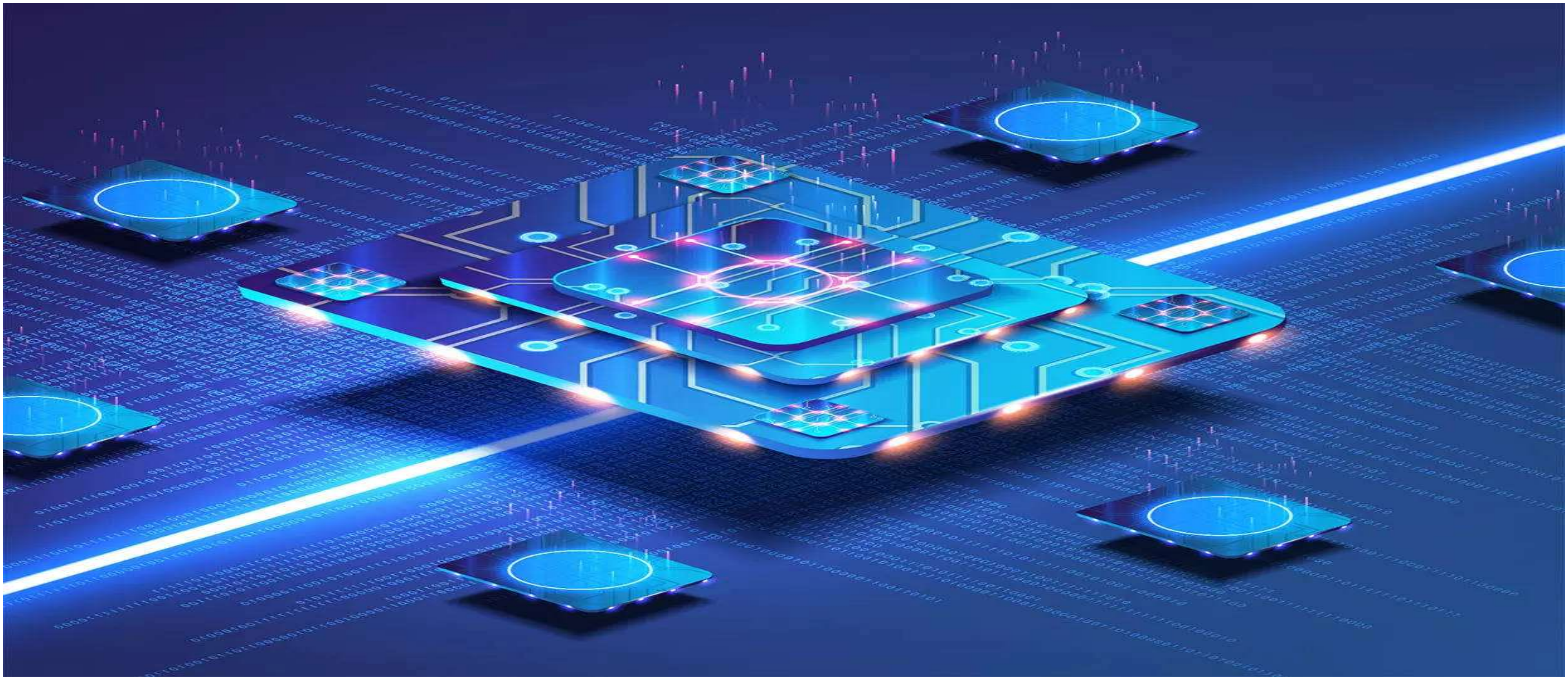


CMOS VLSI and Aspects of ASIC Design

PRESENTED BY:

Dr. Shasanka Sekhar Rout

SYLLABUS

Session 1
Lecture: Metal Oxide Semiconductors (MOS)

- Overview of CMOS Module
- Introduction of MOS devices
- n-MOS, p-MOS and CMOS
- MOS operation

Session 2
Lecture: CMOS Gates and Characteristics

- CMOS Logic Circuits
- CMOS Inverter and I-V Characters

Session 3
Lecture: Design of Complex Functions

- Design of Complex Functions using CMOS

Session 4
Lecture: MOS Cells

- Structure of MOS cells
- Threshold Voltage and DC transfer characteristics
- Introduction to IC Design Cycle and layout

Session 5
Lecture: CMOS functioning and parameters

- CMOS as a switch, estimation of parasitic values
- rise and fall times
- power dissipation
  - ❖ Dynamic and Static
  - ❖ Low power architecture

Session 6
Lecture: Technology Scaling

- ✓ Device sizing, rationed and non-rationed logic
- ✓ Concepts of Interconnect
- ✓ Transistor Scaling and interconnect Scaling

Session 7
Lecture: CMOS Logic Circuits

- ✓ Design of logic circuit, fabrication steps
- ✓ CMOS Combinational Logic circuits
- ✓ Design of NAND and NOR Gates
- ✓ Transmission Gates

Session 8
Lecture: IC Fabrication using CMOS Technology

- ✓ CMOS Technology, wafer formation, photolithography.
- ✓ Well and channel formation, Silicon dioxide and gate oxide.
- ✓ Overview of CMOS Fabrication Steps

Session 9
Lecture: Memory using CMOS

- ✓ Design of Complex circuits
- ✓ SRAM, DRAM & current Mirror
- ✓ Memory cell and their read write operation.

Session 10
Lecture: Advanced Trends in Backend VLSI

- ✓ Sequential CMOS logic
- ✓ Circuit Design of Latches and Flip-Flops
- ✓ Introduction to FinFET technology

Session 11
Lecture: CMOS Design Issues

- ✓ Design issue like Delays, Cross talk , Velocity saturation,
- ✓ Body effect, latch up, electro migration, hot carrier and Channel Length modulation

Session 12
Lecture: Aspects of ASIC Design

- ❖ Introduction of ASIC Design,
- ❖ An overview of Backend VLSI Design Flow – Libraries, Floorplanning, Placement, Routing, Verification, Testing.
- ❖ Flow Diagram, Specifications and Schematic cell Design.
- ❖ Spice simulation. Analysis of analog and digital circuits, Circuits elements, AC and DC analysis.

Session 13
Lecture: Characteristics of Different Technologies

- ❖ Transfer Characteristics, Transient responses, Noise analysis of current and voltage based on different technologies.

Session 14
Lecture: Back-end Processes

- ❖ Fabrication methods of circuit elements, Layout design of different cells.
- ❖ Library cell designing, NAND, NOR, NOT, X-OR etc.
- ❖ Circuit Extraction, Electrical rule check, LVS, Post layout Simulation and parasitic extraction.

Session 15 & 16
Lecture: Back-end Designing and Analysis

- ❖ Design format, timing analysis, back annotation and post layout simulation of silicon circuit
- ❖ Study of design Issues Antenna effect, Electro migration effect body effect, Inductive and capacitive cross talk, Drain punch through, etc.
- ❖ Placement of cells, placing of I/O blocks, Initialization of floor planning, routing and creating design data base
- ❖ DFT Guidelines, Test Pattern and BIST
- ❖ ASIC design implementation

Session 12

Lecture: Aspects of ASIC Design

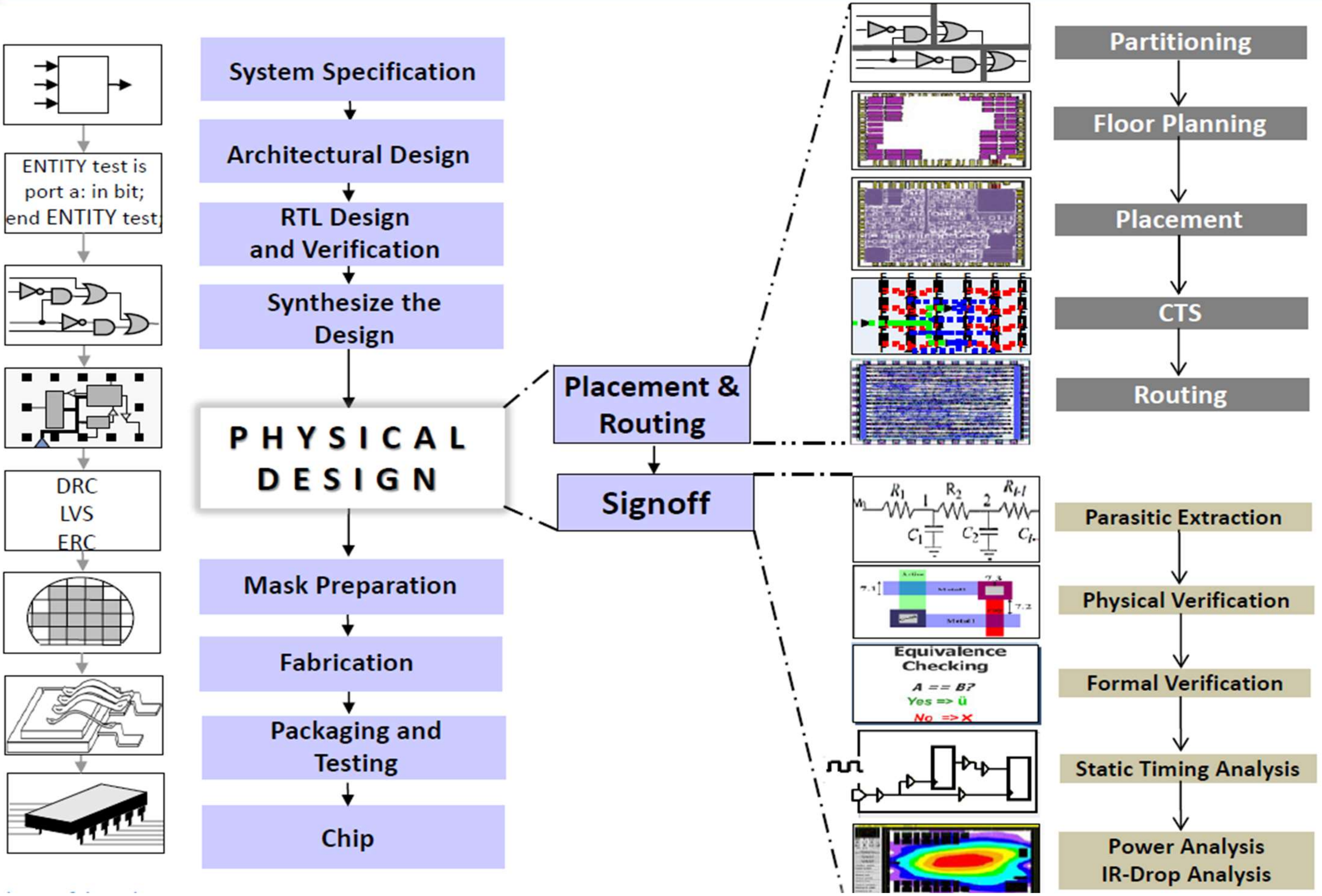
❖Introduction of ASIC Design

❖An overview of Backend VLSI Design Flow – Libraries, Floorplanning, Placement, Routing, Verification, Testing

❖Flow Diagram, Specifications and Schematic cell Design.

❖Spice simulation: Analysis of analog and digital circuits, Circuits elements, AC and DC analysis.

Aspects of ASIC Design



System Specification

Architectural Design

RTL Design and Verification

Synthesize the Design

PHYSICAL DESIGN

Placement & Routing

Signoff

Mask Preparation

Fabrication

Packaging and Testing

Chip

Partitioning

Floor Planning

Placement

CTS

Routing

Parasitic Extraction

Physical Verification

Formal Verification

Static Timing Analysis

Power Analysis IR-Drop Analysis

An Overview of Backend VLSI Design Flow

- ASIC (Application-Specific Integrated Circuit) design involves creating a custom integrated circuit for a specific application or purpose.
- The key aspects of ASIC design are:

1. Specification and Requirement Analysis:

- ➢ Defining the functionality, performance, power consumption, area, and other requirements.
- ➢ Understanding the application and environment where the ASIC will be used.

2. Architecture Design:

- ➢ High-level design, including defining the data path, control units, memory elements, and interfaces.

3. RTL (Register-Transfer Level) Design:

- ➢ Writing the hardware description language (HDL) code (typically Verilog or VHDL) that defines the behavior of the ASIC at the register-transfer level.
- ➢ Implementing algorithms, state machines, and other digital logic components.

An Overview of Backend VLSI Design Flow

4. Verification:

- Functional Verification: Ensuring the RTL design meets the specifications using simulation and formal methods.
- Timing Verification: Ensuring the design meets the required timing constraints using static timing analysis (STA).

5. Synthesize The Design:

- Logic Synthesis: Converting the RTL code into a gate-level netlist that can be implemented in silicon.

6. Physical Design:

- Floorplanning: Determining the placement of major functional blocks and macros within the chip.
- Placement: Placing the standard cells according to the floorplan.
- Clock Tree Synthesis (CTS): Ensuring the clock signal is distributed with minimal skew and meets timing constraints.
- Routing: Connecting all the placed cells and macros with metal layers to form the complete netlist.
- Design Rule Check (DRC) and Layout Versus Schematic (LVS) Check: Ensuring the physical layout adheres to manufacturing constraints and matches the schematic.

An Overview of Backend VLSI Design Flow

7. Power Analysis and Optimization:

- Estimating power consumption and optimizing the design for power efficiency.

8. Mask Preparation:

- Mask preparation, also known as mask making or photomask preparation is used to transfer the circuit patterns onto the silicon wafer during the photolithography process.

9. Fabrication:

- Sending the verified and validated design to a semiconductor foundry for fabrication.
- Working with the foundry to address any issues during the manufacturing process.

10. Packaging and Testing:

- Encapsulating the fabricated die in a suitable package for protection and connectivity.
- Performing final testing to ensure the packaged ASIC functions correctly.

11. Chip (Deployment and Maintenance):

- Integrating the ASIC into the final product (chip) and verifying its performance in the real-world application.

An Overview of Backend VLSI Design Flow

Libraries:

- Logical (Liberty Timing File) (.lib/.db)
- Library exchange format (.lef): shape of the cell, metal layers, pins shape, antenna ratio.

- LIB contains
  - Cell Type and Functionality
  - Delay Models (WLD/ NLDM)
  - Pin/ Cell Timings and design rules
  - PVT Conditions
  - Power Details (Leakage and Dynamic)

- LEF contains
  - Cell Name, Shape, Size, Orientation & Class
  - Port/Pin Name, Direction and Layout Geometries
  - Obstruction/ Blockages
  - Antenna Diff. Area

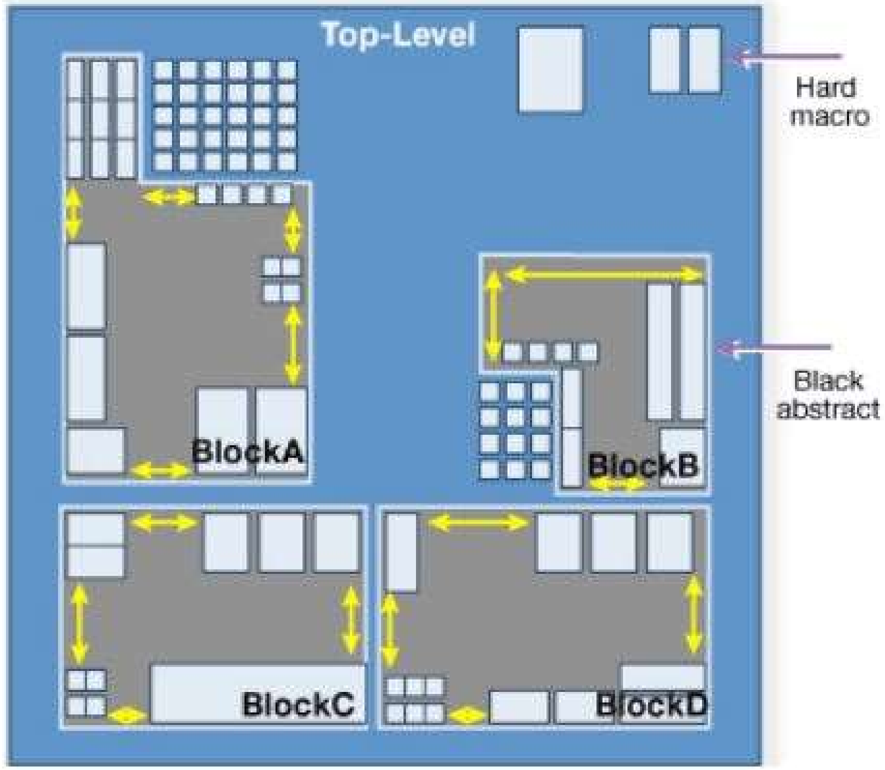
Physical Design (PD) Flow

Physical Design :

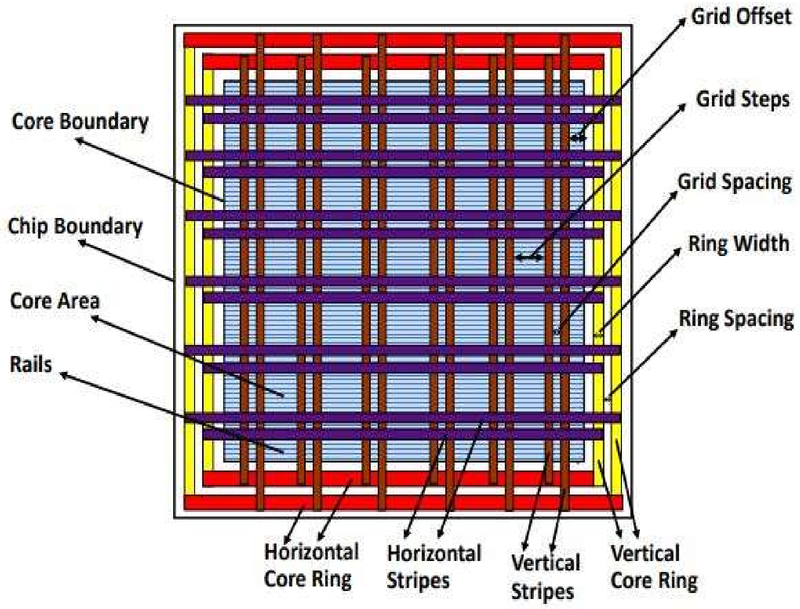
Structural Representation to Physical Implementation i.e., Netlist to GDSII (Geometrical data base standard for information interchange)

PnR Flow: Design planning/Import/Partioning => Floorplan & Powerplan => Placement => CTS => Routing => SignOff

Partitioning: The Hierarchical Partitioning is done prior to Floorplan



- Sub-block Configuration



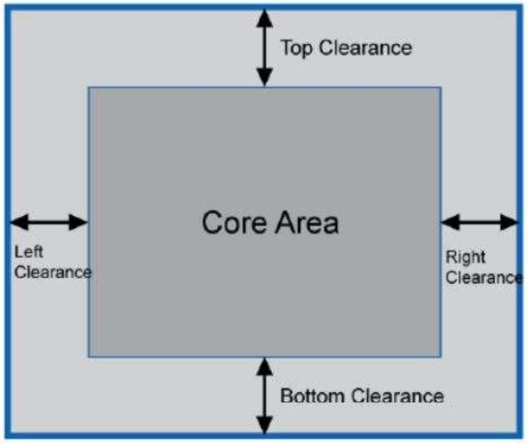
Physical Design (PD) Flow

Floorplan: A floorplan is the process of placing blocks/macros (memory cells) in the chip/core area, thereby determining the routing areas between them.

- Floorplan determines the size of die and creates wire tracks for placement of standard cells. It creates power ground (PG) connections. It also determines the I/O pin/pad placement information.
- Quality of your chip/design implementation depends on how good is your floorplan.
- A good floorplan can be make implementation process (place, cts, route & timing closure) cake walk.
- A bad floorplan can create all kind issues in the design (congestion, timing, noise, IR, routing issues).
- A bad floorplan will blow up the area, power, and affects reliability. Life of an IC and also it can increase the overall IC cost.

Objective of Floorplan:

- ✓ Minimize the die size
- ✓ Meet the timing requirements
- ✓ Maximize the routability
- ✓ Minimize the delays
- ✓ Reduce the wire length
- ✓ Reduce IR drop

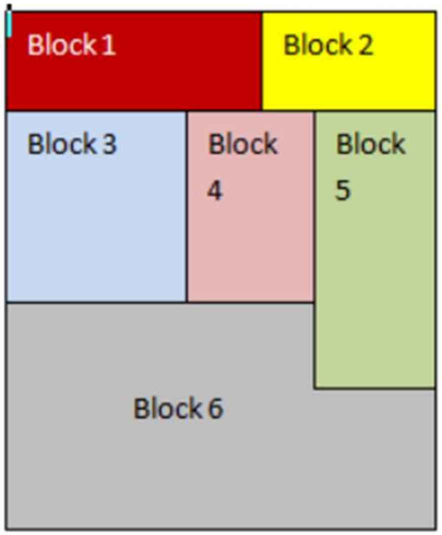


Physical Design (PD) Flow

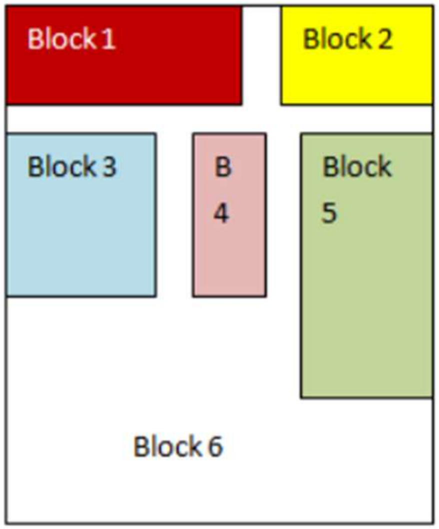
Types of Floorplan Techniques:

- **Abutted:-** When the chip is divided into blocks, in the abutted design there is no gap between the blocks. Channel less placement of blocks.
- **Non abutted:-** In this design there is a gap between blocks. The connection between the blocks is done through the routing nets. Channel based placement of blocks.
- **Mix of Both:** Partially abutted with channels.

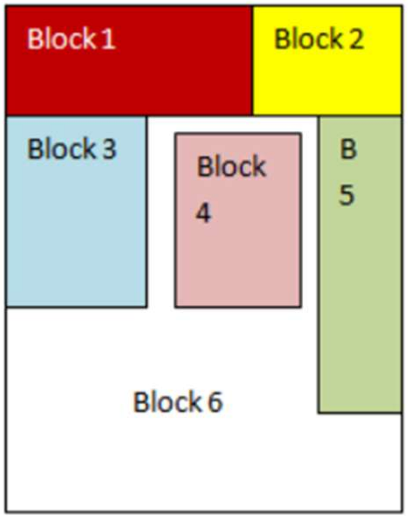
Tools: Cadence Innovus, Synopsys IC Compiler II.



Abutted



Non abutted



Mixed

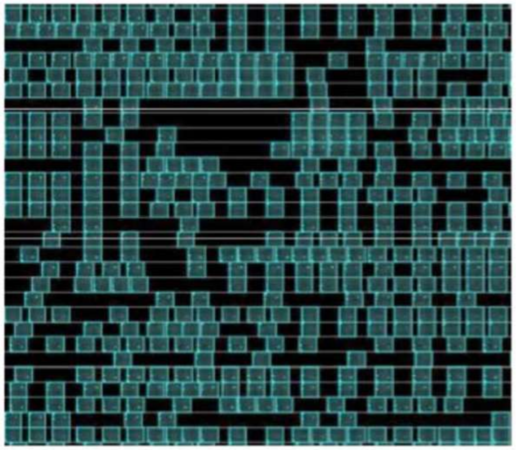
Physical Design (PD) Flow

Utilization: Area occupied by the std cells, macros & blockages.

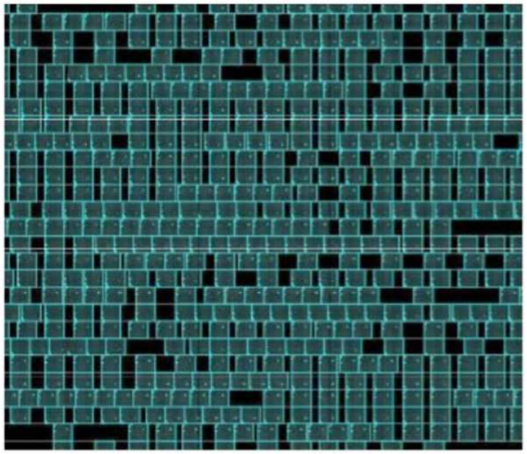
$$\text{Utilization} = \frac{\text{Total Standard Cell Area+Macro Area}}{\text{Total Core Area}} \times 100\,\%$$

$$\text{Aspect Ratio} = \frac{\text{Horizontal Routing Resource}}{\text{Vertical Routing Resource}}$$

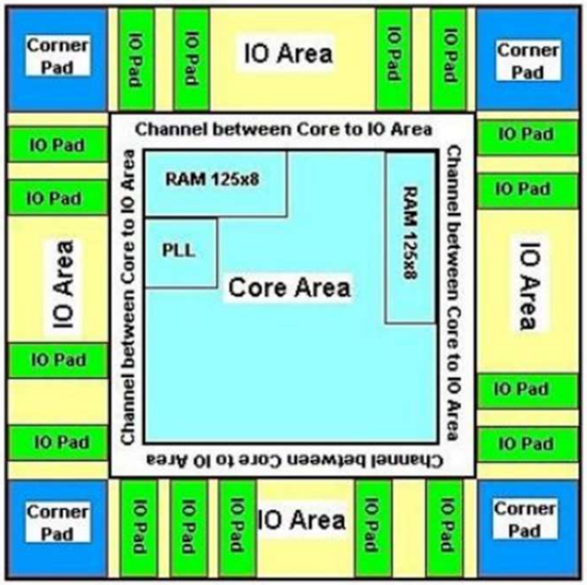
- ➢ U = 0.8, means 80% of area is available for placement of cells, whereas 20% is left for routing.
- ➢ Aspect Ratio: Height/Width: Decides the shape.
- ➢ Full chip Aspect Ratio can have a maximum value of 1.25.



Low standard-cell utilization



High standard-cell utilization

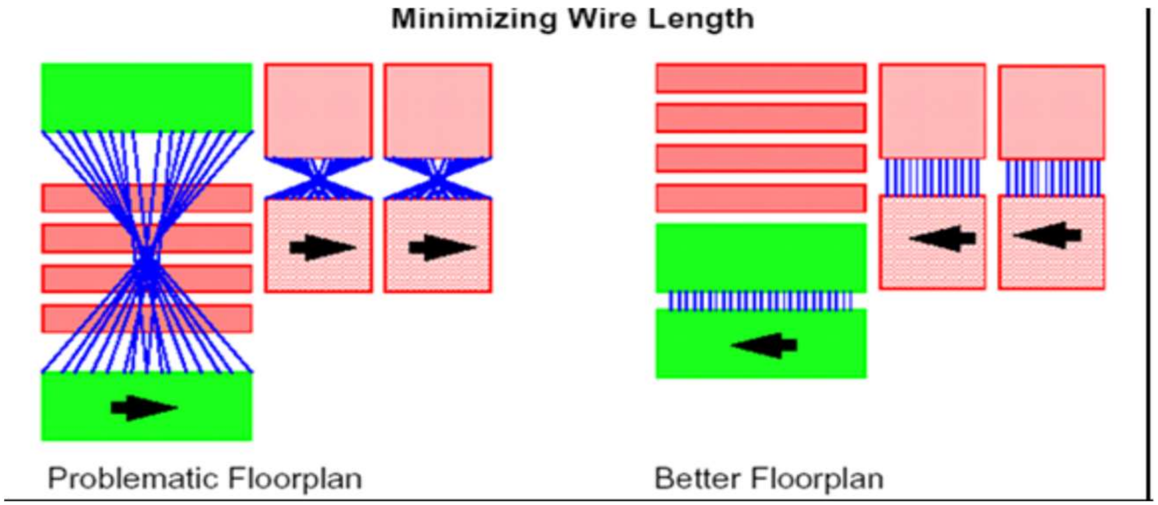


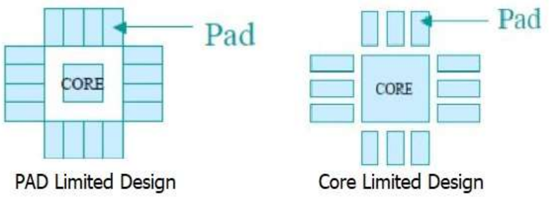
Physical Design (PD) Flow

Macro Placement:

- Fly-line Analysis (For Connectivity information)
- Macro keep-out (For Uniform Standard Cell Region)
- Channel Calculation (Critical for Congestion and Timing)
- Avoid odd shaped area for Standard Cells
- Funnel shaped Macro Placements are preferred
- Fix the Macro locations, so that tool wont alter during Optimization
- Spacing between Macro:

$$\frac{\text{Pitch of the Routing Layers x No.of pins to be Routed}}{\text{Available Routing Layers in the prefered direction}} + \text{Buffer Spacing}$$

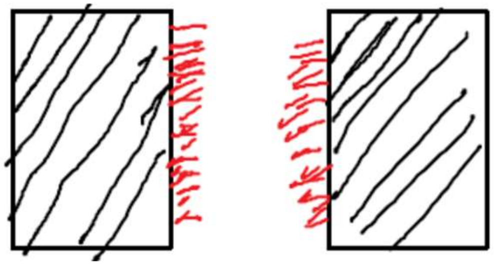




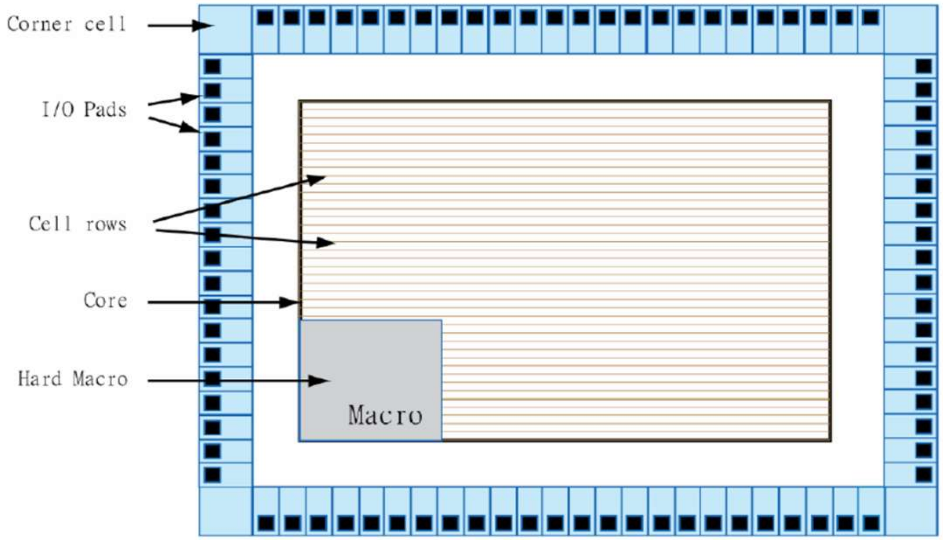
Physical Design (PD) Flow

Spacing between Macro/channel between macros:

Problem: If there are two macros facing each other each have 100 pins and pins are in M4 whose Pitch is 0.08, If it's a 9 metal layer design, then how much spacing need to be given.



Solution: Spacing in micros = (Number of pins * metal pitch) / (available routing/2) = ((100+100)*0.08) / (9/2) = (200*0.08) / 4 = 4



Physical Design (PD) Flow

Placement:

- It is the process of placing standard cells in the design. The tool determines the location of each standard cells in the die. The tool places the cells based on the algorithms which it uses internally.
- It also optimizes the design and determines the routability of the design.

Placement Objectives:

- Initial placement of standard cells.
- Optimization to minimize wire length and improve timing.
- Handle placement constraints like pin density and blockages.

Tools: Cadence Innovus, Synopsys IC Compiler II, Mentor Graphics Olympus-SoC.

Physical Design (PD) Flow

Goals of Placement:

✓ Timing, Area and Power optimizations

✓ Minimum congestion

✓ Minimal cell density, pin density and congestion hot-spots

✓ Minimal timing DRVs

Placement Output:

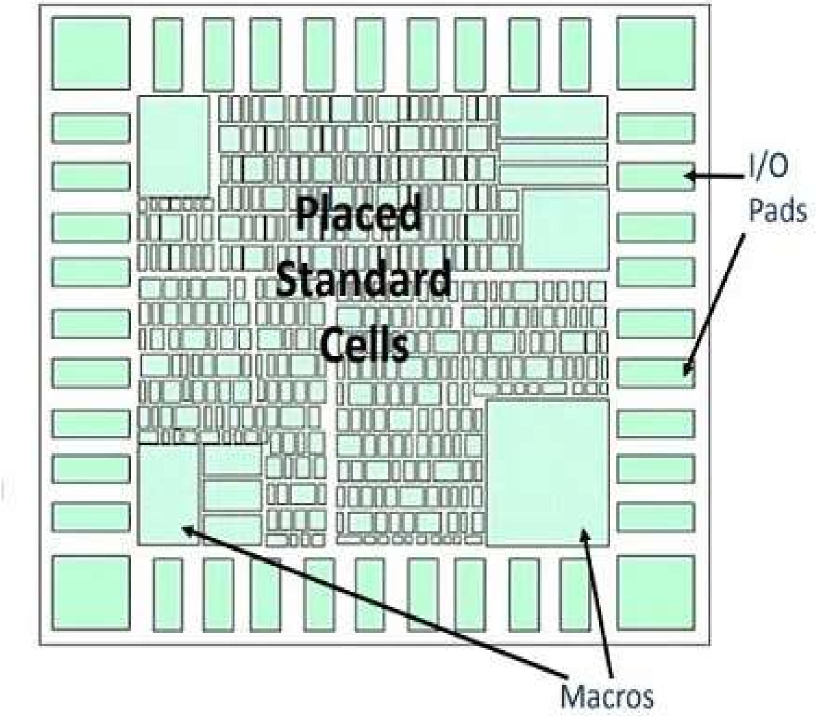
✓ Congestion report

✓ Timing report

✓ Design with all std cells placed in core area

✓ Logs

✓ Placement DEF file



Physical Design (PD) Flow

Clock Tree Synthesis (CTS):

- ✓ The process of distributing the clock and balancing the load.
- ✓ CTS QoR decides timing convergence and power.

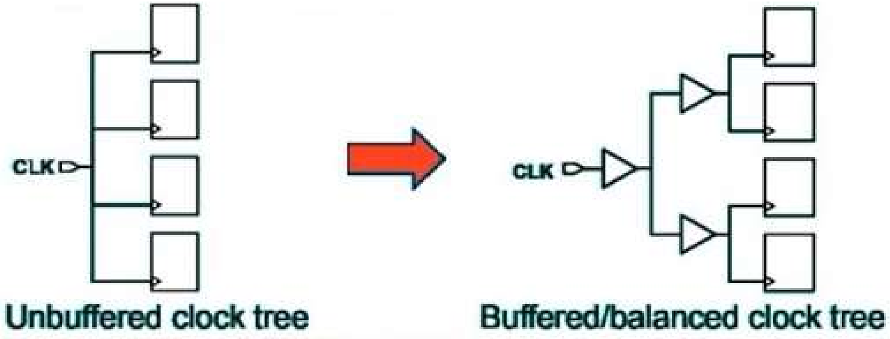
Goals:

- ➢ Minimize the skew and latency
- ➢ Sends clock to all sequential circuits
- ➢ Meets clock DRC (max trans, max cap, max fanout)

Skew: Time intervals in between lunch and capture time

Latency: The delay difference from the clock generation point to the clock endpoints.

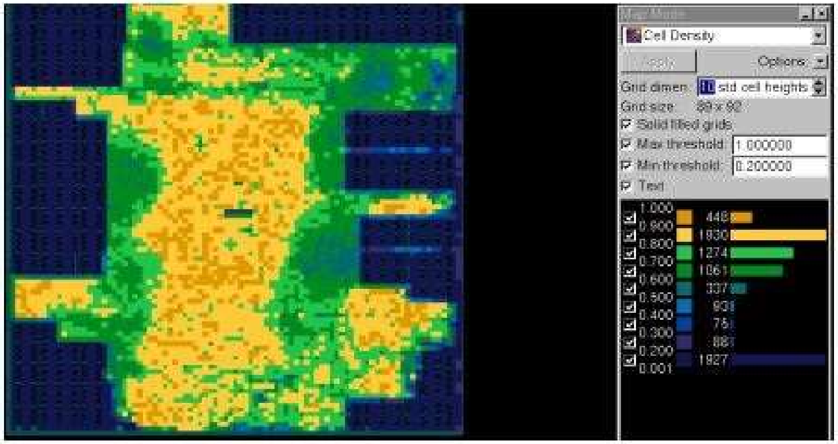
Tools: Cadence Innovus, Synopsys IC Compiler II, Mentor Graphics Olympus-SoC.



Physical Design (PD) Flow

Routing:

- ❖ Connect all placed components according to the netlist.
- ❖ Making physical connections between signal pins using metal layers are called routing.
- ❖ Here, exact paths for the interconnection of standard cells and macros and I/O pins are determined.
- ❖ There are minimal DRC violations while routing.
- ❖ The design is 100% routed with minimal LVS violations.
- ❖ There must be no or minimal congestion hot spots.



Cell Density

Grid dimen: 10 std cell heights

Grid size: 89 x 92

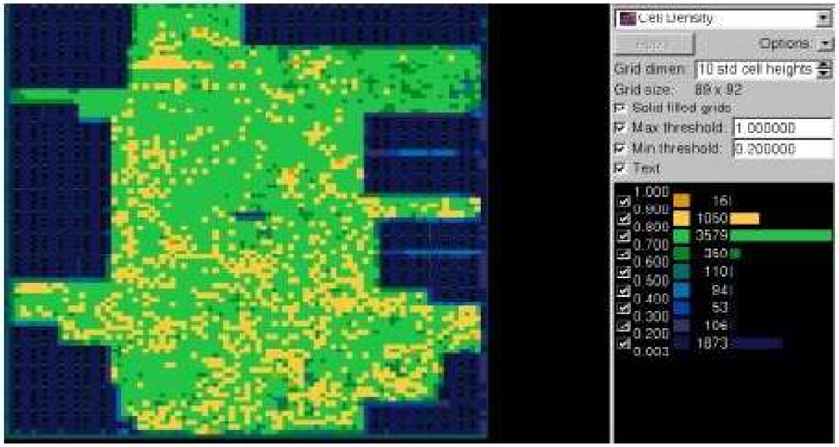
☑ Solid filled grids

☑ Max threshold: 1.000000

☑ Min threshold: 0.200000

☑ Text

Before Fixing



Cell Density

Grid dimen: 10 std cell heights

Grid size: 89 x 92

☑ Solid filled grids

☑ Max threshold: 1.000000

☑ Min threshold: 0.200000

☑ Text

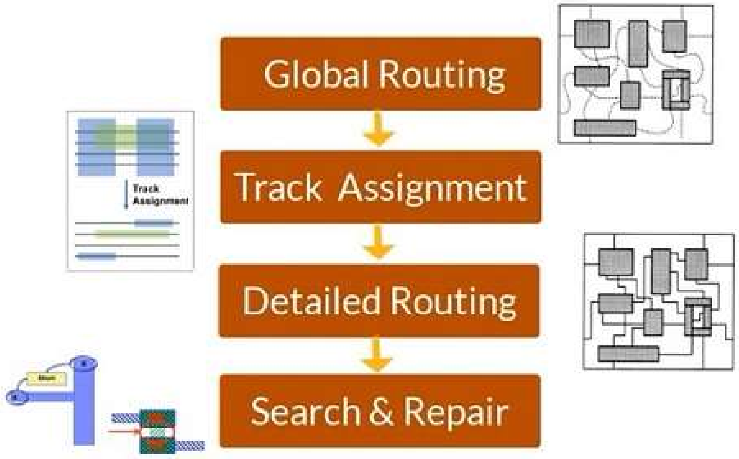
After Fixing

Physical Design (PD) Flow

Routing:

- **Global routing** to determine approximate paths. Identifying routable path for the nets driving/ driven pins in a shortest distance.
- **Track Assignment** takes the Global Routed Layout and assigns each nets to the specific Tracks and layer geometry.
- **Detailed routing** to finalize the exact wire paths. It follows up with the track routed net segments and performs the complete DRC aware and timing driven routing.
- **Search and Repair stage** is performed during detailed routing after the first iteration. Here, shorts and spacing violations are located and rerouting of affected areas to fix all possible violation is executed.

Tools: Cadence Innovus, Synopsys IC Compiler II, Mentor Graphics Olympus-SoC.



Global Routing

Track Assignment

Detailed Routing

Search & Repair

Spice Simulation

Spice Simulation: Analysis of analog and digital circuits, Circuits elements, AC and DC analysis

SPICE (Simulation Program with Integrated Circuit Emphasis):

- ➢ It is a powerful tool used for simulating and analyzing analog and mixed-signal circuits.

- ➢ It helps designers verify circuit performance, validate designs, and identify potential issues before physical prototyping.

- ➢ It is general purpose circuit simulation program for non linear (large-signal) DC, transient, and linear (small-signal) AC analysis.

  – example: Switching power supplies, RAM cells

- ➢ Circuit may contain:

  – Resistors, capacitors, inductors, voltage and current sources, transmission lines, switches, and the common semiconductor devices: diodes, BJT, MOSFETs etc.

Popular Versions of SPICE

- ➢ SPICE3: Unix - University of California Berkeley
- ➢ HSPICE: commercial - (originally from Meta soft) Synopsys
- ➢ PSPICE: Cadence Design Systems
- ➢ Eldo Classic: Mentor graphics
- ➢ LTspice: Analog Devices
- ➢ ngSpice: Open source
- ✓ SPICE1 was written in FORTRAN, SPICE3 in ‘C’.
- ✓ Circuit to be analyzed by SPICE is described through an input file, with set of lines. The input file can have any name.

Spice Input File:

- ✓ Title statement: Title line
- ✓ Comment statement: The asterisk is used
- ✓ Element statement: Circuit description, Element value,
- ✓ Control statement: Analysis Requests
- ✓ End statement: .END

Key Aspects of SPICE Simulation

1. Netlist Creation: It is a text file that describes the circuit components and their connections.

Example:

```
* Simple RC Circuit
V1 1 0 DC 5
R1 1 2 10k
C1 2 0 1uF
.END
```

2. Component Models: SPICE uses mathematical models to represent the behavior of electronic components.

3. Simulation Types:

- **DC Analysis**: Determines the steady-state operating point of the circuit.
- **AC Analysis**: Analyzes the frequency response of the circuit.
- **Transient Analysis**: Simulates the circuit's behavior over time.
- **Noise Analysis**: Evaluates the noise performance of the circuit.
- **Parametric Sweep**: Studies the effect of varying component values or other parameters.

Key Aspects of SPICE Simulation

4. Control Statements: Used to specify the type of analysis and set up simulation parameters.

Example:

```
.DC V1 0 5 0.1
.AC DEC 10 1 1MEG
.TRAN 1us 10ms
.OPTIONS TEMP=27
```

5. Output Commands: Direct the simulator to print or plot specific results.

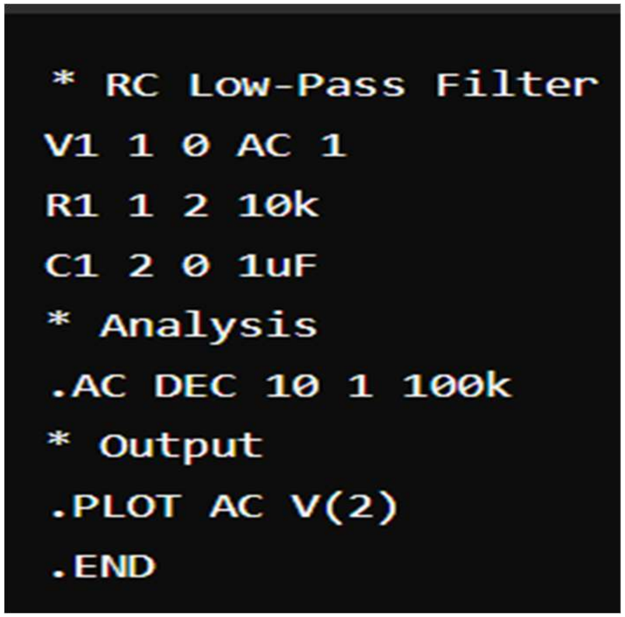
Example:

```
.PRINT DC V(2)
.PLOT TRAN V(2)
```

Example of SPICE Simulation

Let's consider a simple RC low-pass filter circuit.

Netlist:



```
* RC Low-Pass Filter
V1 1 0 AC 1
R1 1 2 10k
C1 2 0 1uF
* Analysis
.AC DEC 10 1 100k
* Output
.PLOT AC V(2)
.END
```

V1 1 0 AC 1 => An AC voltage source (1V amplitude) connected between node 1 and ground.

R1 1 2 10k => A 10k ohm resistor connected between nodes 1 and 2.

C1 2 0 1uF => A 1uF capacitor connected between node 2 and ground.

.AC DEC 10 1 100k => Performs an AC analysis with a decade frequency sweep from 1Hz to 100kHz.

.PLOT AC V(2) => Plots the AC voltage at node 2.

Design Levels for SPICE Simulation

- SPICE can be used at various levels of Design for simulation.

– Schematics level

- Start with a circuit (schematics)
- Describe it by creating/writing a ‘netlist’
- Attach appropriate device models to the schematics
- It becomes input (file) for spice simulation

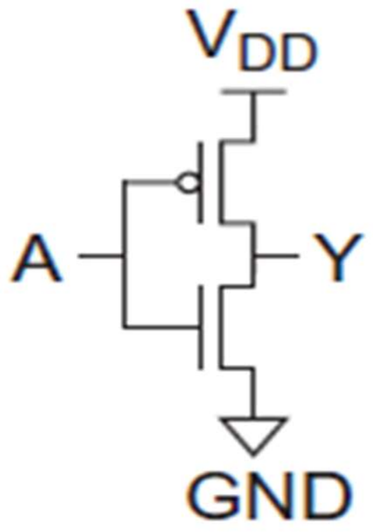
– Layout level

- Layout can be simulated for functionality using SPICE

Spice Code for Inverter

Steps:

1. First line is commented by default.
2. Add model files if any.
3. Information of circuit components and connection.
4. Information of connected sources.
5. Analysis to be performed/control statement.
6. Syntax to see o/p waveform
7. End the code by following syntax.

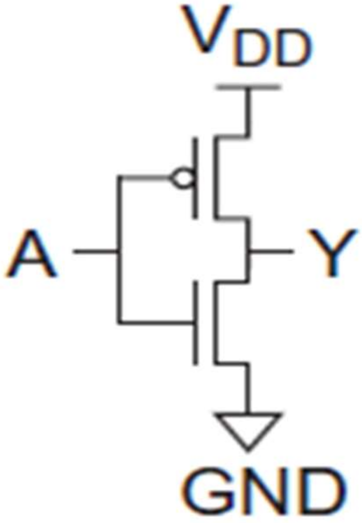


Inverter spice code:

```
.model n1 nmos level = 49 version = 3.3.0
.model p1 pmos level = 49 version = 3.3.0
MN 2 1 0 0 n1
MP 2 1 3 3 p1
VDD 3 0 5
VIN 1 0 Pulse (0 5 10n 0.1n 0.1n 30n 60n)
Tran 1n 70n
.control
Run
Plot v(1) v(2)
.endc
.end
```

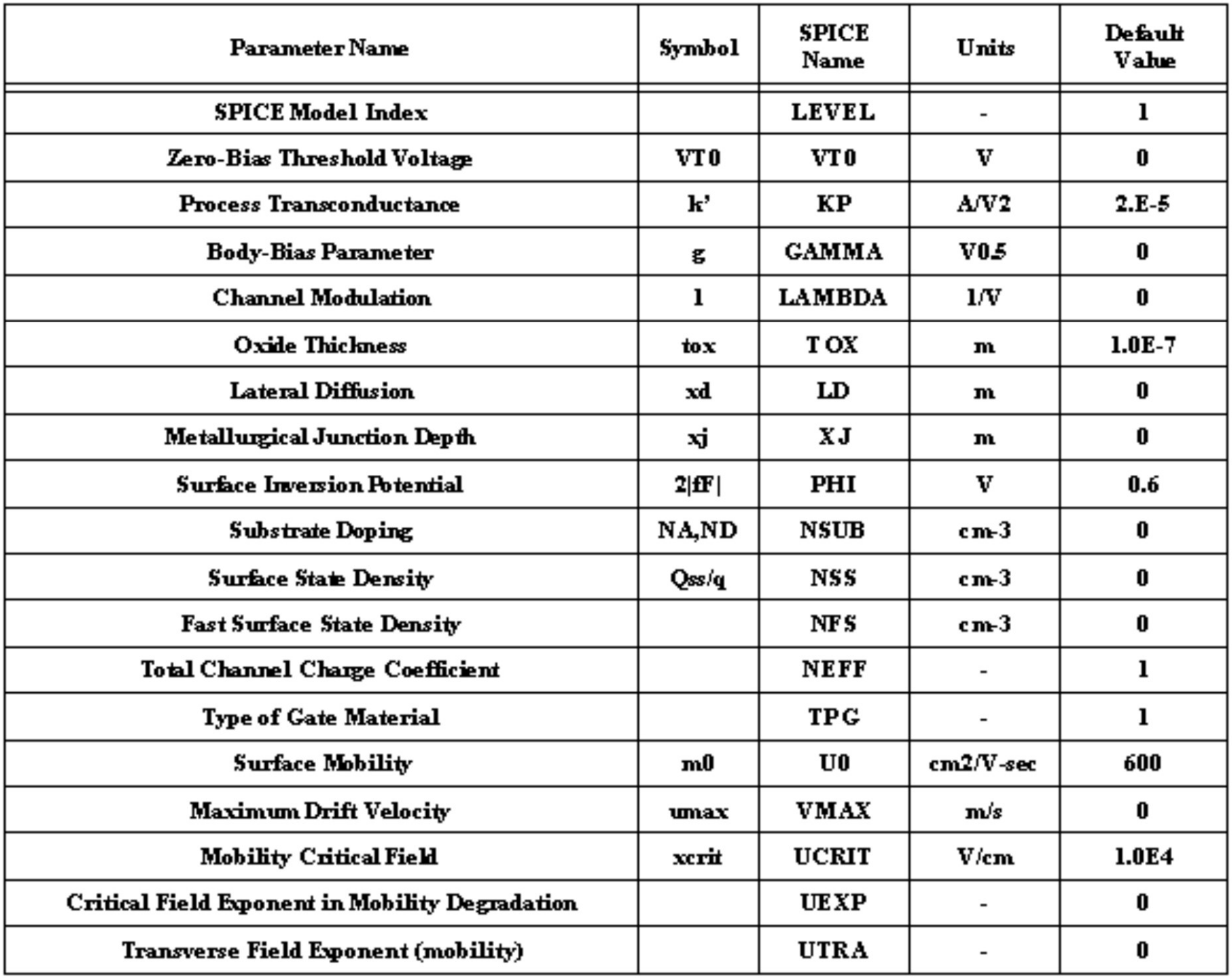
Spice Code for Inverter

Inverter ng spice code:



```
.model n1 nmos level = 49 version = 3.3.0
.model p1 pmos level = 49 version = 3.3.0
Vpower Vdd 0 1.8
Vgnd Vss 0 0
MP out in vdd vdd CMOSP W=3.6u L=0.18u
MN out in vss vss CMOSN W=1.0u L=0.18u
Vin in 0 Pulse (0 1.8 100p 50p 50p 200p 500p)
.tran 3p 600p
.dc Vin 0 1.8 0.1
.control
run
Plot in out
.endc
.end
```

Main MOS SPICE Parameters



Parameter Name	Symbol	SPICE Name	Units	Default Value
SPICE Model Index		LEVEL	-	1
Zero-Bias Threshold Voltage	VT0	VT0	V	0
Process Transconductance	k’	KP	A/V2	2.E-5
Body-Bias Parameter	g	GAMMA	V0.5	0
Channel Modulation	l	LAMBDA	1/V	0
Oxide Thickness	tox	TOX	m	1.0E-7
Lateral Diffusion	xd	LD	m	0
Metallurgical Junction Depth	xj	XJ	m	0
Surface Inversion Potential	2|fF|	PHI	V	0.6
Substrate Doping	NA,ND	NSUB	cm-3	0
Surface State Density	Qss/q	NSS	cm-3	0
Fast Surface State Density		NFS	cm-3	0
Total Channel Charge Coefficient		NEFF	-	1
Type of Gate Material		TPG	-	1
Surface Mobility	m0	U0	cm2/V-sec	600
Maximum Drift Velocity	umax	VMAX	m/s	0
Mobility Critical Field	xcrit	UCRIT	V/cm	1.0E4
Critical Field Exponent in Mobility Degradation		UEXP	-	0
Transverse Field Exponent (mobility)		UTRA	-	0

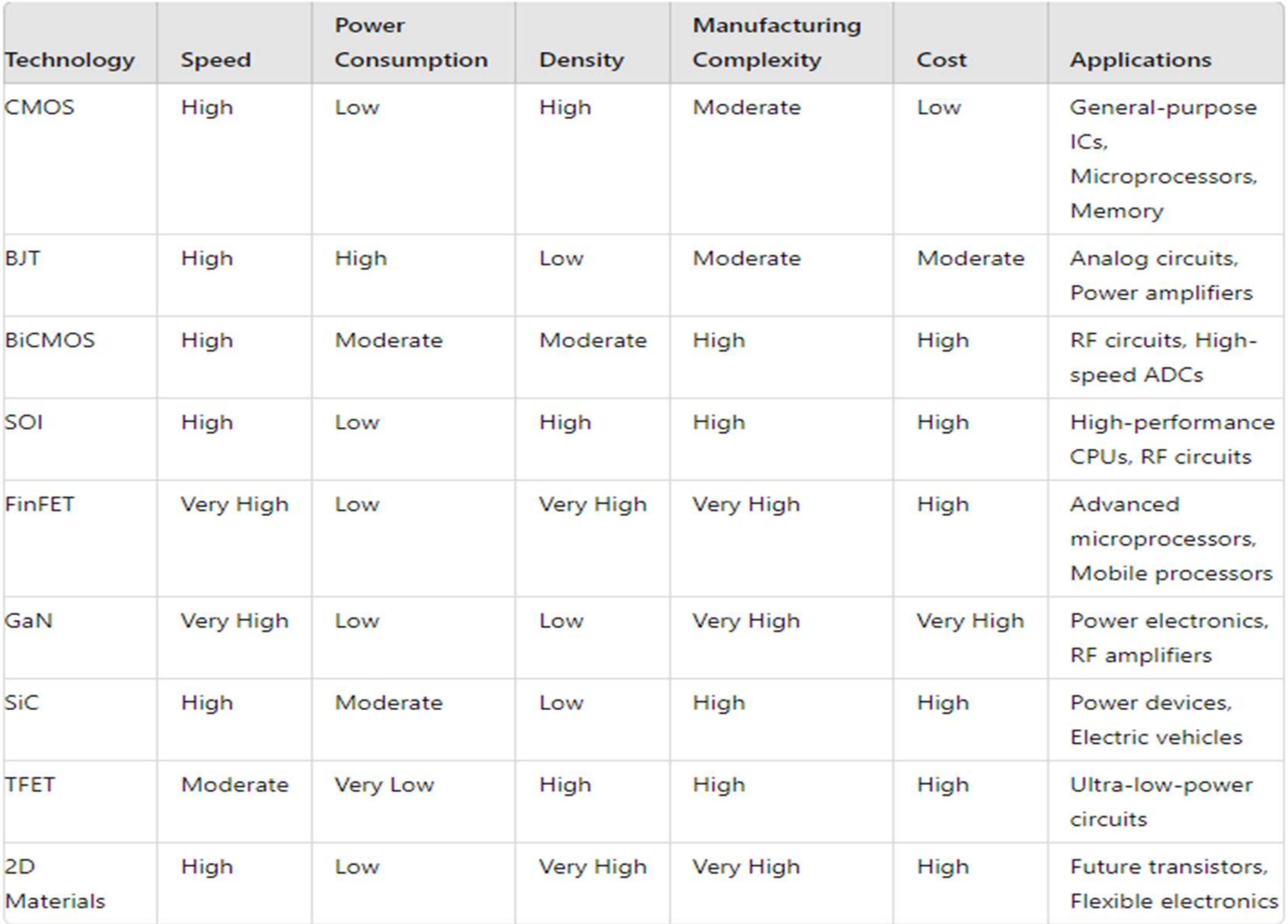
Session 13

Lecture: Characteristics of Different Technologies

- Transfer Characteristics, Transient responses, Noise analysis of current and voltage based on different technologies

Characteristics of Different Technologies

- In the semiconductor design, various technology nodes and types of transistors play a crucial role in determining the characteristics and performance of integrated circuits.

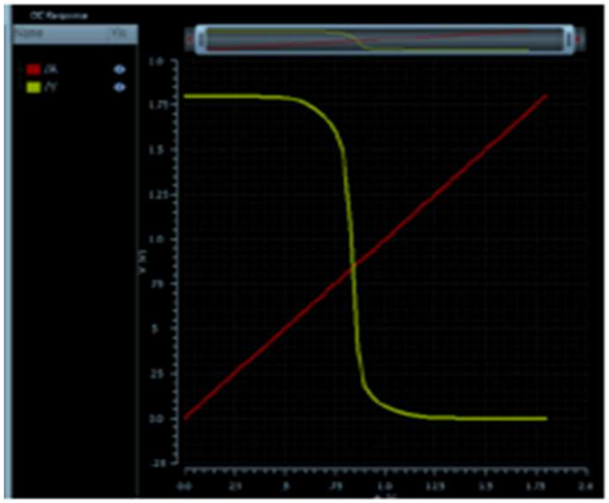


Technology	Speed	Power Consumption	Density	Manufacturing Complexity	Cost	Applications
CMOS	High	Low	High	Moderate	Low	General-purpose ICs, Microprocessors, Memory
BJT	High	High	Low	Moderate	Moderate	Analog circuits, Power amplifiers
BiCMOS	High	Moderate	Moderate	High	High	RF circuits, High-speed ADCs
SOI	High	Low	High	High	High	High-performance CPUs, RF circuits
FinFET	Very High	Low	Very High	Very High	High	Advanced microprocessors, Mobile processors
GaN	Very High	Low	Low	Very High	Very High	Power electronics, RF amplifiers
SiC	High	Moderate	Low	High	High	Power devices, Electric vehicles
TFET	Moderate	Very Low	High	High	High	Ultra-low-power circuits
2D Materials	High	Low	Very High	Very High	High	Future transistors, Flexible electronics

Characteristics of Different Technologies

Transfer Characteristics:

- Transfer characteristics describe the relationship between the input and output of a device, circuit, or system.
- In VLSI design, these characteristics are crucial for understanding and predicting the behavior of various components, especially transistors and logic gates.
- We have already studied the nMOS, pMOS and CMOS transfer characteristics.
- The transfer characteristic curve of a CMOS inverter is a plot of Vout versus Vin. It typically has a sharp transition region where a small change in Vin causes a large change in Vout.



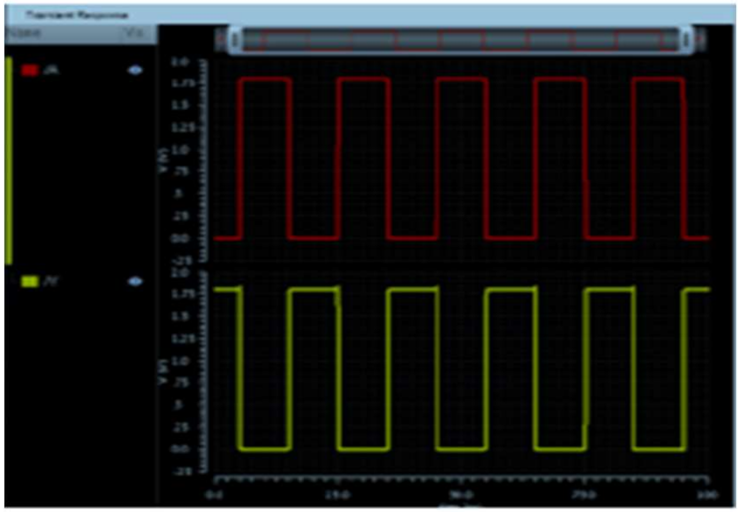
Characteristics of Different Technologies

Transient Responses:

- Transient characteristics describe the relationship between the input and output of a device, circuit, or system.

Key Parameters in Transient Response:

- Rise Time (tr): signal to rise from 10% to 90% of its final value
- Fall Time (tf): signal to fall from 90% to 10% of its final value
- Propagation Delay (tp): input change to produce an output change
- Settling Time: output to settle within a certain percentage of its final value
- Overshoot and Undershoot: when the signal exceeds its final steady-state value, while undershoot is when it drops below it before stabilizing.



Characteristics of Different Technologies

Noise Analysis:

- Noise analysis is an essential aspect of VLSI design, as it helps ensure that circuits function reliably in the presence of various noise sources.
- Noise can affect both current and voltage, leading to potential errors in circuit operation.

Types of Noise in VLSI Circuits:

- **Thermal Noise (Johnson-Nyquist Noise):** Caused by the random motion of electrons in a resistor or semiconductor.
- **Flicker Noise (1/f Noise):** Caused by traps in the semiconductor material that capture and release carriers.
- **Shot Noise:** Caused by the discrete nature of charge carriers in a current flow.
- **Burst Noise (Popcorn Noise):** Caused by the sudden and random changes in the number of charge carriers or their mobility.
- **Power Supply Noise:** Variations in the power supply voltage

Session 14

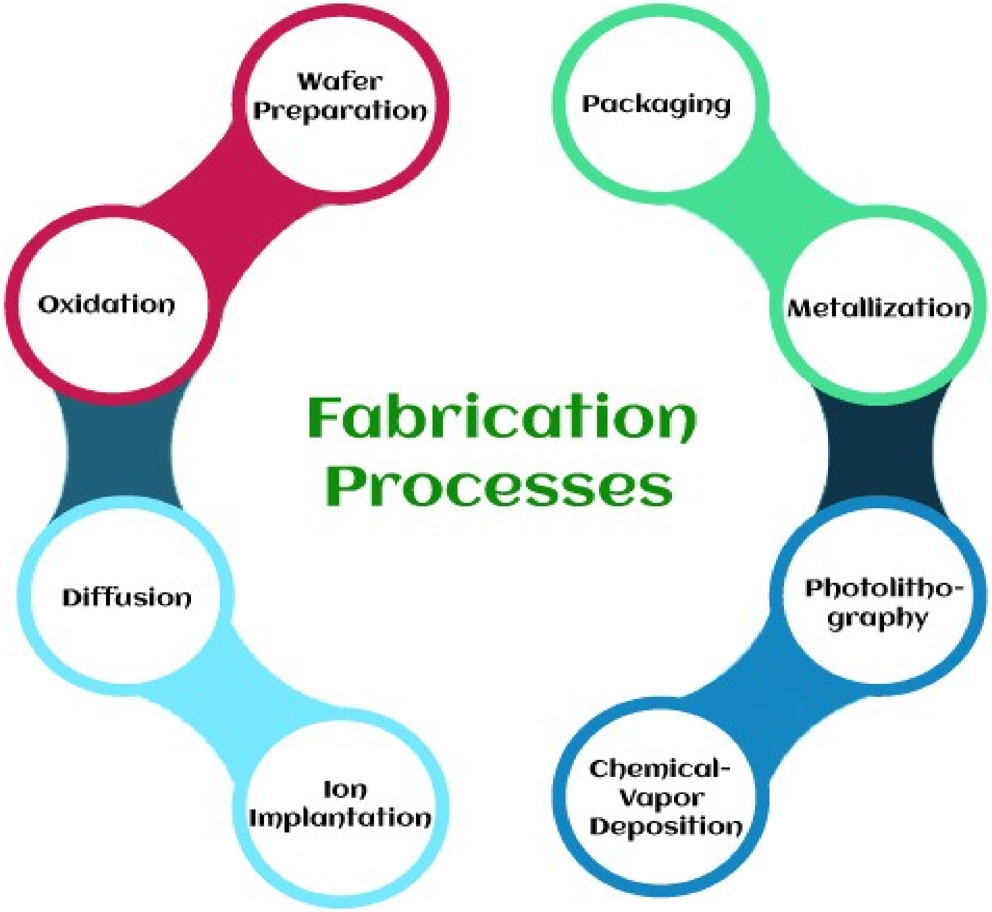
Lecture: Back-end Processes

- Fabrication methods of circuit elements, Layout design of different cells.
- Library cell designing, NAND, NOR, NOT, X-OR etc.
- Circuit Extraction, Electrical rule check, LVS, Post layout simulation and Parasitic extraction

Fabrication Methods of Circuit Elements

Fabrication Methods:

- The fabrication of circuit elements in VLSI involves a series of complex processes to create the tiny components that make up integrated circuits (ICs).
- We have studied briefly in session 8.



Layout Design of Different Cells

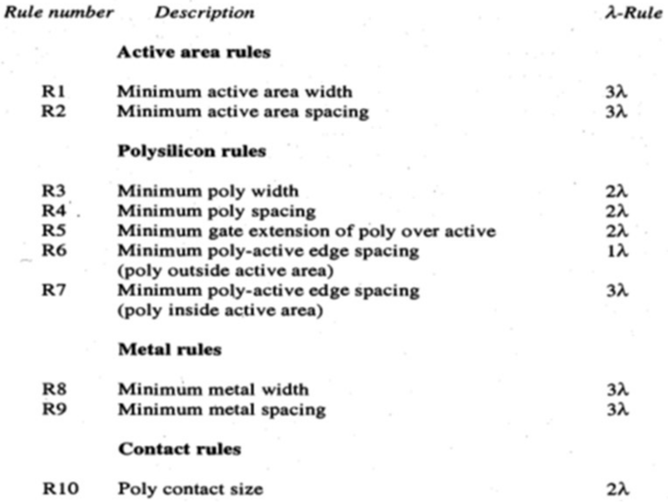
Layout Design:

- The abstract circuit design is translated into a physical layout that can be fabricated on a silicon wafer.
- We have designed NOT, NAND, NOR, AND, OR, EXOR cells in session 7.

Layout Design Rules:

(i) **Micron rules:** The layout constraints such as minimum feature sizes and minimum allowable feature separations are stated in terms of absolute dimensions in micrometers.

(ii) **Lambda rules:** It specify the layout constraints in terms of a single parameter (λ) and thus allow linear, proportional scaling of all geometrical constraints.



Rule number	*Description*	*λ-Rule*
	Active area rules	
R1	Minimum active area width	3λ
R2	Minimum active area spacing	3λ
	Polysilicon rules	
R3	Minimum poly width	2λ
R4	Minimum poly spacing	2λ
R5	Minimum gate extension of poly over active	2λ
R6	Minimum poly-active edge spacing (poly outside active area)	1λ
R7	Minimum poly-active edge spacing (poly inside active area)	3λ
	Metal rules	
R8	Minimum metal width	3λ
R9	Minimum metal spacing	3λ
	Contact rules	
R10	Poly contact size	2λ

Layout Design of Different Cells

Library Cell Designing:

- It involves the creation of a set of standard cells that are used as building blocks for creating larger integrated circuits (ICs).
- These cells include logic gates, flip-flops, multiplexers, and other essential components.
- Proper design of library cells ensures optimized performance, power consumption, and area efficiency**.**

Example: Designing a NAND Gate Cell

1. Specification Definition

- **Function**: 2-input NAND gate.
- **Operating Voltage**: 1.8V.
- **Timing Requirements**: Maximize speed while minimizing power.

2. Schematic Design

- **Transistor Configuration**:
  - Use two NMOS transistors in series and two PMOS transistors in parallel.
- **Sizing**: Optimize transistor sizes to balance speed and power.

Layout Design of Different Cells

3. Layout Design

- **Transistor Placement**: Place NMOS and PMOS transistors close to minimize routing length.
- **Routing**: Connect the gate, source, and drain regions with minimal parasitics.
- **Design Rules**: Ensure adherence to spacing and width rules.

4. DRC and LVS

- **Verification**: Run DRC to check for rule violations and LVS to ensure the layout matches the schematic.

5. Parasitic Extraction and Characterization

- **Extraction**: Extract parasitics for accurate timing and power analysis.
- **Simulation**: Simulate the cell to generate timing and power data.

6. Library Creation

- **Compilation**: Compile the characterized NAND gate into the standard cell library.

Circuit Extraction

Design Rule Checking (DRC):

Purpose: Ensure that the layout adheres to the fabrication process design rules.

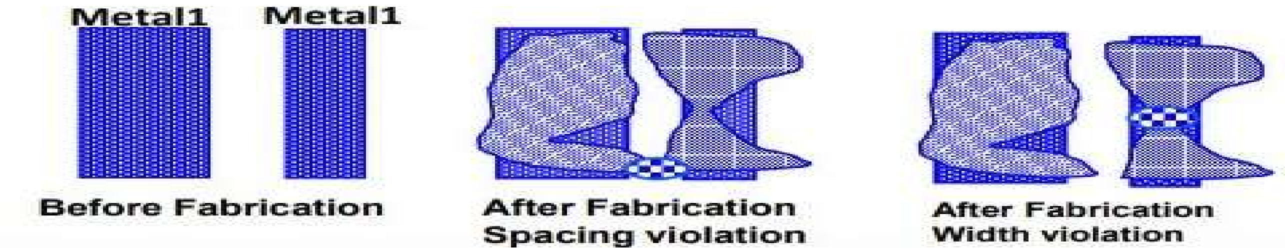
Process: Automated DRC tools check for violations such as minimum spacing, width, and overlap requirements.

Considerations:

- Ensuring no violations to prevent manufacturing defects
- Correcting any identified issues before tape-out

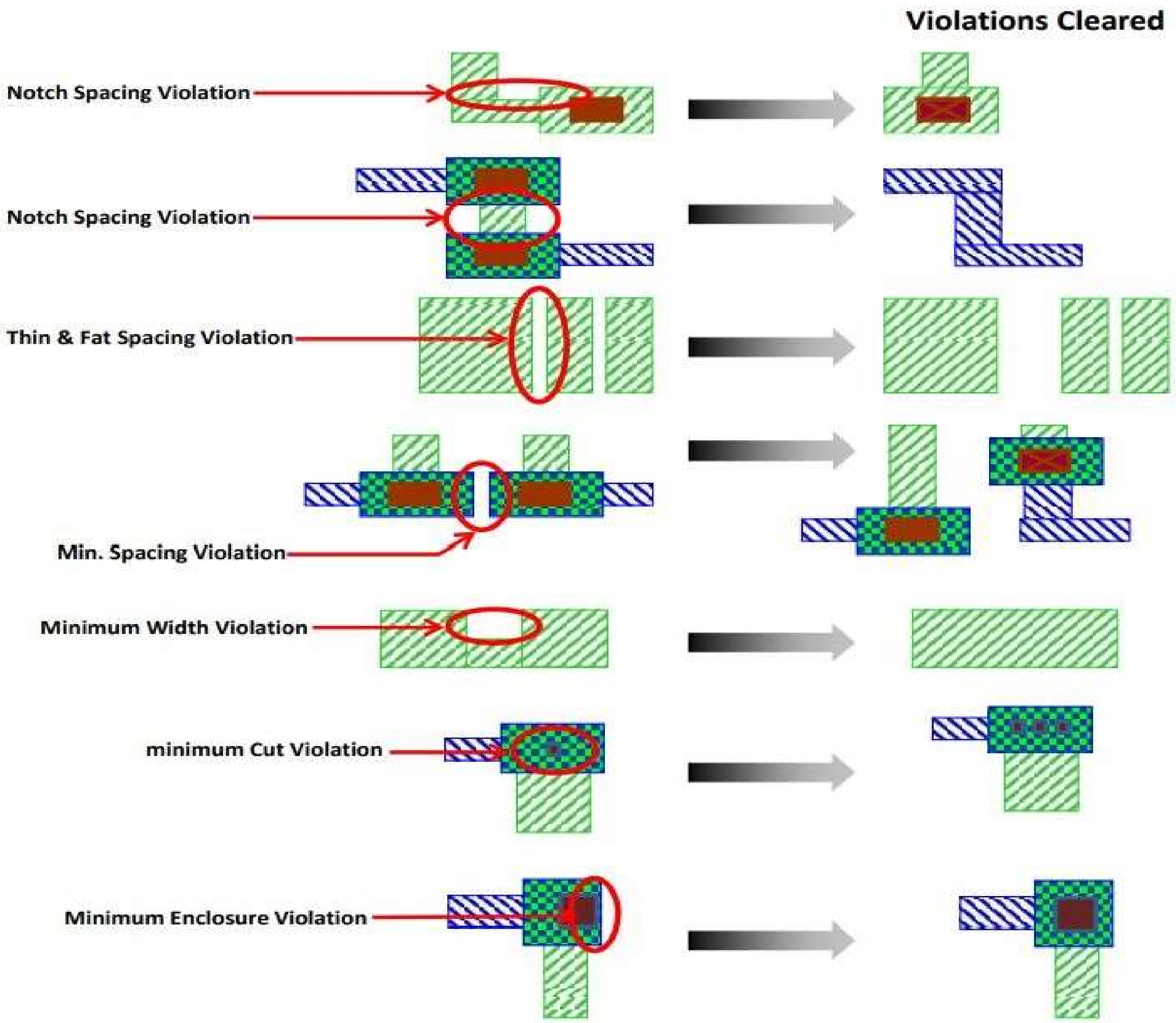
Design Rule Examples:

- ✓ **Minimum Spacing:** The minimum spacing between objects on a single layer.
- ✓ **Minimum Width:** The min width rule specifies the minimum width of individual shapes on a single layer.
- ✓ **Minimum Enclosure/ Overlap**: Implies that the second layer is fully enclosed by the first one.
- ✓ **Notch:** The rule specifies the minimum spacing rule for objects on the same net, including defining the minimum notch on a single-layer, merged object.
- ✓ **Minimum Cut:** The minimum number of cuts a via must have when it is on a wide wire.



Circuit Extraction

Design Rule Checking (DRC):



Circuit Extraction

Layout Versus Schematic (LVS):

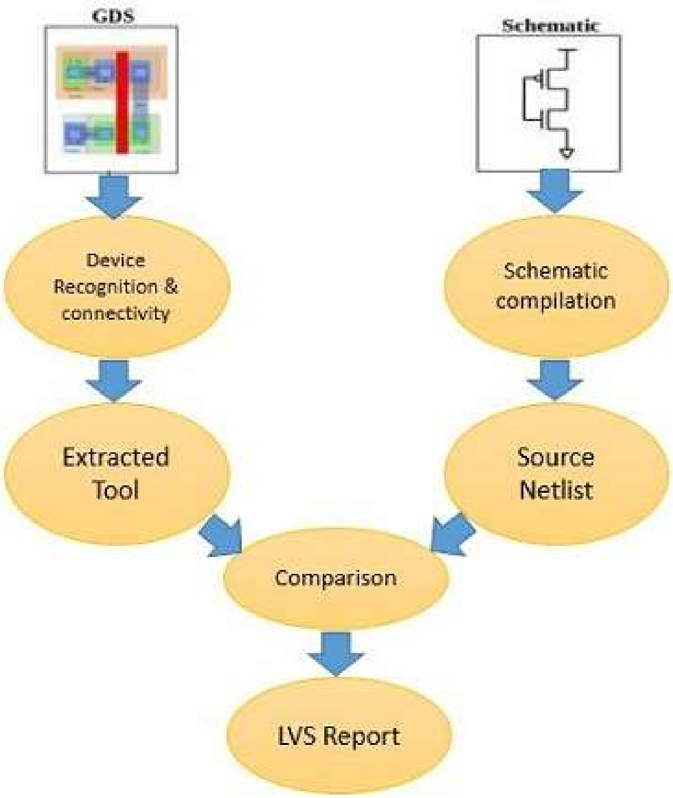
Purpose: Verify that the layout correctly implements the intended circuit design.

Process:

- Compare the layout netlist extracted from the layout with the original schematic netlist.
- Ensure that all components and connections match.

Considerations:

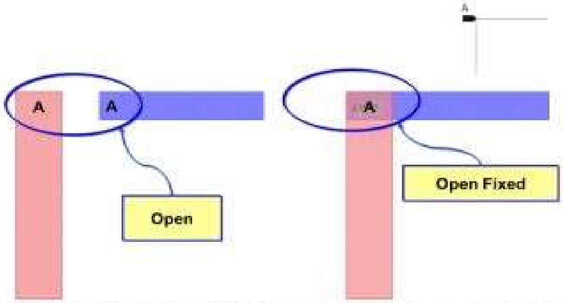
- Identifying and correcting mismatches
- Ensuring functional equivalence



LVS checks examples:

✓ Short Net Error, Open Net Error, Extract errors

Open Net Error:

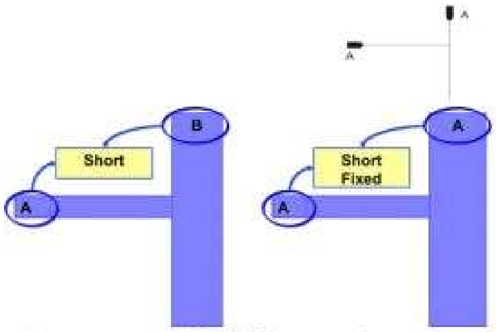


Same net is routed in two different metal layers but not connected

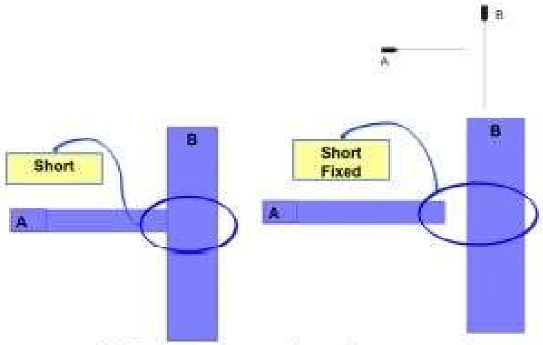
Circuit Extraction

Layout Versus Schematic (LVS):

Short Net Error:



Same net with different pin names

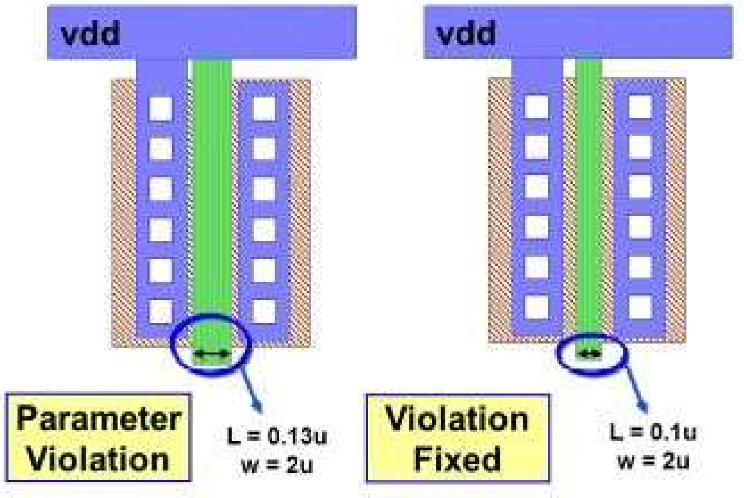


Two different nets shorting together

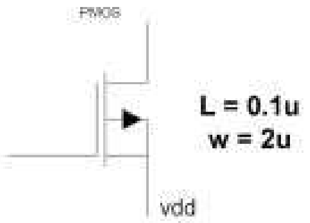
Extract Errors:

- Parameter mismatch.
- Device parameters on schematic and layout are compared.
- LVS check width, length, multiplication factor etc.

Layout :



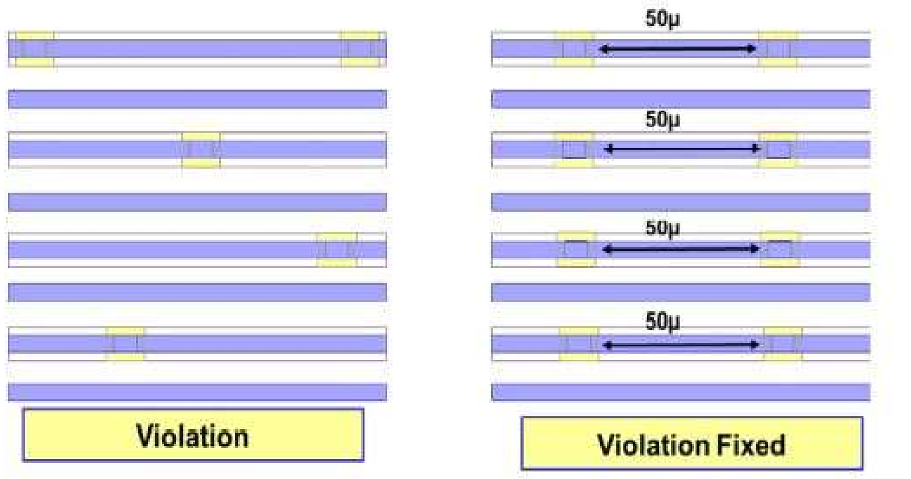
Schematic:

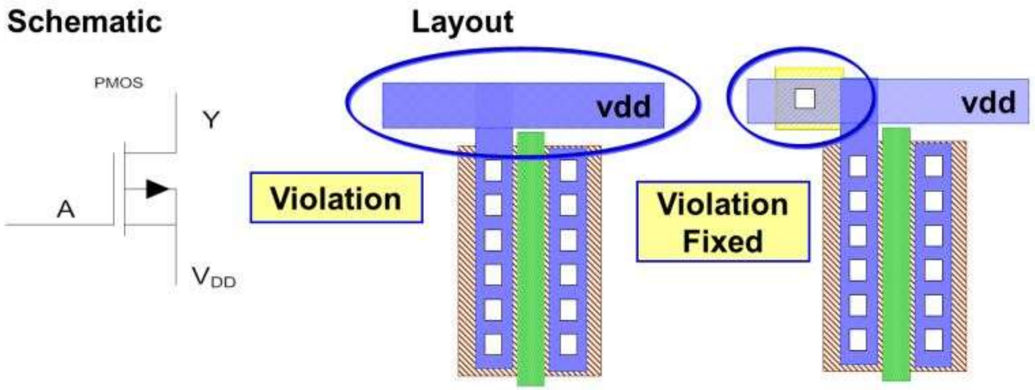


Circuit Extraction

Electrical Rule Check (ERC):

- It focuses on verifying that the electrical properties and connections in the design adhere to predefined rules.
- By verifying voltage levels, current densities, power and ground connections, and other critical electrical parameters, ERC helps identify and correct potential issues before fabrication.
- **ERC checks include:**
  - Voltage Level Check
  - Current Density Check
  - Power and Ground Connection Check
  - Unconnected Pin Check
  - Short Circuit Check





Circuit Extraction

Post Layout Simulation:

Purpose: Verify the performance of the circuit with parasitics included.

Process:

- Perform timing analysis, power analysis, and signal integrity analysis.
- Ensure the design meets all specifications under worst-case scenarios.

Considerations:

- Ensuring timing closure with parasitics
- Validating power consumption and thermal performance

Parasitic Extraction:

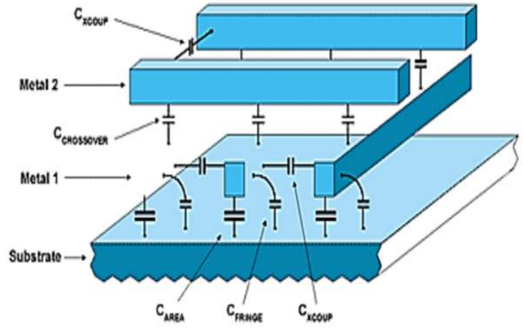
Purpose: Extract parasitic capacitances, resistances, and inductances from the layout.

Process:

- Use extraction tools to generate a detailed parasitic model of the interconnects.
- This model is used for post-layout simulation.

Considerations:

- Accurate modeling of parasitics for precise timing analysis
- Incorporating parasitics into the final verification process



Session 15 & 16

Lecture: Back-end Designing and Analysis

- Design format, timing analysis, back annotation and post layout simulation of silicon circuit
- Study of design Issues: Antenna effect, Electro migration effect, body effect, Inductive and capacitive cross talk, Drain punch through, etc.
- Placement of cells, placing of I/O blocks, Initialization of floor planning, routing and creating design data base
- DFT Guidelines, Test Pattern and BIST
- ASIC design implementation

Back-end Designing and Analysis

Back-end Designing:

- Back-end designing and analysis in VLSI involves the physical implementation of the design, starting from the synthesized netlist to the generation of the final layout ready for fabrication.
- This phase focuses on placing the circuit components, routing the interconnections, optimizing performance, and verifying that the design meets all physical and timing requirements.

Design Format:

- It plays a critical role in ensuring smooth transitions between different stages of the design process.
- Each format serves a specific purpose, from representing the circuit structure and physical layout to modeling timing, power, and parasitic effects.

Back-end Designing and Analysis

Key Design Formats or Inputs of Physical Design:

a) **Gate level netlist (.v):** modules information, ports information, cells (standard cells, macro's, MV cells), nets

b) **Timing constraints (.sdc):** mcmm (multi corner and multi-mode)/scenario

c) **Libraries**

   i) Logical (.lib/.db): max transition/slew, area, functionality, Table: transition, cell delay (NLDM / CCS), net delay (WLM), setup time, hold time, power

   ii) Library exchange format (.lef): shape of the cell, metal layers, pins shape, antenna ratio

d) **Techology file (.tf/.techlef):** units, colors, TILE, via, metal layers: color, Physical DRC's: width, spacing, pitch, area

e) **RC parasitics (.tlup/.spef/.nxtgrd):** Metal layer parasitical information

f) **Scan chain information (.scandef)**

g) **Port location/pad location (.tdf/.io):** tcl file

h) **Design exchange format (.def):** I/O file

i) **Unified power format (.upf):** multi power, power design & power source

j) **Specification file:** power, cts

Back-end Designing and Analysis

Timing Analysis:

- It ensures the correct operation of digital circuits by verifying that timing constraints are met across all paths in the design.
- It involves evaluating the propagation delays of signals through the circuit to ensure that setup and hold times are respected, thus preventing timing violations that can lead to incorrect behavior or even failure of the circuit.

Types of Timing Analysis:

1. Static Timing Analysis (STA)

Purpose: Predicts the timing behavior of a circuit without considering input signal timing variations.

Process:

- ○ **Setup Time Analysis**: Checks if data signals arrive at flip-flops before the clock edge.
- ○ **Hold Time Analysis**: Verifies that data signals remain stable after the clock edge.
- ○ **Propagation Delay Analysis**: Measures the delay through combinational logic paths.

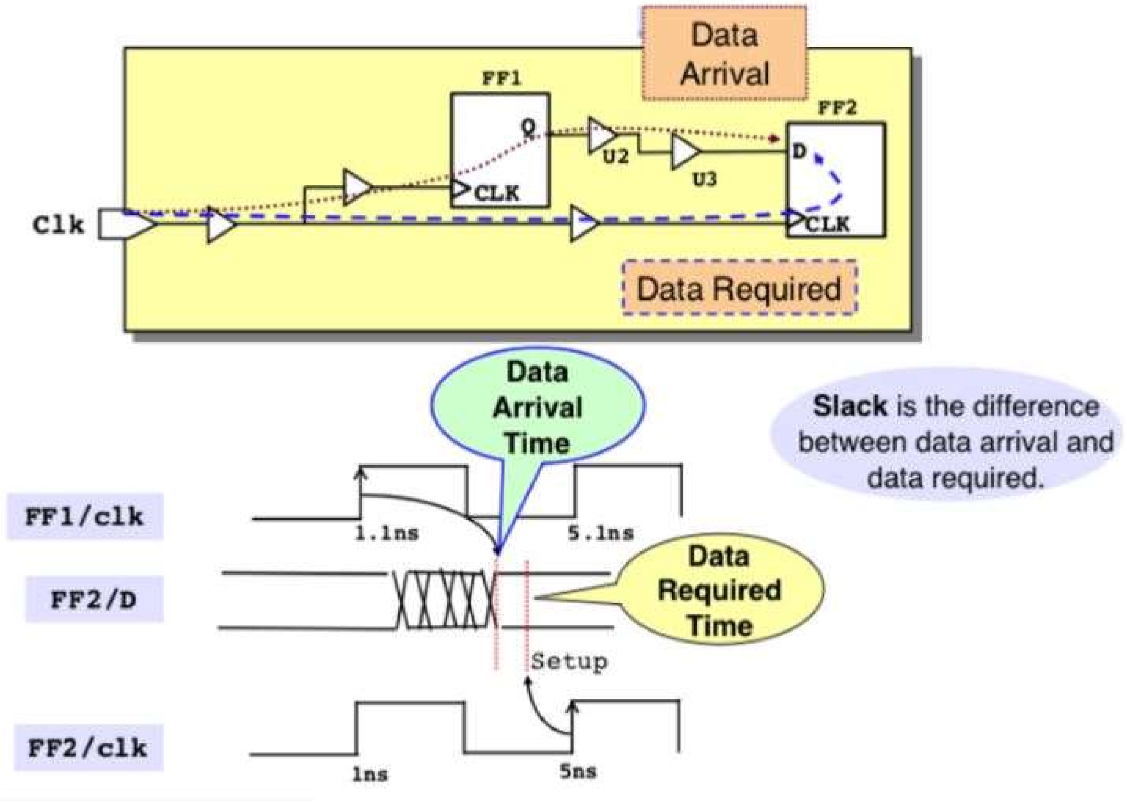
Usage: Used during the back-end design phase to ensure timing closure before tape-out.

Tools: PrimeTime (Synopsys), Tempus (Cadence), Encounter Timing System (Mentor Graphics).

Back-end Designing and Analysis

2. Dynamic Timing Analysis

Purpose: Evaluates timing under changing conditions such as input signal transitions and circuit activity.
Process: Simulates the circuit under different operational scenarios to analyze worst-case timing paths.
Usage: Used for detailed timing verification, especially in high-speed designs or where signal transitions are critical.



Back-end Designing and Analysis

Back Annotation:

- It used to refine the accuracy of simulations and timing analysis by incorporating real-world parasitic effects and other physical characteristics from the layout into the design.
- This ensures that the design behaves as expected when fabricated.

Steps in Back Annotation:

1. **Layout Extraction**
   - Extract parasitic information from the layout using tools like Calibre, StarRC, or Quantus.
2. **Parasitic Extraction**
   - Parasitics include resistances (R), capacitances (C), and sometimes inductances (L) associated with the interconnects.
   - Tools like Synopsys StarRC, Cadence Quantus, and Mentor Graphics Calibre PEX are commonly used for parasitic extraction.

Back-end Designing and Analysis

Steps in Back Annotation:

3. **Back Annotating the Netlist**
   - Modify the original netlist to include the extracted parasitic information.
4. **Timing Analysis**
   - Perform Static Timing Analysis (STA) on the back-annotated netlist to verify that timing constraints are met.
   - Tools like Synopsys PrimeTime, Cadence Tempus, and Mentor Graphics TimeQuest can be used.
5. **Simulation**
   - Use the back-annotated netlist in simulation to verify the functional and timing behavior.
   - Simulators like Cadence Spectre, Synopsys HSPICE, and Mentor Graphics ModelSim are commonly used.

Back-end Designing and Analysis

Post layout simulation of silicon circuit:

- It ensures that the circuit behaves as expected under real-world conditions, incorporating the effects of layout parasitics and other physical characteristics.

Purpose of Post-Layout Simulation

1. **Verification of Timing and Functionality**: Ensures that the circuit meets timing requirements and functions correctly with the actual layout.
2. **Inclusion of Parasitic Effects**: Accounts for parasitic resistances, capacitances, and inductances that can affect signal integrity and timing.
3. **Power Analysis**: Provides a more accurate assessment of power consumption considering the actual layout.
4. **Signal Integrity**: Evaluates crosstalk, noise, and other signal integrity issues that can arise due to the physical layout.

Back-end Designing and Analysis

Study of Design Issues: Antenna effect, Electro migration effect, body effect, Inductive and capacitive cross talk, Drain punch through, etc.

Antenna Effect:

- Long metal lines and vias introduce antenna violations.

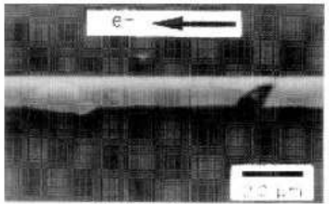
- VLSI process starts from the substrate, device layer and then metal layers. The Etch process builds up the electrical charges on metal layers. These charges cause a high voltage spike, which may damage the gates connected to the metals.

- Three basic techniques are to prevent antenna violations are:
  - Jumper Insertion or Metal hoping
  - Floating gate attachment
  - Dummy transistor insertion
  - Antenna diode insertion

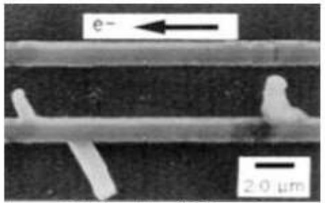
Back-end Designing and Analysis

Electromigration Effect:

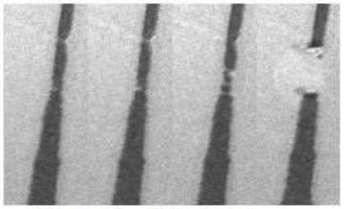
- Electron migration is the movement of atoms based on the flow of current through a material. If the current density is high enough, the heat dissipated within the material will repeatedly break atoms from the structure and move them.
- Displacement of the atoms as a result of current flowing through a conductor which causes voids and failures in a device.
- Electromigration (EM) analysis in VLSI design refers to optimizing IC interconnects to prevent electrochemical growth.
- EM can reduce by wire widening, frequency reduction, lower the supply voltage, short wire length and cell sizing.



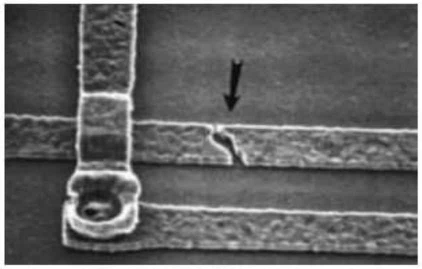
Voids/ Open



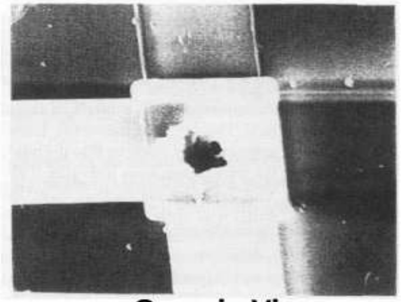
Hillocks/ Short



Short in Metal layer



Open in Metal layer



Open in Via

Back-end Designing and Analysis

Body Effect:

- The "body effect," also known as the "substrate bias effect," is a phenomenon that affects the threshold voltage (Vth) of MOSFET.

- It arises due to the interaction between the substrate (body) of the transistor and the channel region under different biasing conditions.

- The body effect can also vary with temperature due to changes in carrier concentration and mobility in the substrate, influencing transistor performance in temperature-sensitive applications.

- In some advanced CMOS designs, body biasing techniques (forward or reverse biasing of the substrate) are used to dynamically adjust Vth to optimize transistor performance, reduce leakage currents, or improve speed.

Back-end Designing and Analysis

Inductive and Capacitive Cross Talk:

- Crosstalk refers to undesired or unintentional effects, which can cause functional failure in the chips.
- Inductive and capacitive crosstalk are two types of signal interference that occur in integrated circuits (ICs) and printed circuit boards (PCBs).

Aspect	**Capacitive Crosstalk**	**Inductive Crosstalk**
Nature of Coupling	Electric field interaction between adjacent conductors	Magnetic field interaction between current-carrying conductors
Primary Cause	Voltage changes (dV/dt) on the aggressor line	Current changes (dI/dt) on the aggressor line
Effects on Signals	Voltage noise, timing delays, signal distortion	Current noise, timing jitter, power integrity issues
Mitigation Techniques	Increased spacing, shielding, low-swing signals, differential signaling	Controlled impedance, twisted pairs, decoupling capacitors, shorter signal paths

Back-end Designing and Analysis

Drain Punch Through:

- It happens when the depletion regions of the drain and the source extend towards each other and merge, causing a direct path for current to flow from the drain to the source, bypassing the control of the gate.
- This leads to undesirable leakage currents and can significantly affect the device's performance.

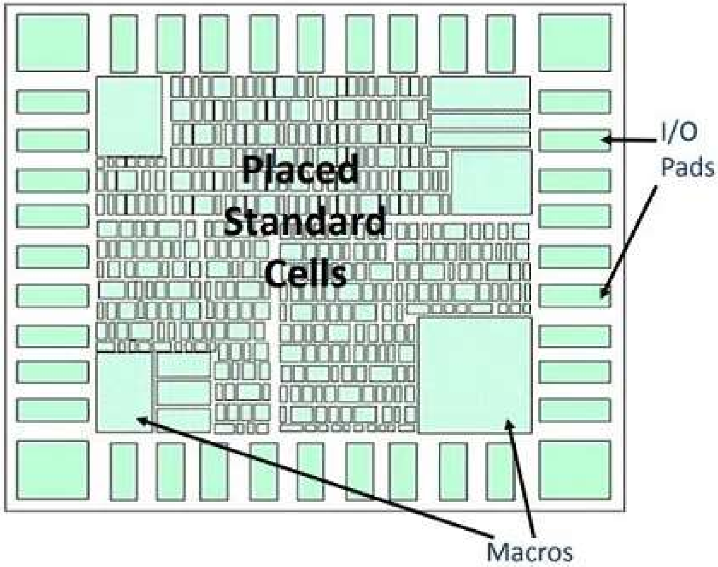
Effects of Drain Punch-Through

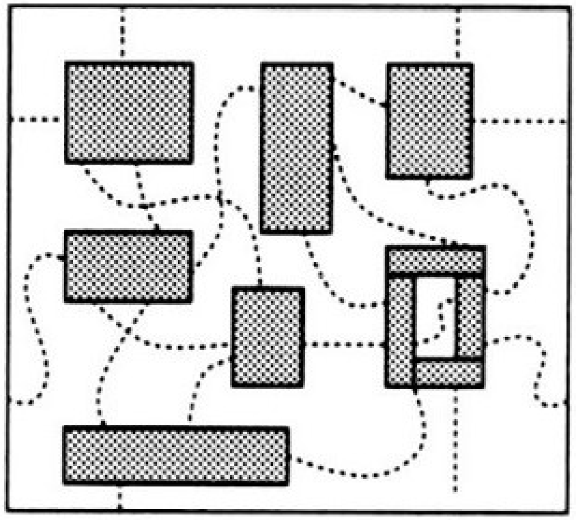
1. **Increased Leakage Currents:** Drain punch-through leads to a significant increase in off-state leakage currents, affecting power consumption and thermal characteristics.
2. **Threshold Voltage Shift:** The threshold voltage (VT) of the MOSFET can be altered due to punch-through, leading to changes in device performance and switching characteristics.
3. **Reduced Reliability:** Increased leakage and altered performance can affect the long-term reliability of the device, causing degradation over time.
4. **Performance Degradation:** The uncontrollable current flow reduces the effective control of the gate, leading to degraded transistor performance, such as reduced gain and speed.

Back-end Designing and Analysis

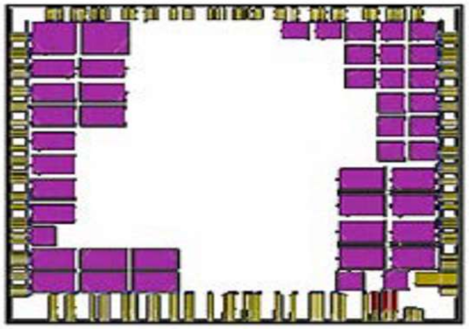
Placement of cells, placing of I/O blocks, Initialization of floor planning, routing and creating design data base:

- It is already discussed briefly in session 12.

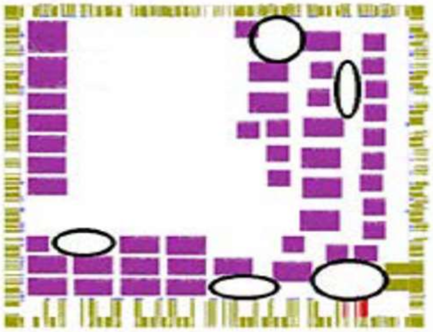




Homogeneous Standard Cell Area With Aligned Macros



Irregular Macro Placement With Traps for Standard Cells



DFT Guidelines, Test Pattern and BIST

Design for Testability (DFT) :

- It is a crucial aspect of IC design that involves incorporating test features into the design to facilitate testing during manufacturing, thereby ensuring high yield and reliability.
- The keys to designing circuits that are testable are **controllability and observability**.
- Restated, controllability is the ability to set (to 1) and reset (to 0) every node internal to the circuit.
- Observability is the ability to observe, either directly or indirectly, the state of any node in the circuit.
- DFT may be categorized as follows:
  - ✓ Ad hoc testing
  - ✓ Scan-based approaches
  - ✓ Built-in self-test (BIST)

1. Ad hoc testing:

- ▪ It is collections of ideas aimed at reducing the combinational explosion of testing.

DFT Guidelines, Test Pattern and BIST

2. Scan Design:

Scan Insertion: Convert flip-flops into scan flip-flops that can form a scan chain during test mode.

Scan Chain Design: Ensure that scan chains are well-structured to minimize routing complexity and timing issues.

Test Access Points: Insert test access points, such as multiplexers, to control and observe internal nodes.

3. Test Patterns:

- ✓ Test patterns are sequences of input vectors applied to an IC during testing to verify its functionality and detect manufacturing defects.
- ✓ These patterns are essential for identifying faults and ensuring that the IC performs as expected under various conditions.
- ✓ Test patterns can be generated for different types of testing, including functional testing, structural testing, and at-speed testing.

DFT Guidelines, Test Pattern and BIST

4. Built-In Self-Test (BIST)

- It a design technique in which test mechanisms are integrated into the hardware, allowing the system to test itself.
- BIST is used to detect and diagnose faults within an integrated circuit (IC) or system-on-chip (SoC) without the need for external test equipment.
- BIST can be applied to various types of circuits, including digital logic, memory, and analog/mixed-signal circuits.

Advantages of BIST

- **Reduced Test Cost**: By embedding the test circuitry within the IC, the need for expensive external test equipment is minimized.
- **Increased Test Coverage**: BIST can achieve high fault coverage by running comprehensive test patterns.
- **Self-Testing Capability**: BIST enables self-testing in the field, allowing for periodic checks and maintenance.
- **Faster Test Times**: BIST can significantly reduce the time required for testing by executing tests at the system's operating speed.
- **Enhanced Reliability**: Continuous or periodic self-testing can enhance the reliability and predictability of the system.

DFT Guidelines, Test Pattern and BIST

Types of BIST

1. **Logic BIST (LBIST)**
   - **Purpose**: Tests the digital logic components of the IC.
   - **Components**: Typically includes a pattern generator, a response analyzer, and a test control unit.
   - **Pattern Generation**: Uses pseudo-random pattern generators (e.g., Linear Feedback Shift Registers, LFSRs) or deterministic pattern generators.
   - **Response Analysis**: Uses signature analyzers (e.g., Multiple Input Signature Registers, MISRs) to compact the test responses and compare them with expected signatures.
2. **Memory BIST (MBIST)**
   - **Purpose**: Tests embedded memory blocks within the IC.
   - **Components**: Includes a test pattern generator, a test controller, and a comparison unit.
   - **Test Algorithms**: Implements various memory test algorithms such as March tests, walking 1s/0s, and checkerboard patterns to detect common memory faults.

DFT Guidelines, Test Pattern and BIST

3. **Analog BIST (ABIST)**
   - **Purpose**: Tests analog and mixed-signal components of the IC.
   - **Components**: May include built-in signal generators, analog-to-digital converters (ADCs), and digital-to-analog converters (DACs) for generating and analyzing test signals.
   - **Test Techniques**: Uses techniques like loopback tests, ramp tests, and frequency response analysis to verify analog functionality.

Components of a BIST System

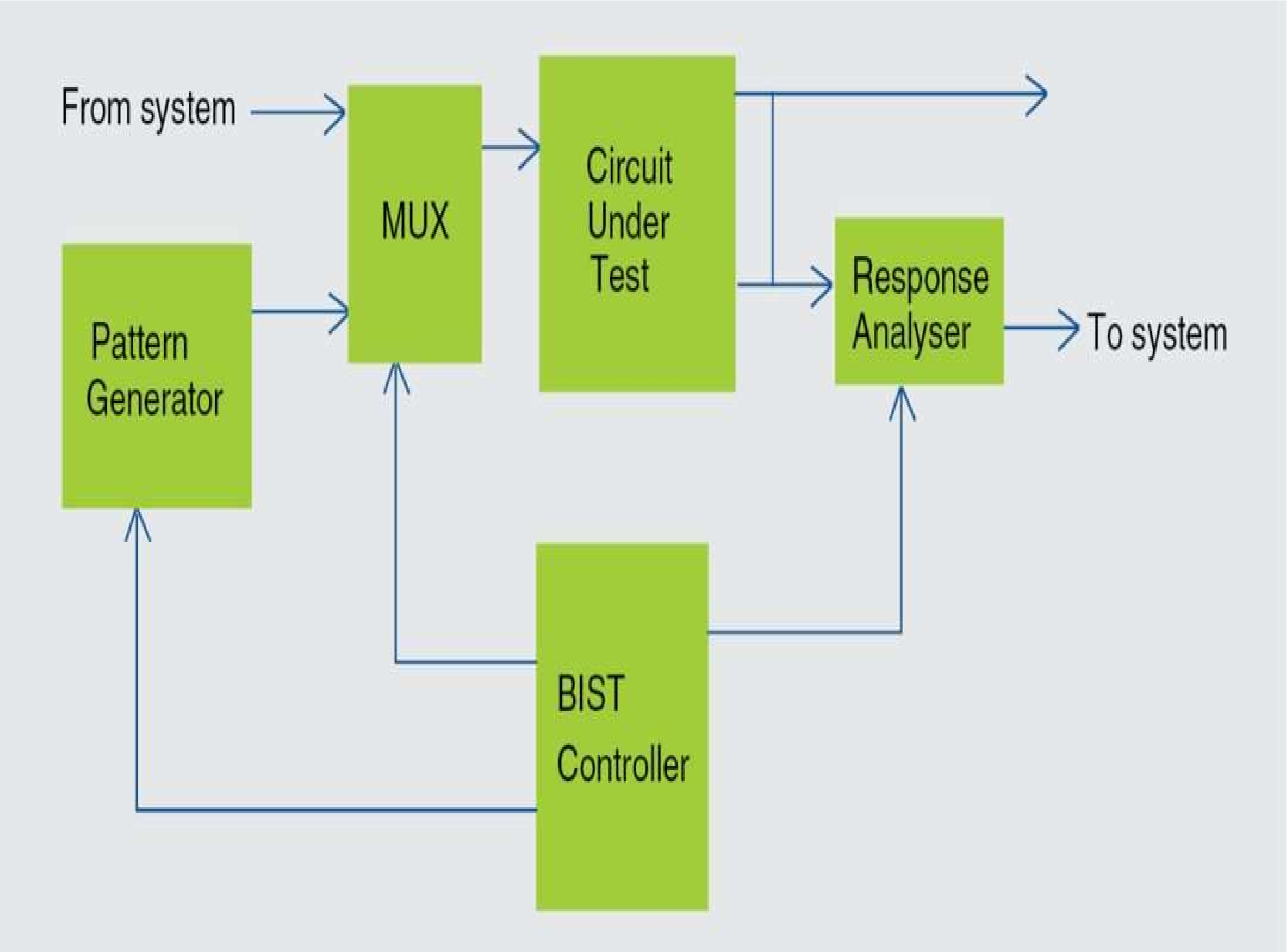
a. **Test Pattern Generator**
   - Generates the test stimuli to be applied to the circuit under test (CUT).

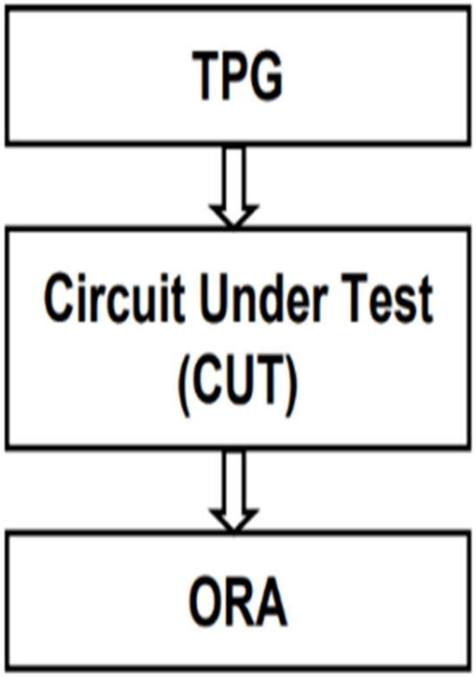
b. **Response Analyzer**
   - Collects and analyzes the output responses from the CUT.

c. **Test Controller**
   - Coordinates the overall test process, including test initialization, pattern application, and result analysis.

d. **Signature Analyzer**
   - Compacts the output responses into a signature that can be compared with a known good signature to detect faults.

DFT Guidelines, Test Pattern and BIST

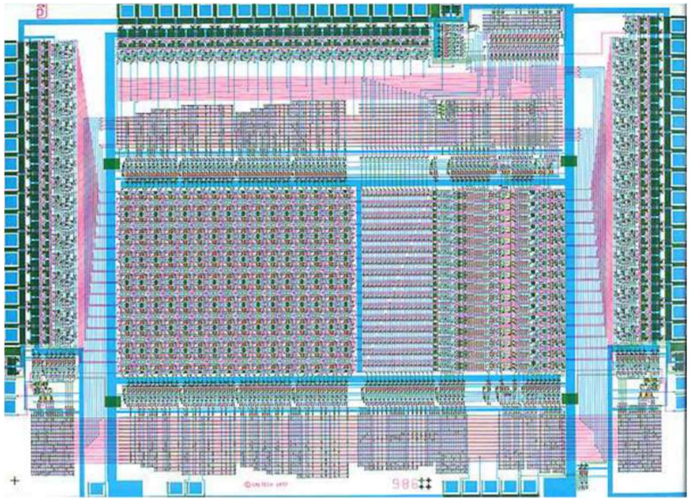




ASIC Design Implementation

ASIC Design Implementation:

- The implementation of an ASIC design involves several steps, from the initial concept to the final chip.
- This process requires a combination of design, verification, and manufacturing techniques to ensure the ASIC meets all functional, performance, and reliability requirements.
- We have already discussed briefly in session 12.



Tools Commonly Used in ASIC Design:

RTL Design: VHDL, Verilog, SystemVerilog

Simulation: ModelSim

Synthesis: Synopsys Design Compiler, Cadence Genus

DFT: Synopsys DFTMAX, Mentor Tessent

Floorplanning and Placement: Cadence Innovus, Synopsys ICC

Routing: Cadence Innovus, Synopsys ICC

Timing Analysis: Synopsys PrimeTime

Physical Verification: Mentor Calibre, Cadence Pegasus

Power Analysis: Cadence Voltus, Synopsys PrimePower

THANK YOU