Lab 22. Advanced Vivado Flow

In this lab, we will play with some advanced Vivado feature. We will reuse the same design as in Lab1: a simple counter that switches on and off the LEDs on the Basys3 board.

Go to `~\labs\lab22` and create a Vivado Project with the following settings

- Name: `lab22`
- RTL Project
- Add Sources: `src/LED_Counter.vhd` (Synthesis and Simulation) and `sim/tb_led_counter.vhd` (Simulation only)
- Add Constraints: `src/Basys3.xdc`
- In the `Default Part` select the `Basys3` from the Boards tab.

1. Simulating the Design

As first step, run the simulation for at least 10 us and check that the LED counter is increased every 100 clocks. Add the `clk_div` signal to the waveform, to monitor the counter.

If the behaviour is as expected, run the synthesis. Once it's finished, run again the Simulation, this time with Post-Synthesis Timing Simulation option. Add again the `clk_div` signal to the waveform.

Do you notice any difference with respect to the previous simulation?

Do you see any delay between data and clocks? Are they safe?

2. Synthesis attributes

Run the report utilization of the synthesis and take notes of the utilized resources. Open also the schematics, and have a look at the implemented logics.

Now open the `LED_Counter.vhd` file, and set the `use_dsp48` attribute for the entity.

```
attribute use_dsp48: string;
attribute use_dsp48 of LED_Counter: entity is "yes";
```

Optionally, you can create a new design run, by clicking on the plus in the *Design Runs* tab, in the bottom pad. Once you are done launch the synthesis again, and compare the utilised resources with the previous run.

What do you observe? Open again the Schematic. Can you identify the two DSP blocks?

3. Recreate the project using Tcl

Close the current project, and open the tcl script `lab22.tcl`, using your favorite text editor.

```
kate lab22.tcl &
```

Modify the script to implement the following features, using the commands shown during the lecture.

- Creates a new Vivado Project, named `lab22_tcl`, targeting the FPGA part: `xc7a35tcpg236-1`, in the `lab22_tcl_dir` folder
- Adds `src/LED_Counter.vhd` to the project in the `sources_1` fileset.
- Adds `src/Basys3.xdc` to the project in the `constrs_1` fileset.
- Adds `sim/tb_led_counter.vhd` to the project in the `sim_1` fileset.
- Runs synthesis, implementation and write_bitstream steps

Once the script is ready, execute it with Vivado

```
vivado -mode batch -source lab22.tcl
```

Wait for the script to finish. Then open the created project, using the Vivado GUI. Open the Hardware Manager, and load the firmware on the board, to verify its functionally.

Optionally, you can try to implement this last step directly in the tcl script.

TIP: Have a look at the tcl console, when programming the FPGA from the graphic interface.