Exam rules for HDL projects

[1] Project assignment

Check the list of the available project on the Teams channel related to the course (930II <academic year>) and send an email to the Professor Sergio Saponara (sergio.saponara@unipi.it) and to the project referent (Luca Crocetti, luca.crocetti@phd.unipi.it, or Stefano Di Matteo, stefano.dimatteo@phd.unipi.it) as Carbon Copy (CC), in order to request the assignment of a specific project.
Once the project is assigned, the project referent will send to each member of the group an archive (.zip) containing the work environment and the documentation for the project.

[2] Project development and rules

1. Extract the archive delivered by email and develop your project by respecting the organization foreseen for folders and files, i.e.:
   - Put the **HDL design file(s)** inside folder **db/**
   - Put the HDL file(s) of **testbench(es)** inside folder **tb/**
   - Put the eventual **test vectors** invoked within testbench(es) inside the folder **modelsim/tv/**
   - Put the eventual design **constraints file(s)** (.sdc) inside the folder **quartus/constr/**
   - Put the **project report** inside the folder **doc/**

   The folders **quartus/** and **modelsim/** will contain **scripts** to automatically generate the Quartus project and the Moldesim project (according to the content of folders db/ and tb/): if required/useful, use the script **build.py** in the corresponding folder, otherwise you can also create the tools projects manually, following the explanations shown during lectures.
   If available, the folder **modelsim/tv/** will contain file(s) to be used as **reference test vectors**.
   Folder doc/ will contain this document, as reminder, a guideline to setup your OS environment in order to use the scripts and the trace of the project assigned to you.

2. The **project report** shall respect the following rules
   - **Name** of the file: **project_report_<student_surname_1>[_<student_surname_2>][_<…>].pdf**
     i. PDF format is not mandatory, but highly suggested
   - The trace of the project report shall follow structure illustrated below ('Project report: structure and content')
     i. **Points 1 to 5 are mandatory** (to consider the project report and the project development as complete)
     ii. Point 6 is optional and is aimed to increase the final evaluation.

   For instance, assuming the project X has been assigned to the students Luca Crocetti and Stefano Di Matteo, they will have to produce a project report named **project_report_Crocetti_DiMatteo.zip**.

[3] Project delivery

Clean the project work environment (i.e., run the script **clean.py** inside both folders quartus/ and modelsim/): the files inside folder db/, doc/, tb/, quartus/constr/ and modelsim/tv/ (in addition to the scripts build.py and clean.py) will be kept and not deleted, so, **pay attention to store the project files in the dedicated folders**!!!

Once cleaned, compress the project work environment into an archive (.zip) and send it by email to:

- Professor Sergio Saponara (sergio.saponara@unipi.it)
- Project referent (as Carbon Copy – CC)

The .zip archive shall have a name similar to the one the project report file, by appending the surname of each member of the group, separated by the underscore character, i.e.:
<original_zip_archive_name>_<student_surname_1>[_student_surname_2>][_<…>].zip

For instance, assuming the project "X" has been assigned to the students Luca Crocetti and Stefano Di Matteo, they will receive an archive named **project_930II_21.22_X.zip**, and when they will have finished the project development, they are required to deliver an archive named **project_930II_21.22_X_Crocetti_DiMatteo.zip**.

Project report: structure and content

1. **Specification analysis**
   Elaboration of specifications and requirements to demonstrate how they have been parsed and used: a sort of introduction which describes the project.

2. **Block diagram and design choices (RTL design module)**
   Description of the design architecture, with block diagrams, schematics (functional and/or architectural), FSM diagrams, etc.

3. **Expected waveforms**
   Snapshot(s) and image(s), by screenshot or export from Modelsim, of waveforms showing significant signals and significant behaviour of signals, with description and/or comments.

4. **Testbench (block diagram, testbench design choices, comments on test-plan)**
   Similar to point 2, architectural and/or functional description of the testbench and verification methodology/approach.

5. **Implementation of RTL design on FPGA and results**
   Results of FPGA implementation of the design by means of Quartus tool, with comments (and snapshot, if suitable) of Quartus reports.

6. **Optional – Static Timing Analysis (STA)**
   Description and comments of STA results (maximum frequency, …), with snapshots, if suitable. Please, attach also the timing constraints file (SDC or PDF format) to the project report.

Note: to generate the schematics or block diagrams for the project report, any software tool for drawing can be used, anyway it is highly suggested to use the Microsoft tool **Visio**. It is included within the package of free licenses that University of Pisa offer to their students for the Microsoft products: for more information refer to http://www.sid.unipi.it/polo6/studenti/licenze-software/. To download such tool, access at https://azureforeducation.microsoft.com/devtools, download the installer and the license activation key.