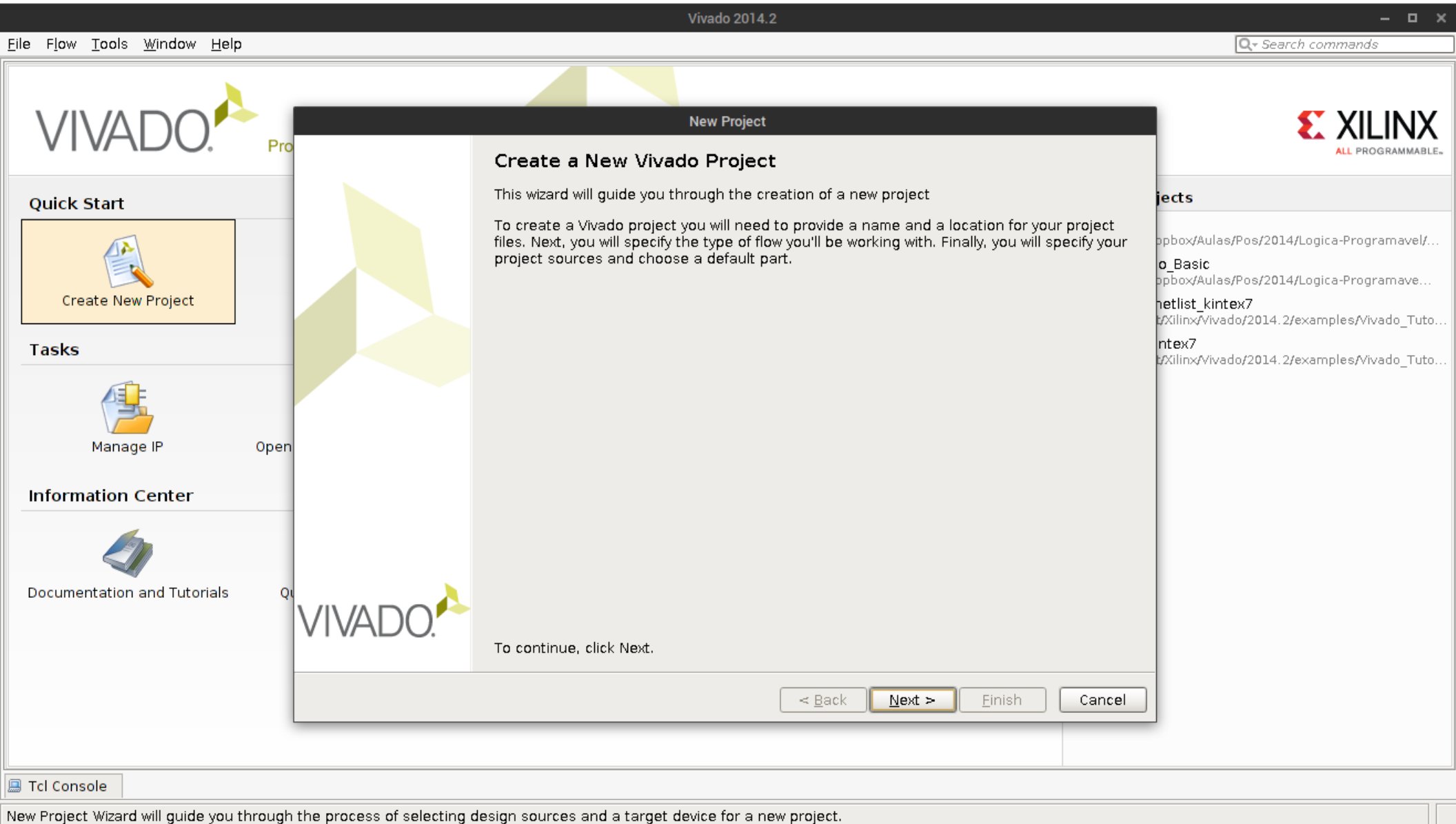


Tutorial Vivado

Introdução a ferramenta Xilinx Vivado
Rafael Corsi – Mauá


Criando um novo projeto



Nome do projeto

New Project

Project Name
Enter a name for your project and specify a directory where the project data files will be stored



Project name:

Project location:


☒ Create project subdirectory

Project will be created at: /home/rafa/aula1

Tipo do projeto

New Project

Project Type
Specify the type of project to create.



☒ **RTL Project**
You will be able to add sources, create block designs in IP Integrator, generate IP, run RTL analysis, synthesis, implementation, design planning and analysis.
☐ Do not specify sources at this time

☐ **Post-synthesis Project**
You will be able to add sources, view device resources, run design analysis, planning and implementation.
☐ Do not specify sources at this time

☐ **I/O Planning Project**
Do not specify design sources. You will be able to view part/package resources.

☐ **Imported Project**
Create a Vivado project from a Synplify, XST or ISE Project File.

< Back

Next >

Finish

Cancel

Adicionando arquivos fontes

New Project

Add Sources

Specify HDL and netlist files, or directories containing HDL and netlist files, to add to your project. Create a new source file on disk and add it to your project. You can also add and create sources later.

Index	Name	Library	HDL Source For	Location
-------	------	---------	----------------	----------

Add Files...

Add Directories...

Create File...

☐ Scan and add RTL include files into project

☒ Copy sources into project

☒ Add sources from subdirectories

Target language: VHDL

Simulator language: Mixed

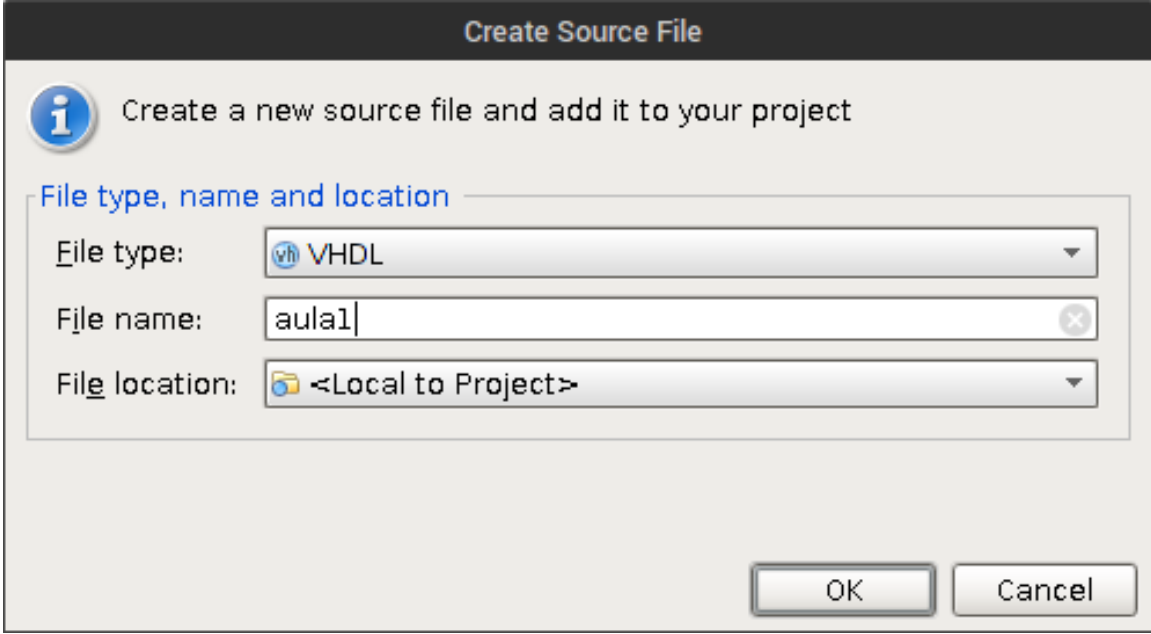
< Back

Next >

Finish


Cancel

Nome e tipo do arquivo, repare nos diferentes tipos que podemos escolher.





The image shows a 'Create Source File' dialog box. It has a title bar with the text 'Create Source File'. Below the title bar is an information icon (a lowercase 'i' in a blue circle) followed by the text 'Create a new source file and add it to your project'. Below this is a section titled 'File type, name and location' in blue text. This section contains three input fields: 'File type:' with a dropdown menu showing 'VHDL' and a small icon; 'File name:' with a text box containing 'aula1' and a clear button (an 'x' in a circle); and 'File location:' with a dropdown menu showing '<Local to Project>' and a folder icon. At the bottom right of the dialog are two buttons: 'OK' and 'Cancel'.


Create Source File

 Create a new source file and add it to your project

File type, name and location

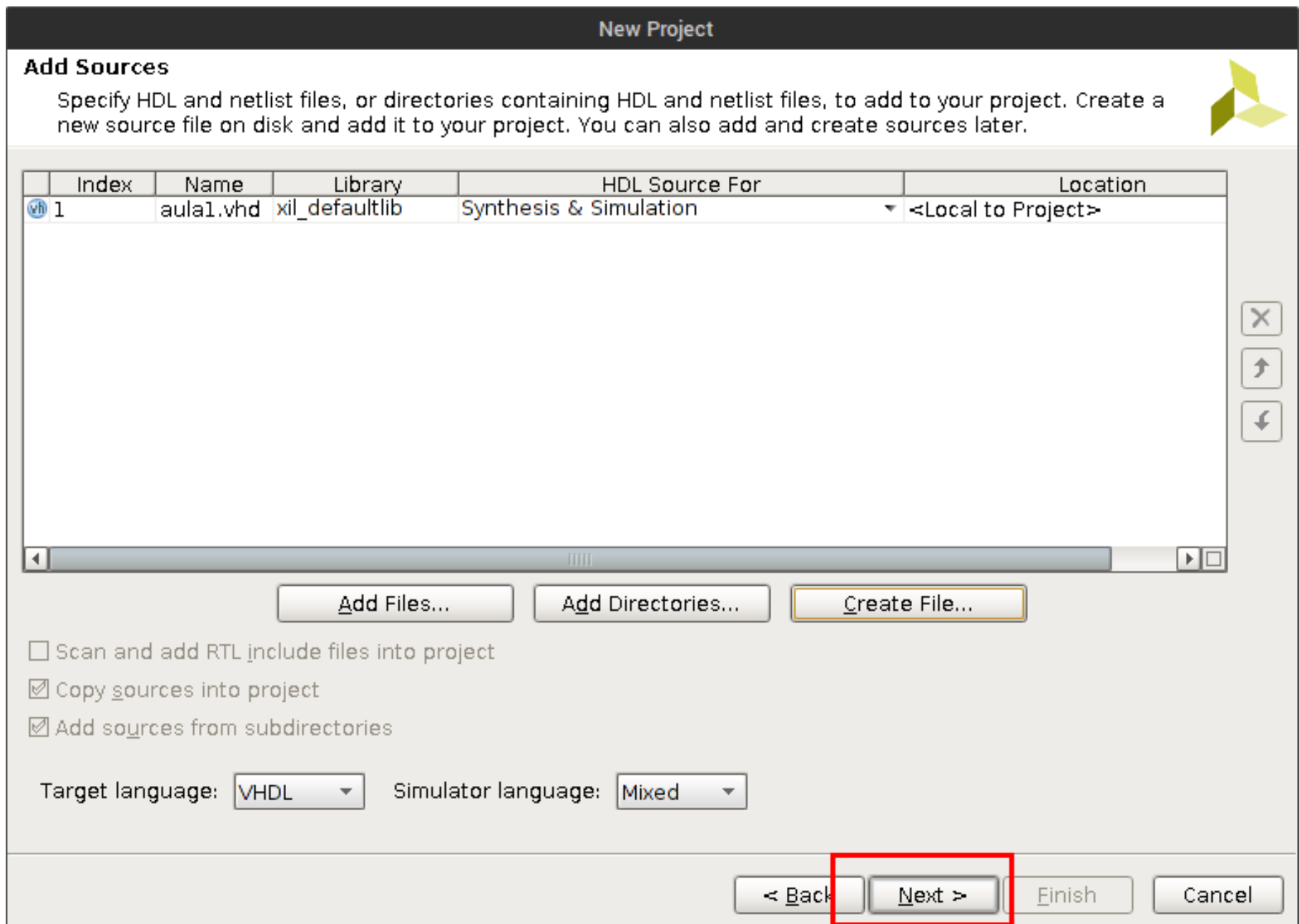
File type:  VHDL

File name: aula1 

File location:  <Local to Project>

OK Cancel

Verificamos que o novo arquivo foi automaticamente adicionado ao projeto



New Project

Add Sources

Specify HDL and netlist files, or directories containing HDL and netlist files, to add to your project. Create a new source file on disk and add it to your project. You can also add and create sources later.

Index	Name	Library	HDL Source For	Location
1	aula1.vhd	xil_defaultlib	Synthesis & Simulation	<Local to Project>


☐ Scan and add RTL include files into project
☒ Copy sources into project
☒ Add sources from subdirectories


Target language: Simulator language:




Adicionando arquivos de configuração (XCD)



New Project

Add Constraints (optional)
Specify or create constraint files for physical and timing constraints.




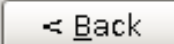


Constraint File	Location
 io_basicos.xdc	<Local to Project>

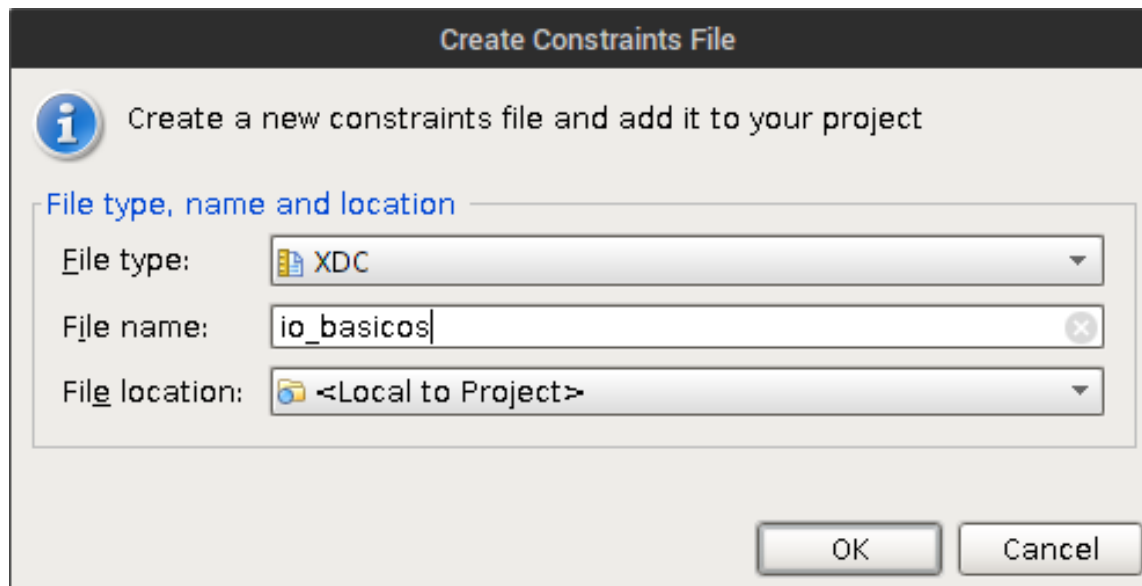




☒ Copy constraints files into project



Nome do novo arquivo, repare nos tipos.



Selecionando a pastilha

New Project

Default Part

Choose a default Xilinx part or board for your project. This can be changed later.

Specify **Filter**

Parts **Boards**


Product category **All** Package **All**

Family **All** Speed grade **All**

Sub-Family **All** Temp grade **All**

Reset All Filters

Search: (1 match)

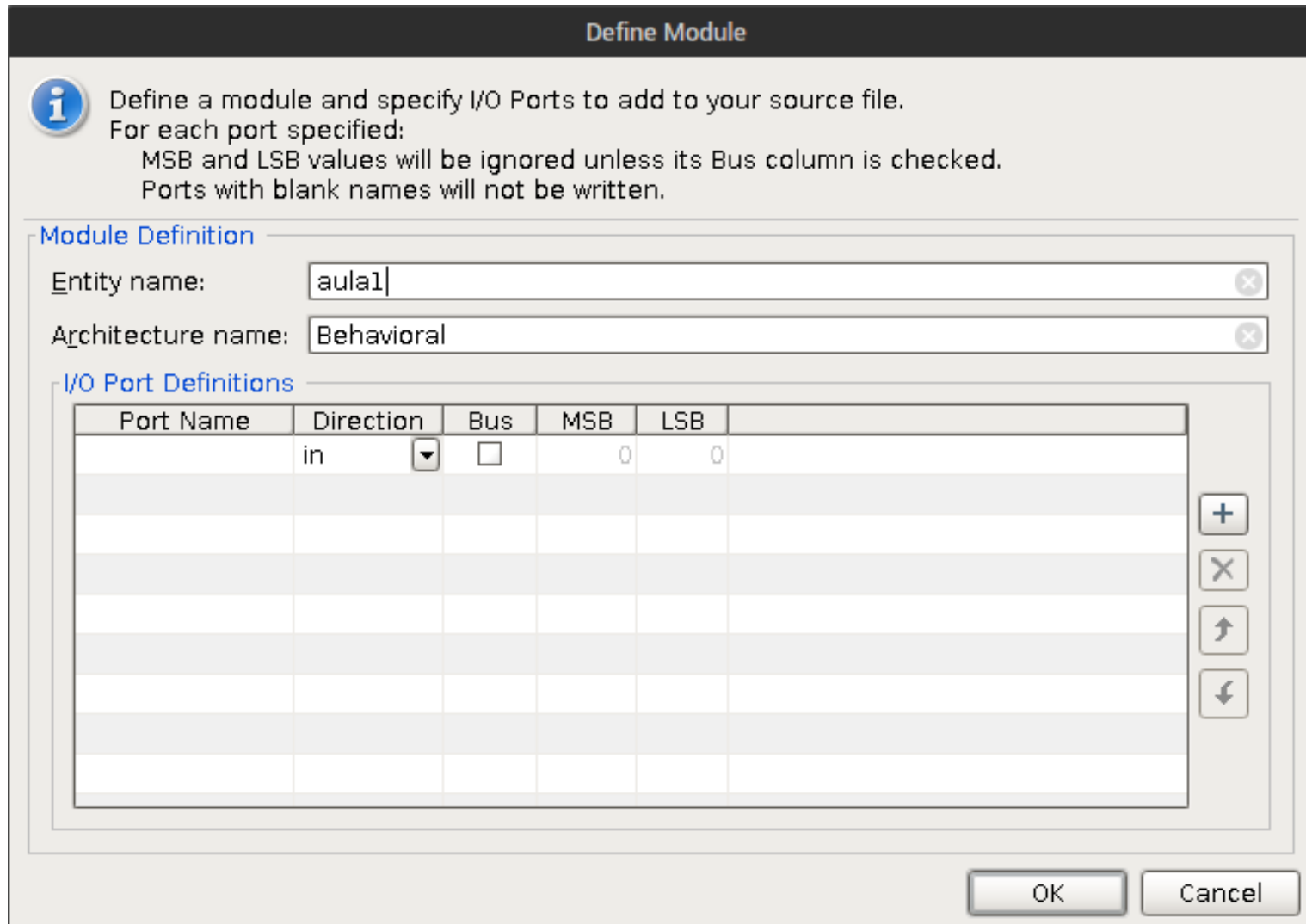
Part	I/O Pin Count	Available IOBs	LUT Elements	FlipFlops	Block RAMs	DSPs	Gb Transceivers	GTPE2 Transceiver
 xc7a100tcsg324-1	324	210	63400	126800	135	240	0	0

< Back **Next >** **Finish** **Cancel**

Resumo do projeto




O Viado fornece uma ferramenta para início rápido de projetos, aqui podemos configurar as entradas e saídas.



The image shows a 'Define Module' dialog box. It has a title bar 'Define Module'. Below the title bar is an information icon and text: 'Define a module and specify I/O Ports to add to your source file. For each port specified: MSB and LSB values will be ignored unless its Bus column is checked. Ports with blank names will not be written.' Below this is a 'Module Definition' section with two text boxes: 'Entity name:' containing 'aula1' and 'Architecture name:' containing 'Behavioral'. Below that is an 'I/O Port Definitions' section containing a table with 6 columns: 'Port Name', 'Direction', 'Bus', 'MSB', 'LSB', and an empty column. The first row has 'in' in the 'Port Name' column, a dropdown arrow in 'Direction', an unchecked checkbox in 'Bus', and '0' in both 'MSB' and 'LSB'. To the right of the table are four buttons: '+', 'X', '↑', and '↓'. At the bottom right are 'OK' and 'Cancel' buttons.

Define Module

 Define a module and specify I/O Ports to add to your source file.
For each port specified:
MSB and LSB values will be ignored unless its Bus column is checked.
Ports with blank names will not be written.





Module Definition

Entity name:

Architecture name:

I/O Port Definitions


Port Name	Direction	Bus	MSB	LSB	
	in	<input type="checkbox"/>	0	0	

OK Cancel

Definindo 3 entradas e 3 saídas

Define Module

 Define a module and specify I/O Ports to add to your source file.
For each port specified:
MSB and LSB values will be ignored unless its Bus column is checked.
Ports with blank names will not be written.






Module Definition

Entity name:

Architecture name:

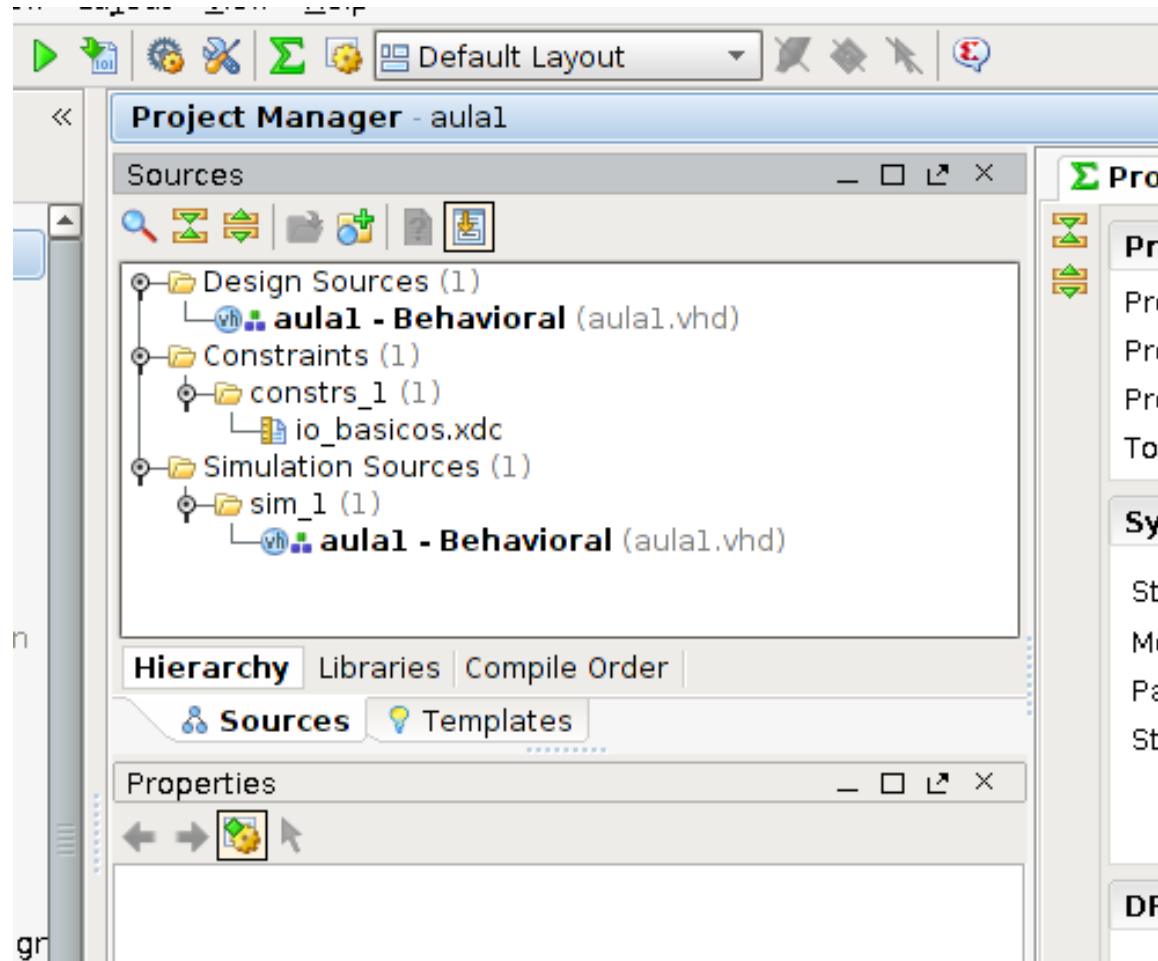
I/O Port Definitions

Port Name	Direction		Bus	MSB	LSB
led0	out	▼	<input type="checkbox"/>	0	0
led1	out	▼	<input type="checkbox"/>	0	0
led2	out	▼	<input type="checkbox"/>	0	0
chave0	in	▼	<input type="checkbox"/>	0	0
chave1	in	▼	<input type="checkbox"/>	0	0
chave2	in	▼	<input type="checkbox"/>	0	0

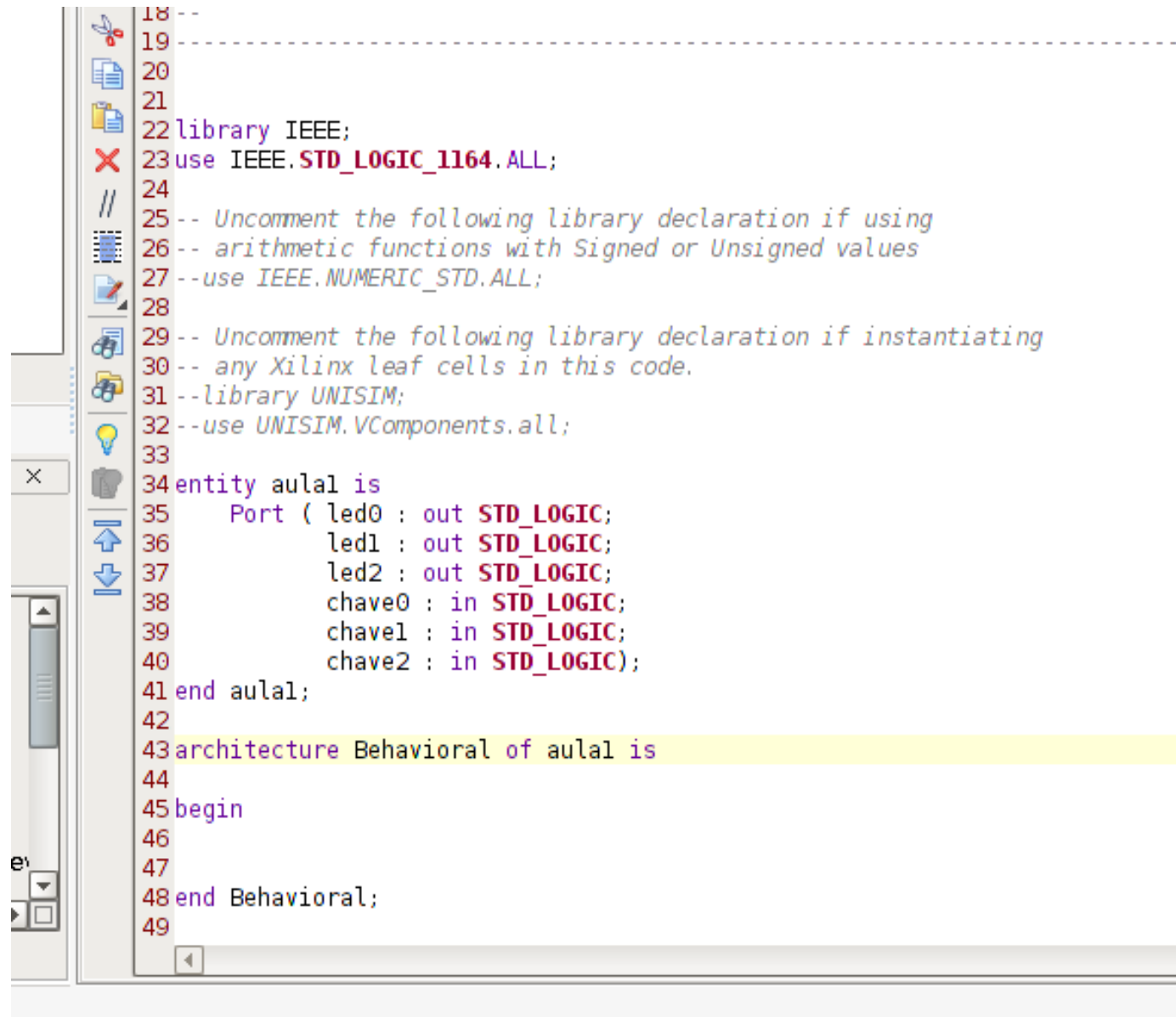


OKCancel

Resumo dos arquivos adicionados ao projeto



Vhdl criado pela ferramenta

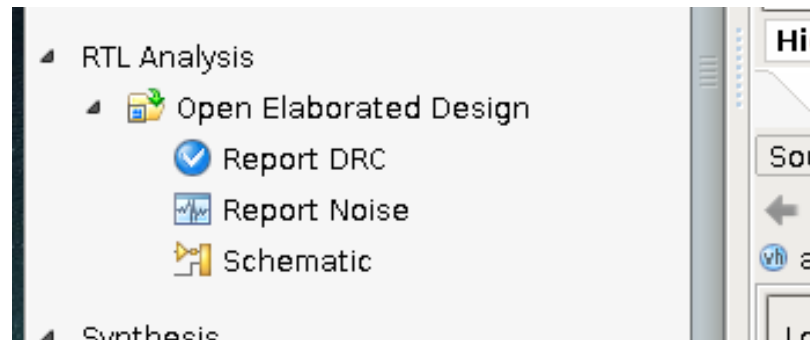


```
18 --
19 -----
20
21
22 library IEEE;
23 use IEEE.STD_LOGIC_1164.ALL;
24
25 -- Uncomment the following library declaration if using
26 -- arithmetic functions with Signed or Unsigned values
27 --use IEEE.NUMERIC_STD.ALL;
28
29 -- Uncomment the following library declaration if instantiating
30 -- any Xilinx leaf cells in this code.
31 --library UNISIM;
32 --use UNISIM.VComponents.all;
33
34 entity aula1 is
35     Port ( led0 : out STD_LOGIC;
36           led1 : out STD_LOGIC;
37           led2 : out STD_LOGIC;
38           chave0 : in STD_LOGIC;
39           chave1 : in STD_LOGIC;
40           chave2 : in STD_LOGIC);
41 end aula1;
42
43 architecture Behavioral of aula1 is
44
45 begin
46
47
48 end Behavioral;
49
```

Modificação proposta

```
21
22 library IEEE;
23 use IEEE.STD_LOGIC_1164.ALL;
24
25 -- Uncomment the following library declaration if using
26 -- arithmetic functions with Signed or Unsigned values
27 --use IEEE.NUMERIC_STD.ALL;
28
29 -- Uncomment the following library declaration if instantiating
30 -- any Xilinx leaf cells in this code.
31 --library UNISIM;
32 --use UNISIM.VComponents.all;
33
34 entity aul1 is
35     Port ( led0 : out STD_LOGIC;
36           led1 : out STD_LOGIC;
37           led2 : out STD_LOGIC;
38           chave0 : in STD_LOGIC;
39           chave1 : in STD_LOGIC;
40           chave2 : in STD_LOGIC);
41 end aul1;
42
43 architecture Behavioral of aul1 is
44
45 begin
46
47 led0 <= chave0;
48 led1 <= NOT chave1;
49 led2 <= chave2 AND chave1;
50
51
52 end Behavioral;
53
```


Vamos verificar agora o RTL desse projeto, rode o “schematic” no “RTL Analysis”



Verificamos a lógica interpretada pela ferramenta

aula1 - [/home/rafa/aula1/aula1.xpr] - Vivado 2014.2

File Edit Flow Tools Window Layout View Help

Search commands

Ready

Flow Navigator

- Project Settings
- Add Sources
- IP Catalog
- IP Integrator
 - Create Block Design
 - Open Block Design
 - Generate Block Design
- Simulation
 - Simulation Settings
 - Run Simulation
- RTL Analysis**
 - Elaborated Design
 - Report DRC
 - Report Noise
 - Schematic
- Synthesis
 - Synthesis Settings
 - Run Synthesis
 - Open Synthesized Design
- Implementation
 - Implementation Settings
 - Run Implementation
 - Open Implemented Design
- Program and Debug

Elaborated Design - xc7a100tcsg324-1 (active)

RTL Netlist

- aula1
 - Nets (5)
 - Leaf Cells (2)

Sources RTL Netlist

Source File Properties

Design Runs

Name	Constraints	WNS	TNS	WHS	THS	TPWS	Failed Routes	LUT	FF	BRAM	DSP	St
synth_1	constrs_1											
impl_1	constrs_1											

RTL Schematic

2 Cells 6 I/O Ports 5 Nets

chave0 led0

chave1 led1

chave2 led2

led1_i

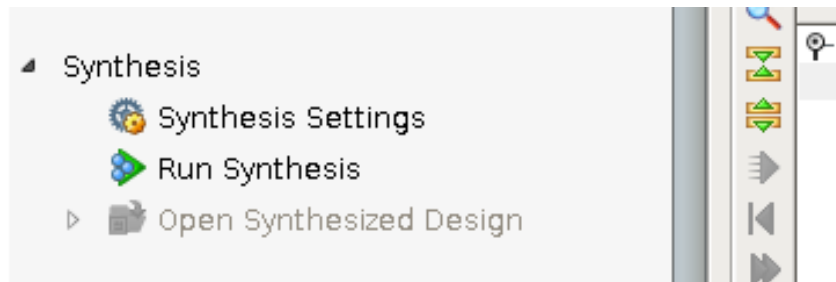
RTL_INV

led2_i

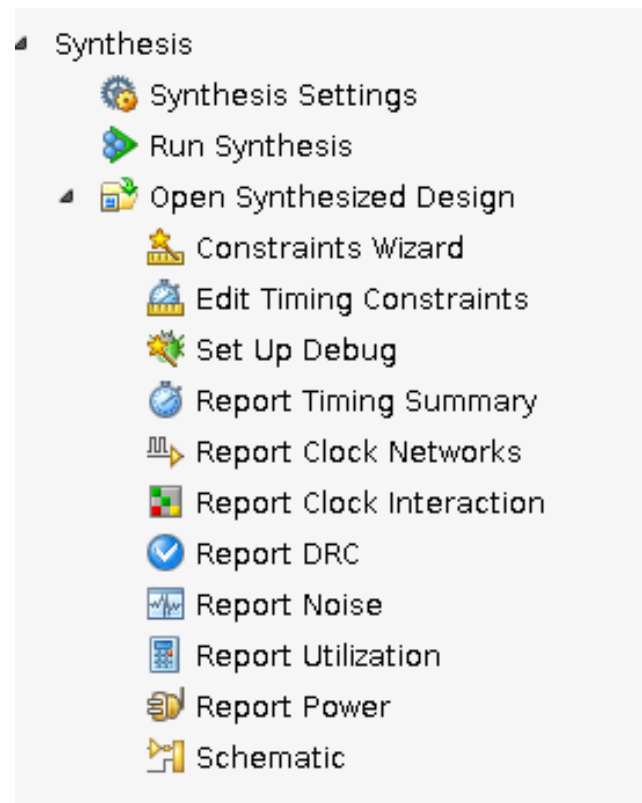
RTL_AND

Tcl Console Messages Log Reports Design Runs

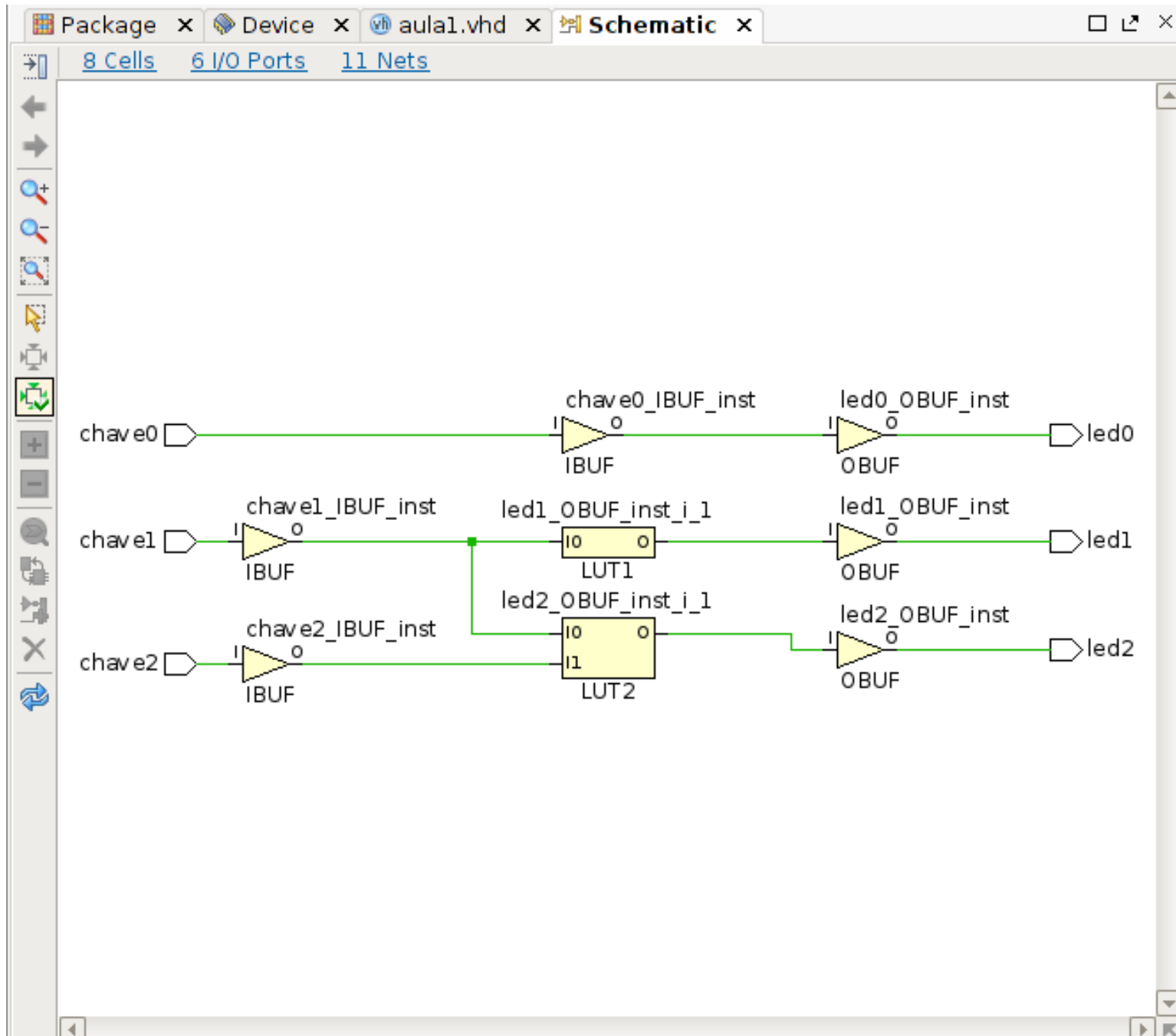
Agora podemos executar a “Synthesis” → run Synthesis.



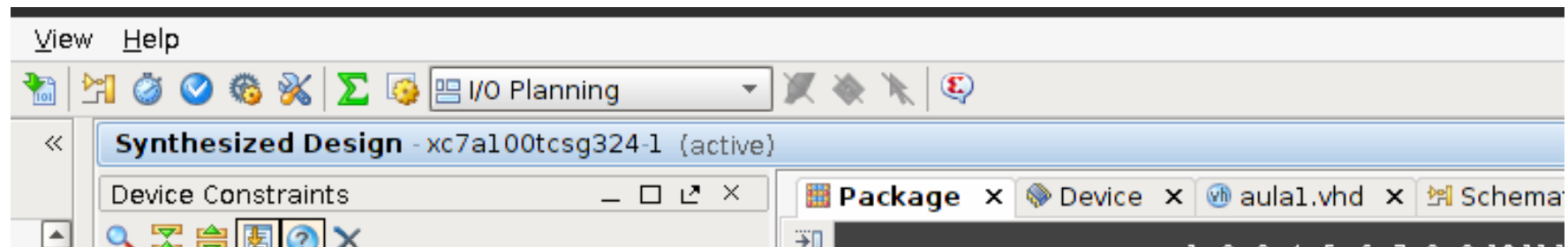
Verificamos agora o Schematic da synthesis



Aqui verificamos como a lógica RTL será implementada



Precisamos definir os pinos e tipos de sinais de cada porta da entidade aula1, mude o layout da ferramenta para “I/O Planning”



Devemos verificar essa janela no I/O Planning. Note que as portas não estão mapeadas

aula1 - [/home/rata/aula1/aula1.xpr] - Vivado 2014.2

Help Search commands

I/O Planning

Synthesis Complete

Synthesized Design - xc7a100tcs9324-1 (active)

Device Constraints

Internal VREF

- 0.0V
- NONE (5)
 - I/O Bank 14
 - I/O Bank 15
 - I/O Bank 16
 - I/O Bank 34
 - I/O Bank 35

Drop I/O banks on voltages or the "NONE" folder to set/unset Internal VREF.

Sources Netlist Device C..

Properties

Properties Clock Regions

Package x Device x aula1.vhd x Schematic x

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

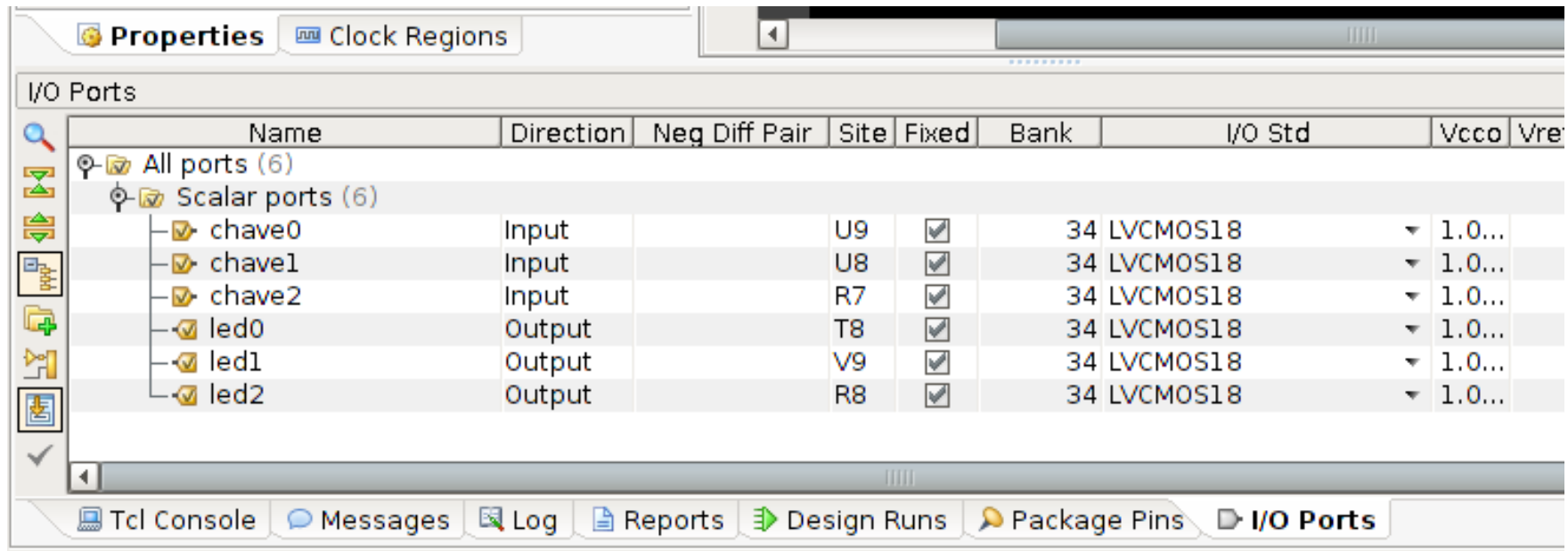
A B C D E F G H I J K L M N P R T U V

I/O Ports

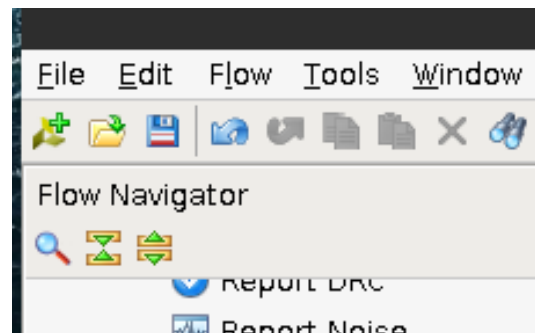
Name	Direction	Neg Diff Pair	Site	Fixed	Bank	I/O Std	Vcco	Vref	Drive Strength	Slew Type	Pull Type
All ports (6)											
Scalar ports (6)											
chave0	Input					default (LVCMOS18)	1.0...				NONE
chave1	Input					default (LVCMOS18)	1.0...				NONE
chave2	Input					default (LVCMOS18)	1.0...				NONE
led0	Output					default (LVCMOS18)	1.0...		12	SLOW	NONE
led1	Output					default (LVCMOS18)	1.0...		12	SLOW	NONE
led2	Output					default (LVCMOS18)	1.0...		12	SLOW	NONE

Tcl Console Messages Log Reports Design Runs Package Pins I/O Ports

Modifique conforme exemplo abaixo:




Após a modificação, salve as alterações




Deve aparecer essa janela, selecione o .xdc criado no inicio do projeto


Save Constraints

 Select a target file to write new unsaved constraints to. Choosing an existing file will update that file with the new constraints.


☐ Create a new file

File type:  XDC

File name:

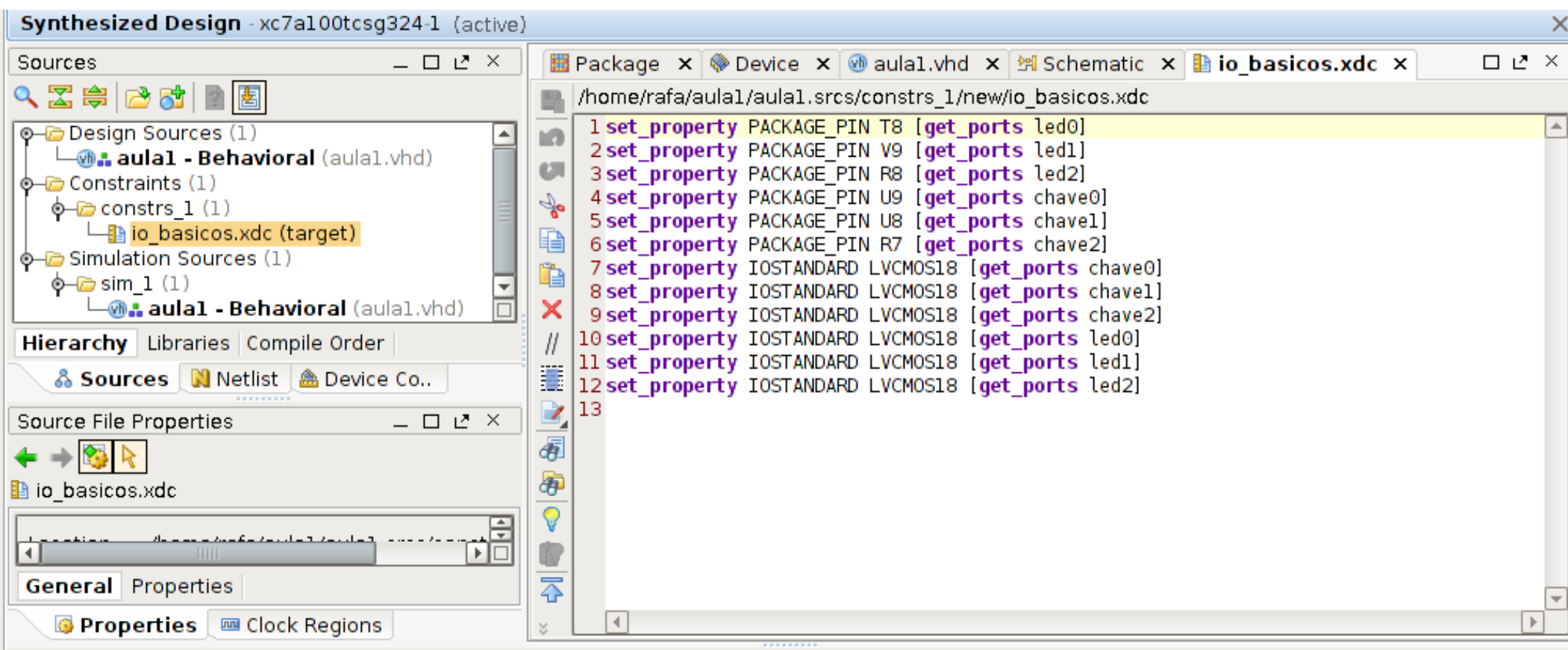
File location:  <Local to Project>

☒ Select an existing file

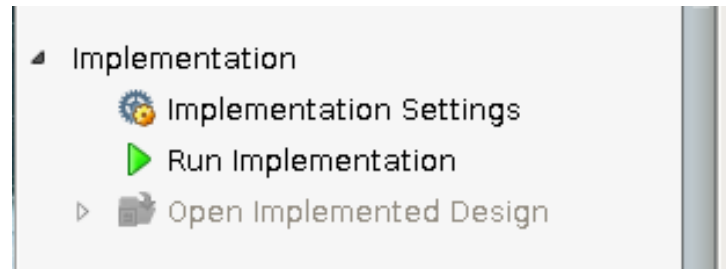
 io_basicos.xdc

OK Cancel

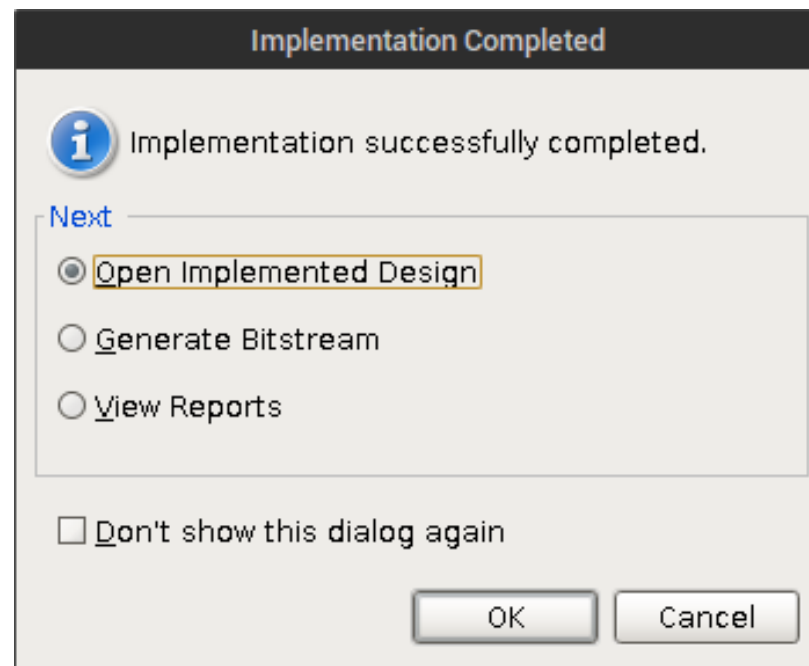
O .xdc foi alterado pela ferramenta, note que poderíamos ter editado o .XDC sem usar a ferramenta gráfica.



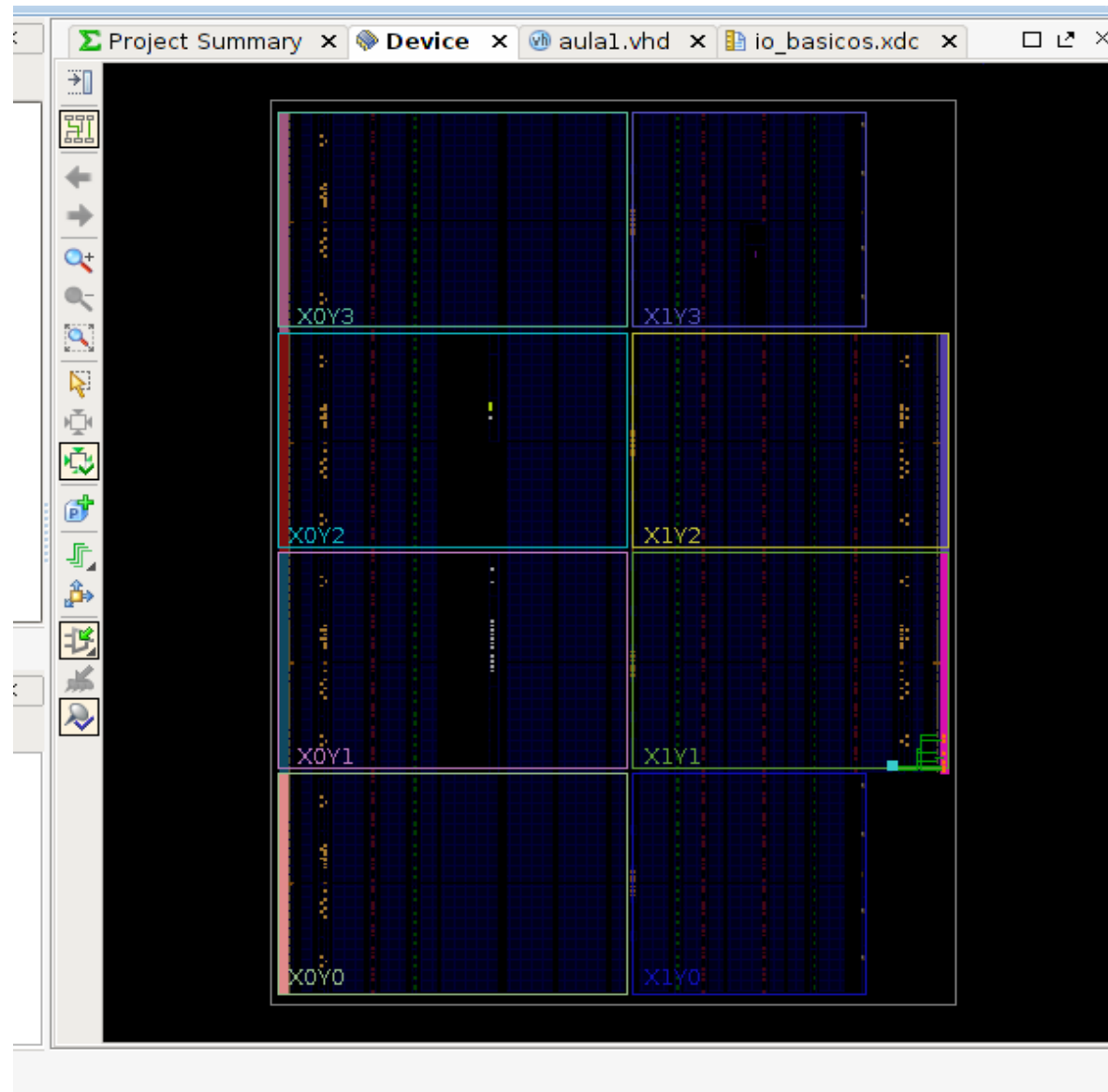
Execute a implementação



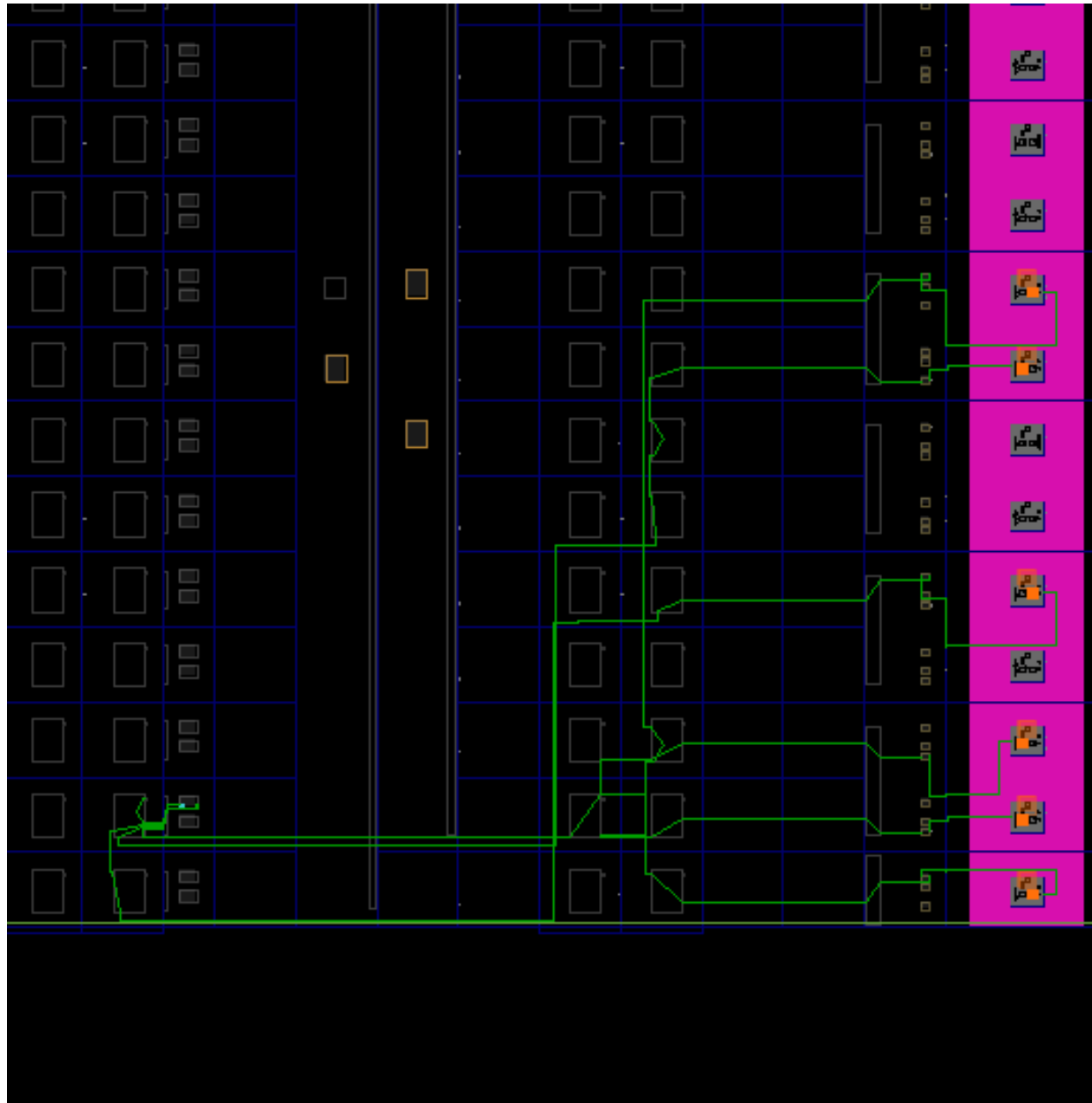
E abra o seu resultado



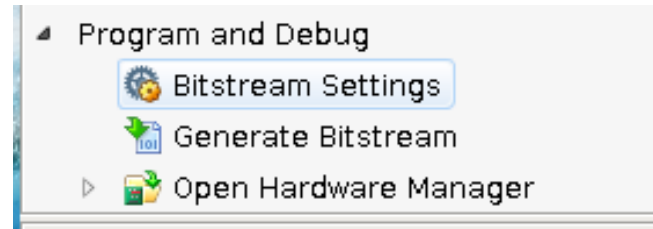
Aqui verificamos todas as ligações internas realizadas para implementar o projeto.



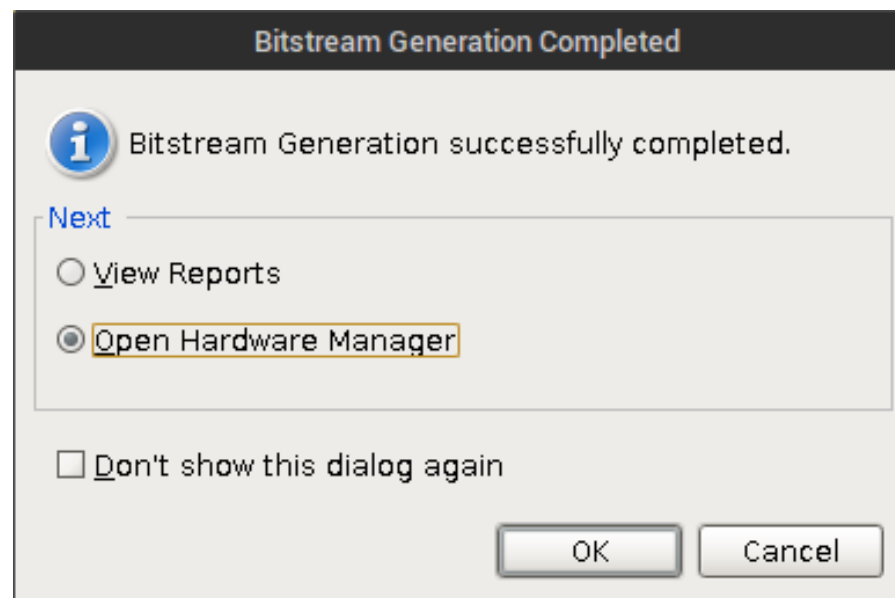
Podemos verificar as ligações e os I/Os



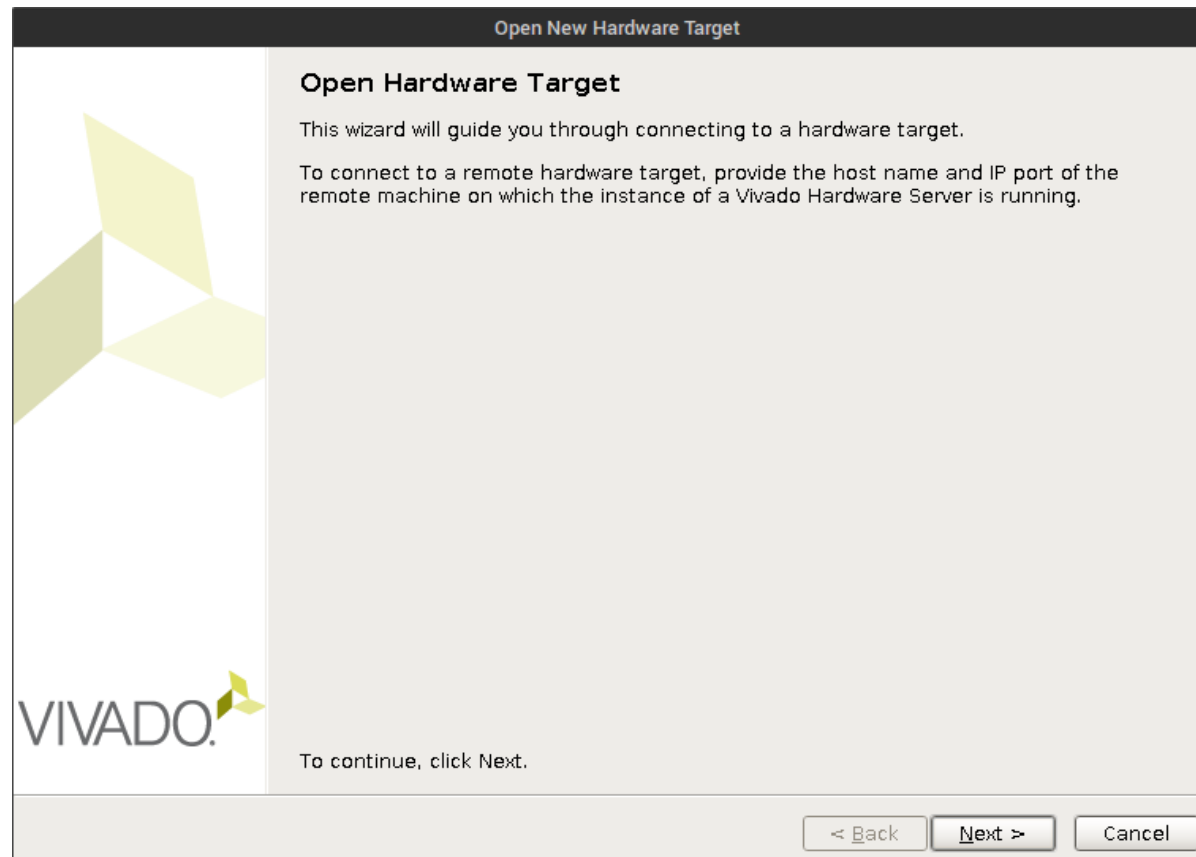
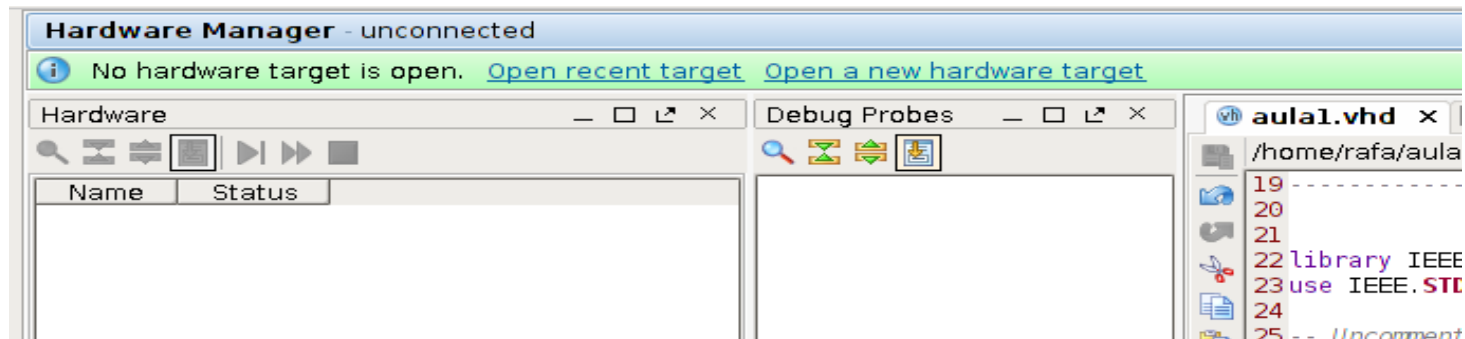
Agora vamos gravar o projeto na FPGA, antes é preciso gerar o arquivo de gravação, chamado de Bitstream:



E abriremos a interface de gravação



Na primeira vez é preciso configurar o hardware, → Open a new hardware target



Open New Hardware Target

Hardware Server Settings

Select local or remote hardware server, then configure the host name and port settings. Use Local server if the target is attached to the local machine; otherwise, use Remote server.



Connect to:

Click Next to launch and/or connect to the vcse_server (port 60001) and hw_server (port 3121) applications on the local machine.

< Back

Next >

Finish

Cancel

Open New Hardware Target

Select Hardware Target

Select a hardware target from the list of available targets, then set the appropriate JTAG clock (TCK) frequency. If you do not see the expected devices, decrease the frequency or select a different target.



Hardware Targets

Type	Port	Name	JTAG Clock Frequency
xilinx_tcf		Digilent/210274505220A	15000000

Hardware Devices (for unknown devices, specify the Instruction Register (IR) length)

Name	ID Code	IR Length
xc7a100t_0	13631093	6

VCSE server: localhost:60001

Hardware server: localhost:3121

VCSE server: localhost:60001

< Back

Next >

Finish

Cancel

Open New Hardware Target

Open Hardware Target Summary

- ❖ Hardware Server Settings:
 - Server: localhost:3121
- ❖ VCSE Server Settings:
 - Server: localhost:60001
 - Version: 20
- ❖ Target Settings:
 - Target: xilinx_tcf/Digilent/210274505220A
 - Frequency: 15000000



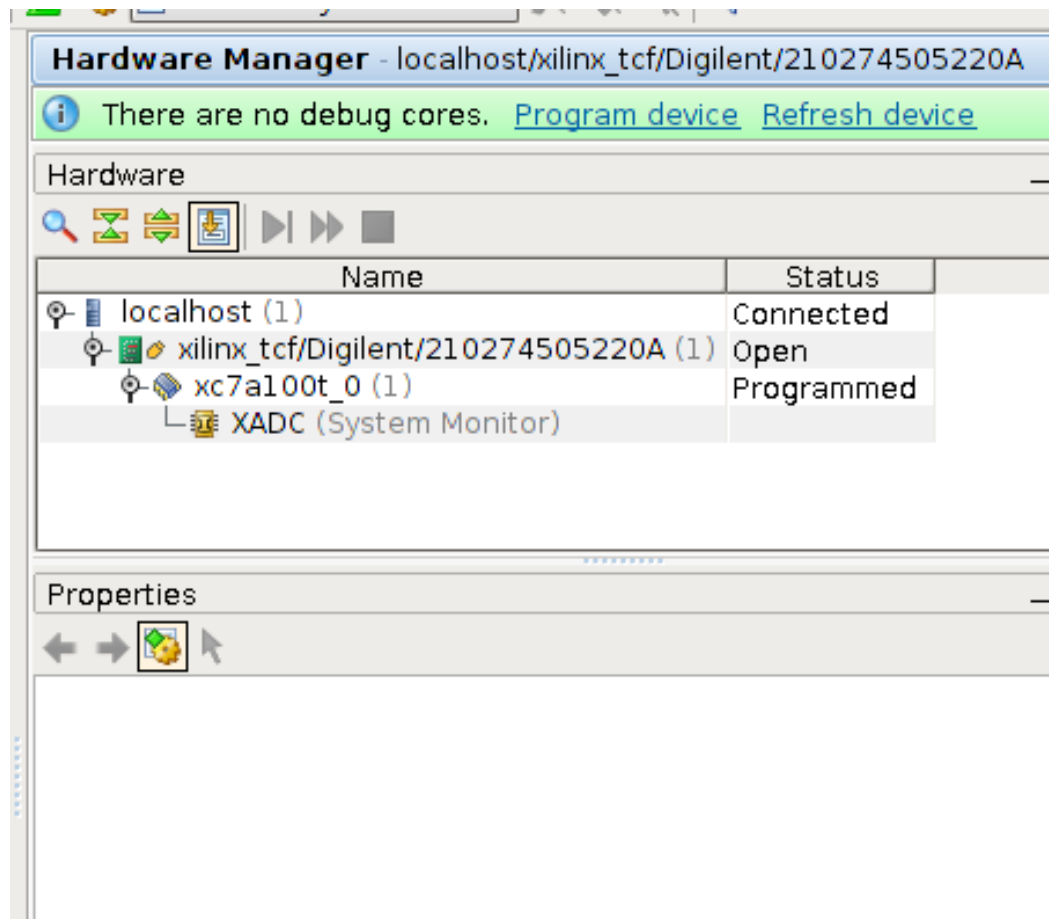
To connect to the hardware described above, click Finish

< Back

Finish

Cancel

Com o HW configurado, basta clicarmos em “Program device” para gravarmos o programa na FPGA



Escolha o Bitstream e clique em program



FIM