Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ingeniería Eléctrica
IEE2463 Sistema electrónicos programables

Laboratorio 02

ZYBO-Z7 Vivado Simulation

Prof. Dr.-Ing. Félix Rojas - felix.rojas@uc.cl

1. Laboratory Goals

After this laboratory, the student should learn:

- To identify differences between behavioral, post-synthesis and post-implementation simulations in Vivado.
- To recognize and use the setting and navigation panel of a Simulation environment.
- To manually modify signals and clocks into the simulation environment to test user designed IPs.
- To design and implement a Testbench for fast and comprehensive simulation of user designed IPs.

2. Previous Requirements

These requirements are mandatory to perform the laboratory. Not accomplishing them count as missing the laboratory.

- You must have previously installed vitis/vivado version 2020.1.
- You must read the ZYbo Z7 Board Reference.

Note: **Version of the software 2020.1 is mandatory**. This is important to avoid compatibility problems.

3. Laboratory Activities

- Create a new Vivado project associated to ZyboZ7.
- Include ZyboZ7 constrains into the vivado project.
- Include a new source into the vivado project. Use the provided source for implementing this lab (compare.vhd).
- Run a behavioral simulation:
    - Modify all input signals manually to test your VHDL code. Check outputs and validate the logic of your design.
    - Modify the simulated time.
- Create a Testbench for your source file and test the unit. Run a behavioral simulation.
- Run project synthesis and then run a post-synthesis simulation. Modify the inputs of the IP every 2ns and 2us, see if there is any difference into the simulation. Also study the change on the synchronized output.

4. Complementary Homework

To fulfill this homework is mandatory, but not evaluated.

- Associate two switches and one led of the ZYboZ7 to your code. Use the switches as input and led as output of your compare.vhdl source validate it. Make all needed changes into the compare.vhd file.
- Run project synthesis & implementation, then run a post-implementation simulation. Compare the results with post-synthesis simulation.

Information about Simulation:

Post-Synthesis Simulation: You can simulate a synthesized netlist to verify that the synthesized design meets the functional requirements and behaves is expected. Although it is not typical, you can perform timing simulation with estimated timing numbers at this simulation point.

The functional simulation netlist is a hierarchical, folded netlist expanded to the primitive module and entity level; the lowest level of hierarchy consists of primitives and macro primitives.

These primitives are contained in the UNISIMS_VER library for Verilog, and the UNISIM library for VHDL.

Post-Implementation Simulation You can perform functional or timing simulation after implementation. Timing simulation is the closest emulation to actually downloading a design to a device. It allows you to ensure that the implemented design meets functional and timing requirements and has the expected behavior in the device.

Important: Performing a thorough timing simulation ensures that the completed design is free of defects that could otherwise be missed, such as:

- Post-synthesis and post-implementation functionality changes that are caused by:
    - Synthesis properties or constraints that create mismatches (such as full_case and parallel_case)
    - UNISIM properties applied in the Xilinx Design Constraints (XDC) file
    - The interpretation of language during simulation by different simulators
- Dual port RAM collisions
- Missing, or improperly applied timing constraints
- Operation of asynchronous paths
    - Functional issues due to optimization techniques

Note: For Versal devices, post synthesis and post implementation simulation are supported only for fabric logic (PL) and not supported for designs with Hard Blocks (NoC/AIE/PS). Only behavioral simulation is supported for designs utilizing Hard Blocks.