Intel® FPGA Technical Training

Introduction to Timing Analysis with TimeQuest

Course Description

This course provides all theoretical and practical know-how start writing sdc files and analyze your design in TimeQuest tool.

The course combines 50% theory with 50% practical work in every meeting. The practical labs cover all the theory.

The course starts with an overview of what need to be constrained in every design, the timing terminology used by the tools.

The training continuous by introducing the clock constraints, I/O constraints and exception constraints.

The training explains how to generate and read the various reports in TimeQuest in order to solve timing issues.

Course Duration

2 days

Goals

1. Become familiar with timing analysis types and terminology
2. Become familiar with the TimeQuest tool
3. Write constraints for clocks, reset, I/O, and exceptions
4. Generate various reports and analyze timing results
5. Be able to solve timing issues

Intended Users

Digital hardware engineers who program with VHDL/Verilog languages and would like to constrain and analyze their design timing.

Previous Knowledge

FPGA design.

FPGA architecture.

Quartus Prime software.

Course Material

1. Synthesizer and Place & Route: Quartus Prime Standard
2. Course book (including labs)

Table of Contents

Day #1

- **Introduction to Timing Analysis**
  - TimeQuest tool overview
  - Basic steps to using TimeQuest (generate timing netlist, enter SDC constraints, update timing netlist, generate timing reports)
  - Using TimeQuest in Quartus Prime flow
  - Timing analysis basics (Launch Vs Latch edges, setup and hold times, data and clock arrival time, data required time, setup and hold slack analysis, I/O analysis, recovery and removal, timing models)

- **Timing Reports**
  - Reporting in Quartus Prime Vs reporting in TimeQuest
  - Custom, summary and diagnostic reports
  - Clock transfer, datasheet, Fmax reports
  - Slack histogram report
  - Detailed slack/path report, further path analysis

- **Introduction to Timing Constraints**
  - Importance of constraining
  - Enter constraints
  - SDC netlist terminology
  - Collections

- **SDC Timing Constraints for Clocks**
  - Internal and virtual clocks
  - Generated clocks (inverted clocks, phase shifted clocks
  - PLL clocks and derive_pll_clocks Altera SDC extension
  - Automatic clock detection and creation
  - Non ideal clock constraints (Jitter, latency on PCB)
  - Common clock path pessimism removal
  - Checking clock constraints
  - Report clocks

Day #1 Labs

❖ Lab #1: TimeQuest Interface & Timing Reports Generation

- Start the TimeQuest GUI and create timing netlist for analysis
- Use TimeQuest reports to verify that design meets timing
- Use the SDC file to guide the Quartus Prime fitter

❖ Lab #2: Timing Analysis: Clock Constraints

- Create SDC file for a given design
- Create base and generated clock constraints
- Analyze the clock constrained design

Day #2

- **SDC Timing Constraints for I/O**
  - Combinational I/O interface constraints (max & min delay constraints)
  - Constraining synchronous I/O
  - I/O timing: virtual clocks
  - Synchronous inputs constraints (setup and hold time calculations, set_input_delay max & min)
  - Synchronous outputs constraints (set_output_delay min& max, when to use each constraint)
  - Constrain I/O using Tcl variables
  - Checking I/O constraints (report SDC, report unconstrained path, report ignored constraint)

- **Asynchronous Paths**
  - TimeQuest & asynchronous ports
  - Recovery and removal
  - Externally registered
  - Internally registered
  - Checking asynchronous control constraints
  - What about truly asynchronous control inputs?

- **Timing Exceptions**
  - Timing exceptions types
  - False paths
  - set_clock_groups command
  - Verifying false paths and groups
  - Report exceptions
  - Multicycle types (setup, hold, end, start)
  - Case 1: opening the window
  - Case 2: shifting the window
  - Determining and applying multicycles
  - Multicycle path constraints
  - Reporting multicycles in TimeQuest
  - Exception priorities

Day #2 Labs

- **Lab #1: Timing Analysis: Synchronous I/O Constraints**
  - Constrain synchronous input paths using SDC
  - Constrain synchronous output paths using SDC
  - Generate reports and analyze results

- **Lab #2: Timing Analysis: Timing Exceptions & Analysis**
  - Constrain asynchronous input signals
  - Eliminate timing violations by using timing exceptions