

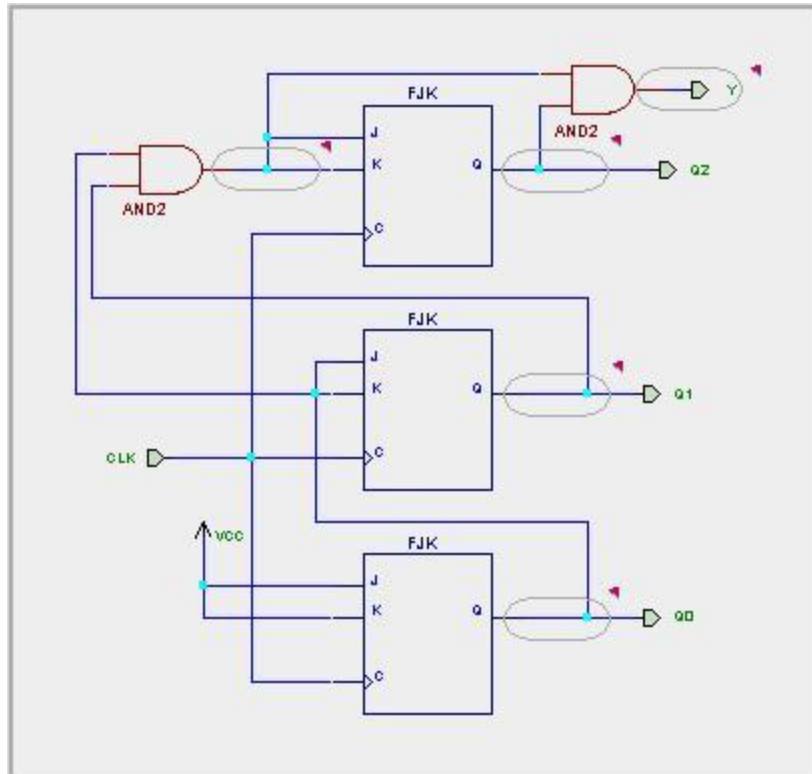
# Verilog HDL: A solution for Everybody

**Pravin Zode**

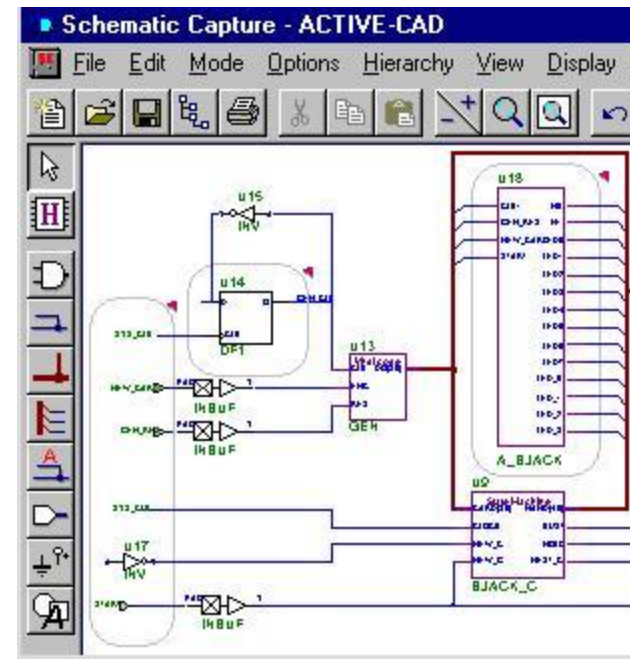
# Outline

- Traditional Design Approach
- Complexity and Design
- Design Abstraction
- Modern Digital Design Flow
- Importance of HDLs
- Comparison of HDLs

# Traditional Design approaches



Gate Level Design

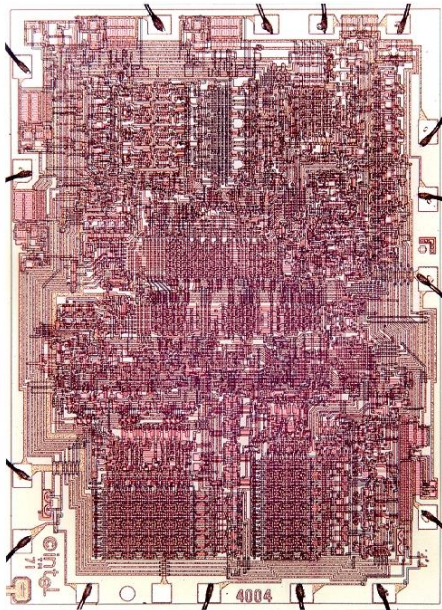


Schematic Design

# Where is the problem?

- System specification is behavioral
- Manual Translation of design in Boolean equations
- Handling of large Complex Designs
- Can we still use SPICE for simulating Digital circuits?

# Advancements over the years

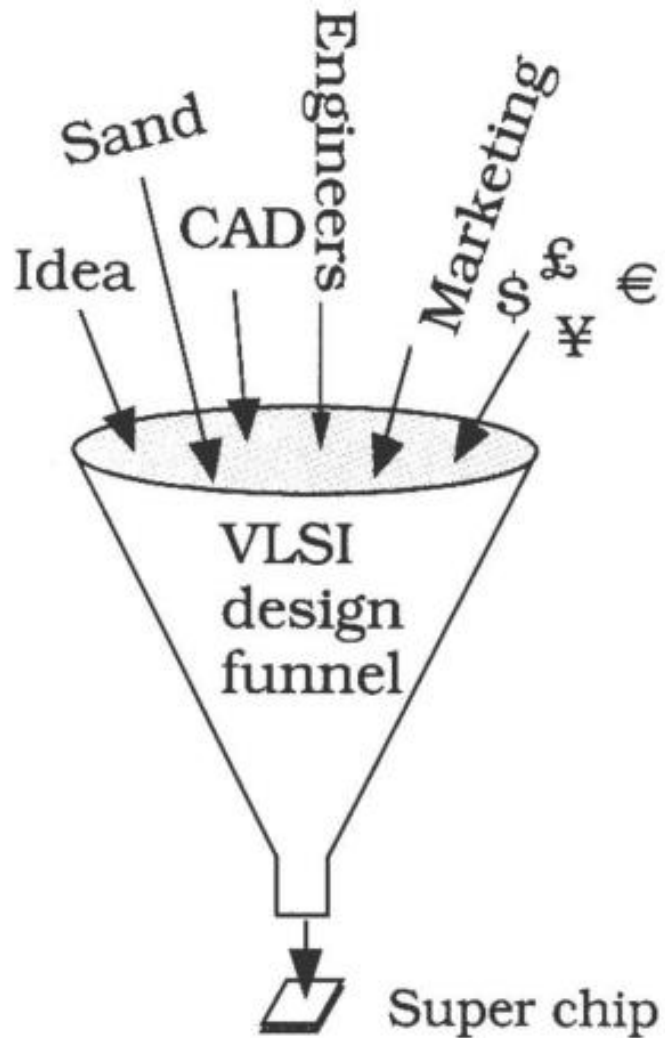


- © Intel 4004 Processor
- Introduced in 1971
- 2300 Transistors
- 108 KHz Clock



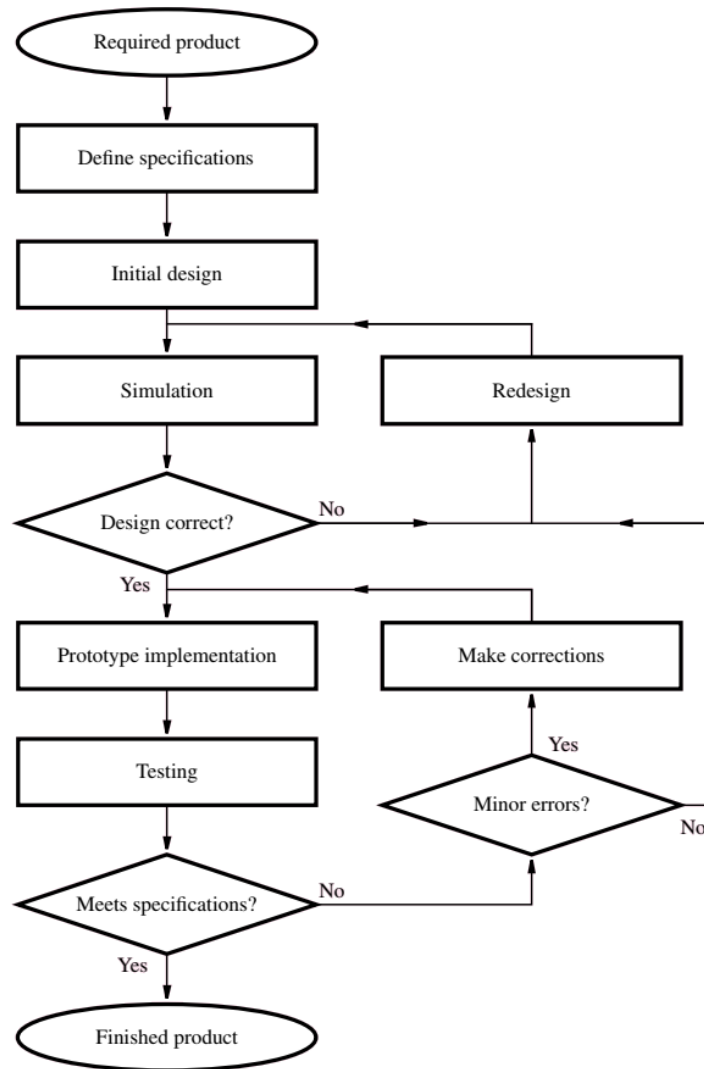
- © Intel P4 Processor
- Introduced in 2000
- 40 Million Transistors
- 1.5GHz Clock

# Complexity and Design

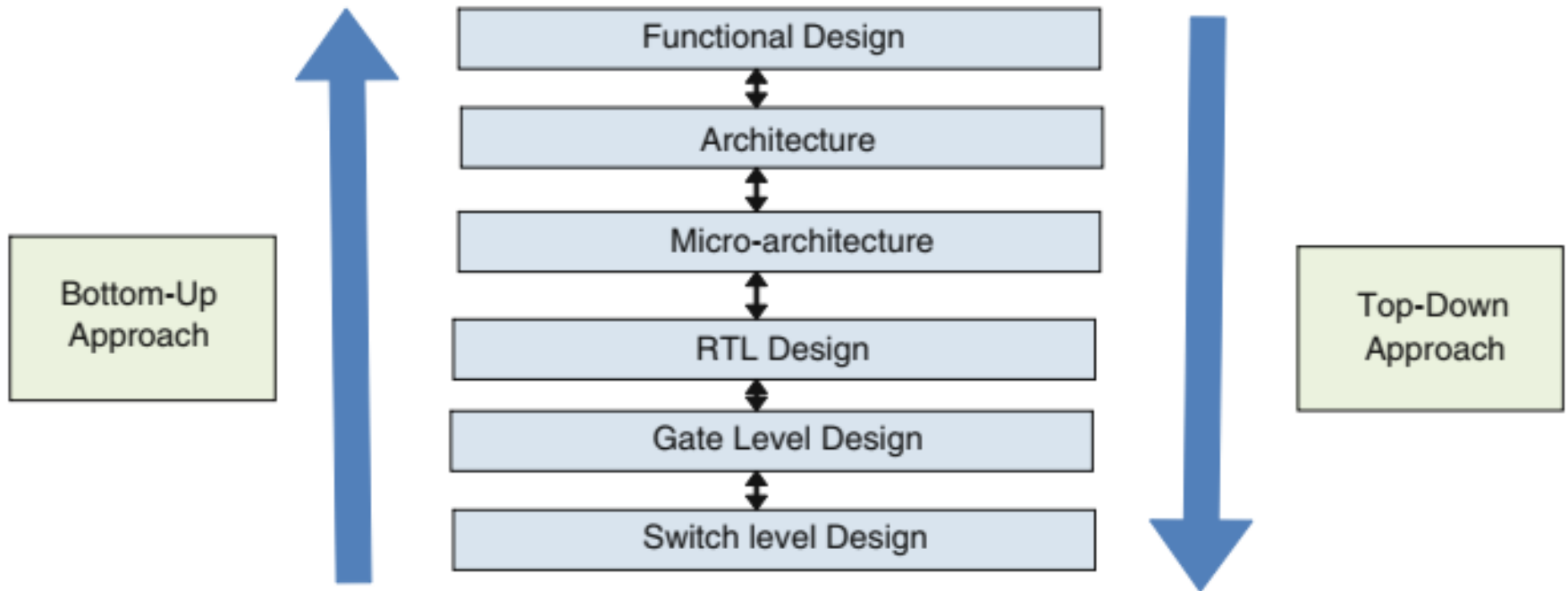


- Creating a design team provides a realistic approach to approaching a VLSI project
- it allows each person to study small sections of the system
- Needing hundreds of engineers, scientists, and technicians
- Needing hierarchy design and many different Level Views
- Everyone of each level depends upon the CAD tools

# Development Process



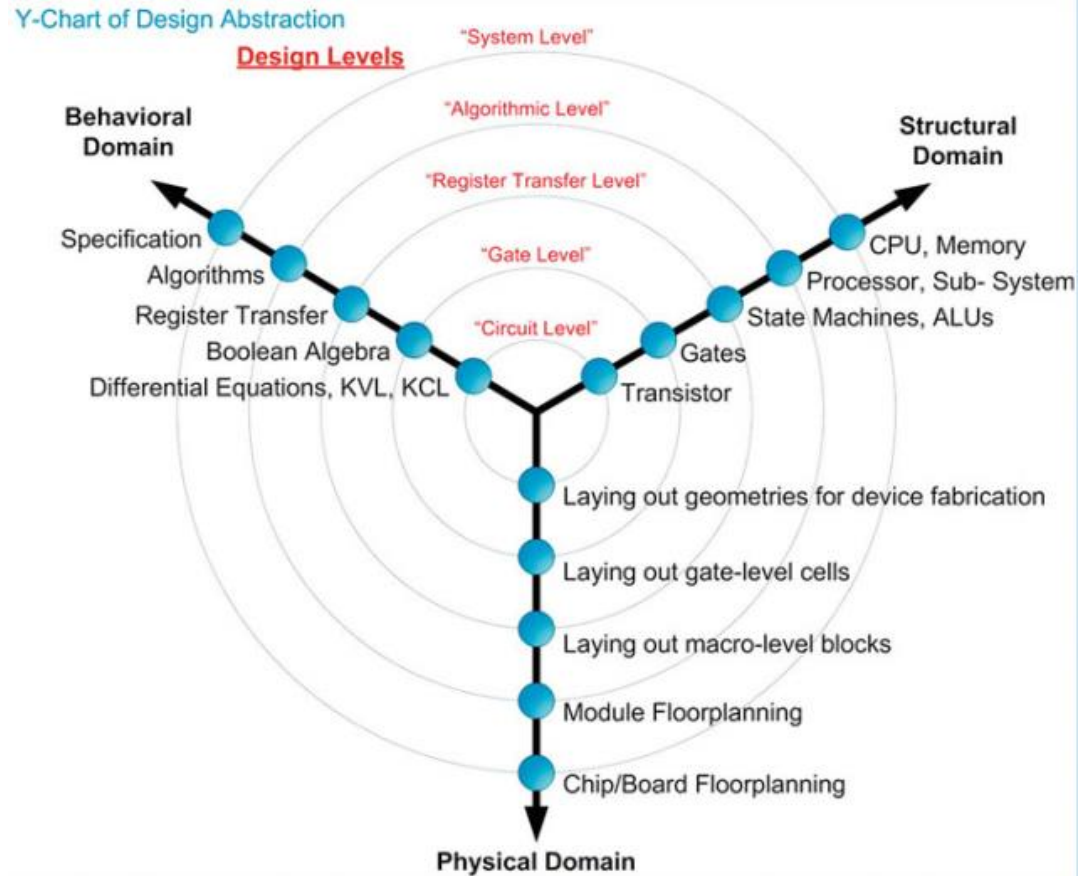
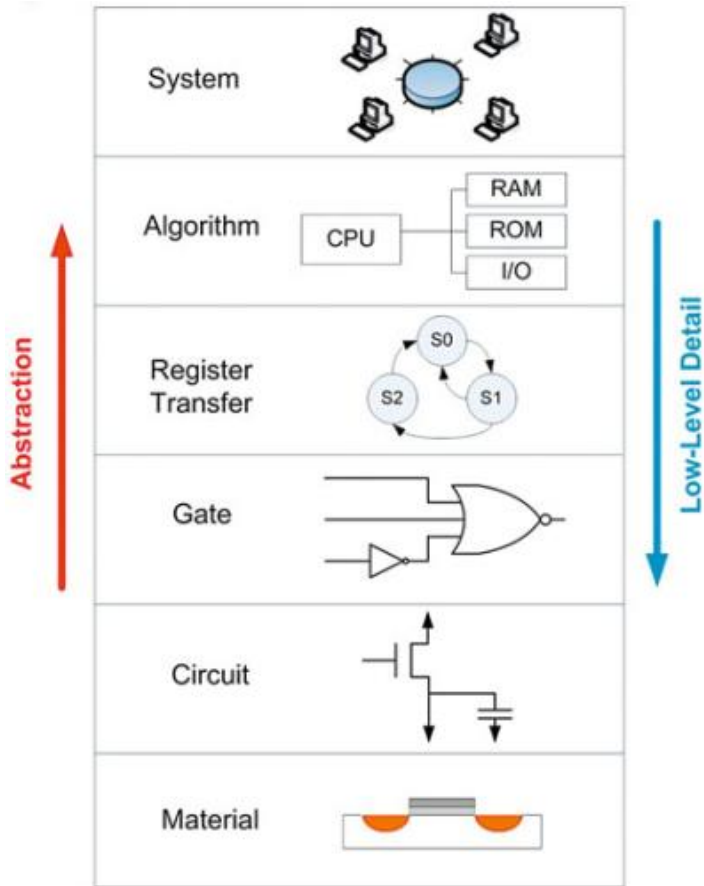
# Design Abstraction



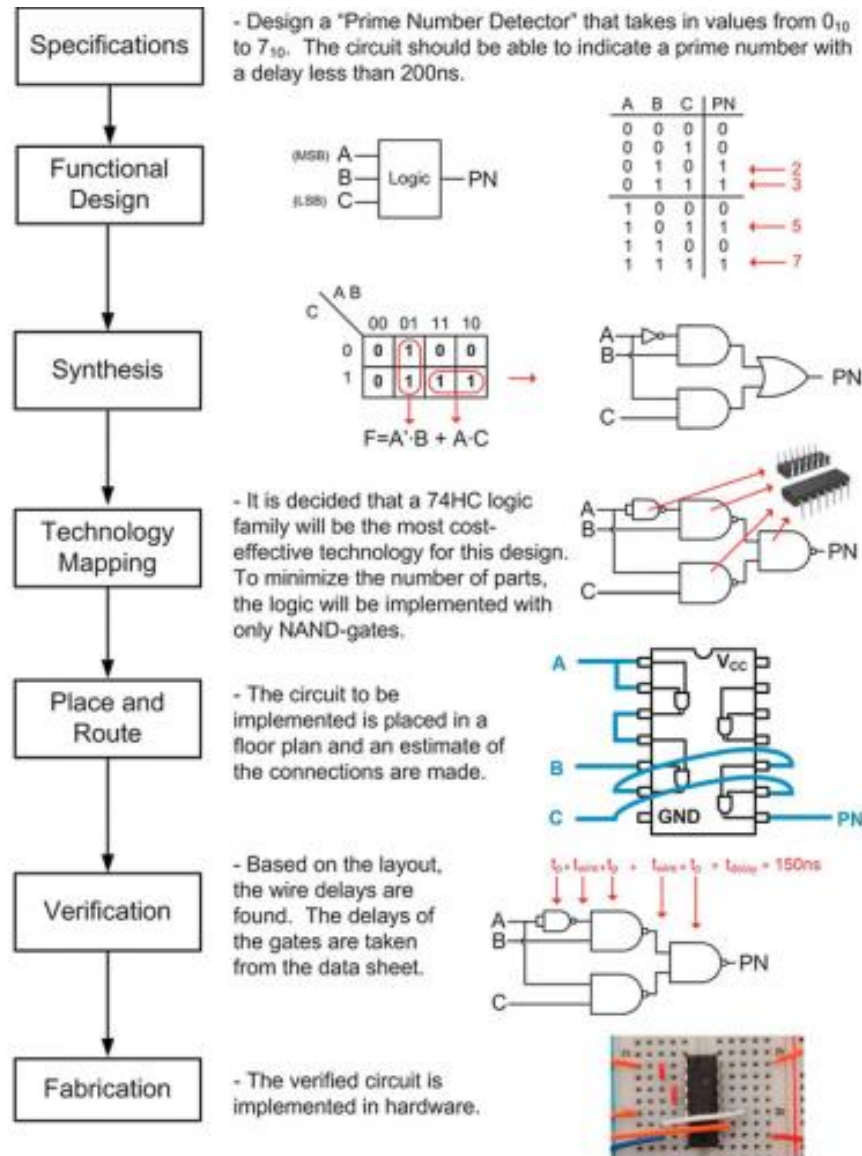
Design Abstraction



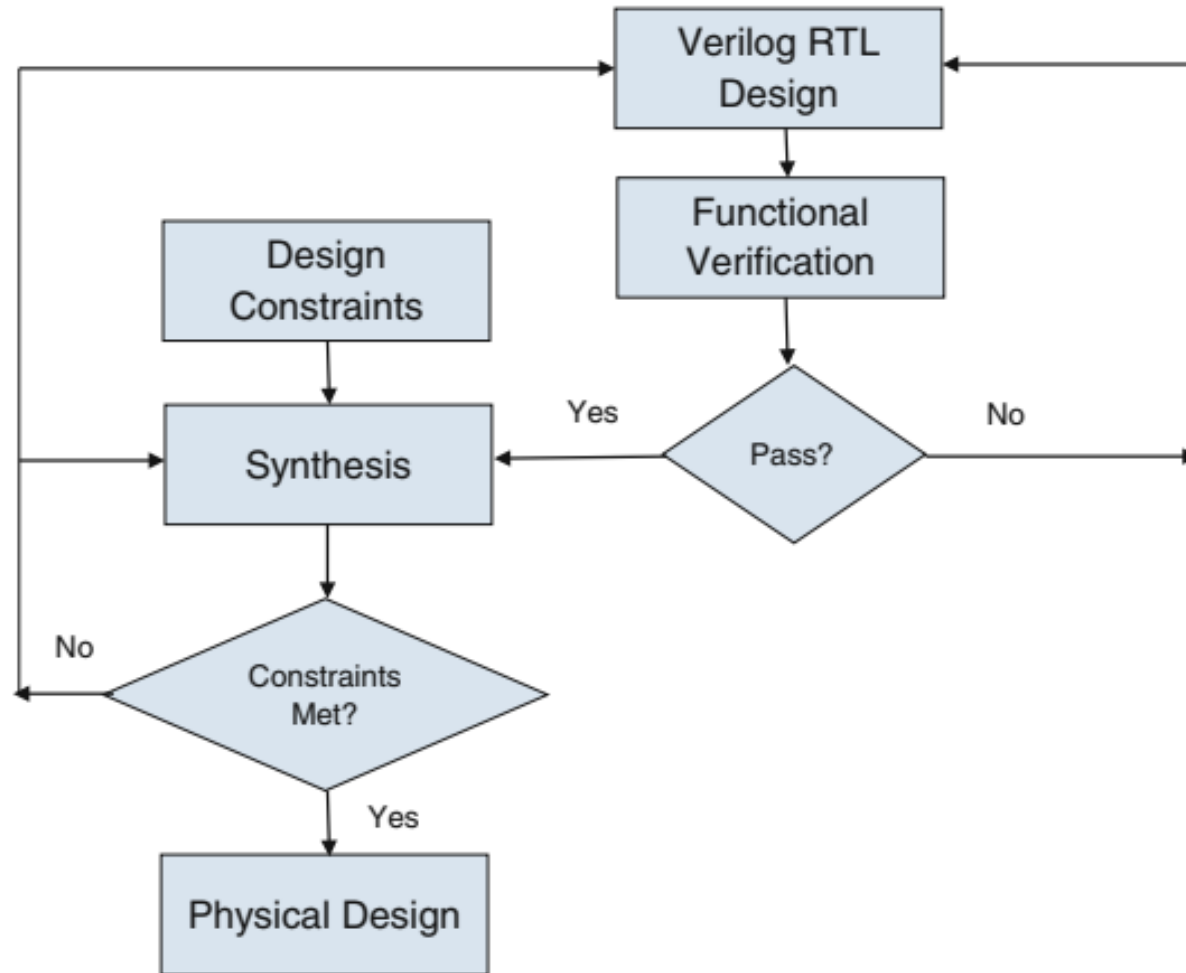
# Design Abstraction



# Classical Digital Design Flow

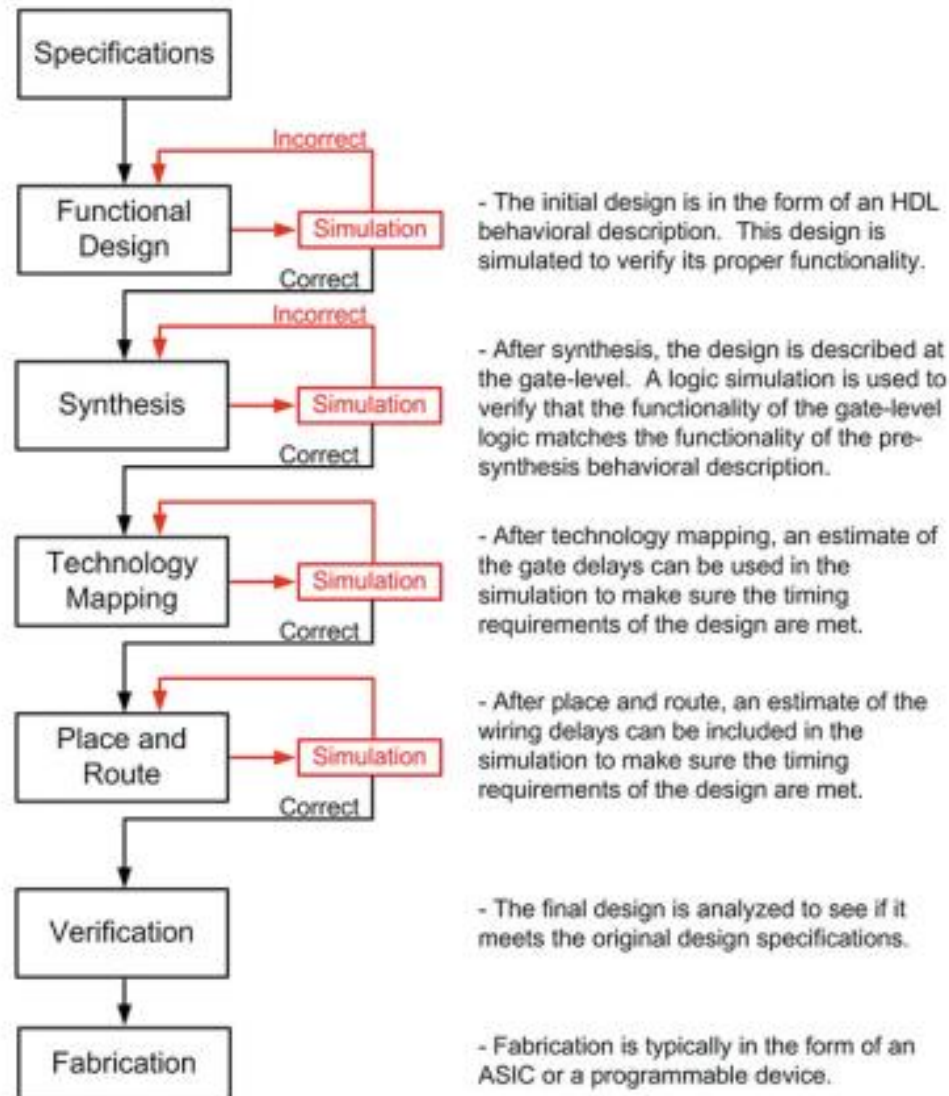


# Modern Digital Design Flow



Simulation and Synthesis Flow

# Modern Digital Design Flow



# Importance of HDLs

- Designs can be described at a very abstract level using HDLs
- RTL descriptions are independent of specific fabrication technologies
- Logic synthesis tools convert designs to any fabrication technology
- New technologies only require re-synthesis, not redesign
- Circuits are optimized for area and timing with new technologies

# Early Functional Verification

- Functional verification can be done early in the design cycle
- Designers work at the RTL level to optimize and modify designs
- Most design bugs are eliminated early, reducing later-stage errors
- Shortens the overall design cycle

# Development and Debugging Efficiency

- Designing with HDLs is similar to computer programming
- Textual descriptions with comments make development and debugging easier
- Provides a concise representation compared to gate-level schematics
- Gate-level schematics become impossible to understand for complex designs.

# Popularity of Verilog

- Verilog HDL is a general-purpose hardware description language, easy to learn and use
- Similar syntax to C programming, making it accessible to C programmers
- Supports mixed levels of abstraction within the same model: switches, gates, RTL, or behavioral code
- One language suffices for stimulus and hierarchical design
- Supported by most popular logic synthesis tools, ensuring wide adoption
- Fabrication vendors provide Verilog HDL libraries for post-logic synthesis simulation



# Comparison of HDLs

| Feature                   | Verilog                                   | VHDL                                      |
|---------------------------|---|---|
| <b>Syntax</b>             | C-like syntax; easy to learn              | Complex syntax; derived from ADA          |
| <b>Suitability</b>        | Best for simulation and hardware modeling | Excellent for large-scale/system modeling |
| <b>Industry Adoption</b>  | Popular in U.S. semiconductor industry    | Popular in Europe and defense industries  |
| <b>Type System</b>        | Less strict                               | Strongly typed, reducing ambiguity        |
| <b>Libraries</b>          | Limited libraries                         | Rich set of libraries                     |
| <b>Simulation Speed</b>   | Faster for large designs                  | Slower compared to Verilog                |
| <b>Abstraction Levels</b> | Strong support for mixed-level modeling   | Focus on high-level abstractions          |

# Hardware Description Languages

## Verilog





|                         |   |
|-------------------------|---|
| <b>Name:</b>            | Verilog HDL   |
| <b>Coach:</b>           | Open Verilog International ( <a href="http://www.ovi.org">www.ovi.org</a> )   |
| <b>Characteristics:</b> | Has very good acceptance in ASIC, particularly lower level designs (register level transfer and below); results in fast simulations, relatively simple, easy in first contacts, especially for C Language users. In long term might have problems with handling system level designs. |
| <b>Fan clubs:</b>       | Mostly in North America and Asia Japan, especially among industrial supporters. Not popular in Europe.  |



## VHDL

|                         |   |
|-------------------------|---|
| <b>Name:</b>            | VHDL  |
| <b>Coach:</b>           | VHDL International ( <a href="http://www.vhdl.org">www.vhdl.org</a> )   |
| <b>Characteristics:</b> | Relatively weaker in lower level designs, but superior in higher- and system level designs; results in slower simulations, but constantly improving; very flexible, but also difficult, complex character; very popular in academia; lots of his skills were taught by Special Forces Officer Ada; many believe that in long term presents better condition and adaptability than its competitor. |
| <b>Fan clubs:</b>       | Especially in Europe, but significant number also in US and Canada. Disliked in Japan, but gaining popularity worldwide.  |



# Hardware Description Languages

## Verilog

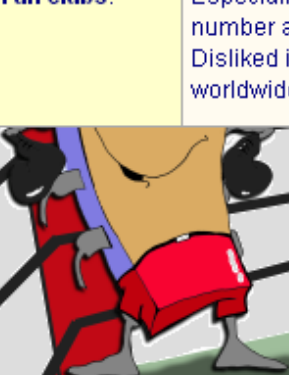



|                         |   |
|-------------------------|---|
| <b>Name:</b>            | Verilog HDL   |
| <b>Coach:</b>           | Open Verilog International ( <a href="http://www.ovi.org">www.ovi.org</a> )   |
| <b>Characteristics:</b> | Has very good acceptance in ASIC, particularly lower level designs (register level transfer and below); results in fast simulations, relatively simple, easy in first contacts, especially for C Language users. In long term might have problems with handling system level designs. |
| <b>Fan clubs:</b>       | Mostly in North America and Asia Japan, especially among industrial supporters. Not popular in Europe.  |



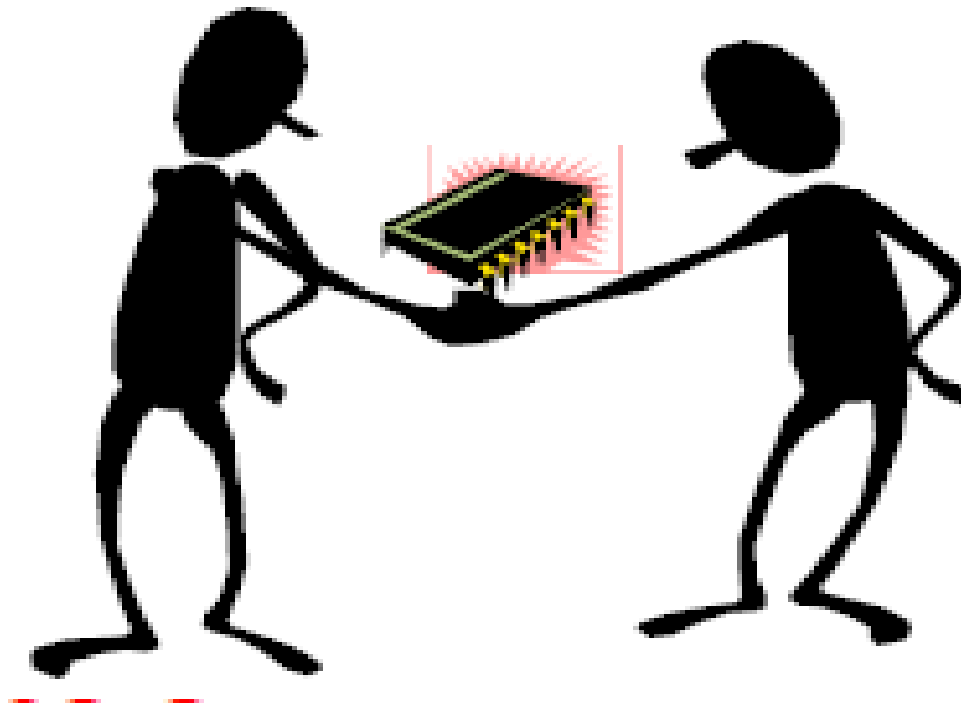
## VHDL

|                         |   |
|-------------------------|---|
| <b>Name:</b>            | VHDL  |
| <b>Coach:</b>           | VHDL International ( <a href="http://www.vhdl.org">www.vhdl.org</a> )   |
| <b>Characteristics:</b> | Relatively weaker in lower level designs, but superior in higher- and system level designs; results in slower simulations, but constantly improving; very flexible, but also difficult, complex character; very popular in academia; lots of his skills were taught by Special Forces Officer Ada; many believe that in long term presents better condition and adaptability than its competitor. |
| <b>Fan clubs:</b>       | Especially in Europe, but significant number also in US and Canada. Disliked in Japan, but gaining popularity worldwide.  |



# Summary

- **Traditional Design Approach** is Error-prone and time-consuming
- Increasing circuit complexity necessitates structured design and efficient methodologies for scalability.
- **Multi-level abstraction** (behavioral, RTL, gate-level, physical) simplifies large-scale system development
- **The modern digital design flow** follows systematic steps (Specification → RTL Design → Synthesis → Verification → Fabrication) to ensure efficiency and correctness in complex designs.
- **HDLs (VHDL, Verilog)** enable simulation and synthesis, making them essential for automation in modern design tools.
- **Verilog is easier for beginners and widely used in industry**, while **VHDL has stronger typing and is suited for complex designs**; selection depends on project requirements



**Thank you !**

**Happy Learning**