



VHDL: Simulation to Synthesis

Chapter 1: Introduction



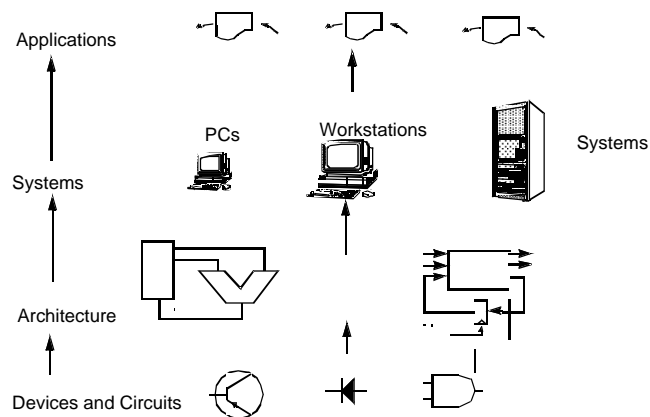
Traditional vs. Hardware Description Languages

- Procedural programming languages provide the *how* or recipes
 - for computation
 - for data manipulation
 - for execution on a specific hardware model
- Hardware description languages *describe* a system
 - many different facets
 - behavior
 - structure
 - functional properties
 - physical properties

Why do we Describe Systems?

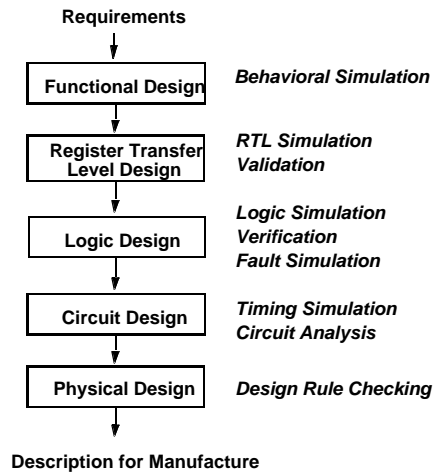
- Design Specification
 - unambiguous definition of components and interfaces in a large design
- Design Simulation
 - verify system/subsystem/chip performance prior to design implementation
- Design Synthesis
 - automated generation of a hardware design

Describing Digital Systems

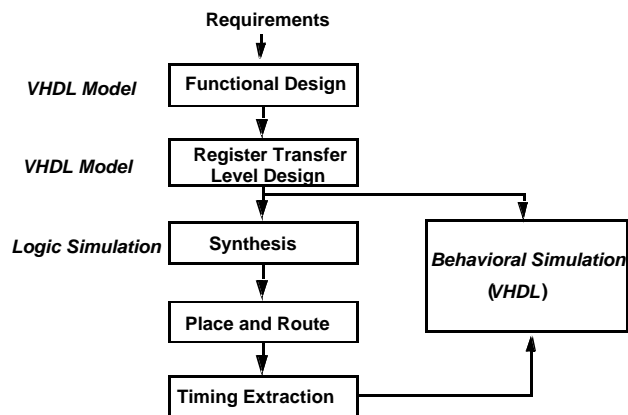


- Systems may be described at multiple levels of abstraction

A Digital System Design Flow

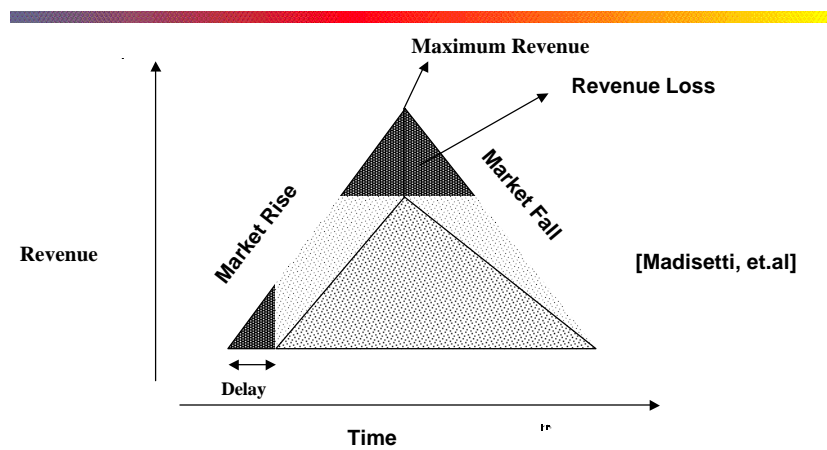


A Synthesis Design Flow



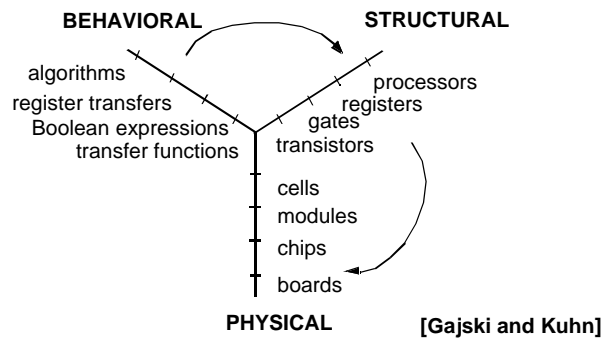
- Automation of design refinement steps
- Feedback for accurate simulation
- Example targets: ASICs, FPGAs

Impact of the Marketplace



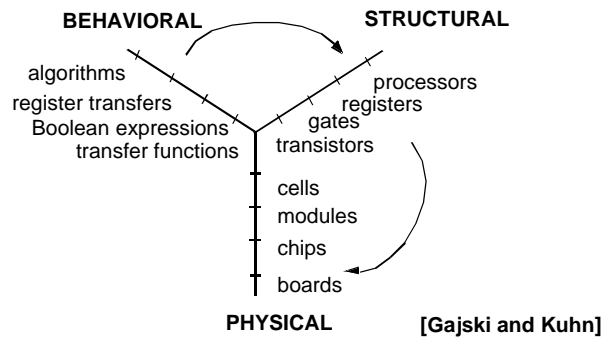
- First 10%-20% of the design cycle can set 70%-80% of system cost
- Reqs can determine 70% of the manufacturing costs
- Pressure is to shorten the design cycle and track reqs

The Role of Hardware Description Languages



- Design is structured around a hierarchy of representations
- HDLs can describe distinct aspects of a design at multiple levels of abstraction

The Role of Hardware Description Languages



- Interoperability: models at multiple levels of abstraction
- Technology independence: portable model
- Design re-use and rapid prototyping

The VHDL Language

V *Very High Speed Integrated Circuit*

H *Hardware*

D *Description*

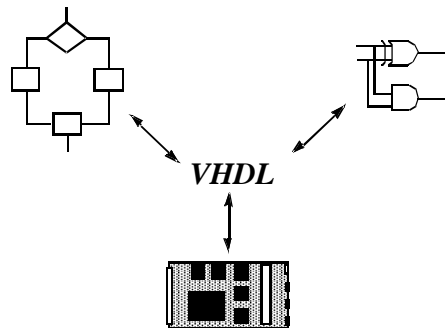
L *Language*

- Interoperability between design tools: standardized portable model of electronic systems
- Technology independent description
- Reuse of components described in VHDL

History of VHDL

- Designed by IBM, Texas Instruments, and Intermetrics as part of the DoD funded VHSIC program
- Standardized by the IEEE in 1987: IEEE 1076-1987
- Enhanced version of the language defined in 1993: IEEE 1076-1993
- Additional standardized packages provide definitions of data types and expressions of timing data
 - IEEE 1164 (data types)
 - IEEE 1076.3 (numeric)
 - IEEE 1076.4 (timing)

Role of VHDL



- System description and documentation
- System simulation
- System synthesis