# Vivado STA, XDC, and Advanced Tools and Techniques of Vivado Design Suite

**FPGA 4**

FPGA-STAXDCADV-ILT (v1.0)

**Course Specification**

## Course Description

This course will update experienced ISE® software users to utilize the Vivado® Design Suite. Learn the underlying database and static timing analysis (STA) mechanisms. Utilize Tcl for navigating the design, creating Xilinx design constraints (XDC), and creating timing reports. Learn to make appropriate timing constraints for SDR, DDR, source-synchronous, and system-synchronous interfaces for your FPGA design.

You will also learn to make path-specific, false path, and min/max timing constraints, as well as learn about timing constraint priority in the Vivado timing engine. Finally, you will learn about the scripting environment of the Vivado Design Suite and how to use the project-based scripting flow.

You will also learn the FPGA design best practices and skills to be successful using the Vivado Design Suite. This includes the necessary skills to improve design speed and reliability, including: system reset design, synchronization circuits, optimum HDL coding techniques, and timing closure techniques using the Vivado software. This course encapsulates this information with an UltraFast™ design methodology case study. The UltraFast design methodology checklist is also introduced.

**Level** – FPGA 2

**Course Duration** – 3 days

**Price** – $2,700 or 27 Xilinx Training Credits

**Course Part Number** – FPGA-VAXDC4ISE-ILT

**Who Should Attend?** – Existing Xilinx ISE Design Suite FPGA designers

**Prerequisites**

- Completion of the Vivado Design Suite for ISE Project Navigator Users course is strongly recommended.
- Designing FPGAs Using the Vivado Design Suite 2 course
- Designing FPGAs Using the Vivado Design Suite 3 course
- Working HDL knowledge (VHDL or Verilog)
- Digital Design Experience

**Optional Video**

- Basic HDL Coding Techniques

**Software Tools**

- Vivado Design or System Edition 2018.1

**Hardware**

- Architecture: UltraScale™ and 7 series FPGAs*
- Demo board: None*

* This course focuses on the UltraScale and 7 series 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:

- Create appropriate clock and input, output delay constraints and describe timing reports that involve input and output paths
- Analyze different timing reports
- Define a properly constrained design
- Describe setup and hold checks and describe the components of a timing report
- Identify key areas to optimize your design to meet your design goals and performance objectives
- Describe all of the options available with the `report_timing` and `report_timing_summary` commands
- Build a more reliable design that is less vulnerable to metastability problems and requires less design debugging later in the development cycle
- Describe the timing constraints required to constrain system-synchronous and source-synchronous interfaces
- Identify timing closure techniques using the Vivado® Design Suite
- Describe how the UltraFast design methodology techniques work effectively through case studies and lab experiences

## Course Outline

**Day 1**

- Static Timing Analysis and Metastability {Lecture}
- Create Clock using four methods {Demo}
- Timing Constraint Wizard {Lecture/Demo}
- Timing Constraint Editor {Lecture/Demo}
- Clocking Wizard {Demo}
- Constraint sets {Demo}
- Long Lab 1: System and source synchronous timing with readable/maintainable constraints.

**Day 2**

- Optional: Vivado Logic Analyzer Insertion flow {Demo, Lab continuation}
- Report Datasheet {Lecture, Demo}
- Baselining {Lecture, Lab, Demo}
- Introduction to Timing Exceptions {Lecture, Lab, Demo}
- ASYNC_REG, HARD_SYNC, MTBF for 7-series and UltraScale {Demo}
- Report Clock Interaction {Lecture, Demo}
- Report Clock Networks {Lecture, Demo}

**Day 3**

- Scripting in Vivado Design Suite Non-Project Mode {Lecture, Lab}
- Hierarchical Design {Lecture}
- Managing Remote IP {Lecture, Lab}
- Timing Constraints Priority {Lecture}
- Introduction to Floorplanning {Lecture}
- Design Analysis and Floorplanning {Lecture, Lab}
- Incremental Compile Flow {Lecture, Lab}
- Physical Optimization {Lecture, Lab}
- Incremental Compile Flow {Lecture, Lab}

## Self-study

The following material and lab instructions are also provided to the student. Material not covered above may be used for self-study (SS) or to refresh your memory on some of the more esoteric techniques covered below.

**SS Day 1**

- Introduction to Clock Constraints – Apply clock constraints and perform timing analysis.
- Generated Clocks – Use the report clock networks report to determine if there are any generated clocks in a design.
- Report Clock Networks – Use report clock networks to view the primary and generated clocks in a design.
- Clock Group Constraints – Apply clock group constraints for asynchronous clock domains.
- I/O Constraints and Virtual Clocks – Apply I/O constraints and perform timing analysis.
- Timing Constraints Wizard – Use the Timing Constraints Wizard to apply missing timing constraints in a design.
- Introduction to Vivado Reports – Generate and use Vivado timing reports to analyze failed timing paths.
- Setup and Hold Timing Analysis – Understand setup and hold timing analysis.

- Timing Summary Report – Use the post-implementation timing summary report to sign-off criteria for timing closure.
- Report Clock Interaction – Use the clock interaction report to identify interactions between clock domains.
- Introduction to Timing Exceptions – Introduces timing exception constraints and applying them to fine tune design timing.
- Timing Constraints Priority – Identify the priority of timing constraints.

**SS Day 2**

- Synchronization Circuits – Use synchronization circuits for clock domain crossings.
- Report Datasheet – Use the datasheet report to find the optimal setup and hold margin for an I/O interface.
- UltraFast Design Methodology: Implementation – Introduces the methodology guidelines covered in this course.
- Baselining – Use Xilinx-recommended baselining procedures to progressively meet timing closure.
- Pipelining – Use pipelining to improve design performance.
- I/O Timing Scenarios – Overview of various I/O timing scenarios, such as source- and system-synchronous, direct/MMCM capture, and edge/center aligned data.
- System-Synchronous I/O Timing – Apply I/O delay constraints and perform static timing analysis for a system-synchronous input interface.
- Source-Synchronous I/O Timing – Apply I/O delay constraints and perform static timing analysis for a source-synchronous, double data rate (DDR) interface.
- Introduction to Floorplanning – Introduction to floorplanning and how to use Pblocks while floorplanning.
- Congestion - Identifies congestion and addresses congestion issues.
- Physical Optimization – Use physical optimization techniques for timing closure.
- UltraFast Design Methodology: Design Closure – Introduces the methodology guidelines covered in this course.

**SS Day 3**

- UltraFast Design Methodology: Design Closure – Introduces the UltraFast™ design methodology guidelines covered in this course.
- Scripting in Vivado Design Suite Non-Project Mode – Write Tcl commands in the non-project batch flow for a design.
- Hierarchical Design – Overview of the hierarchical design flows in the Vivado Design Suite.
- Managing Remote IP – Store IP and related files remote to the current working project directory.
- I/O Timing Scenarios – Overview of various I/O timing scenarios, such as source- and system-synchronous, direct/MMCM capture, and edge/center aligned data.
- System-Synchronous I/O Timing – Apply I/O delay constraints and perform static timing analysis for a system-synchronous input interface.
- Source-Synchronous I/O Timing – Apply I/O delay constraints and perform static timing analysis for a source-synchronous, double data rate (DDR) interface.
- Timing Constraints Priority – Identify the priority of timing constraints.
- Case Analysis – Understand how to analyze timing when using multiplexed clocks in a design.
- Introduction to Floorplanning – Introduction to floorplanning and how to use Pblocks while floorplanning.
- Design Analysis and Floorplanning – Explore the pre- and post-implementation design analysis features of the Vivado IDE.
- Congestion - Identifies congestion and addresses congestion issues.
- Introduction to the Xilinx Tcl Store – Introduces the Xilinx Tcl Store.
- Incremental Compile Flow – Utilize the incremental compile flow when making last-minute RTL changes.

**SS Day 4**

- Physical Optimization – Use physical optimization techniques for timing closure.
- Vivado Design Suite ECO Flow – Use the ECO flow to make changes to a previously implemented design and apply changes to the original design.
- Power Management Techniques – Identify techniques used for low power design.
- Daisy Chains and Gangs in Configuration – Introduces advanced configuration schemes for multiple FPGAs.
- Bitstream Security – Understand the Xilinx bitstream security features such as readback disable, bitstream encryption, and authentication.
- Vivado Design Suite Debug Methodology – Understand and follow the debug core recommendations. Employ the debug methodology for debugging a design using the Vivado logic analyzer.
- Trigger and Debug at Device Startup – Debug the events around the device startup.
- Trigger Using the Trigger State Machine in the Vivado Logic Analyzer – Use trigger state machine code to trigger the ILA and capture data in the Vivado logic analyzer.
- Debugging the Design Using Tcl Commands – Use Tcl scripting for VLA designs for adding probes and making connections to probes.
- Using Procedures in Tcl Scripting - Employ procedures in Tcl scripting.
- Using Lists in Tcl Scripting – Employ lists in Tcl scripting.
- Using regexp in Tcl Scripting – Use regular expressions to find a pattern in a text file while scripting an action in the Vivado Design Suite.
- Debugging and Error Handling in Tcl Scripts – Understand how to debug errors in a Tcl script.

## Register Today

Morgan Advanced Programmable Systems, Inc. (MAPS, Inc.) 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

- At Morgan Advanced Programmable Systems, we understand that from time to time engineers will have unexpected issues to deal with at work or at home.

- Consequently, 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.

## 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 Advanced Programmable Systems, Inc., 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.
- In rare events such as instructor injury, accident, or family/health issues, or natural disasters, acts of war, and other serious events, training may also be rescheduled or cancelled within 7 days of training. In this case, students are entitled to a full refund.
- Under no circumstances is Morgan Advanced Programmable Systems, Inc., responsible or liable for travel, lodging or other incidental costs or losses due to cancelled or postponed training. Please be aware of this cancellation policy when making your arrangements.
- For additional information or to schedule a private class contact us here.