

VHDL Training Overview

SynthWorks offers a wide range of VHDL classes that are appropriate for either new or experienced engineers. Class topics include the VHDL language, VHDL coding for synthesis, and VHDL coding for testbenches and verification. Our classes focus on VHDL syntax and coding techniques that engineers need to effectively design and verify their next FPGA and/or ASIC.

Our instructors are VHDL experts who have solved difficult design and test coding problems and can answer your questions in detail. We are vendor independent and teach coding styles that are portable between different EDA tools. We are part of the VHDL standards team and are helping make sure that VHDL meets your current and future needs.

All of our offerings are modular and flexible to allow us to train your engineers your way. Below you will find our core course offering as well as our common combinations and customizations of our core courses.

Core Course Offerings

Comprehensive VHDL Introduction - 4 days

Introductory Level, 50% Lecture, 50% Labs

An in-depth introduction to VHDL and its application to design and verification of digital hardware (FPGAs and ASICs). Students will gain a strong foundation in VHDL RTL and testbench coding techniques. Lectures contain numerous examples that show both syntax and coding style guidelines. Labs give students hands-on experience writing RTL code, writing testbenches, running your simulator and synthesis tools, and programming our FPGA based lab board. Recommended as a first course for design and verification engineers who need a solid foundation in VHDL.

Learn VHDL from a designer's perspective with SynthWorks

www.SynthWorks.com

Core Course Offerings ... continued ...

VHDL Testbenches and Verification - 4 days

Advanced Level, 50% Lecture, 50% Labs

An in-depth study of advanced VHDL coding styles and methodologies to verify digital hardware (FPGAs and ASICs). This course starts with simple testbenches and progressively increases the level of abstraction. Along the way students learn about subprogram usage, TEXTIO, modeling issues, transaction-based tests, data structures (linked-lists, scoreboards, memories), algorithmic and random test generation, protocol checking, result checking, and handshaking methods. The final result is a transaction-based, system-level, self-checking test environment. Our test methodology shows how to pre-use the system-level test environment for subblock tests. Labs track with lecture giving students the opportunity to apply what they learn.

As new verification features are integrated into VHDL as part of the VHDL-2006 language revision effort, we expect this course to become 5 days in length.

VHDL Coding Styles for Synthesis - 4 days

Advanced Level, 50% Lecture, 50% Labs

An in-depth study of VHDL RTL (FPGA and ASIC) coding styles, methodologies, design techniques, problem solving techniques, and advanced language constructs to produce better, faster, and smaller logic. Class topics focus on mapping digital hardware structures to vendor independent VHDL code. Detailed do's and don'ts of synthesis coding styles are discussed. Lecture and laboratory materials illustrate the optimization differences achieved by different VHDL coding styles. Students will learn proven coding practices that result in smaller and faster designs. The numerous examples in this course make it suitable for a student with limited VHDL. The application focus of this course results in the student being ready for VHDL based ASIC or FPGA design.

Common Combinations and Customizations

Our course offerings are modular and flexible to allow us to train your engineers your way. If you don't find what you need here, we can customize a class for you.

VHDL Introduction for Verilog Designers - 1 to 4 days

Introductory Level

Comprehensive VHDL Introduction customized for students with Verilog experience.

Intermediate VHDL Coding Styles for Synthesis - 2 days

Intermediate Level, 50% Lecture, 50% Labs

First half of VHDL Coding for Synthesis that focuses on coding styles, methodologies, and design techniques used to efficiently synthesize digital hardware (ASICs and FPGAs).

Advanced VHDL Coding Styles for Synthesis - 2 days

Advanced Level, 50% Lecture, 50% Labs

Second half of VHDL Coding for Synthesis that focuses on coding techniques, advanced language features (subprograms, generics, generate, packages), issue identification and problem solving.

Intermediate VHDL for Synthesis and Verification - 5 days

Intermediate Level, 50% Lecture, 50% Labs

Covers both testbench and intermediate RTL coding topics. Covers all topics from Intermediate VHDL Coding Styles for Synthesis and the VHDL Testbenches and Verification class.

Advanced VHDL for Synthesis and Verification - 5 days

Intermediate Level, 50% Lecture, 50% Labs

Covers both testbench and advanced RTL coding topics. Covers all topics from Advanced VHDL Coding Styles for Synthesis and the VHDL Testbenches and Verification class.

Quick VHDL Introduction - 2 days

Introductory Level, 60% Lecture, 40% Labs

Comprehensive VHDL Introduction shortened for support engineers and managers who only need to understand the basics of using VHDL in a design environment. Only recommended for design and verification engineers when they plan on immediately following it with additional training.

Comments about SynthWorks' Class Materials

Emphasized key concepts, labs drove home ideas.

Excellent insight into unseen / unexpected gotcha's

The slides were the best I have ever seen. Very effective use of powerpoint.

The lab problems were ideal - a happy medium between real-world application and trivial application. They were a great way to reinforce the material.

Enjoyed having a designer teach the course.

Public VHDL Training Sessions

The latest information on our public class sessions is posted at:

http://www.synthworks.com/public_vhdl_courses.htm

More Information

To schedule a class, for more information, or to get on our email class announcement list, contact Jim Lewis at (800) 505-8435, (503) 590-4787 or jim@SynthWorks.com. Complete class information is available at: <http://www.SynthWorks.com>.