Course Description

This course provides a thorough introduction to the VHDL language. The emphasis is on:

- Writing efficient hardware designs
- Performing high-level HDL simulations
- Employing structural, register transfer level (RTL), and behavioral coding styles
- Targeting Xilinx devices specifically and FPGA devices in general
- Utilizing best coding practices

Level – FPGA 1
Course Duration – 3 days
Course Part Number – LANG-VHDL
Who Should Attend? – Engineers who want to use VHDL effectively for modeling, design, and synthesis of digital designs

Prerequisites
- Basic digital design knowledge

Software Tools
- Vivado® Design or System Edition 2019.1

Hardware
- Architecture: N/A*
- Demo board: Kintex® UltraScale™ FPGA KCU105*

* This course does not focus on any particular architecture. Check with your local Authorized Training Provider for the specifics of the in-class lab board or other customizations.

After completing this comprehensive training, you will have the necessary skills to:

- Implement the VHDL portion of coding for synthesis
- Identify the differences between behavioral and structural coding styles
- Distinguish coding for synthesis versus coding for simulation
- Use scalar and composite data types to represent information
- Use concurrent and sequential control structure to regulate information flow
- Implement common VHDL constructs (finite state machines [FSMs], RAM/ROM data structures)
- Simulate a basic VHDL design
- Write a VHDL testbench and identify simulation-only constructs
- Identify and implement coding best practices
- Optimize VHDL code to target specific silicon resources within the Xilinx FPGA
- Create and manage designs within the Vivado Design Suite environment

Course Outline

Day 1

- Introduction to VHDL
  Discusses the history of the VHDL language and provides an overview of the different features of VHDL. (Lecture)

- VHDL Design Units
  Provides an overview of typical VHDL code, covering design units such as libraries, packages, entities, architectures, and configuration. (Lecture, Lab)

Day 2

- Conditional Statements in VHDL: if/else, case
  Describes conditional statements such as if/else and case statements. (Lecture, Lab)

- Sequential Looping Statements
  Introduces the concept of looping in both the simulation and synthesis environments. (Lecture, Lab)

- Delays in VHDL: Wait Statements
  Covers the wait statement and how it controls the execution of the process statement. (Lecture)

- Introduction to the VHDL Testbench
  Introduces the concept of the VHDL testbench to verify the functionality of a design. (Lecture, Lab)

- VHDL Assert Statements
  Describes the concept of VHDL assertions. (Lecture)

- VHDL Attributes
  Describes attributes, both predefined and user defined. (Lecture)

- VHDL Subprograms
  Covers the use of subprograms in verification and RTL code to model functional blocks. (Lecture)

- VHDL Functions
  Describes functions, which are integral to reusable and maintainable code. (Lecture, Lab)

- VHDL Procedures
  Describes procedures, common constructs that are also important for reusing and maintaining code. (Lecture)

Day 3

- VHDL Libraries and Packages
  Demonstrates how libraries and packages are declared and used. (Lecture, Lab)

- Interacting with the Simulation
  Describes how to interact with a simulation via text I/O. (Lecture)

- Finite State Machine Overview
  Provides an overview of finite state machines, one of the more commonly used circuits. (Lecture)
Designing with VHDL
FPGA 1

LANG-VHDL (v1.0) Course Specification

- **Mealy Finite State Machine**
  Describes how to implement a Mealy state machine in which the output is dependent on both the current state and the inputs. (Lecture)

- **Moore Finite State Machine**
  Demonstrates how to implement a Moore state machine in which the output is dependent on the current state only. (Lecture, Lab)

- **FSM Coding Guidelines**
  Describes the guidelines and recommendations for using one or more procedural blocks when coding a finite state machine. (Lecture)

- **Vivado Simulator and Race Conditions in VHDL**
  Introduces the Vivado simulator simulation environment. Race conditions are also discussed. (Lecture)

- **Writing a Good Testbench**
  Explores how time-agnostic, self-checking testbenches can be written and applied. (Lecture, Lab)

- **Targeting Xilinx FPGAs**
  Focuses on Xilinx-specific implementation and chip-level optimization. (Lecture, Lab)

**Register Today**
Visit the Xilinx Customer Training Center to view schedules and register online.