AMD
together we advance_

Designing with the IP Integrator Tool

FPGA-IPI (v1.0)

Course Specification

Course Description

Explore the Vivado™ IP integrator tool and its features to gain the expertise needed to develop, implement, and debug different IP integrator block designs using the Vivado Design Suite.

This course focuses on:

- Creating an IP integrator block design using the Vivado Design Suite
- Creating your own custom IP via the IP packaging flow
- Using the block design container (BDC) and module referencing features of the IP integrator
- Using the IP integrator to add and configure the Versal™ device CIPS block and then to export the generated programmable device image (PDI)
- Configuring the AXI network on chip (NoC) to access DDR memory controllers in Versal devices

What's New for 2024.2

- All labs have been updated to the latest software versions

Level – FPGA 1

Course Details

- 2 days ILT or 3 sessions

Course Part Number – FPGA-IPI

Who Should Attend? – Software and hardware developers, system architects, and anyone who wants to learn about the Vivado Design Suite IP integrator tool

Prerequisites

- Basic knowledge of AMD FPGAs and adaptive SoCs

Software Tools

- Vivado Design Suite 2024.2
- Vitis Unified IDE 2024.2

Hardware

- Architecture: UltraScale™ family and Versal adaptive SoCs
- Demo board: Zynq™ UltraScale+™ ZCU104 board

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

- Describe the Vivado tool flow for RTL-based and IP-based design flows
- Create a Vivado IP integrator block design using the Vivado Design Suite
- Describe the block design container feature in the IP integrator
- Package custom IP and add it to the IP catalog repository or manage it from a remote location
- Add an RTL module or a block design (BD) into a block design by using RTL module referencing
- Add and configure the Versal device CIPS block and export the generated hardware
- Configure the AXI NoC to access DDR memory controllers in Versal devices
- Debug a design by adding debug cores using the IP integrator
- Use a revision control system in the Vivado Design Suite flows

Course Outline

Day 1

Vivado IP Catalog

- **Vivado IP Flow**

  Demonstrates how to customize IP, instantiate IP, and verify the hierarchy of your design IP. {Lecture, Demo}

Using the Vivado IP Integrator

- **Getting Started with Vivado IP Integrator**

  Introduces the Vivado IP integrator tool and its features. Also reviews creating and working with block designs. {Lecture, Demo, Lab}
- **Designing IP Subsystems Using Vivado IP Integrator**

  Illustrates designing with processor-based subsystems and working with custom RTL code. Also explains how to create Vitis platforms using Vivado IP integrator. {Lecture, Lab}
- **Block Design Containers in the Vivado IP Integrator**

  Describes the block design container (BDC) feature and shows how to create a BDC in the IP integrator. {Lecture, Lab}
- **Creating and Packaging Custom IP**

  Covers creating your own IP and package and including it in the Vivado IP catalog. {Lecture, Lab}
- **Module Referencing in IP Integrator**

  Shows how to quickly add an RTL module or a block design (BD) into a block design by using RTL module referencing. {Lab}

Day 2

- **Versal Adaptive SoC: Hardware Platform Development Using the Vivado IP Integrator**

  Describes the different Versal device design flows and covers the platform creation process using the Vivado IP integrator. {Lecture, Lab}
- **Versal Adaptive SoC: NoC Introduction and Concepts**

  Reviews the basic vocabulary and high-level operations of the NoC. {Lecture, Labs}

Debugging

- **Debug Flow in an IP Integrator Block Design**

  Shows how to insert the debug cores into IP integrator block designs. {Lecture, Lab}

Version Control Systems

- **Revision Control Systems in the Vivado Design Suite**

  Investigates using version control systems with the Vivado design flows. {Lecture, Lab}

Vivado IP Catalog

- **Managing IP in Remote Locations**

  Covers storing IP and related files that are remote to the current working project directory. {Lecture}