Outline

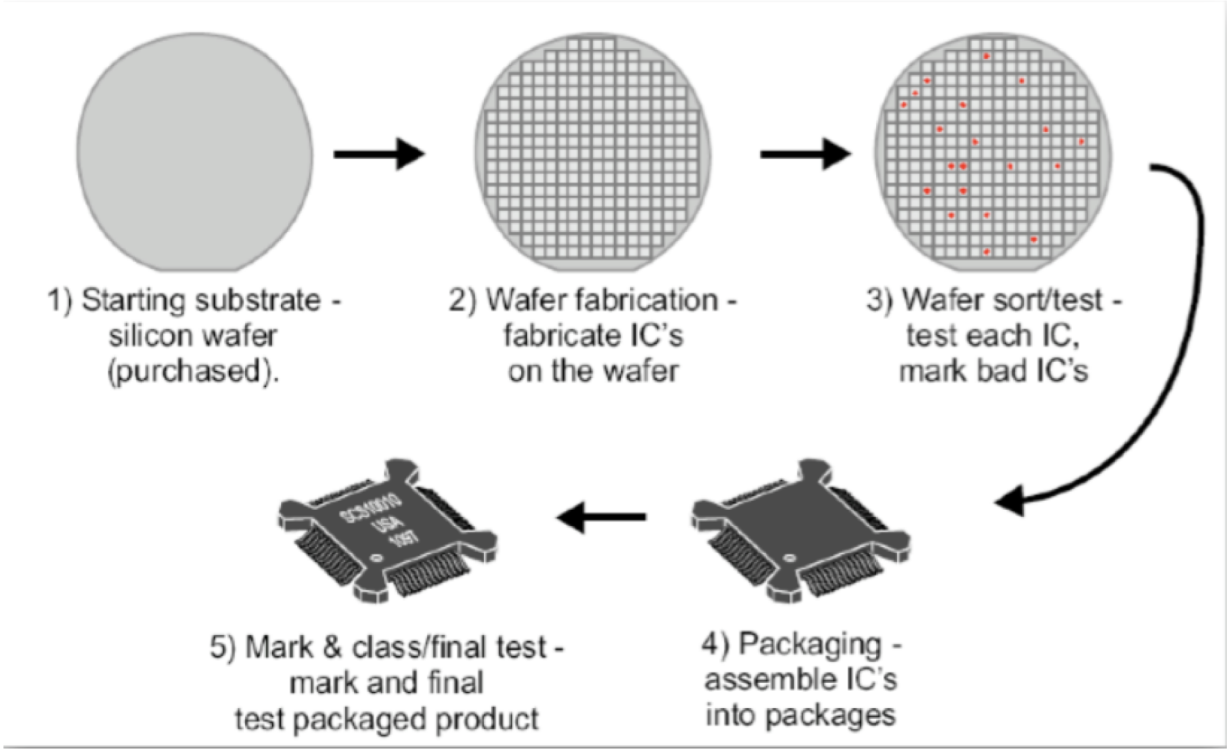
- Brief IC history
- **CMOS fabrication**
- Design partitioning
- Simple example of custom design

Measure of IC

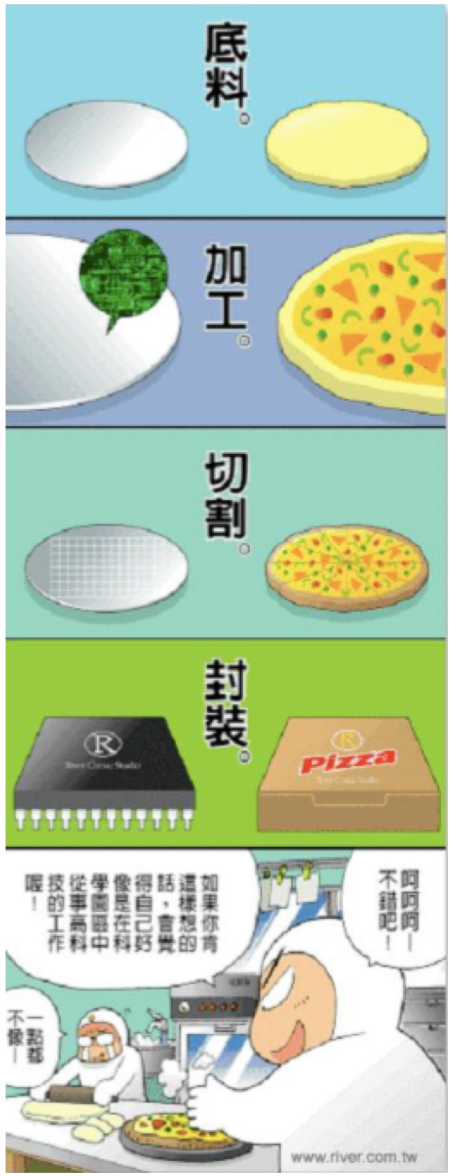
- Gate equivalent, two-input NAND gate
  (of 4 transistors)

- Terms
  - SSI (small-scale IC: 60's): 10 gates
  - MSI (medium-scale IC: early 70's): 1000 gates
  - LSI (large-scale IC: late 70's): 10000 gates
  - VLSI (very-large-scale IC: early 80's): > LSI
  - ULSI (ultra-large-scale IC): millions gates

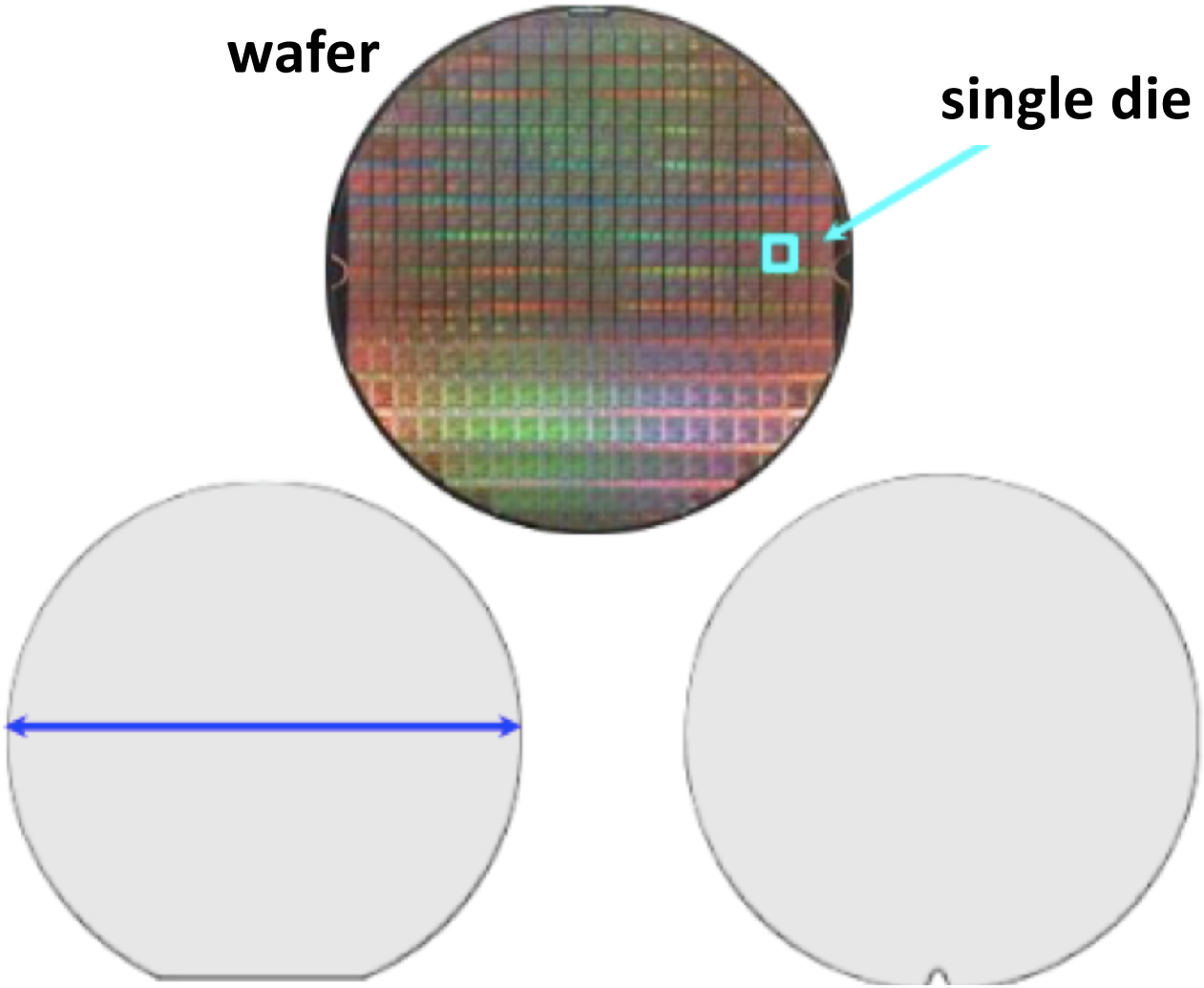
Wafer Fabrication Process



Ref: http://www.icknowledge.com



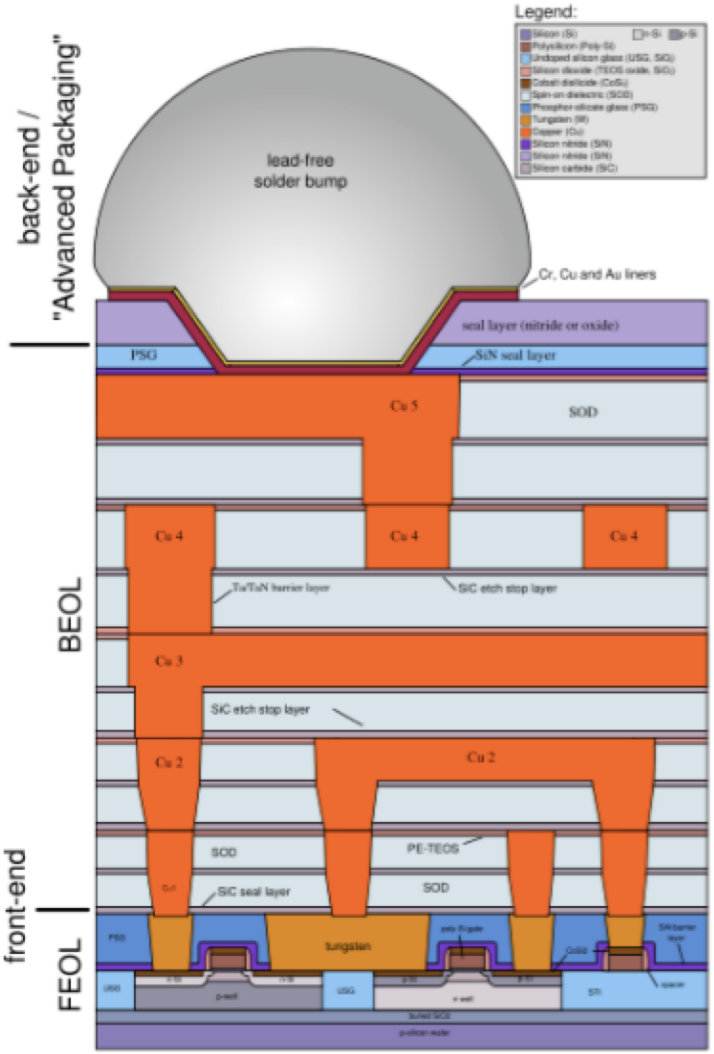
Wafer and Die

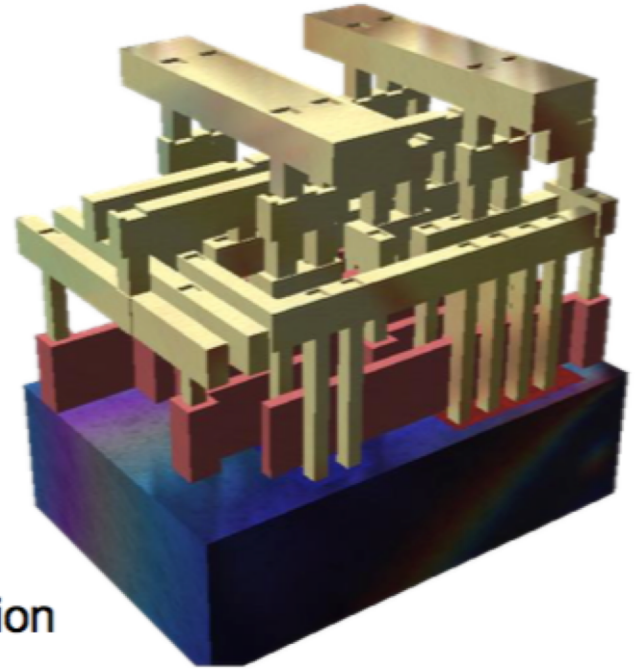


wafer with flat < 150 mm (6”)

wafer with notch > 200 mm (8”)

Integrated Circuit Cross-Section

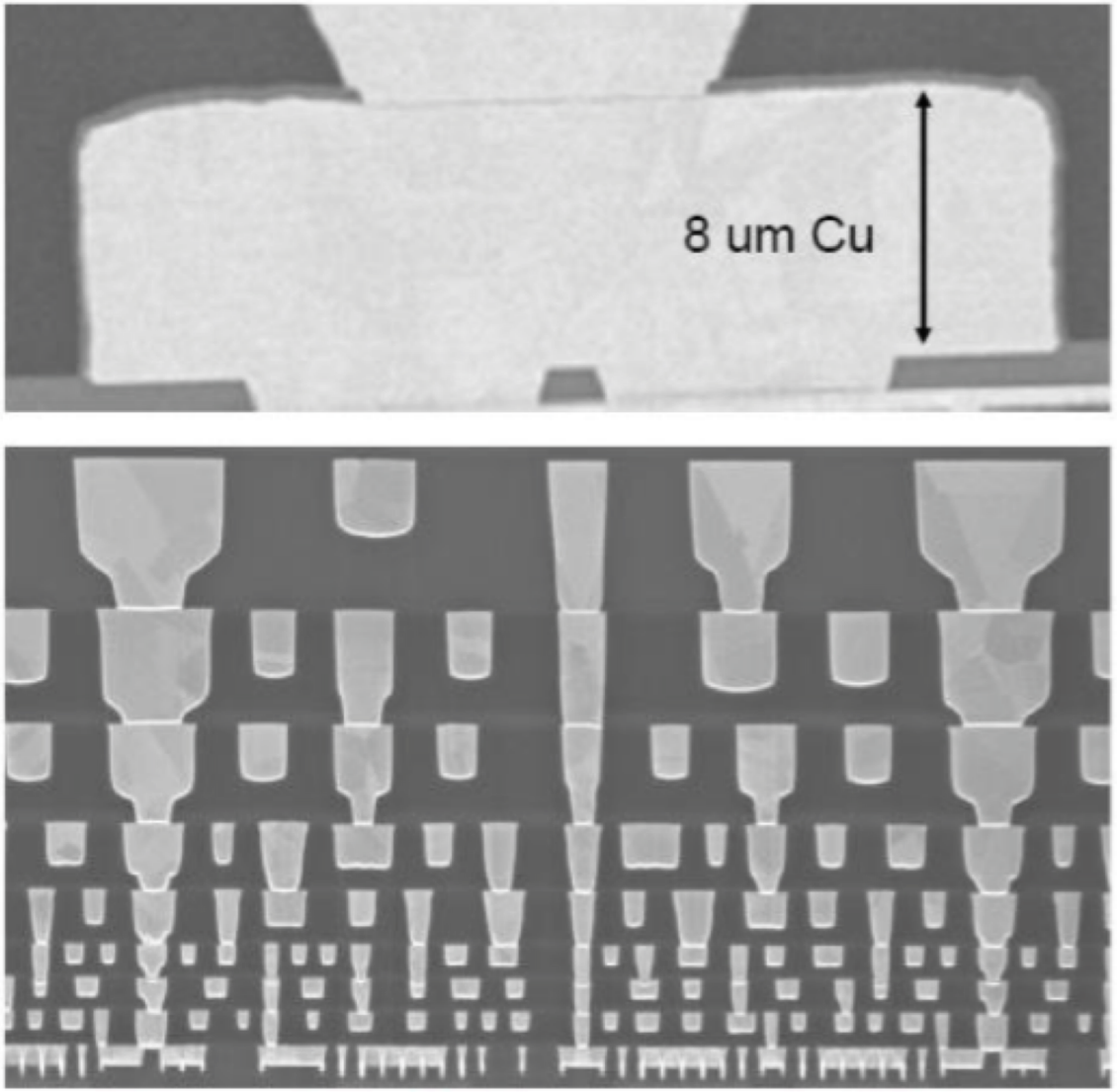




3D model :
3 metal layers interconnection

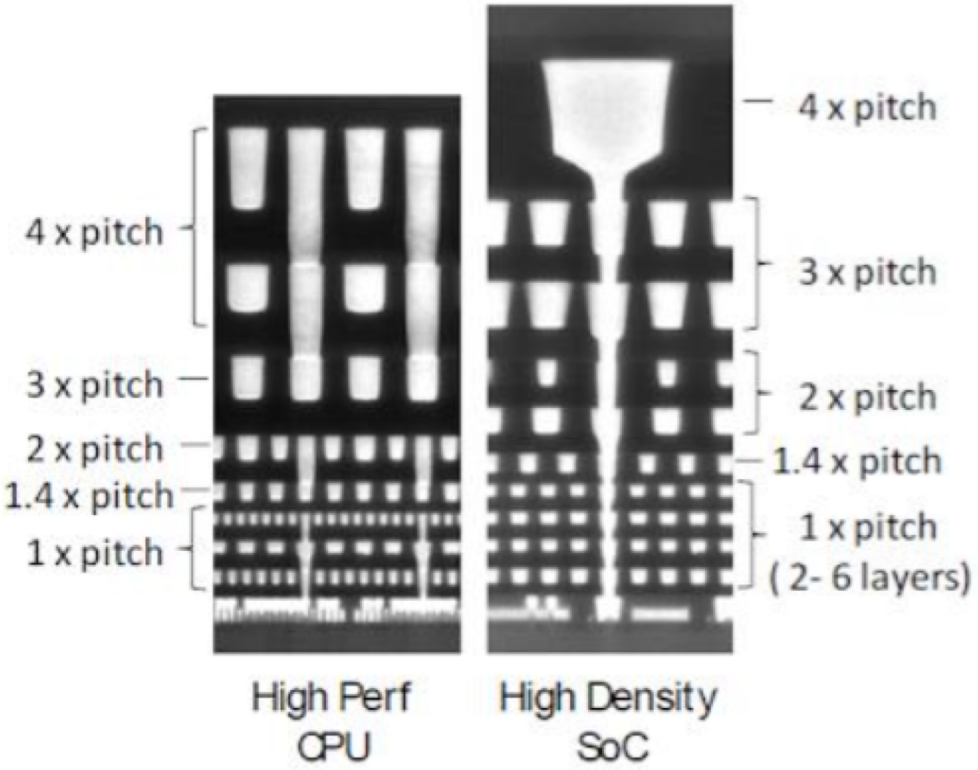
Ref: http://en.wikipedia.org/wiki/integrated_circuit

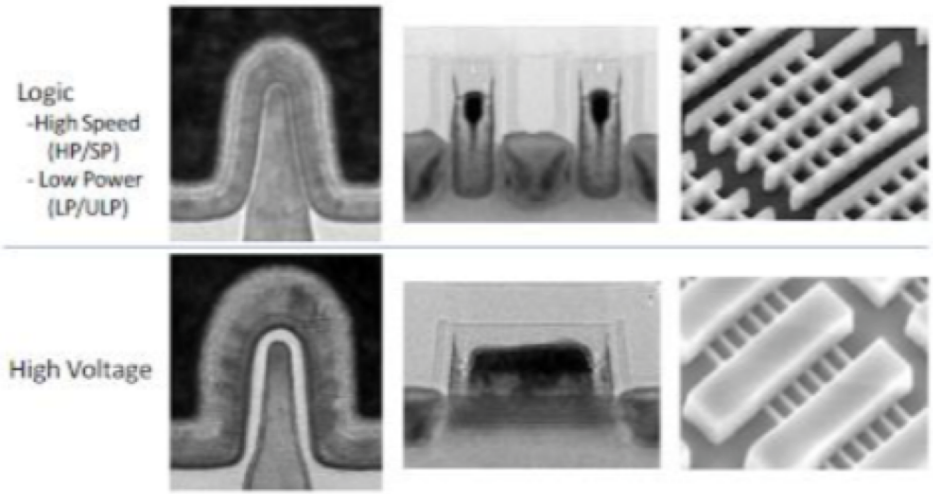
32 nm Interconnect



Hierarchical interconnect pitches

Intel 22 nm Trigate Process





Cost of Integrated Circuits

- Non-recurrent engineering (NRE) costs
  - One-time cost factors
  - **Design effort, design time, mask generation**
- Recurrent costs
  - Proportional to chip area and volume
  - **Silicon processing, packaging, testing**

Fabrication Cost

Year	1999	2002	2005	2008	2011	2014
Feature size (nm)	180	130	100	70	50	35
Chip size (mm^2)	170	214	235	269	308	354
Logic trans/Chip (M)	15	60	235	925	3,650	14,400

- Cost of building factories increases 2X every 3 years
  - By 2010 a fab (12”) may cost $3 billion
  - By 2014 (18”) it may cost 8~10 billion
- Mask costs are growing rapidly adding more to upfront NRE for new designs
  - Next-generation lithography methods require expensive complex masks (optical proximity correction (OPC) and phase shift (PSM)) to have low error tolerances
  - Multiple masks that require longer write times increase mask production costs

Mask Cost

Technology	Mask Cost
250nm	$100K *
180nm	$350K *
130nm	$750K *
90nm	$1.5M *
65nm	$3M **
35nm	$6M **

- Double Jeopardy!
  1) The potential for bugs goes up
  2) The cost of re-spinning the chip goes up

→ **You cannot afford hardware bugs!!**

Charles Moore, "Managing the Transition from Complexity to Elegance", ISCAS 2003

Design Productivity Crisis

Year	Tech. (nm)	Complexity	Frequency	3 Yr. Design Staff Size	Staff Costs
1997	350	13 M Tr.	400 MHz	210	$90 M
1998	250	20 M Tr.	500 MHz	270	$120 M
1999	180	32 M Tr.	600 MHz	360	$160 M
2002	130	130 M Tr.	800 MHz	800	$360 M

- 1996: 100 people in P6 team
- 2007: 1600 people in P10 team
- Question: ?? People in P38 team?

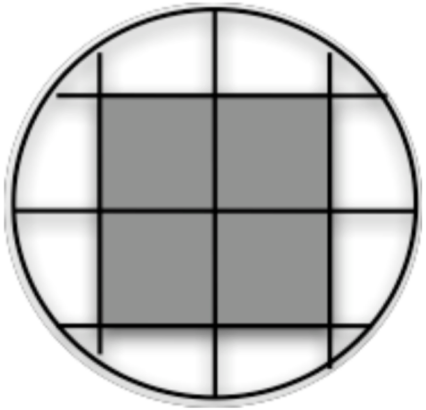
- Improve productivity through design animation

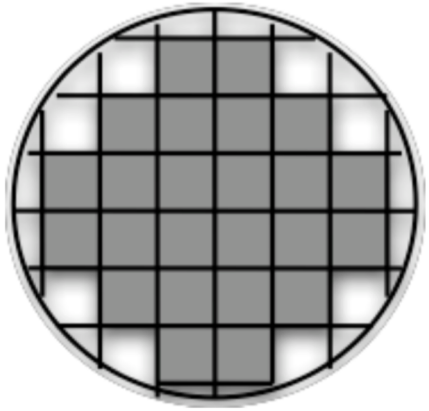
Yield

$$Y = \frac{\text{No. of good chips per wafer}}{\text{Total number of chips per wafer}} \bullet 100\%$$

$$\text{Die cost} = \frac{\text{wafer cost}}{\text{Dies per wafer} \times \text{Die yield}}$$

$$\text{Dies per wafer} = \frac{\pi \times (\text{wafer diameter}/2)^2}{\text{die area}} - \frac{\pi \times \text{wafer diameter}}{\sqrt{2 \times \text{die area}}}$$



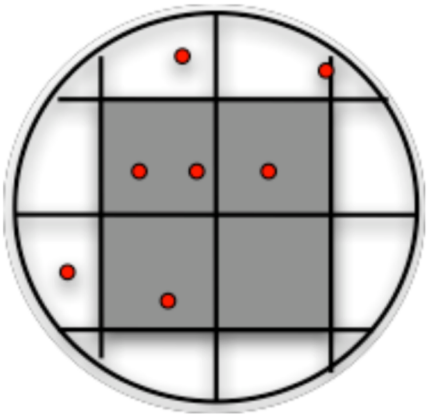


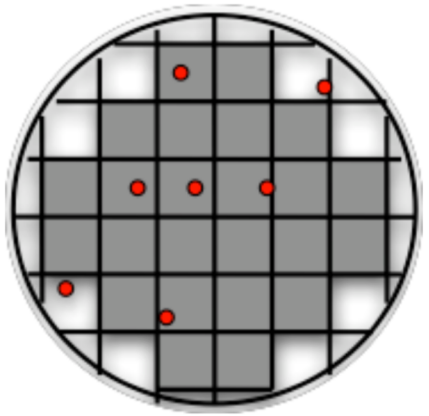
Defects

$$\text{Die yield} = \left(1 + \frac{\text{defects per unit area} \times \text{die area}}{\alpha}\right)^{-\alpha}$$

$$\text{Die cost} = \frac{\text{wafer cost}}{\text{Dies per wafer} \times \text{Die yield}}$$

$$\text{if } \alpha = 3, \text{ die cost} = f(\text{die area})^4$$





Yield Example

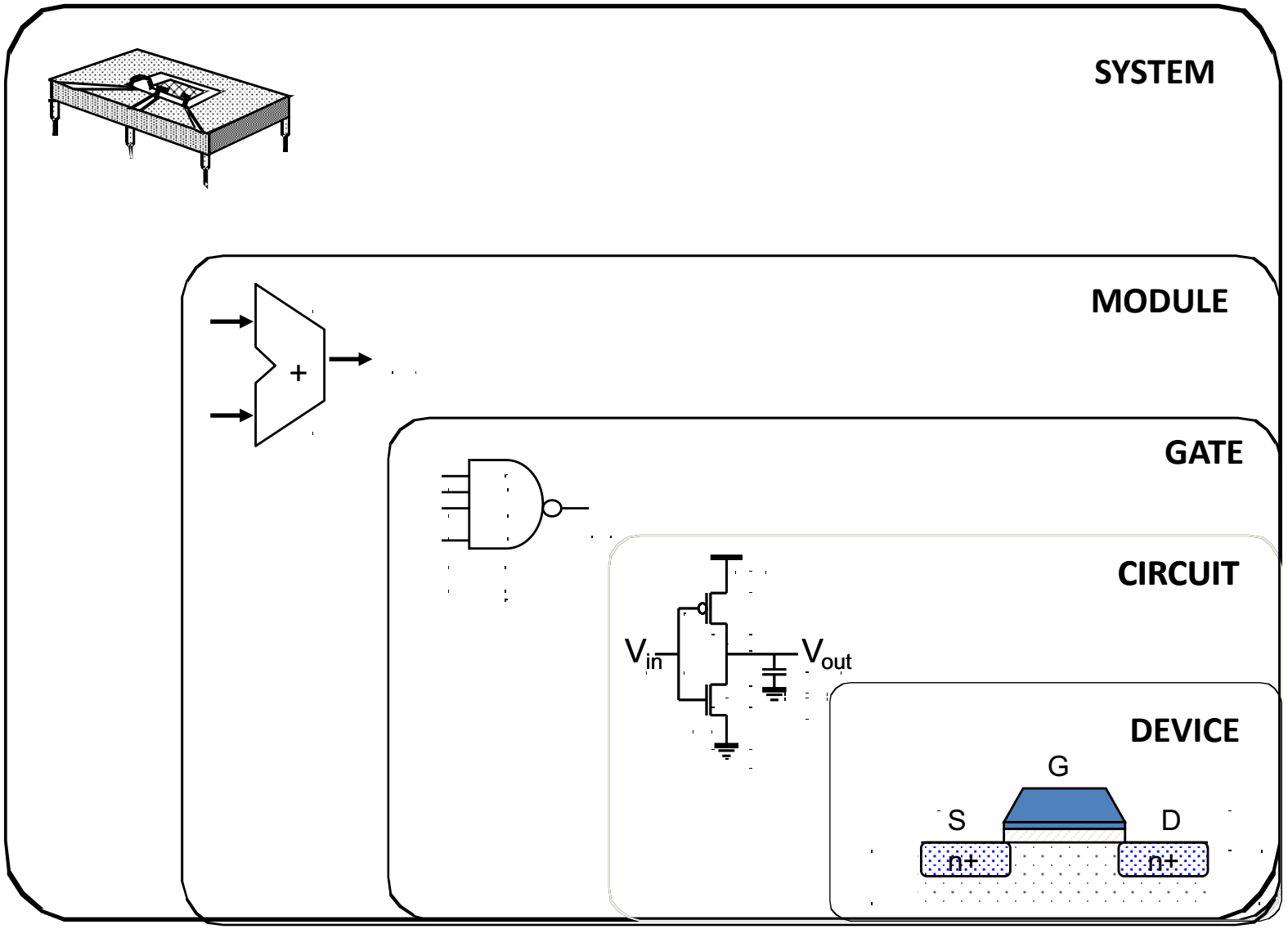
- Wafer size of 12”, die size of 2.5 cm^2, 1 defects/cm^2, $\alpha = 3$ (measure of manufacturing process complexity)
- 252 dies/wafer
  - Wafers are round & dies are square
- Die yield of 16%
- 252 x 16% = only 40 dies/wafer die yield!

- Die cost is strong function of die area
  - Proportional to the third or fourth power of the die area

Outline

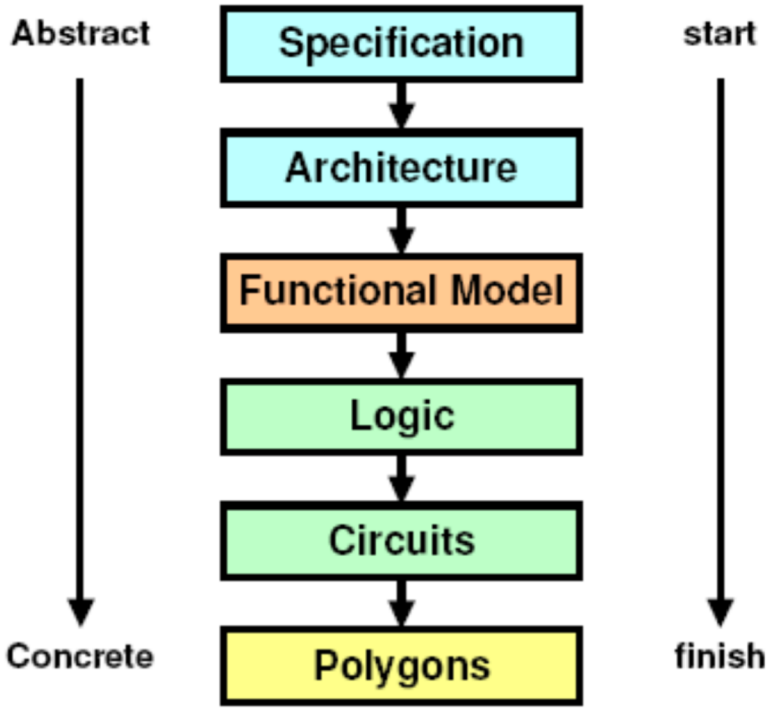
- Brief IC history
- CMOS fabrication
- **Design partitioning**
- Simple example of custom design

Design Abstraction Levels



From Specifications to Hardware

- Target: **spec in → hardware out**
- ***Profitable*** hardware
  - Performance, yield and cost
- Design flow
  - You need good methodology & tools to help
- Validation
  - Functional & power
  - Testing efficiency & coverage



Design Metrics

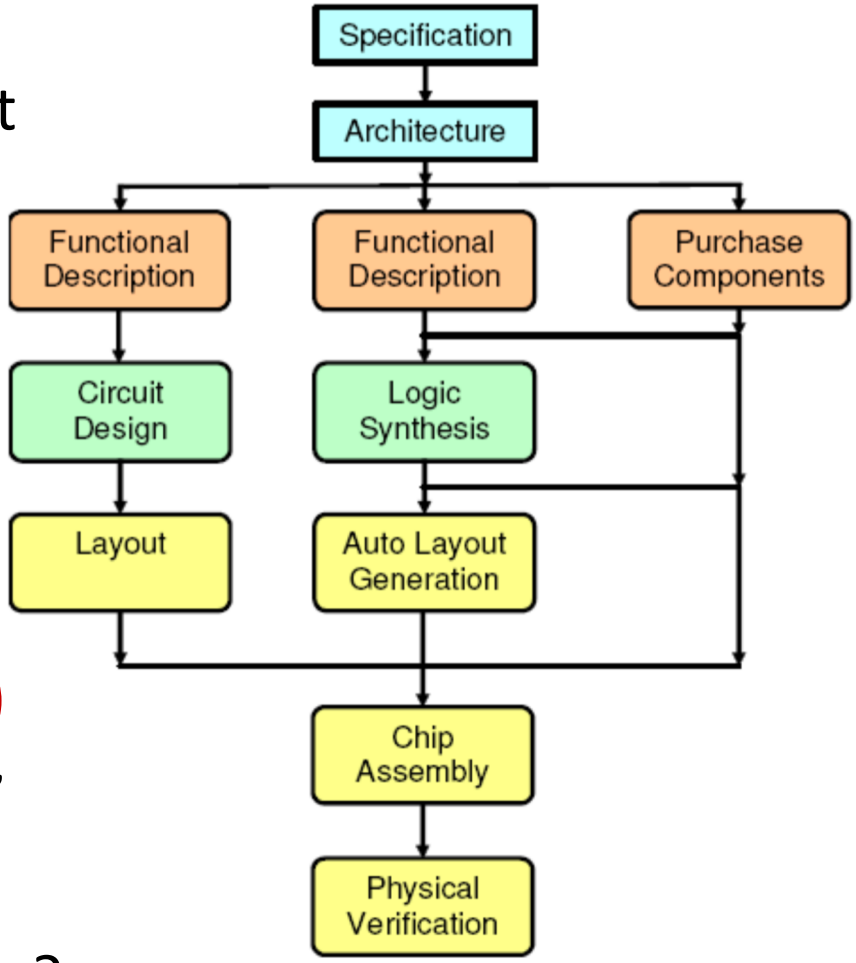
- How to evaluate performance of a circuit?
  - Speed (delay, operating frequency)
  - Design complexity
  - Power dissipation
  - Energy to perform a function
  - Cost (NRE and recurring)
  - Reliability (noise margin and immunity)
  - Scalability
  - Time-to-market

Design Types

- **FPGA design** (EE2230 Logic Design Lab)
  - Configurable logic blocks, interconnects and I/Os
- **Gate array design**
  - Two step manufacturing, shorter silicon finished time
- **Cell-based design** (EE4292 IC Design Lab)
  - "Describe-and-synthesize" paradigm
- **Full-custom design** (EE5250, EE4292 IC Design Lab)
  - "Capture-and-simulate" paradigm
- **SoC design**
  - Platform-based design (reuse)

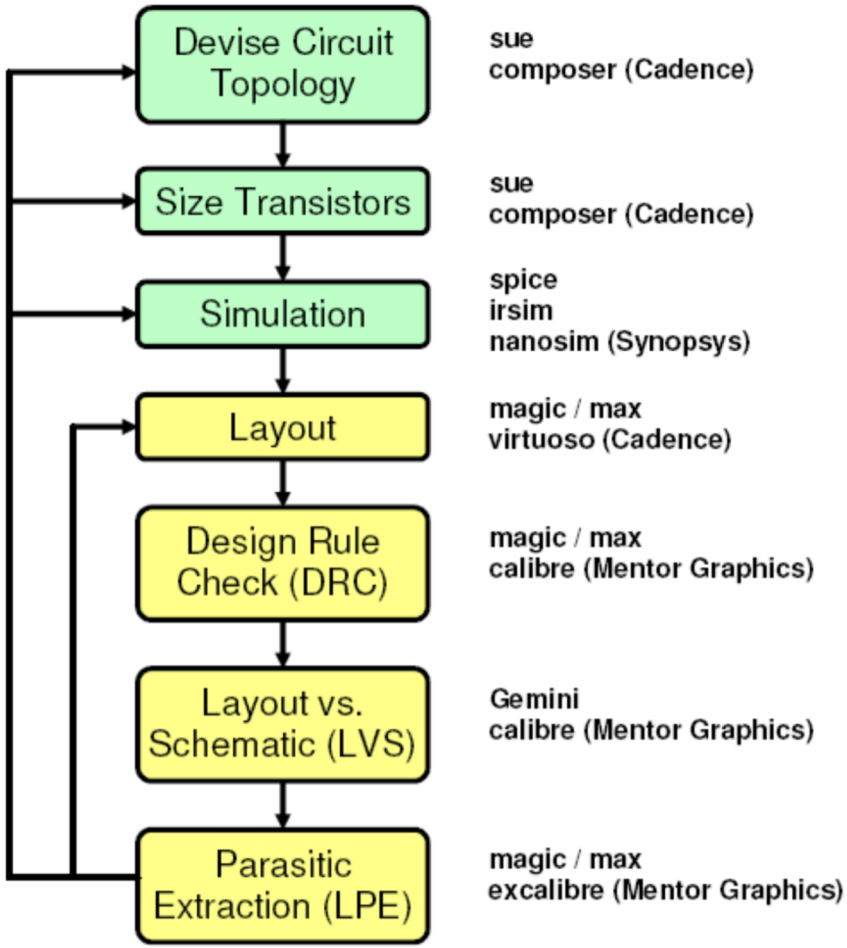
Design Flow

- **Left:** Full Custom
  - Performance and area driven
  - Hand-made design and layout
- **Center:** ASIC (Application-Specific Integrated Circuit)
  - Cell library + customized interconnect
- **Right:** SoC (System-on-Chip)
  - Silicon "printed circuit board"
  - Higher integration level = better cost and performance?



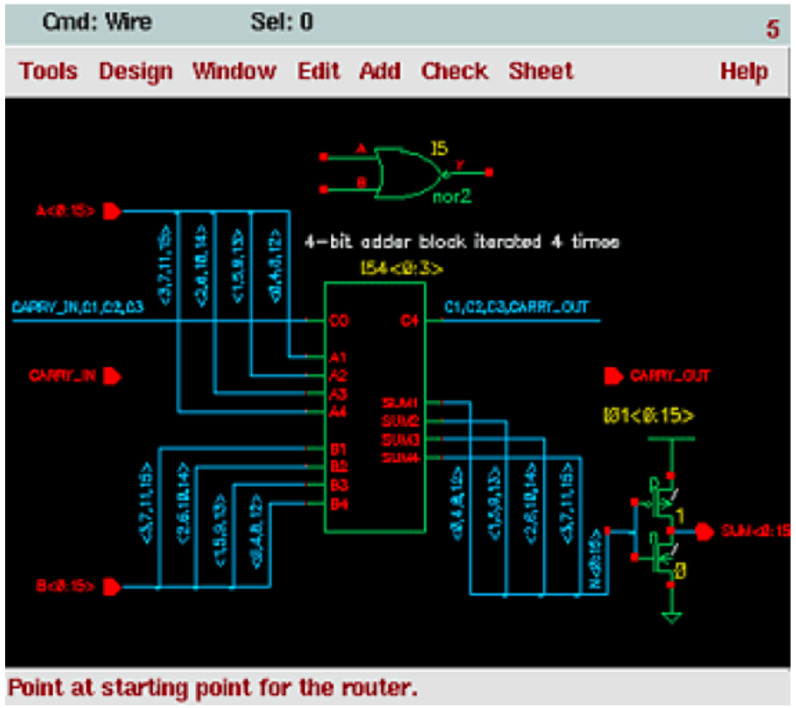
Full-Custom Design Flow

- Analog or mixed-mode design flow
  - Customized function, technology, shape and spec
- Pros
  - Optimized area and performance
- Cons
  - Time and human source
  - May not re-usable



Schematic and Simulation

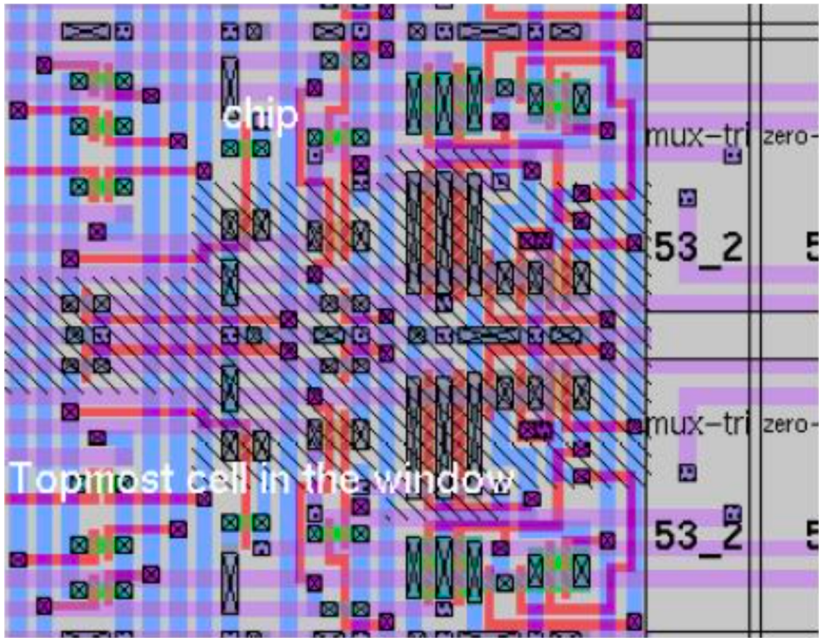
- GUI schematic entry
  - MOS (Transistor) → Gate
    → Sub-circuit (Symbol)
    → Full-chip (System)
  - Easy for hieratical design and debugging
- Netlist Generation
  - For SPICE simulation and LVS
  - **DO NOT** use text editor to create your netlist



Cadence Virtuoso Schematic Composer

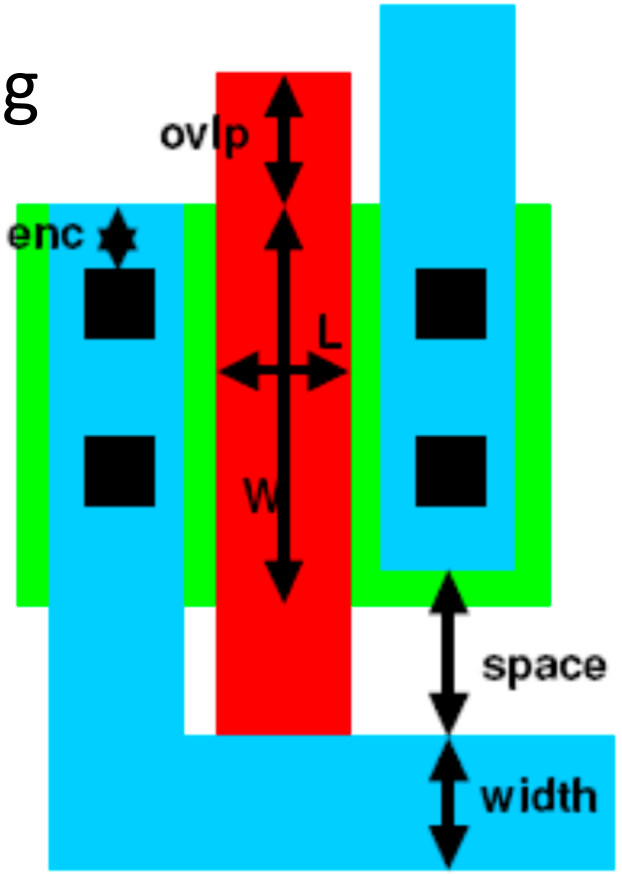
Layout

- Polygons entry
  - Transformed from device netlist to polygons
  - Keep device's 3D cross section view in mind when drawing 2D polygons
  - Try to match schematic's hierarchy
  - Need to consider matching, routing parasitics, and coupling coupling, etc.
- Gds file Generation
  - For MASK layer definition



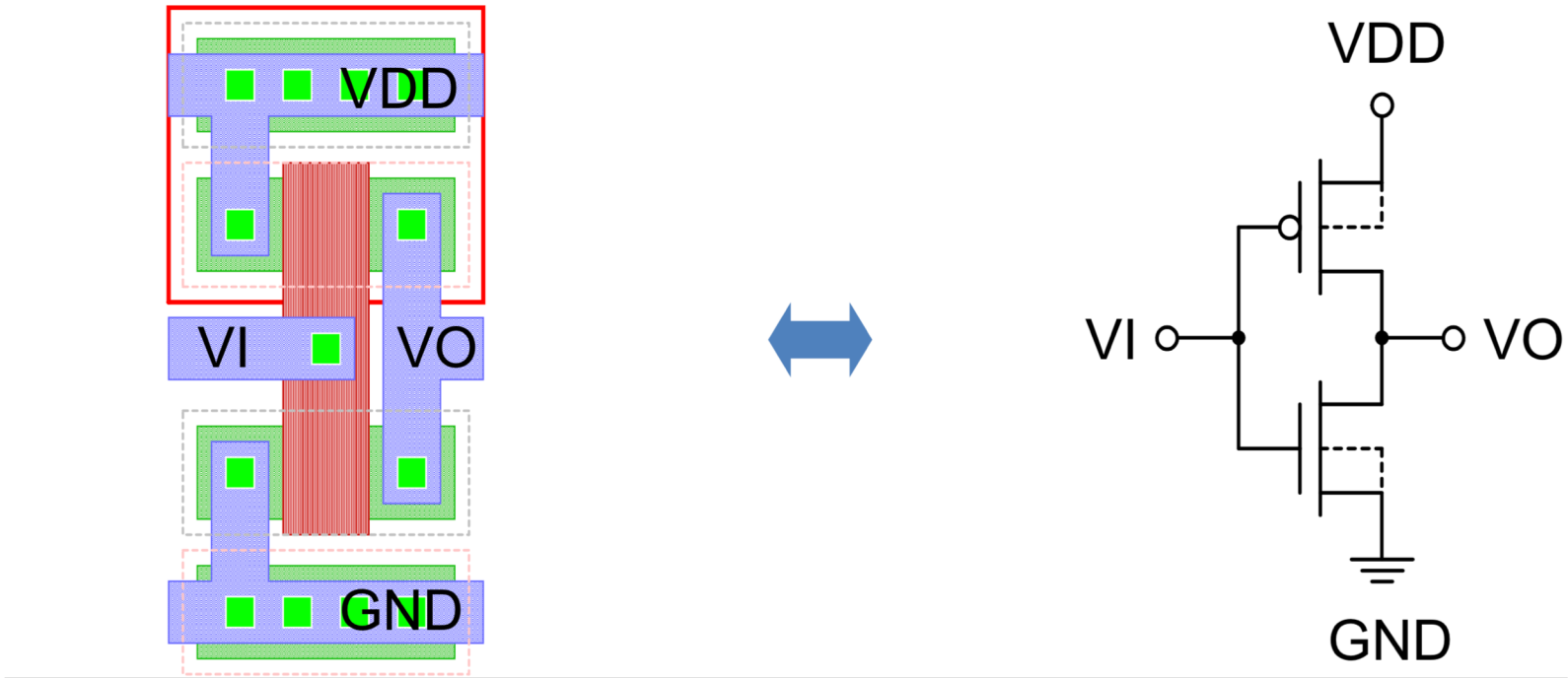
Design Rule Check (DRC)

- Design rules are defined by foundry for semiconductor manufacturing process
- Study DRC rules before start Layout
- Need to know the physical meaning
- Design rule violation is **NOT ALLOWED** unless you are doing some experiments



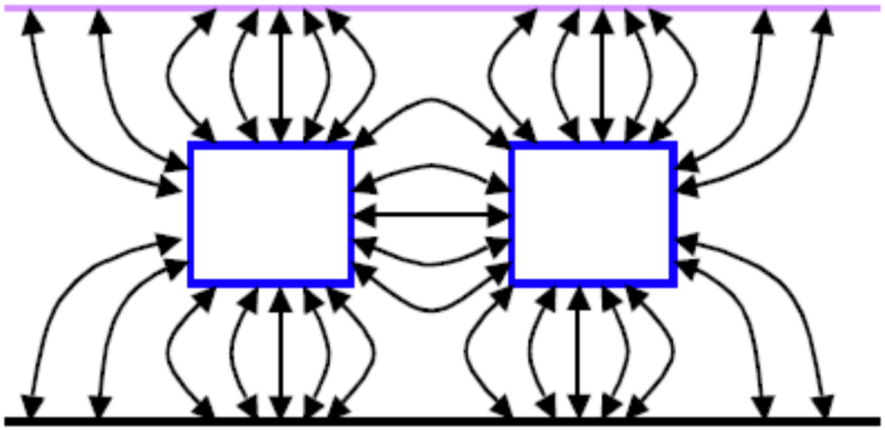
Layout vs. Schematic (LVS)

- Make sure the layout matches your schematics
  - Debugging requires good methodology and experience



Layout Parasitic Extraction (LPE)

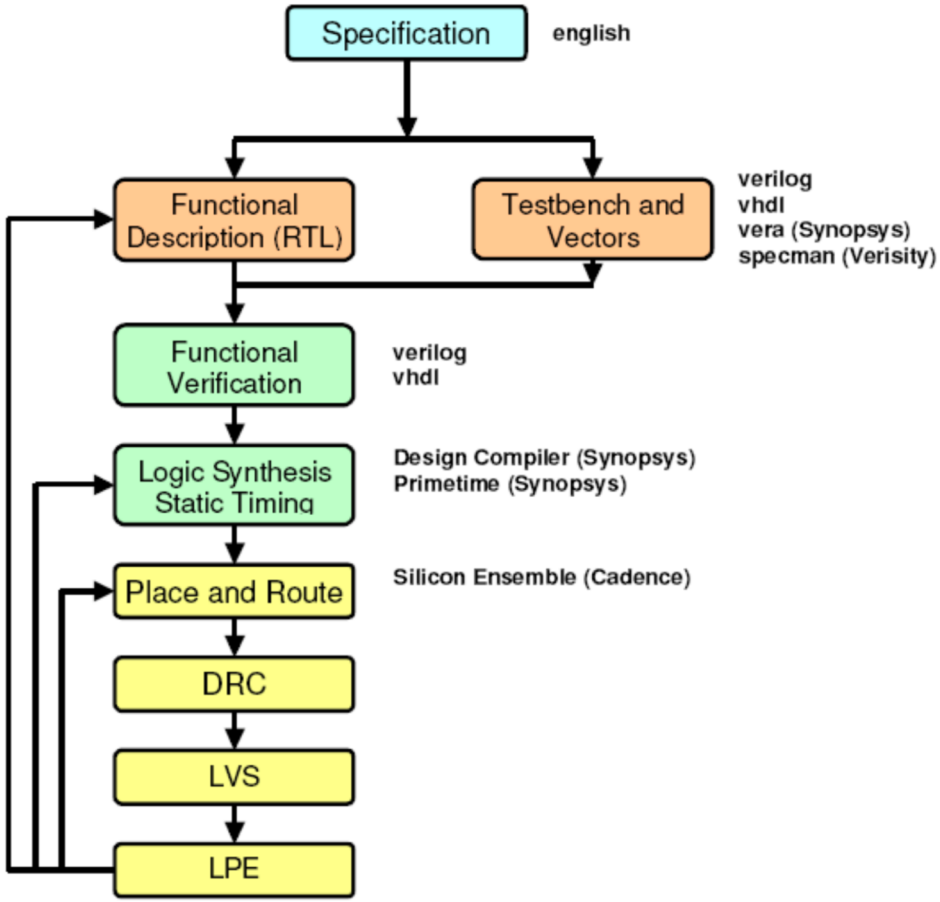
- Extract parasitic resistance and capacitance
- Parasitic: the effect you didn't consider at the the first design phase (pre-layout simulation)
- Use LPE netlist to do post-layout simulation and check the close-to-real performance
- Post-simulation is NECESSARY and IMPORTANT



ASIC Design Flow

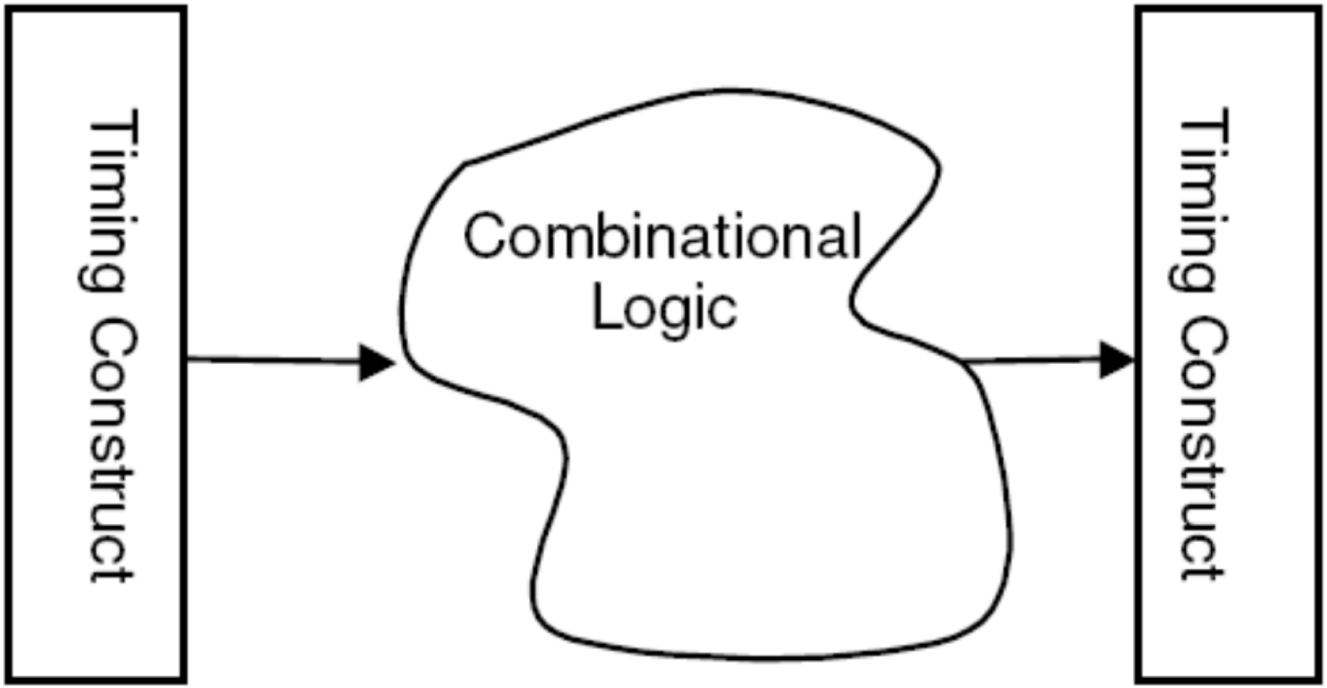
- Digital design dlow
- Design based on standard cell library

- Pros
  - Reliable and expectable chip performance
- Cons
  - Not area and performance optimized
  - Need IP cost



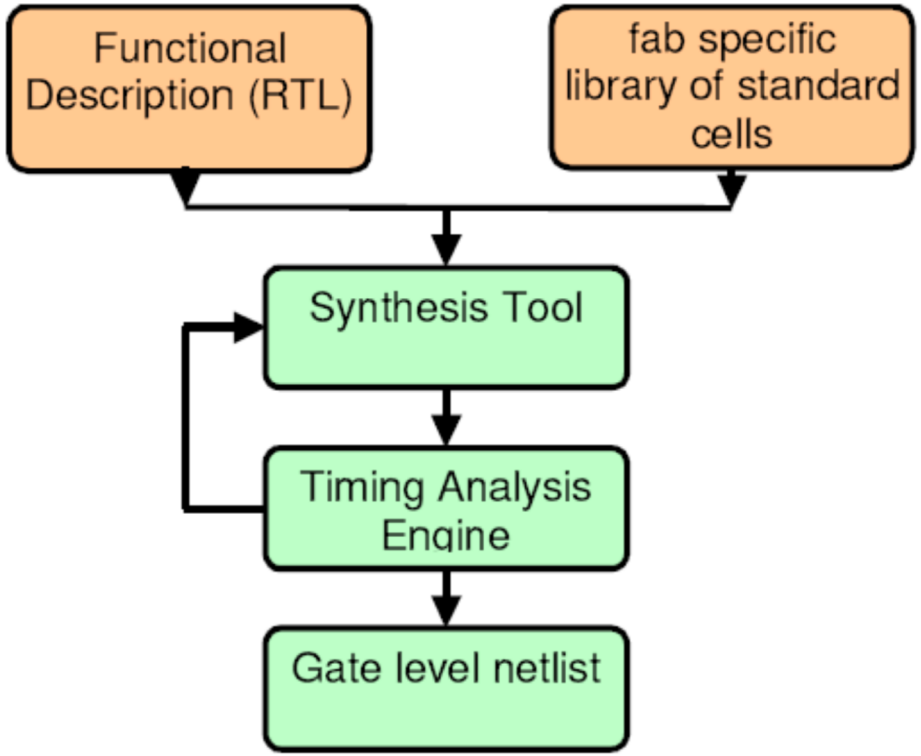
Register Transfer Level

- Separate sections of combinational logic by timing statements
- Define behavior of each part on clock boundary (edge)



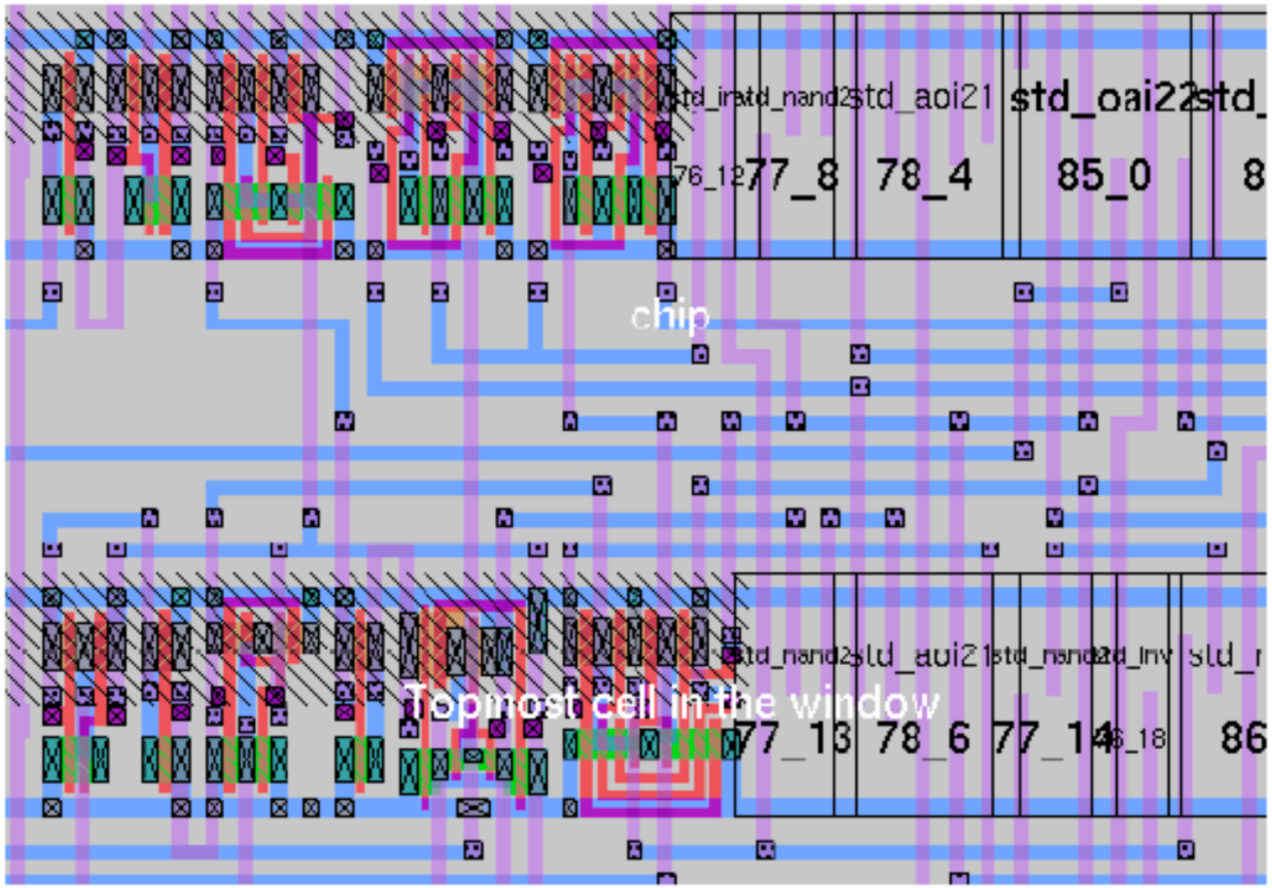
Logic Synthesis

- Synthesize combinational functions into standard cells (gates) from fab-specific library
- Choose IP based on timing concern (rise/falling, delay)



Auto Place and Route (APR)

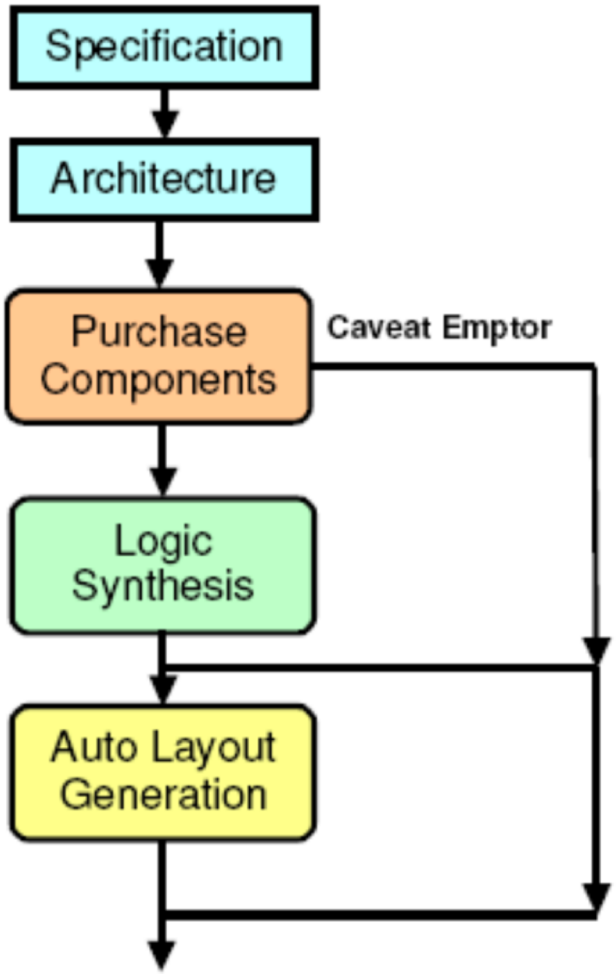
- Cell placement and routing generated by CAD
- Area/delay minimization and loading optimization



"Introduction to VLSI Design 2008", Prof. Subhasish Mitra @ Stanford

System-on-Chip Design Flow

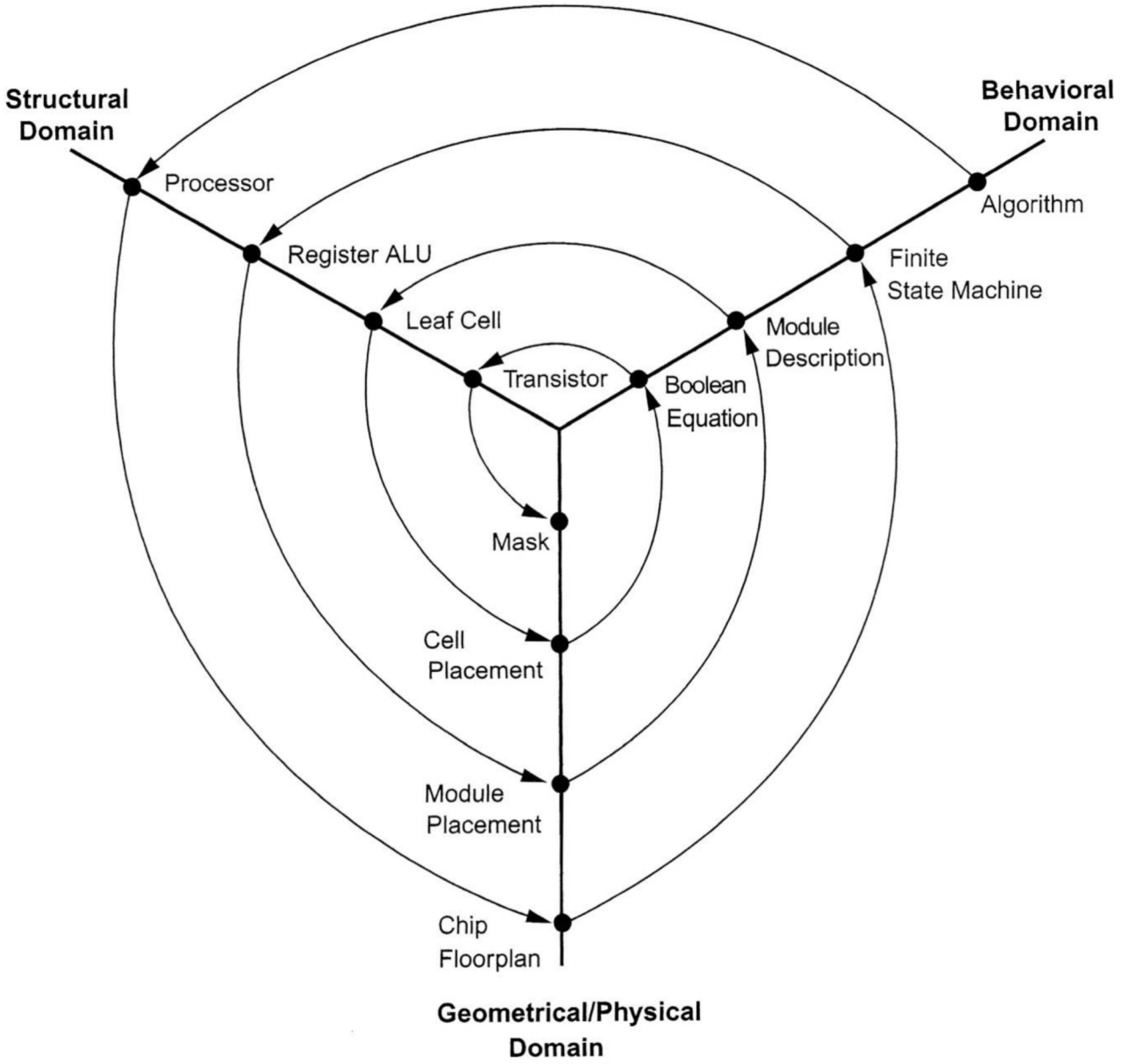
- Architecture plan,sub-block spec and interface definition
- Buy **Intellectual Property (IP)** from various vendors
- **Soft IP:**
  RTL of gate level description
  - Synthesize and APR for your process
  - **Ex** USB controller
- **Hard IP:**
  Polygon level description
  - Just customized place and route
  - **Ex** memory



Can CAD Tool Do Everything?

- Tools are smart and stupid
  - It can't think for you, be aware of the limitations
- Tools do output but do not judge
  - Need to know how to use tool correctly
  - Need to be able to identify the validity
- Tools always update
  - Need to follow the steps
  - Choose the right tool for you
- You can even create new tools
  - Startup opportunity

Y-Chart


Structural Domain
Behavioral Domain
Processor
Algorithm
Register ALU
Finite State Machine
Leaf Cell
Module Description
Transistor
Boolean Equation
Mask
Cell Placement
Module Placement
Chip Floorplan
Geometrical/Physical Domain

Structured Design

- Hierarchy: Divide and Conquer
  - Recursively divide system into modules
- Regularity
  - Reuse modules wherever possible
- Modularity: well-defined Interfaces
  - Allow modules to be treated as black boxes
- Locality
  - Physical and temporal