Page 1: Introduction to VLSI

What is VLSI?

Very Large Scale Integration (VLSI) is the process of integrating thousands to millions of transistors on a single silicon semiconductor microchip. It is the fundamental technology behind all modern electronic devices, enabling compact, high-performance, and low-power integrated circuits (ICs).

History:

- SSI (Small Scale Integration): ~10 transistors

- MSI (Medium Scale Integration): 100s of transistors

- LSI (Large Scale Integration): 1000s of transistors

- VLSI: 10,000 to billions of transistors

Applications:

- Microprocessors
- ASICs (Application-Specific ICs)
- FPGAs (Field-Programmable Gate Arrays)
- SoCs (System on Chip)

Page 2: VLSI Design Flow
1. **Specification**
- Define system functionality, performance, and power requirements.
2. **Architectural Design**
- Choose datapaths, control units, memory organization.
3. **RTL Design (Register Transfer Level)**
- Code written in Verilog/VHDL to define logic.
4. **Functional Verification**
- Simulate and verify the correctness of RTL.
5. **Synthesis**
- Convert RTL into gate-level netlist.
6. **Static Timing Analysis (STA)**
- Analyze and validate timing of design paths.
- MIGINES GIIU VAIIUAIS IIIIIIIU VI USSIUI DAIIIS.

7. **Physical Design**

- Floorplanning, placement, routing.

- DRC, LVS, power checks, then send to foundry.

8. **Signoff and Fabrication**

Role: Translate design specifications into RTL code.
Tools:
- Synopsys Design Compiler
- Cadence Genus
Key Skills:
- Verilog/VHDL
- FSM design
- Pipelining, parallelism
- Writing constraints (SDC)
- Understanding of synthesis reports
Output: Gate-level netlist

Page 3: Domain 1 - RTL Design and Synthesis (Front-End)

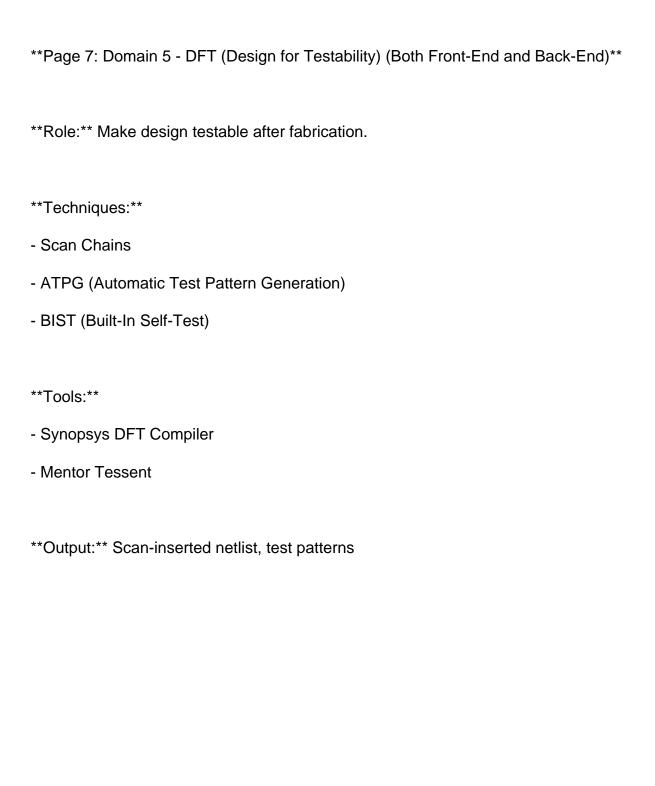
Page 4: Domain 2 - Verification (Front-End)
Role: Ensure the RTL behaves as per specifications.
Types:
- Functional Verification
- Formal Verification
- Low-power Verification
Tools:
- Synopsys VCS, Cadence Xcelium, Mentor Questa
Methodologies:
- UVM (Universal Verification Methodology)
- SystemVerilog Assertions (SVA)
- Coverage-driven verification
Output: Testbenches, coverage reports, bug reports

Role: Convert netlist to physical layout.
Phases:
1. Floorplanning
2. Placement
3. Clock Tree Synthesis (CTS)
4. Routing
5. Signoff (DRC, LVS)
Tools:
- Cadence Innovus
- Synopsys IC Compiler
Output: GDSII file for fabrication

Page 5: Domain 3 - Physical Design (Back-End)

Role: Ensure timing requirements are met.
Key Concepts:
- Setup/Hold time
- Clock skew and jitter
- Constraints and exceptions
Tools:
- PrimeTime
- Tempus
Output: Timing reports, constraint files

Page 6: Domain 4 - Static Timing Analysis (STA) (Back-End)



Page 8: Domain 6 - Analog and Mixed-Signal Design (Analog Domain)
Role: Design analog blocks or integrate with digital blocks.
Circuits:
- Op-amps, comparators
- ADCs/DACs
Key Concepts:
- Noise, gain, bandwidth
- Layout matching techniques
Tools:
- Cadence Virtuoso
Output: Schematics, layout, simulation results

Page 9: Domain 7 - FPGA Design (Front-End)
Role: Prototype or develop reprogrammable logic.
Languages: Verilog/VHDL
Tools:
- Xilinx Vivado
- Intel Quartus
Use Cases:
- Rapid prototyping
- Emulation
Output: Bitstream file for FPGA

