Quartus II Design Flow: Basic Steps Summary

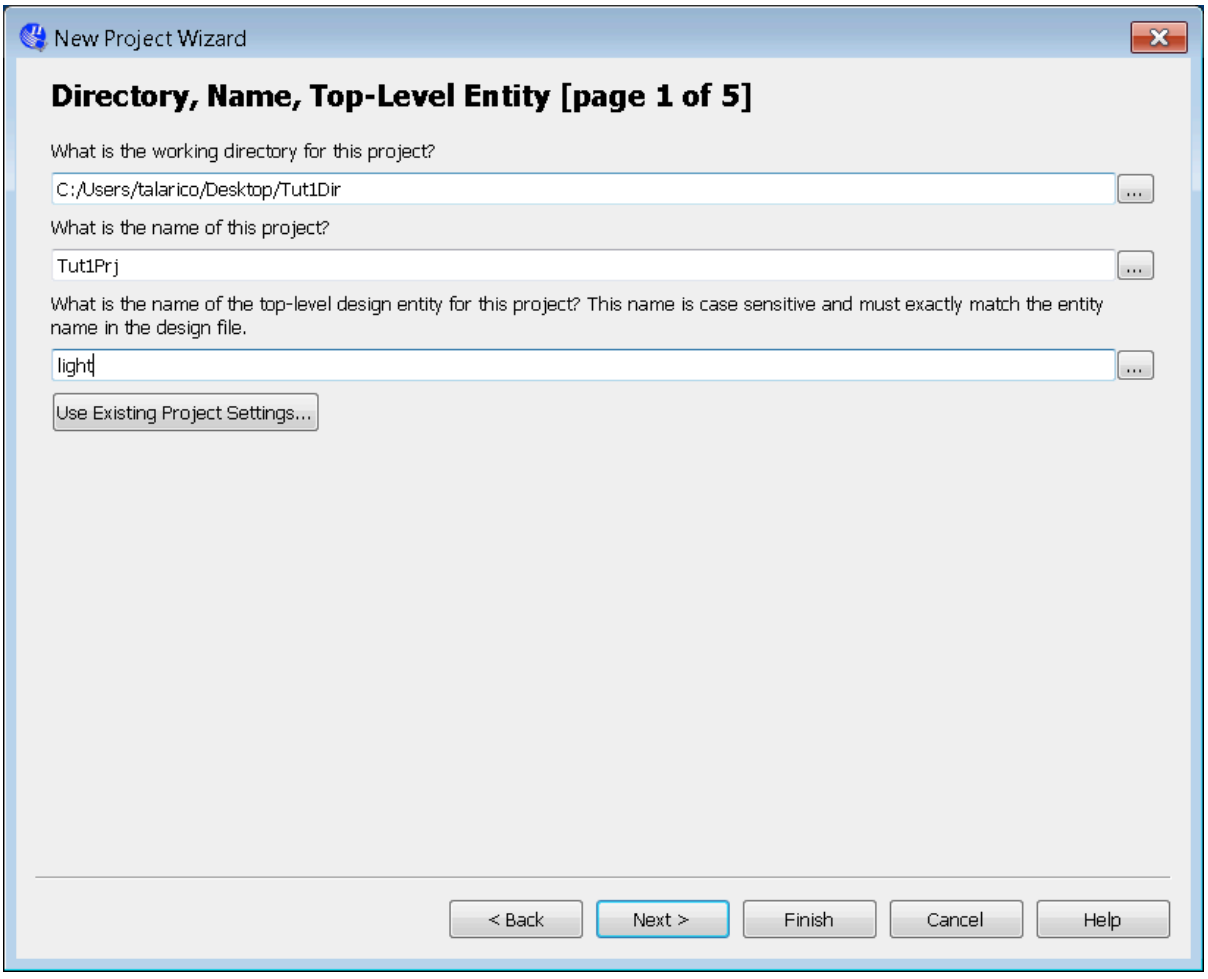
1. Files

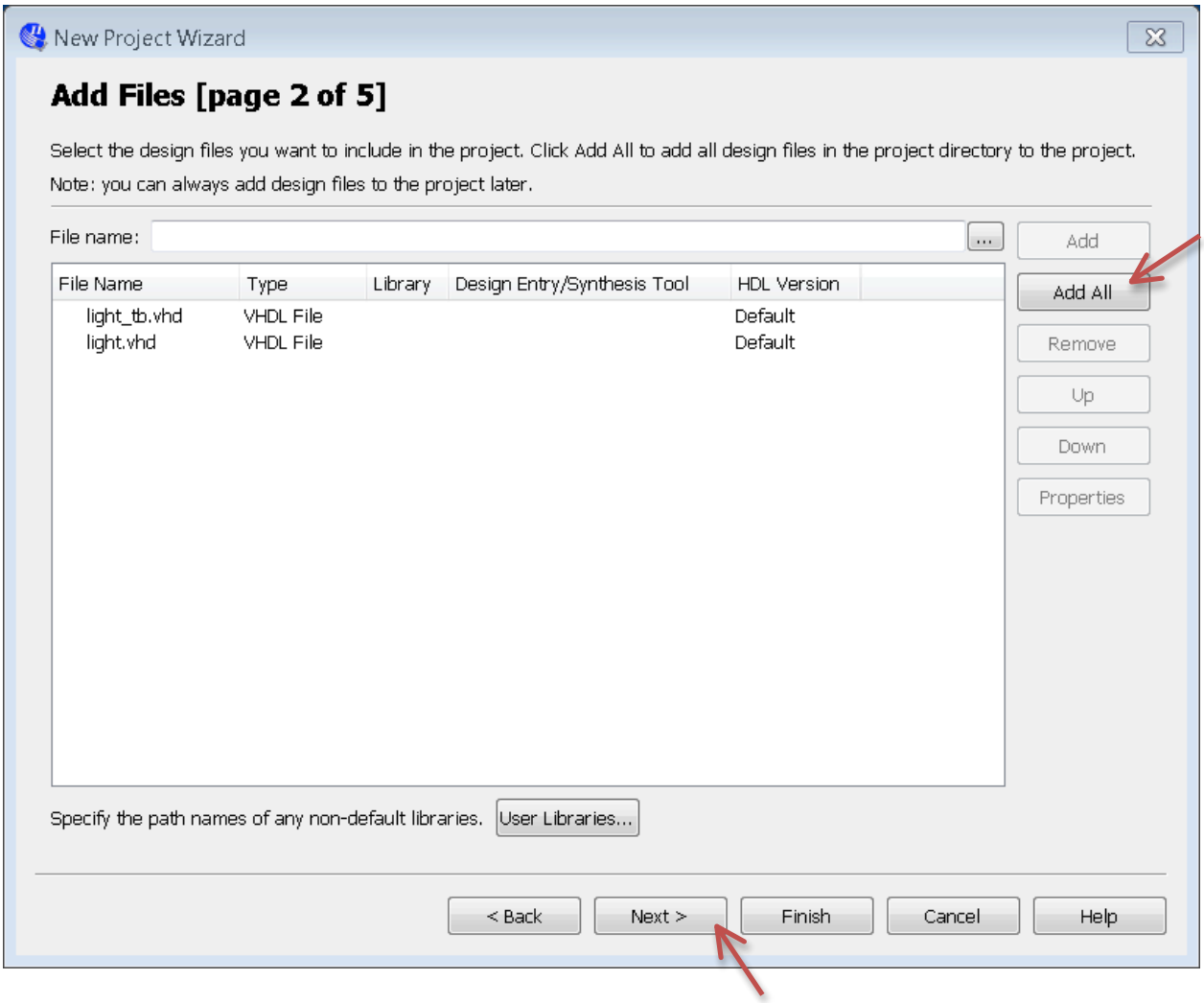
```
$ pwd
/Users/talarico/Desktop/Tut1Dir
$ ls -1
light.csv
light.vhd
light_tb.vhd
```

2. Create Project

* Project Name and Directory
* Name of the top-level design **entity**
* Project Files and Libraries
* Target Device Family and device
* EDA Tool Setting

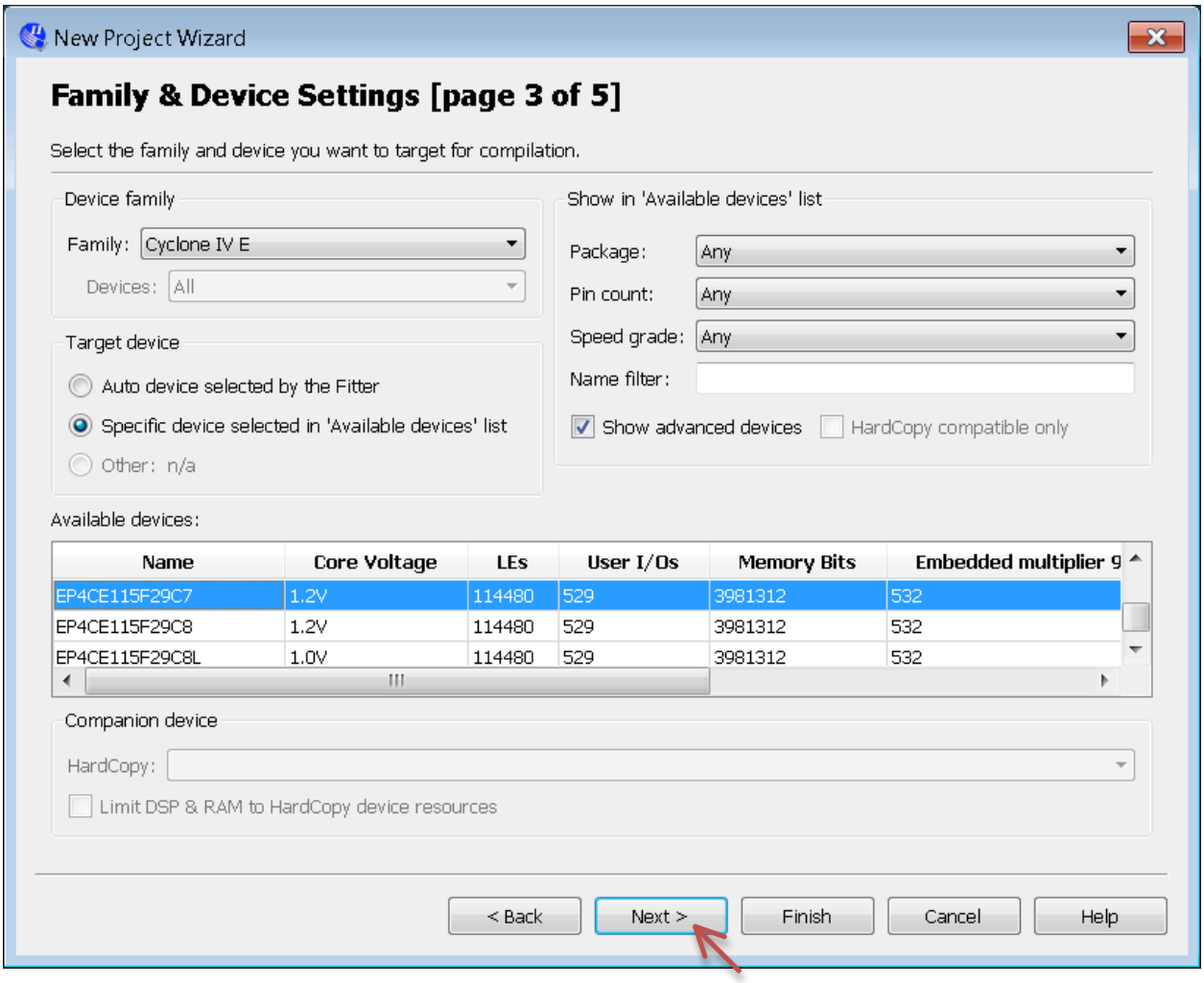
File > New Project Wizard





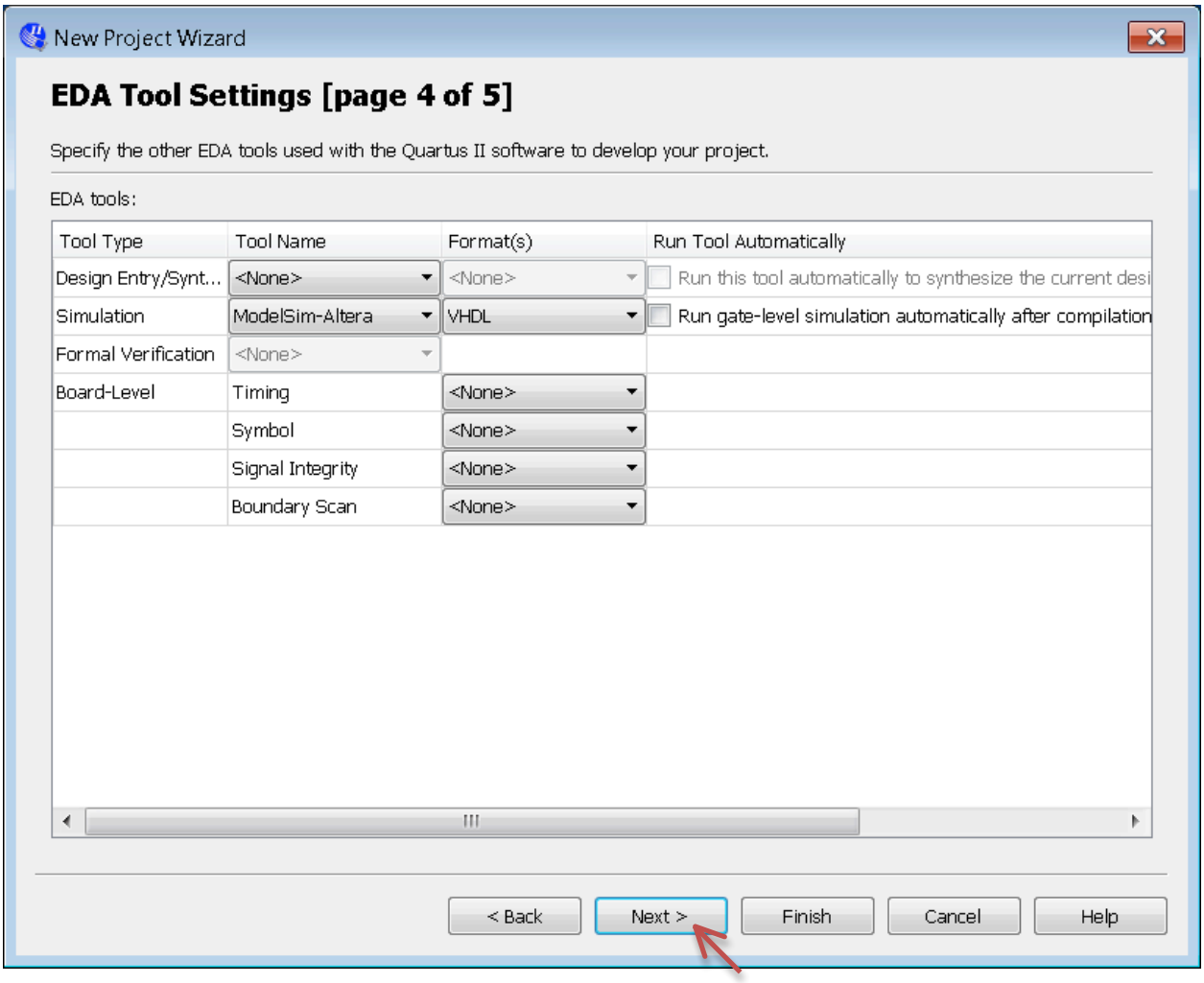
To modify a project:
Project > Add/Remove Files in Project

To edit a new VHDL File with Quartus Text Editor:
File > New > (Design Files > VHDL File)

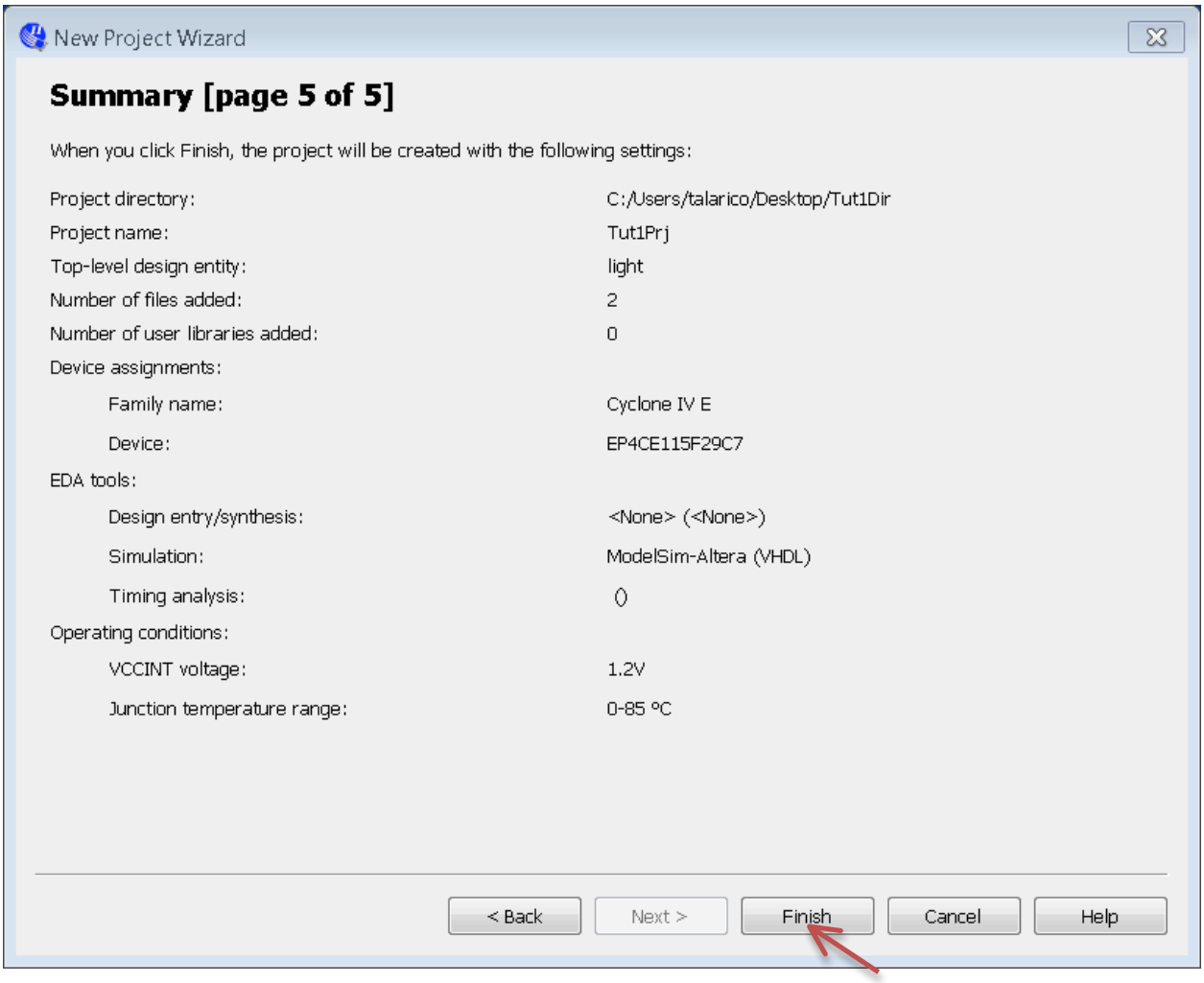


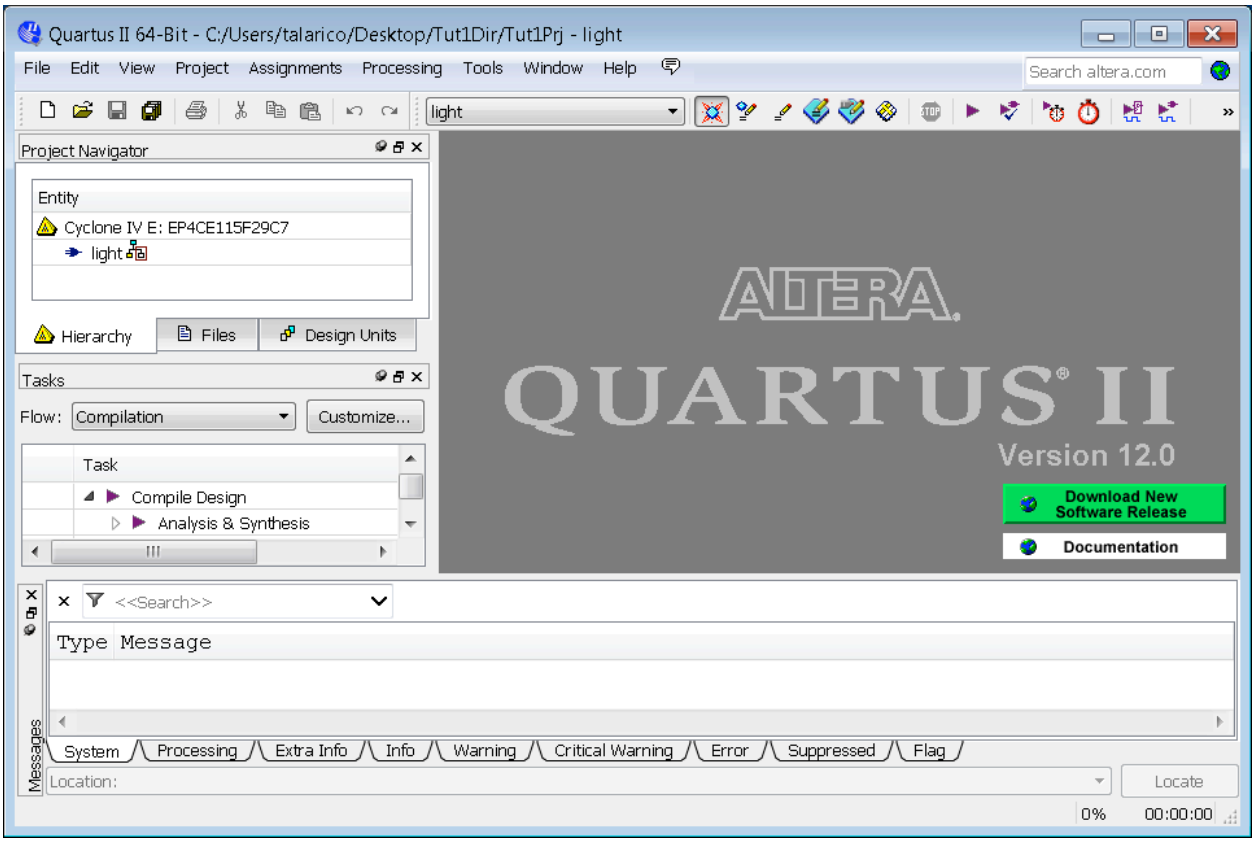
Family: Cyclone IV E
Device: EP4CE115F29C7

Assignments > Device



Assignments > Settings


New Project Wizard
Summary [page 5 of 5]
When you click Finish, the project will be created with the following settings:
Project directory: C:/Users/talarico/Desktop/Tut1Dir
Project name: Tut1Prj
Top-level design entity: light
Number of files added: 2
Number of user libraries added: 0
Device assignments:
Family name: Cyclone IV E
Device: EP4CE115F29C7
EDA tools:
Design entry/synthesis: <None> (<None>)
Simulation: ModelSim-Altera (VHDL)
Timing analysis: ()
Operating conditions:
VCCINT voltage: 1.2V
Junction temperature range: 0-85 °C
< Back
Next >
Finish
Cancel
Help



3. Files

```
$ ls -1p
Tut1Prj.qpf
db/
light.csv
light.qsf
light.vhd
light_tb.vhd
```

Quartus project file

```
$ cat Tut1Prj.qpf
```

QUARTUS_VERSION = "13.1"
DATE = "09:45:00 April 29, 2014"
# Revisions
PROJECT_REVISION = "light"

Quartus settings file

```
$ cat light.qsf
```

set_global_assignment -name FAMILY "Cyclone IV E"
set_global_assignment -name DEVICE EP4CE115F29C7
set_global_assignment -name TOP_LEVEL_ENTITY light
set_global_assignment -name ORIGINAL_QUARTUS_VERSION 13.1
set_global_assignment -name PROJECT_CREATION_TIME_DATE "09:45:00 APRIL 29, 2014"
set_global_assignment -name LAST_QUARTUS_VERSION 13.1

```
set_global_assignment -name VHDL_FILE light_tb.vhd
set_global_assignment -name VHDL_FILE light.vhd
set_global_assignment -name PROJECT_OUTPUT_DIRECTORY output_files
set_global_assignment -name MIN_CORE_JUNCTION_TEMP 0
set_global_assignment -name MAX_CORE_JUNCTION_TEMP 85
set_global_assignment -name ERROR_CHECK_FREQUENCY_DIVISOR 1
set_global_assignment -name NOMINAL_CORE_SUPPLY_VOLTAGE 1.2V
set_global_assignment -name EDA_SIMULATION_TOOL "ModelSim-Altera (VHDL)"
set_global_assignment -name EDA_NETLIST_WRITER_OUTPUT_DIR simulation/modelsim -section_id eda_simulation
set_global_assignment -name EDA_OUTPUT_DATA_FORMAT VHDL -section_id eda_simulation
```

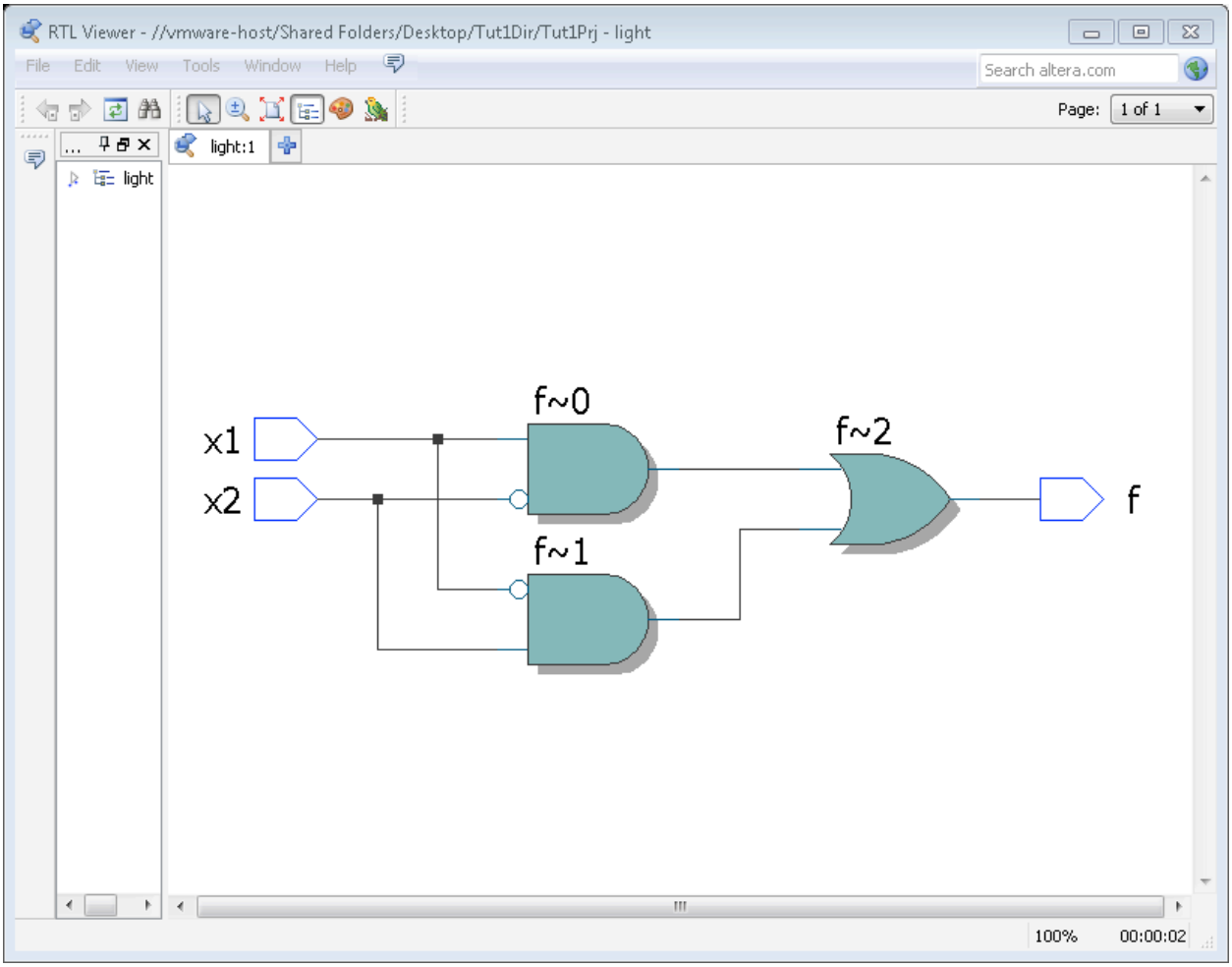
4. Start Compiling the Design

Processing > Analyze Current File > Start > Analysis & Elaboration

Check syntax & translate HDL into generic Boolean eqns.

Make sure to verify all Info/Warnings/Errors

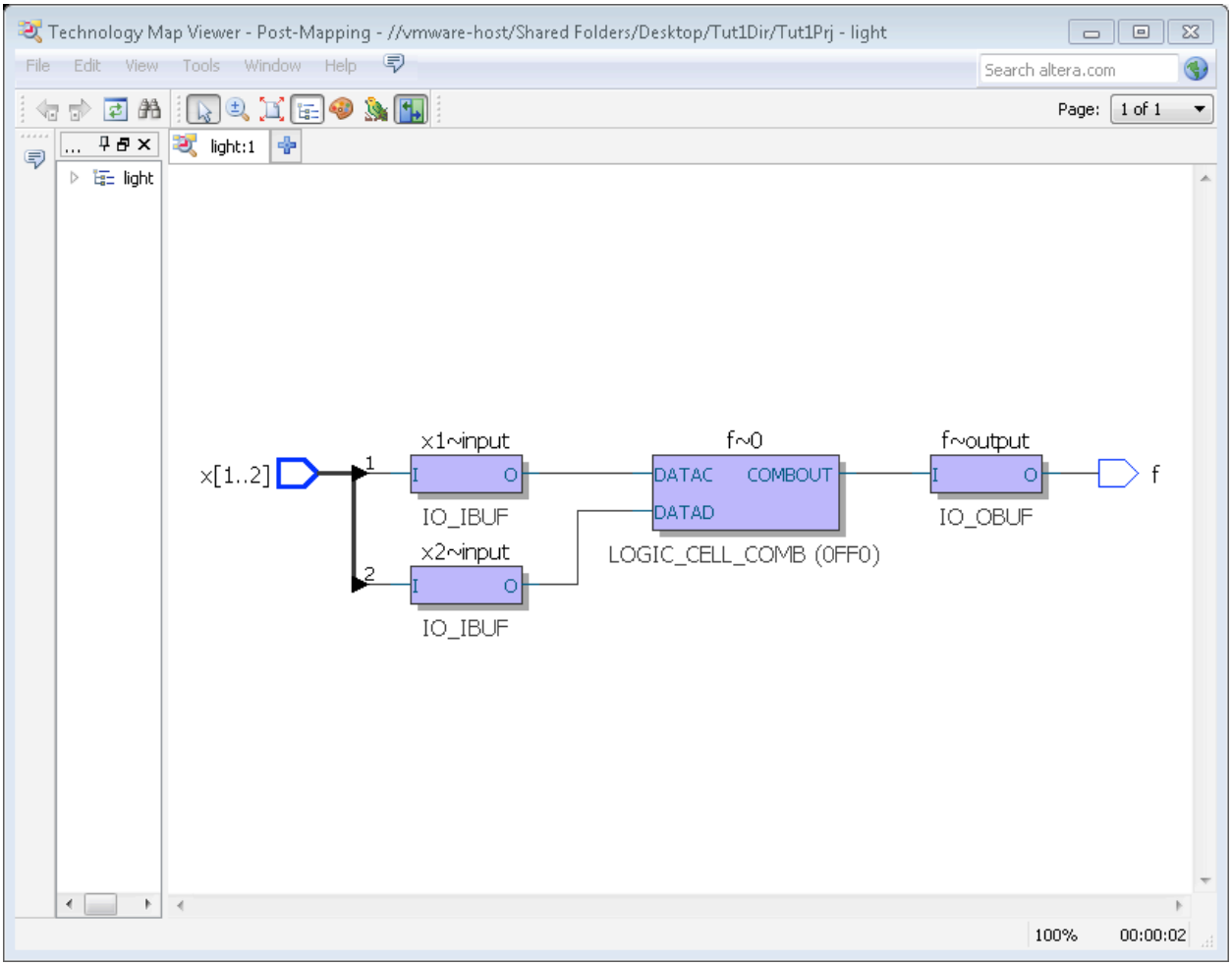
Tools > Netlist Viewers > RTL viewer



Processing > Analyze Current File > Start > Analysis & Synthesis

Synthesize HDL into target Technology

Tools > Netlist Viewers > Technology Map Viewer (Post Mapping)

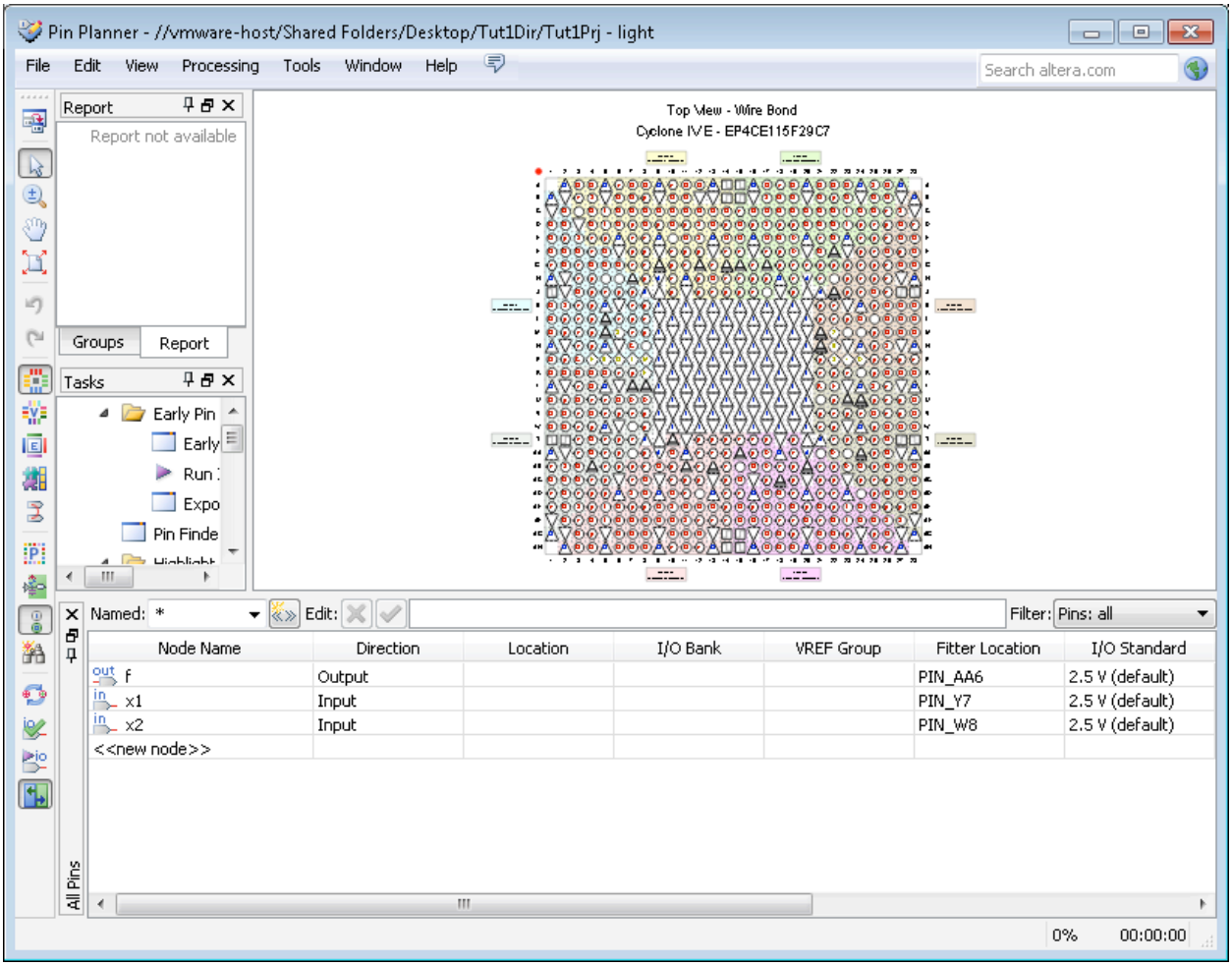


4. Files

```
$ ls -p1
Tut1Prj.qpf
db/
incremental_db/
light.csv
light.qsf
light.vhd
light_tb.vhd
output_files/
simulation/
```

5. Pin Assignments

Assignments > Pin Planner

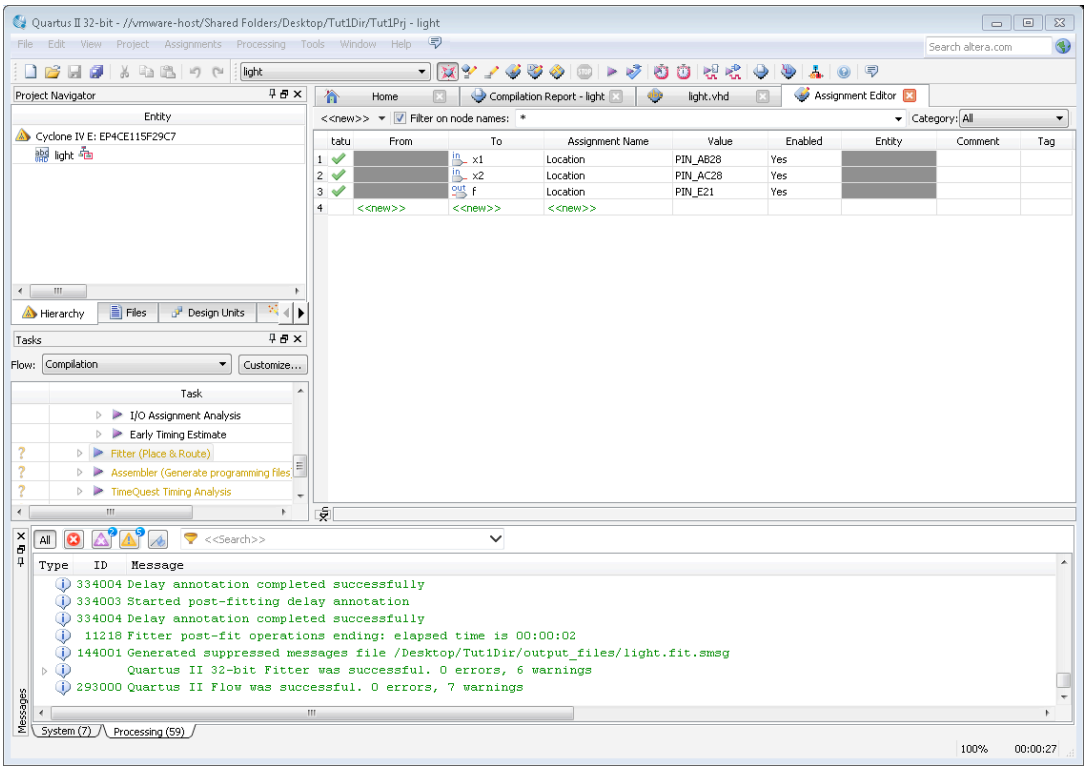


If the Pin Planner has issues loading automatically the pinout assignments saved in light.csv put the assignments in a text file light.txt and load it manually.

```
$ cat light.txt
# Pin Assignments

To, Direction, Location
x1, input, PIN_AB28
x2, input, PIN_AC28
f, output, PIN_E21
```

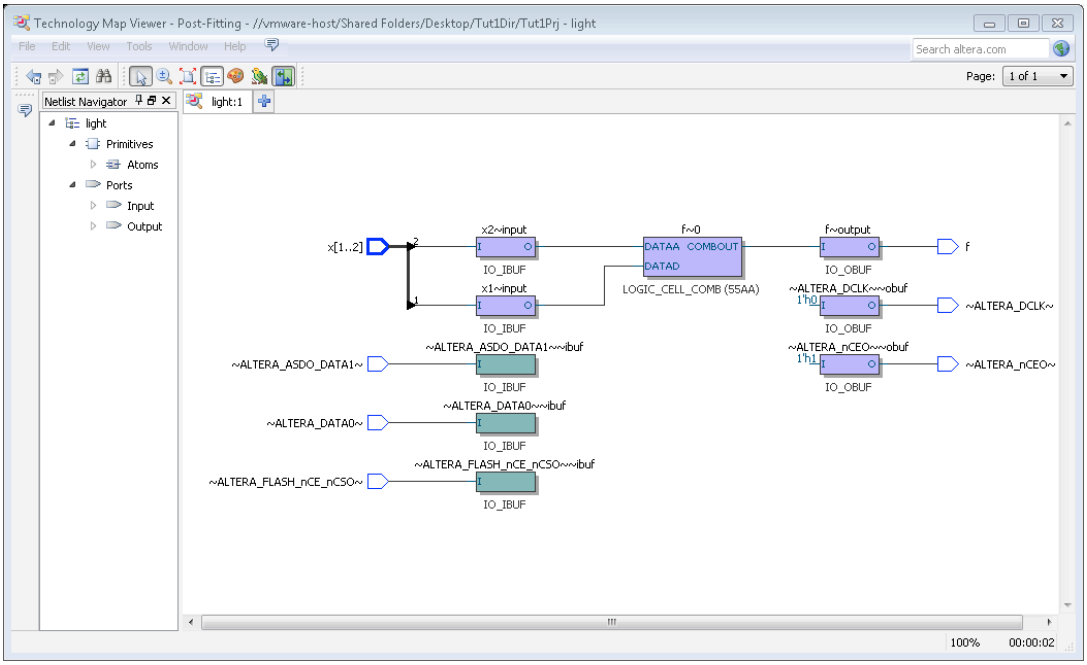
Assignments > Input Assignments (file name light.txt)



6. Continue Compiling the Design

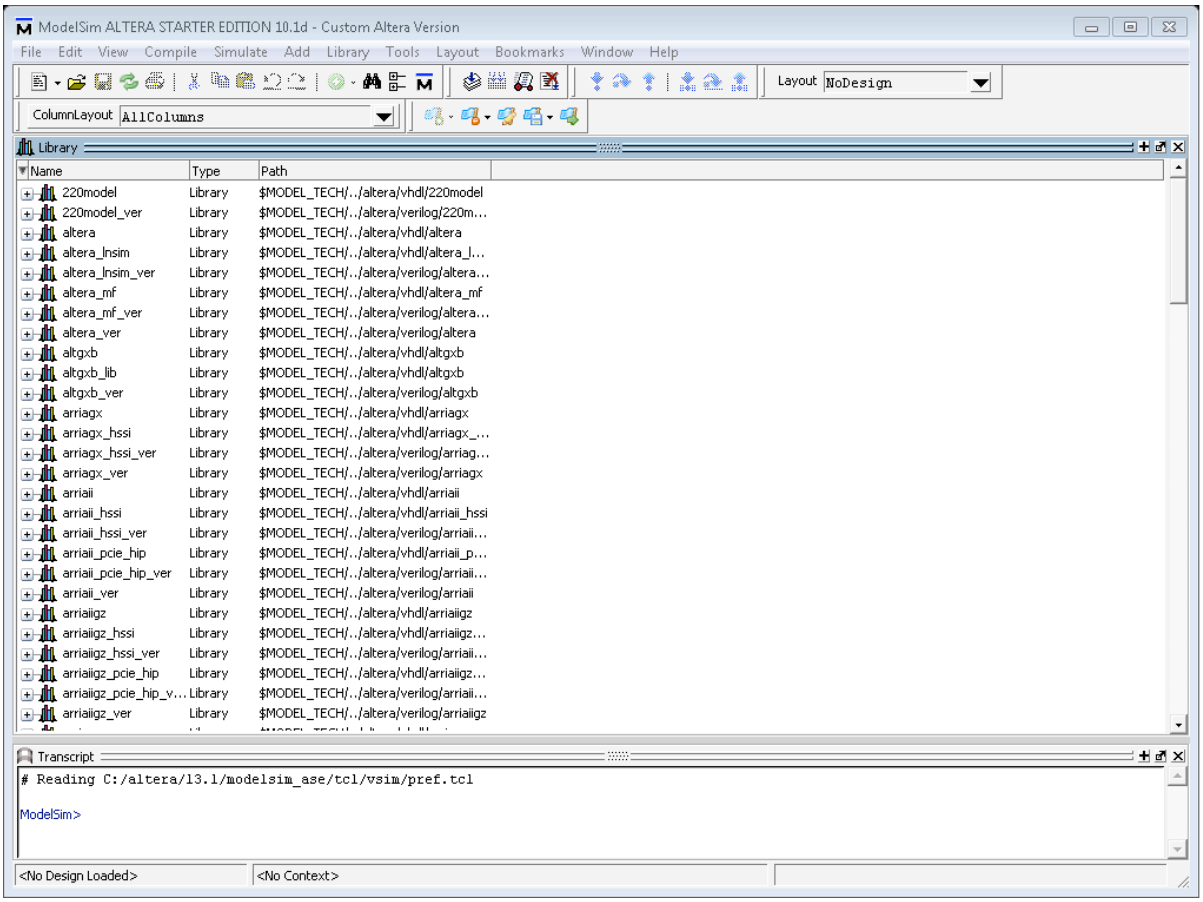
Processing > Start > Start Fitter P & R

Tools > Netlist Viewers > Technology Map Viewer (Post Fitting)



7. Simulation

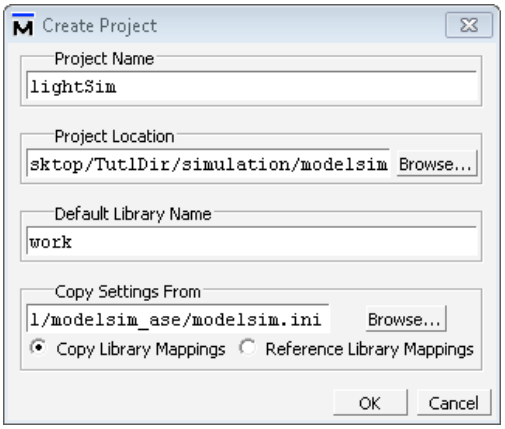
Invoke Modelsim



File > New > Project

Project name: lightSim

Project Location: C:/Users/talarico/Desktop/Tut1Dir/simulation/modelsim

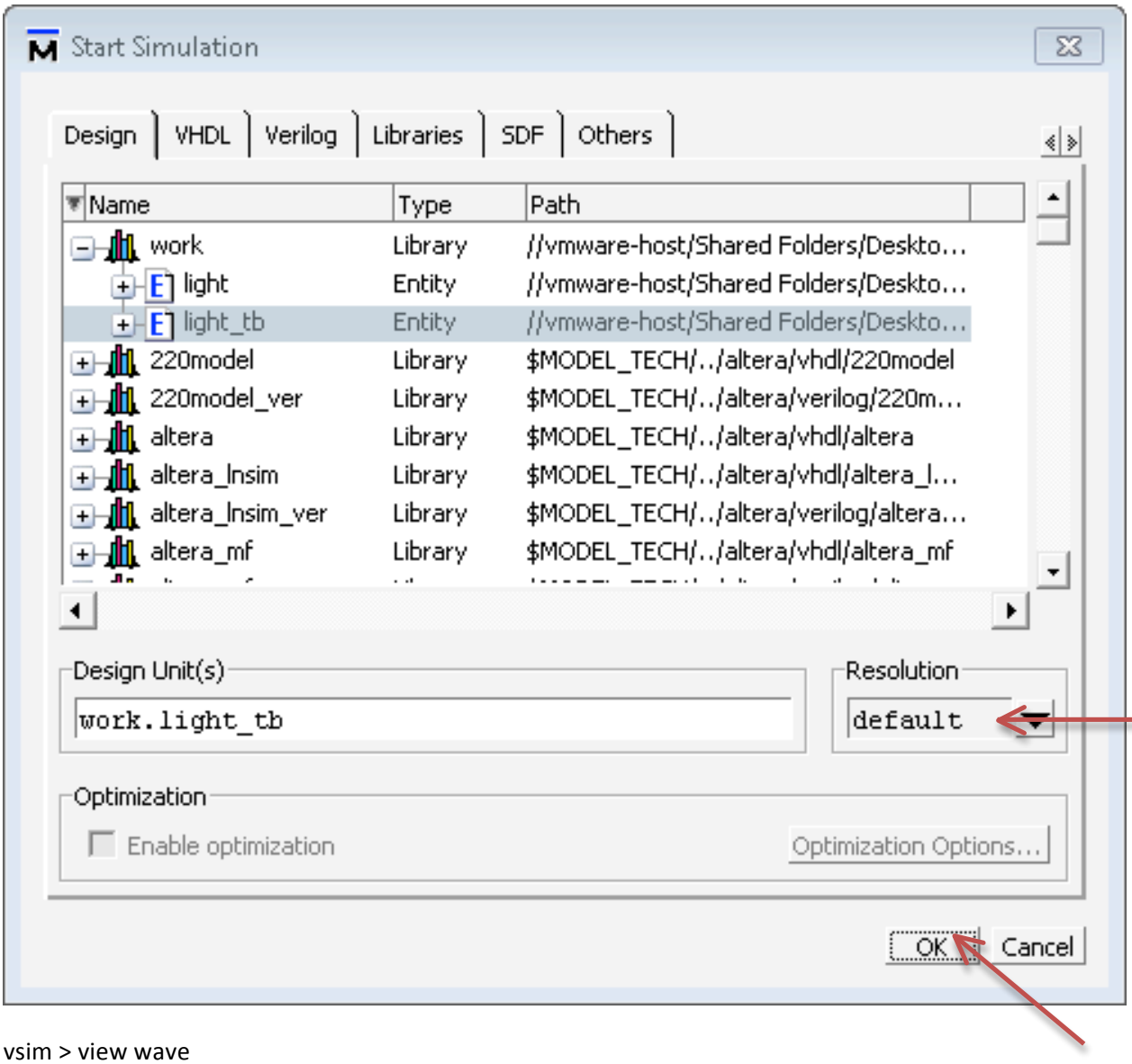


Project > Add to project > Existing File:
C:/Users/talarico/Desktop/Tut1Dir/light.vhd
C:/Users/talarico/Desktop/Tut1Dir/light_tb.vhd

Project > Add to project > New File

Compile > compile all

Simulate > Start Simulation
+ work
  + light
  + light_tb



vsim > view wave

vsim > add wave sim: /light_tb/*

vsim > run –all  The simulation runs forever !!!

Simulate > break

To automatically end the simulator use assertion in the testbench code.

$ cat light_tb.vhd

```
library ieee;
use ieee.std_logic_1164.all;
entity light_tb is
end light_tb;

architecture beh of light_tb is
component light
port(
  x1 : in std_logic;
  x2 : in std_logic;
  f  : out std_logic);
end component;

signal x1_sig : std_logic := '0';
signal x2_sig : std_logic := '0';
signal f_sig : std_logic;

begin
--instantiate the component
light_inst: light port map(
x1 => x1_sig,
x2 => x2_sig,
f => f_sig);

-- ALTERA's testbench processes
-- process
--   begin
--   x1_sig <= '0';
--   wait for 20 ns;
--   x1_sig <= '1';
--   wait for 20 ns;
-- end process;

-- process
--   begin
--   x2_sig <= '0';
--   wait for 40 ns;
--   x2_sig <= '1';
--   wait for 20 ns;
-- end process;
-- END of ALTERA's testbench processes

-- Claudio's testbench process
process
  begin
    x1_sig <= '0';
    x2_sig <= '0';
    wait for 20 ns;
    x1_sig <= '0';
```

```
    x2_sig <= '1';
    wait for 20 ns;
    x1_sig <= '1';
    x2_sig <= '0';
    wait for 20 ns;
    x1_sig <= '1';
    x2_sig <= '1';
    wait for 40 ns;

    assert false
    report "End of Testbench"
    severity failure;

end process;
-- END Claudio's of testbench process

end beh;
```

Simulate > Runtime options > Message Severity

vsim > restart –force
vsim > run –all
vsim > wave zoomout
vsim > wave zoomfull

8. Finish Compiling the Design

Processing > Start > Assembler  Generating Programming files (light.sof and light.jdi)

sof = SRAM object file
jdi = JTAG Debugging Information

9. View Reports

Make sure to read carefully the reports generated at each step of the design process.
The reports can be accessed through the Quartus' workspace window or directly in the output_files folder.

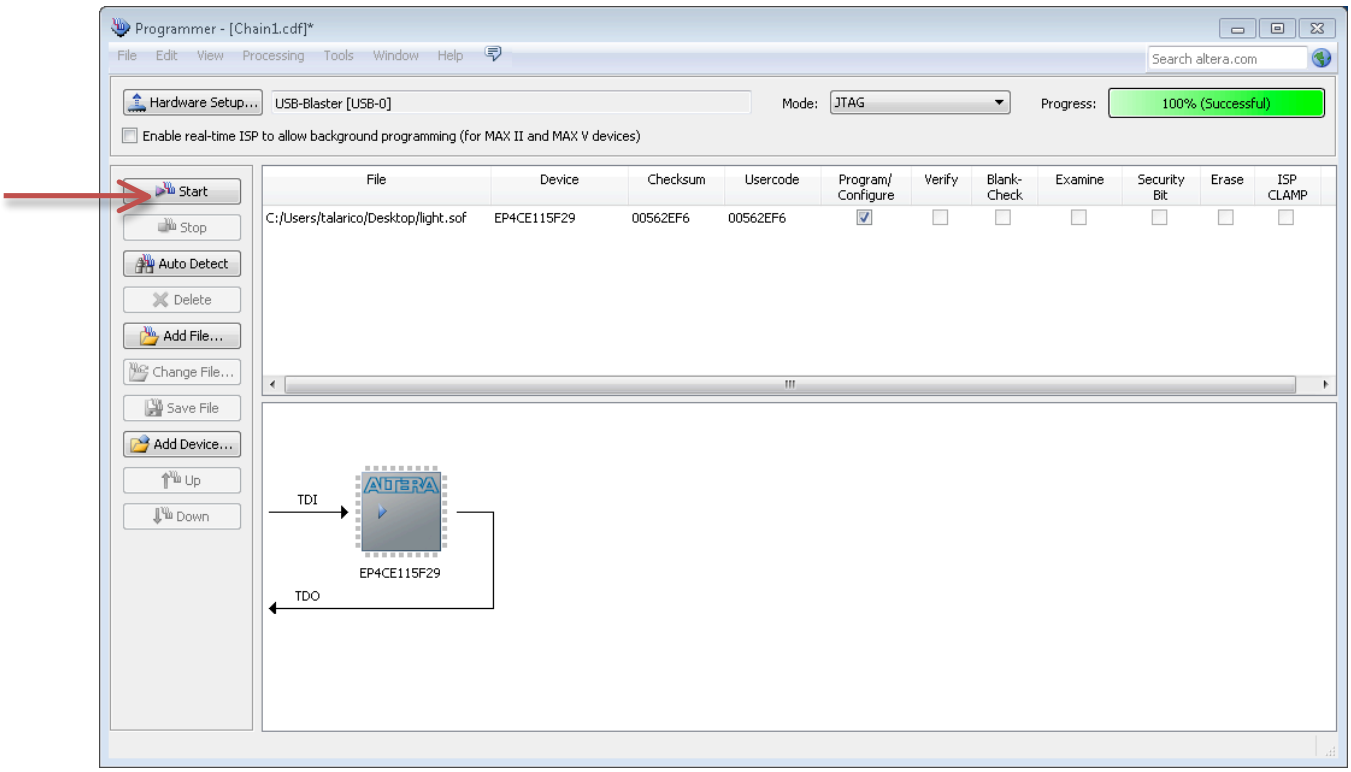
10. Programming and Configuring the FPGA

Before Programming the FPGA make sure to set the board in RUN mode (JTAG mode).

Tools > Programmer

Edit > Add File (C:/Users/talarico/Desktop/output_files/light.sof)

The hardware Setup must be USB-Blaster [USB-0]



11. Testing the Circuit on the Development Board