Faysal Aziz

Beginning Verilog

Digital-design modules and introductory simulation

FA-0587 · 2 days

Software Development

Write and simulate simple Verilog modules, organise hierarchical designs and create introductory testbenches.

Introduction

This two-day beginner course introduces Verilog as a language for describing digital hardware. Participants move from modules, ports and basic logic to hierarchical design, parameterisation and simple behavioural modelling.

Guided exercises use a prepared simulator to examine module behaviour and testbench results. The course is an introduction to HDL coding and simulation, not complete ASIC/FPGA implementation training. Illustrative applications connect the language concepts to digital-system design without relying on claims about individual trainer achievements.

Learning outcomes

The course teaches participants to:

- Explain the purpose of Verilog and distinguish an HDL from a general-purpose application language.
- Use basic module, port and data-type constructs.
- Write and simulate small logic examples.
- Create a basic testbench and inspect simulation results.
- Use module instantiation, parameters and introductory procedural blocks in a small hierarchical design.

Prerequisites

- General familiarity with computer hardware and basic digital logic.

- No previous Verilog experience is required.
- A laptop with a suitable prepared Verilog simulation environment.
- Interest in learning digital-design coding and simulation.

Training outline

1. Day 1 — Syntax, Logic and First Simulations

Morning — Verilog and Basic Constructs

- What Verilog describes and how it is used in digital design.
- An introductory comparison with other hardware-description languages.
- Introduction to modules, ports, and I/O types.
- Basic Verilog data types and how they affect a small design.

Afternoon — Simple Designs and Testbenches

- Writing your first Verilog module.
- Exploring basic logic gates and their implementation in Verilog.
- What is a testbench and why is it important?
- Creating and running a basic testbench.

2. Day 2 — Hierarchy and Behavioural Modelling

Morning — Modules and Simulation Review

- Hierarchical design and module instantiation.
- Module parameters and configurable examples.
- Simulate a slightly larger hierarchical example.
- Analyzing simulation outputs.

Afternoon — Procedural Blocks and Applications

- Introduction to procedural blocks.
- Use a simple always block and inspect its behaviour in simulation.
- Review clearly illustrative uses of Verilog in digital systems.

- Discuss coding and simulation assumptions and next learning steps.
- Questions and review of the guided examples.

© 2026 Faysal Aziz · Kuala Lumpur