

AS-07 Lecture Notes

Handling Placement Constraints in Analog Layout Synthesis

在“類比佈局”做“元件擺放”

Mark Po-Hung Lin

College of Artificial Intelligence

National Yang Ming Chiao Tung University

Handling Placement Constraints in Analog Layout Synthesis

❖ Lecture 1

Introduction to Analog Layout Synthesis and Placement Constraints

❖ Lecture 2

Analog Placement Method Considering Symmetry Constraints

☞ 很多跟 Analog Placement 有關的研究，都會考慮這個 constraint

❖ Lecture 3

Analog Placement Method Considering Other Placement Constraints

Handling Placement Constraints in Analog Layout Synthesis

教育部智慧晶片系統與應用人才培育計畫
111年度模組教材發展計畫

Lecture 1

Introduction to Analog Layout Synthesis and Placement Constraints

Mark Po-Hung Lin

College of Artificial Intelligence

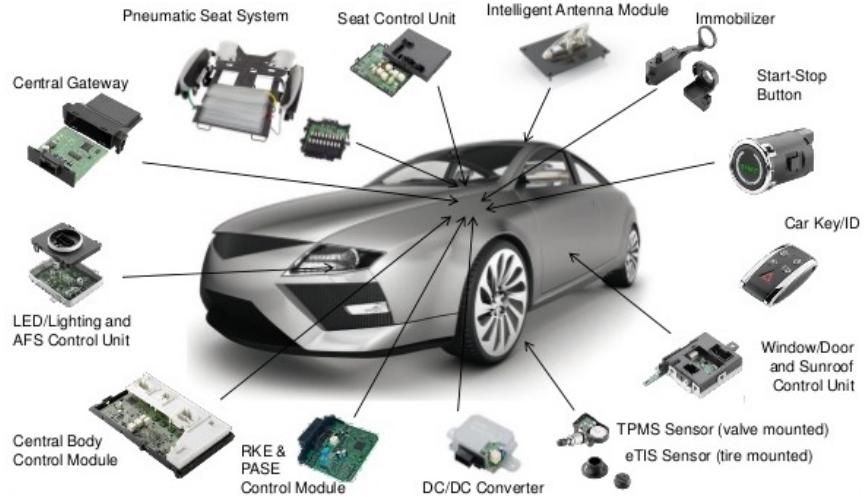
National Yang Ming Chiao Tung University

Analog IC Applications

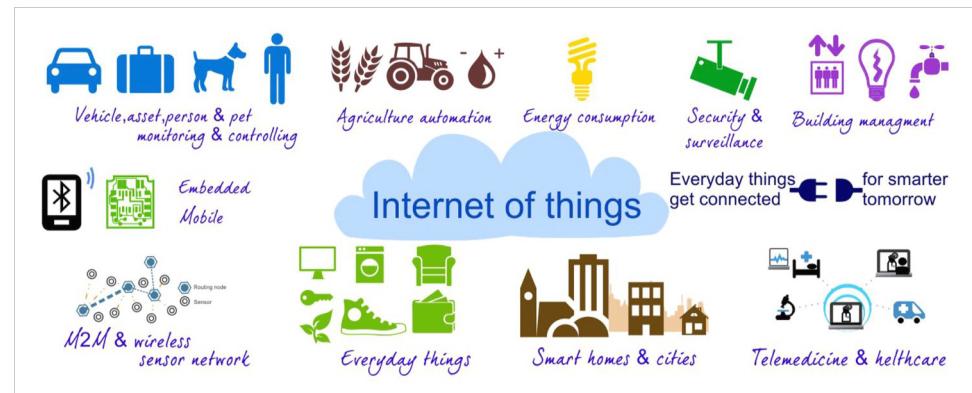
Consumer Electronics



Automotive Electronics



Bioelectronics



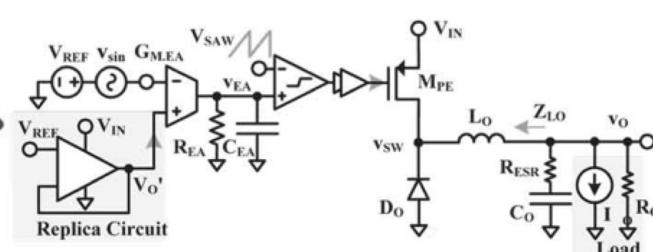
Internet of Things

Design Procedure for Analog ICs

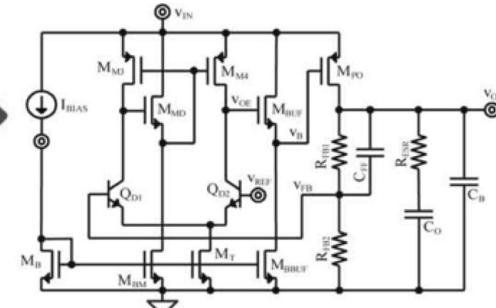
設計流程
(很多地方都是手工)



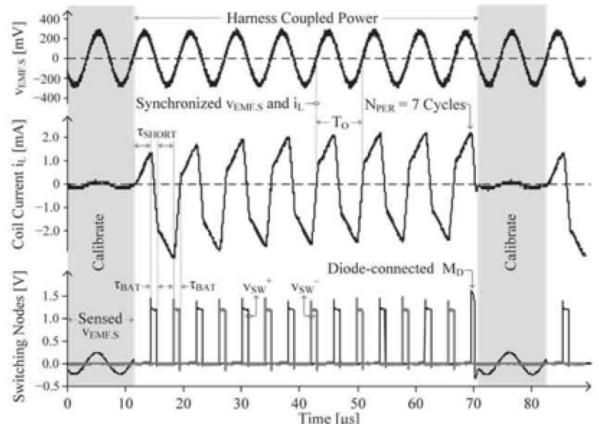
Brainstorm
Ideas



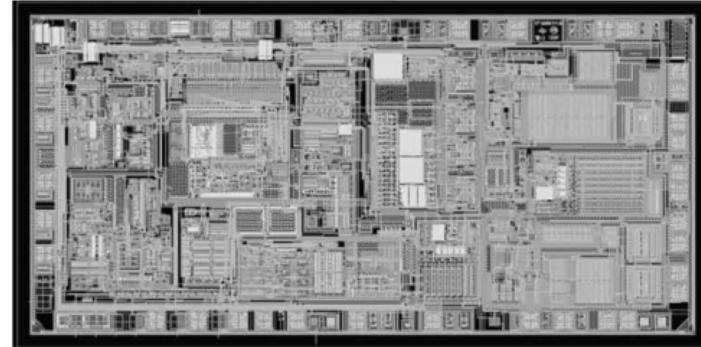
Design Block-Level System



Design Component-
Level Circuit

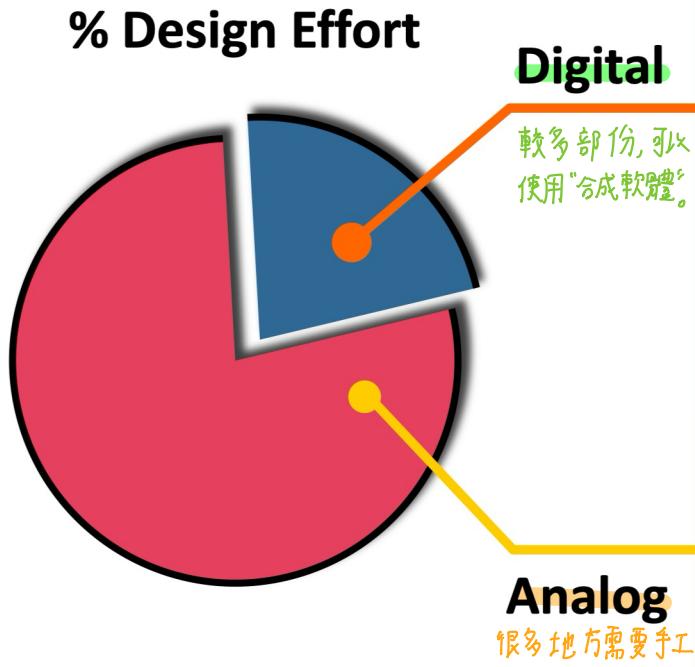


Verify Design (with Simulations)

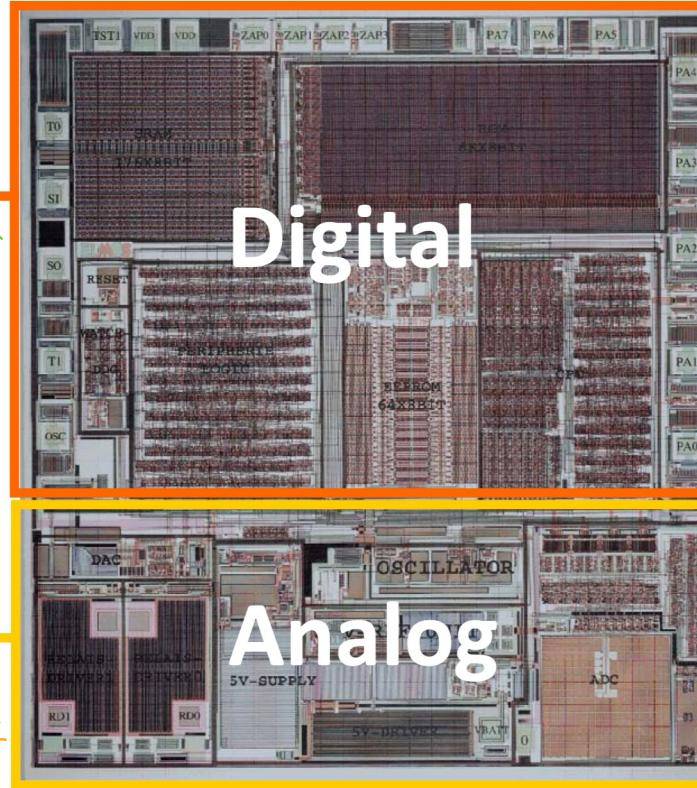


Design Physical Translation (Layout)

Large Design Effort for Analog ICs



Commercial Mixed Signal ASIC



→ 一個晶片內，大部份都是“數位電路”，“類比電路”不會佔很多。

[Rutenbar, ISPD'10]

Difficulties in Analog Layout Synthesis

- ❖ Optimization experts \neq Analog design experts \Rightarrow 不同領域
- ❖ Analog layout design == Artworks
- ❖ Analog layout synthesis == Aesthetic engineering

Hey, that looks
strange, right?



[Rutenbar, ISPD'10]

Analog Layout Synthesis Methodologies

❖ Layout generation from scratch

⇒ 根據 netlist, technology file, 用演算法產生 layout。

❖ Layout migration from a legacy layout

⇒ 從舊的 layout, 產生新的 layout, 且要合法。(不同 layout 可能用到不同製程)

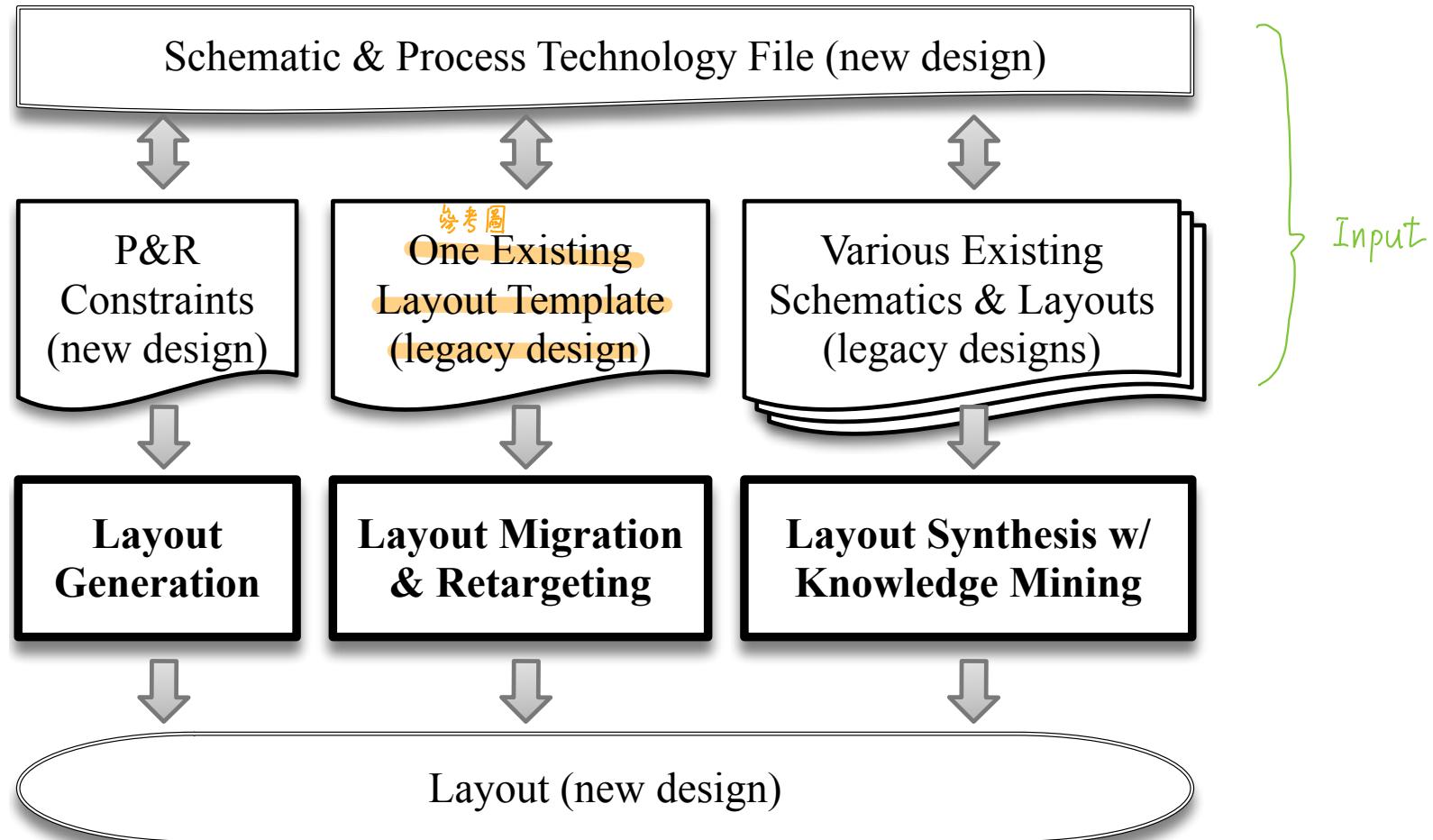
❖ Layout synthesis by learning from design repository

(machine learning)

(設計資料庫：存很多不同的 design)

Recent research development and new challenges in analog layout synthesis. ASP-DAC 2016: 617-622

Analog Layout Synthesis Methodologies



Analog Layout Generation

類似“數位的 standard cell”，但沒有固定的長寬。

❖ Device/Building block generation

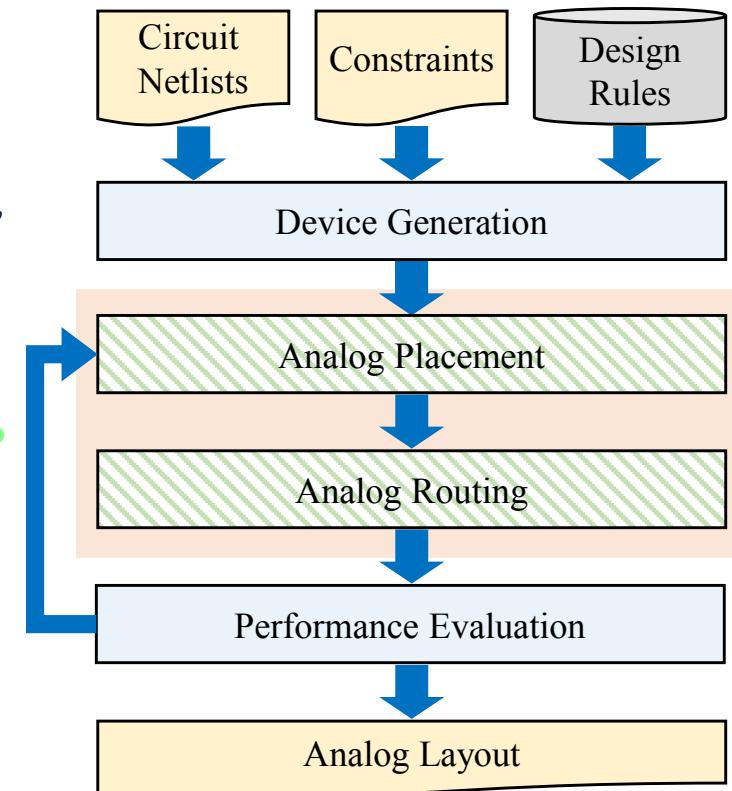
- ▶ Create physical layouts of each device or device groups with different variants, including some internal routing

❖ Analog placement

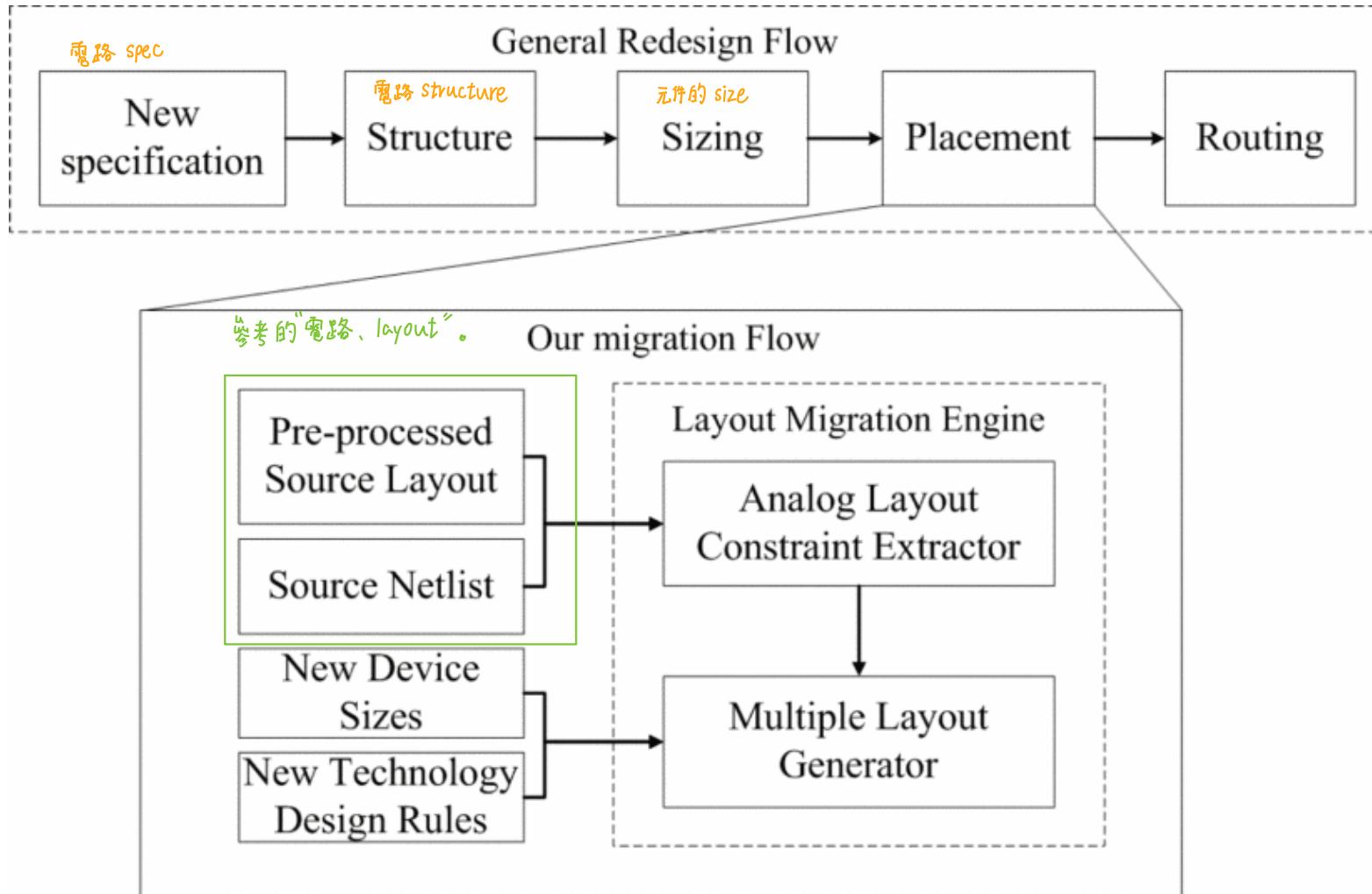
- ▶ Determine physical positions of devices according to the given layout area or aspect ratio, and placement constraints while minimizing the layout area and estimated wirelength.

❖ Analog routing

- ▶ Finalize the interconnections according to the routing constraints while minimizing the interconnecting wirelength

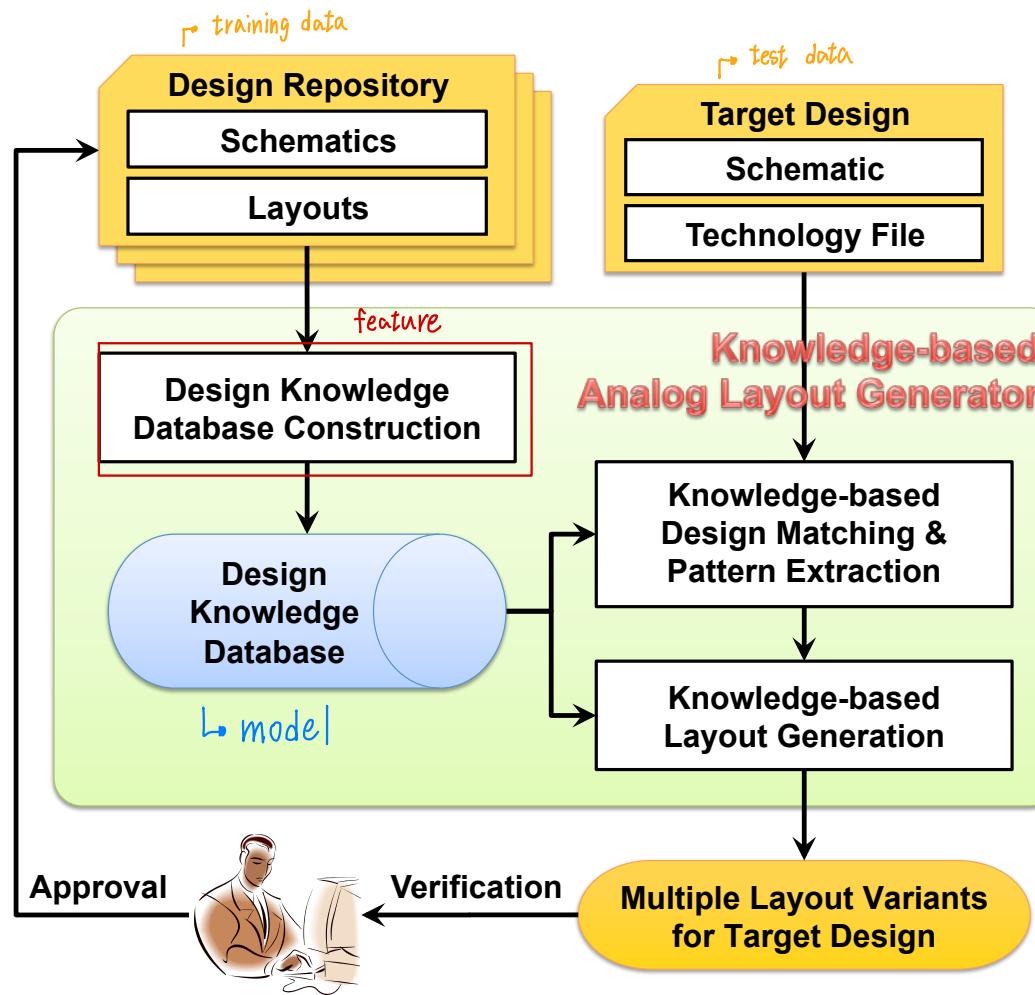


Analog Layout Migration



Fast analog layout prototyping for nanometer design migration. ICCAD 2011: 517-522

Analog Layout Synthesis w/ Knowledge Mining



A Novel Analog Physical Synthesis Methodology Integrating Existent Design Expertise. IEEE TCAD, 34(2): 199-212 (2015)

Ongoing Academic Research

- ❖ **BAG/BAG2 (Berkeley Analog Generator)**

UC Berkeley

- ❖ **ALIGN**

U of Minnesota

- ❖ **MAGICAL**

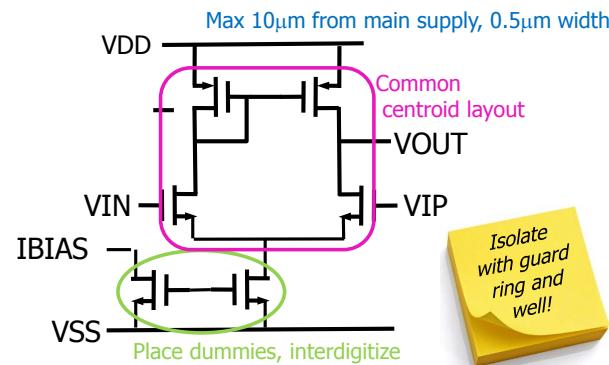
UT Austin



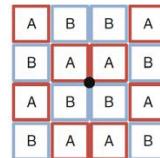
Fully automated digital AND analog layout!

Today

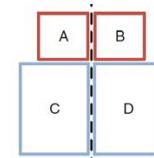
Designer provides manual constraints
to layout person (or tool)



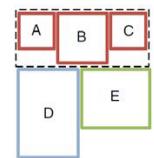
Common Vocabulary of Strategies



Centroid

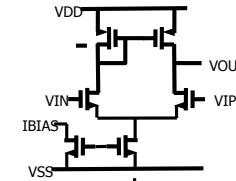


Mirroring

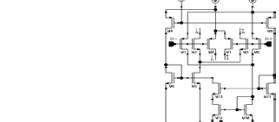


Isolation

IDEA



Circuit Classifier



Novelty:
Auto create
layout
constraints by
classifying
circuit patterns
and applying
strategies from
knowledge
database.

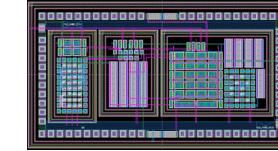
Model

Training

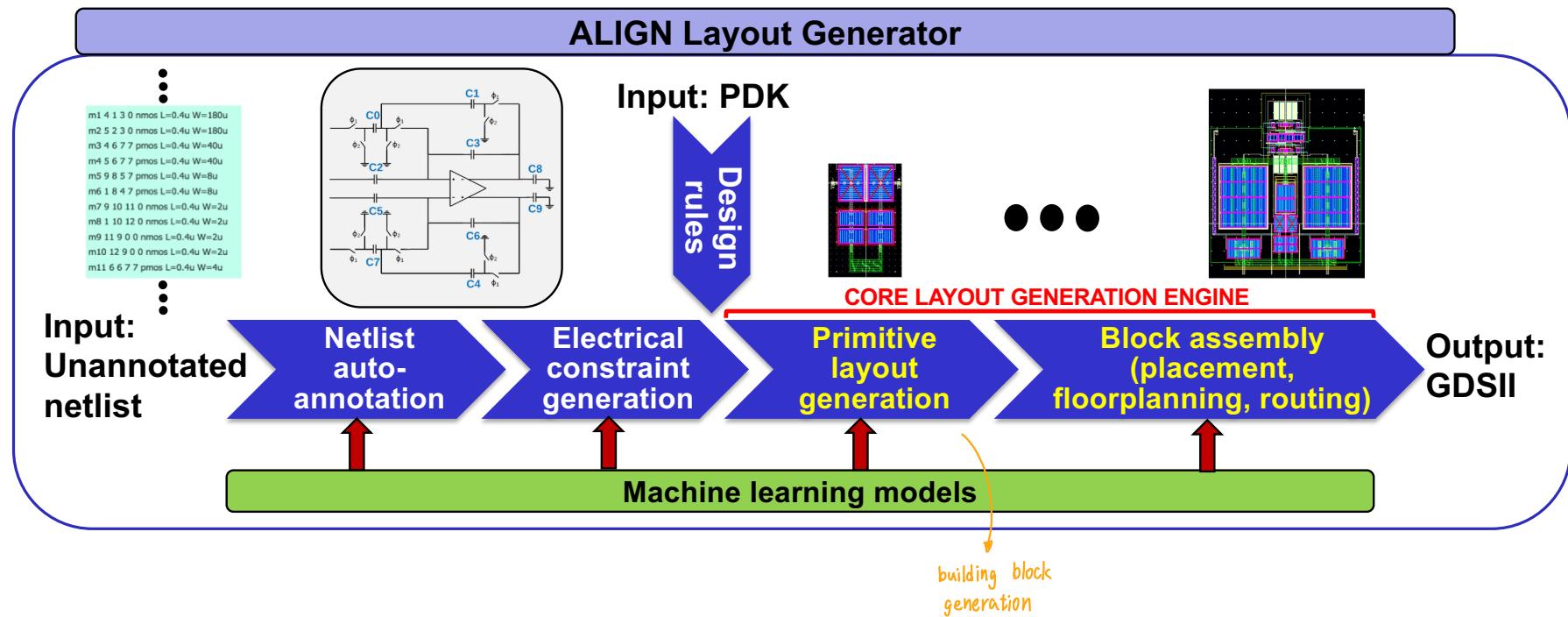
Assign Strategies

Auto-Placement

Auto-Routing

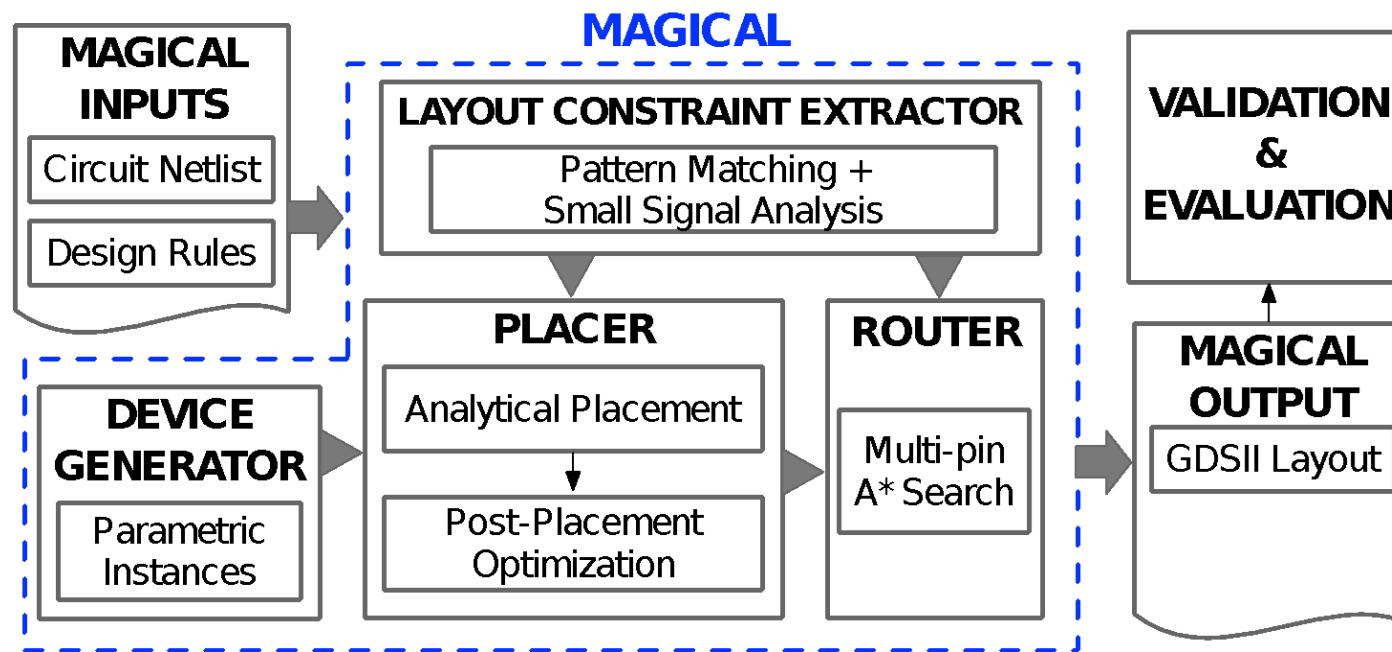


ALIGN (Analog Layout, Intelligently Generated from Netlists)



ALIGN: A System for Automating Analog Layout. IEEE Des. Test 38(2): 8-18 (2021)

MAGICAL – Fully Automated Analog IC Layout System from Netlist to GDSII



MAGICAL: An Open-Source Fully Automated Analog IC Layout System from Netlist to GDSII. IEEE Des. Test 38(2): 19-26 (2021)

Analog Placement Constraints

☞重要的 constraint

- ✓ ♦ Symmetry
- ✓ ♦ Symmetry island
- ✓ ♦ Common centroid ⇒ 在 building block generation，會做此 constraint
- ✓ ♦ Proximity, layout design hierarchy ⇒ 有些 block 會有“前後關係”。相同 type 的 block 要擺一起。
- ♦ Regularity ⇒ 先進製程
- ♦ Boundary ⇒ 在較大的“混合訊號電路”會考慮。（“數位的 block”和“類比的 block”可能會有連線）
- ♦ Current/signal path ⇒ 考慮訊號、電流的流向
- ♦ Thermal ⇒ 跟 power IC 有關的 類比電路，避免“熱效應影響”。
- ♦ ...

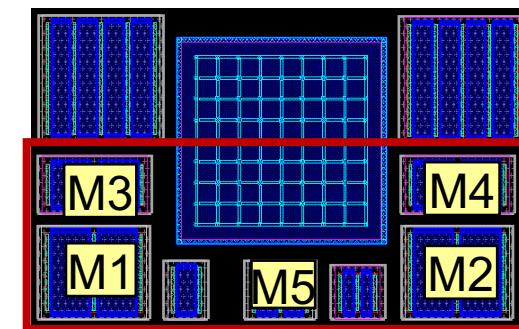
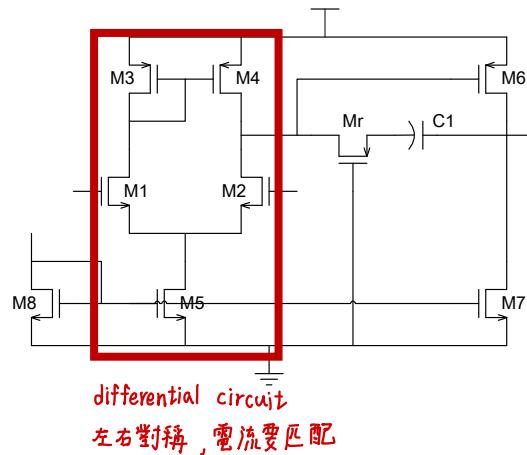
Symmetry v.s. Symmetry Island

2個元件對稱擺放

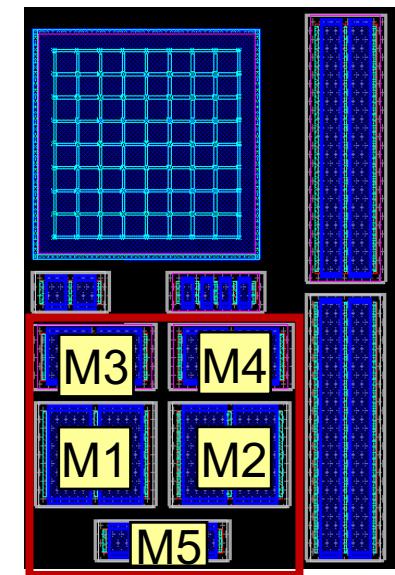
❖ Variance of the electrical property, P , of two identical devices [Pelgrom et al., JSSC'89]

$$\sigma^2(\Delta P) = \frac{A_P^2}{WL} + S_P^2 \underbrace{D_x^2}_{\text{目標：希望“變異小一點”。}} \Rightarrow \text{目標：希望“變異小一點”。}$$

- W, L – device size ↗ 重點
- D_x – distance between two devices



不好
symmetric, but not
symmetry-island placement
• 不在意“要對稱的元件”離對稱軸多遠。
⇒ 會被其他元件分開

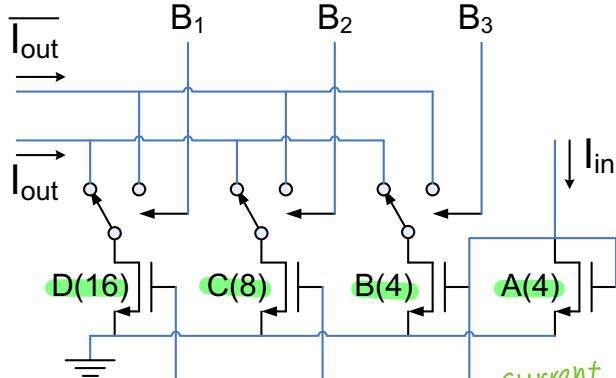


較好
symmetry-island placement
• 希望“要對稱的元件”離對稱軸愈近愈好。
⇒ 不會被其他元件分開

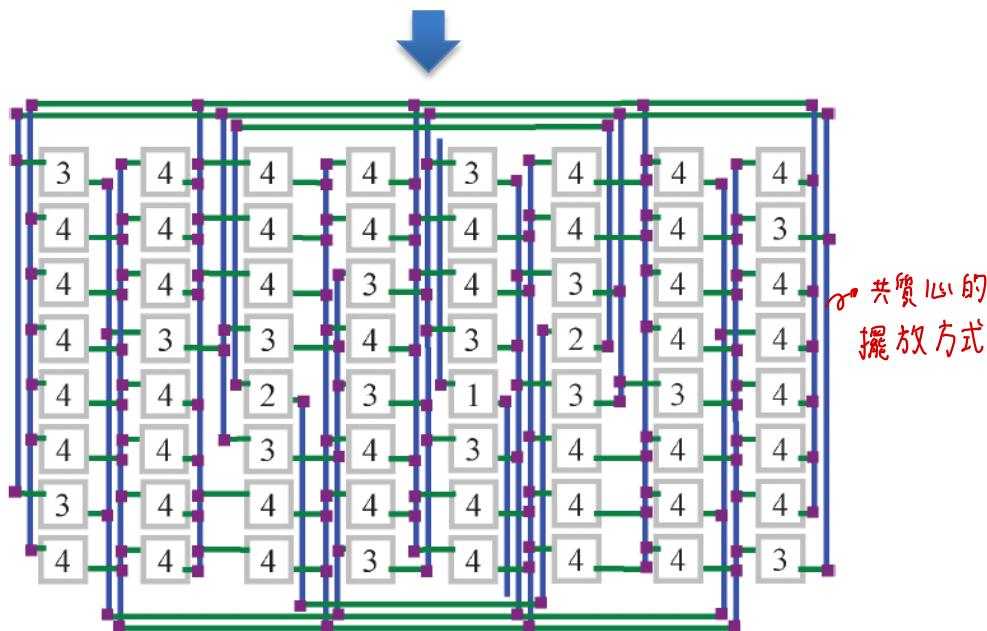
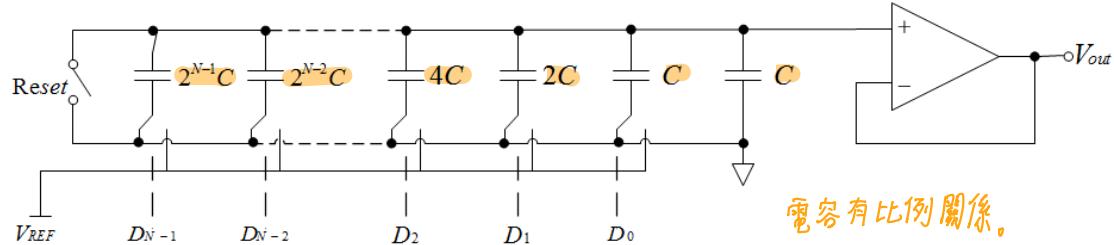
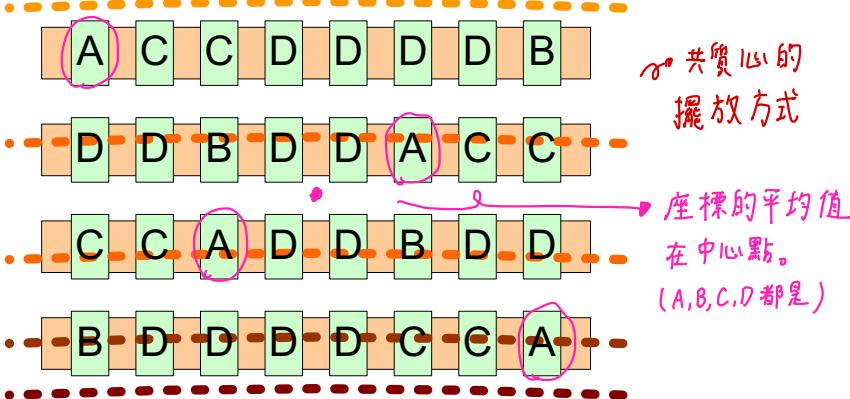
Analog Placement Based on Novel Symmetry-Island Formulation. DAC 2007: 465-470

Common Centroid

☞ 共質心，用在 current mirror 的電路架構。



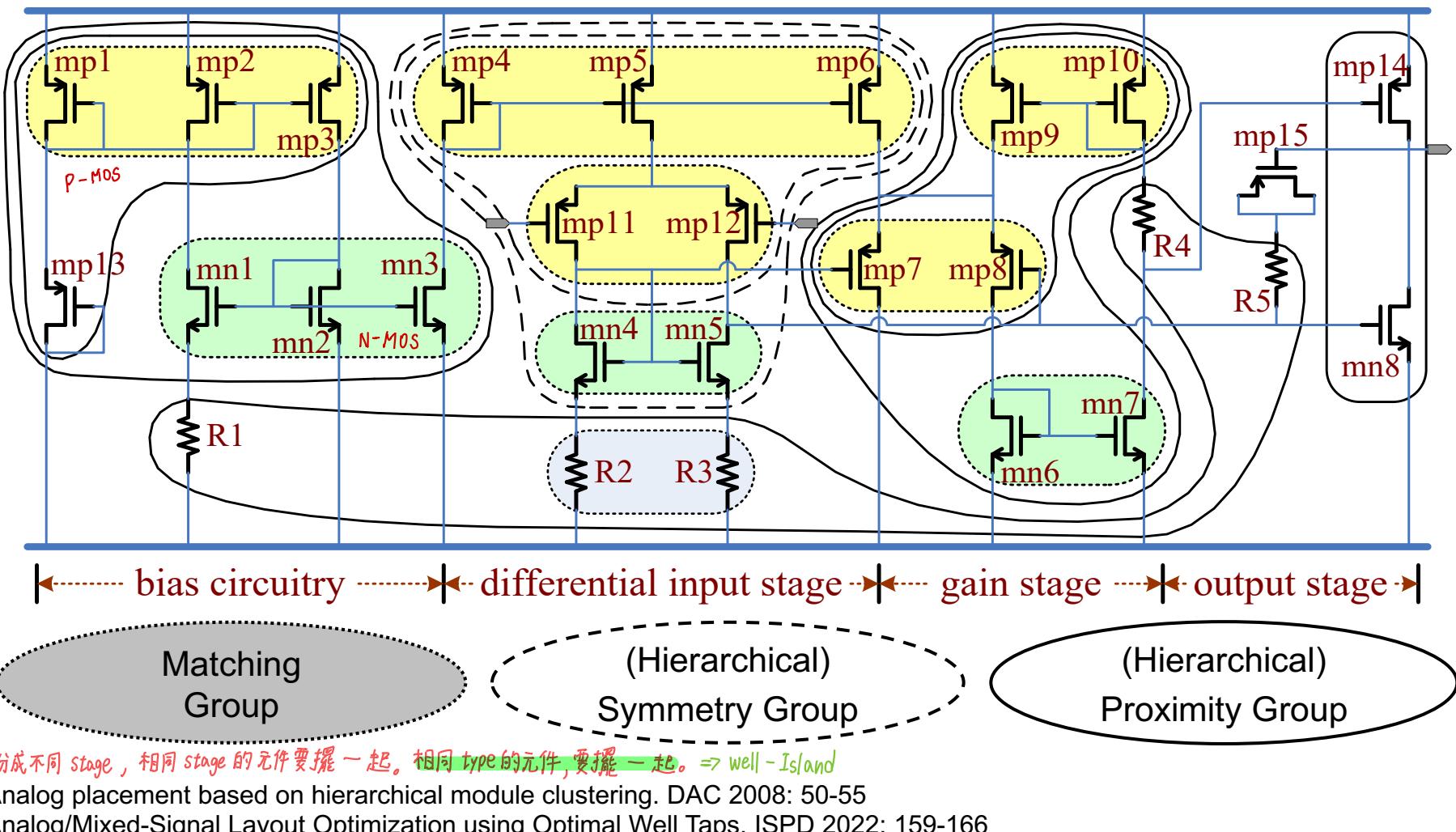
current mirror
(電晶體有比例關係)
⇒ 電流也是



Common-Centroid FinFET Placement Considering the Impact of Gate Misalignment. ISPD 2015: 25-31

Common-Centroid Capacitor Layout Generation Considering Device Matching and Parasitic Minimization. IEEE TCAD, 32(7): 991-1002 (2013)

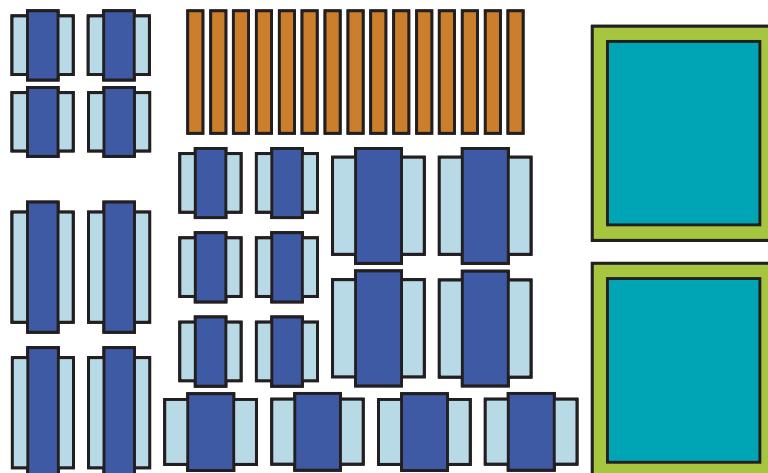
Proximity (Well-island) Constraints



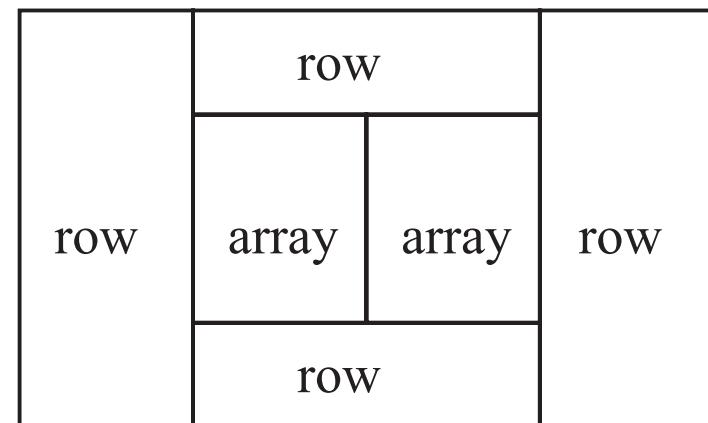
Regularity

→ 用於先進製程

- ❖ **Regular placement (rows or arrays) in all levels, including devices, cells, and blocks**
 - ❖ **Resulting in more compact layout, better routability, and less sensitivity to process variation**



Typical analog placement



Different regular placement structures

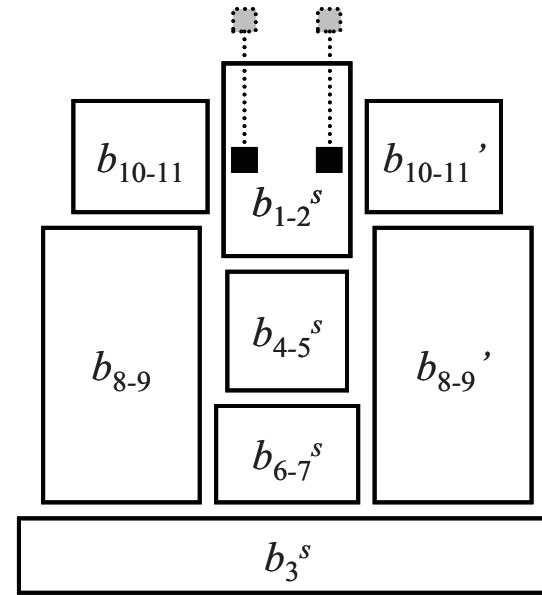
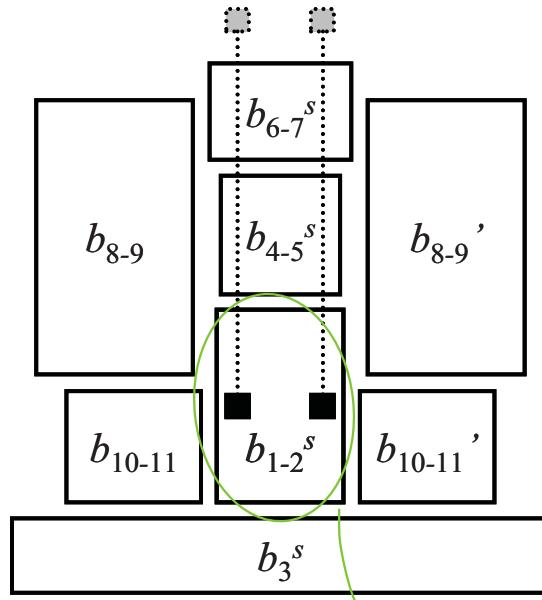
Regularity-oriented analog placement with diffusion sharing and well island generation. ASP-DAC 2010: 305-311

Boundary

- ❖ Devices connected to external signals must be on the boundary.

■ input node of the op-amp

.....□ connection to external signal source



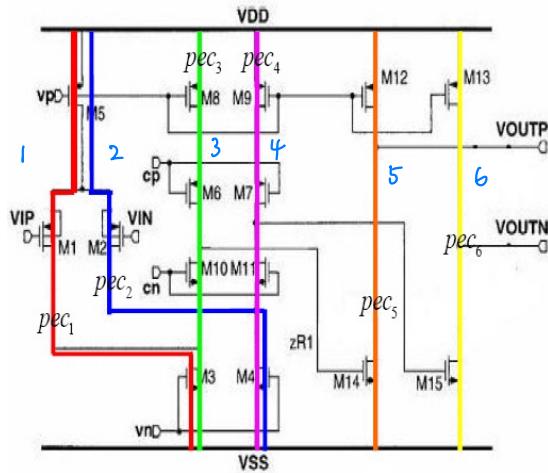
Device 有外部連線
Device 要放 boundary。

Performance-driven analog placement considering boundary constraint. DAC 2010: 292-297

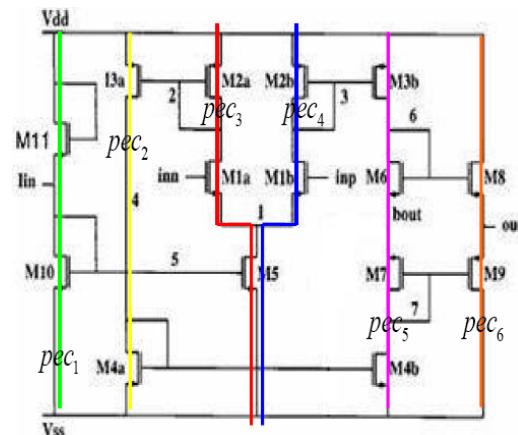
Current Path

~ 電流路徑

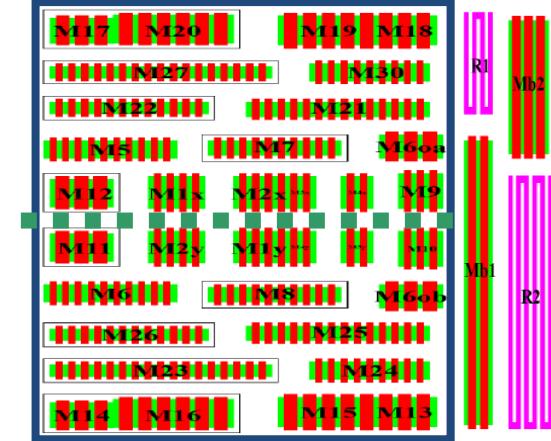
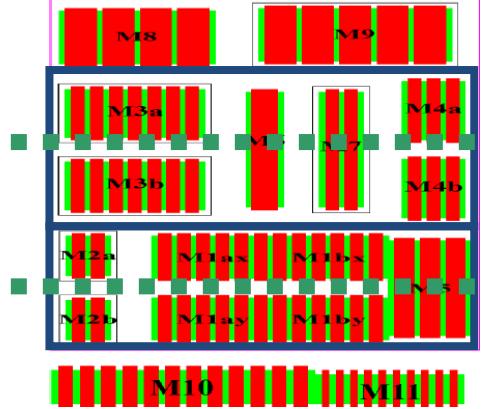
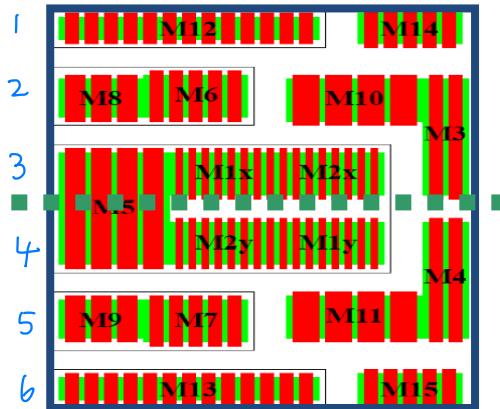
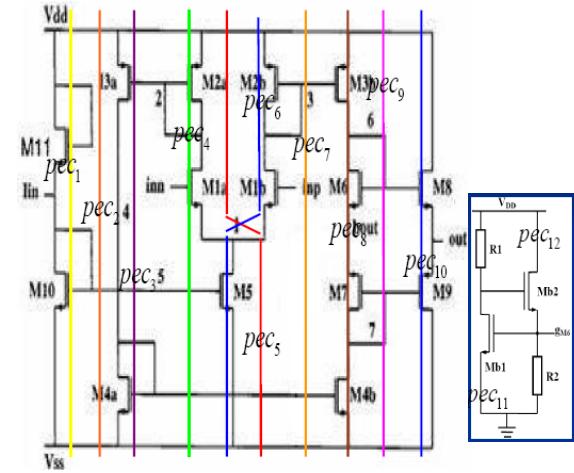
Two-stage amplifier



Comparator



Open-loop Opamp



Signal-path driven partition and placement for analog circuit. ASP-DAC 2006: 694-699

Monotonic Current Path

電流流向可以轉彎

❖ Straight current path

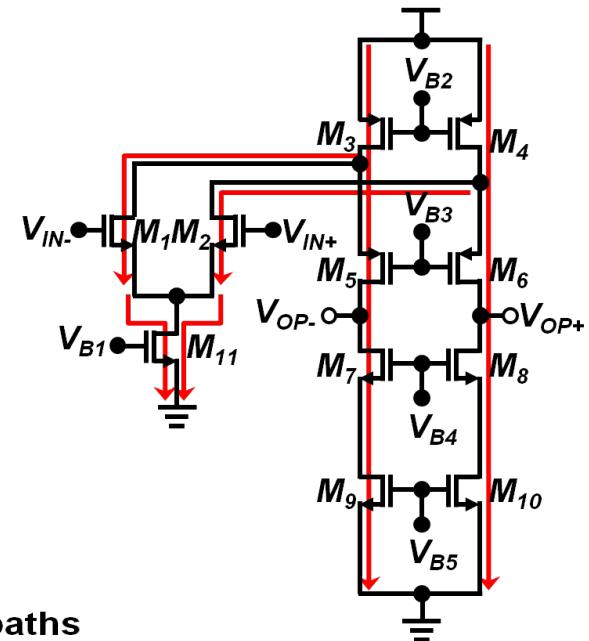
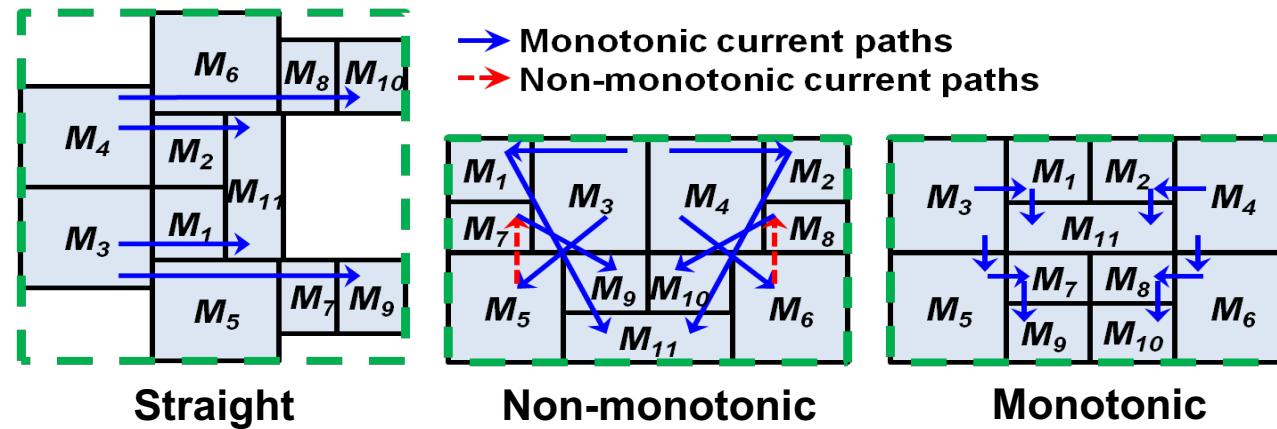
- ▶ Larger area

❖ Non-monotonic current path

- ▶ Larger routing-induced parasitic

❖ Monotonic current path

- ▶ Smaller area and routing-induced parasitic

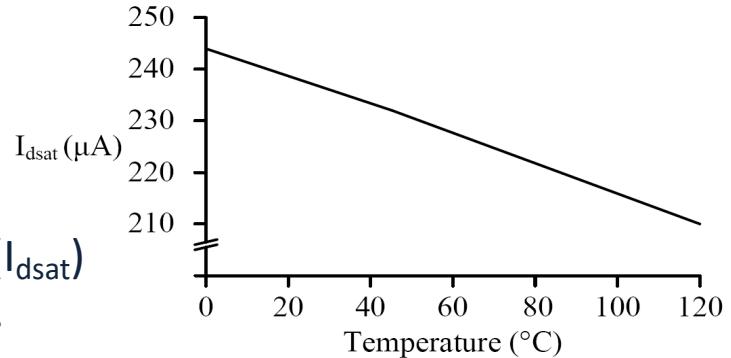


Performance-driven analog placement considering monotonic current paths. ICCAD 2012: 613-619

Thermal Effect & Device Mismatch

❖ Thermal effect

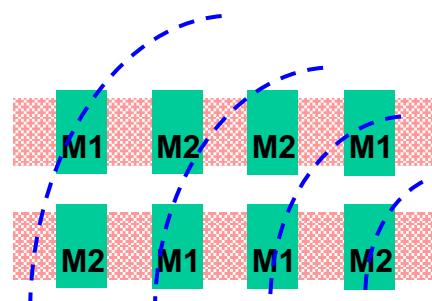
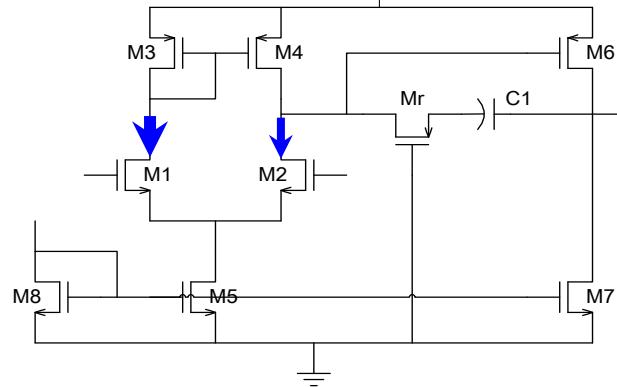
- ▶ Some electrical properties may vary with temperature.
 - E.g. The saturation current of MOSFET (I_{dsat}) decreases with increasing temperature.



Device mismatch

□ $\because M_1, M_2$ 離熱源的距離可能不同。

- ▶ E.g. M1 and M2 can be mismatched due to the thermal gradient.



Placement under thermal gradient

Thermal-Driven Analog Placement Considering Device Matching. IEEE TCAD, 30(3): 325-336 (2011)

Lecture 2

Analog Placement Method Considering Symmetry Constraints

Mark Po-Hung Lin

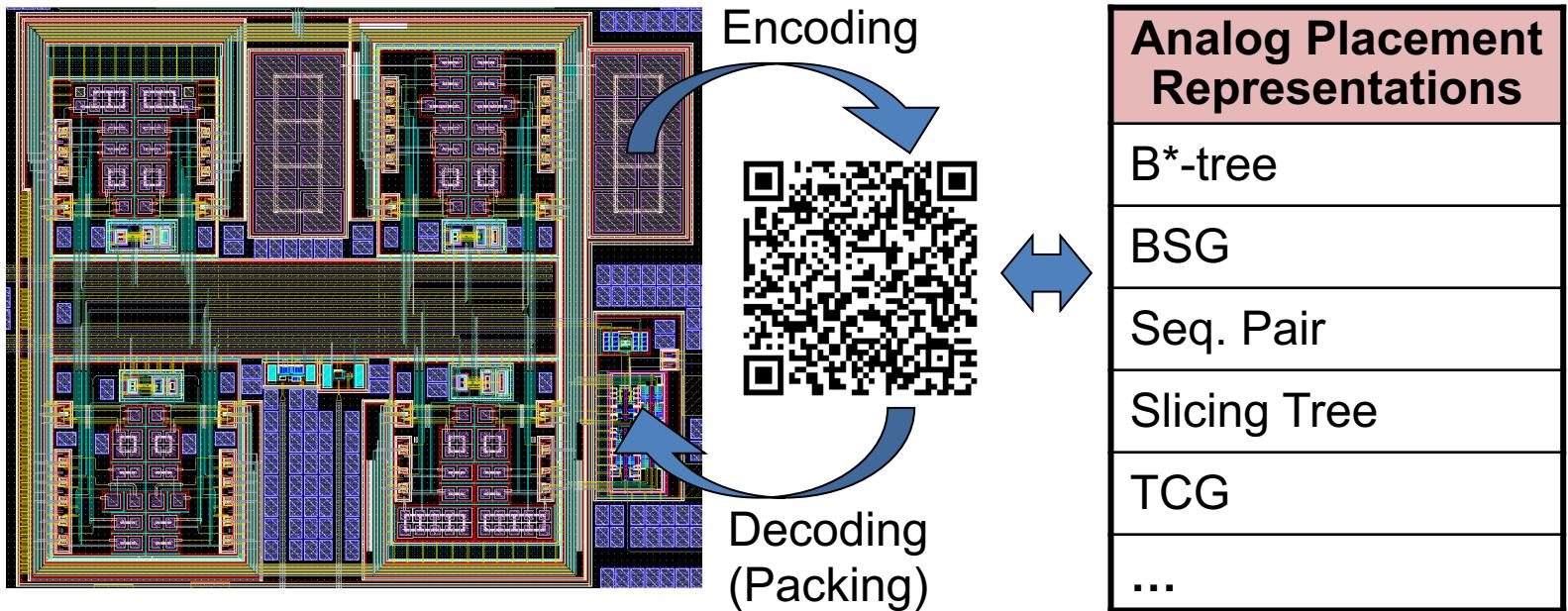
College of Artificial Intelligence

National Yang Ming Chiao Tung University

Basic Idea of Analog Placement

↳ 用到很多 Floorplan 的表示法。

- ❖ Map placement solutions to some kind of **representations**
- ❖ Embed various layout constraints in the **representation**
- ❖ Find out the best configurations in the **representation** in terms of **placement area and interconnecting wirelength**



Analog Placement Representations

❖ **Absolute representation** ⇒ “表示法”內，有絕對座標。

- ▶ Each device/block is associated with an absolute coordinate on a gridless plane.
- ▶ No packing is needed, but overlapping may happen.
- ▶ A postprocessing step is required to eliminate overlapping.

❖ **Topological representations** ⇒ “表示法”內，沒有絕對座標。Floorplan 的技術。

- ▶ Topological relationships are defined in the representation without absolute coordinates.
- ▶ A packing procedure is required to convert the representation into absolute coordinates, which guarantee non-overlapping.
- ▶ Examples: Slicing tree, B*-tree, Sequence Pair (SP)

Analog Placement Representations

❖ B*-tree and Sequence Pair

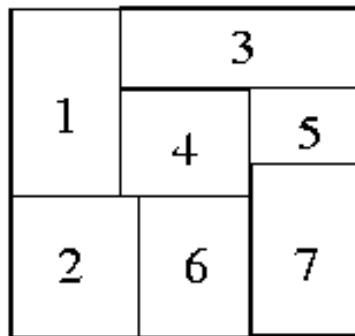
- ▶ Most popular analog placement representations
- ▶ Capable of handling various analog placement constraints

Constraint \ Representation	B*-tree	CBL	Sequence Pair	Slicing Tree	TCG
Symmetry	[16]–[25]	[26]	[27]–[30],	[31], [32]	[33], [34]
Symmetry Island	[17]–[25]			[31], [32]	[34]
Common Centroid	[16]	[30], [35]	[29]		
Proximity	[16], [18], [21], [24]		[27], [30]		
Regularity	[23]		[40], [41]		
Boundary	[20], [24]		[27], [30]		
Current/Signal Path	[25]			[31], [32]	
Thermal	[19], [22]	[26]			

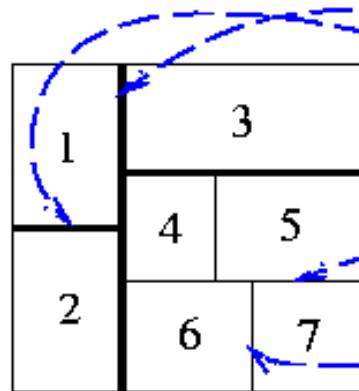
Recent research development and new challenges in analog layout synthesis. ASP-DAC 2016: 617-622

Slicing Tree

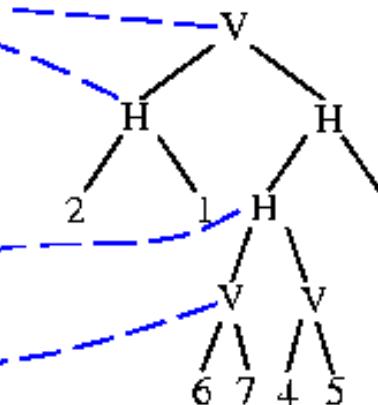
- ❖ **Slicing structure:** a rectangular dissection that can be obtained by repetitively subdividing rectangles horizontally or vertically.
- ❖ **Slicing tree:** A binary tree, where each internal node represents a vertical cut line or horizontal cut line, and each leaf a basic rectangle.
- ❖ **Skewed slicing tree:** One in which no node and its **right child** are the same.



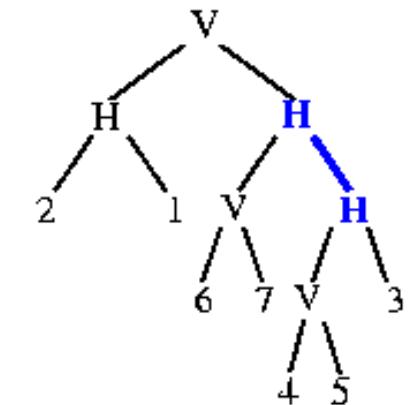
Non-slicing floorplan



Slicing floorplan



A slicing tree (skewed)



Another slicing tree
(non-skewed)

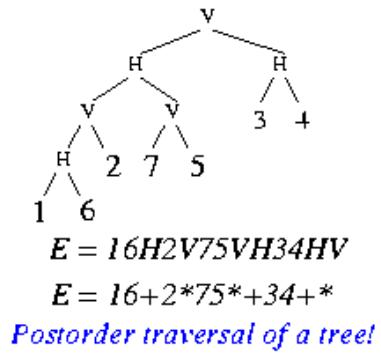
Automatic floorplan design. DAC 1982: 261-267

Polish Expression for Slicing Tree

- An expression $E = e_1 \ e_2 \dots \ e_{2n-1}$, where $e_i \in \{1, 2, \dots, n, H, V\}$, $1 \leq i \leq 2n-1$, is a **Polish expression** of length $2n-1$ iff
 - every operand j , $1 \leq j \leq n$, appears exactly once in E ;
 - (the balloting property) for every subexpression $E_i = e_1 \dots e_i$, $1 \leq i \leq 2n-1$, # operands > # operators.

1 6 H 3 5 V 2 H V 7 4 H V
of operands = 4 = 7
of operators = 2 = 5

7	5	4
6	2	3
1		



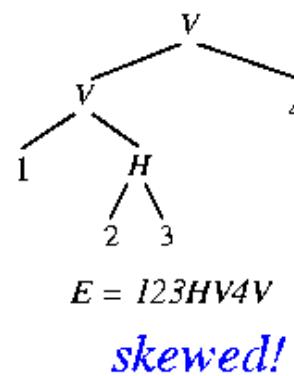
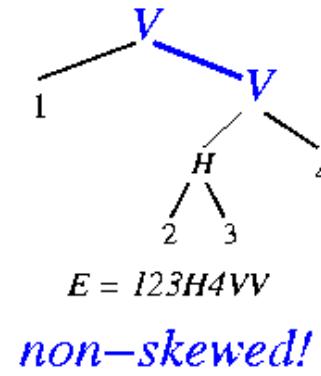
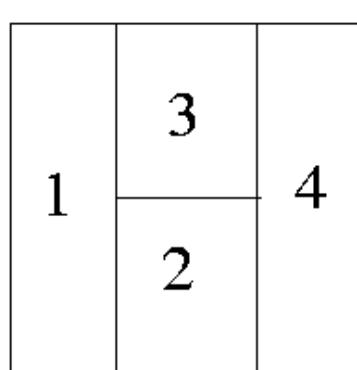
Postorder traversal of a tree!

- Polish expression \leftrightarrow Postorder traversal.
- ijH : rectangle i on bottom of j ; ijV : rectangle i on the left of j .

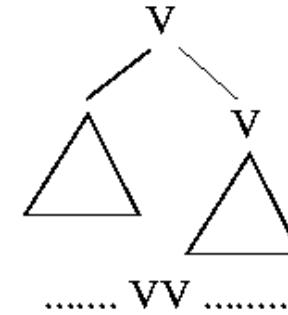
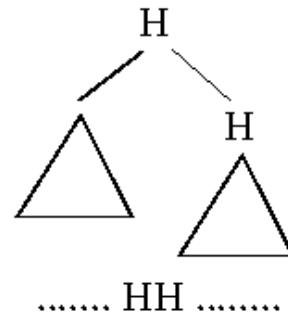
A new algorithm for floorplan design. DAC 1986: 101-107

Redundant Representations

- ❖ Question: How to eliminate ambiguous representation?



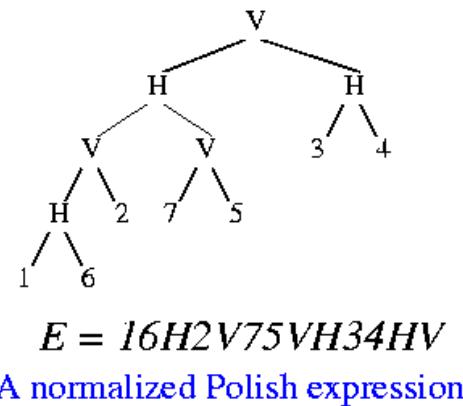
Non-skewed cases



Normalized Polish Expression

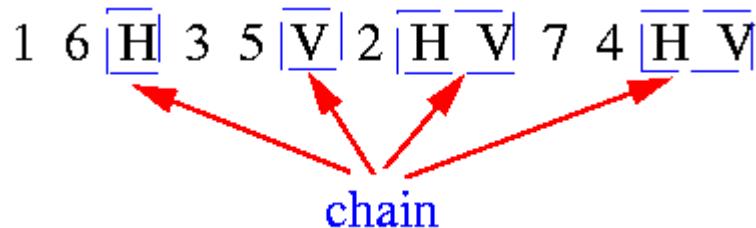
- ❖ A Polish expression $E = e_1 e_2 \dots e_{2n-1}$ is called **normalized** iff E has no consecutive operators of the same type (H or V).
- ❖ Given a **normalized Polish expression**, we can construct a **unique rectangular slicing structure**.

7	5	4
6	2	3
1		



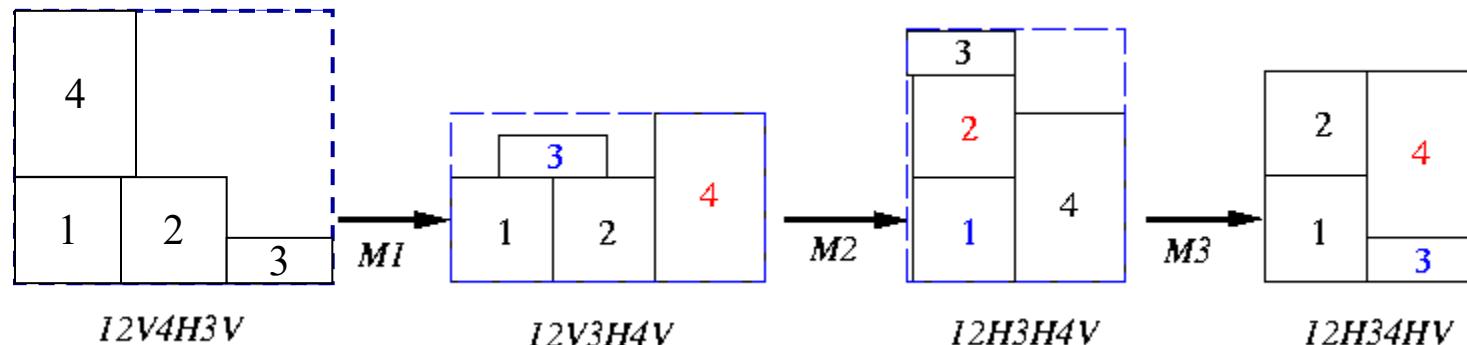
Neighborhood Structure

- ❖ **Chain:** $HVHVH \dots$ or $VHVVH \dots$



- ❖ **Adjacent:** 1 and 6 are adjacent operands; 2 and 7 are adjacent operands; 5 and V are adjacent operand and operator.
- ❖ **3 types of moves:**
 - ▶ **M1 (Operand Swap):** Swap two adjacent operands.
 - ▶ **M2 (Chain Invert):** Complement some chain ($V = \overline{H}$, $H = \overline{V}$).
 - ▶ **M3 (Operator/Operand Swap):** Swap two adjacent operand and operator.

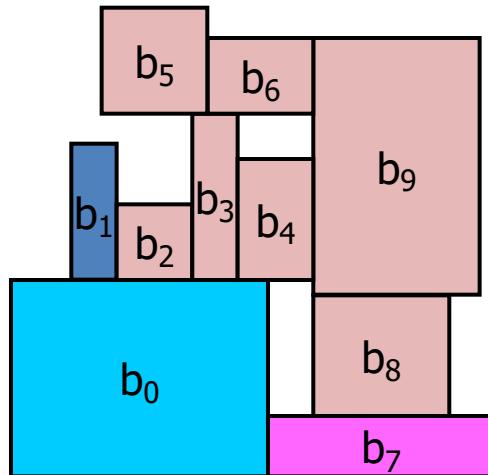
Effects of Perturbation



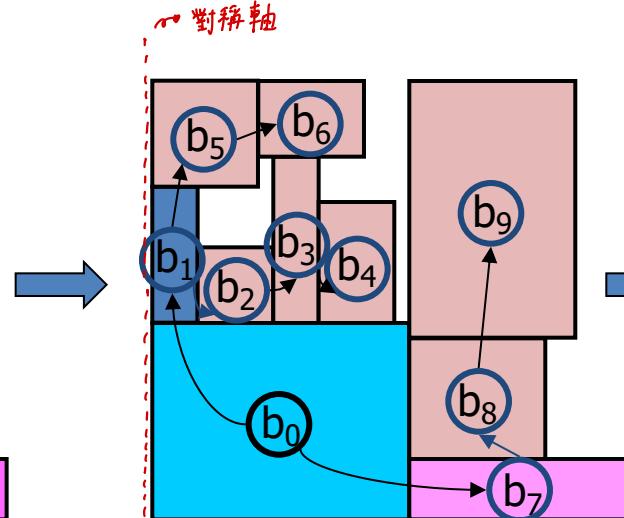
- ❖ **Question: The balloting property holds during the moves?**
 - ▶ $M1$ and $M2$ moves are OK.
 - ▶ **Check the $M3$ moves!** Reject “illegal” $M3$ moves.
- ❖ **Check $M3$ moves:** Assume that $M3$ swaps the operand e_i with the operator e_{i+1} , $1 \leq i \leq k-1$. Then, the swap will not violate the balloting property iff $2N_{i+1} < i$.
 - ▶ N_k : # of operators in the Polish expression $E = e_1 e_2 \dots e_k$, $1 \leq k \leq 2n-1$

B*-Tree

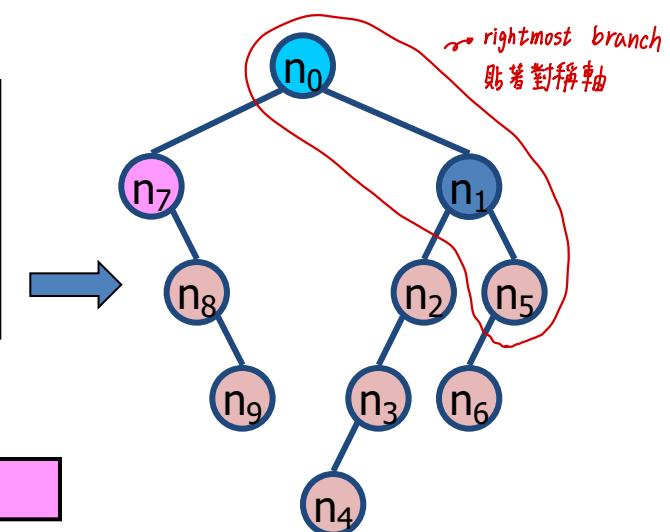
- ❖ Compact modules to left and bottom.
- ❖ Construct an ordered binary tree (B*-tree).
 - Left child: the lowest, adjacent block on the right ($x_j = x_i + w_i$).
 - Right child: the first block above, with the same x-coordinate ($x_j = x_i$).



A non-slicing floorplan



Compact to left and down

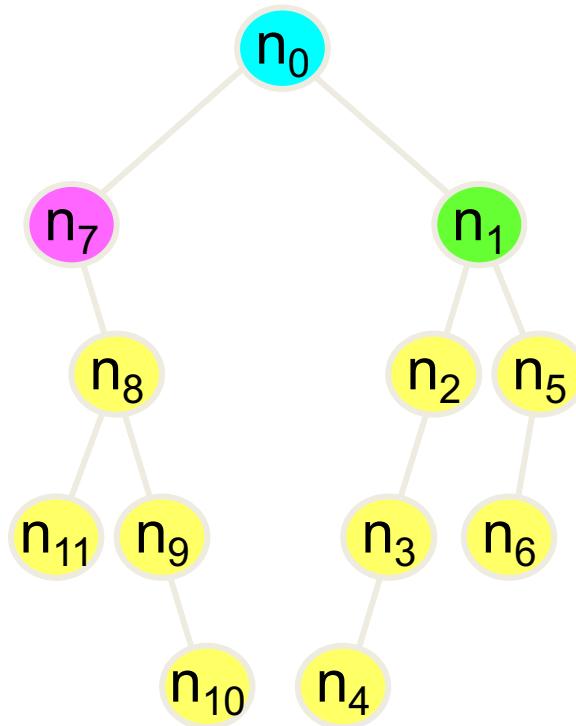
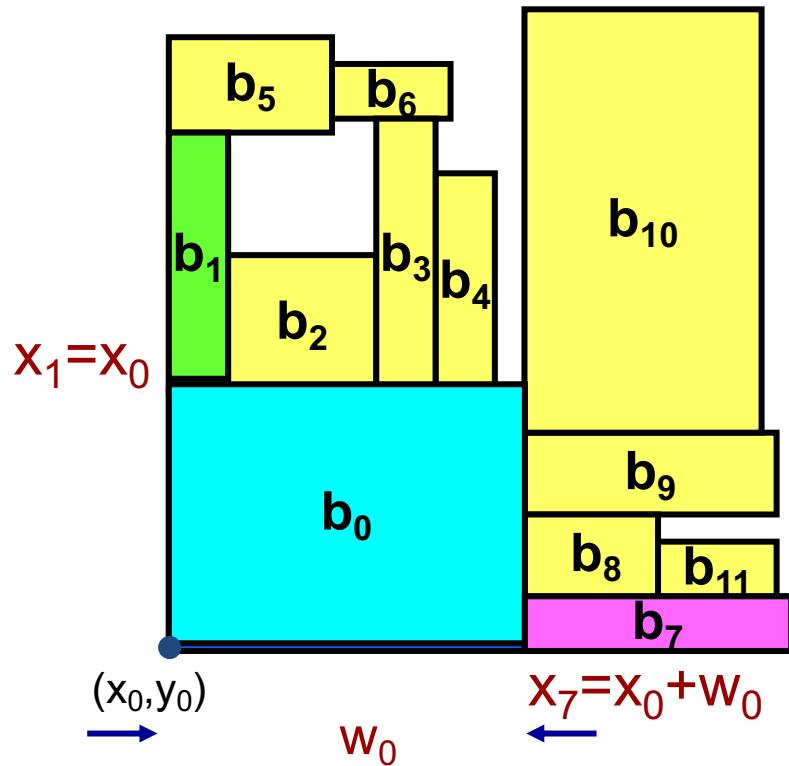


B*-tree

B*-Trees: a new representation for non-slicing floorplans. DAC 2000: 458-463

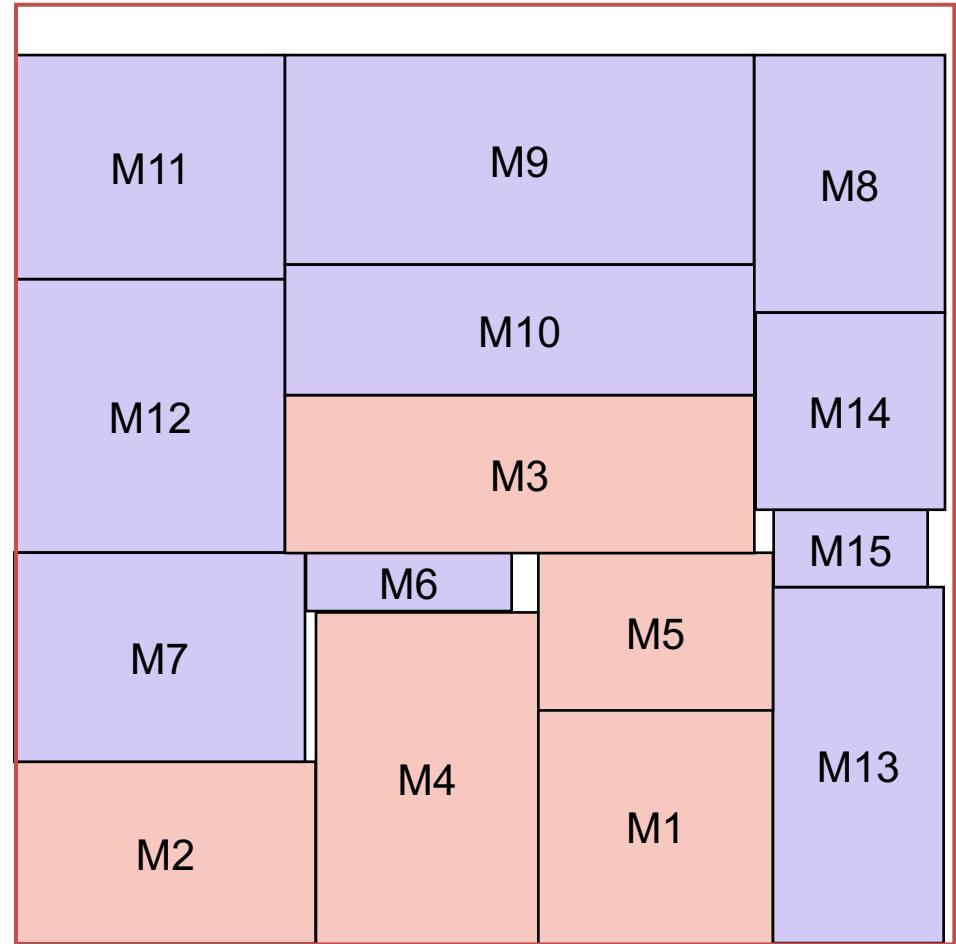
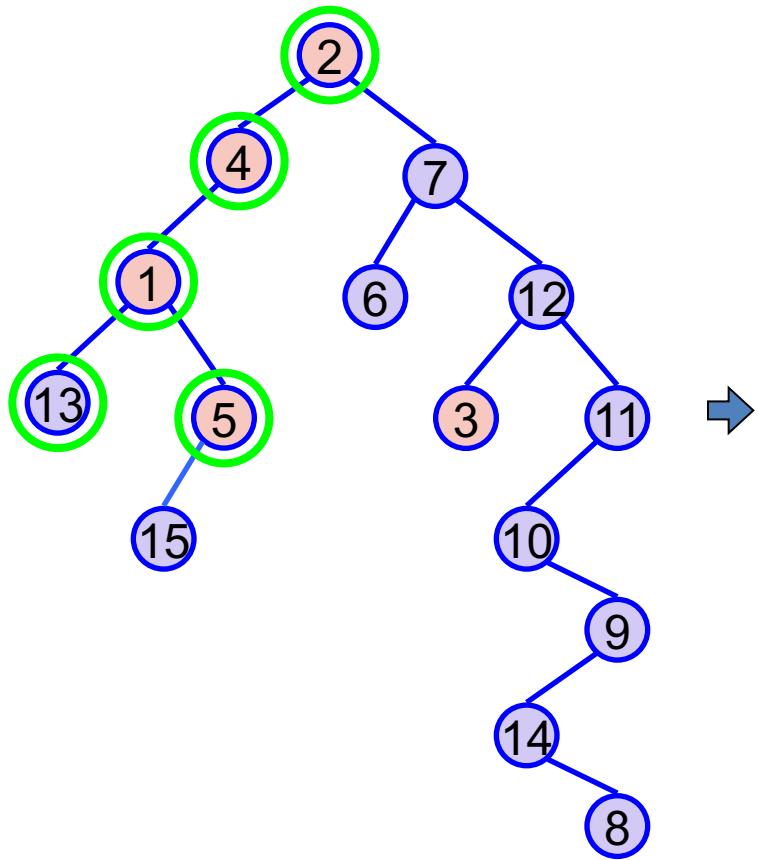
B*-Tree Packing

- ❖ **x-coordinates can be determined by the tree structure.**
 - ▶ Left child: the lowest, adjacent block on the right ($x_j = x_i + w_i$).
 - ▶ Right child: the first block above, with the same x-coordinate ($x_j = x_i$).
- ❖ **y-coordinates?**



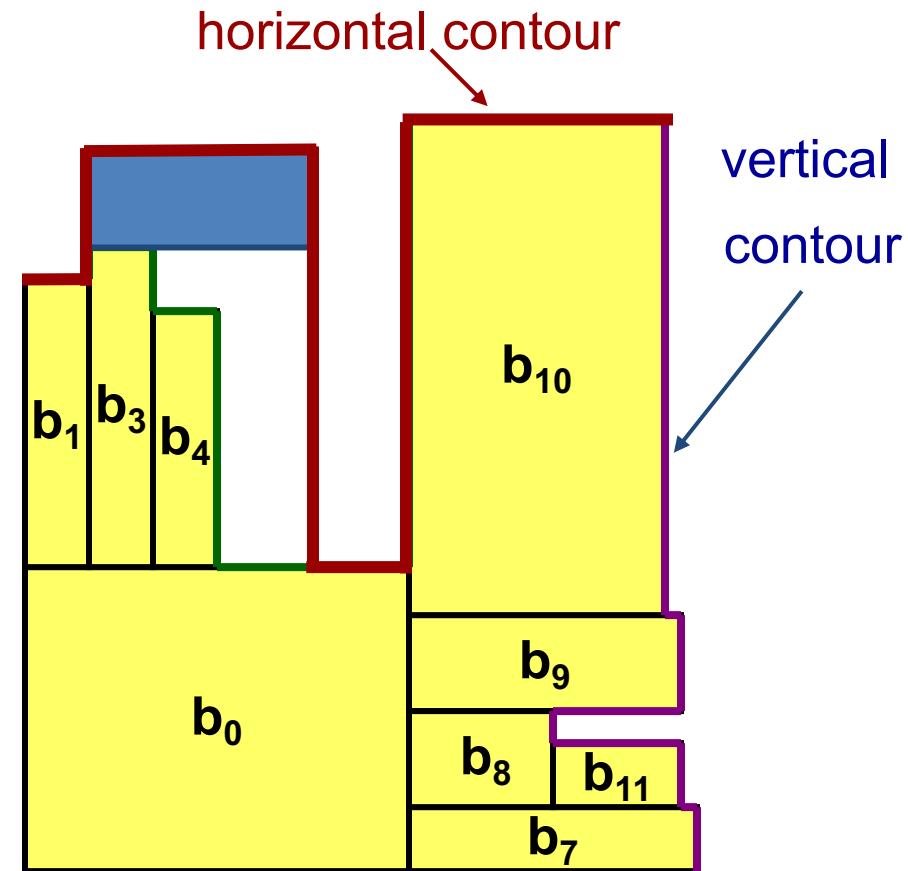
B*-Tree Packing

❖ Preorder traversal



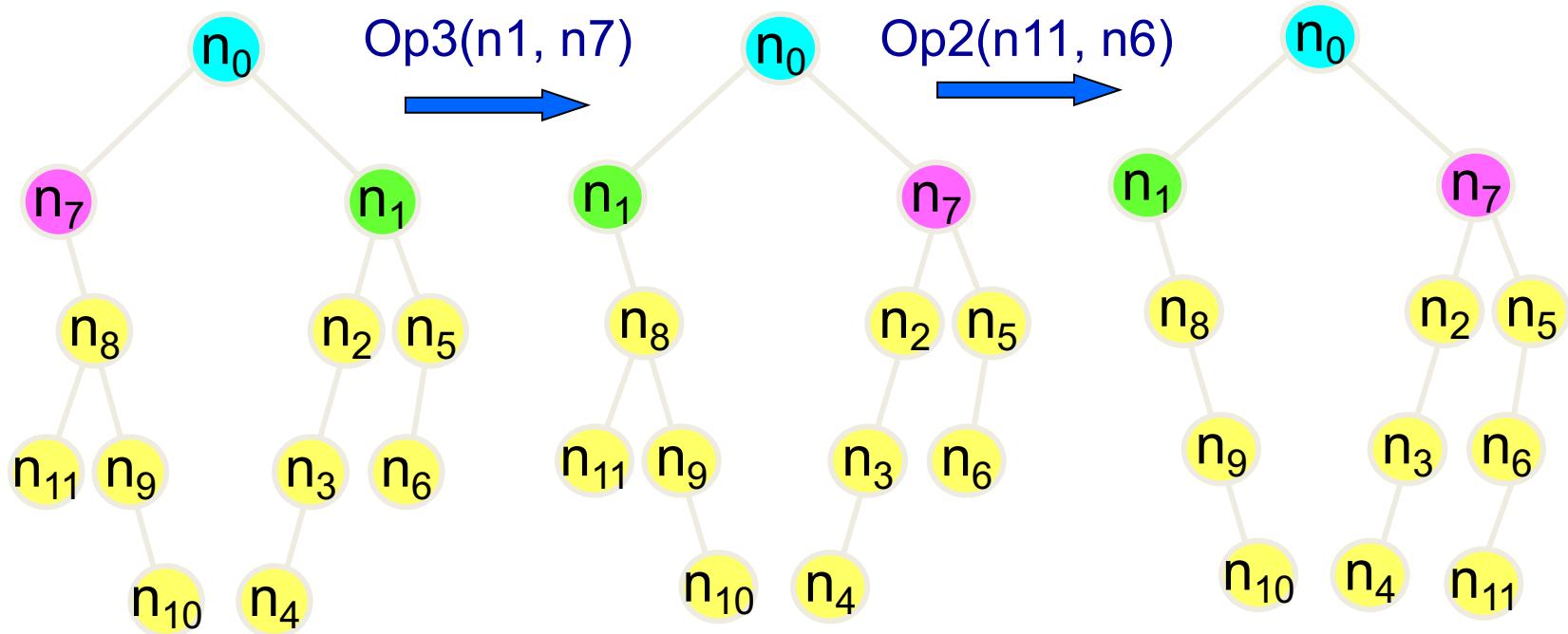
Computing y-coordinates

- ❖ **Horizontal contour:**
Use a doubly linked list to record the current maximum y-coordinate for each x-range
- ❖ Reduce the complexity of computing a y-coordinate to amortized O(1) time.



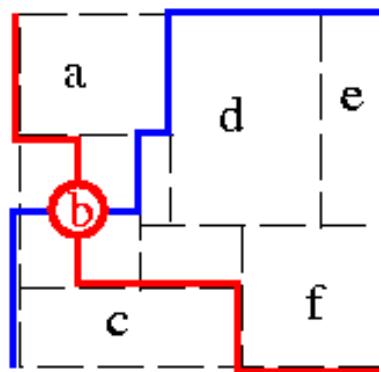
B*-Tree Perturbation

- ❖ Op1: rotate a macro
- ❖ Op2: delete & insert } analog placement 只能用這 2 個操作。
- ❖ Op3: swap 2 nodes
- ❖ Op4: resize a soft macro

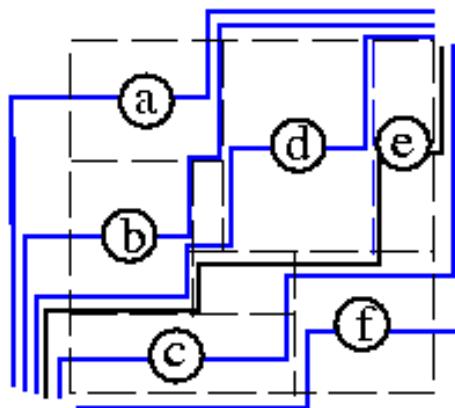


Sequence Pair (SP)

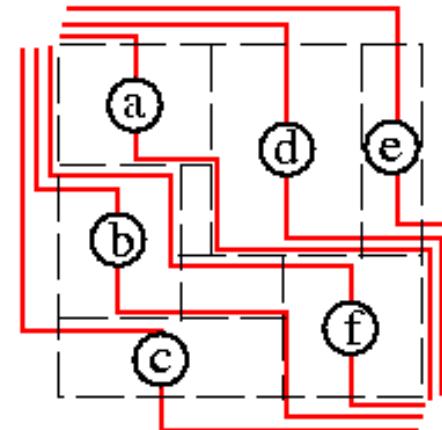
- ❖ Represent a packing by a pair of module-name sequences (e.g., $(abdecf, cbfade)$).
 - ▶ Size of the solution space: $(n!)^2$
- ❖ Positive locus Γ_+ : Union of right-up locus and left-down locus.
- ❖ Negative locus Γ_- : Union of up-left locus and down-right locus.



Loci of module b



Positive loci: abdecf

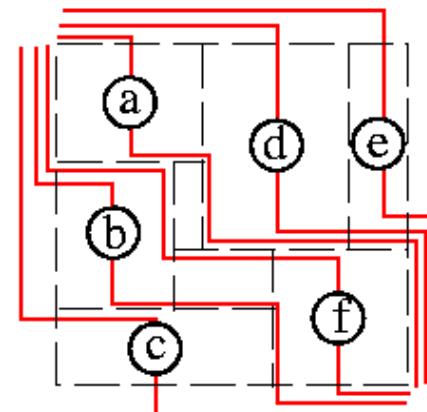
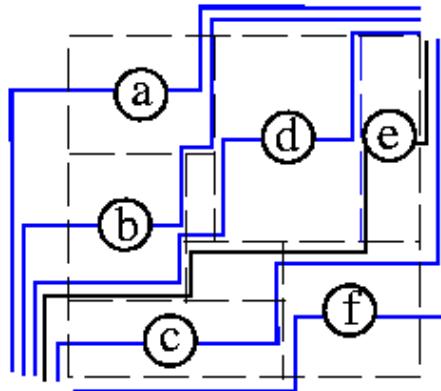
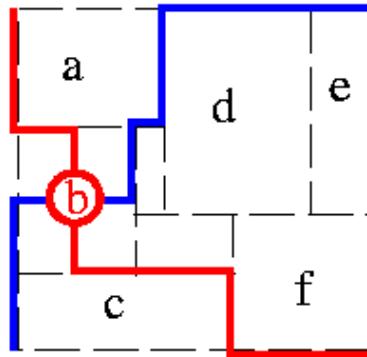


Negative loci: cbfade

Rectangle-packing-based module placement. ICCAD 1995: 472-479

Geometrical Information

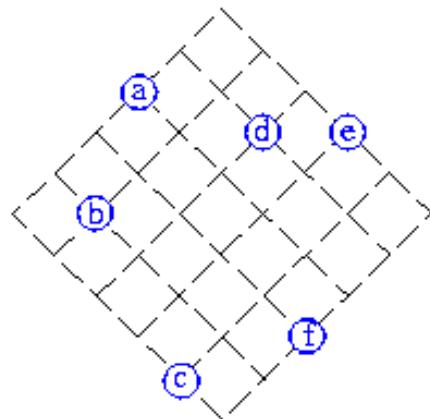
- ❖ No pair of positive (negative) loci cross each other,
i.e., loci are linearly ordered.
- ❖ SP uses two sequences (Γ_+, Γ_-) to represent a floorplan.
 - ▶ H-constraint: $(\dots a \dots b \dots, \dots a \dots b \dots)$ iff a is on the left of b
 - ▶ V-constraint: $(\dots a \dots b \dots, \dots b \dots a \dots)$ iff b is below a



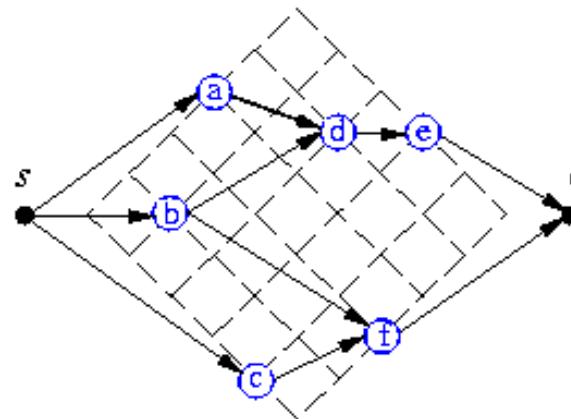
$$(\Gamma_+, \Gamma_-) = (\textcolor{red}{abdecf}, \textcolor{green}{cbfade})$$

Packing for Sequence Pair

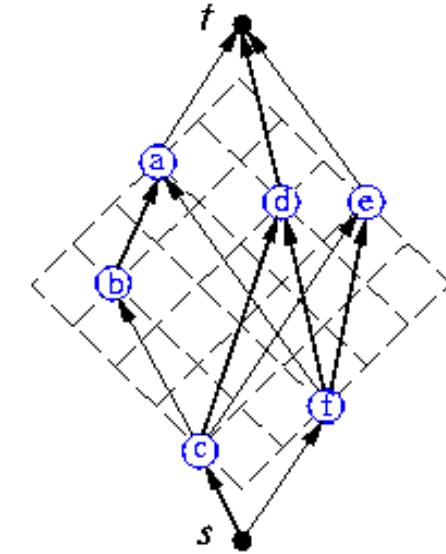
- ❖ $O(n^2)$ time by applying a longest path algorithm on a vertex-weighted directed acyclic graph
- ❖ $O(n \lg \lg n)$ time by computing the longest common subsequence of the two sequences



Packing for sequence pair:
(abdecf, cbfude)



Horizontal constraint graph
(Transitive edges are not shown)



Vertical constraint graph
(Transitive edges are not shown)

Fast evaluation of sequence pair in block placement by longest common subsequence computation. IEEE TCAD, 20(12): 1406-1413 (2001)

Symmetric-Feasible B*-tree

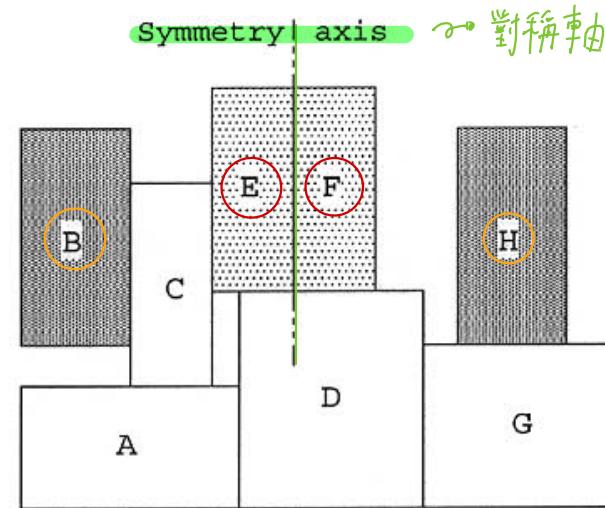
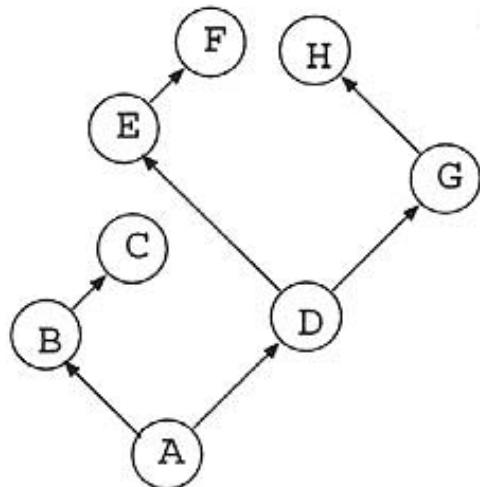
↳ Block

- ❖ (A, sym(A)) and (B, sym(B)) are **symmetry pairs**. \Rightarrow 有一個共用的對稱軸。

- ▶ The corresponding symmetric-feasible B*-tree must satisfy the property

$$A \xleftarrow{\text{inorder}} B \iff \text{sym}(B) \xleftarrow{\text{preorder}} \text{sym}(A)$$

A在B前面。
sym(B)在sym(A)前面。



Symmetric-Feasible Sequence-Pair

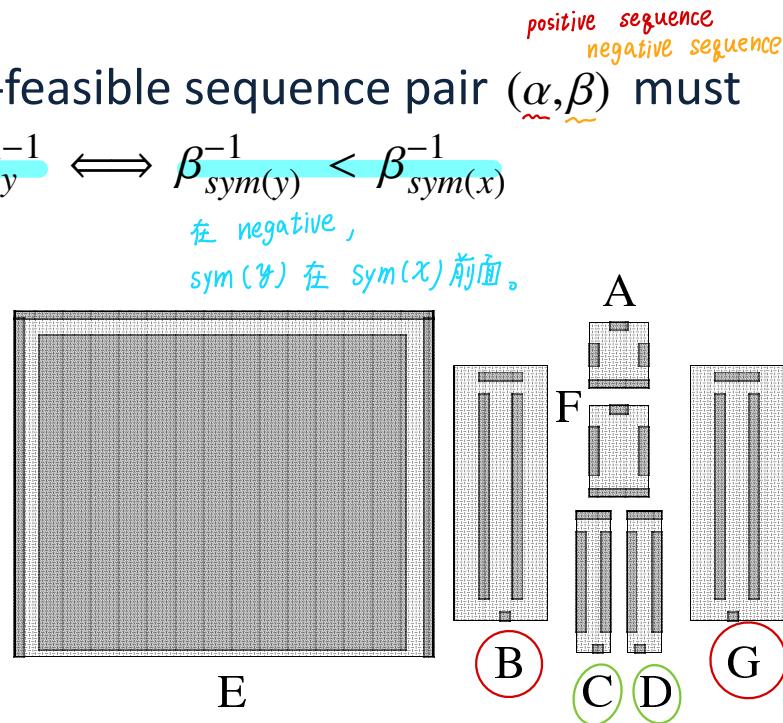
❖ (x, y) is a symmetry pair.

- ▶ The corresponding symmetric-feasible sequence pair (α, β) must satisfy the property $\alpha_x^{-1} < \alpha_y^{-1} \iff \beta_{\text{sym}(y)}^{-1} < \beta_{\text{sym}(x)}^{-1}$

在 positive,
 x 在 y 前面

在 negative,
 $\text{sym}(y)$ 在 $\text{Sym}(x)$ 前面。

$(E\underline{\text{B}}A\underline{\text{F}}C\underline{\text{D}}G, EBC\underline{\text{D}}F\underline{\text{A}}G)$



Symmetry within the sequence-pair representation in the context of placement for analog design. IEEE TCAD, 19(7): 721-731 (2000)

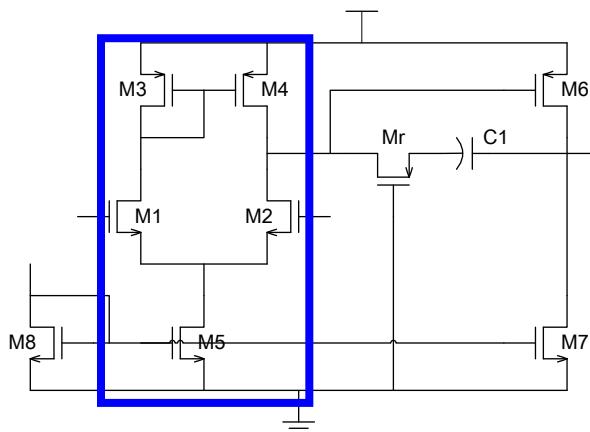
Matching Property of Symmetric Devices

❖ Matching property of MOS transistors

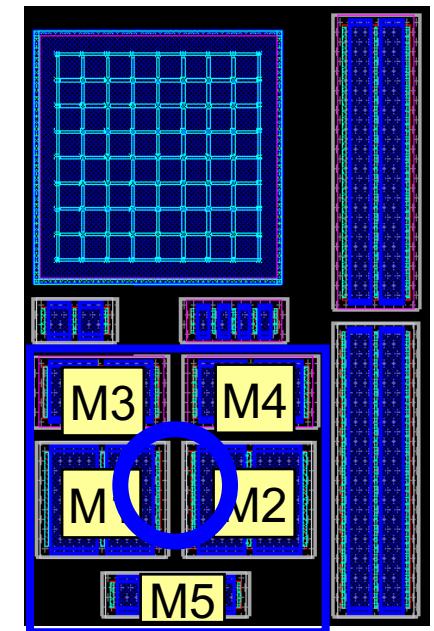
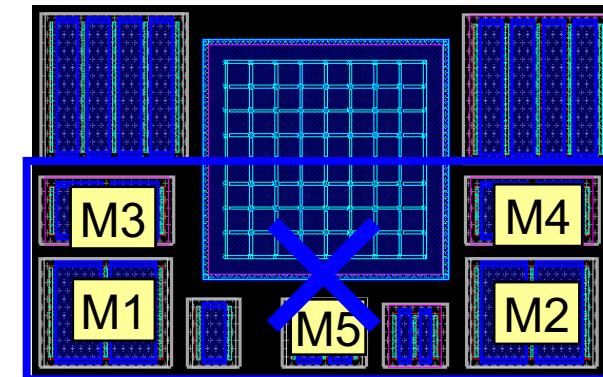
- ▶ Variance of the electrical property P of two identical devices

$$\sigma^2(\Delta P) = \frac{A_P^2}{WL} + S_P^2 D_x^2$$

- W, L – device size
- D_x – distance between two devices



Symmetry group
 $S = \{(M1, M2), (M3, M4), M5\}$



Need **close proximity** of symmetric devices
to get less impacts from process variation

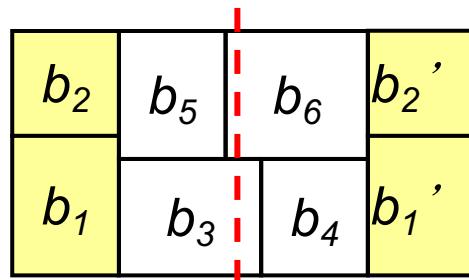
Symmetry Island

❖ A symmetry island is a placement of a symmetry group

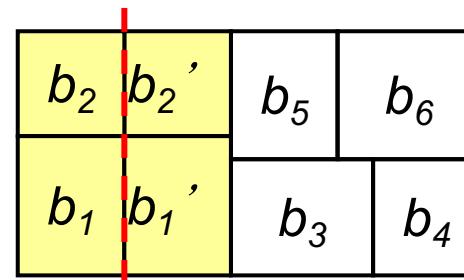
- ▶ Each module in the group abuts at least one of the other modules in the same group
- ▶ The modules in the same group form a connected placement

❖ Example

- ▶ A symmetry group $S = \{(b_1, b_1'), (b_2, b_2')\}$



S does not form a symmetry island.



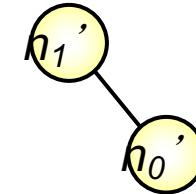
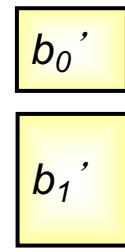
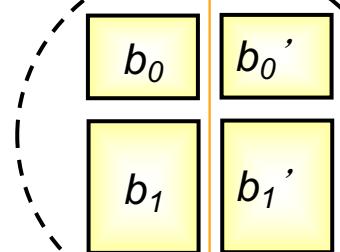
S forms a symmetry island.

↳ “對稱的元件”離對稱軸很近。

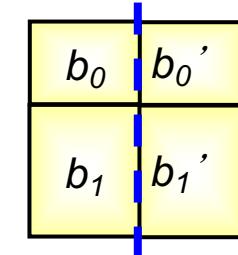
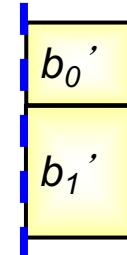
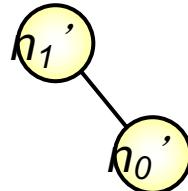
Symmetric-Island-feasible B*-tree

Construction

分成 2 邊，只考慮某一邊。
(根據對稱關係)



Packing



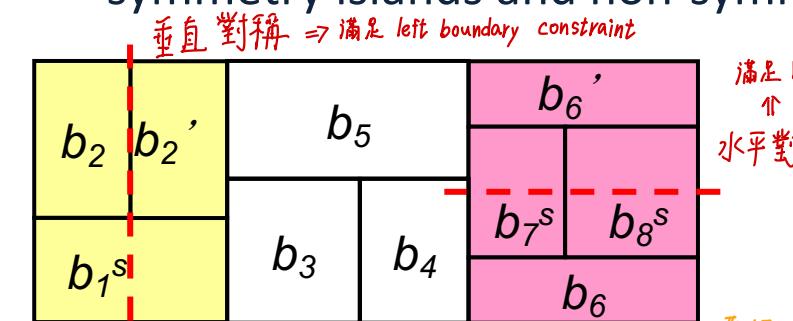
Hierarchical B*-tree (HB*-tree)

❖ Hierarchy node

- ▶ Represent a symmetry island
- ▶ Contain a corresponding ASF-B*-tree

❖ HB*-tree

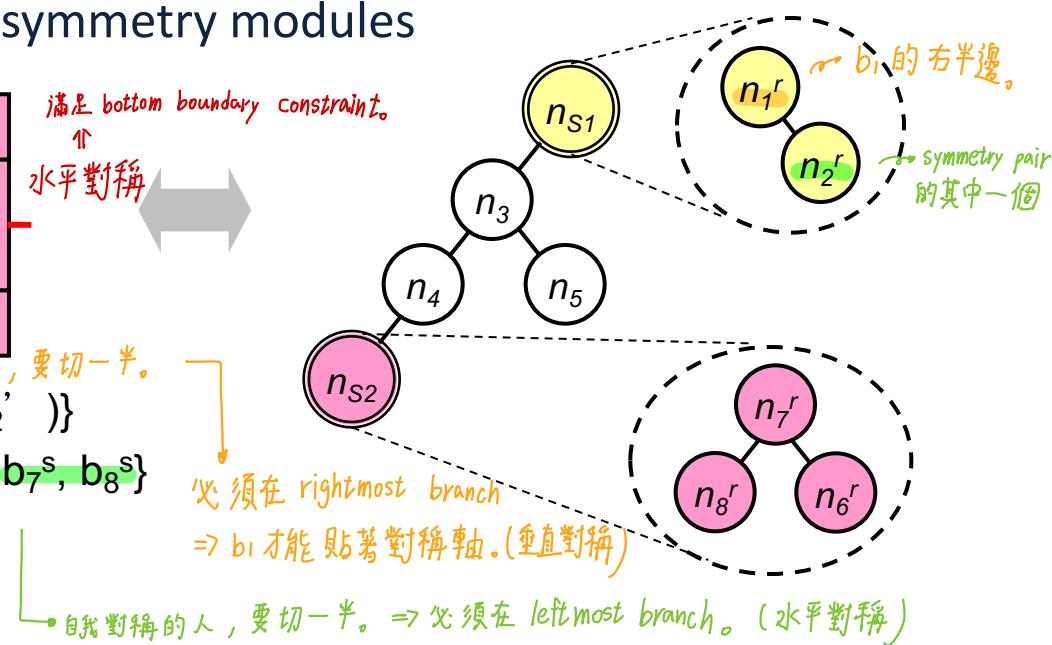
- ▶ Handle the simultaneous placement optimization of modules in the symmetry islands and non-symmetry modules



Symmetry groups: $S_1 = \{b_1^s, (b_2, b_2')\}$

$S_2 = \{(b_6, b_6'), b_7^s, b_8^s\}$

Non-symmetry modules: b_3, b_4, b_5

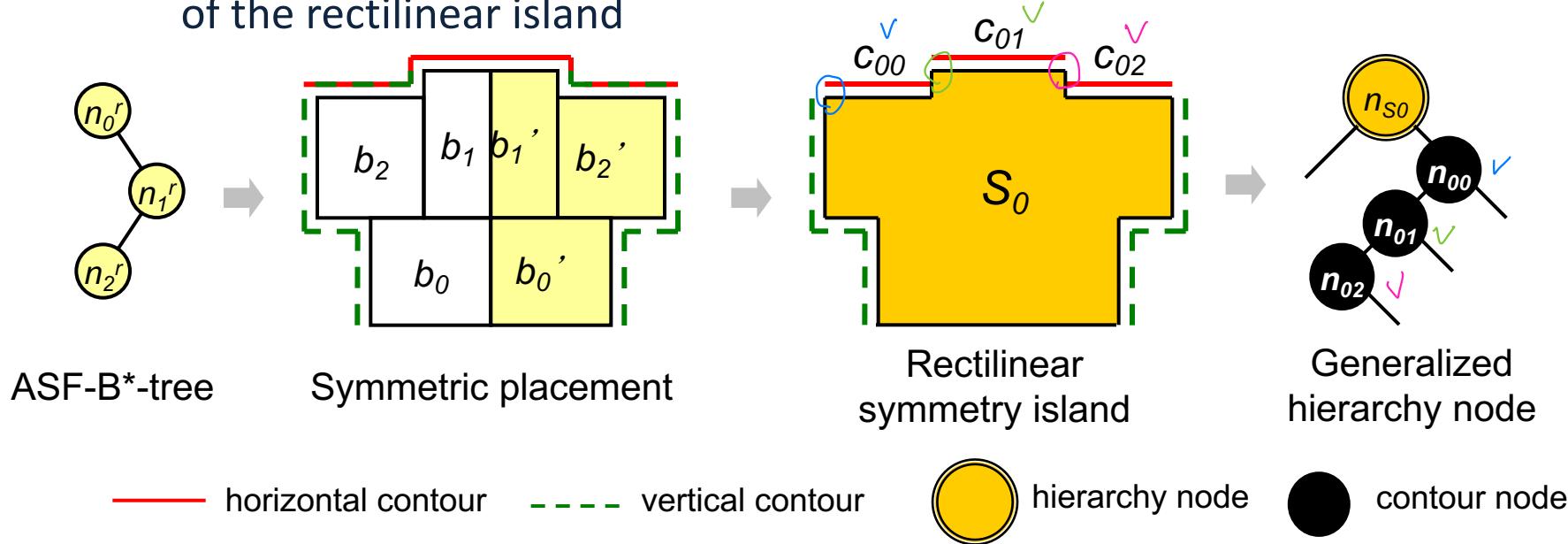


Generalized Hierarchy Node

❖ Rectilinear symmetry island

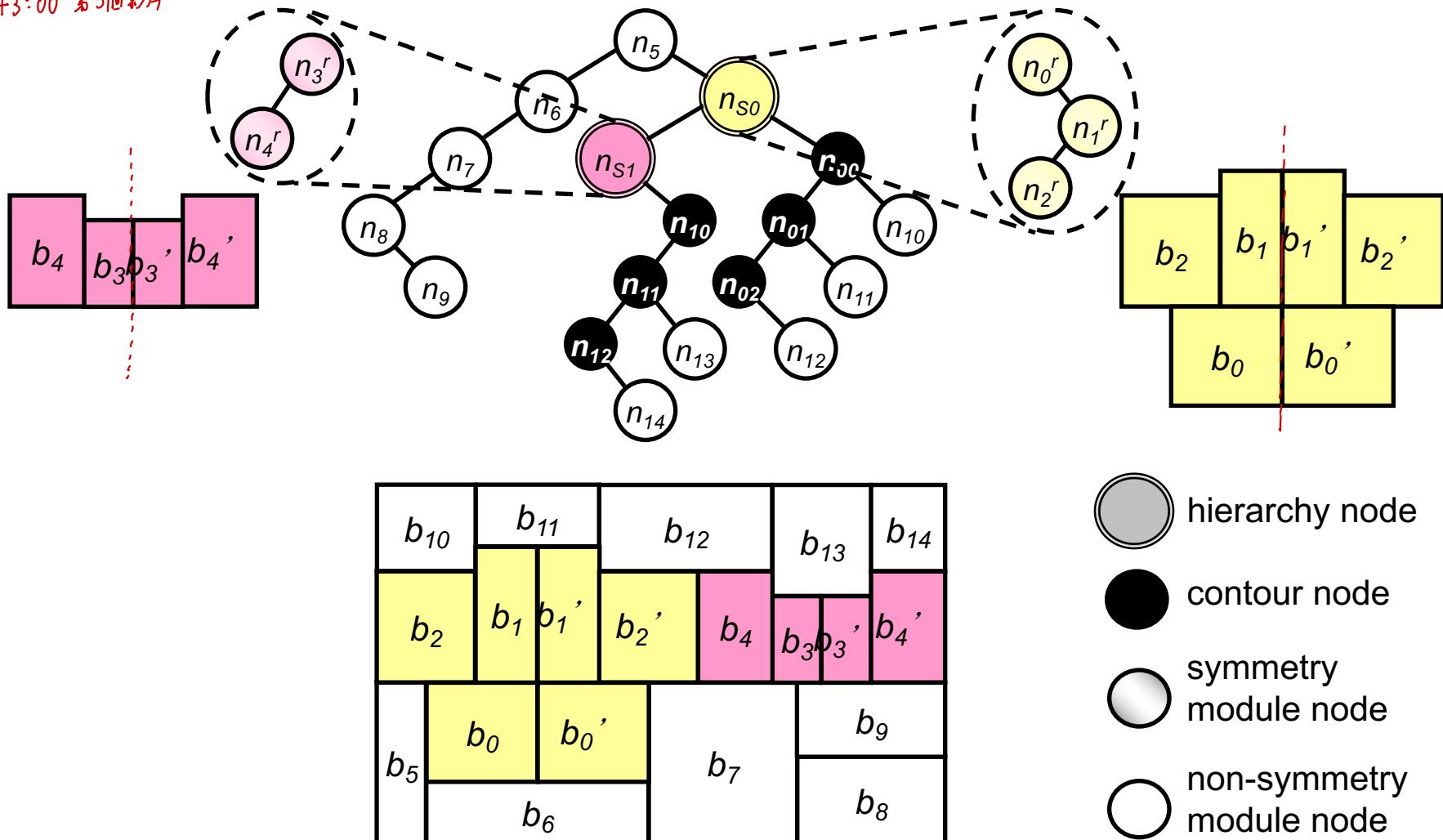
“symmetry Island 擺出來不一定是矩形
要避免 dead space 過多。

- ▶ Keep the horizontal contour and dual vertical contours of the symmetry island in the hierarchy node
- ▶ Introduce **contour nodes** to represent the horizontal contour segments of the rectilinear island

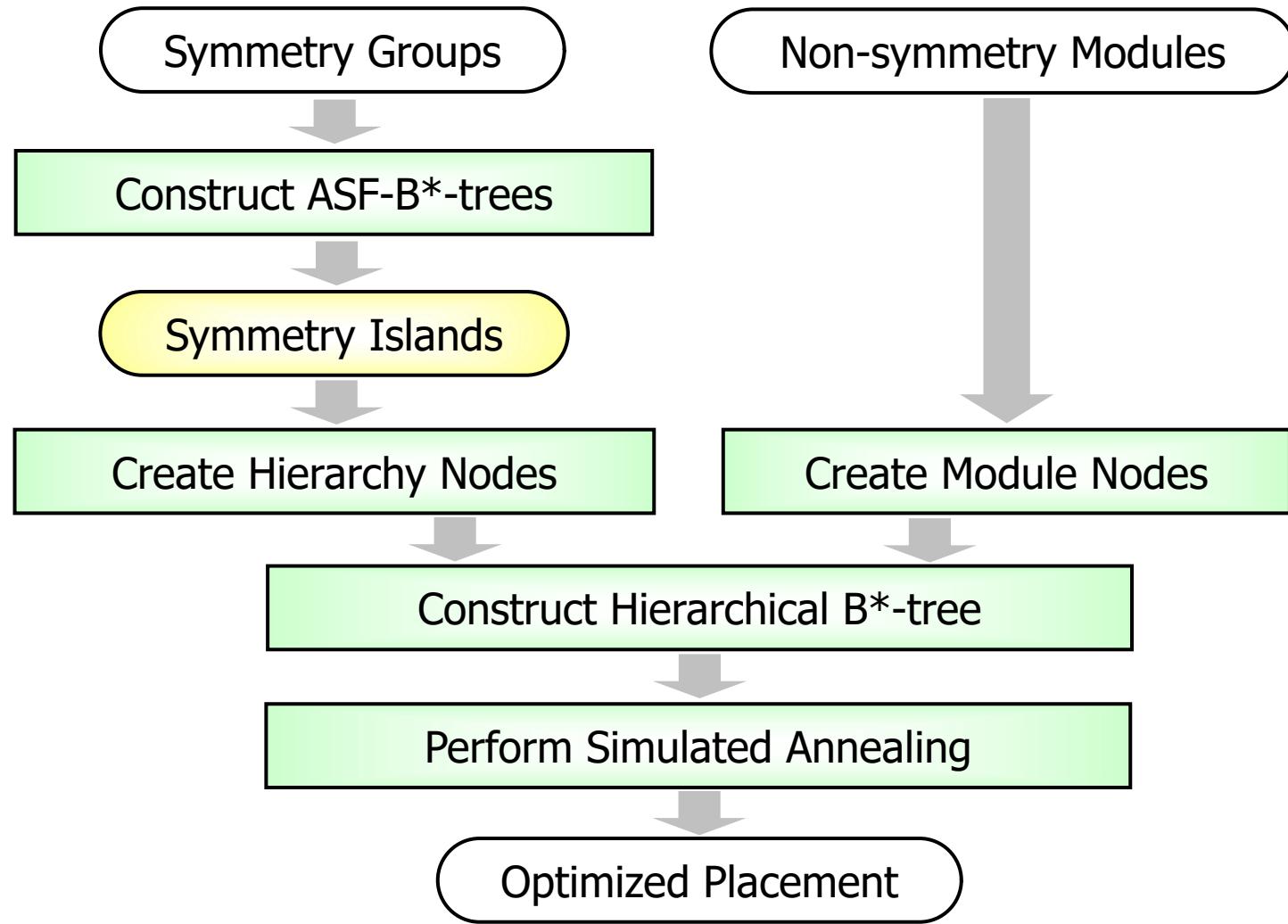


HB*-tree Packing

43:00 第3個影片



Placement Flow with Symmetry Islands

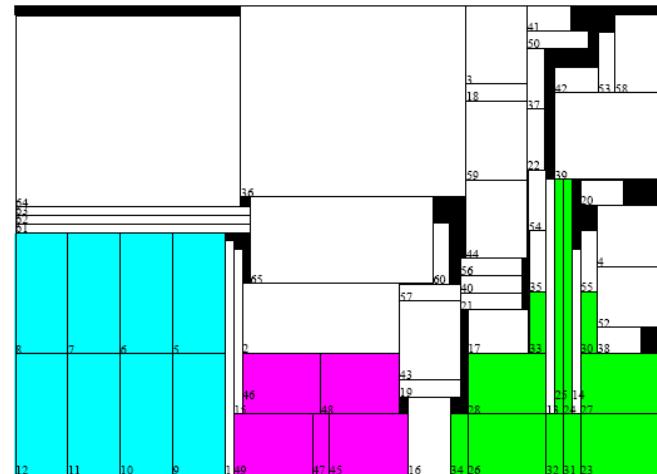


Comparison of Sym-feasible SP, Sym-feasible B*-tree and Sym-island-feasible B*-Tree

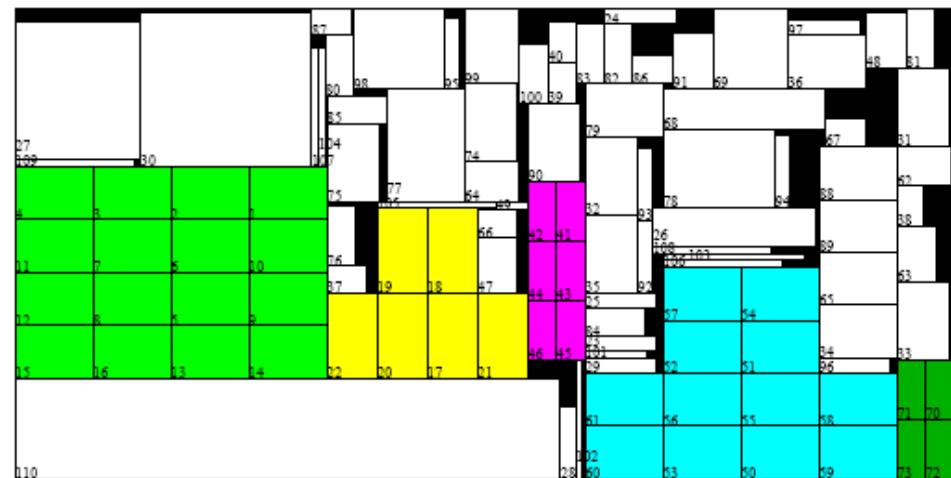
MCNC Circuit	# Mod.	# Sym. Mod.	Sym-feasible SP		Sym-feasible B*-tree		Sym-island-feasible B*-tree	
			Area (mm ²)	Time (s)	Area (mm ²)	Time (s)	Area (mm ²)	Time (s)
Apte	9	8	48.12	25	47.52	11	46.92	2
Hp	11	8	9.84	138	9.71	62	9.35	2
Ami33	33	6	1.24	684	1.23	307	1.23	12
Ami49	49	4	37.82	2038	37.31	983	36.85	20
Comp.			1.03	-	1.02	-	1	1

Resulting Placements w/ Sym-island-feasible B*-Tree

Circuit	biasynth_2p4g
# of Mod.	65
# of Sym. Mod.	8 + 5 + 12
Area utilization	95.53%
Runtime	22 sec



Circuit	Inamixbias_2p4g
# of Mod.	110
# of Sym. Mod.	16 + 6 + 6 + 12 + 4
Area utilization	94.59%
Runtime	43 sec



Analog Placement Based on Symmetry-Island Formulation. IEEE TCAD, 28(6): 791-804 (2009)

Comparison of Time Complexities

Topological Representations for Analog Placement Considering Symmetry Constraints	Perturbation Time	Packing Time
ASF-B*-tree + HB*-tree [Lin et al., DAC'07 & TCAD'09]	$O(\lg n)$	$O(n)$
B*-tree [Balasa et al., ICCAD'00]	$O(\lg n)$	$O(n^2)$
B*-tree w/ Seg. Tree [Balasa et al., ICCAD'02 & TCAD'04]	$O(\lg n)$	$O(n \lg n)$
B*-tree w/ RB tree [Balasa et al., ASPDAC'03]	$O(\lg n)$	$O(n \lg n)$
B*-tree w/ Skip list [Balasa et al., ASICON'05]	$O(\lg n)$	$O(n \lg n)$
O-tree [Pang et al., DAC'00]	$O(\lg n)$	$O(n^2)$
SP [Balasa & Lampaert, DAC'99 & TCAD'00]	$O(1)$	$O(n^2)$
SP w/ LP [Kouda et al., ISPD'06 & TCAD'07]	$O(1)$	$O(n^2)$
SP w/ Dummy [Tam et al., ICCAD'06]	$O(1)$	$O(n^2)$
SP w/ Priority Queue [Krish. et al., ISCAS'07]	$O(1)$	$O(mn \lg \lg n)$
TCG-S [Lin et al., ASPDAC'05]	$O(n^2)$	$O(n^2)$
TCG [Zhang et al., ASPDAC'08]	$O(n)$	$O(n^2)$

Handling Placement Constraints in Analog Layout Synthesis

教育部智慧晶片系統與應用人才培育計畫
111年度模組教材發展計畫

1:22:20

Lecture 3

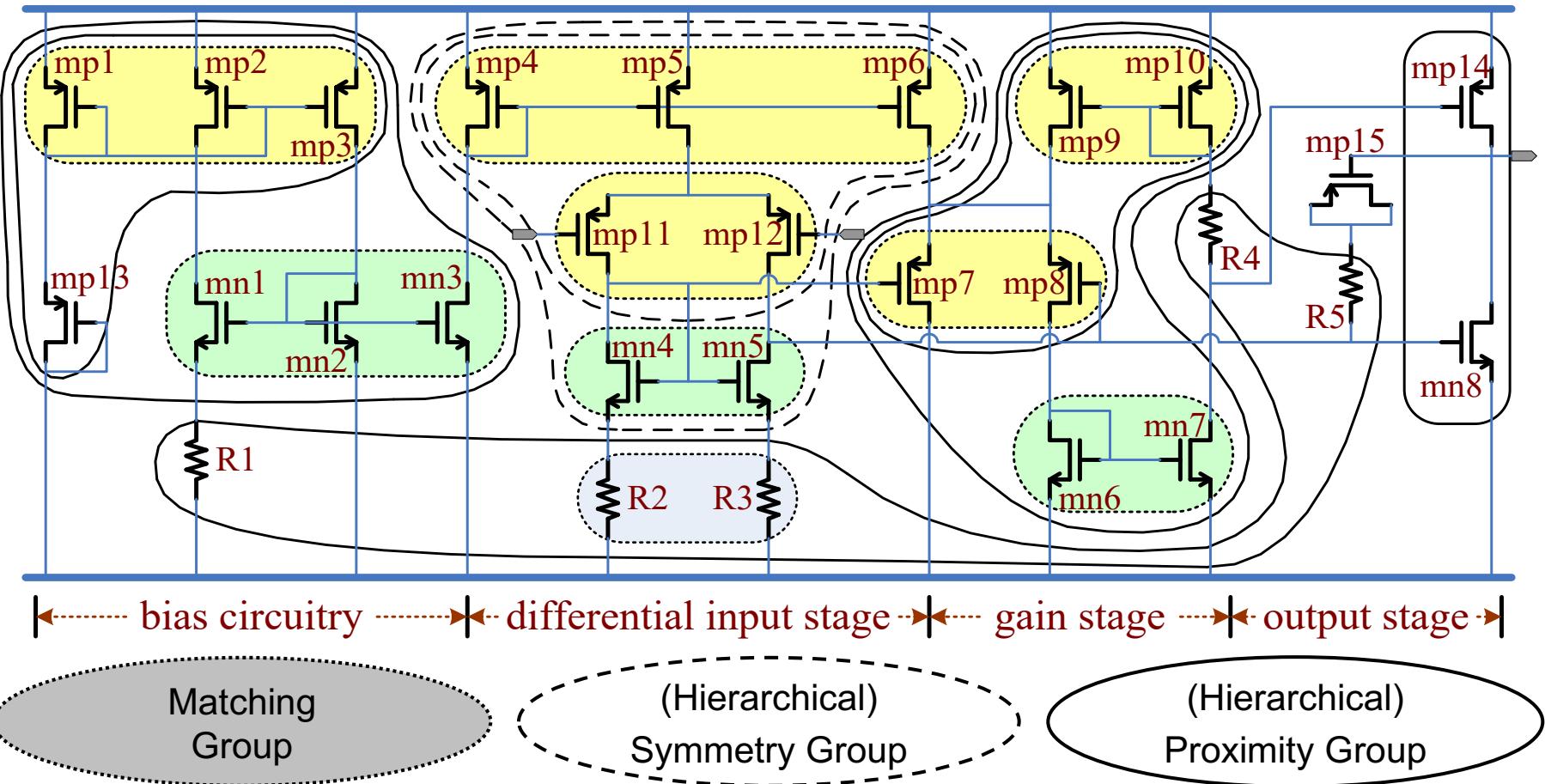
Analog Placement Method Considering Other Placement Constraints

Mark Po-Hung Lin

College of Artificial Intelligence

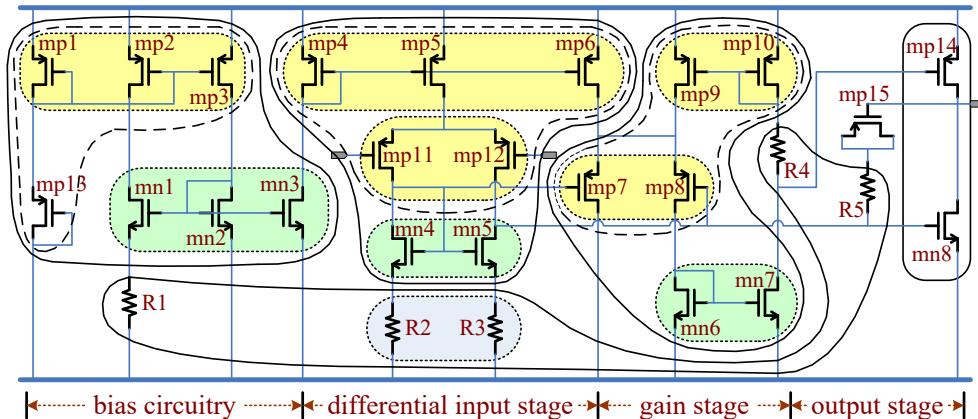
National Yang Ming Chiao Tung University

Proximity Constraints

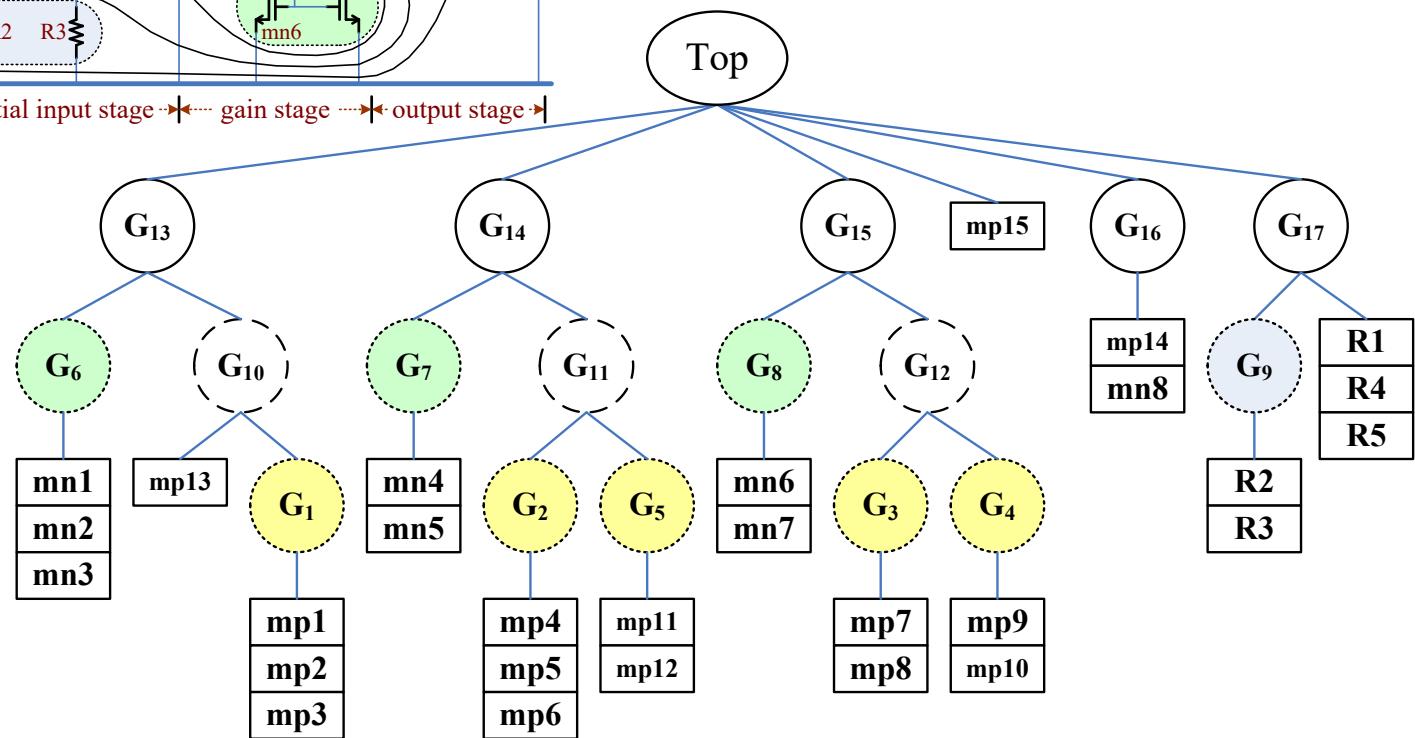


Analog placement based on hierarchical module clustering. DAC 2008: 50-55

Netlist & Device Constraint Hierarchy

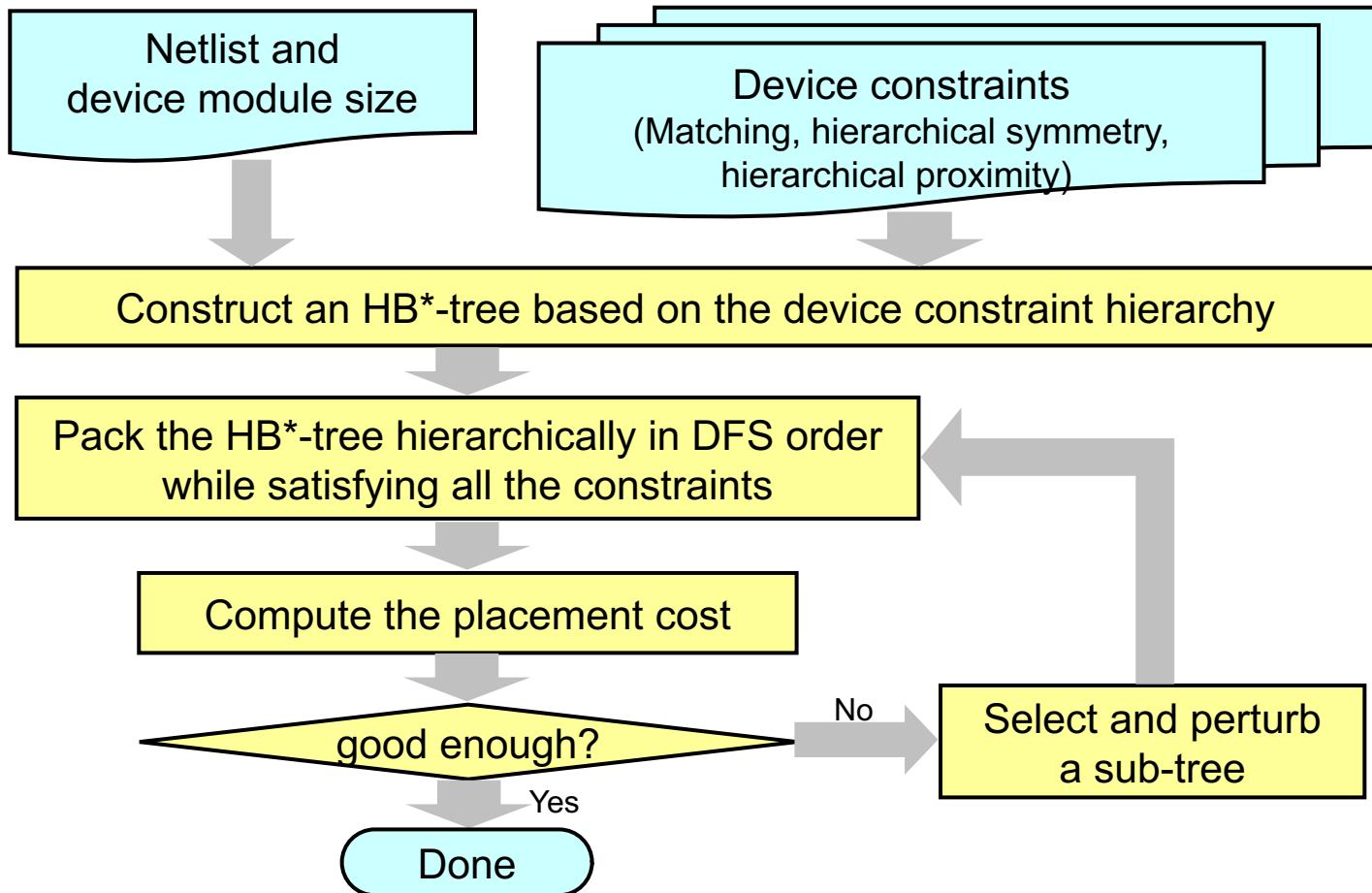


9 matching groups
2 symmetry groups
6 proximity groups



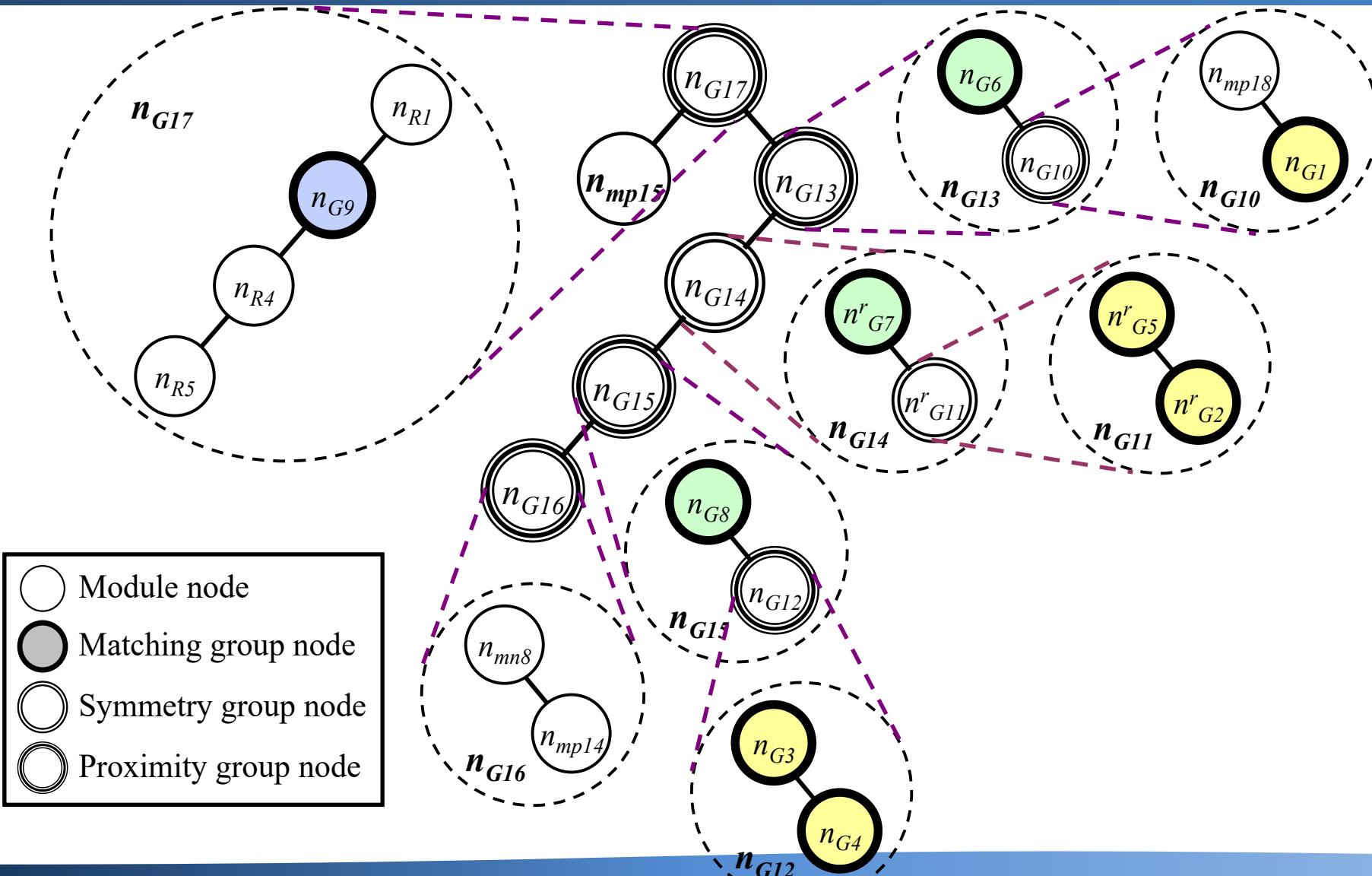
Analog placement based on hierarchical module clustering. DAC 2008: 50-55

Placement Flow

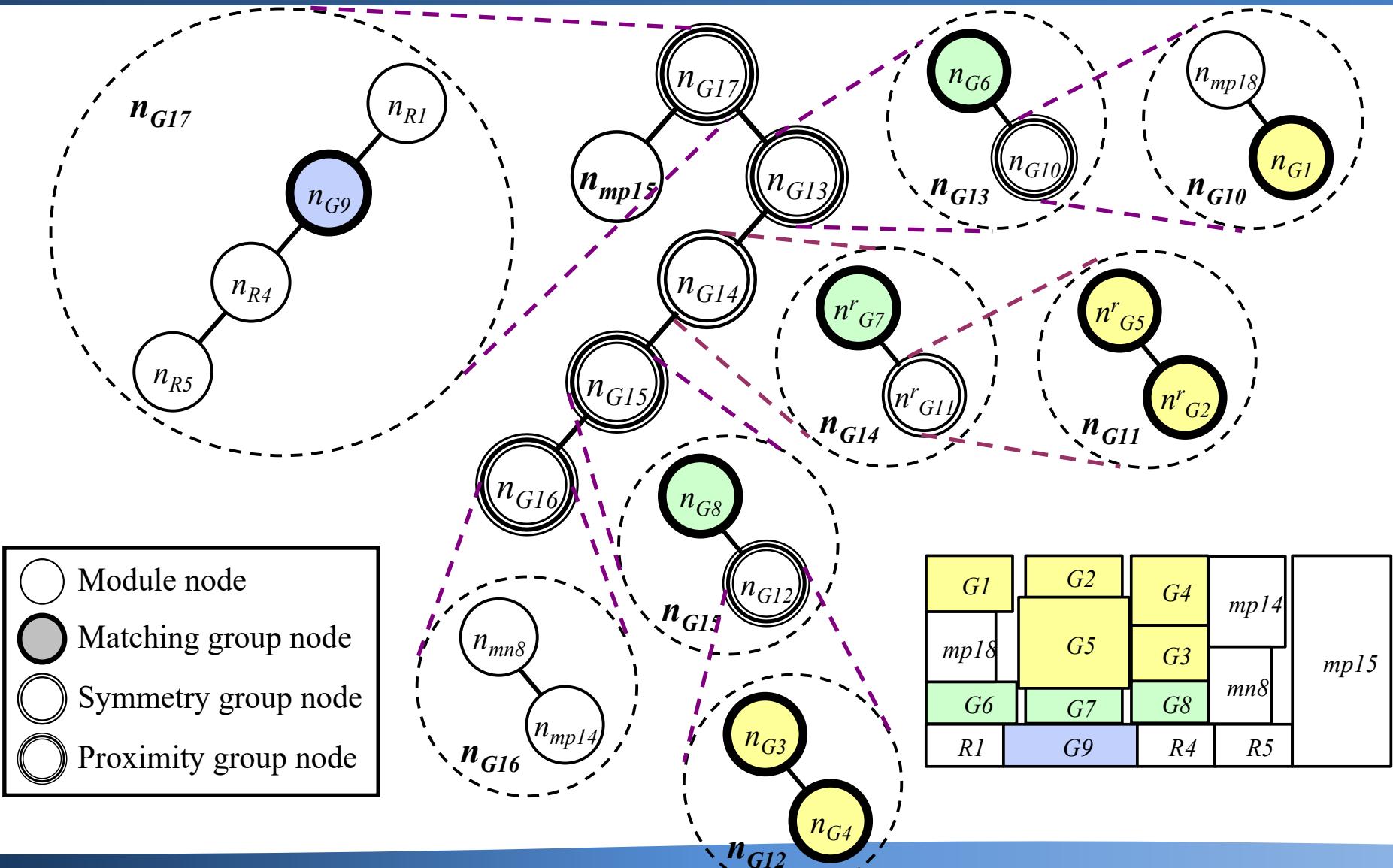


Analog placement based on hierarchical module clustering. DAC 2008: 50-55

HB*-tree Representation



HB*-tree Packing



Current-Path Constraints

❖ Straight current path

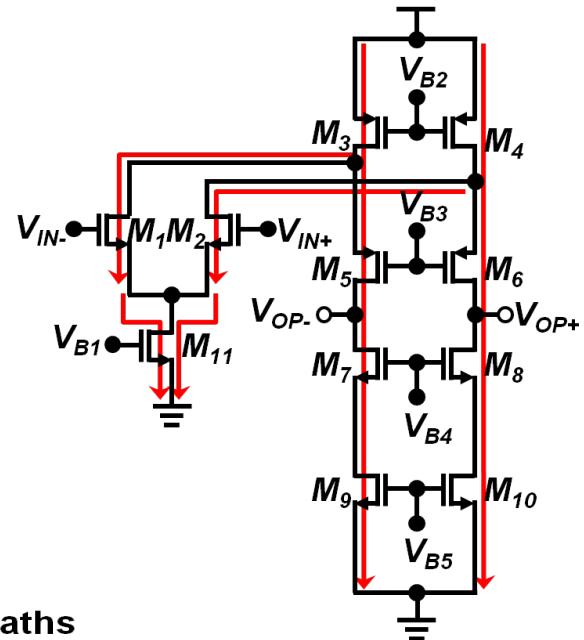
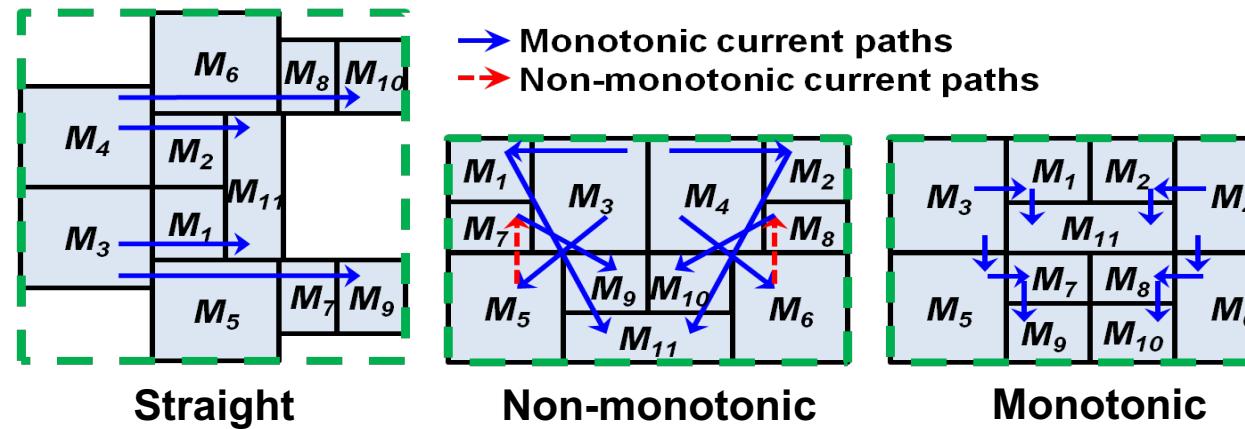
- ▶ Larger area

❖ Non-monotonic current path

- ▶ Larger routing-induced parasitic

❖ Monotonic current path

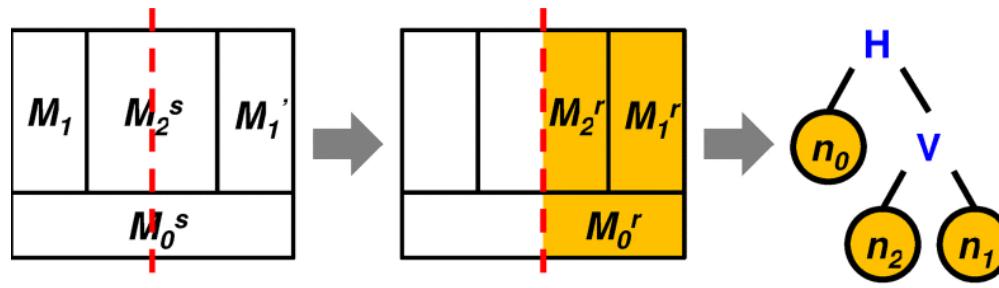
- ▶ Smaller area and routing-induced parasitic



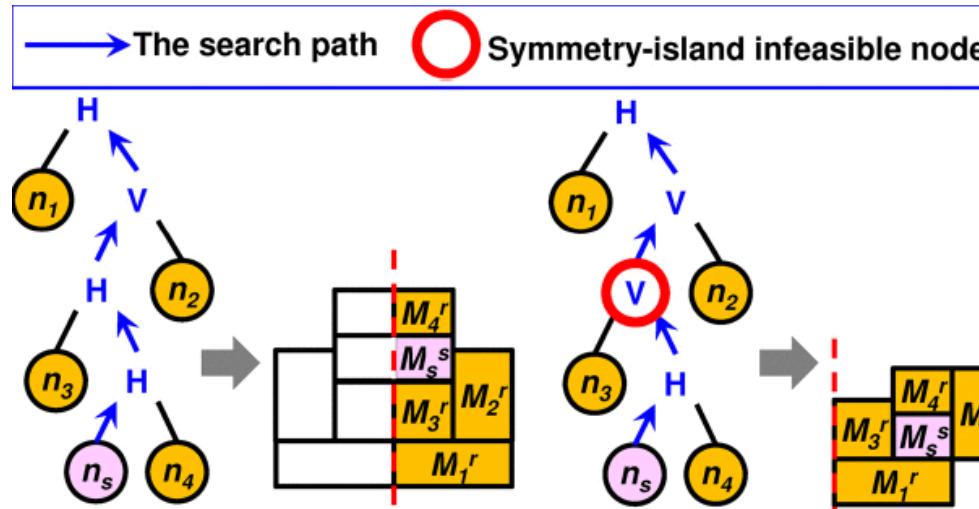
Performance-driven analog placement considering monotonic current paths. ICCAD 2012: 613-619

Symmetry-island-feasible Slicing Tree

❖ Symmetry-island-feasible slicing tree



❖ Symmetry-island-infeasible slicing tree



Symmetry-island-feasible Slicing Tree Validation

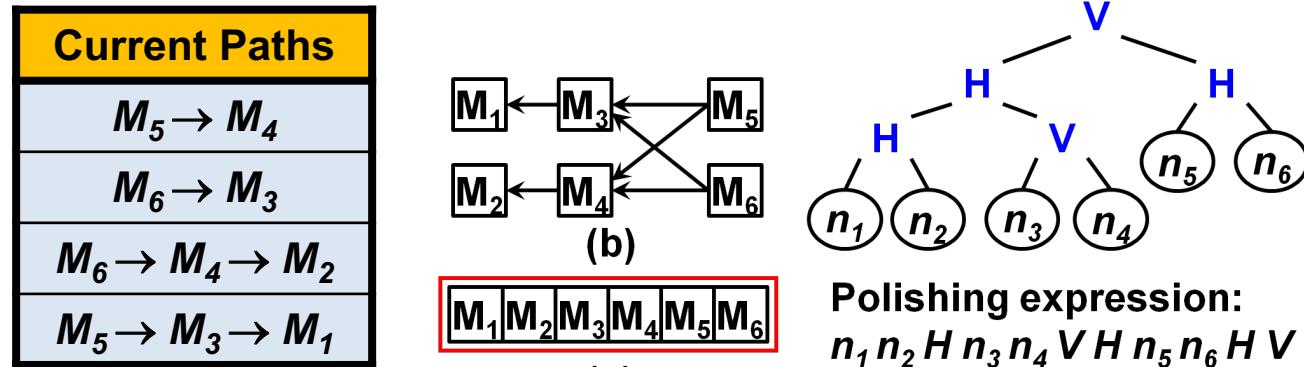
- ❖ For each representative node in the RST, find a search path passing through all its ancestors starting from the representative node to the root of the RST
- ❖ If there is an ancestor labeled V in the search path, check whether the node right before the ancestor in the search path is the right child of the ancestor
- ❖ If the condition in the previous step is true, the RST is not an SIFST. Otherwise, the RST is also an SIFST.

Monotonic-current-path Placement Conditions

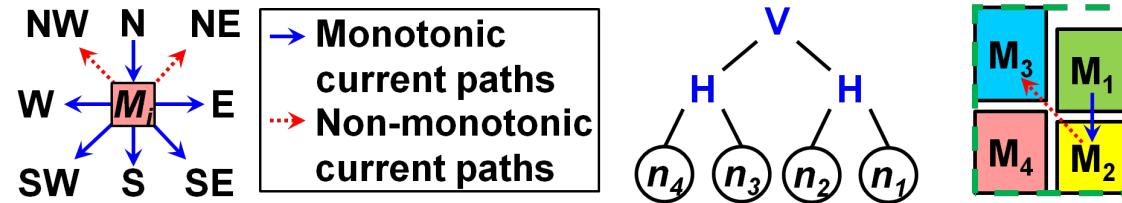
- ❖ Given a current path among three modules, $M_i \rightarrow M_j \rightarrow M_k$, the following topological monotonic-current-path constraints must be satisfied to avoid nonmonotonic current paths in the resulting placement.
 - ▶ If M_i is North to M_j , M_k must not be North to M_j
 - ▶ If M_i is South to M_j , M_k must not be South to M_j
 - ▶ If M_i is East to M_j , M_k must not be East to M_j
 - ▶ If M_i is West to M_j , M_k must not be West to M_j

Current-Path aware Slicing Tree

- ❖ Sufficient condition of monotonic current path in a slicing tree



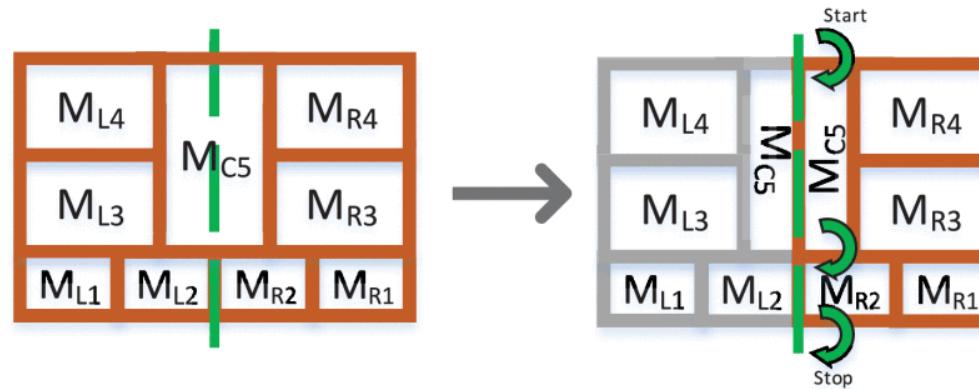
- ❖ Local movement to satisfy monotonic-current-path constraints



Exploring Feasibilities of Symmetry Islands and Monotonic Current Paths in Slicing Trees for Analog Placement. IEEE TCAD, 33(6): 879-892 (2014)

Symmetry-island-feasible Sequence Pair

- ❖ The representative SP corresponding to the right-half of the symmetric placement is $(RSP+; RSP-) = (54321; 21534)$.
- ❖ The curved arrow from start to stop represents the symmetric-axis search path passing from MC5 and MR2



Exploring Multiple Analog Placements With Partial-Monotonic Current Paths and Symmetry Constraints Using PCP-SP.
IEEE TCAD, 39(12): 5056-5068 (2020)