Designing FPGAs Using the Vivado Design Suite 4

FPGA-VDES4 (v1.0)

Course Specification

Course Description

Learn how to use the advanced aspects of the Vivado® Design Suite and AMD Xilinx hardware.

The focus is on:

- Applying techniques to reduce delay and to improve clock skew and clock uncertainty
- Utilizing floorplanning techniques
- Employing advanced implementation options
- Utilizing AMD Xilinx security features
- Identifying advanced FPGA configurations
- Debugging a design at the device startup phase
- Using Tcl scripting in non-project batch flows

This is the final course in the *Designing FPGAs Using the Vivado Design Suite* series.

What's New for 2022.2

- All labs have been updated to the latest software versions

Level – FPGA 4

Course Details

- 2 days live instructor led training (online or in person)
  - 33 lectures
  - 12 labs
  - 4 demos

Price – $1,600 or 16 Xilinx Training Credits

Course Part Number – FPGA-VDES4

Who Should Attend? – Engineers who seek advanced training in using AMD Xilinx tools to improve FPGA performance and utilization while also increasing productivity

Prerequisites

- Intermediate HDL knowledge (Verilog or VHDL)
- Sound digital design background
- Designing FPGAs Using the Vivado Design Suite 1
- Designing FPGAs Using the Vivado Design Suite 2
- Designing FPGAs Using the Vivado Design Suite 3

Software Tools

- Vivado Design Suite 2022.2

Hardware

- Architecture: UltraScale™ FPGAs and Versal® ACAPs*
- Demo board: Zynq® UltraScale+™ ZCU104 board

* This course focuses on the UltraScale and Versal architectures. Check with Morgan Advanced Programmable Systems, Inc. for the specifics of the in-class lab board or other customizations.

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

- Analyze a timing report to identify how to center the clock in the data eye
- Apply appropriate techniques to reduce logic and net delay and to improve clock skew and clock uncertainty
- Implement Intelligent Design Runs (IDR) to automate analysis and timing closure for complex designs
- Utilize floorplanning techniques to improve design performance
- Employ advanced implementation options, such as incremental compile flow, physical optimization techniques, and re-entrant mode as last mile strategies
- Utilize security features, bitstream encryption, and authentication using AES for design and IP security
- Identify advanced FPGA configurations, such as daisy chains and gangs, for configuring multiple FPGAs in a design
- Debug a design at the device startup phase to debug issues related to startup events, such as MMCM lock and design coming out of reset
- Use Tcl scripting in non-project batch flows to synthesize, implement, and generate custom timing reports

Course Outline

Day 1

UltraFast Design Methodology (UFDM)

- **UltraFast Design Methodology: Design Closure**
  Introduces the UltraFast methodology guidelines on design closure. {Lecture}

Vivado Tool Flow

- **Hierarchical Design**
  Provides an overview of the hierarchical design flows in the Vivado Design Suite. {Lecture}
- **Incremental Compile Flow**
  Demonstrates how to utilize the incremental compile flow when making last-minute RTL changes. {Lecture, Lab}
- **Vivado Design Suite ECO Flow**
  Illustrates using the ECO flow for making changes to a previously implemented design and applying the changes to the original design. {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, Lab}

Timing – Advanced

- **Timing Closure Using Physical Optimization Techniques**
  Describes physical optimization techniques for timing closure. {Lecture, Lab}
- **Case Analysis**
  Highlights how to analyze timing when using multiplexed clocks in a design. {Lecture}
- **Reducing Logic Delay**
  Describes how to optimize regular fabric paths and paths with dedicated blocks and macro primitives. {Lecture}
- **Reducing Net Delay**
  Reviews different techniques to reduce congestion and net delays. {Lecture}
- **Improving Clock Skew**
  Describes how to apply various techniques to improve clock skew. {Lecture}
- **Improving Clock Uncertainty**
  Reviews various flows for improving clock uncertainty, including using parallel BUFGCE_DIV clock buffers, changing MMCM or PLL settings, and limiting synchronous clock domain crossing (CDC) paths. {Lecture, Lab}

Design Runs

- **Intelligent Design Runs (IDR)**
  Introduces Intelligent Design Runs (IDR), which are special types of implementation runs that use a complex flow to attempt to close timing. {Lecture, Lab}

Day 2

Floorplanning

- **Introduction to Floorplanning**
  Introduces floorplanning and how to use Pblocks while floorplanning. {Lecture}
- **Design Analysis and Floorplanning**
  Highlights the pre- and post-implementation design analysis features of the Vivado IDE. {Lecture, Lab}
- **Congestion**
  Identifies congestion and addresses congestion issues. {Lecture}

Power

- **Power Management Techniques**
  Describes the techniques used for low power design. {Lecture}
- **Versal ACAP: Power Management**
  Discusses power domains and how they can be controlled along with basic techniques used to lower overall power consumption. {Lecture}

Configuration

- **Daisy Chains and Gangs in Configuration**
  Introduces advanced configuration schemes for multiple FPGAs. {Lecture}
- **Bitstream Security**
  Reviews AMD Xilinx bitstream security features, such as readback disable, bitstream encryption, and authentication. {Lecture, Demo}

Debugging

- **Vivado Design Suite Debug Methodology**
  Covers debug core recommendations and how to employ the debug methodology for debugging a design using the Vivado logic analyzer. {Lecture}
- **Trigger and Debug at Device Startup**
  Shows how to debug the events around device startup. {Lecture, Demo}
- **Trigger Using the Trigger State Machine in the Vivado Logic Analyzer**
  Illustrates using trigger state machine code to trigger the ILA and capture data in the Vivado logic analyzer. {Lecture, Lab}
- **Versal ACAP: Debugging**
  Covers the Versal ACAP debug interfaces, such as the test access port (TAP), debug access port (DAP) controller, and high-speed debug port (HSDP). {Lecture}

Vivado Store

- **Introduction to the Vivado Store**
  Introduces the Vivado Store. {Lecture, Demo}

Tcl Commands

- **Scripting in Vivado Design Suite Non-Project Mode**
  Demonstrates how to write Tcl commands in the non-project batch flow for a design. {Lecture, Lab}
- **Debugging the Design Using Tcl Commands**
  Reviews how to use Tcl scripting for VLA designs when adding probes and making connections to probes. {Lecture, Lab}
- **Using Procedures in Tcl Scripting**
  Discusses employing procedures in Tcl scripting. {Lecture}
- **Debugging and Error Handling in Tcl Scripts**
  Describes how to debug errors in a Tcl script. {Lecture}

Register Today

Morgan Advanced Programmable Systems, Inc. (Morgan A.P.S.) delivers public and private courses in locations throughout the central US region; including Iowa, Illinois, Kansas, Minnesota, Missouri, Nebraska, North Dakota, South Dakota, and Wisconsin.

Visit **morgan-aps.com/training**, for full course schedule and training information.

Morgan
Advanced Programmable Systems

You must have your tuition payment information available when you enroll. We accept credit cards (Visa, MasterCard, or American Express) as well as purchase orders and Xilinx training credits.

Student Cancellation Policy

- Student cancellations received more than 7 days before the first day of class are entitled to a 100% refund. Refunds will be processed within 14 days.
- Student cancellations received less than 7 days before the first day of class are entitled to a 100% credit toward a future class.
- Student cancellations must be sent here.

Morgan A.P.S. Course Cancellation Policy

- We regret from time-to-time classes will need to be rescheduled or cancelled.
- In the event of cancellation, live on-line training may be offered as a substitute.
- Morgan A.P.S. may cancel a class up to 7 days before the scheduled start date of the class; all students will be entitled to a 100% refund.
- Under no circumstances is Morgan A.P.S. responsible or liable for travel, lodging or other incidental costs. Please be aware of this cancellation policy when making your arrangements.
- For additional information or to schedule a private class contact us here.

Online training with real hardware

During the Covid-19 period, some companies do not allow their staff to participate in live in-person training.

- Consequently, Morgan Advanced Programmable Systems, Inc. has set up a training VPN where engineer participants can take classes online using the same computers and devCards used during in-person training.
- Even better, and upon request, you can use these computers after hours on training days to experiment with labs. This is not possible for in-person training.
- Additionally, just like in-person training, the laptops and devCards, tools, OS, and licensing are set up in advance.
- In some ways, live online-training is better than in-person…for example, you can grant the instructor permission to look at your Vivado, PetaLinux terminal, or Vitis for extended periods of time if your lab is not going exactly has planned to a missed step.
- This is often more comfortable than two engineers crowding around a laptop screen.

- Taking remote training also allows you to learn some tips and tricks for working remote. Whether your devCard is in the lab down the hall, or across the world via VPN, you can control your Xilinx based device quickly and efficiently.