CS 6135
VLSI Physical Design Automation
VLSI實體設計自動化
Fall 2022

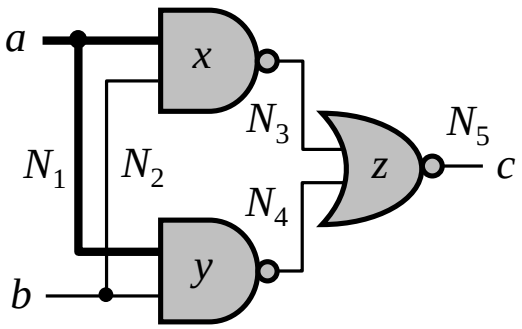
Course Information

- Time: T3T4R3
- Room: Delta 109
- Instructor: Ting-Chi Wang (王廷基), EECS 643/Delta 527,
  (03) 5742963/5731070, tcwang@cs.nthu.edu.tw
- TAs: Chen-Han Lu (呂宸漢, love0416much@gmail.com) &
  En-Yu Liao (廖恩宇, danielliaotpe@gmail.com),
  EECS 229/230, ext. 33536/33570
- Website: eeclass (https://eeclass.nthu.edu.tw)
- Material: technical papers selected from major EDA (Electronic Design Automation) conference proceedings and journals
- Grading policy
  - Homework assignments: 80%
  - One exam: 20%

Physical Design

- Physical design is the process of converting a circuit netlist into a geometric description (i.e., determining where to put components and how to connect them).
- The description is used to manufacture a chip.
- Objectives: area, performance, and power, etc.
- Constraints: components may not be too close, and wires cannot cross, etc.

Netlist



(**a: N_1**)
(b: N_2)
(c: N_5)
(**x: IN1 N_1**, IN2 N_2, OUT N_3)
(**y: IN1 N_1**, IN2 N_2, OUT N_4)
(z: IN1 N_3, IN2 N_4, OUT N_5)

Pin-Oriented Netlist

(**N_1: a, x.IN1, y.IN1**)
(N_2: b, x.IN2, y.IN2)
(N_3: x.OUT, z.IN1)
(N_4: y.OUT, z.IN2)
(N_5: z.OUT, c)

Net-Oriented Netlist

Computer-Aided Design (CAD)

- Physical design is very complicated:
  - Millions of components
  - Multiple objectives
  - Multiple constraints
- Chip designers need help from CAD tools.

Course Objectives

- Understanding the problems arising in the physical design of VLSI circuits.
- Understanding various CAD algorithms for automating the physical design process.

Course Topics

- Introduction
- Partitioning
- Floorplanning
- Placement
- Routing
- Post-routing optimization

Target Audience

Students who want to be:

- VLSI CAD Engineers & Researchers
  - Development and Implementation of CAD tools
- VLSI Designers
  - Designing VLSI chips using CAD tools

Expected Background

- Digital Logic Design
- Algorithms (or Data Structures)
- Programming Languages such as C or C++

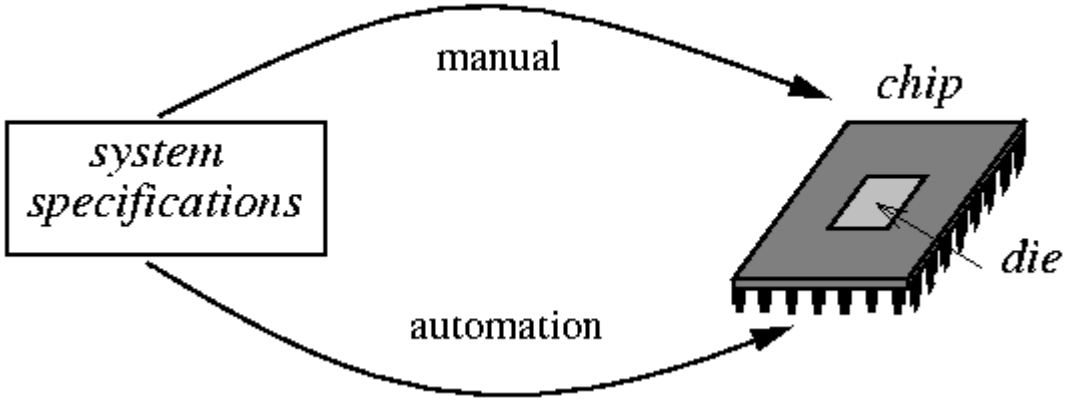
Major Conferences and Journals

- Conferences
  - Design Automation Conference (DAC)
  - International Conference on Computer-Aided Design (ICCAD)
  - Asia and South Pacific Design Automation Conference (ASP-DAC)
  - Design, Automation and Test in Europe (DATE)
  - International Symposium on Physical Design (ISPD)
- Journals
  - IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD)
  - ACM Transactions on Design Automation of Electronic Systems (TODAES)
  - IEEE Transactions on Very Large Scale Integration (VLSI) Systems (TVLSI)
  - INTEGRATION: The VLSI Journal

Related Books

- A. B. Kahng, J. Lienig, I. L. Markov, J. Hu, *VLSI Physical Design: From Graph Partitioning to Timing Closure*, Springer, 2011.
- *Handbook of Algorithms for Physical Design Automation*, edited by C. J. Alpert, D. P. Mehta, S. Sapatnekar, CRC Press, 2009.
- *Electronic Design Automation: Synthesis, Verification, and Test*, edited by L.-T. Wang, Y.-W. Chang, K.-T. Cheng, Morgan Kaufmann, 2009.
- S. K. Lim, *Practical Problems in VLSI Physical Design Automation*, Springer, 2008
- N. Sherwani, *Algorithms for VLSI Physical Design Automation*, 3rd Edition, Kluwer Academic Publishers, 1999.
- M. Sarrafzadeh and C. K. Wong, *An Introduction to VLSI Physical Design*, McGraw-Hill, 1996.
- S. M. Sait and H. Youssef, *VLSI Physical Design Automation: Theory and Practice*, McGraw-Hill, 1995.

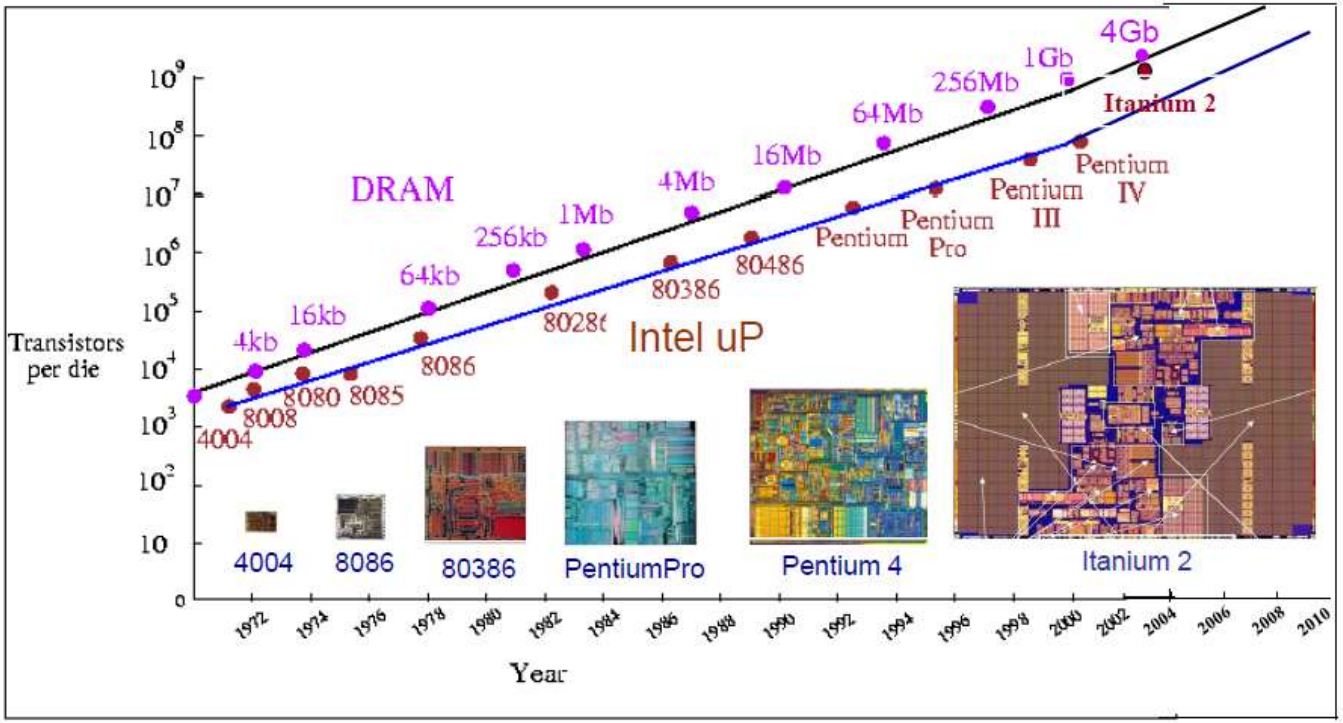
VLSI Design Considerations



- Several conflicting considerations
  - Design complexity: large number of devices/transistors
  - Performance: optimization requirements for high performance
  - Time-to-market: about a 15% gain for early birds
  - Cost: die area, packaging, testing, etc
  - Others: power consumption, noise, reliability, etc

Technology Advancement

- In 1965, Gordon Moore stated that the number of transistors on an IC would double every year. Ten years later, he revised his statement, asserting that they double every 18 months. Since then, this “rule” has been famously known as Moore’s Law.
- Logic capacity doubles per IC at regular intervals.

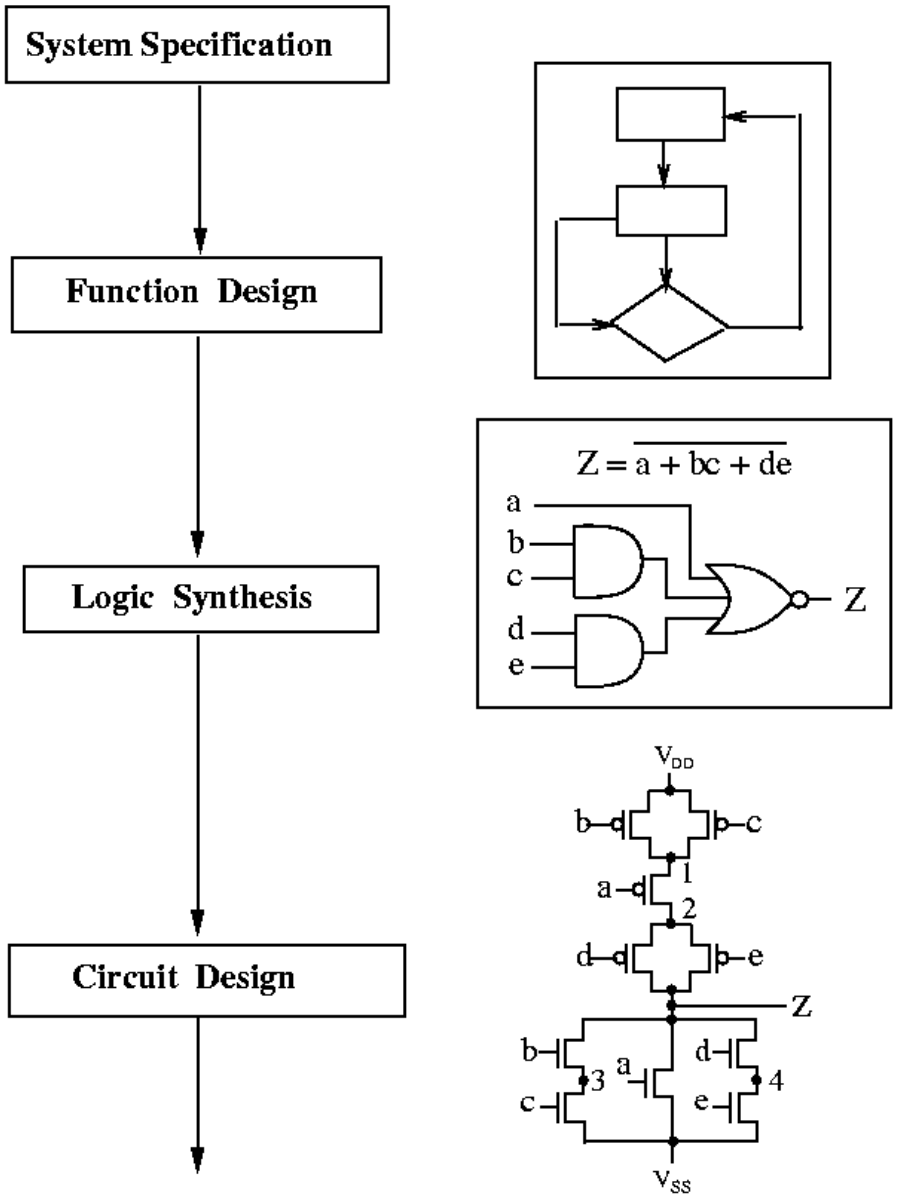


Problems with Future Technology

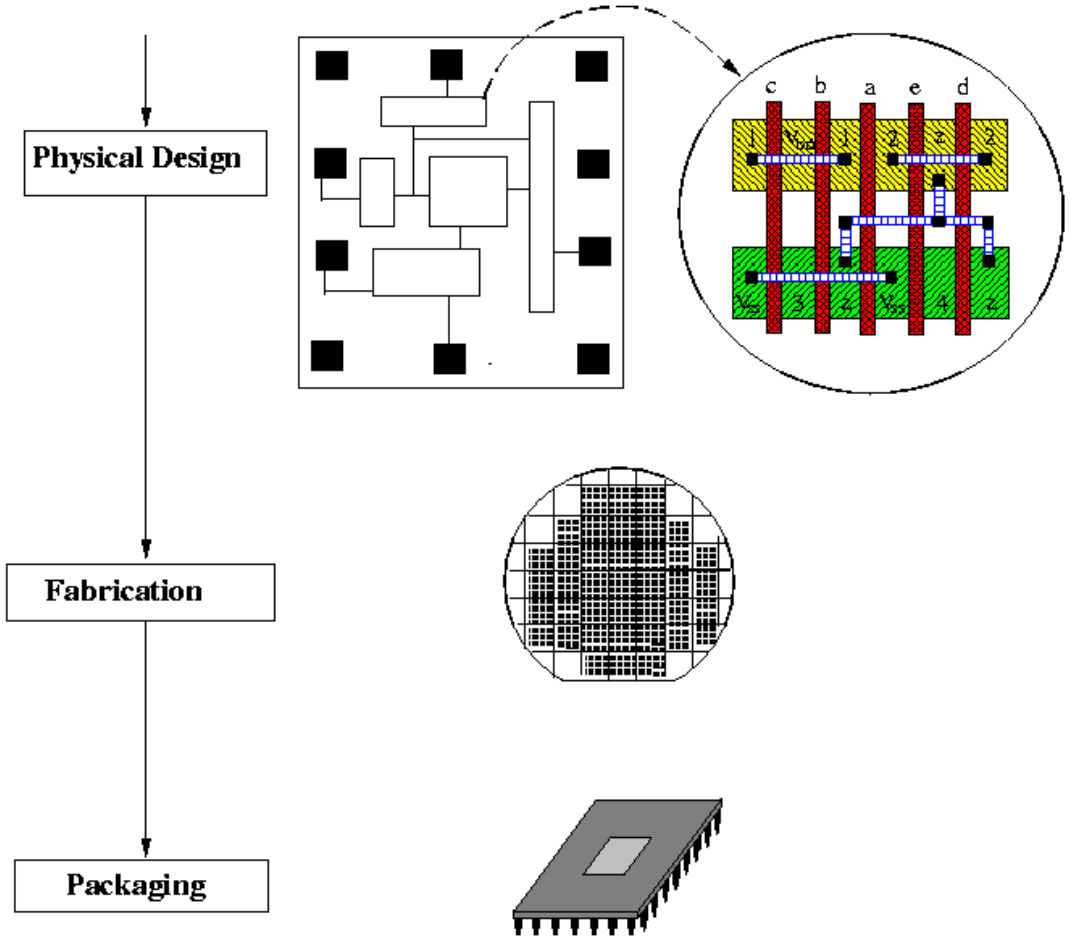
- Deep submicron technology: node (feature size) < 0.25 μm
- Nanometer technology: node < $0.1\mu m$.
- International Technology Roadmap for Semiconductors (http: www.itrs2.net/)
- International Roadmap for Devices and Systems (http://irds.ieee.org/)
- Designs are too complicated to be handled manually
- Solutions:
  - CAD
  - Hierarchical design
  - Design reuse

Traditional VLSI Design Flow

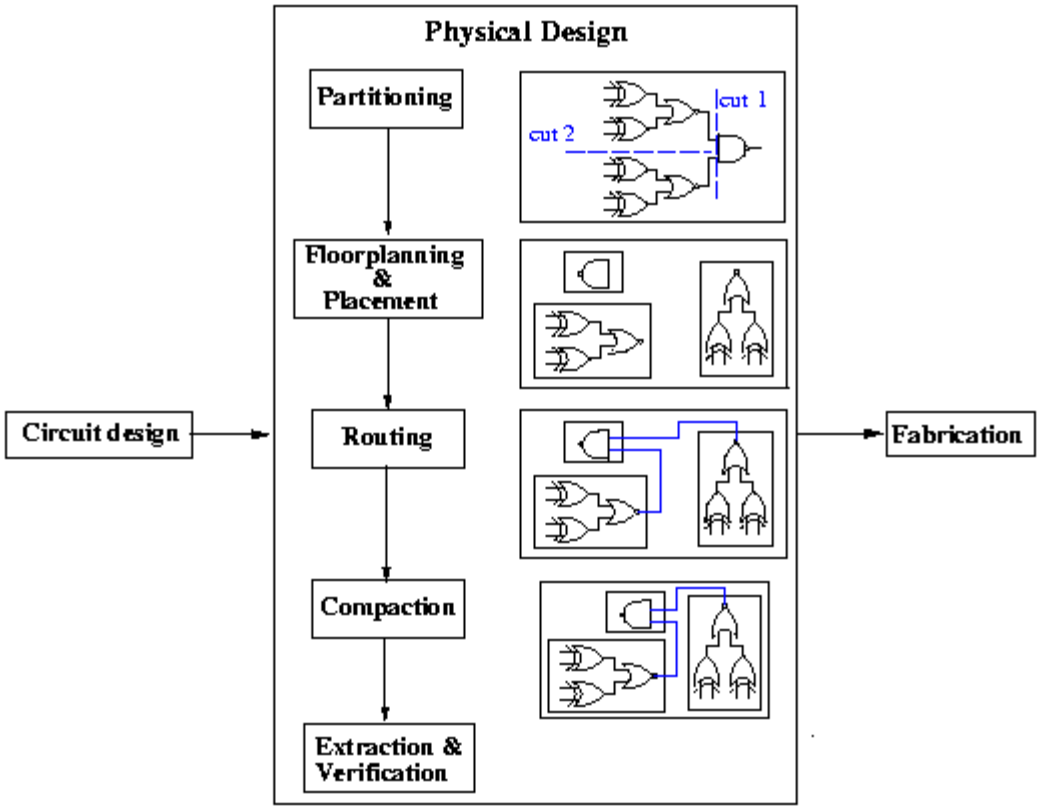
- 1. System specification
- 2. Functional design
- 3. Logical synthesis
- 4. Circuit design
- 5. **Physical Design**
- 6. Fabrication
- 7. Packaging
- Other tasks involved: function/timing verification, etc.


System Specification
Function Design
Logic Synthesis
Circuit Design
$Z = \overline{a + bc + de}$
a
b
c
d
e
Z
V_{DD}
b
c
1
a
2
d
e
Z
b
3
a
d
4
c
e
V_{SS}

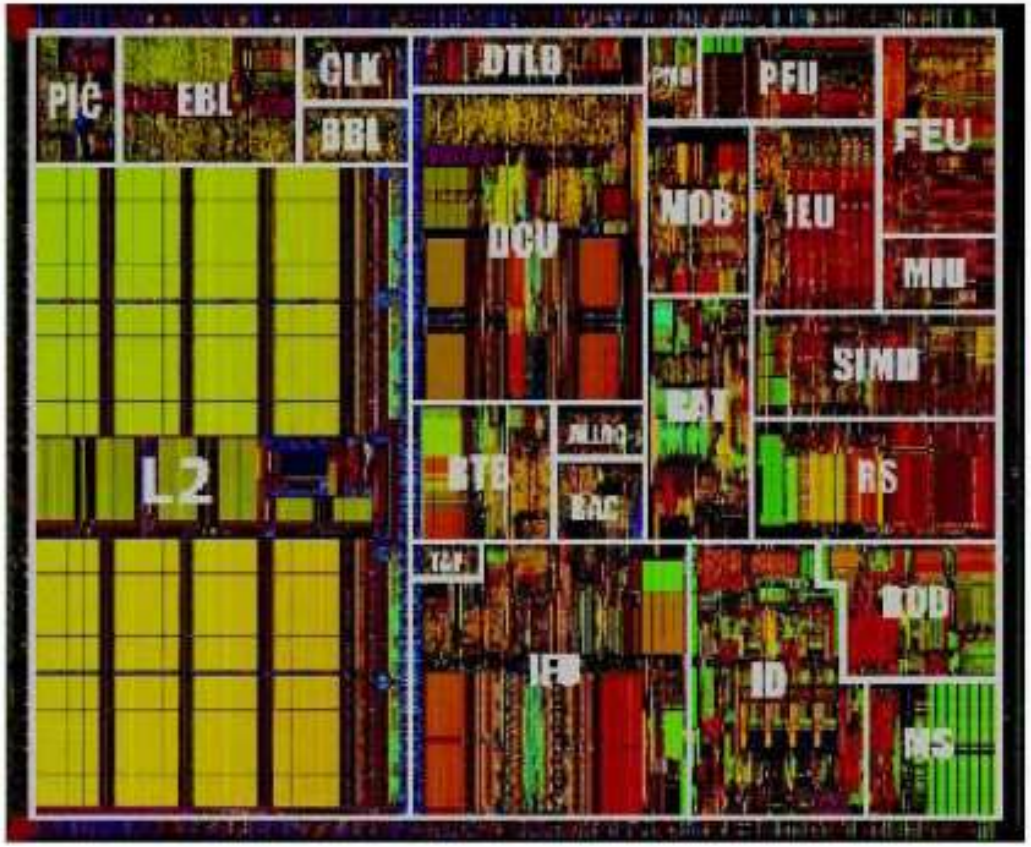
Traditional VLSI Design Flow (Cont'd)



Physical Design Flow

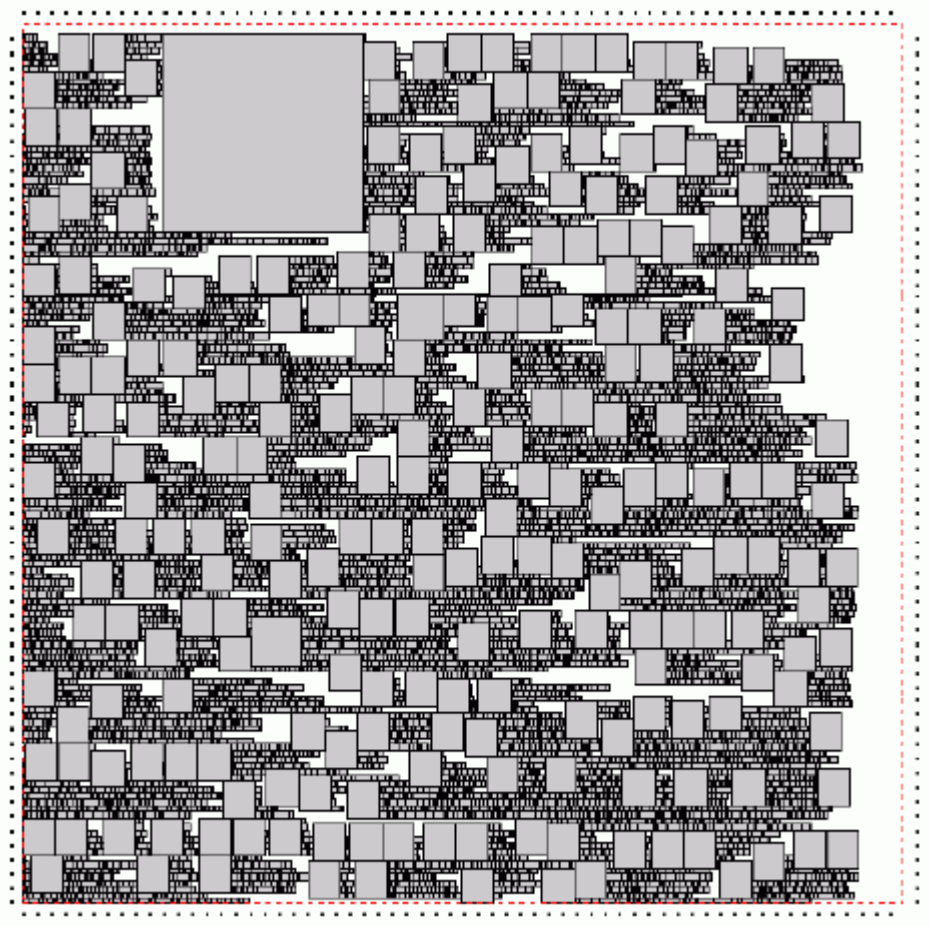


Floorplan Example

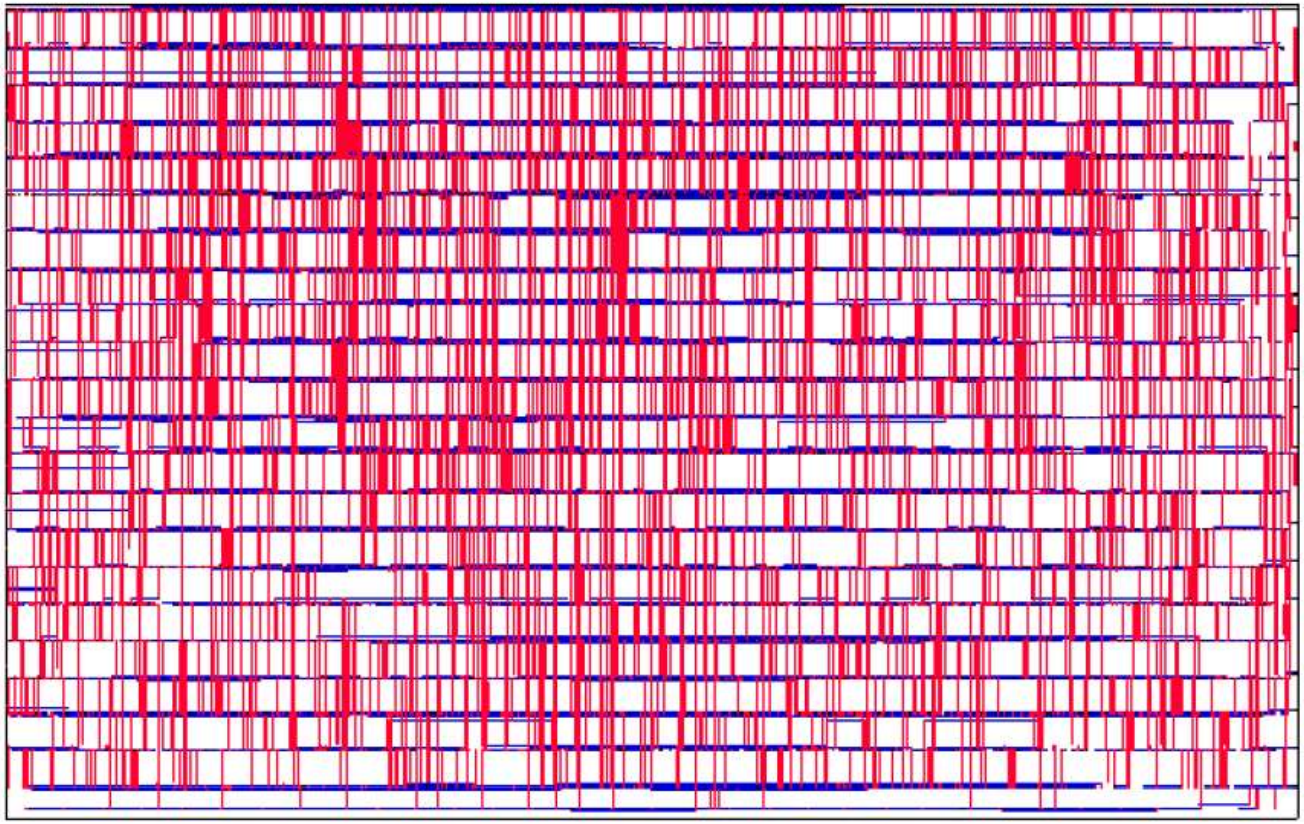


Intel
Pentium 4

Placement Example



Routing Example

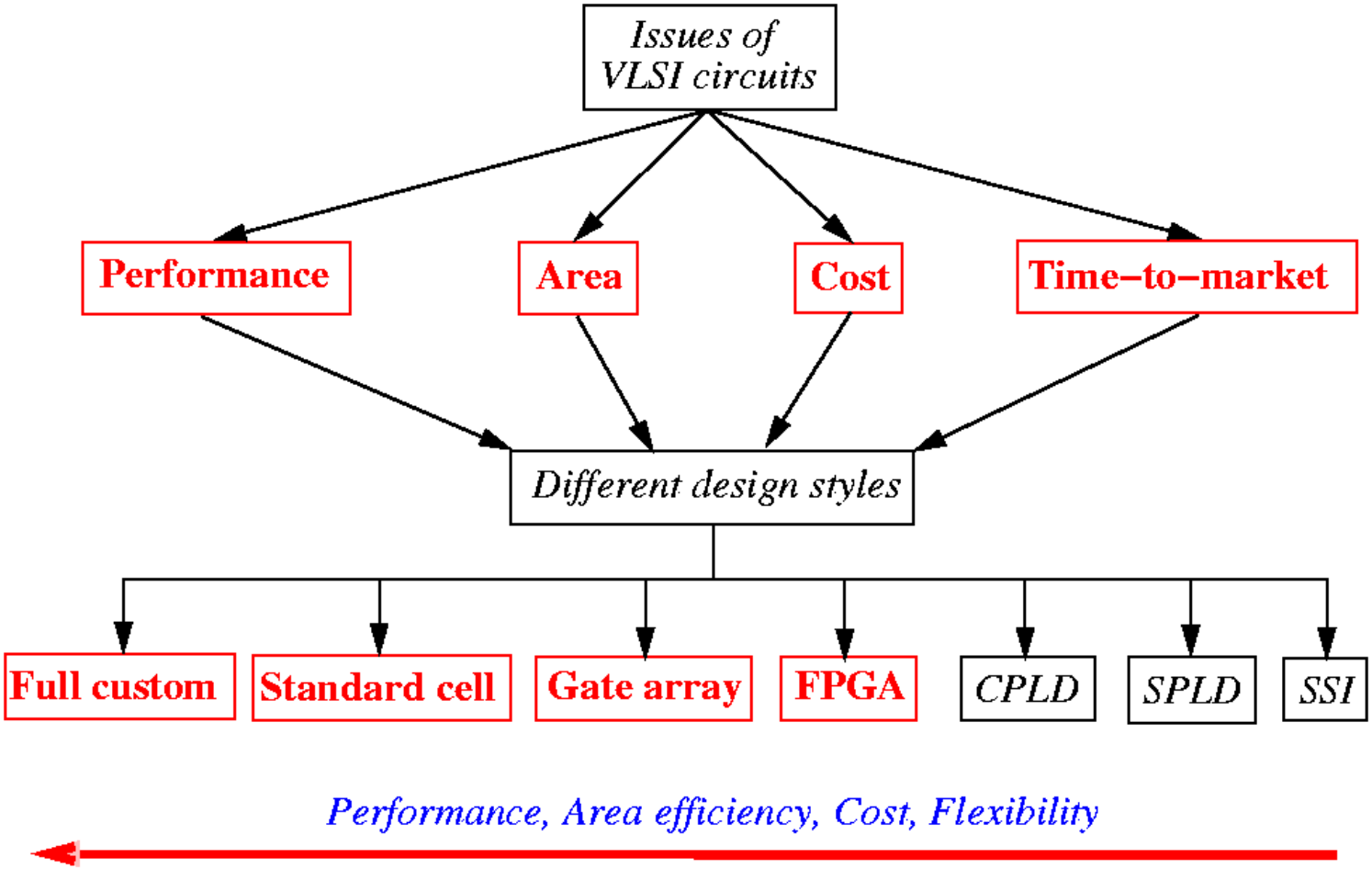


Design Styles

- Restricting design styles to reduce complexity.
- Choosing design style according to

  design time, performance, size and cost, etc.

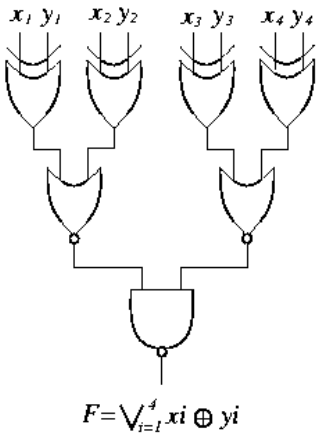
Design Styles (Cont'd)



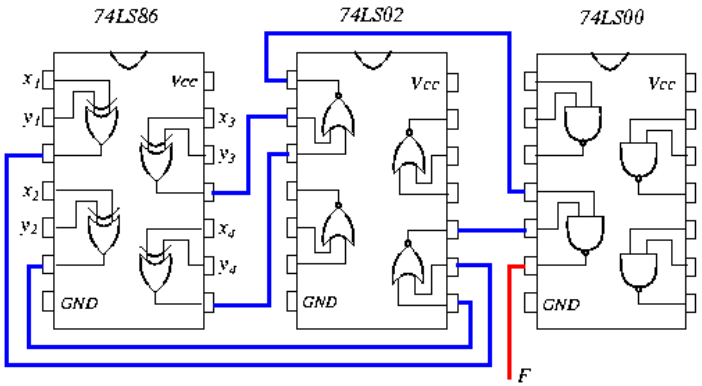
Design Styles (Cont'd)

- Full Custom
  - Fewest constraints
- Standard Cell
  - A library of cells designed in (multiples of) a fixed height
  - A design consisting of rows of cells
- Gate Array
  - A design consisting of an array of identical pre-fabricated cells with transistors but no connections
  - Requiring intra-cell routing (creating a cell by connecting certain transistors) and inter-cell routing (connecting cells to form nets from the netlist)
  - Fabricating routing layers after chip-specific requirements are known
- Field Programmable Gate Array (FPGA)
  - Pre-fabricated cells and interconnects
  - Programmable cells and interconnects

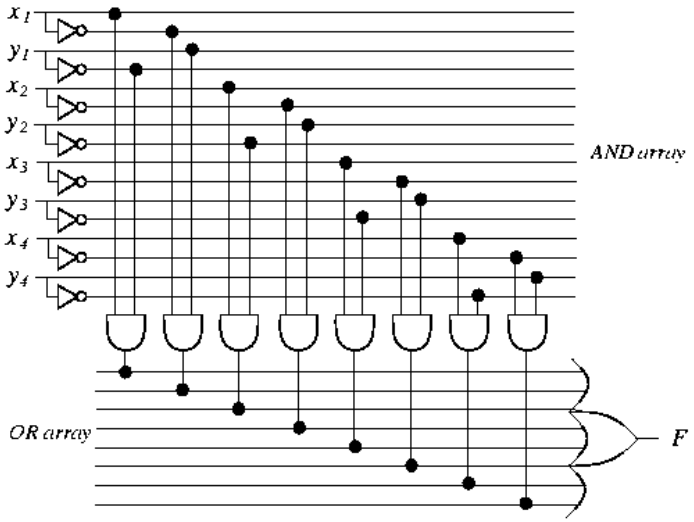
SSI & SPLD Designs



(a) 4–bit comparator.



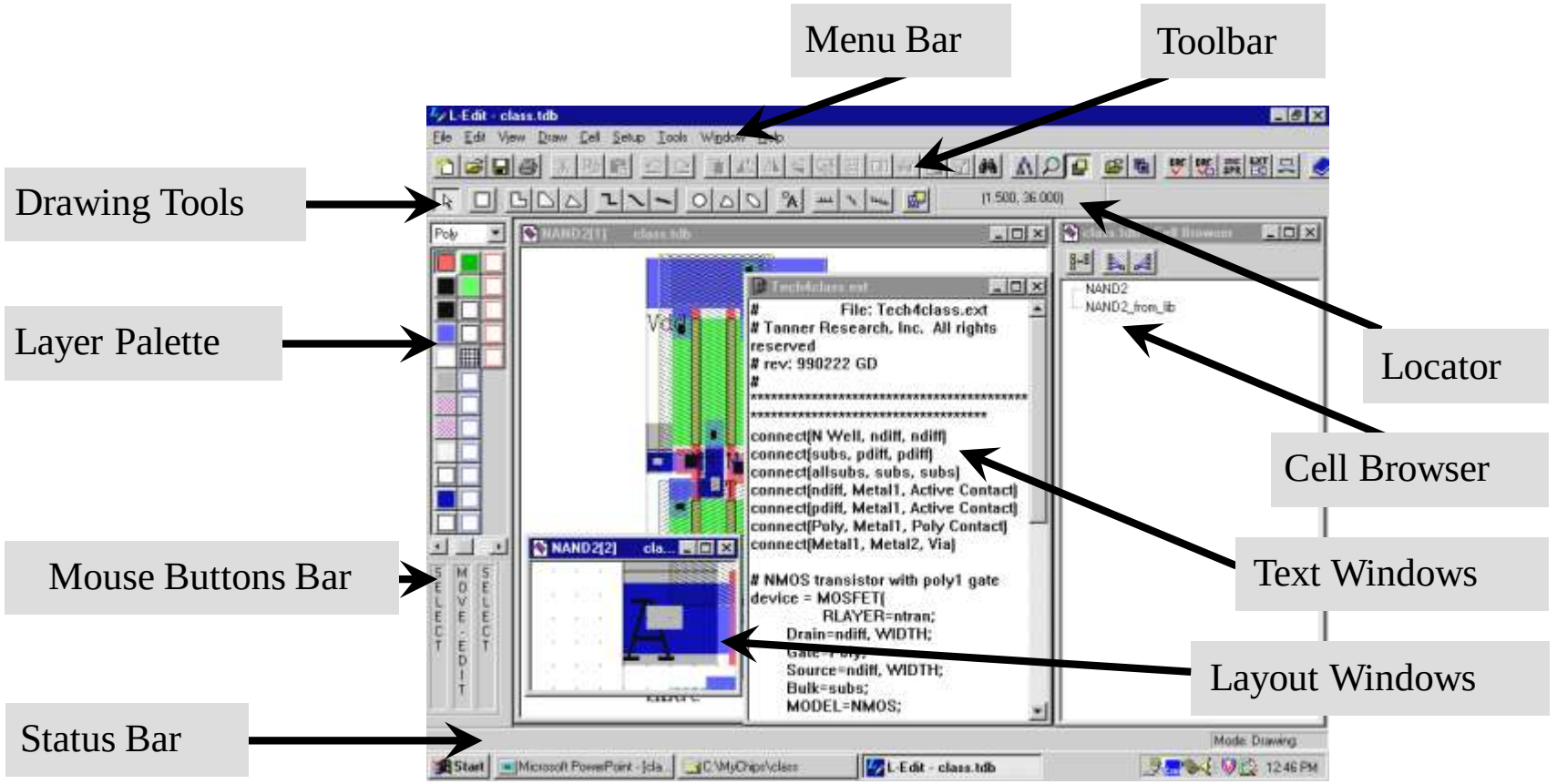
(b) SSI implementation.



(c) SPLD (PLA) implementation.

Full Custom Design

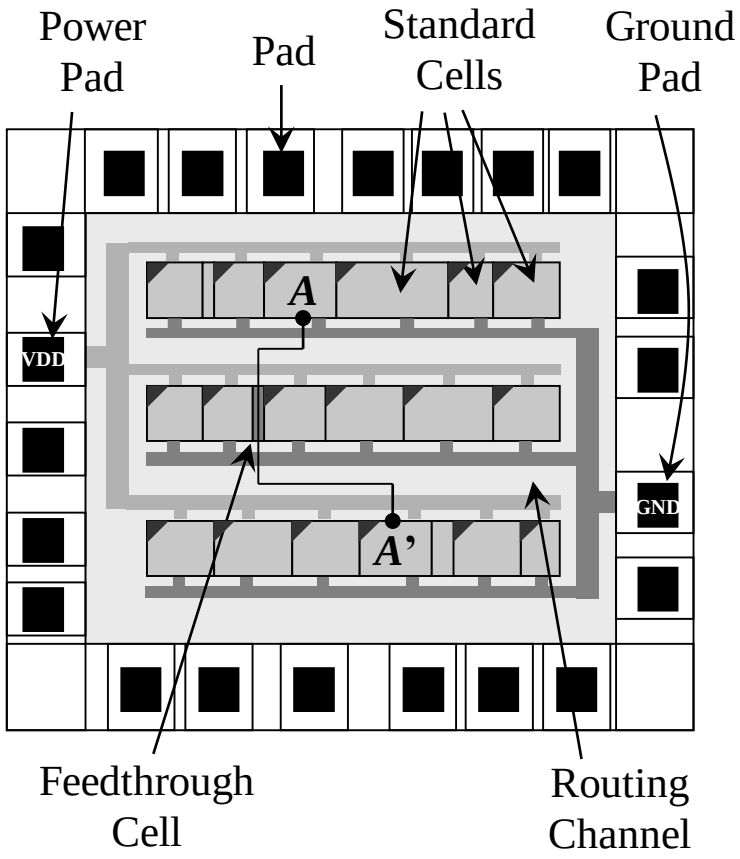
- Layout editor is an essential tool



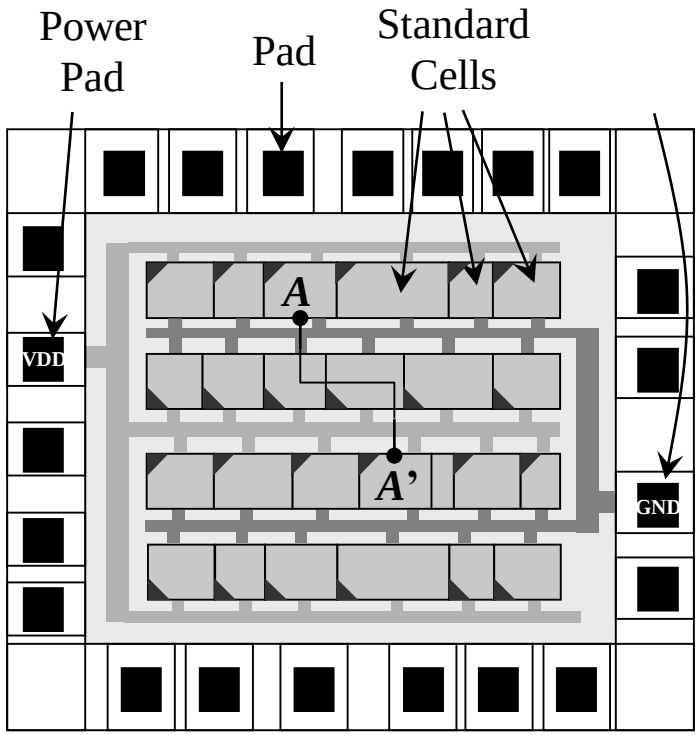
Source: L-Edit, Tanner Research, Inc.

Standard Cell Design Style

Using feedthrough cells

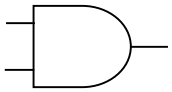


Using over-the-cell (OTC) routing



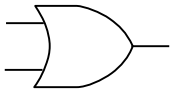
Common Digital Cells

AND



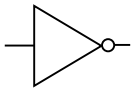
IN1	IN2	OUT
0	0	0
1	0	0
0	1	0
1	1	1

OR



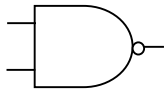
IN1	IN2	OUT
0	0	0
1	0	1
0	1	1
1	1	1

INV



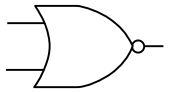
IN	OUT
0	1
1	0

NAND

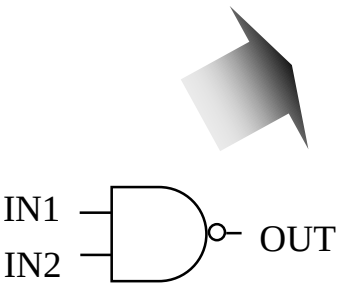


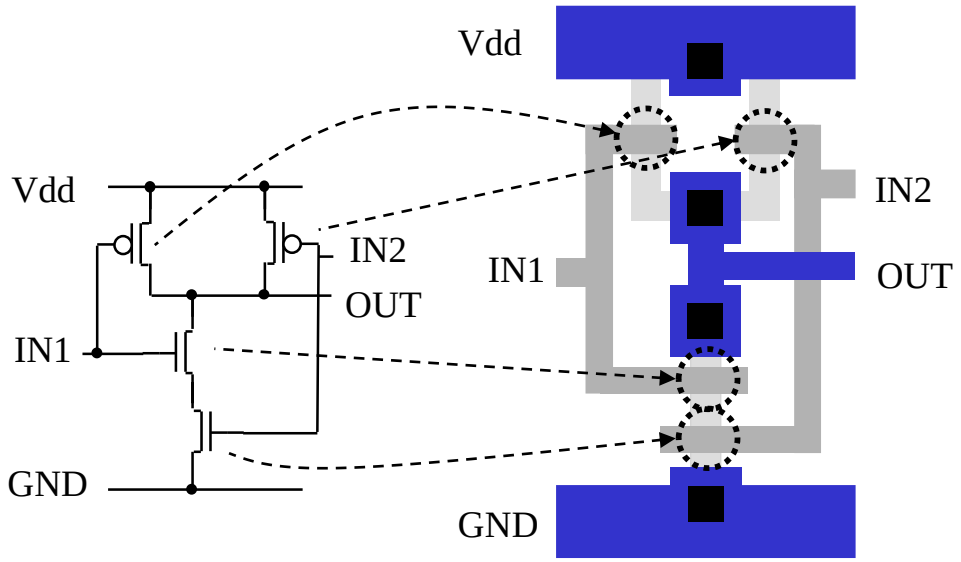
IN1	IN2	OUT
0	0	1
1	0	1
0	1	1
1	1	0

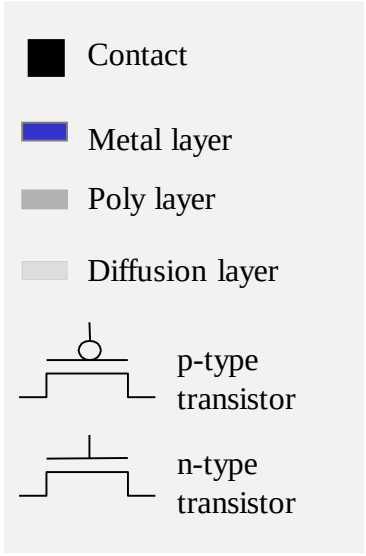
NOR

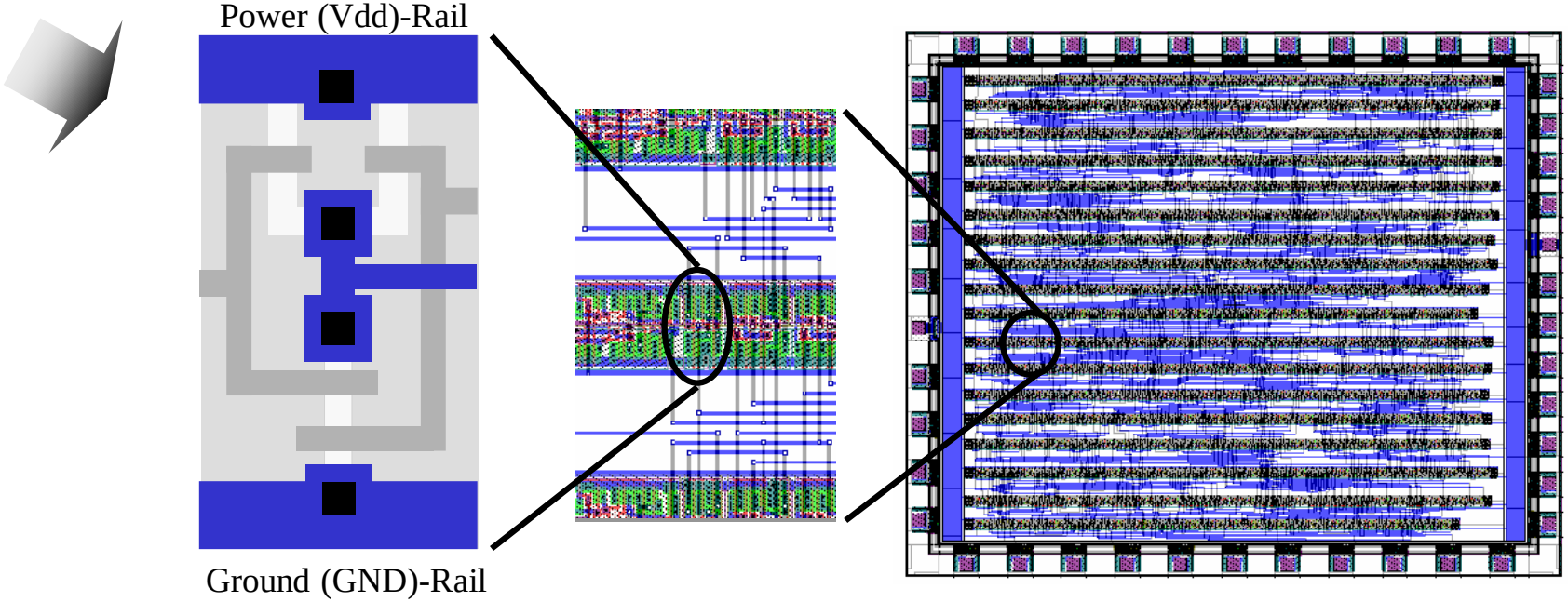


IN1	IN2	OUT
0	0	1
1	0	0
0	1	0
1	1	0

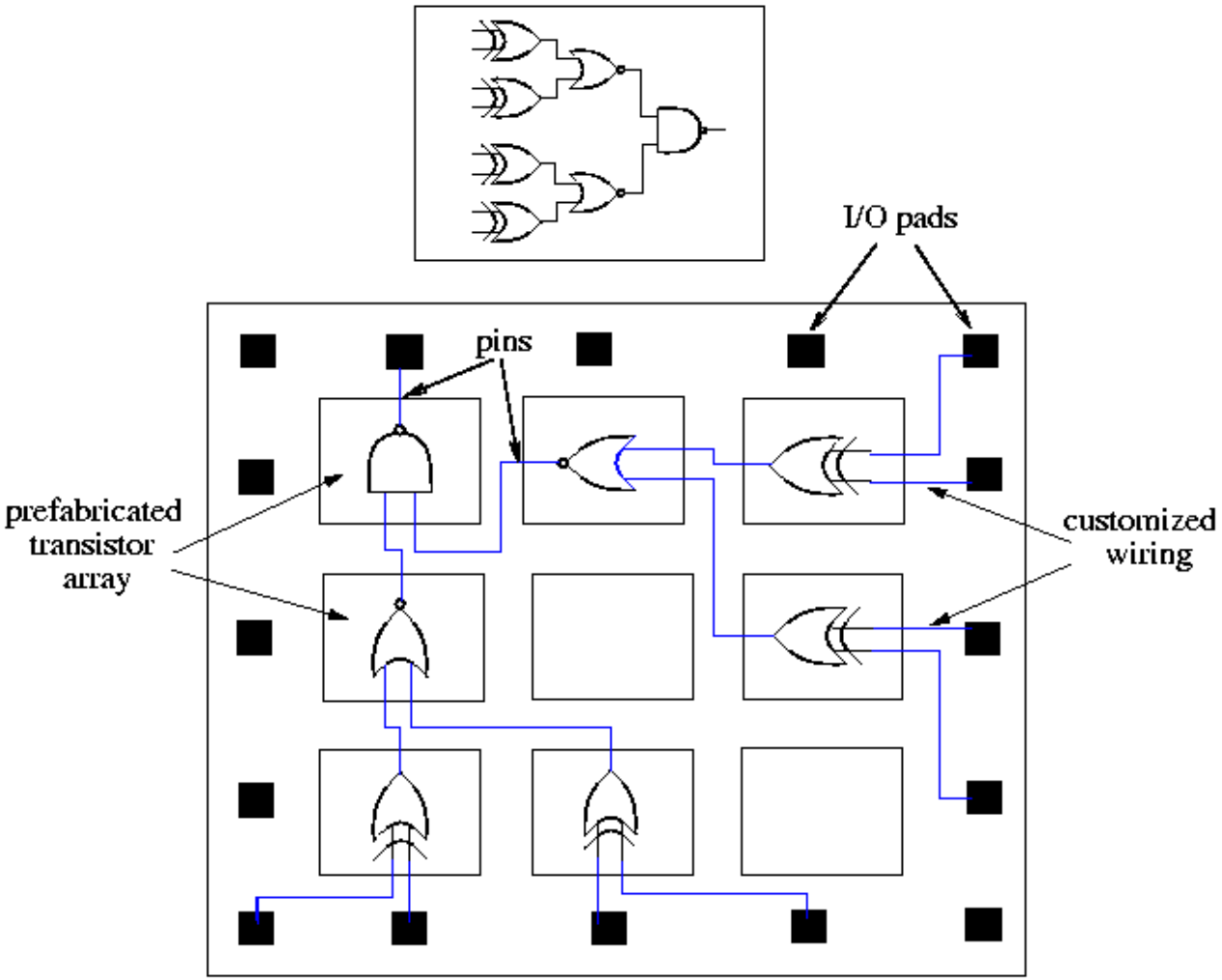

IN1
IN2
OUT


Vdd
Vdd
IN2
OUT
IN1
GND
IN1
IN2
OUT
GND


Contact
Metal layer
Poly layer
Diffusion layer
p-type transistor
n-type transistor

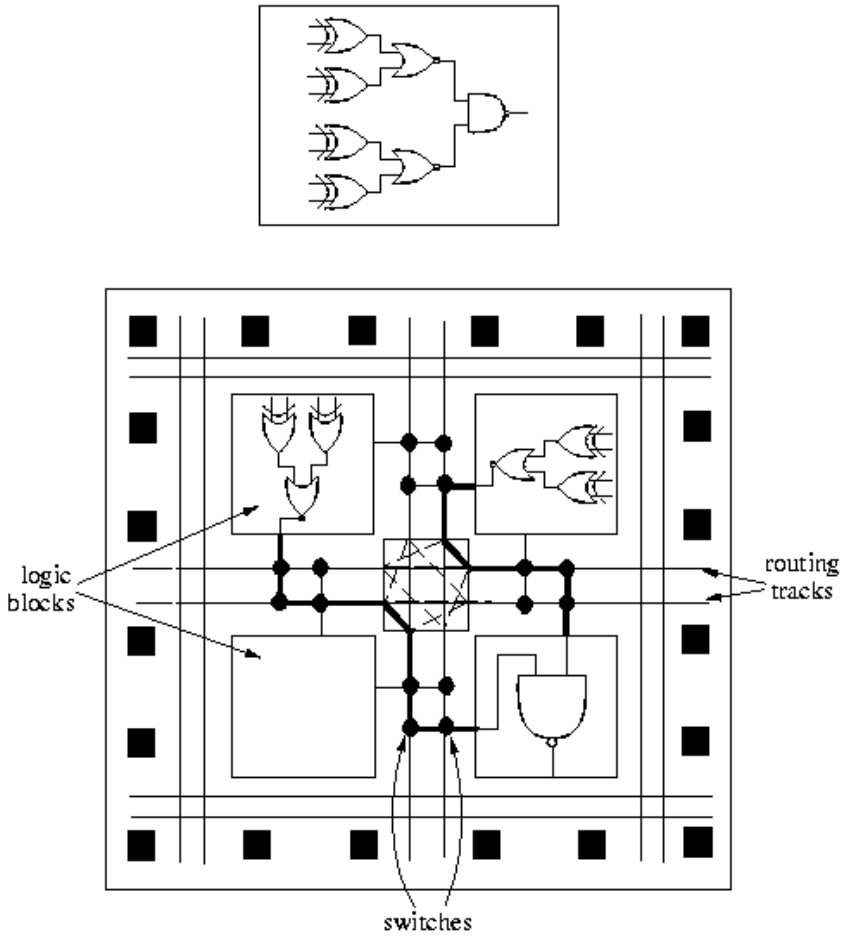

Power (Vdd)-Rail
Ground (GND)-Rail

Gate Array Design



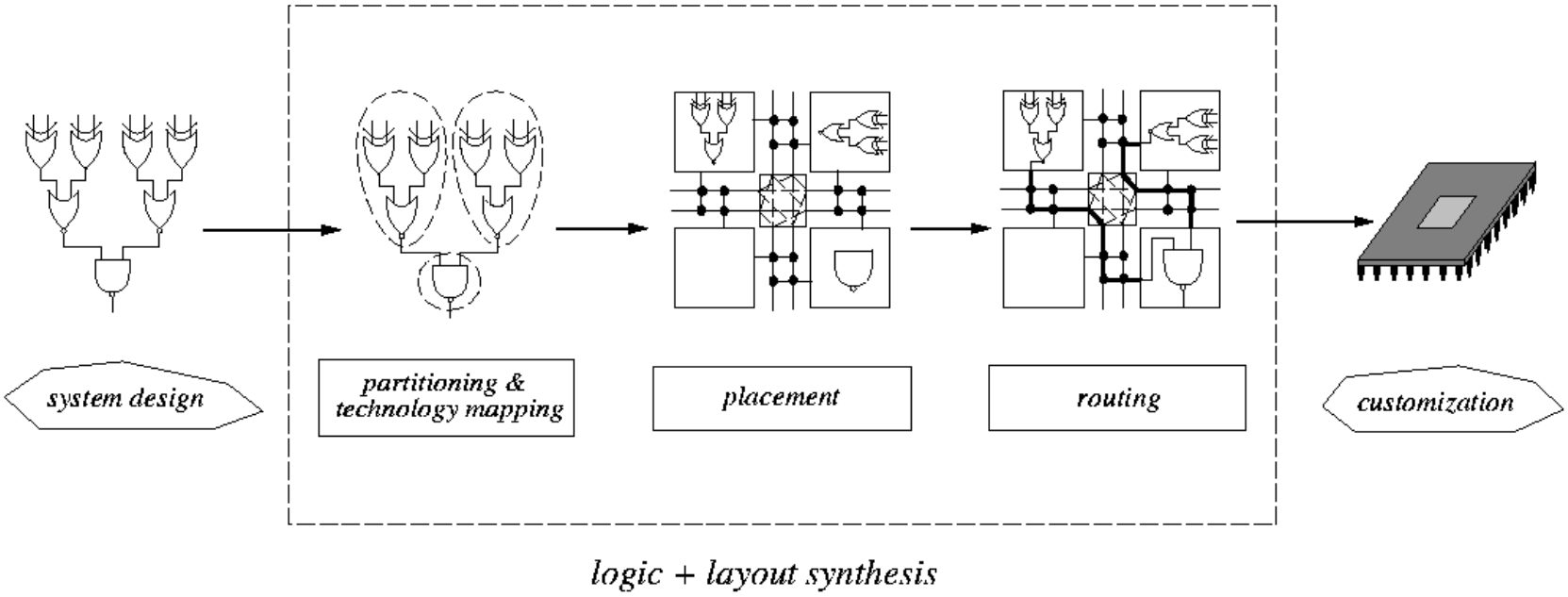
FPGA Design

- Illustrated by a symmetric array-based FPGA:



FPGA Design Process

- Illustrated by a symmetric array-based FPGA:



Comparisons of Design Styles

	Full custom	Standard cell	Gate array	FPGA	SPLD
Cell size	variable	fixed height*	fixed	fixed	fixed
Cell type	variable	variable	fixed	programmable	programmable
Cell placement	variable	in row	fixed	fixed	fixed
Interconnections	variable	variable	variable	programmable	programmable

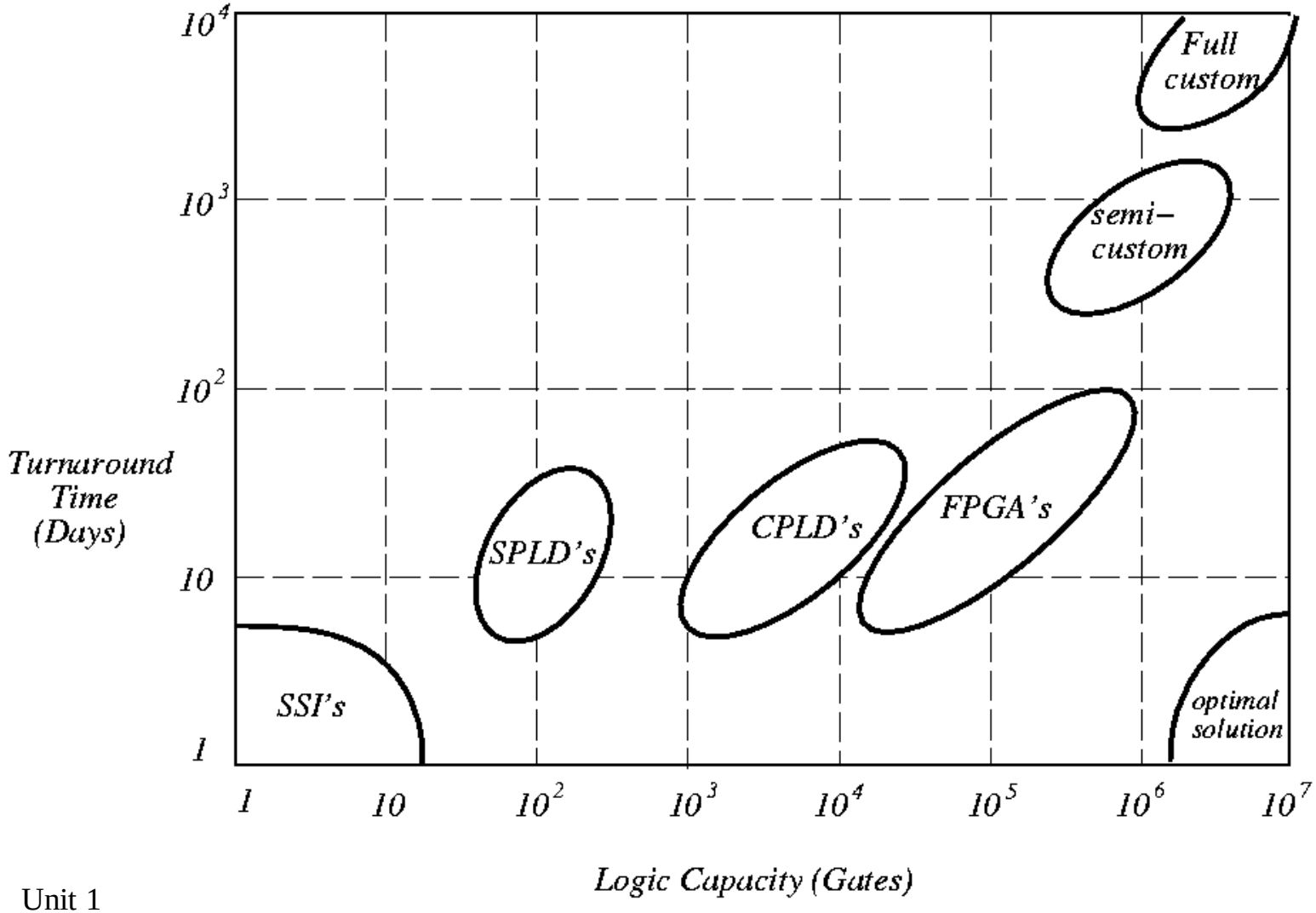
* Uneven height cells are also used.

Comparisons of Design Styles (Cont'd)

	Full custom	Standard cell	Gate array	FPGA	SPLD
Fabrication time	− − −	− −	+	+++	++
Packing density	+++	++	+	− −	− − −
Unit cost in large quantity	+++	++	+	− −	−
Unit cost in small quantity	− − −	− −	+	+++	++
Easy design and simulation	− − −	− −	−	++	+
Easy design change	− − −	− −	−	++	++
Accuracy of timing simulation	−	−	−	+	++
Chip speed	+++	++	+	−	−−

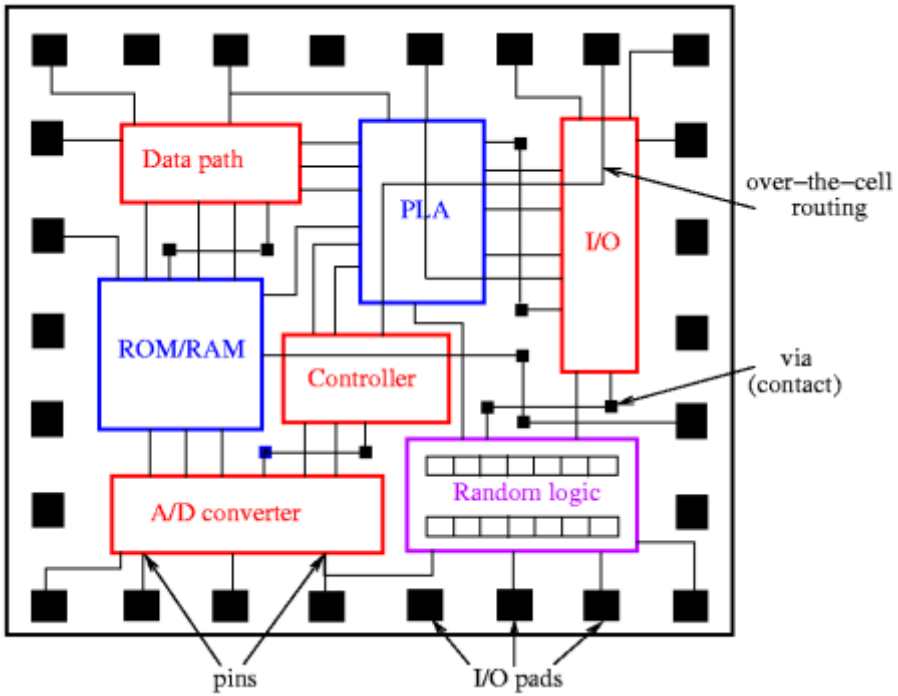
+ desirable; − not desirable

Design Style Trade-offs



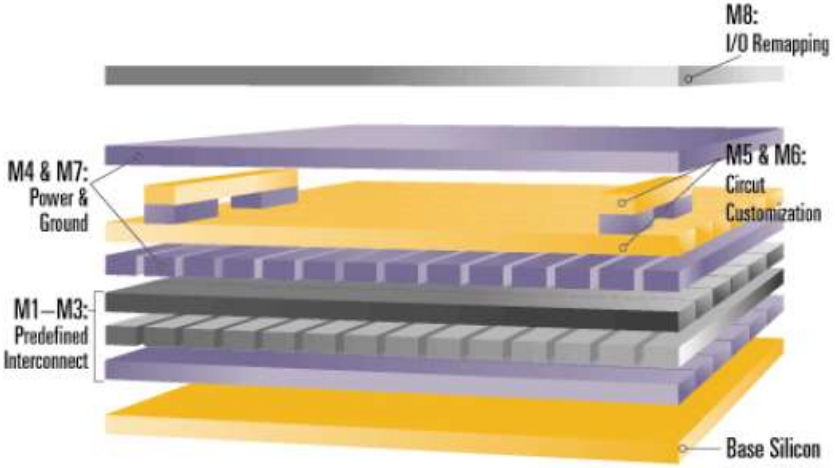
Design with Macro Cells

- Macro cells
  - Are typically large pieces of logic that perform a reusable functionality
  - Range from simple (a couple of standard cells) to highly complex (the scale of an embedded processor or memory block)
  - Vary greatly with respect to shapes and sizes
  - Can be placed anywhere in most cases

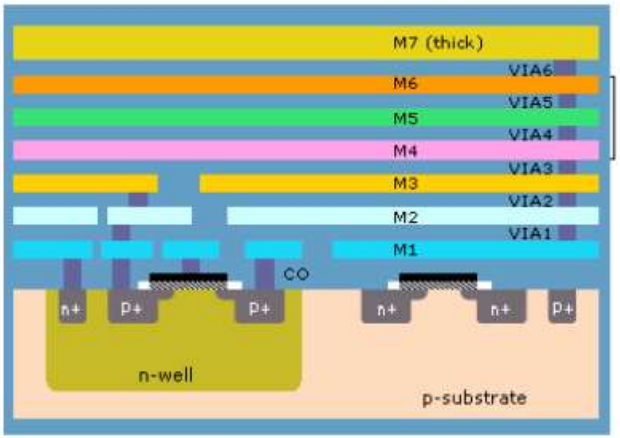


Structured ASIC

- Structured ASIC
  - Consisting of predefined metal and via layers, as well as a few of them for customization
  - Power distribution and local communications on predefined layers
  - Advantages: fewer masks (lower cost), easier physical extraction and analysis
  - Popular for engineering change order (ECO)



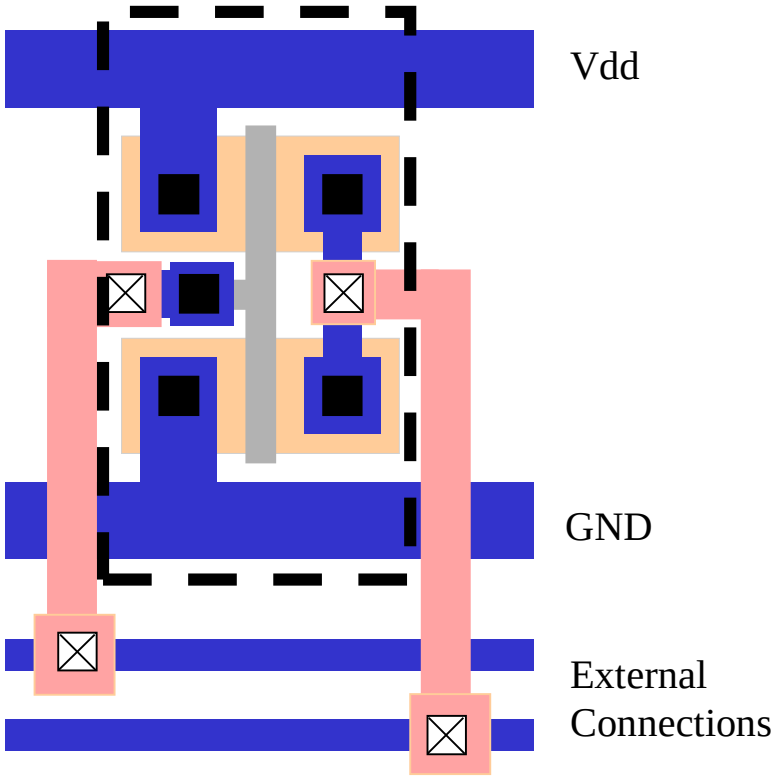
A structured ASIC
(M5 & M6 can be customized)

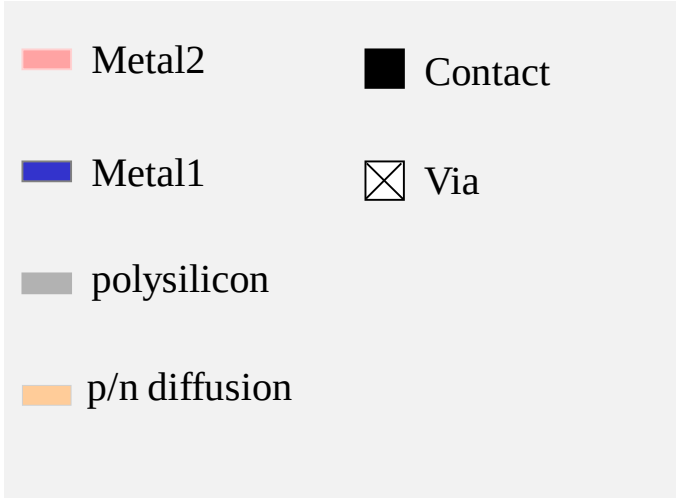


Faraday's 3MPCA structured ASIC
(M4--M6 can be customized)

Layout Layers

- An inverter cell with external connections

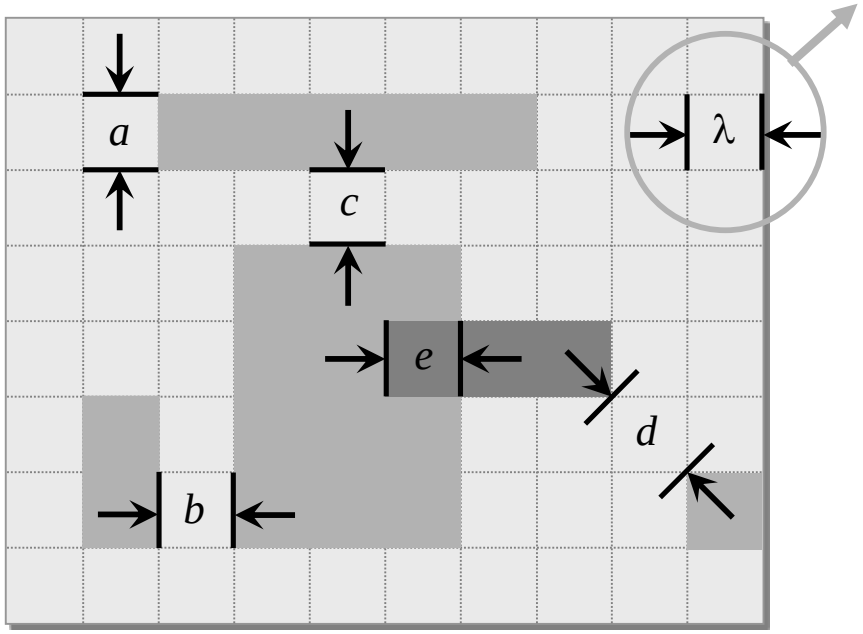




Design Rules

- Size rules (such as minimum width)
  - The dimensions of any component (shape), e.g., length of a boundary edge or area of the shape, cannot be smaller than given minimum values.
  - These values vary across different metal layers.
- Separation rules (such as minimum separation)
  - Two shapes, either on the same layer or on adjacent layers, must be a minimum (rectilinear or Euclidean diagonal) distance apart.
- Overlap rules (such as minimum overlap)
  - Two connected shapes on adjacent layers must have a certain amount of overlap due to inaccuracy of mask alignment to previously-made patterns on the wafer.

Design Rules (Cont'd)



λ: smallest meaningful technology-dependent unit of length

Minimum Width: *a*

Minimum Separation: *b*, *c*, *d*

Minimum Overlap: *e*

Constraint Types

- During physical design optimization, three types of constraints must be satisfied:
  - Technology constraints
    - Enabling fabrication for a specific technology node and derived from technology restrictions
    - Examples: minimum layout widths and spacing values between layout shapes
  - Electrical constraints
    - Ensuring the desired electrical behavior of a design
    - Examples: meeting maximum timing constraints for signal delay and staying below maximum coupling capacitances
  - Geometry (design methodology) constraints
    - Introduced to reduce the overall complexity of the design process
    - Examples: preferred wiring directions during routing, and the placement of standard cells in rows