HLS01: High-Level Synthesis for AMD devices with the Vitis HLS Tool

HLS01: Síntesis de alto nivel para FPGAs de AMD con Vitis-HLS

Language: The classes are in Spanish, but working material is in English (available also in English at In-Company).

Who Should Attend? Hardware, firmware, and system design engineers who are interested in use high-level synthesis and accelerate the hardware development process.

Duration: 16 h (2 days, 8 h/day).

Prerequisites: Notions of Digital and AMD-Xilinx FPGA design. C, C++ or system-C.

Introduction: Digital design is moving from RTL design levels using HDLs to more productive tool that uses higher level of abstractions. The course introduces the Vitis High-Level Synthesis (HLS) tool. This course covers synthesis strategies, features, improving throughput, area, interface creation, latency, testbench coding, and coding tips. Utilize the Vitis HLS tool to optimize code for high-speed performance in an embedded environment and download for in-circuit validation.

The focus of this course is on: Converting C/C++ designs into RTL implementations; Learning the Vitis HLS tool flow; Creating I/O interfaces for designs by using the Vitis HLS tool; Applying different optimization techniques; Improving throughput, area, latency, and logic by using different HLS pragmas/directives; Exporting IP that can be used with the Vivado IP catalog; Downloading for in-circuit validation.

Skills Gained: After completing this training, you will know how to:

- Enhance productivity in HW development by using the Vitis HLS tool
- Know the high-level synthesis flow and how to optimize it.
- Use the Vitis HLS tool for a first project
- Identify the importance of the testbench and use in real case
- Use directives to improve performance and area and select RTL interfaces
- Identify common coding pitfalls as well as methods for improving code for RTL/HW.
- Perform system-level integration of IP generated by the Vitis HLS tool.

Material: Each student will have a computer with the development tools (Vivado, Vitis and Vitis-HLS 2022.x), documentation, repository whit exercises (and solutions) and a FPGA development board for exercises that require it.

Course Outline: This course was updated for the new tool version including RTL as blackbox, system integration with Zynq US+, new HLS task library and a module of Abstract Parallel Programming Model for HLS.

- Introduction to High-Level Synthesis: Overview of the Vitis HLS tool flow, and the verification advantage.
- Vitis HLS Tool Flow: Explores the basics of high-level synthesis and the Vitis HLS tool.
- Abstract Parallel Programming Model for HLS: Describes the structuring of a design at a high level using an abstract parallel programming model.
- Design Exploration with Directives: Explores different optimization techniques that can improve the design performance.
- Vitis HLS Tool Command Line Interface: Describes the Vitis HLS tool flow in command prompt mode.
- Introduction to Vitis HLS Design Methodology: Introduces the methodology

guidelines covered in this course and the HLS Design Methodology steps.

- Introduction to I/O Interfaces: Explains interfaces such as the block-level and port-level protocols abstracted by the Vitis HLS tool from the C design.
- Block-Level Protocols: Explains the different types of block-level protocols abstracted by the Vitis HLS tool.
- Port-Level I/O Protocols: Describes the port-level interface protocols abstracted by the Vitis-HLS tool from the C design.
- AXI Adapter Interface Protocols: Explains the different AXI interfaces (such as AXI4-Master, AXI4-Lite (Slave), and AXI4-Stream) supported by the Vitis HLS tool.
- Port-Level I/O Protocols - Memory Interfaces: Describes the memory interface port-level protocols (such as block RAM and FIFO) abstracted by the Vitis HLS tool from the C design.
- Pipeline for Performance - PIPELINE: Describes the PIPELINE directive for improving the throughput of a design.
- Pipeline for Performance - DATAFLOW: Describes the directive for improving the throughput of a design by pipelining the functions to execute as soon as possible.
- Optimizing for Throughput: Identify the performance limitations caused by arrays in your design. You will also explore optimization techniques to handle arrays for improving performance.
- Optimizing for Latency - Default Behavior and Reduction: Describes the default behavior of the Vitis HLS and how to optimize the C design to improve latency.
- Optimizing for Area and Logic: Describes different methods for improving resource utilization and explains how some of the directives have impact on the area utilization.
- Migrating to the Vitis HLS Tool: Reviews key considerations when moving from the Vivado HLS tool to the Vitis HLS tool.
- HLS Design Flow – System Integration: Describes the traditional RTL flow versus the Vitis HLS tooldesign flow.
- Vitis HLS Tool C++ Libraries: Arbitrary Precision
- Describes Vitis HLS tool support for the C/C++ languages as well as arbitrary precision data types.
- Hardware Modeling: Describes hardware modeling with streaming data types and shiftregister implementation using the ap_shift_reg class.
- Using Pointers in the Vitis HLS Tool: Explains the use of pointers in the design and workarounds for some of the limitations.

Related Courses:

Today new tools and methodologies of Xilinx FPGA design relates with High Level Synthesis (HLS). It is in the hart of Vivado-HLx and Vitis.

Other Xilinx Technologies courses:

Please visit our web site.

Dates, location and registration:

Visit www.electratraining.org

Prices and Discounts:

HLS01: 990 €
HLS01 coming from VIV-x or SoC-x: 840€ (-15%)
HLS01 coming from other Xilinx course: 890 € (-10%)

For more than one engineer from same company / institution additional discounts.