Tutorial

on

Simulation using Aldec Active-HDL

ver. 2.6

Spring 2012
Preparing the Input: Go to the link given above and download following files from the following link:

http://ece.gmu.edu/coursewebpages/ECE/ECE448/S11/labs/448_lab1.htm

VHDL Source Files:

1. MLU.vhd
2. MLU_TB.vhd
3. MLU_TB2.vhd

Current Version of Tools: This tutorial has been tested using the following tools CAD Tools
• ActiveHDL Versions: 7.2, 8.2, 8.3, 9.1

The combinations of tools supported as of Spring 2012 are as follows:

At home:
Aldec Active-HDL Student Edition ver. 7.2

At GMU:
Aldec Active-HDL ver. 9.1 SP1
**Introduction:**

Active-HDL is an integrated environment designed for development of VHDL designs. The core of the system is a VHDL simulator. Along with debugging and design entry tools, it makes up a complete system that allows you to write, debug and simulate VHDL code. Based on the concept of a design, Active-HDL allows you to organize your VHDL resources into a convenient and clear structure.

Students can perform the following tasks:

- Development of VHDL based design,
- Functional simulation of their code,
- Functional simulation of the synthesized code,
- Timing simulation of the hardware implementation.

**Objective:**

This tutorial helps you to

- Create a new design or add .vhd files to your design
- Compile and debug your design
- Perform Simulation

Note: This tutorial does not explain the synthesis or implementation steps.

**Start-up**

1. Start => VLSI Tools => Active-HDL 9.1
Click Next

License Configuration

Select one of the Active-HDL product configurations and click [Next].

EDU Mixed Design Entry

Reserve simulation features at startup

Simulation features

For more details click license information.

License information...

<Back  Next  Cancel>
a. Select “Create new workspace” and click OK.
b. If you select “open existing workspace”, you can choose any previous workspace that already exists.

2. Select a name and location for the workspace and click ok.
3. Select “Create an Empty Design with Design Flow” and click OK.

4. Choose the block diagram configuration as “Default HDL Language” and default HDL Language as “VHDL”, Select the target technology and click “Next”.

5. Select a name of the design and click OK.

click “Finish”
6. In the Design Browser window on the left hand side, current workspace and design are displayed.

You can either add any existing file to the workspace or create a new file using the Tool’s text editor.

1. To create a VHDL source file, either use the HDL Editor from Design flow or go to Add New File => New => VHDL Source

Click on this button to open HDL Editor. This will guide you to create a template for new VHDL source file.
1. Choose an appropriate name for the “Source File”. Select a name of the “Entity”. For consistency, please pick the same name for the VHDL source file and name of the Entity. Choose a name for the “Architecture”
1. You can also add Input and Output ports from the dialogue box below by selecting a Name, Port Direction, Array Index and Type. For now, leave any input and output selection. In the end, click “Finish”.

Remove the new source file “MLU.VHD” for now. It was created to guide through the process of creating a new source file.
2. To add existing files Go to Add New Files => Add Files to Design, and browse the directory containing the source files.
**Compile:** Once all files are added to the workspace, now you have to “Compile” them to create the simulation model of the described circuit. The compiler checks all the syntax and writes all the necessary information in internal binary format.

3. Right Click the file you want to compile and select “Compile”. You can also combine all source files by clicking “Compile All”.

---

**Diagram Image:**

- **Workspace:**
  - `Lab1_Design`
  - `MUL Tek v5`
  - `MUL Tek v6`
  - `MUL Tek v6`
  - `Add New Library`
  - `Lab1_Design library`

- **Options:**
  - `Flow Settings`
  - `Tools`
  - `Analysis`

- **Menu:**
  - `Compile`
  - `Analyze`
  - `Compile All`
  - `Open`
A Tick in green color appears next to the successfully compiled file.

Click on “options” button to specify the inputs files for Functional Simulation.
For Function simulation, it is important to specify your top level. We have two test bench files MLU_TBB.VHD and MLU_TBB2 therefore we will select them one by one and then follow their simulation in two parts. Right now, select MLU_TBB as your top level unit and add it.
It is also important to set MLU_TN as a Top-Level from the Lab1_Design library. Otherwise, your design will only show MLU as your Top-Level file and now signal related to test bench will appear.
**Simulation:** Once you have all the source files compiled, the design can be simulated for functional correctness.

Click on button to start “new waveform”

4. From the menu bar, select “Simulation => Initialize Simulation”,

![Simulink screenshot](image-url)
5. Now you have to select the signals you want to monitor, copy them to the simulation window and proceed. For this example, we select all signals of the design and paste them to the simulation window.

Select the signals from Design Browser window

Drag the signals to the waveform window or right click the signals to choose the option “Add to waveform”
1. Click on button to start the simulation, Press if you want to stop the simulation. You can also press , if you want the simulation to run for the specified amount of time.

Now click on UUT: MLU (mlu_dataflow) to see the signals related MLU source file in the Design Browser window.

Drag these signals to the waveform window as well. At first, there will not be any waveform related these signals.
Click on Restart button to apply the changes to the waveform.
Save your waveform in two of the native formats for Aldec (.awc and .asdb).

End the simulation using command “End Simulation”
Now select the test bench file “MLU_TB2” as your Top-Level file.

All the signals related to MLU_TB2 will be displayed in Design Browser window.
Just like MLU_TK, again start a new waveform, drag the required signals to the waveform, initialize and run the simulation.

Chose the print preview option from File menu. You can take the print out and 3\textsuperscript{rd} party tools to create and pdf file of the waveform.
Page Setup option provides you with several options to display the output in output file.

Save the simulation files for both the test bench files (MLU_TB and MLU_TB2) in a folder.
Use the Language Assistant from the shortcut menu. This will be of great help in writing VHDL code and help you with all constructs and templates for VHDL coding.
Overall view of the simulation waveform for MLU_TB2: