ALTERA FPGA Design Using VHDL

The course is designed to teach students and professionals how to design using ALTERA FPGA in Quartus II.

The course provides a 50% discount on all courses related to FPGA and VHDL.

The course is suitable for beginners and professionals in the field of VLSI.

The course includes:
1. Getting Started with VHDL
2. Understanding VHDL
3. Designing with VHDL
4. Using VHDL in Digital Design
5. FPGA Development with VHDL
6. Advanced VHDL Techniques
7. VHDL Libraries and Tools
8. VHDL Simulation and Verification
9. VHDL Hardware Description Language
10. VHDL SystemVerilog
11. VHDL and SystemC Integration

Specialized Curriculum for CPLD and FPGA Design

The course provides specialized curriculum for CPLD and FPGA design, taught by experts in the field.

Contact Us:
HandsOnTraining.co.il | 077-4702742 | 052-5816791 | info@HandsOnTraining.co.il
Day #1

• Introduction to Programmable Logic Devices
  o CPLD architecture and design consideration
  o FPGA architecture
    ▪ LUT
    ▪ FF
    ▪ PLL
    ▪ DSP Block
    ▪ Embedded RAM
    ▪ Embedded Processor
    ▪ FPGA Programming process

• Introduction to VHDL Language
  o VHDL history
  o Digital design
  o FPGA design
    ▪ Simulation
    ▪ Synthesis
    ▪ Place & Route
    ▪ Programming
    ▪ Verification
  o Advantages of VHDL
  o Simulation & Synthesis
  o Demonstration of whole process on board

• VHDL Basic Structures Overview
  o Entity
  o Component
  o Architecture
  o Process
  o Functions & Procedures
  o Package & Package Body

When innovation meets expertise...
• Library
• Configuration
• Top down design

• VHDL Design Units – Building a Hierarchy
  o Building MUX from its primitives
  o Port Map
  o Test bench and simulation

Day #2

• More on Entities
  o Ports name
  o Direction
    ▪ In
    ▪ Out
    ▪ InOut
    ▪ Buffer
  o Data types
  o Generic
  o Generic map

• Architecture Bodies
  o Architecture Declarative Part
  o Behavioral Description
  o Data Flow Description
  o Structural Description

• Concurrent Statements
  o Simple signal assignment
  o Concurrent signal assignment
  o Implementation of signals
  o Conditional signal assignments
    ▪ When-Else
    ▪ With-Select
  o Examples of mux and encoder

• VHDL Timing Model
  o Inertial delay
  o Transport delay
  o Delta delay
  o Reject reserved word

When innovation meets expertise...
• **Grammar and Declarations**
  o Data Types & Data Objects
  o Enumeration Types
  o Attributes, Subtype
  o Numeric Data Types (Integer & Real)
  o Physical Data Types
  o Composite Data Types
    ▪ Array
    ▪ Array attributes
    ▪ Record
    ▪ Aggregate
    ▪ Multi-Dimensional Array

**Day #3**

• **Sequential Processing**
  o Process definition
  o Sensitivity list
  o Declaration area
  o Statement area
  o Sequential execution
  o Demonstration
  o Concurrent signal assignment versus process

• **Sequential Control Statements**
  o If-Elseif-Else statement
  o Case Statement
  o Loop statement
  o Next & Exit Loop control statements
  o Null
  o Assert statement
  o Wait statement
Day #4

• Process Behavior
  o Signal assignment inside and outside a process
  o Signal and variable assignment differences
  o Process communication
  o Understanding the simulator algorithm
  o Passive process

• Modeling Finite State Machines
  o FSM concept
  o Mealy & Moore Models
  o HDL coding style
    ▪ One process
    ▪ Two processes
    ▪ Three processes
    ▪ Mealy & Moore
  o State encoding
    ▪ Sequential
    ▪ Johnson
    ▪ One Hot
    ▪ Two Hot
    ▪ Defined by user
    ▪ Defined by synthesis
  o Handling the unused states
  o Reset & Fail Safe Behavior
  o Interactive State Machines
    ▪ Unidirectional
    ▪ Bi-Directional
Day #5

- **Introduction to ALTERA FPGAs**
  - CPLD : MAX V, MAX II
  - FPGA : Cyclone IV, Cyclone V, ARIA II, ARIA V, STRATIX IV, STRATIX V, SoC ARM based

- **Introduction to Quartus II IDE**
  - Subscription Vs Web edition
  - Project creation and the main windows in Quartus II
  - Quartus II project files and folders
  - Project management (archive, restore, copy, revisions)
  - Design entry methods (schematic, HDL, EDIF)
  - Synthesis and P&R
  - Programmer

- **RTL Coding Guidelines for QUARTUS II Integrated Synthesis**
  - Synthesis directives and attributes
  - Fixed output registers
  - Combinational circuits inference
  - Sequential circuits inference
  - RAM inference
  - Latch inference
  - Combinational loops problem
  - Finite state machine coding styles

- **Compilation Process**
  - Processing options
  - Compilation design flows
  - Viewing compilation results
  - Netlist viewers
  - State machine viewer
  - Chip planner
  - Resource property editor
Day #6

- **Example: Cyclone IV Architecture Deep Dive**
  - Logic elements and logic array blocks (LE modes, LAB topology and control signals)
  - Memory blocks (control signals, parity bit support, byte enable support, packed mode, address clock enable, mixed width, asynchronous reset, memory modes: FIFO, ROM, Shift register, DPR, SDR, clocking modes, design considerations)
  - Embedded multipliers (architecture, operational modes)
  - Clock networks and PLLs (GCLK networks, PLLs hardware, programmable bandwidth, phase shift implementation, PLL cascading, PLL reconfiguration, spread spectrum clocking)
  - I/O interfaces (I/O element features: slew rate control, programmable current strength, open drain output, bus hold, programmable pull-up resistor, programmable delay, PCI clamp diode, on chip termination, I/O standards, I/O banks, clock pins functionality, high speed I/O interface, true differential output buffer support, design guidelines)

- **IP Megafuntions**
  - MegaCore, AMPP, OpenCore IPs
  - MegaWizard plug-in manager tool
  - Generated files

- **Memories**
  - Memory editor
  - Create memory initialization file
  - Using memory file in design
Day #7

- **Assignment Editor**
  - Synthesis and fitting control
  - Synthesis settings
  - Fitter settings
  - Assignment editor features
  - Design assistance

- **I/O Management**
  - Pin planner
  - I/O assignment analysis
  - Using synthesis attributes in HDL
  - Live I/O checking

- **Introduction to FPGA Verification**
  - Functional versus gate level simulation
  - Simulating ALTERA IP cores
  - System debugging tools overview (in-system sources and probes, in-system memory content editor, signal probe, SignalTap II, LAI)

Day #8

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

When innovation meets expertise...
• **Introduction to Timing Constraints**
  o Importance of constraining
  o Enter constraints
  o SDC netlist terminology
  o Collections

• **SDC Timing Constraints**
  o Internal and virtual clocks
  o Generated clocks (inverted clocks, phase shifted clocks
  o PLL clocks and derive_pll_clocks Altera SDC extension
  o Non ideal clock constraints (Jitter, latency on PCB)
  o Checking clock constraints
  o Report clocks

• **SDC Timing Constraints for I/O**
  o Combinational I/O interface constraints (max & min delay constraints)
  o Synchronous inputs constraints (setup and hold time calculations,
    set_input_delay max & min, set_output_delay min& max, when to
    use each constraint)
  o Checking I/O constraints (report SDC, report unconstrained path,
    report ignored constraint)