Lab 03 - Vivado

GitHub repository

Table with connections

Our used connection

Port	Connected to	FPGA pin
a_i[0]	SW[0]	J15
a_i[1]	SW[1]	L16
b_i[0]	SW[2]	M13
b_i[1]	SW[3]	R15
c_i[0]	SW[4]	R17
c_i[1]	SW[5]	T18
d_i[0]	SW[6]	U18
d_i[1]	SW[7]	R13
sel_i[0]	SW[14]	U11
sel_i[1]	SW[15]	V10
f_o[0]	LED[0]	H17
f_o[1]	LED[1]	K15

Development board connection

Name	FPGA pin	FPGA package pin name
SW[0]	IO_L24N_T3_RS0_15	J15
SW[1]	IO_L3N_T0_DQS_EMCCLK_14	L16
SW[2]	IO_L6N_T0_D08_VREF_14	M13
SW[3]	IO_L13N_T2_MRCC_14	R15
SW[4]	IO_L12N_T1_MRCC_14	R17
SW[5]	IO_L7N_T1_D10_14	T18
SW[6]	IO_L17N_T2_A13_D29_14	U18
SW[7]	IO_L5N_T0_D07_14	R13
SW[8]	IO_L24N_T3_34	T8

Name	FPGA pin	FPGA package pin name
SW[9]	IO_25_34	U8
SW[10]	IO_L15P_T2_DQS_RDWR_B_14	R16
SW[11]	IO_L23P_T3_A03_D19_14	T13
SW[12]	IO_L24P_T3_35	H6
SW[13]	IO_L20P_T3_A08_D24_14	U12
SW[14]	IO_L19N_T3_A09_D25_VREF_14	U11
SW[15]	IO_L21P_T3_DQS_14	V10

Name	FPGA pin	FPGA package pin name
LED[0]	IO_L18P_T2_A24_15	H17
LED[1]	IO_L24P_T3_RS1_15	K15
LED[2]	IO_L17N_T2_A25_15	J13
LED[3]	IO_L8P_T1_D11_14	N14
LED[4]	IO_L7P_T1_D09_14	R18
LED[5]	IO_L18N_T2_A11_D27_14	V17
LED[6]	IO_L17P_T2_A14_D30_14	U17
LED[7]	IO_L18P_T2_A12_D28_14	U16
LED[8]	IO_L16N_T2_A15_D31_14	V16
LED[9]	IO_L14N_T2_SRCC_14	T15
LED[10]	IO_L22P_T3_A05_D21_14	U14
LED[11]	IO_L15N_T2_DQS_DOUT_CSO_B_14	T16
LED[12]	IO_L16P_T2_CSI_B_14	V15
LED[13]	IO_L22N_T3_A04_D20_14	V14
LED[14]	IO_L20N_T3_A07_D23_14	V12
LED[15]	IO_L21N_T3_DQS_A06_D22_14	V11

2-bit wide 4-to-1 multiplexer

VHDL architecture

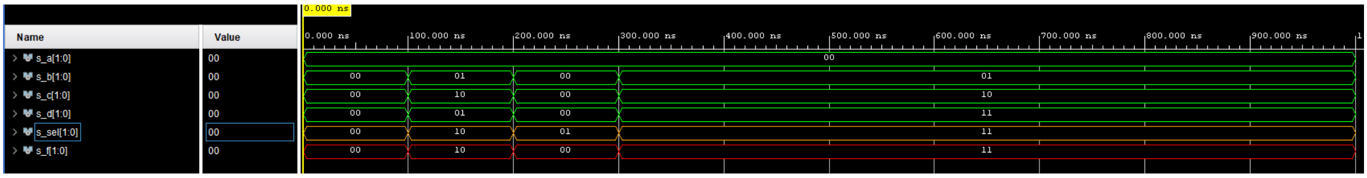
```
architecture Behavioral of mux_2bit_4to1 is
begin
    f_o <= a_i when (sel_i = "00") else
           b_i when (sel_i = "01") else
           c_i when (sel_i = "10") else
           d_i;

end architecture Behavioral;
```

Testbench file

```
library ieee;
use ieee.std_logic_1164.all;

------------------------------------------------------------------------
-- Entity declaration for testbench
------------------------------------------------------------------------
entity tb_mux_2bit_4to1 is
    -- Entity of testbench is always empty
end entity tb_mux_2bit_4to1;

------------------------------------------------------------------------
-- Architecture body for testbench
------------------------------------------------------------------------
architecture testbench of tb_mux_2bit_4to1 is

    -- Local signals
    signal s_a       : std_logic_vector(2 - 1 downto 0);
    signal s_b       : std_logic_vector(2 - 1 downto 0);
    signal s_c       : std_logic_vector(2 - 1 downto 0);
    signal s_d       : std_logic_vector(2 - 1 downto 0);
    signal s_sel     : std_logic_vector(2 - 1 downto 0);

    signal s_f       : std_logic_vector(2 - 1 downto 0);

begin
    -- Connecting testbench signals with mux_2bit_4to1 entity (Unit Under Test)
    uut_mux_2bit_4to1 : entity work.mux_2bit_4to1
        port map(
            a_i   => s_a,
            b_i   => s_b,
            c_i   => s_c,
            d_i   => s_d,
            sel_i => s_sel,
            f_o   => s_f

        );
    --------------------------------------------------------------------
    -- Data generation process
    --------------------------------------------------------------------
    p_stimulus : process
    begin
        report "Stimulus process started" severity note;

        s_d <= "00";s_c <= "00";s_b <= "00";s_a <= "00";s_sel <= "00";wait for 100
```

```
ns;
        s_d <= "01";s_c <= "10";s_b <= "01";s_a <= "00";s_sel <= "10";wait for 100
ns;
        s_d <= "00";s_c <= "00";s_b <= "00";s_a <= "00";s_sel <= "01";wait for 100
ns;
        s_d <= "11";s_c <= "10";s_b <= "01";s_a <= "00";s_sel <= "11";wait for 100
ns;

        report "Stimulus process finished" severity note;
        wait;
    end process p_stimulus;

end architecture testbench;
```

Waveform from testbench simulation



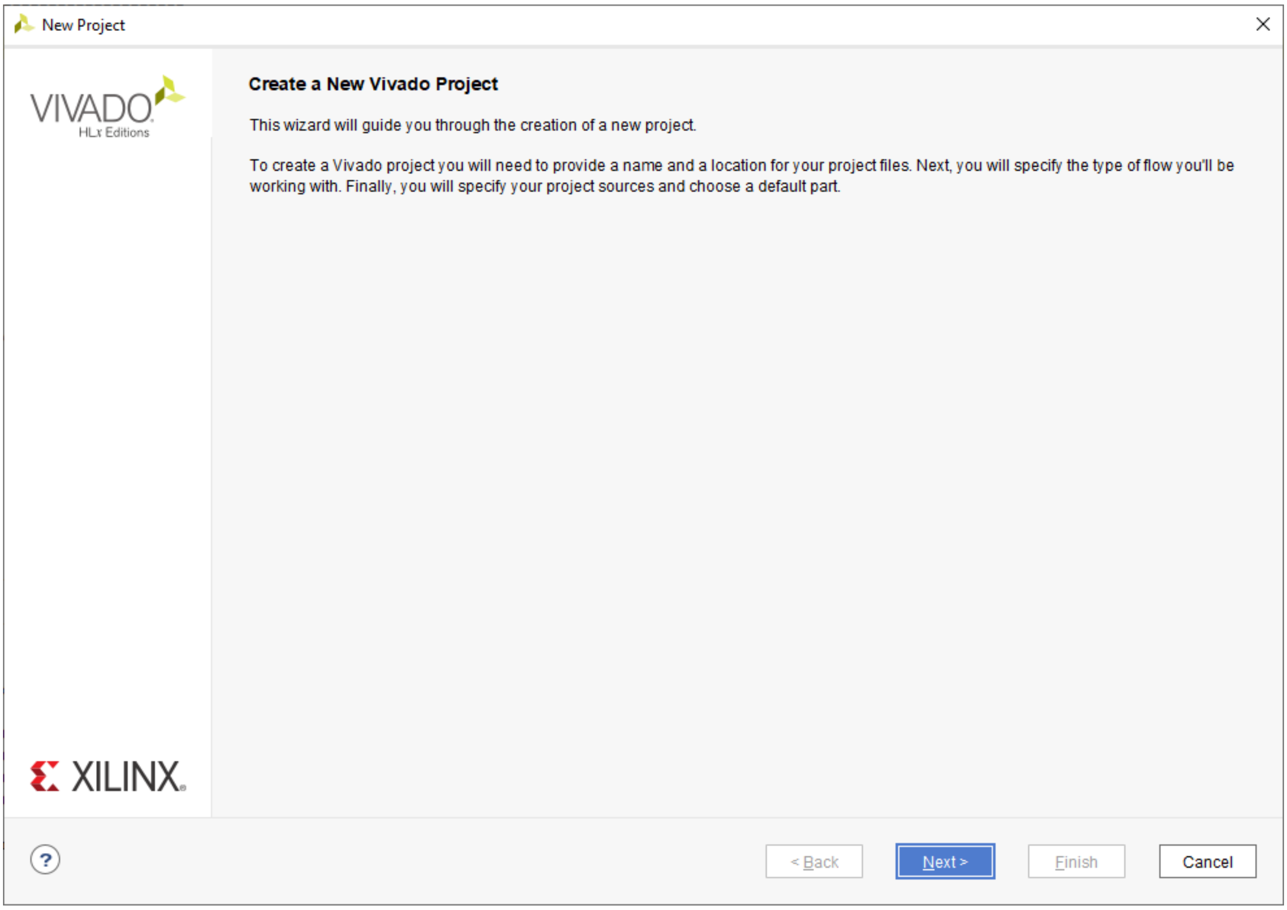
Vivado tutorial

Project creation

1. Open "**Vivado 2020.2**" (or your installed version)

2. "**File**" -> "**Project**" -> "**New...**"

3. It will open you wizard for creating new project

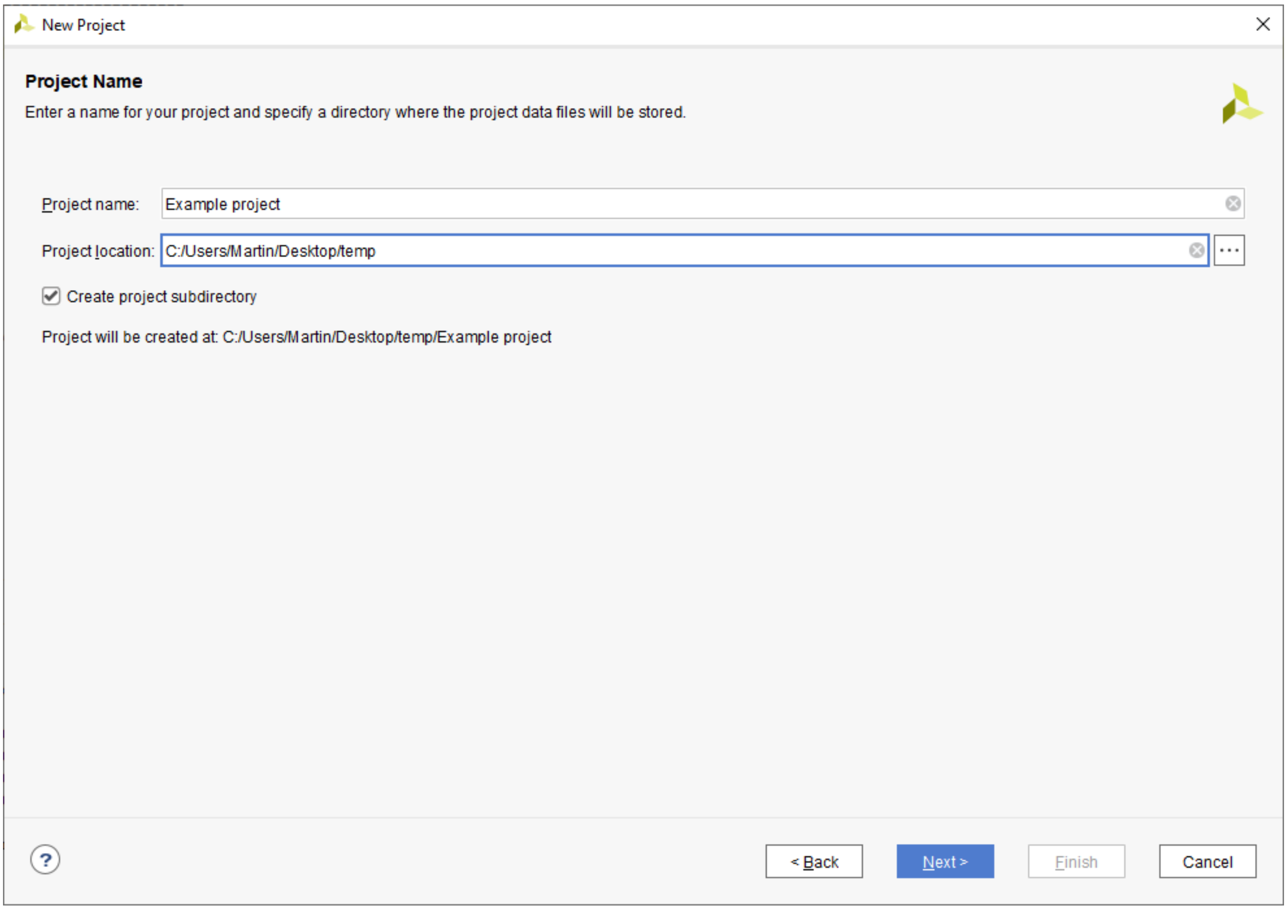


Create a New Vivado Project

This wizard will guide you through the creation of a new project.

To create a Vivado project you will need to provide a name and a location for your project files. Next, you will specify the type of flow you'll be working with. Finally, you will specify your project sources and choose a default part.

4. Select your project name and project location



Project Name

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

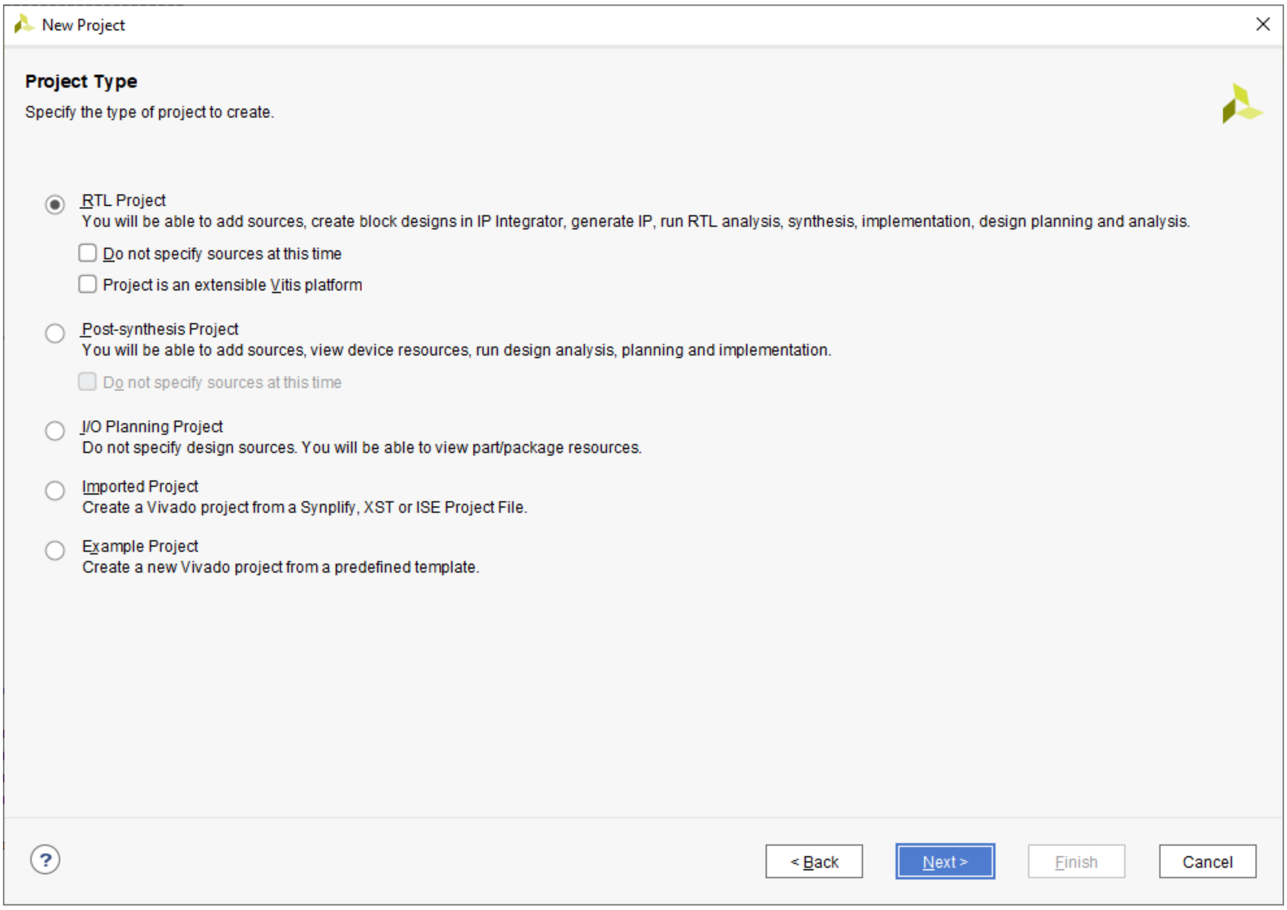
Project name: Example project

Project location: C:/Users/Martin/Desktop/temp

☑ Create project subdirectory

Project will be created at: C:/Users/Martin/Desktop/temp/Example project

5. Select project type (we are using "**RTL Project**")



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
□ Project is an extensible Vitis platform

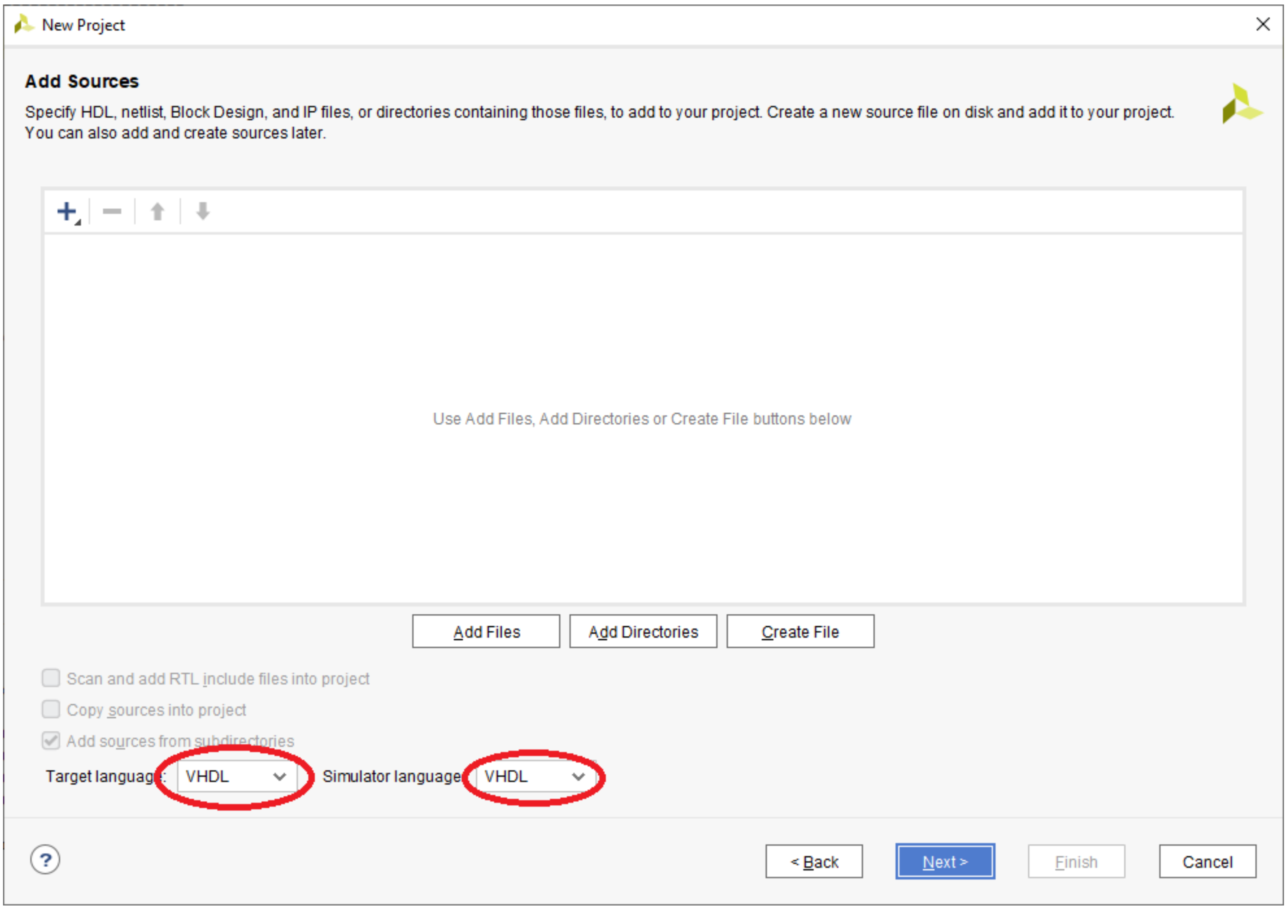
○ Post-synthesis Project
You will be able to add sources, view device resources, run design analysis, planning and implementation.
□ Do not specify sources at this time

○ 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.

○ Example Project
Create a new Vivado project from a predefined template.

6. Now select "**Target language**" and "**Simulator language**" as "**VHDL**"



New Project

Add Sources

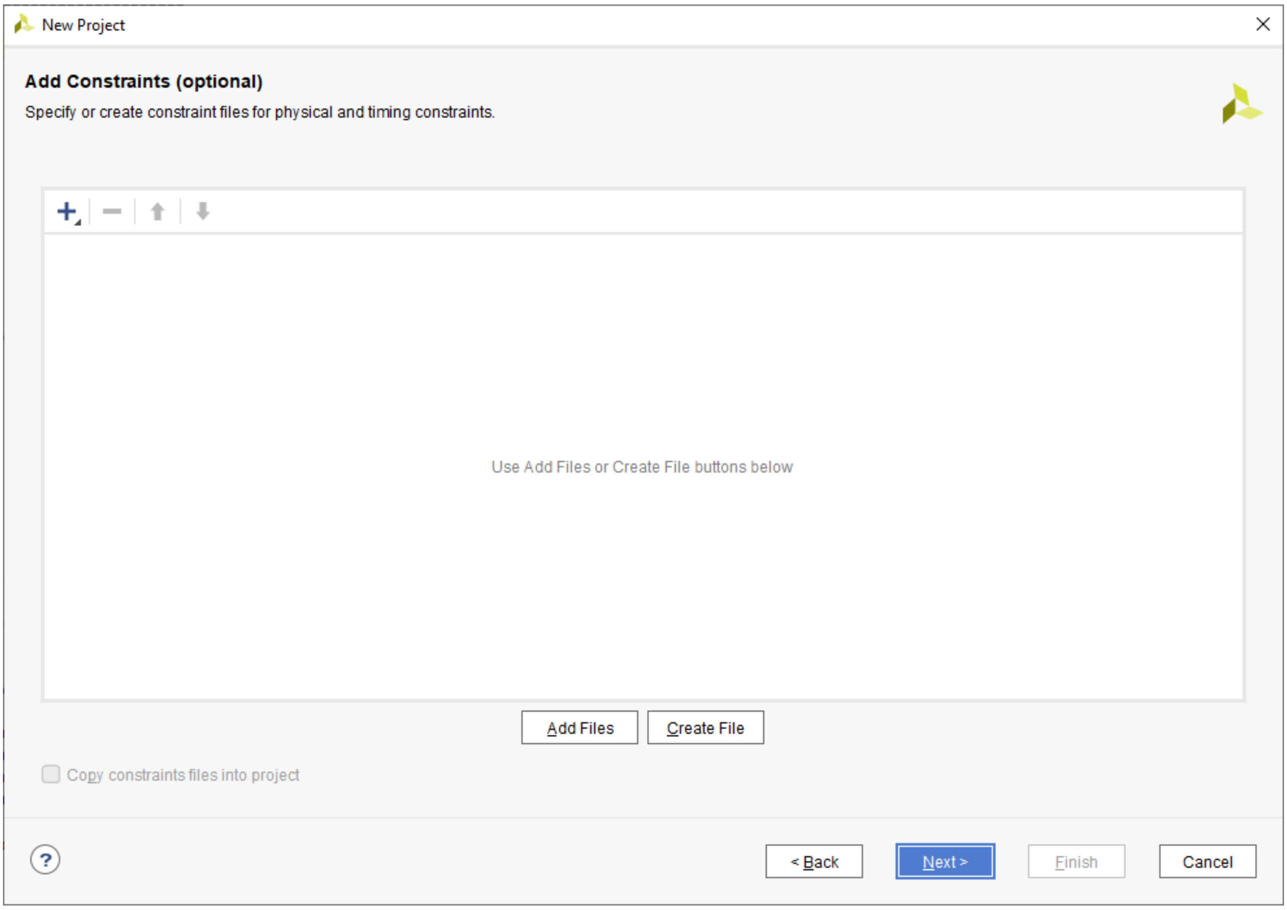
Specify HDL, netlist, Block Design, and IP files, or directories containing those files, to add to your project. Create a new source file on disk and add it to your project. You can also add and create sources later.

Use Add Files, Add Directories or Create File buttons below

□ Scan and add RTL include files into project
□ Copy sources into project
☑ Add sources from subdirectories

Target language: VHDL Simulator language: VHDL

7. Skip this windows with "**Next**" button



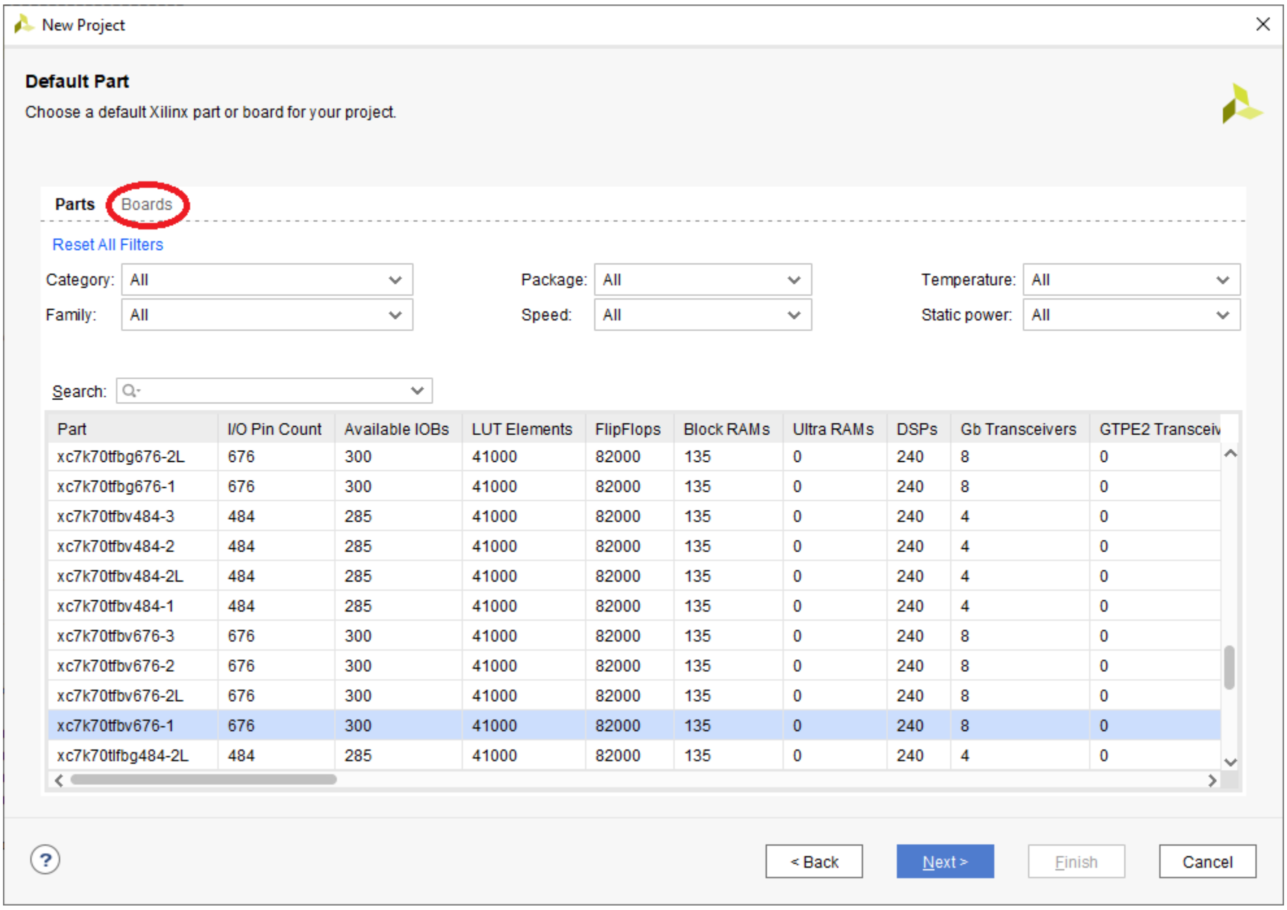
New Project

Add Constraints (optional)

Specify or create constraint files for physical and timing constraints.

Use Add Files or Create File buttons below

8. In this window get into board selection



New Project

Default Part

Choose a default Xilinx part or board for your project.

Category: All

Family: All

Package: All

Speed: All

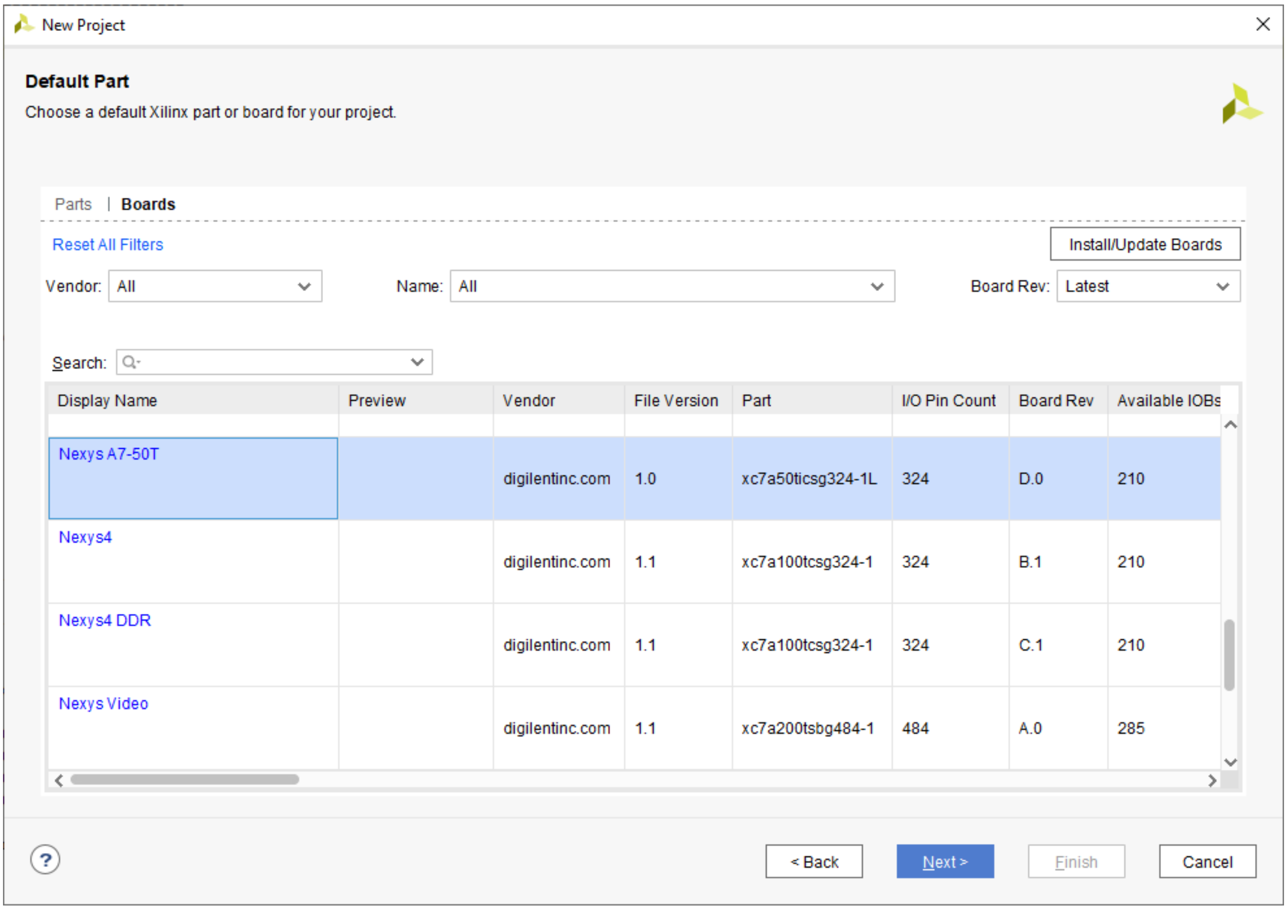
Temperature: All

Static power: All

Search:

Part	I/O Pin Count	Available IOBs	LUT Elements	FlipFlops	Block RAMs	Ultra RAMs	DSPs	Gb Transceivers	GTPE2 Transceiv
xc7k70tfbg676-2L	676	300	41000	82000	135	0	240	8	0
xc7k70tfbg676-1	676	300	41000	82000	135	0	240	8	0
xc7k70tfbv484-3	484	285	41000	82000	135	0	240	4	0
xc7k70tfbv484-2	484	285	41000	82000	135	0	240	4	0
xc7k70tfbv484-2L	484	285	41000	82000	135	0	240	4	0
xc7k70tfbv484-1	484	285	41000	82000	135	0	240	4	0
xc7k70tfbv676-3	676	300	41000	82000	135	0	240	8	0
xc7k70tfbv676-2	676	300	41000	82000	135	0	240	8	0
xc7k70tfbv676-2L	676	300	41000	82000	135	0	240	8	0
xc7k70tfbv676-1	676	300	41000	82000	135	0	240	8	0
xc7k70tlfbg484-2L	484	285	41000	82000	135	0	240	4	0

9. Choose "**Nexys A7-50T**"



Default Part

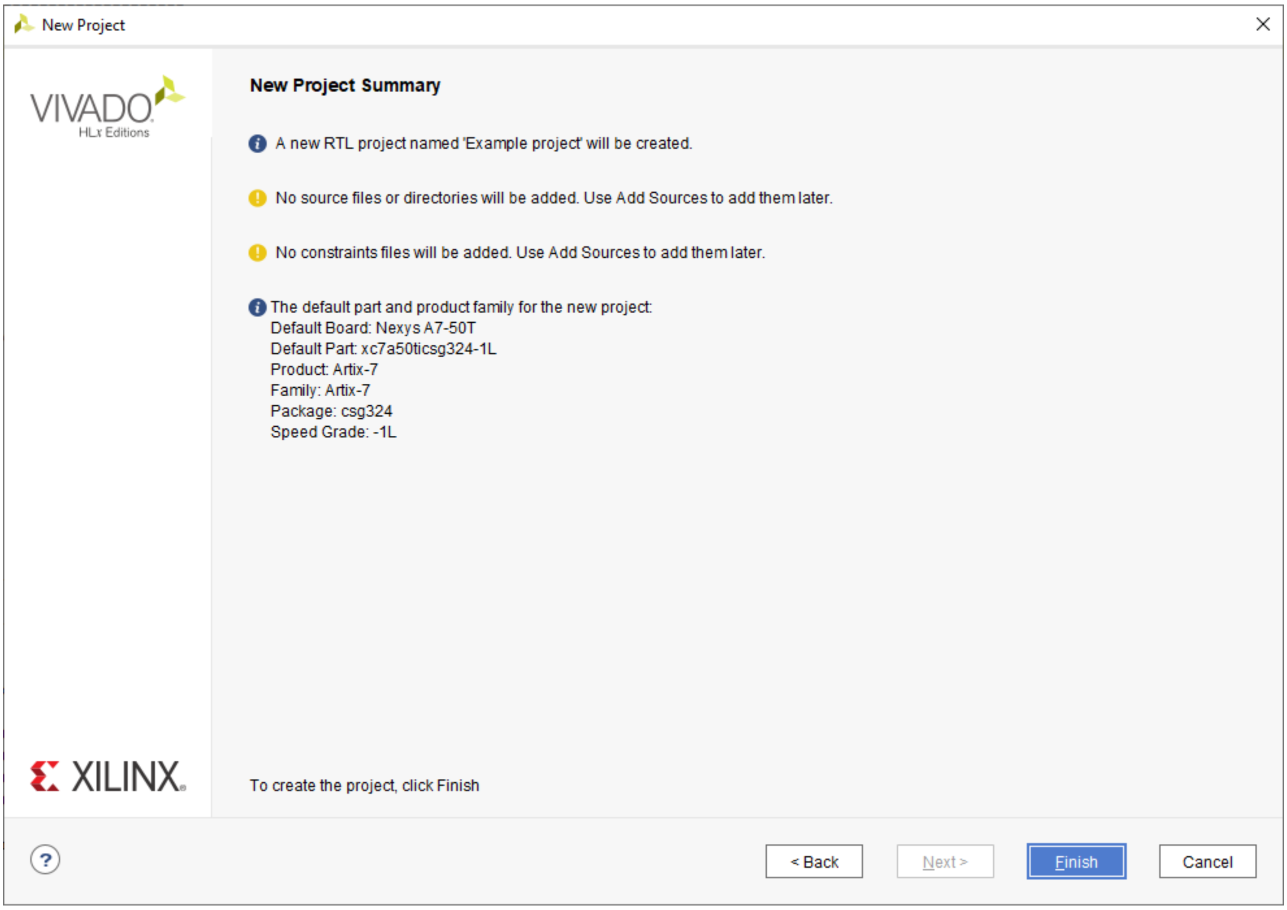
Choose a default Xilinx part or board for your project.

Vendor: All   Name: All   Board Rev: Latest

Search:

Display Name	Preview	Vendor	File Version	Part	I/O Pin Count	Board Rev	Available IOBs
Nexys A7-50T		digilentinc.com	1.0	xc7a50ticsg324-1L	324	D.0	210
Nexys4		digilentinc.com	1.1	xc7a100tcsg324-1	324	B.1	210
Nexys4 DDR		digilentinc.com	1.1	xc7a100tcsg324-1	324	C.1	210
Nexys Video		digilentinc.com	1.1	xc7a200tsbg484-1	484	A.0	285

10. Now you will see project summary. Finish with clicking at "**Finish**" button



New Project Summary

A new RTL project named 'Example project' will be created.

No source files or directories will be added. Use Add Sources to add them later.

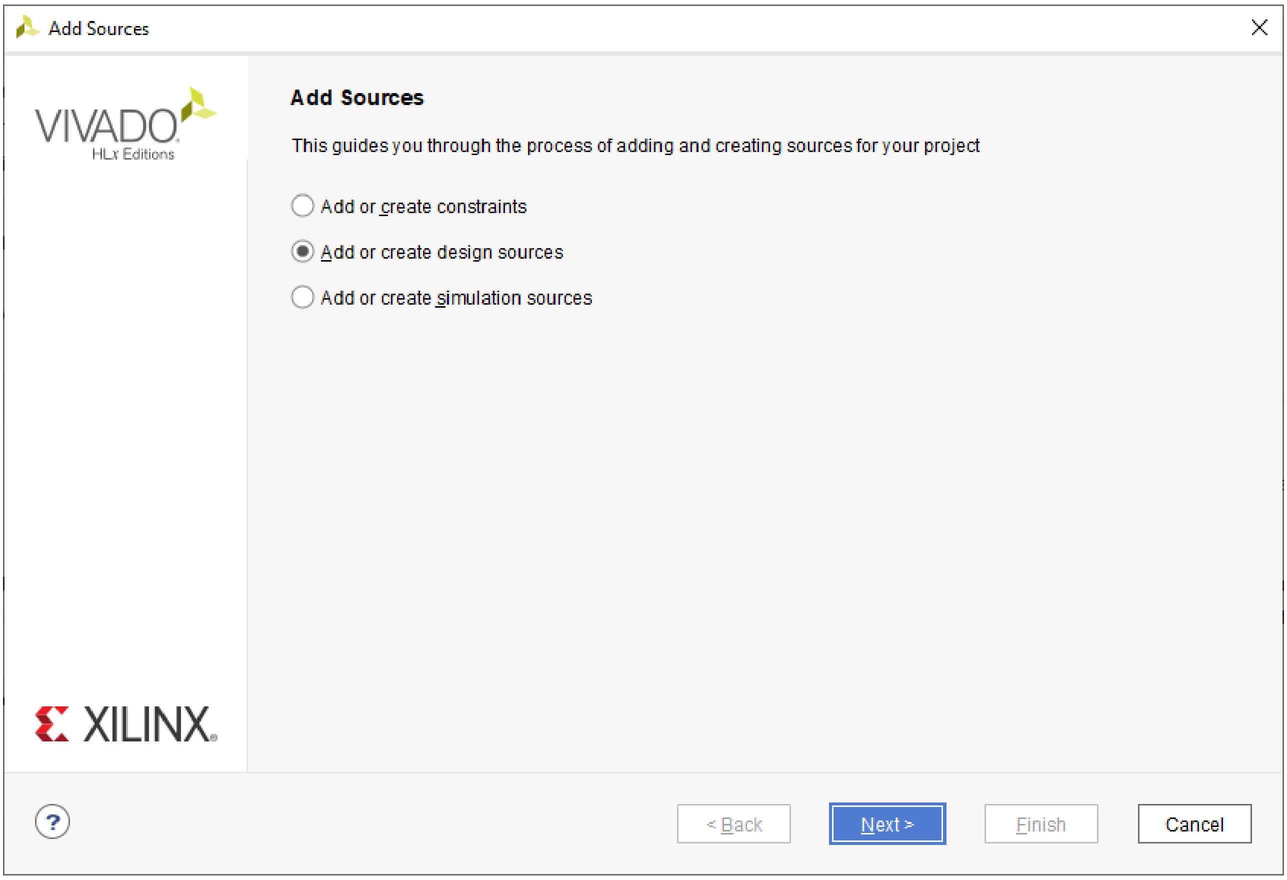
No constraints files will be added. Use Add Sources to add them later.

The default part and product family for the new project:
Default Board: Nexys A7-50T
Default Part: xc7a50ticsg324-1L
Product: Artix-7
Family: Artix-7
Package: csg324
Speed Grade: -1L

To create the project, click Finish

Adding source file

1. Open your project
2. "**File**" -> "**Add Sources...**"
3. It will open you new wizard. Select "**Add or create design sources**"



Add Sources

Add Sources

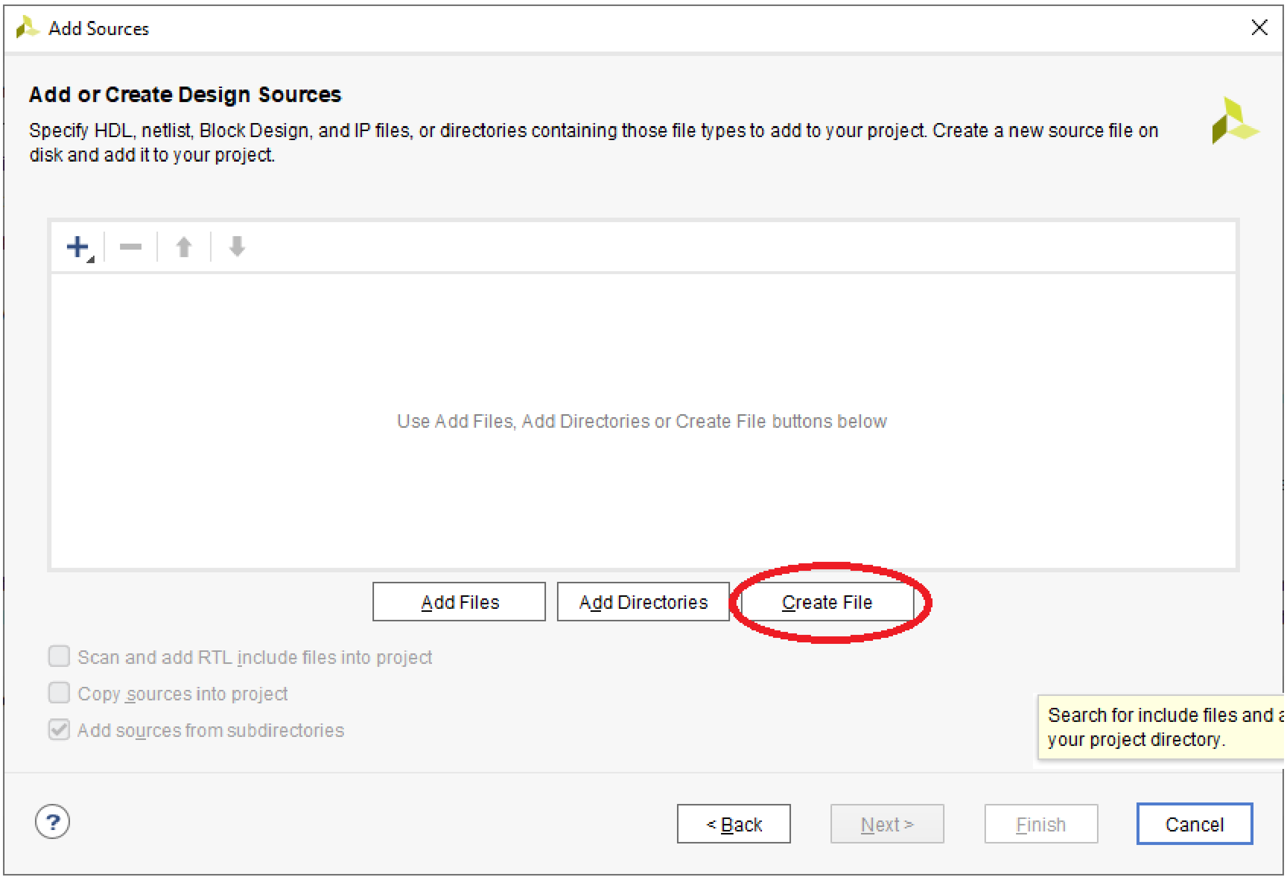
This guides you through the process of adding and creating sources for your project

○ Add or create constraints

● Add or create design sources

○ Add or create simulation sources

4. Now add source file with "**Create File**" button



Add Sources

Add or Create Design Sources

Specify HDL, netlist, Block Design, and IP files, or directories containing those file types to add to your project. Create a new source file on disk and add it to your project.

Use Add Files, Add Directories or Create File buttons below

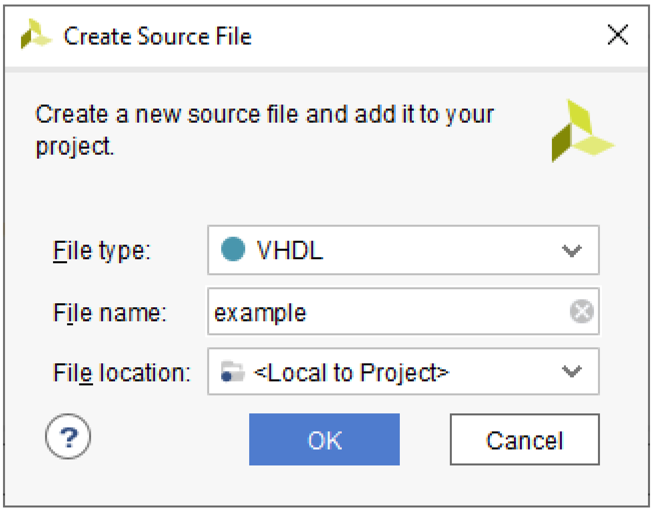
□ Scan and add RTL include files into project

□ Copy sources into project

☑ Add sources from subdirectories

Search for include files and a
your project directory.

5. Select "**VHDL**" file type and type in filename



Create Source File

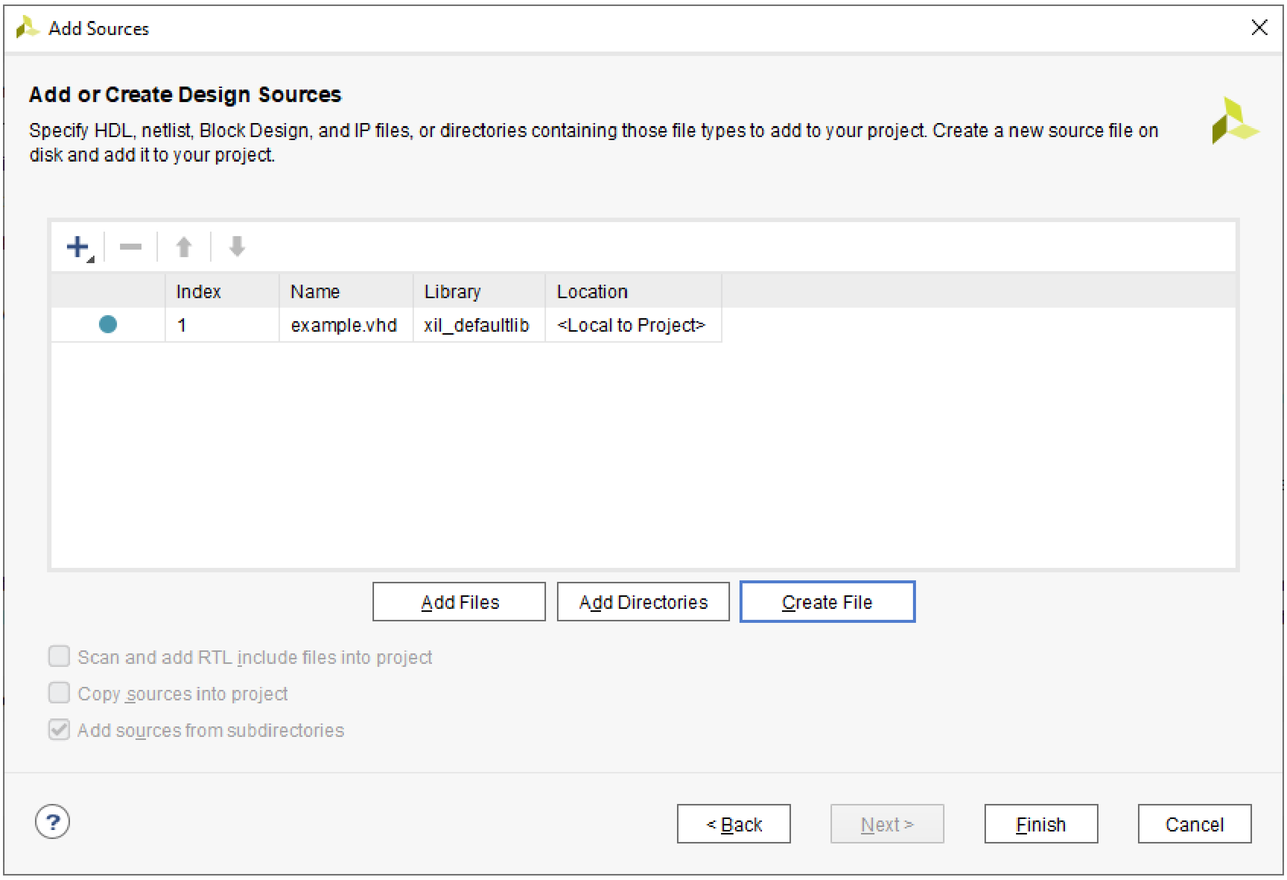
Create a new source file and add it to your project.

File type: ● VHDL

File name: example

File location: <Local to Project>

6. Now you can overview all created files



Add Sources

Add or Create Design Sources

Specify HDL, netlist, Block Design, and IP files, or directories containing those file types to add to your project. Create a new source file on disk and add it to your project.

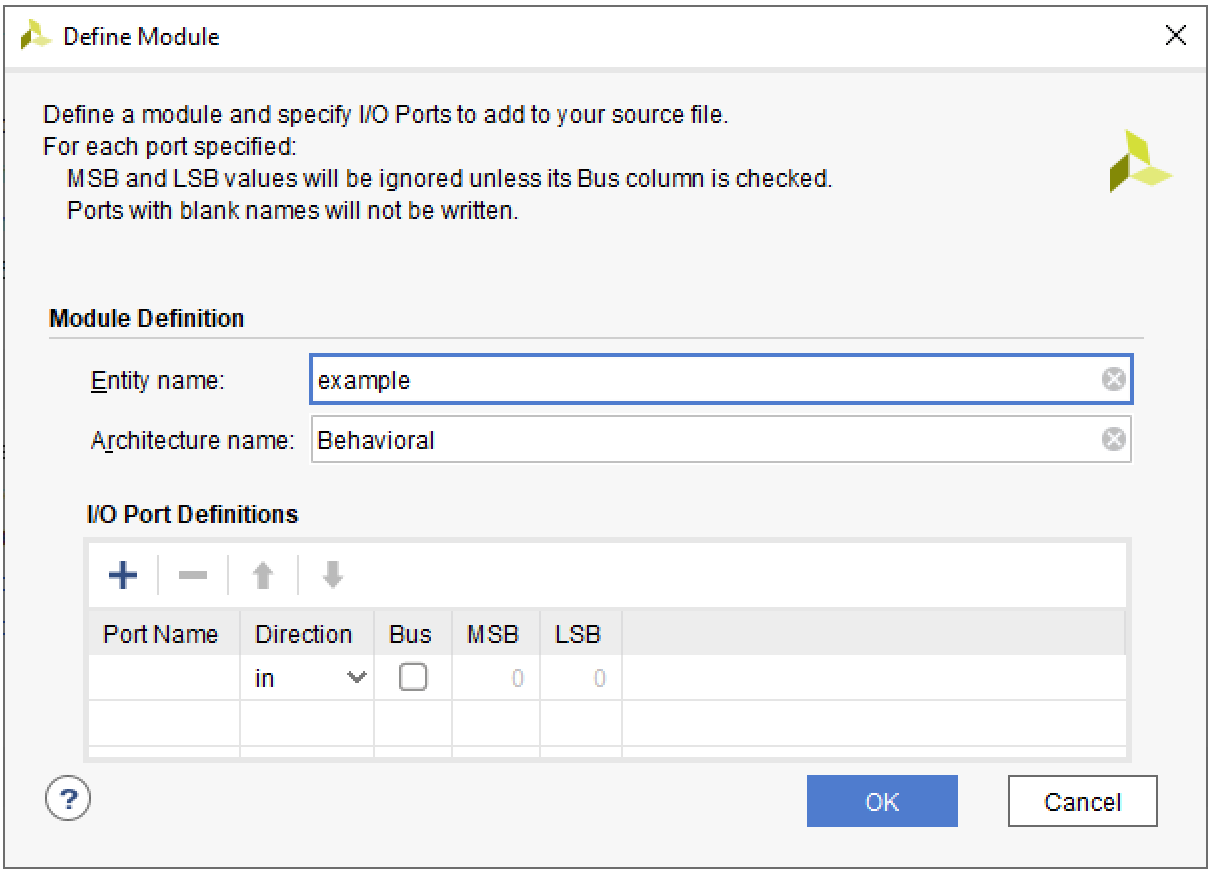
	Index	Name	Library	Location
●	1	example.vhd	xil_defaultlib	<Local to Project>

□ Scan and add RTL include files into project

□ Copy sources into project

☑ Add sources from subdirectories

7. Here you can define real connection with FPGA pins



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: example

Architecture name: Behavioral

I/O Port Definitions

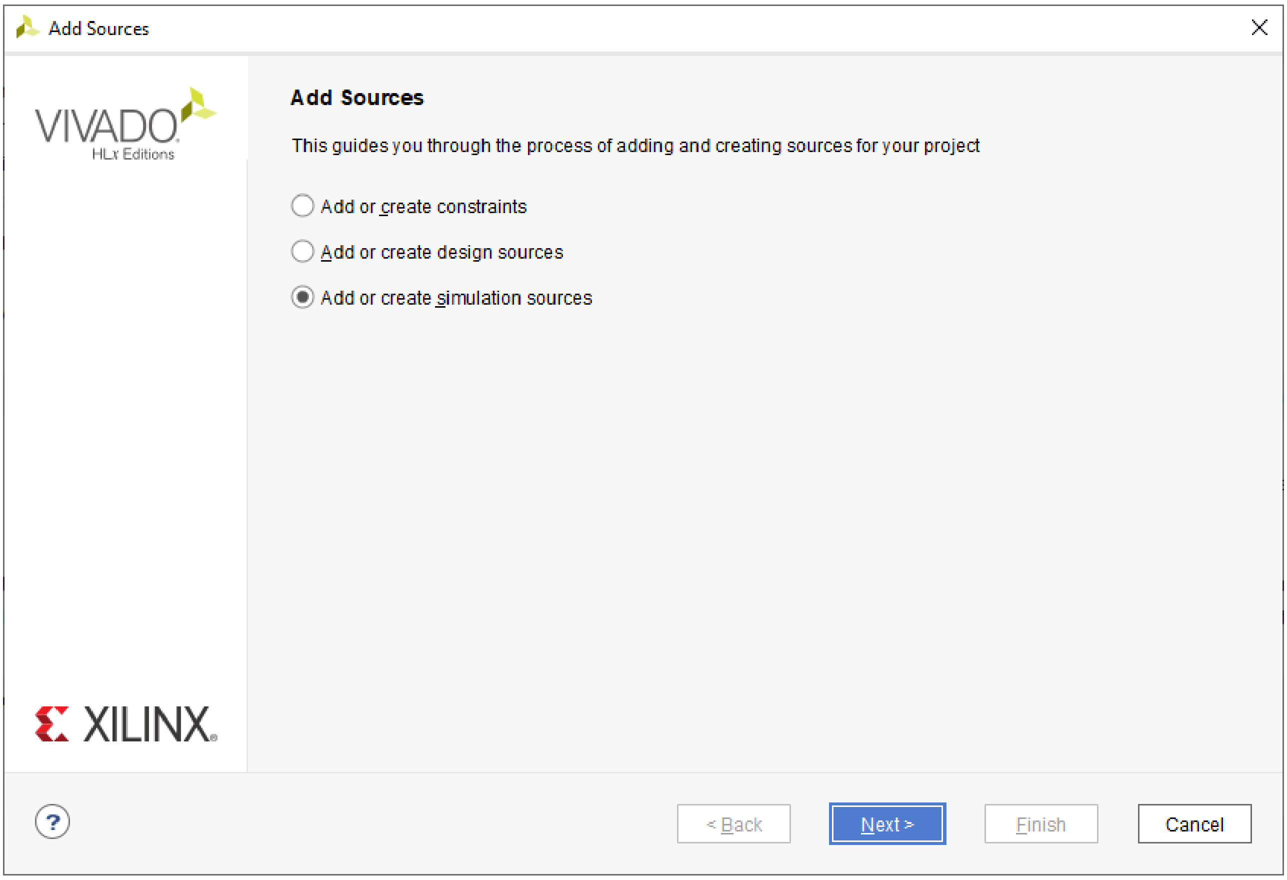
Port Name	Direction	Bus	MSB	LSB
	in	□	0	0

Adding testbench file

1. Open your project

2. "**File**" -> "**Add Sources...**"

3. It will open you new wizard. Select "**Add or create simulation sources**"

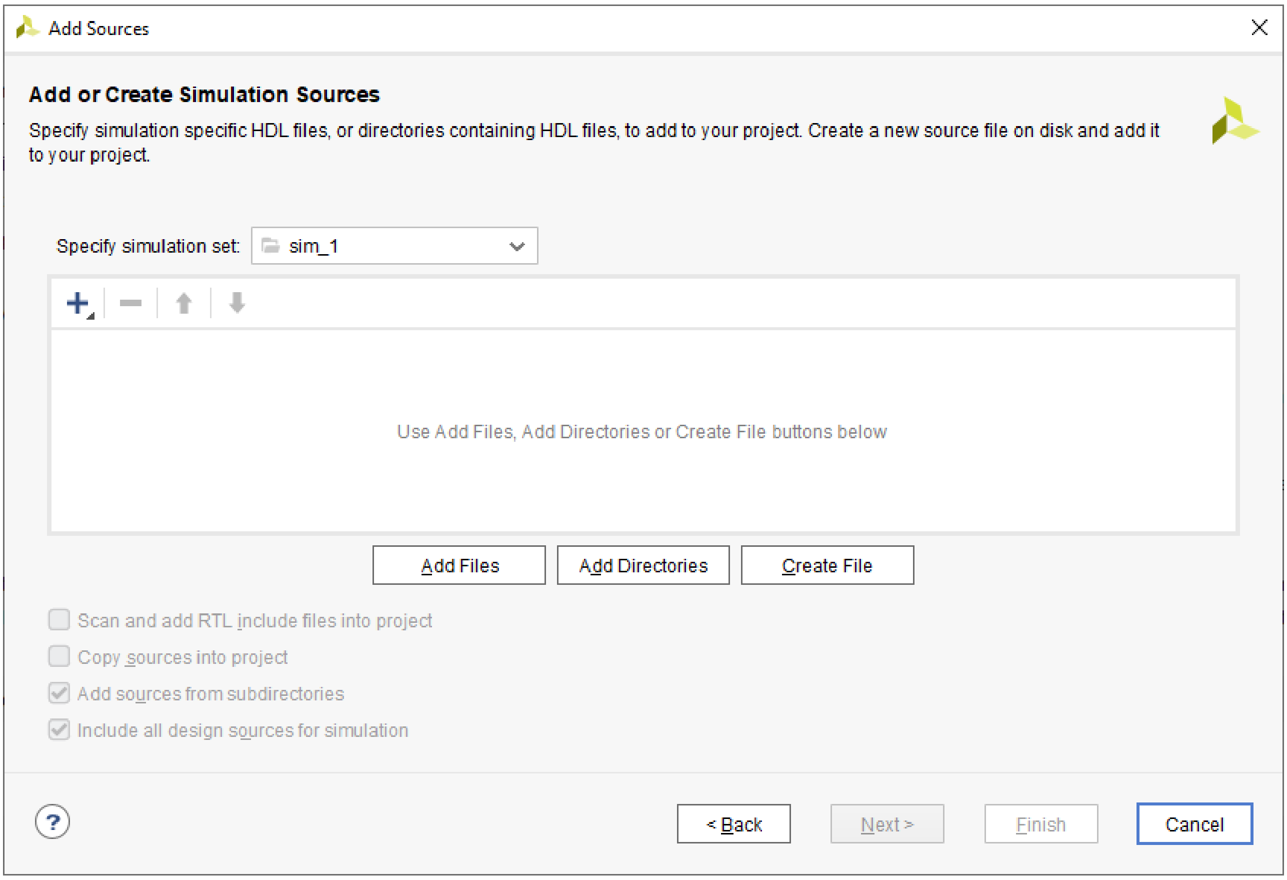


Add Sources

This guides you through the process of adding and creating sources for your project

○ Add or create constraints
○ Add or create design sources
● Add or create simulation sources

4. Now add source file with "**Create file**" button



Add or Create Simulation Sources

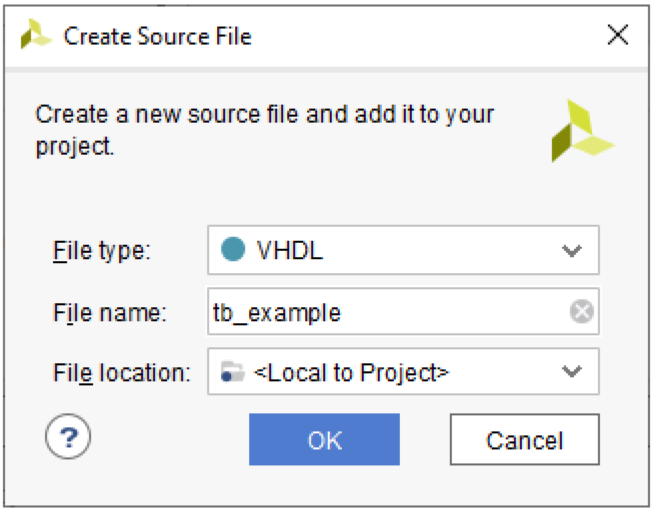
Specify simulation specific HDL files, or directories containing HDL files, to add to your project. Create a new source file on disk and add it to your project.

Specify simulation set: sim_1

Use Add Files, Add Directories or Create File buttons below

□ Scan and add RTL include files into project
□ Copy sources into project
☑ Add sources from subdirectories
☑ Include all design sources for simulation

5. Select "**VHDL**" file type and type in filename



Create Source File

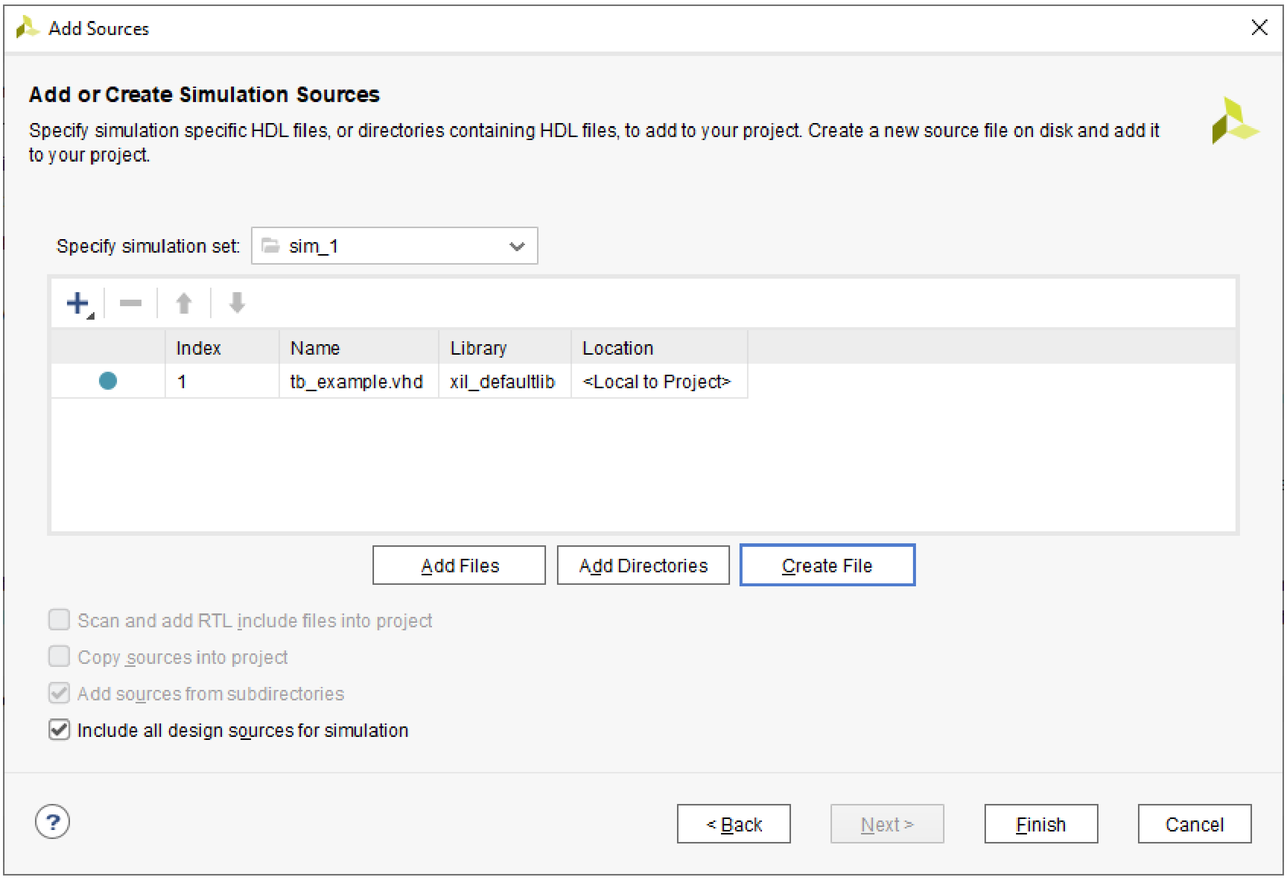
Create a new source file and add it to your project.

File type: ● VHDL

File name: tb_example

File location: <Local to Project>

6. Now you can overview all created files



Add Sources

Add or Create Simulation Sources

Specify simulation specific HDL files, or directories containing HDL files, to add to your project. Create a new source file on disk and add it to your project.

Specify simulation set: sim_1

	Index	Name	Library	Location
●	1	tb_example.vhd	xil_defaultlib	<Local to Project>

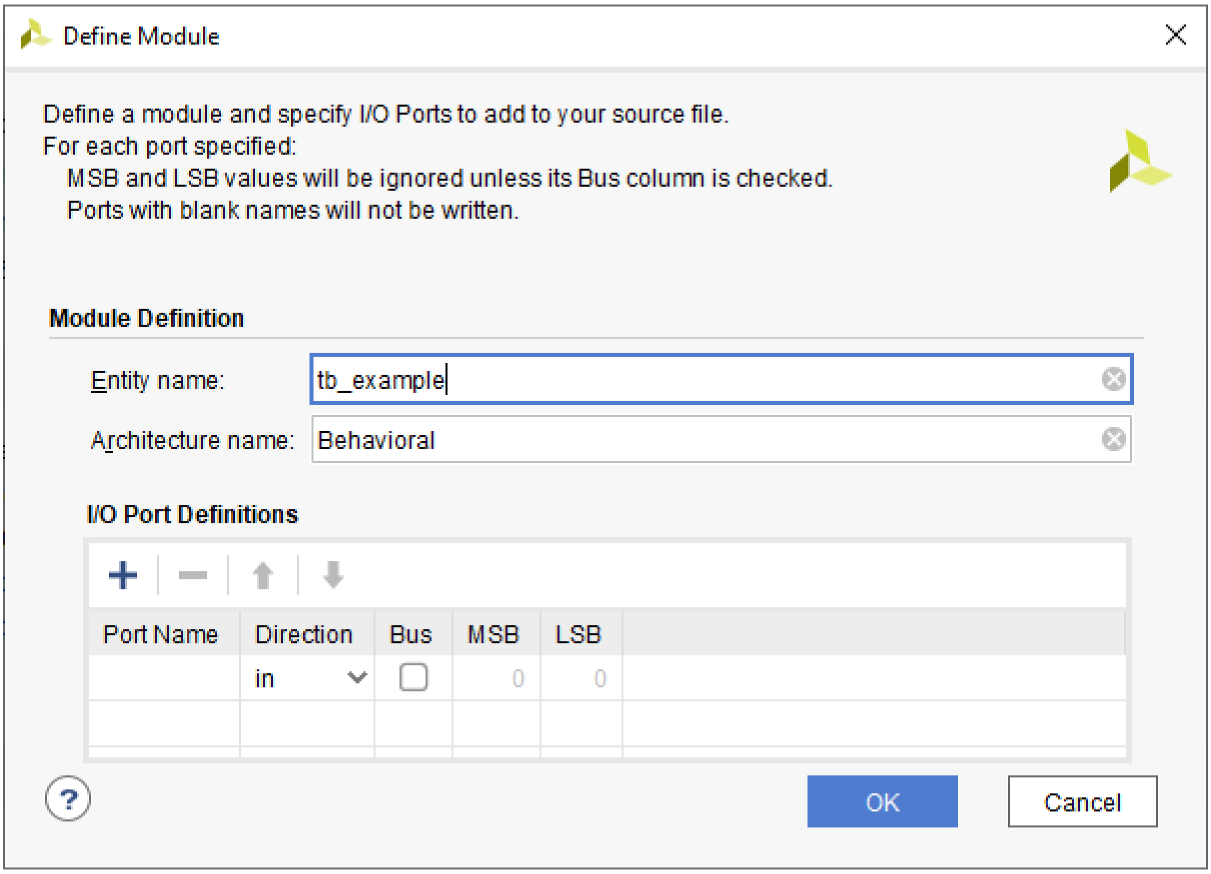
☐ Scan and add RTL include files into project

☐ Copy sources into project

☑ Add sources from subdirectories

☑ Include all design sources for simulation

7. Here you can define real connection with FPGA pins



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: tb_example

Architecture name: Behavioral

I/O Port Definitions

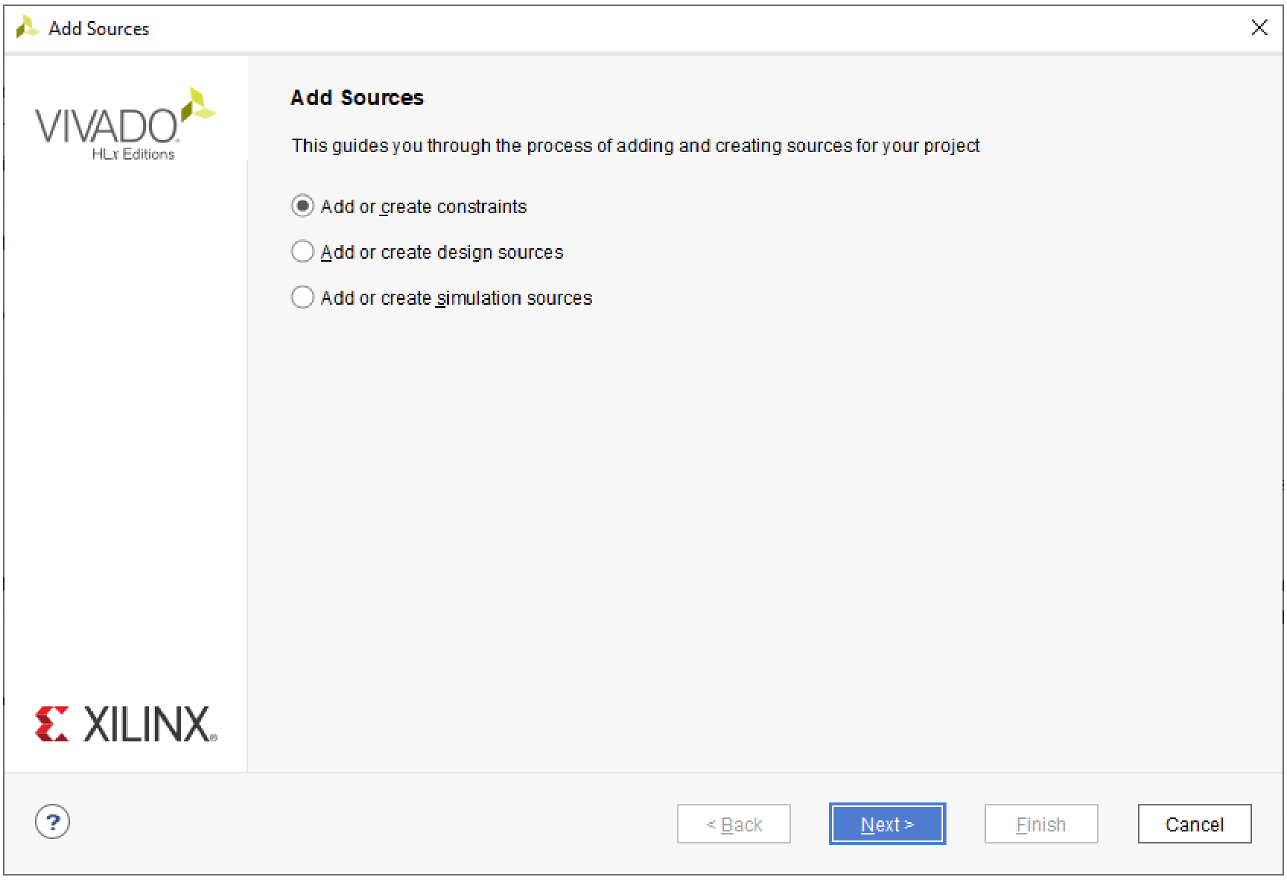
Port Name	Direction	Bus	MSB	LSB
	in	□	0	0

Adding constraints (XDC) file

1. Open your project

2. "**File**" -> "**Add Sources...**"

3. It will open you new wizard. Select "**Add or create constraints**"



Add Sources

Add Sources

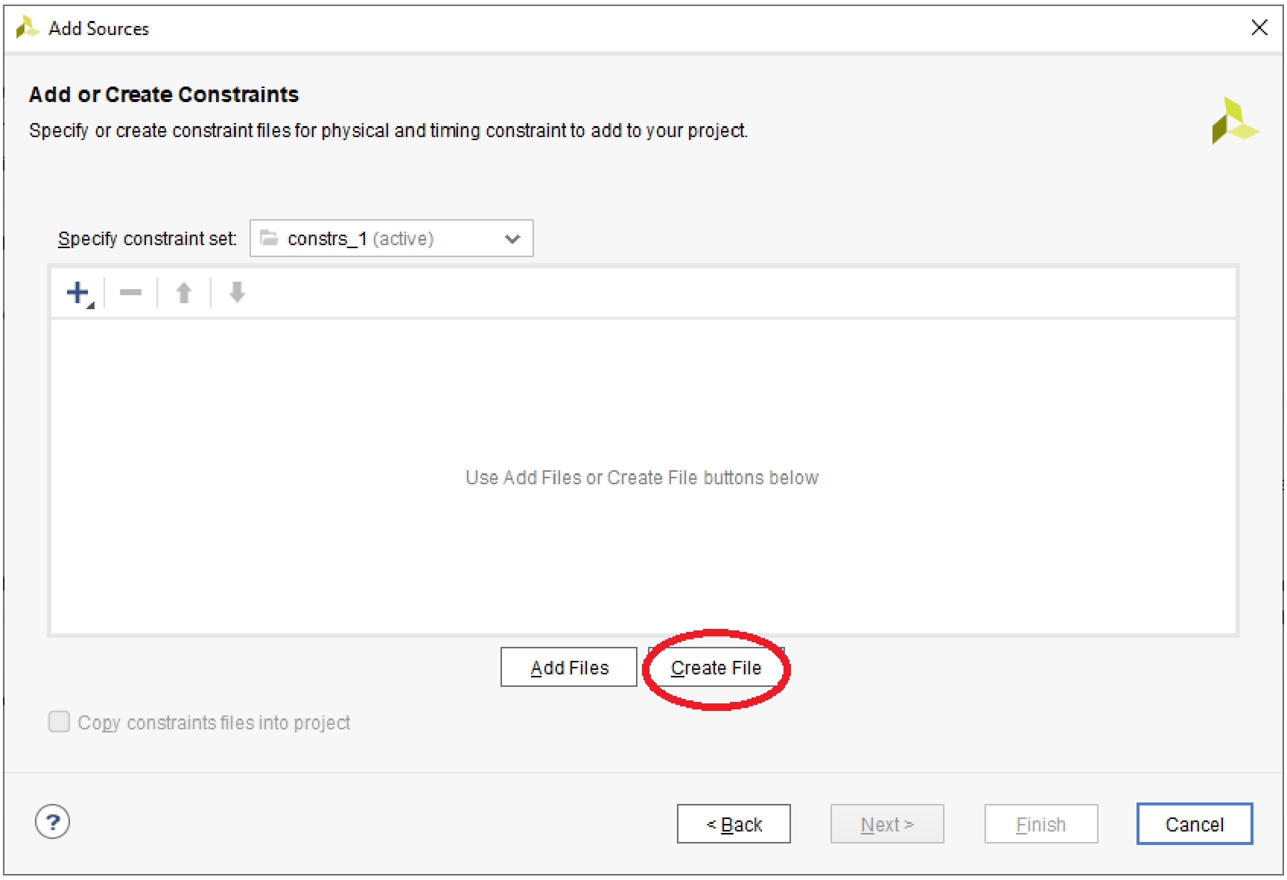
This guides you through the process of adding and creating sources for your project

● Add or create constraints

○ Add or create design sources

○ Add or create simulation sources

4. Now add source file with "**Create file**" button



Add Sources

Add or Create Constraints

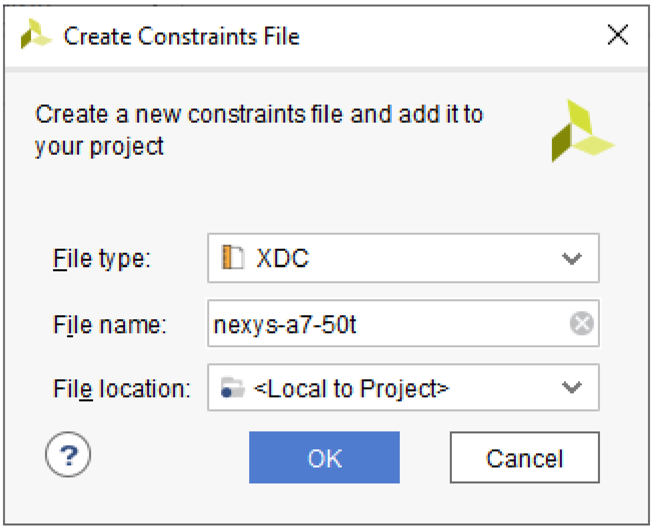
Specify or create constraint files for physical and timing constraint to add to your project.

Specify constraint set: constrs_1 (active)

Use Add Files or Create File buttons below

□ Copy constraints files into project

5. Select file type as "**XDC**" and type in file name (in this case same as board name)



Create Constraints File

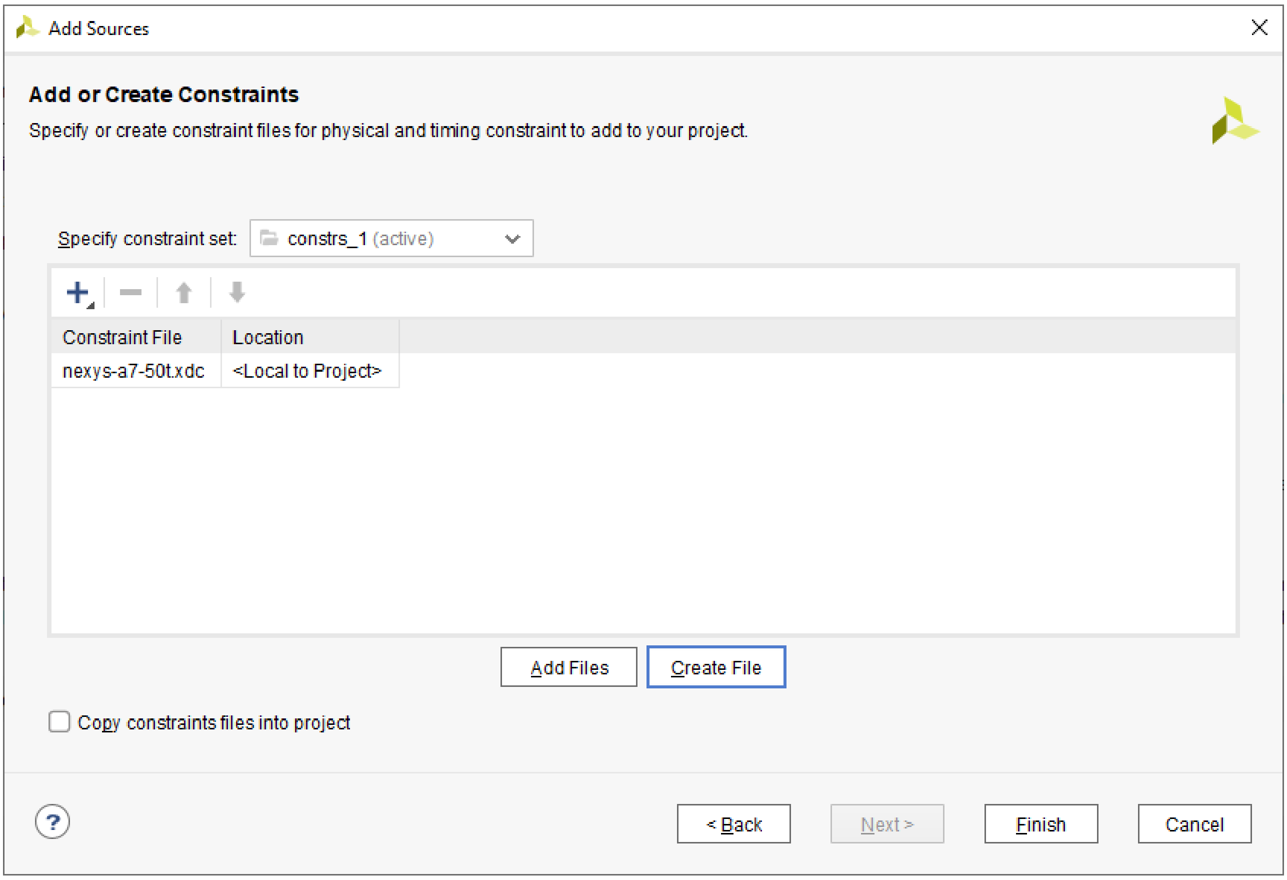
Create a new constraints file and add it to your project

File type: XDC

File name: nexys-a7-50t

File location: <Local to Project>

6. Now you can overview all created files



Add Sources

Add or Create Constraints

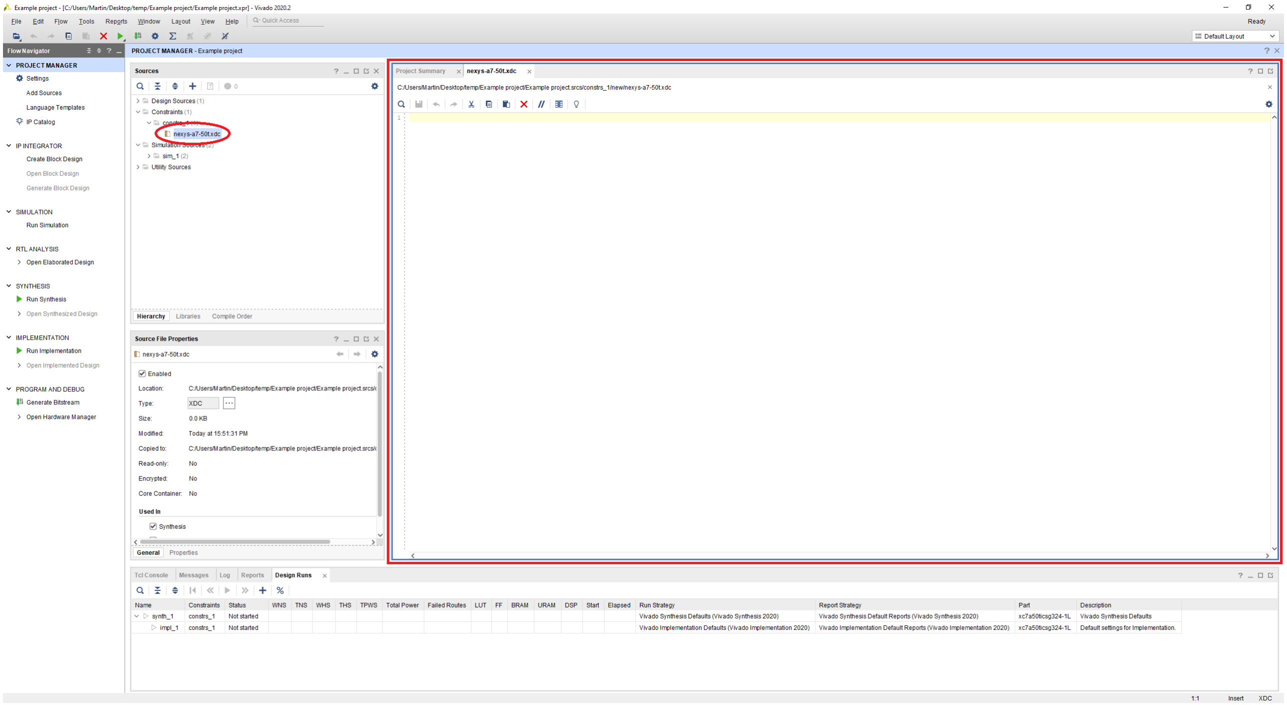
Specify or create constraint files for physical and timing constraint to add to your project.

Specify constraint set: constrs_1 (active)

Constraint File	Location
nexys-a7-50t.xdc	<Local to Project>

☐ Copy constraints files into project

7. Now in main window open this created file and type pin definition in editor window. You can also copy .xdc file from board manafacturer



C:/Users/Martin/Desktop/temp/Example project/Example project.srcs/constrs_1/new/nexys-a7-50t.xdc

Source File Properties: nexys-a7-50t.xdc

☑ Enabled

Location:	C:/Users/Martin/Desktop/temp/Example project/Example project.srcs/
Type:	XDC
Size:	0.0 KB
Modified:	Today at 15:51:31 PM
Copied to:	C:/Users/Martin/Desktop/temp/Example project/Example project.srcs/
Read-only:	No
Encrypted:	No
Core Container:	No

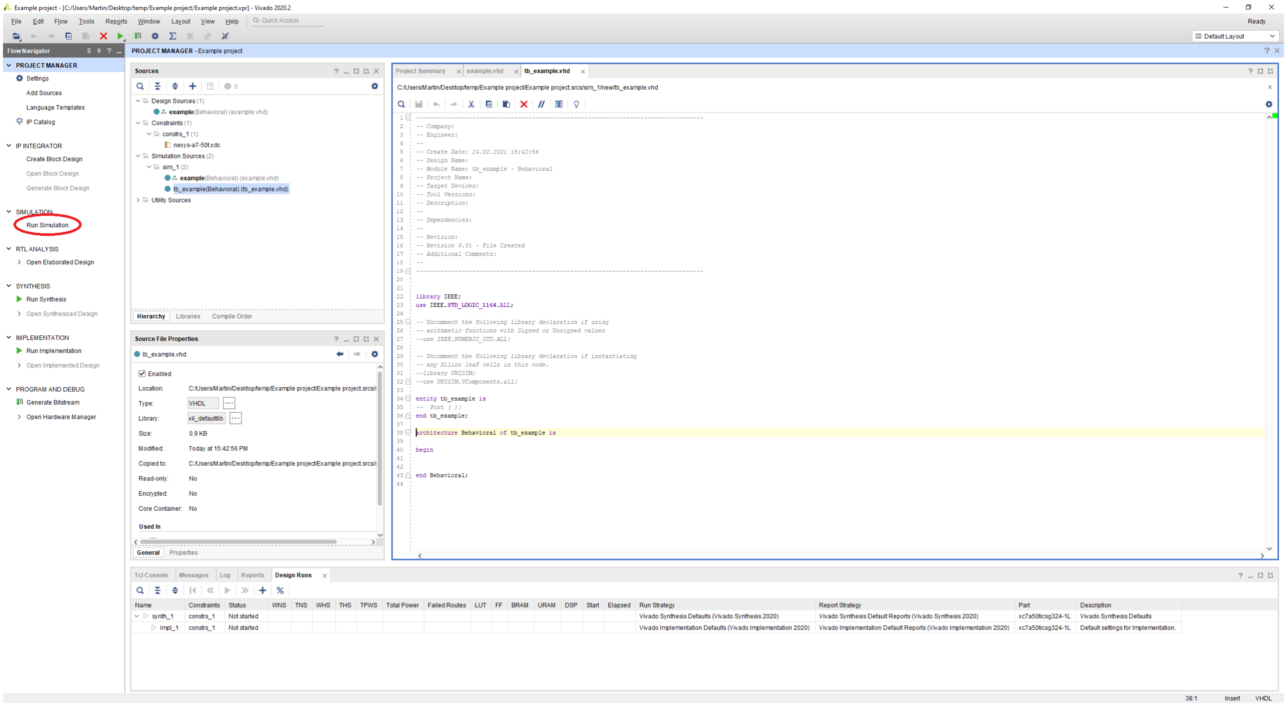
Used In

☑ Synthesis

Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAM	URAM	DSP	Start	Elapsed	Run Strategy	Report Strategy	Part	Description
synth_1	constrs_1	Not started															Vivado Synthesis Defaults (Vivado Synthesis 2020)	Vivado Synthesis Default Reports (Vivado Synthesis 2020)	xc7a50ticsg324-1L	Vivado Synthesis Defaults
impl_1	constrs_1	Not started															Vivado Implementation Defaults (Vivado Implementation 2020)	Vivado Implementation Default Reports (Vivado Implementation 2020)	xc7a50ticsg324-1L	Default settings for Implementation.

Running simulation

1. Open your project
2. If you have setup design sources and testbench files, you can run simulation



C:/Users/Martin/Desktop/temp/Example project/Example project.srcs/sim_1/new/tb_example.vhd

```vhdl
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 24.02.2021 15:42:56
-- Design Name:
-- Module Name: tb_example - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity tb_example is
--  Port ( );
end tb_example;

architecture Behavioral of tb_example is

begin


end Behavioral;

```

Source File Properties: tb_example.vhd

☑ Enabled

Location:	C:/Users/Martin/Desktop/temp/Example project/Example project.srcs/
Type:	VHDL
Library:	xil_defaultlib
Size:	0.9 KB
Modified:	Today at 15:42:56 PM
Copied to:	C:/Users/Martin/Desktop/temp/Example project/Example project.srcs/
Read-only:	No
Encrypted:	No
Core Container:	No

Used In

Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAM	URAM	DSP	Start	Elapsed	Run Strategy	Report Strategy	Part	Description
synth_1	constrs_1	Not started															Vivado Synthesis Defaults (Vivado Synthesis 2020)	Vivado Synthesis Default Reports (Vivado Synthesis 2020)	xc7a50ticsg324-1L	Vivado Synthesis Defaults
impl_1	constrs_1	Not started															Vivado Implementation Defaults (Vivado Implementation 2020)	Vivado Implementation Default Reports (Vivado Implementation 2020)	xc7a50ticsg324-1L	Default settings for Implementation.