ESE382 Digital System Design Using VHDL and PLDs

Spring 22, Ken Short
April 22, 2022 6:47 pm

PRELIMINARY PRELIMINARY PRELIMINARY PRELIMINARY PRELIMINARY

Laboratory 12: Controller for TC514 Precision Analog Front End II - Sequential Testbenches and Simulation

This laboratory is to be performed the week starting May 1st.

Prerequisite Reading

1. Chapter 14 of the text.
2. TC514 Precision Analog Front End data sheet (on Blackboard).

Purpose

The purpose of this laboratory is to provide you with some experience in writing testbenches for sequential circuits. Whereas an exhaustive testbench for a combinational circuit is, typically, easy to write, this is not the case with a non-trivial sequential circuit. This is because of the extremely large number of input sequences that would have to be generated and applied to the UUT to exhaust the number of different possible sequences.

Instead of an exhaustive simulation the goal with most sequential systems is to produce a comprehensive simulation. A comprehensive simulation is one that provides a high degree of confidence that the design is correct if the simulation is passed.

For non-trivial sequential systems using stimulator is not practicable. Instead, one or more sequential testbenches must be written. Every sequential testbench will require a reset signal and a system clock as inputs to the UUT. In addition control and data inputs must be provided to the UUT and sequenced in a manner that is appropriate to the operation of the UUT. Outputs from the UUT must be verified at the times when the should be valid, if the testbench is to be self-checking.

In some applications, the device(s) that the UUT interfaces to may be complex enough that, instead of trying to generate their inputs directly for the simulation, it is more appropriate to write a VHDL model of the system that the UUT interfaces to and make the testbench a structural system that interconnects the UUT with the VHDL model. Often, the system that the UTT connects to is so complex that it would be a significant effort to model it completely in VHDL. Or, the system the UUT connects to may be primarily analog and not able to be modeled in VHDL. In these cases, the VHDL model of the system that the UUT connects to is often what is called a bus functional model. A bus functional model is one that accurately models the system at its external ports, but not internally. For example, in the case of the “Controller for TC514” a bus function model for the TC514 could be created. You will be provided with such a bus functional for the last task in this laboratory so that you can study it and see how such a model may be constructed.

Design Tasks

For each of the component design entities for Laboratory 11 you must write a comprehensive testbench. For the top-level design entity for Laboratory 11, you will be provided with a bus functional model of the TC514 and must use that to create a comprehensive self-checking testbench.

An extra credit task is provided at the end involving extending the bus functional model of the TC514 to make it more comprehensive.

Design Task 1: Self-checking Testbench for the Frequency Divider

Write a self-checking testbench named `freq_div_`*`TB`* that verifies the functionality of your design entity `freq_div` from Laboratory 11. If you are not successful in writing a self-checking testbench for this entity, then write a non-self checking one for partial credit.

Submit your testbench design description and simulation output waveforms as part of your prelab.

Design Task 2: Self-checking Testbench for the 16-Bit Binary Counter

Write a self-checking testbench named `binary_cntr_`*`TB`* that verifies the functionality of your design entity `binary_cntr` from Laboratory 11. If you are not successful in writing a self-checking testbench for this entity, then write a non-self checking one for partial credit.

Submit your testbench design description and simulation output waveforms as part of your prelab.

Design Task 3: Self-checking Testbench for the Output Register

Write a self-checking testbench named `out_reg_`*`TB`* that verifies the functionality of your design entity `out_reg` from Laboratory 11. If you are not successful in writing a self-checking testbench for this entity, then write a non-self checking one for partial credit.

Submit your testbench design description and simulation output waveforms as part of your prelab.

Design Task 4: Top-Level TC514cntrl and Bus Functional Self-checking Testbench

Write a top-level structural design description for the design entity `TC514cntrl` that uses the design entity components from Laboratory 11 to implement the controller for the TC514 Precision Analog Front End.

Use the TC514 VHDL model provided to create a self-checking testbench for the design entity `TC514cntrl`.

Submit your design description source code for the `TC514_cntrl`, bus functional testbench code using the TC514 VHDL model, and simulation output waveforms as part of your prelab.

Laboratory Tasks

Using Aldec Active-HDL create a new workspace named `TC514_contrller`. In this workspace, create a separate design for each entity. For each design entity, import and compile your VHDL source file. When each compilation is successful, using a value of $n = 16$, functionally simulate each design entity using your self-checking testbench. After each component design entity has been simulated, simulate the top-level entity using your self-checking testbench that uses the bus functional model provided.

Have a TA verify that each of your design entity simulations generates the appropriate output waveforms.

With generic *n* set to 16, compile your `TC514cntrl` design description. Place and route your design to a Lattice LCMXO3L-6900C FPGA. Pin assignments will be provided in the laboratory. Place your programmed FPGA into the test fixture provided and demonstrate that it operates properly.

Have a TA verify that your programmed FPGA functions properly in the test fixture.