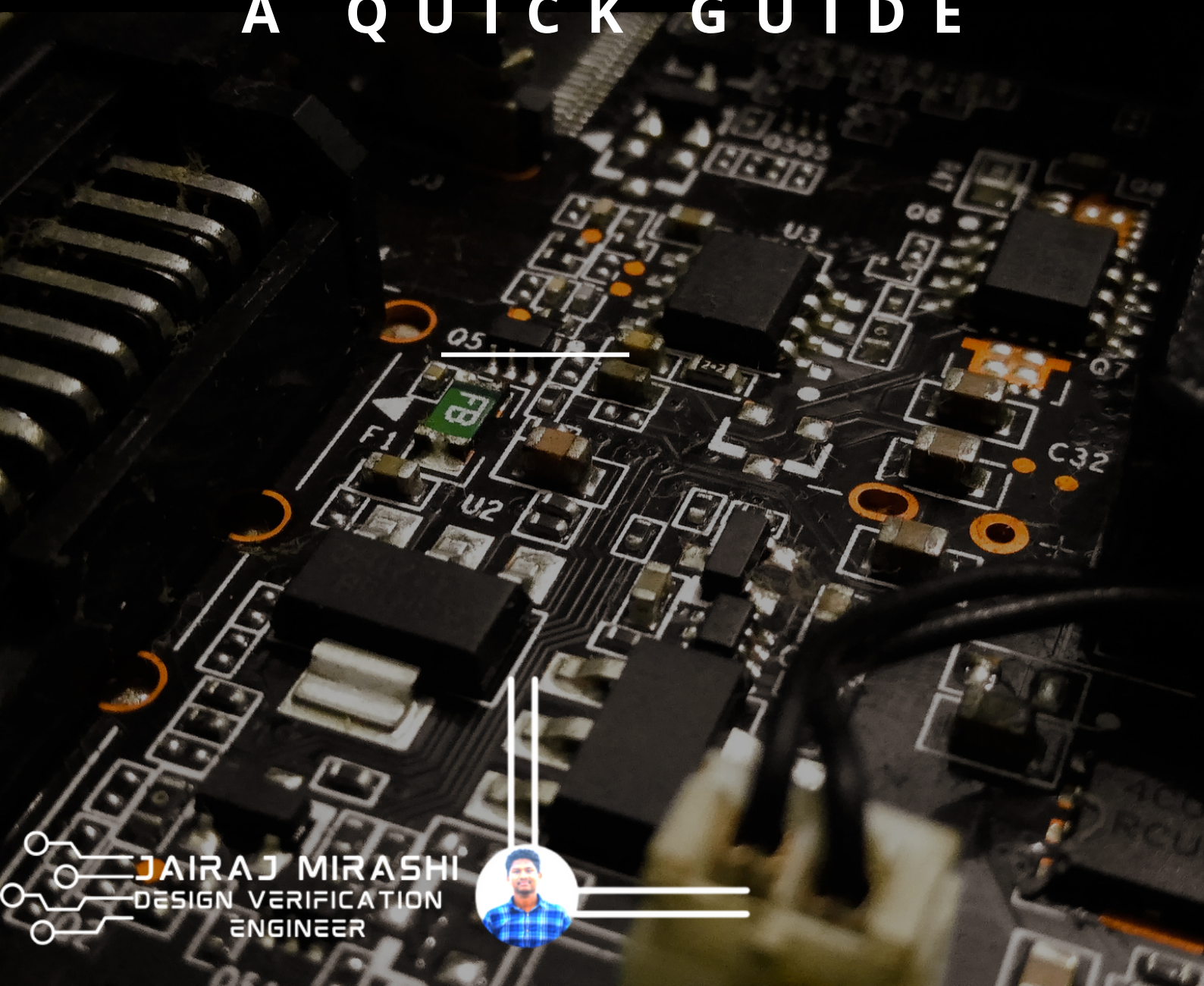
ASIC DESIGN FLOW

A QUICK GUIDE

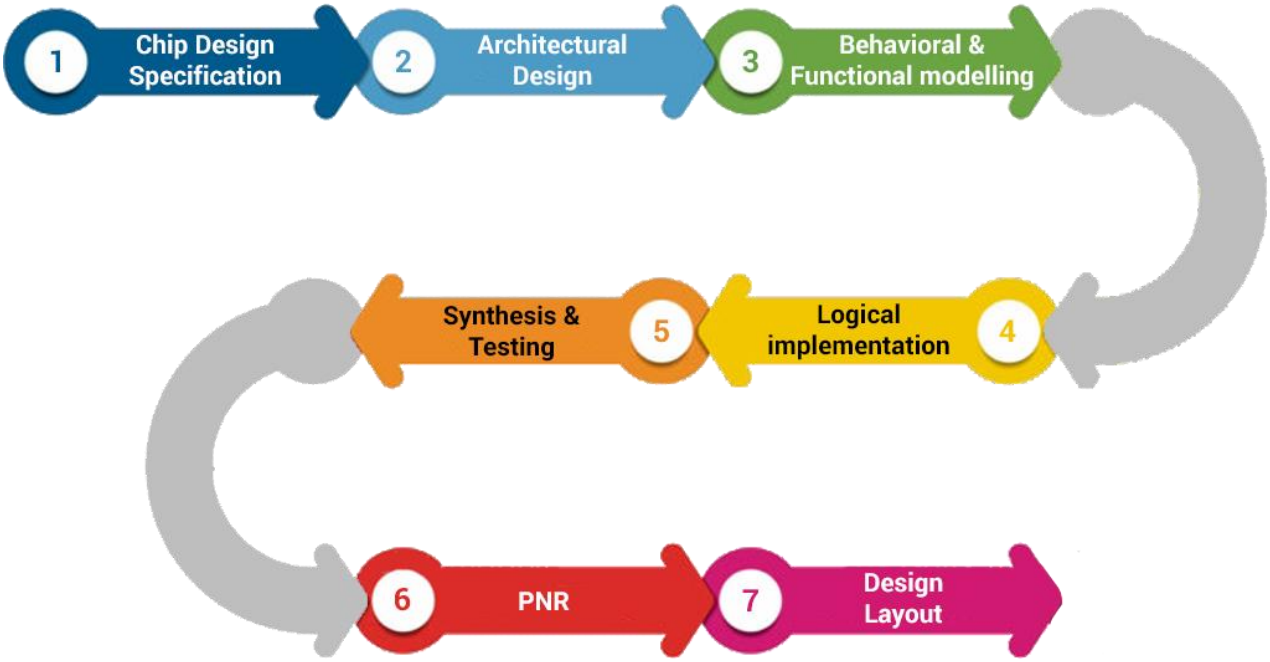


JAIRAJ MIRASHI
DESIGN VERIFICATION ENGINEER

ASIC Design Flow in VLSI Engineering Services – A Quick Guide

Today, ASIC design flow is a very mature process in silicon turnkey design. The ASIC design flow and its various steps in VLSI engineering that we describe below are based on best practices and proven methodologies in ASIC chip designs. This PDF attempts to explain different steps in the ASIC design flow, starting from ASIC design concept and moving from specifications to benefits.

Why to adopt the ASIC design flow?

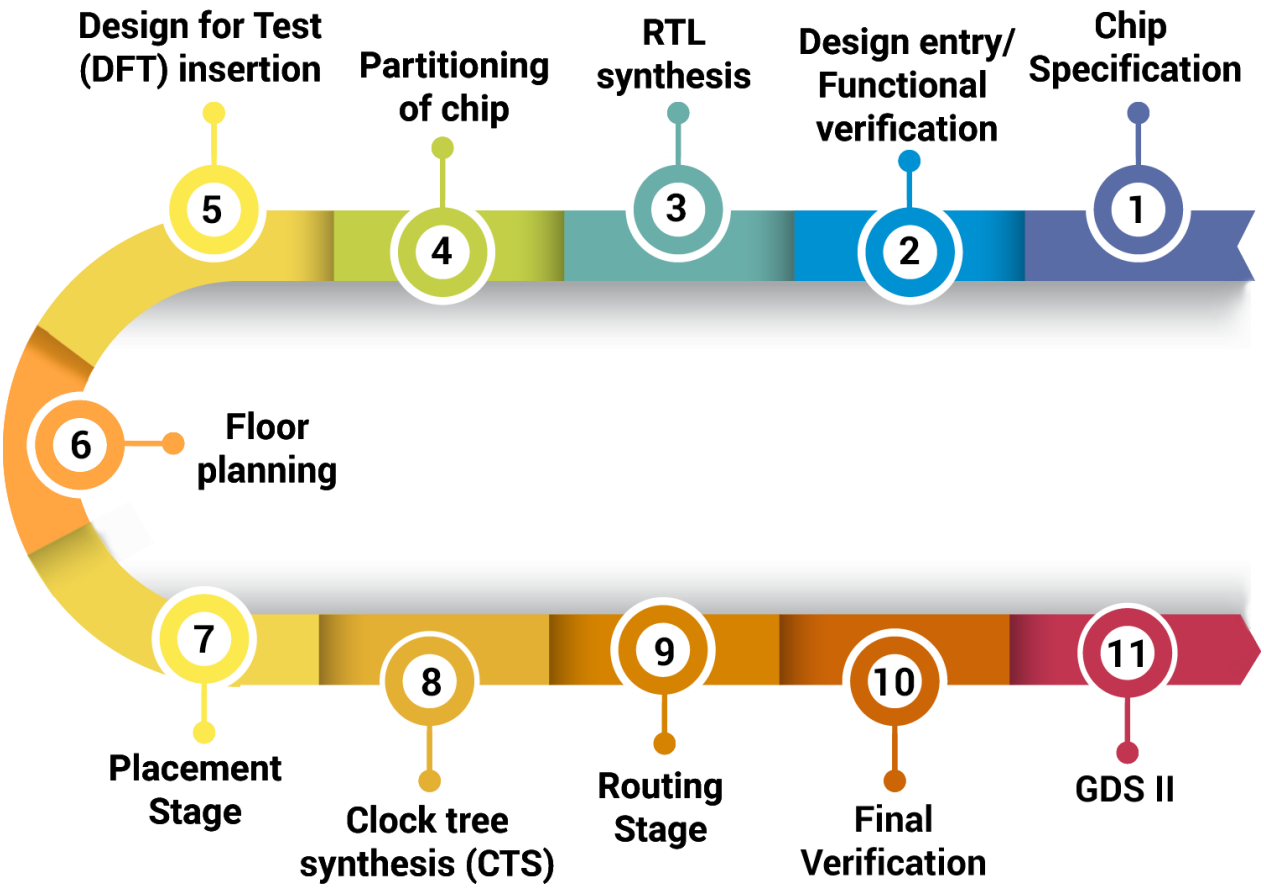


To ensure successful ASIC design, engineers must follow a proven ASIC design flow which is based on a good understanding of ASIC specifications, requirements, low power design and performance, with a focus on meeting the goal of right time to market. Every stage of ASIC design cycle has EDA tools that can help to implement ASIC design with ease.

How does the ASIC design cycle work?

In order to fulfill futuristic demands of chip design, changes are required in design tools, methodologies, and software/hardware capabilities. For those changes, ASIC design flow adopted by engineers for efficient structured ASIC chip architecture and focus on its design functionalities

ASIC design flow is a mature and silicon-proven IC design process which includes various steps like design conceptualization, chip optimization, logical/physical implementation, and design validation and verification. Let's have an overview of each of the steps involved in the process.



Step 1. Chip Specification

This is the stage at which the engineer defines features, microarchitecture, functionalities (hardware/software interface), specifications (Time, Area, Power, Speed) with design guidelines of ASIC. Two different teams are involved at this juncture:

- Design team: Generates RTL code.
- Verification team: Generates testbench (SV TB LINK, UVM TB LINK).

Step 2. Design Entry / Functional Verification

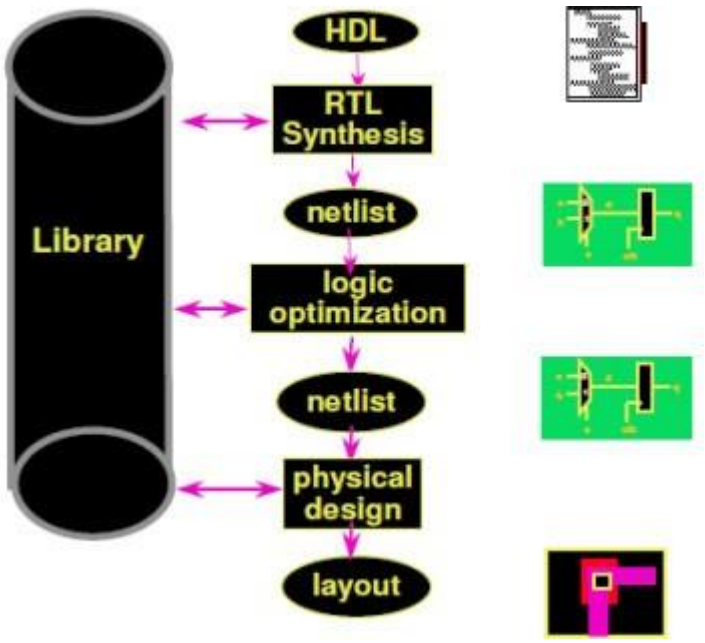
Functional verification confirms the functionality and logical behavior of the circuit by simulation on a design entry level. This is the stage where the design team and verification team come into the cycle where they generate RTL code using test-benches. This is known as ***behavioral simulation***.

In this simulation, once the RTL code (RTL code is a set of code that checks whether the RTL implementation meets the design verification) is done in HDL, a lot of code coverage metrics proposed for HDL. Engineers aim to verify correctness of the code with the help of test vectors and trying to achieve it by 95% coverage test. This code coverage includes statement coverage, expression coverage, branch coverage, and toggle coverage.

There are two types of simulation tools:

- **Functional simulation tools**: After the testbench and design code, functional simulation verifies logical behavior and its implementation based on design entry.
- **Timing simulation tools**: Verifies that circuit design meets the timing requirements and confirms the design is free of circuit signal delays.

Step 3. RTL block synthesis / RTL Function



Once the RTL code and testbench are generated, the RTL team works on RTL description – they translate the RTL code into a gate-level netlist using a logical synthesis tool that meets required timing constraints. Thereafter, a synthesized database of the ASIC design is created in the system. When timing constraints are met with the logic synthesis, the design proceeds to the design for testability (DFT) techniques.

Step 4. Chip Partitioning

This is the stage wherein the engineer follows the ASIC design layout requirement and specification to create its structure using EDA tools and proven methodologies. This design structure is going to be verified with the help of HLL programming languages like C++ or System C.

After understanding the design specifications, the engineers partition the entire ASIC into multiple functional blocks (hierarchical modules), while keeping in mind ASIC's best performance, technical feasibility, and resource allocation in terms of area, power, cost and time. Once all the functional blocks are implemented in the architectural document, the engineers need to brainstorm ASIC design partitioning by reusing IPs from previous projects and procuring them from other parties.

Step 5. Design for Test (DFT) Insertion

With the ongoing trend of lower technology nodes, there is an increase in system-on-chip variations like size, threshold voltage and wire resistance. Due to these factors, new models and techniques are introduced to high-quality testing.

ASIC design is complex enough at different stages of the design cycle. Telling the customers that the chips have fault when you are already at the production stage is embarrassing and disruptive. It's a situation that no engineering team wants to be in. In order to overcome this situation, design for test is introduced with a list of techniques:

- **Scan path insertion**: A methodology of linking all registers elements into one long shift register (scan path). This can help to check small parts of design instead of the whole design in one go.
- **Memory BIST (built-in Self-Test)**: In the lower technology node, chip memory requires lower area and fast access time. MBIST is a device which is used to check RAMs. It is a comprehensive solution to memory testing errors and self-repair proficiencies.
- **ATPG (automatic test pattern generation)**: ATPG is a method of creating test vectors / sequential input patterns to check the design for faults generated within various elements of a circuit.

Step 6. Floor Planning (blueprint your chip)

After, DFT, the physical implementation process is to be followed. In physical design, the first step in RTL-to-GDSII design is floorplanning. It is the process of placing blocks in the chip. It includes: block placement, design portioning, pin placement, and power optimization.

Floorplan determines the size of the chip, places the gates and connects them with wires. While connecting, engineers take care of wire length, and functionality which will ensure signals will not interfere with nearby elements. In the end, simulate the final floor plan with post-layout verification process.

A good floorplanning exercise should come across and take care of the below points; otherwise, the life of IC and its cost will blow out:

- Minimize the total chip area
- Make routing phase easy (routable)
- Improve signal delays

Step 7. Placement

Placement is the process of placing standard cells in row. A poor placement requires larger area and also degrades performance. Various factors, like the timing requirement, the net lengths and hence the connections of cells, power dissipation should be taken care. It removes timing violation.

Step 8. Clock tree synthesis

Clock tree synthesis is a process of building the clock tree and meeting the defined timing, area and power requirements. It helps in providing the clock connection to the clock pin of a sequential element in the required time and area, with low power consumption.

In order to avoid high power consumption, increase in delays and a huge number of transitions, certain structures can be used for optimizing CTS structure such as Mesh Structure, H-Tree Structure, X-Tree Structure, Fishbone Structure and Hybrid structure.

With the help of these structures, each flop in the clock tree gets the clock connection. During the optimization, tools insert the buffer to build the CTS structure. Different clock structures will build the clock tree with a minimum buffer insertion and lower power consumption of chips.

Step 9. Routing

1. **Global Routing**: Calculates estimated values for each net by the delays of fan-out of wire. Global routing is mainly divided into line routingand maze routing.
2. **Detailed Routing**: In detailed routing, the actual delays of wire is calculated by various optimization methods like timing optimization, clock tree synthesis, etc.

As we are moving towards a lower technology node, engineers face complex design challenges with the need for implanting millions of gates in a small area. In order to make this ASIC design routable, placement density range needs to be followed for better QoR. Placement density analysis is an important parameter to get better outcomes with less number of iterations.

Step 10. Final Verification (Physical Verification and Timing)

After routing, ASIC design layout undergoes three steps of physical verification, known as signoff checks. This stage helps to check whether the layout working the way it was designed to. The following checks are followed to avoid any errors just before the tapeout:

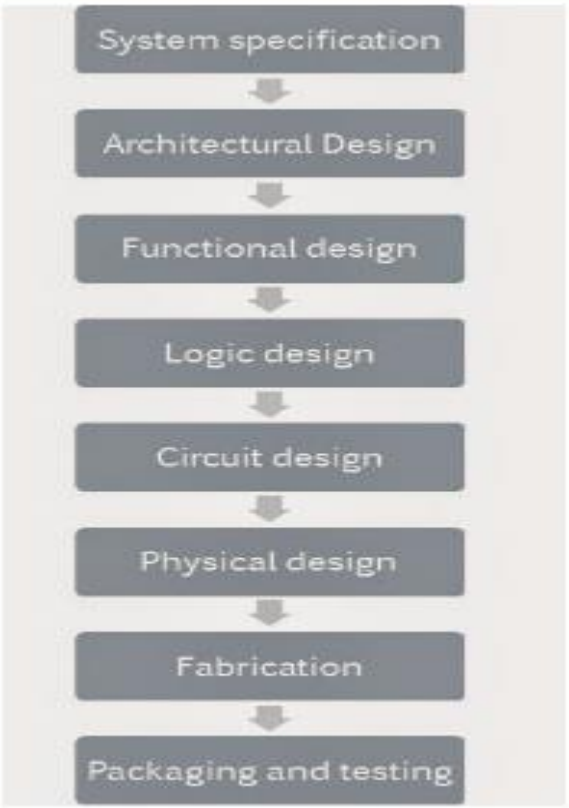
1. Layout versus schematic(LVS) is a process of checking that the geometry/layout matches the schematic/netlist.
2. Design rule checks(DRC) is the process of checking that the geometry in the GDS file follows the rules given by the foundry.
3. Logical equivalence checks(LVC) is the process of equivalence check between pre and post design layout.

Step 11. GDS II – Graphical Data Stream Information Interchange

In the last stage of the tapeout, the engineer performs wafer processing, packaging, testing, verification and delivery to the physical IC. GDSII is the file produced and used by the semiconductor foundries to fabricate the silicon and handled to client.

VLSI Design Flow

Flow chart



VLSI design Flow

System Specification

- The first step is to lay down the specifications of the system. System specification is a high level representation of the system.
- The factors to be considered in this process include: **performance, functionality, and physical dimensions (size of the die (chip))**. The fabrication technology and design techniques are also considered.
- The specification of a system is a compromise between market requirements, technology and economical viability.
- The end results are specifications for the size, speed, power, and functionality of the VLSI system.

Architectural Design

- The basic architecture of the system is designed in this step. This includes, such decisions as RISC or CISC, number of ALUs, Floating Point units, number and structure of pipelines, and size of caches among others.
- The outcome of architectural design is a Micro-Architectural Specification (MAS). Architects can accurately predict the performance, power and die size of the design based on such a description.

Behavioural or Functional Design

- In this step, main functional units of the system are identified. The area, power, and other parameters of each unit are estimated.
- The behavioural aspects of the system are considered without implementation specific information. The key idea is to specify behaviour, in terms of input, output and timing of each unit, without specifying its internal structure.
- The outcome of functional design is usually a timing diagram or other relationships between units. This information leads to improvement of the overall design process and reduction of the complexity of subsequent phases

Logic Design

- In this step the control flow, word widths, register allocation, arithmetic operations, and logic operations of the design that represent the functional design are derived and tested.
- This description is called Register Transfer Level (RTL) description. RTL is expressed in a Hardware Description Language (HDL), such as VHDL or Verilog. This description can be used in simulation and verification.

Circuit Design

- The purpose of circuit design is to develop a circuit representation based on the logic design.
- The Boolean expressions are converted into a circuit representation by taking into consideration the speed and power requirements of the original design.
- *Circuit Simulation* is used to verify the correctness and timing of each component.
- The circuit design is usually expressed in a detailed circuit diagram. This diagram shows the circuit elements (cells, macros, gates, transistors) and interconnection between these elements. This representation is also called a *netlist.*

Physical Design

- In this step the circuit representation (or netlist) is converted into a geometric representation. As stated earlier, this geometric representation of a circuit is called a *layout*.
- In many cases, physical design can be completely or partially automated and layout can be generated directly from netlist by *Layout Synthesis* tools. Layout synthesis tools, works fast, do have an area and performance penalty, which limit their use to some designs.

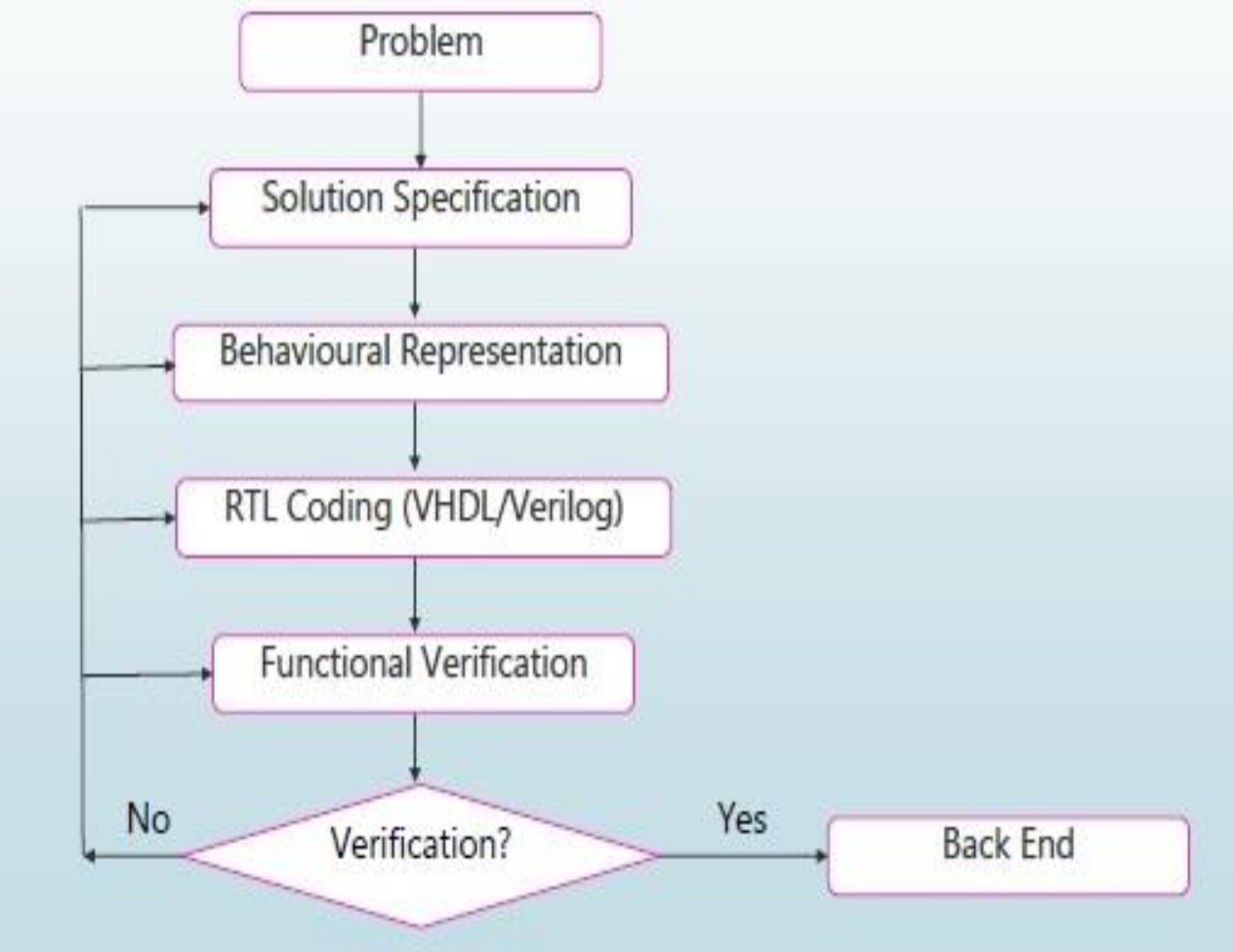
Fabrication

- After layout and verification, the design is ready for fabrication. Since layout data is typically sent to fabrication on a tape, the event of release of data is called *Tape Out.*
- Layout data is converted into photo-lithographic masks, one for each layer. Silicon crystals are grown and sliced to produce wafers. Extremely small dimensions of VLSI devices require that the wafers be polished to near perfection. The fabrication process consists of several steps involving deposition, and diffusion of various materials on the wafer. During each step one mask is used. Several dozen masks may be used to complete the fabrication process.

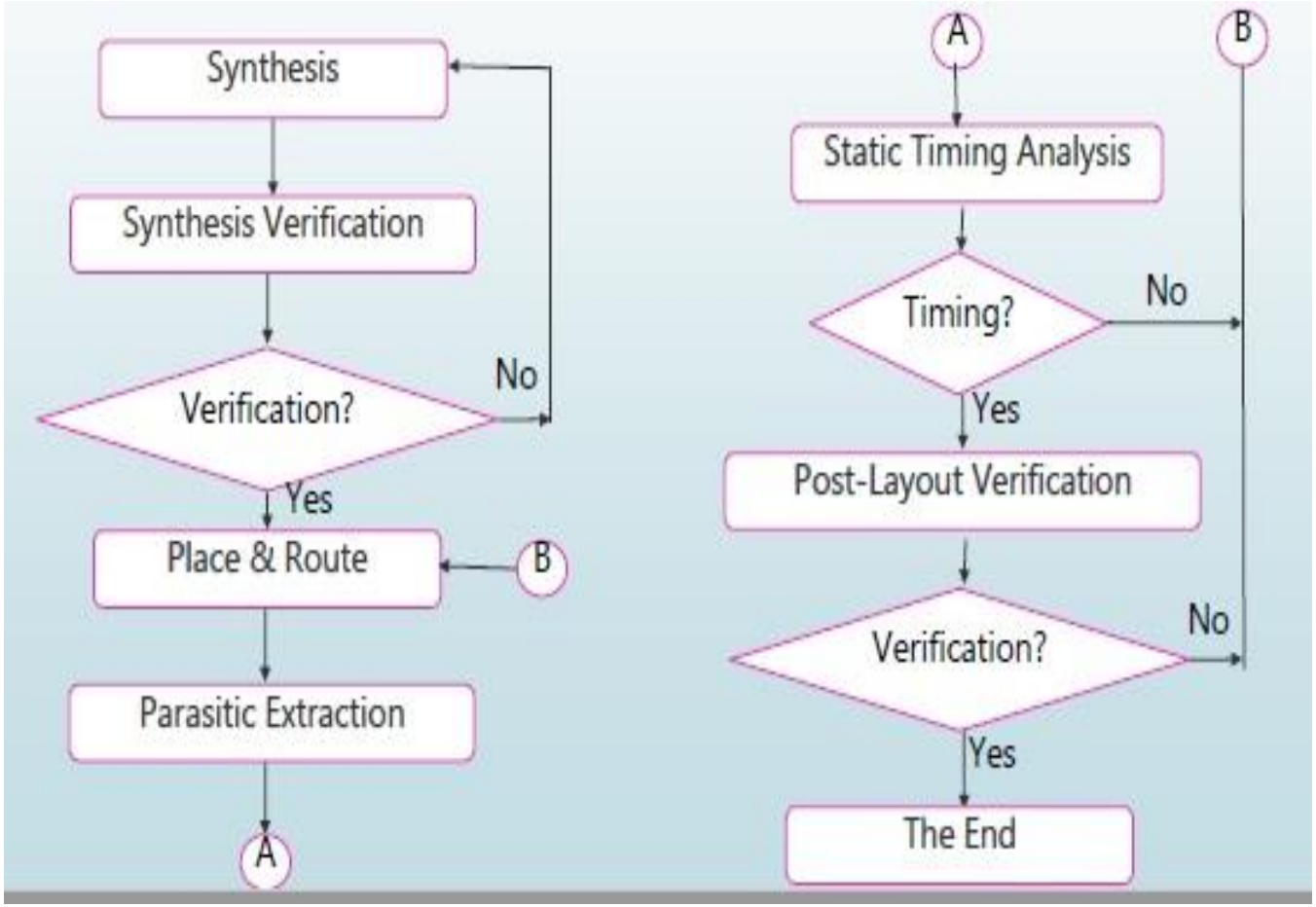
Packaging, Testing and Debugging

- Finally, the wafer is fabricated and diced into individual chips in a fabrication facility.
- Each chip is then packaged and tested to ensure that it meets all the design specifications and that it functions properly.
- Chips used in Printed Circuit Boards (PCBs) are packaged in Dual In-line Package (DIP), Pin Grid Array (PGA), Ball Grid Array (BGA), and Quad Flat Package (QFP).

Front end design



Back end design flow



Synthesis

- Synthesis is responsible for converting the RTL description into a structural gate level based netlist. This netlist instantiates every element that compose the circuit and its connections.
- Synthesis = Translation + Optimization + Mapping

Synthesis Verification

- The first step is to verify a set of reports, which have information about timing, area, fan-out
- These reports must be interpreted to check if there are violations
- In case of violations one can try to fix them by running optimization algorithms.
- If cannot fix the violations, one must go back to RTL coding.
- The final verification before proceeding to Place & Route is to run Formality, which is a logical verification tool.
- It takes the final net list generated and checks the logical equivalence with the RTL description.

Place & Route

- Partitioning is the task of dividing a circuit in such a way so that the area of each sub system is well within specified system range and number of interconnects between sub circuits is also minimized.
- Floorplanning is the step to determine the shape of each subcircuit module and pin location at their boundary and find of the approximate location of each module in rectangular chip.
- Placement is the problem of determination of best position of each module .

Routing

- Routing is the method of interconnection of different circuit components, with the aim to minimize the chip area and also reduce the total wire length.
- Global routing: Develop a routing plan so that each net is assigned a particular region.
- Channel routing: connect all the nets with minimum possible area.

Parasitic Extraction

- Parasitic extraction has the objective to create an accurate RC model of the circuit so that future simulations and timing, power and IR Drop analyses can emulate the real circuit response.
- Only with this information, all the analyses and simulations can report results close to the real functioning of the circuit. This way this stage needs to precede all signoff analyses.

Static Timing Analysis

- STA is a method to obtain accurate timing information without the need to simulate the circuit.
- It allows detecting setup and hold times violations, as well as skew and slow paths that limit the operation frequency.

Post Layout Verification

- The huge number of transistors in a circuit can make the voltage level drop below a defined margin that ensures that the circuit works properly.
- DRC (Design Rules Checking) checks if the foundry geometric and connectivity rules are met.
- LVS (Layout Versus Schematic) checks if the physical circuit corresponds to the original circuit schematic.