Digital Design with VERILOG

Highly Successful Verilog course intended for companies or individuals who want to evolve into Verilog development. The course is aimed at learning the Verilog language and is independent of any particular simulation tool. The learner will be able to use any simulation tools that have Verilog capabilities. The main objective to create quality manpower in the field of Verilog-HDl design

Learning Objectives

"Ceda Digital Design with VERILOG training course will help you to learn to understand, design, test, and implement complex digital hardware. You will learn to use Verilog, one of the two modern, industry-standard hardware description languages to develop digital hardware at a much higher level of abstraction than is possible with schematic-based design methods.

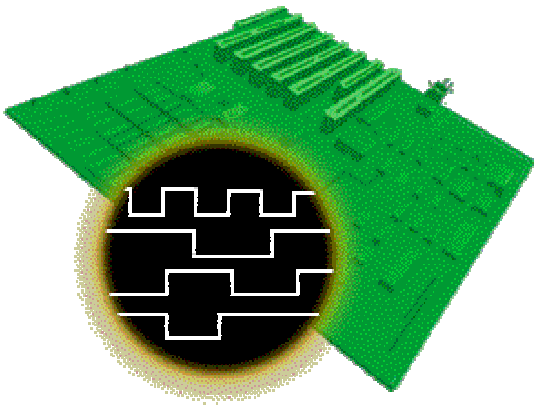
Course Prerequisites:
Knowledge of digital design is useful to fully benefit from this course. However, no hardware description language experience is necessary.

Target Audience:
Engineers who are new in Hardware Description Languages and need to use Verilog as part of their University program/JOB

Course Overview

Introduction to VLSI
- Application of VLSI
- Design Process of VLSI
- Scope of VLSI
- Introduction to VLSI Design flow

Introduction to HDL
- History of HDL
- Need and Scope Of HDL
- Evolution of CAD



Hierarchical Modeling Concepts
- Design Methodology
- Module and Module instances
- Components of Verilog Module
- Design Blocks and Stimulus Block

Basic Concepts
- Types of Modeling
- Data Types in Verilog, System Task
- Logic Values
- Port Definition, Declaration
- Port Connection Rule

Gate Level Modeling
- Gate Types
- Gate Delays
- Vectors in Verilog
- Test Bench Basics
- Writing Verilog Modules
- Gate Level Modeling Examples

Dataflow Modeling
- Type of Operators
- Continuous assignment statement
- Regular assignment delay
- Examples Of Data Flow Modeling

Behavioral Modeling
- Initial and Always Statements
- Procedural Assignment Statements
- Timing Control Statements
- If__ else Statements
- Case statement
- Block Statements
- Loops
- Design of Flip-flops
- Design of Digital circuits using Behavioral Modeling

Digital Design with VERILOG

Professional course with World best Simulator

"Ceda Digital Design with VERILOG training course will help you to learn to understand, design, test, and implement complex digital hardware. You will learn to use Verilog, one of the two modern, industry-standard hardware description languages to develop digital hardware at a much higher level of abstraction than is possible with schematic-based design methods.

TASKS and Function

State Machines

- Moore's Machines
- Mealy Machines
- Examples of Moore and Mealy Machines

Delay Models in Verilog

- Distributed
- Lumped
- Pin to Pin
- Specify block

Minor Projects

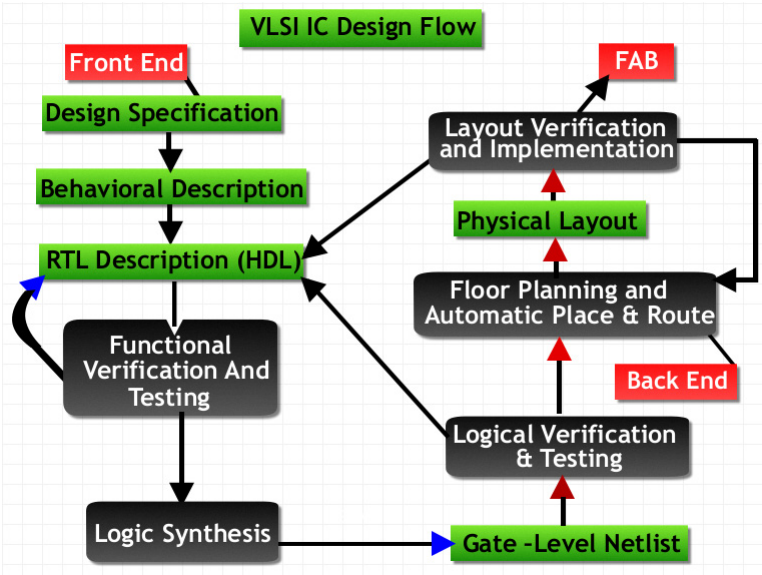
- Traffic Light Controller
- Simple ALU design

Tool Used: ModelSim

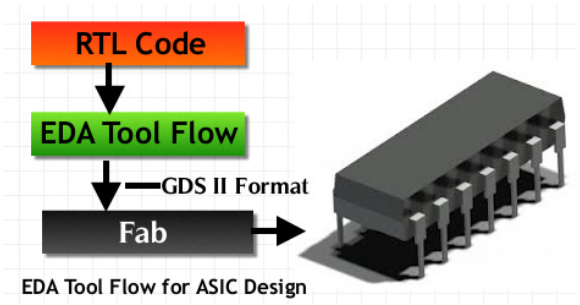
Modelsim is a Simulation tool in which we write the code to check the Desired functionality and if require we debug the code.

ModelSim ***the simulator of choice for both ASIC and FPGA design***

Snapshot of VLSI Flow



ASIC Design Flow



EDA Tool Flow for ASIC Design

Course Materials:

Course attendees will receive a color fully indexed and cross-referenced course manual, and a certificate of attendance.
To book a course, please contact us

VERILOG & implementation in FPGA

Highly Successful Verilog course intended for companies or individuals who want to evolve into Verilog development. The course is aimed at learning the Verilog language and is independent of any particular simulation tool. The learner will be able to use any simulation tools that have Verilog capabilities. The main objective to create quality manpower in the field of Verilog-HDl design

Learning Objectives

"Ceda VERILOG & Implementation in FPGA training course will help you to learn to understand, design, test, and implement complex digital hardware. You will learn to use Verilog, to develop digital hardware at a much higher level of abstraction than is possible with schematic-based design methods AND implementation of Verilog in a FPGA

Course Prerequisites:
Knowledge of digital design is useful to fully benefit from this course. However, no hardware description language experience is necessary.

Target Audience:
Engineers who are new in Hardware Description Languages and need to use Verilog as part of their University program/JOB

Course Overview

Introduction to VLSI
- Application of VLSI
- Design Process of VLSI
- Scope of VLSI
- Introduction to VLSI Design flow

Introduction to HDL
- History of HDL
- Need and Scope Of HDL
- Evolution of CAD

Hierarchical Modeling Concepts
- Design Methodology
- Module and Module instances
- Components of Verilog Module
- Design Blocks and Stimulus Block

Basic Concepts
- Types of Modeling
- Data Types in Verilog, System Task
- Logic Values
- Port Definition, Declaration
- Port Connection Rule

Gate Level Modeling
- Gate Types
- Gate Delays
- Vectors in Verilog
- Test Bench Basics
- Writing Verilog Modules
- Gate Level Modeling Examples

Dataflow Modeling
- Type of Operators
- Continuous assignment statement
- Regular assignment delay
- Examples Of Data Flow Modeling

Behavioral Modeling
- Initial and Always Statements
- Procedural Assignment Statements
- Timing Control Statements
- If__ else Statements
- Case statement
- Block Statements
- Loops
- Design of Flip-flops
- Design of Digital circuits using Behavioral Modeling

TASKS and Function

VERILOG & implementation in FPGA

State Machines
- Moore's Machines
- Mealy Machines
- Examples of Moore and Mealy Machines

Delay Models in Verilog
- Distributed
- Lumped
- Pin to Pin
- Specify block

Minor Projects
- Traffic Light Controller
- Simple ALU design

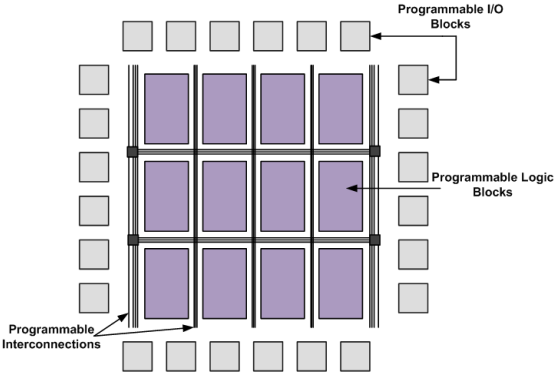
Project Implementation in FPGA
- Traffic Light Controller
- Simple ALU design

Field Programmable Gate Array (FPGA)

Field Programmable Gate Arrays popularly known as FPGAs is an alternative for implementation of digital logic in systems. They are prefabricated silicon chips that can be programmed electrically to implement any digital design.

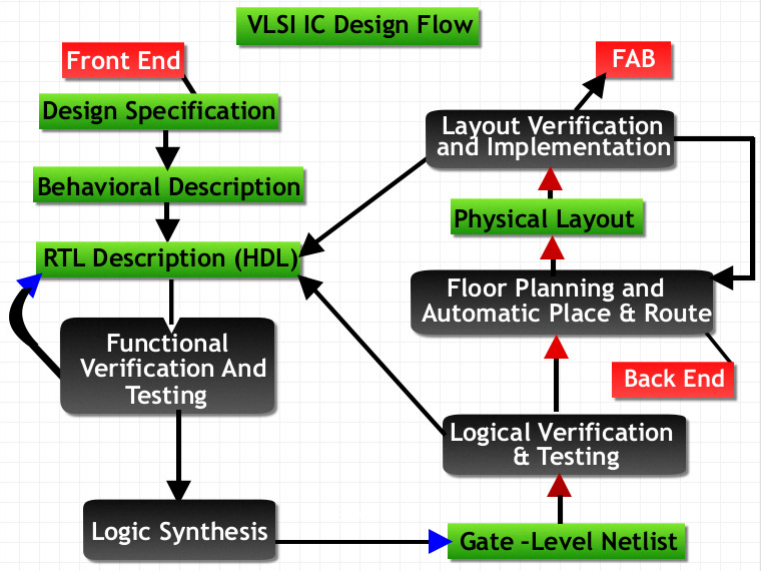
FPGA now can contain approximately 330,000 logic blocks and around 1100 inputs and outputs. The basic architecture of FPGA consists of three major components: programmable logic blocks which implements the logic functions, programmable routing (interconnects) to implement these functions and I/O blocks to make off-chip connections.

An illustration of typical FPGA architecture is shown in figure

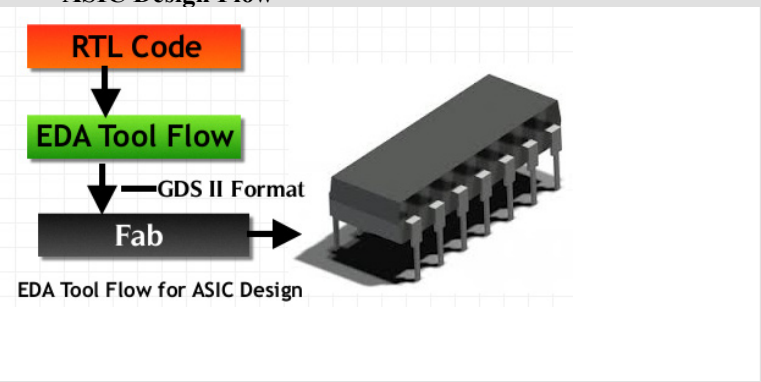


Internal Architecture of FPGA

Snapshot of VLSI Flow



ASIC Design Flow



Tool Used:

ModelSim Modelsim is a Simulation tool in which we write the code to check the Desired functionality and if require we debug the code.

Xilinx ISE Xilinx ISE is an Integrated Software Environment use to check HDL error and little bit simulation (Not as good as compare to Modelsim) plus it will synthesize the code and generate the required files for the Xilinx Device. **Therefore**, ISE does Simulation+Synthesis+FPGa File generation for Hardware configuration.

CEDA www.ceda.in/course.html