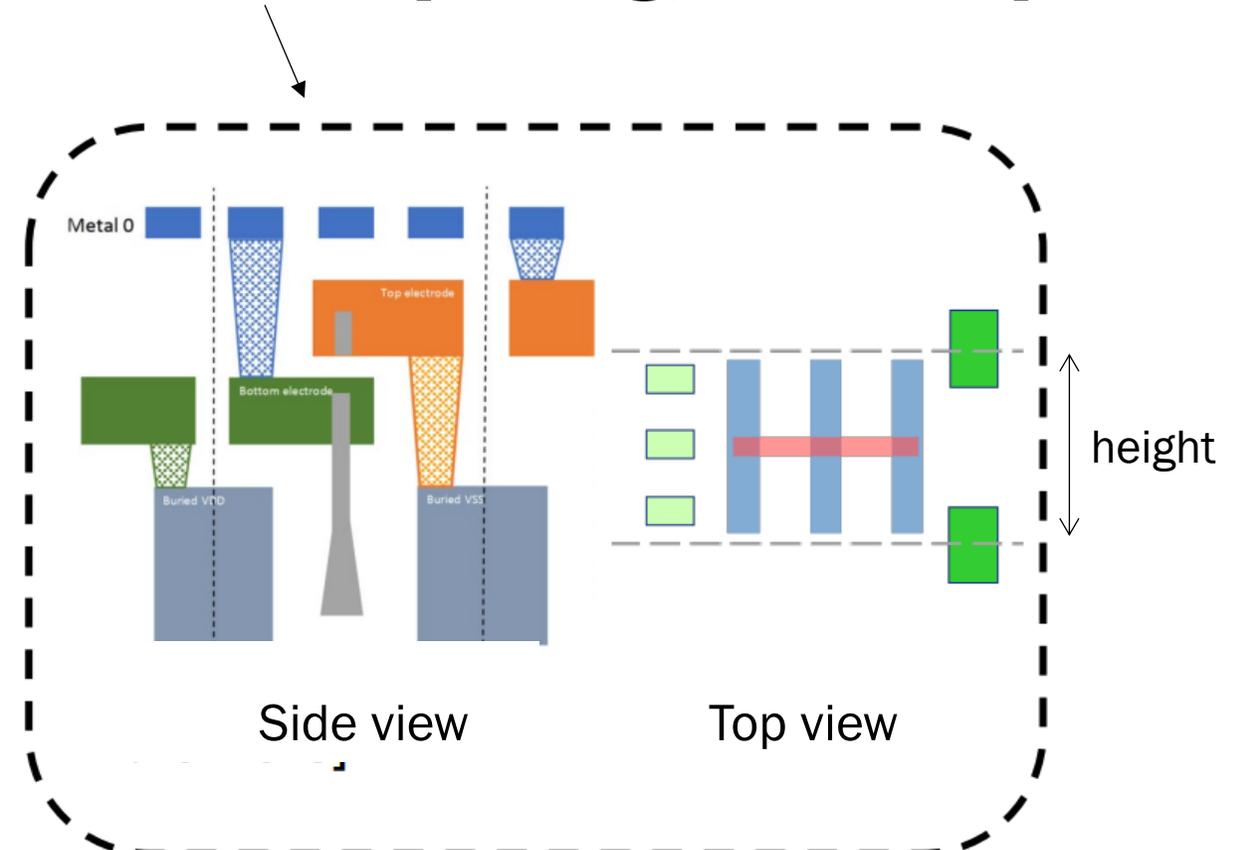
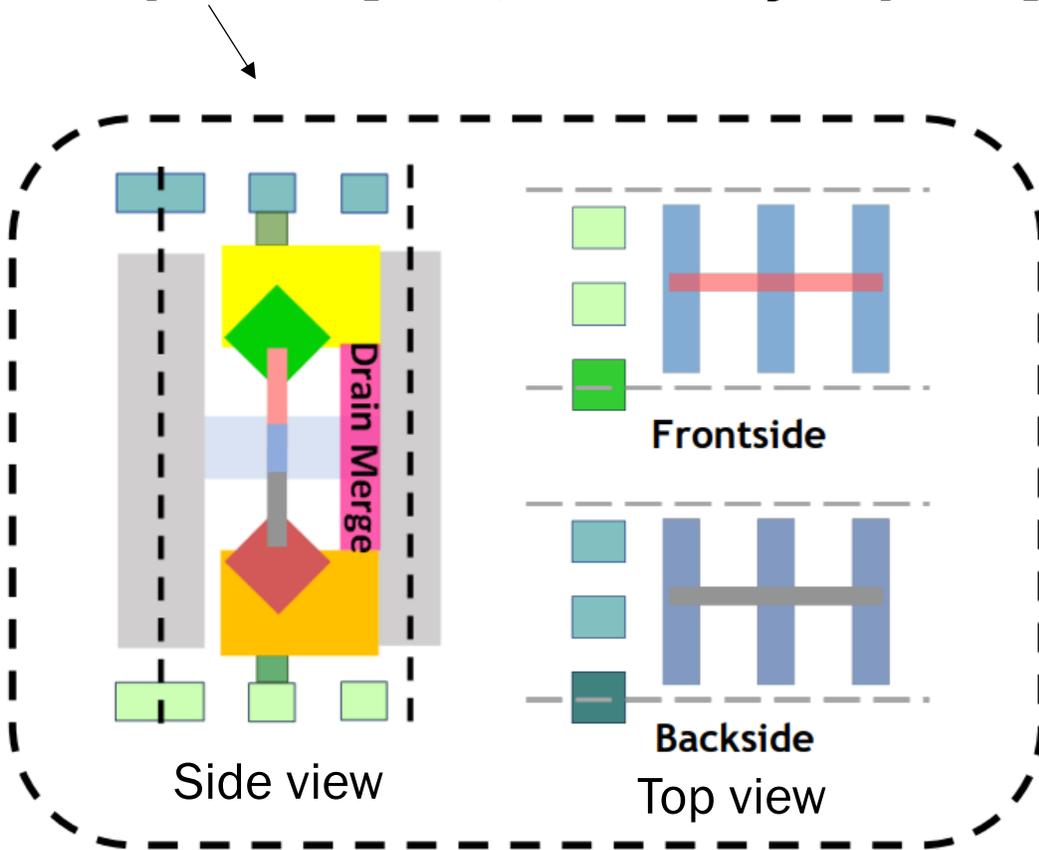


GNN based EP clustering

Backside Roadmap



2.5T Flip-FET [Liu, VLSI Symp24] vs. 3T CFET [Cheng, TVLSI21]



- Smaller cell height
- More local routing resources
 - 2 signal tracks and 1 shared power rail on each side → 4 signals in total 😎
 - 3T CFET with BS power rail → 3 signals in total 😞

Summary

- An effective and efficient DTCO framework is essential for developing advanced new technology nodes in industry.
 - Fast and accurate evaluation of DTCO parameter is essential.
 - ML-driven Cell/Chip PPA prediction models are required.
- Automatic cell generation is **a core part** for the development of DTCO framework.
 - Multi-row cells increases the synthesis complexity as well as degrades the placement legalization quality.
- Intensive use of MBFFs adversely affect **WL** and **timing**.
 - A considerable attention has been paid to how the diverse structures of MBFFF cells can be leveraged on the DTCO framework.
- CFET technology is **a new challenge** in EDA, which drastically impacts chip PPA.
 - The core issues are the CFET cell generation and the utilization of back interconnects for PDN, clock, and signal routing.

Thank you!!
Q&A

A blue abstract graphic with a textured, crystalline appearance, located in the bottom right corner of the slide.