Sharad Seth <sharadcseth@gmail.com>

lab3 -- synthesis part

Sina Balkir <sbalkir@unl.edu> **Wed, Oct 6, 2010 at 7:29 PM**
To: Sharad Seth <seth@cse.unl.edu>

OK. Lab3 will not be exactly duplicated but to a high degree it will be similar. There are profound reasons for this which are related to specific version of Cadence utilities being used in different schools etc. Anyway, here is what the students should do for lab3.

For section III (synthesis) of lab 3, keeping in mind that we are using "mudd" instead of "IC_CAD":

1- Students first need to copy a new version of mudd.rc from /doppler/cadence/mudd_labs/ into their home and replace the old one with the new one (cp /doppler/cadence/mudd_labs/mudd.rc ~/.) . Since we do not have synopsys rtl compiler we have to use the rtl compiler of cadence, and this new file points into that in addition to previous settings.

2- ignore all the synopsys related instructions in the lab3 handout...

3- instead, we use a new .tcl script for invoking cadence rtl compiler. For this we need a cadence script, which is located at /doppler/cadence/mudd_labs/lab3. Its name is "Start.tcl"

4- Once students create a new directory called "synth" under their mudd directory (mkdir ~/mudd/synth), they will need to copy the file mentioned in step 3 to this directory. Also, copy the "mips.sv" file from lab2 to this directory.

5- Ignore any line in the labs3 handout saying syndc.tcl

6- Instead invoke rtl compiler (from under the newly formed synth directory), by typing the command " rc " from the command line. After a few moments you will be in the system prompt of the Cadence RTL compiler. It will look like this: rc1>

7- From this prompt (under the newly formed synth directory), type the command: source Start.tcl

8- This will invoke the RTL - to - logic synthesis of the mips.sv description.

9- Once successfully executed, the RTL compiler will dump two important files. One is the structural verilog netlist which is synthesized by the RTL compiler and the other one is timing constraint file for place and route. See page 4 of lab3 handouts for the names.

---------- end of synthesis ----------

Sharad Seth <sharadcseth@gmail.com>

lab 3 - place and route

Sina Balkir <sbalkir@unl.edu> **Wed, Oct 6, 2010 at 7:51 PM**
To: Sharad Seth <seth@cse.unl.edu>

Ok. This flow will not be 100% duplicated as the versions of tools are not fully matched at different lelves of abstraction. But, we will be able to place and route a std cell layout and call it into the icfb environment -- but no LVS will be performed, which is not really necessary, as an LVS equivalent step is already done in the place route module. Here it is:

Section IV of lab3:

1- create a new directory called soc under the ~/mudd directory.

2- There is a new cds library configuration file placed under /doppler/cadence/mudd_labs, named cds-encounter.lib

3- IMPORTANT!!!: First, this file should *not* be copied over the existing cds.lib file under the ~/mudd directory of each user. Instead, the entries for basic and analogLib should be added to their existing cds.lib file without altering anyting else. This can be done by a text editor like vi or else.

4- Follow the steps of page 4 and page 5 of lab3, keeping in mind that IC_CAD is mudd in your case.

5- Also, /courses/e158/10/lab3 path should be changed to /doppler/cadence/mudd_labs/lab3 on page 5

6- Also, we use the command "encounter" from the system prompt instead of "cad-soc".

7- Rest of the flow on page 5 and page 6 until section V should work as is.

------- end of place and route -----------

Lab 3 -- importing into the virtuoso environment (section V)

Sina Balkir <sbalkir@unl.edu> **Wed, Oct 6, 2010 at 8:27 PM**
To: Sharad Seth <seth@cse.unl.edu>

OK. This is the last step where we will have significant differences.

First, pay attention to the author's typo in writing the mudlib10; it should be muddlib10

Second, there is a profound logistic issue, when the author says to set the target library name to lab3_xx toward the bottom of page 6. This is not quite that simple. First, the lab3_xx target library is *not* the lab3_xx directory. We first have to create a new library named lab3_xx using the library manager tool of the icfb main command window and attach to it the UofUtah AMI0.5 tech library (similar steps found in lab1). Without this, the rest of the procedure will fail. To do this, first go to your top level mudd directory (cd ~/mudd), but do *not* go to soc or synth etc.

Once a new library named lab3_xx is formed by using the new library function of the icfb window, under ~/mudd, you can then invoke icfb from under the top level mudd directory as usual and follow the steps in the last paragraph of page 6.

IMPORTANT: due to a profound difference in the assign statements of the structural verilog, the schematic formation step fails to perform. So, will skip this part but it is good practice to see how it fails.

Next, on top of page 7 of lab3, execute the import steps as instructed and you should be able to see the layout eventually.

Finally, the DRC may work but LVS will fail as there is no schematic to compare against. However, keep in mind that in the encounter there was already a verification of the connectivity done so we should be OK without this step.

Note: If all the steps are done with being careful and with no typing mistakes etc. these steps should work. -- well, mine did :-) Hope this helps

-------end----------