**A Brief Tutorial for Xilinx Vivado**

## VE270 TA Group

**NOTE: The following tutorial is based on Vivado version 2017.2, you can now download Vivado version 2018.1 from the official website. The usage is similar.**

1. **Introduction**

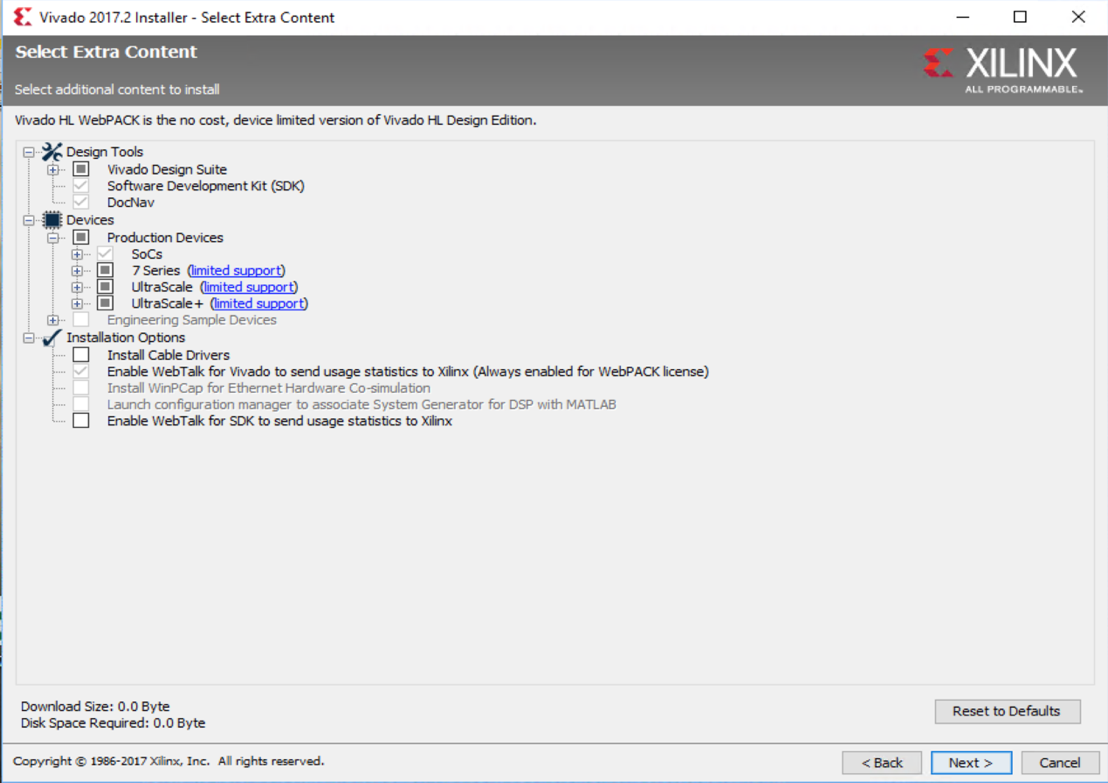
Xilinx Vivado is a newer version of Xilinx software. In this semester, it’s mainly used for burn the program to the board as well as Verilog programming.

1. **Installation**
2. Download & Sign Up

Download ***Vivado HL WebPACK Edition*** from [Xilinx Website](https://www.xilinx.com/products/design-tools/vivado/vivado-webpack.html) (xilinx.com).

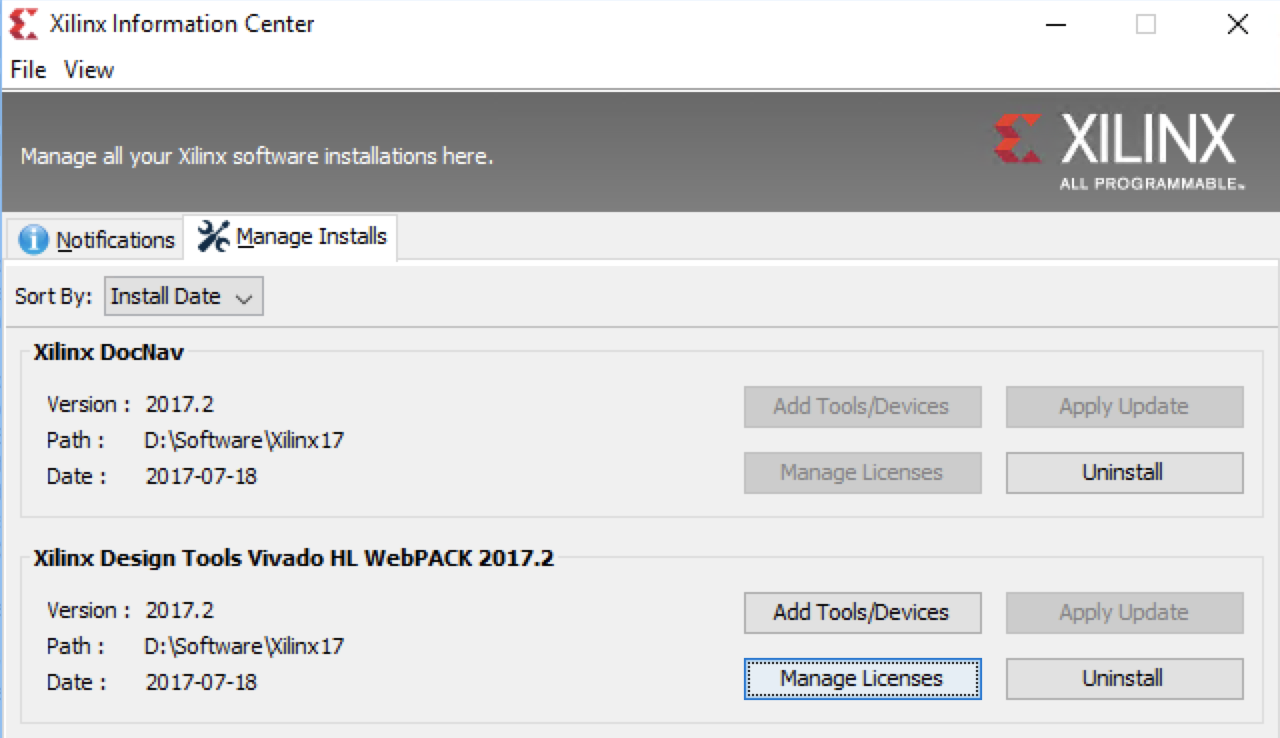
***Vivado HL WebPACK Edition*** is free to use, but you need to [sign up](https://www.xilinx.com/registration/create-account.html) at xilinx.com.

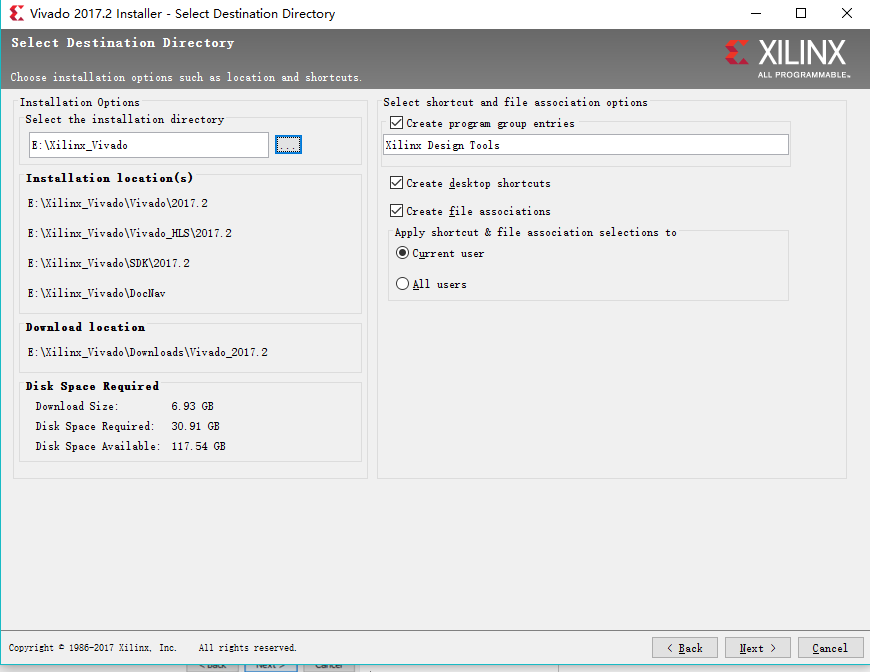
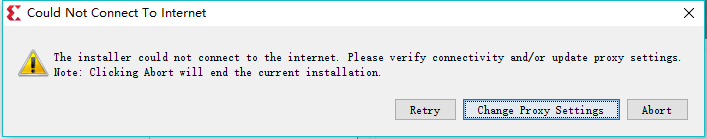
1. Install Vivado WebPack with the downloaded installer. You could choose the tools you need, but default tools are enough for our labs.



Tips:

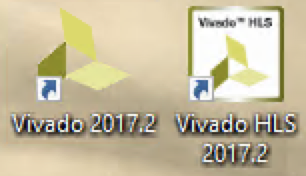
1. If you want to modify the installed content (for example like SDK etc.), you could manage them through **Xilinx Information Center – Add Tools/Devices**.



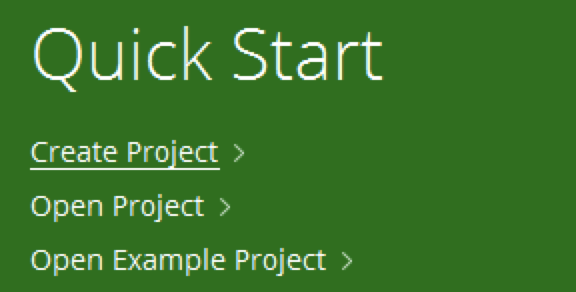
1. If you want to modify the installation directory, make sure it does not contain any spacing inside the directory. Namely “E:\Xilinx Vivado” is illegal and the correct way is shown in the picture below.
2. When downloading, there may be warnings as the figure below,

If the installer is still running and the progress bar is progressing, please just ignore it or click “Retry” several minutes later, it will not affect the downloading process.

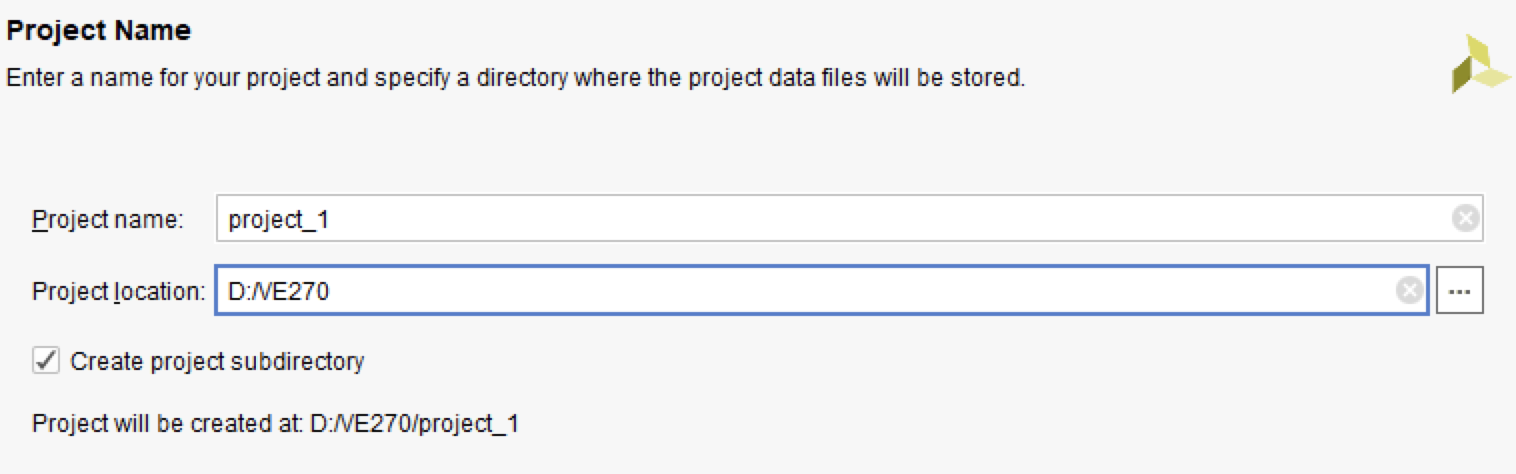
1. If you find the downloading speed to be slow, connect to SJTU VPN may help.
2. **Create Project**
3. Open **Vivado** to open the software.



1. Click **Create Project** in the left side panel.

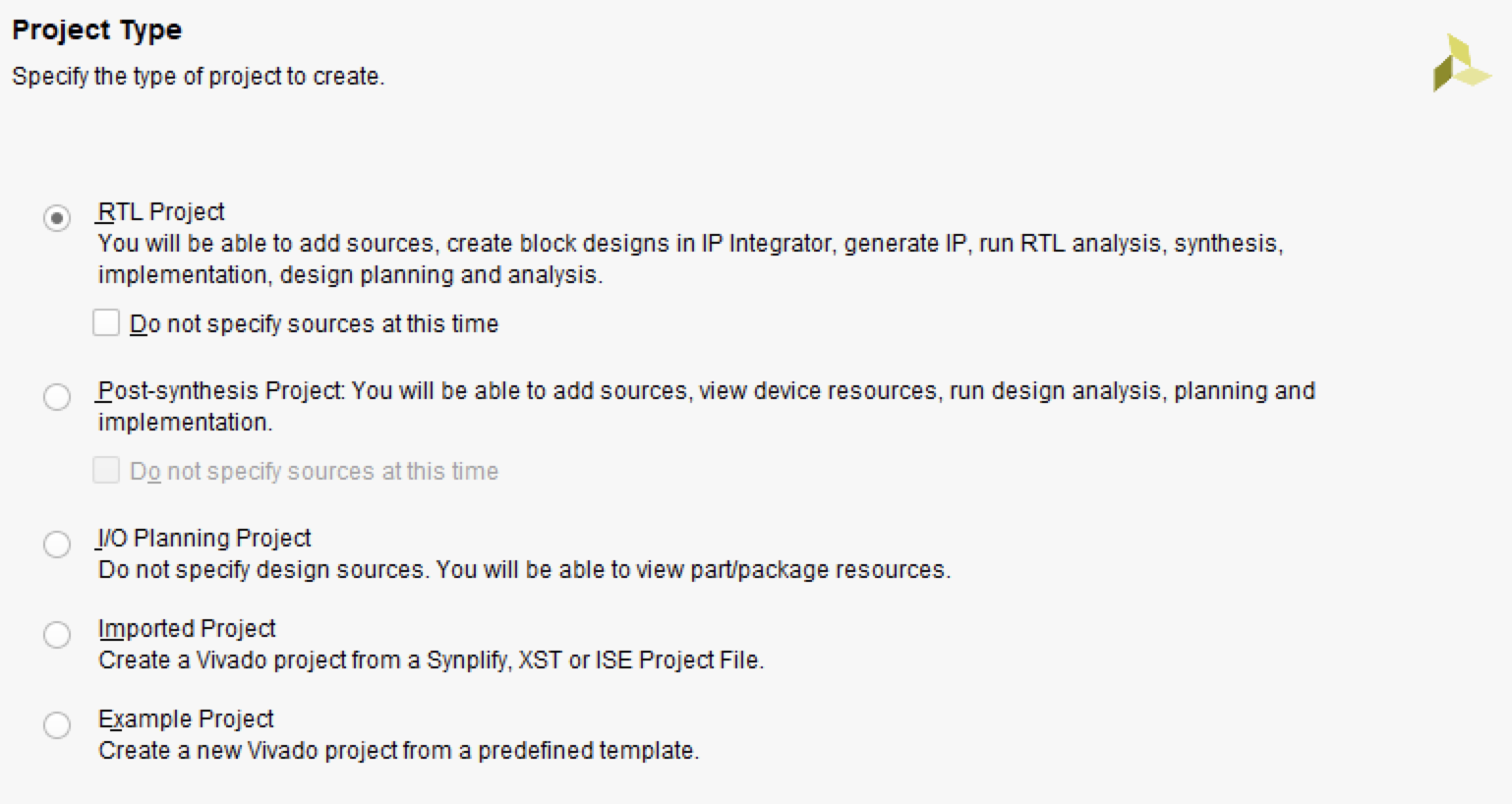


1. Follow the navigator to choose **project file location and project name**:



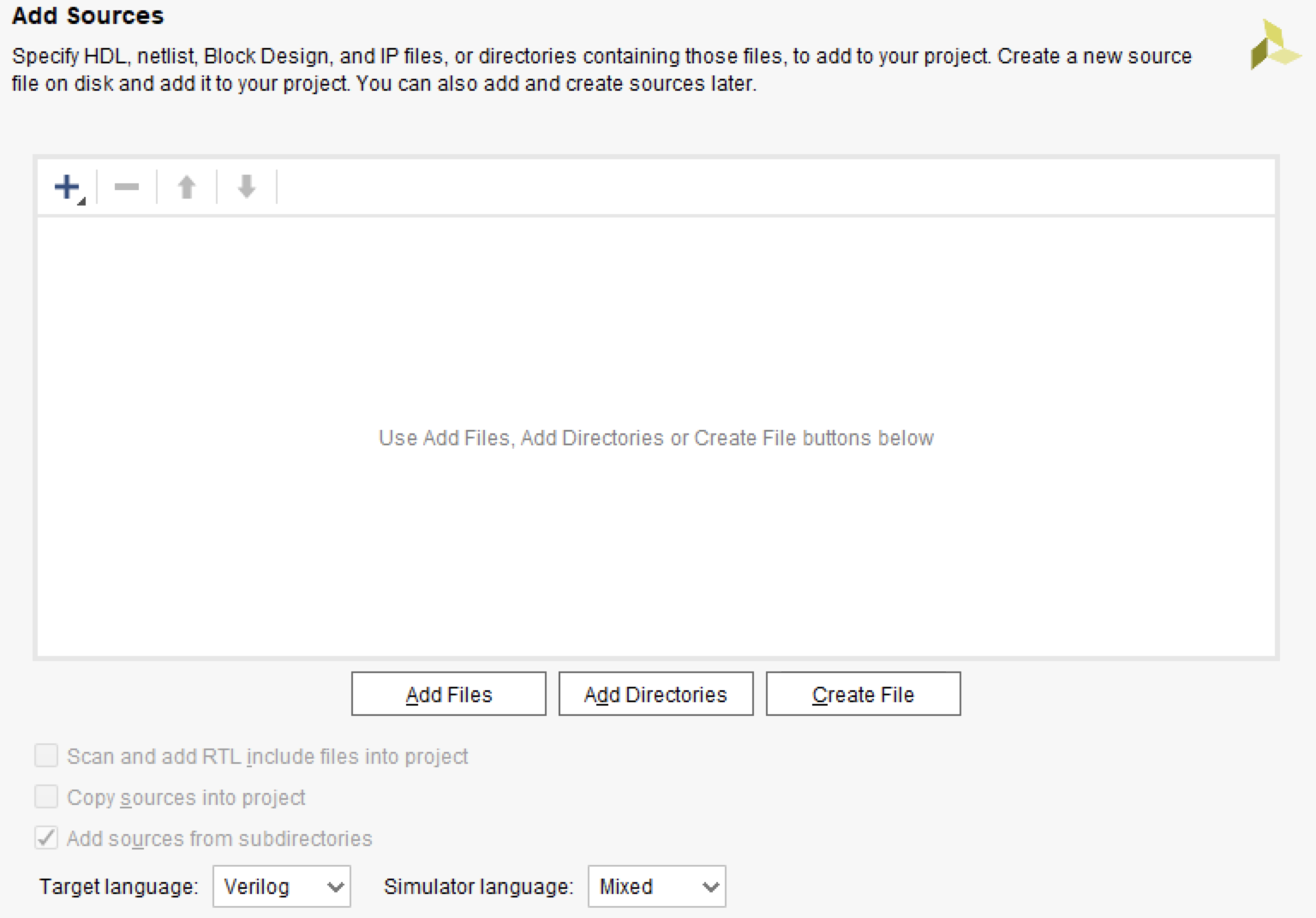
**Project Type**:

Select **RTL Project** option in the *Project Type* form, and click **Next**.



you could check “Do not specify sources at this time” box if you want to add source file later.

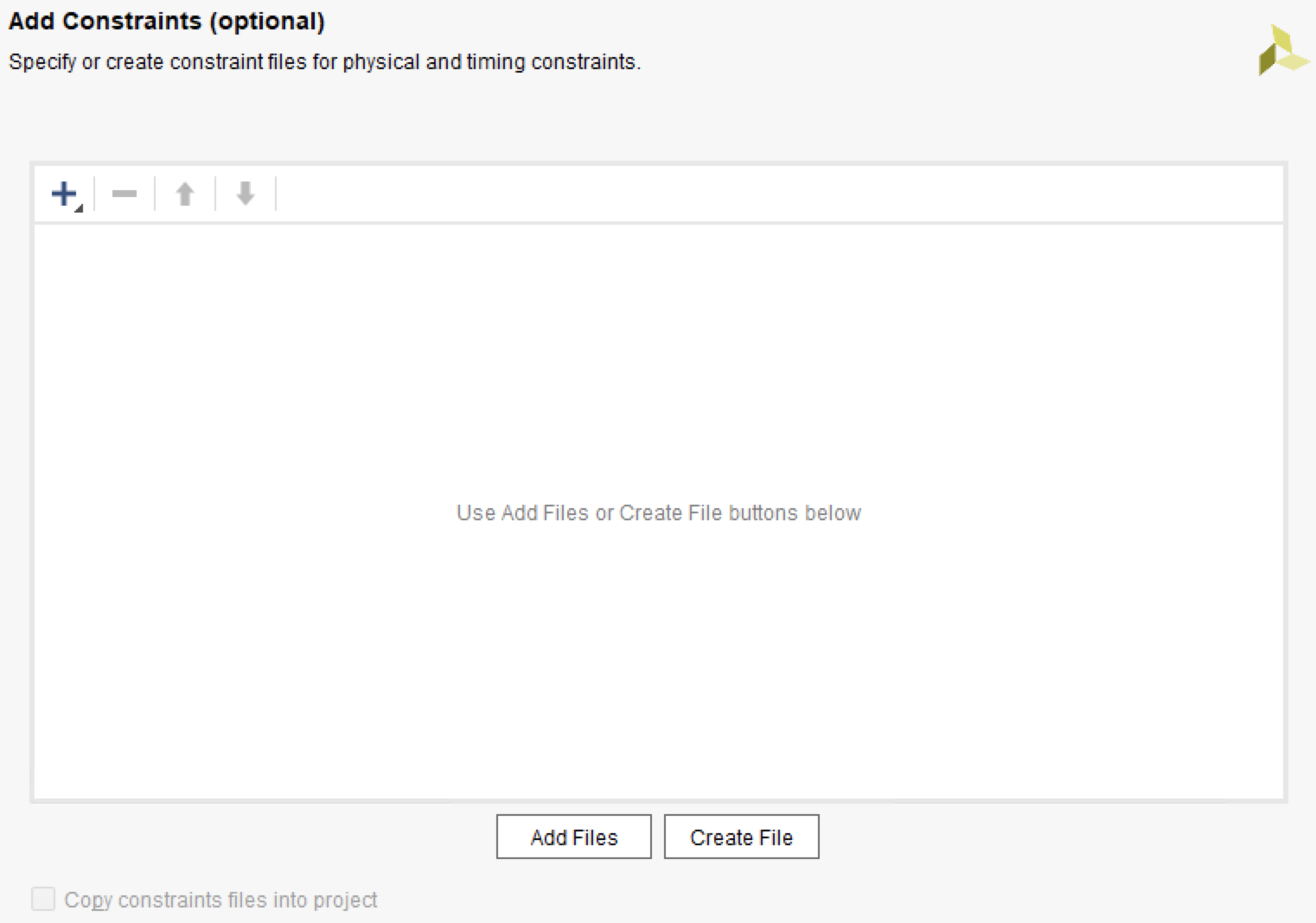
**Add Source (Can Skip):**



Optional:

Using the drop-down buttons, select **Verilog** as the *Target Language* and *Simulator Language* in the *Add Sources* form.

**Add Constrain (Can Skip):**



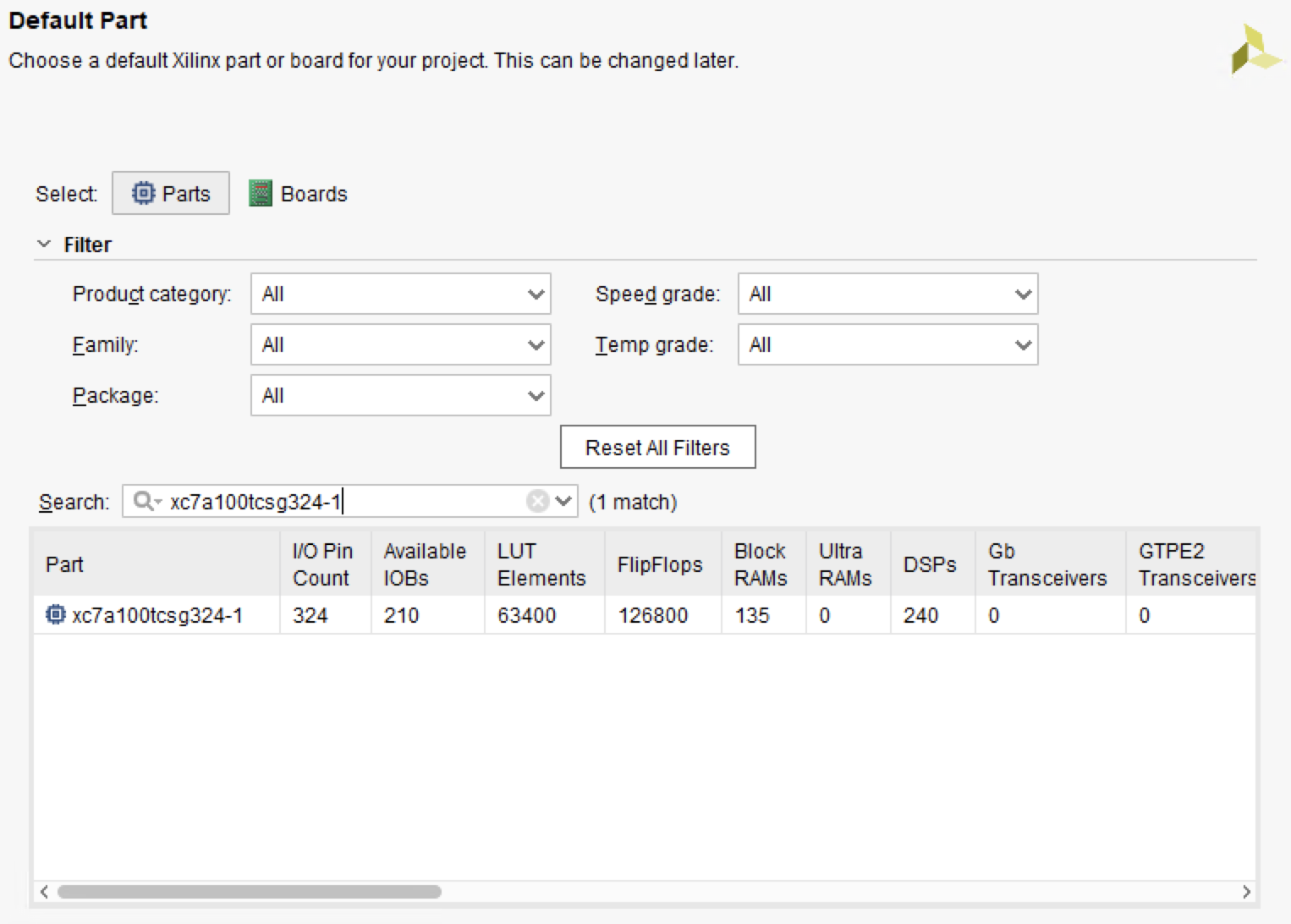
**Choose Default Part:**

There are two ways to choose parts, select by Parts or by Boards:

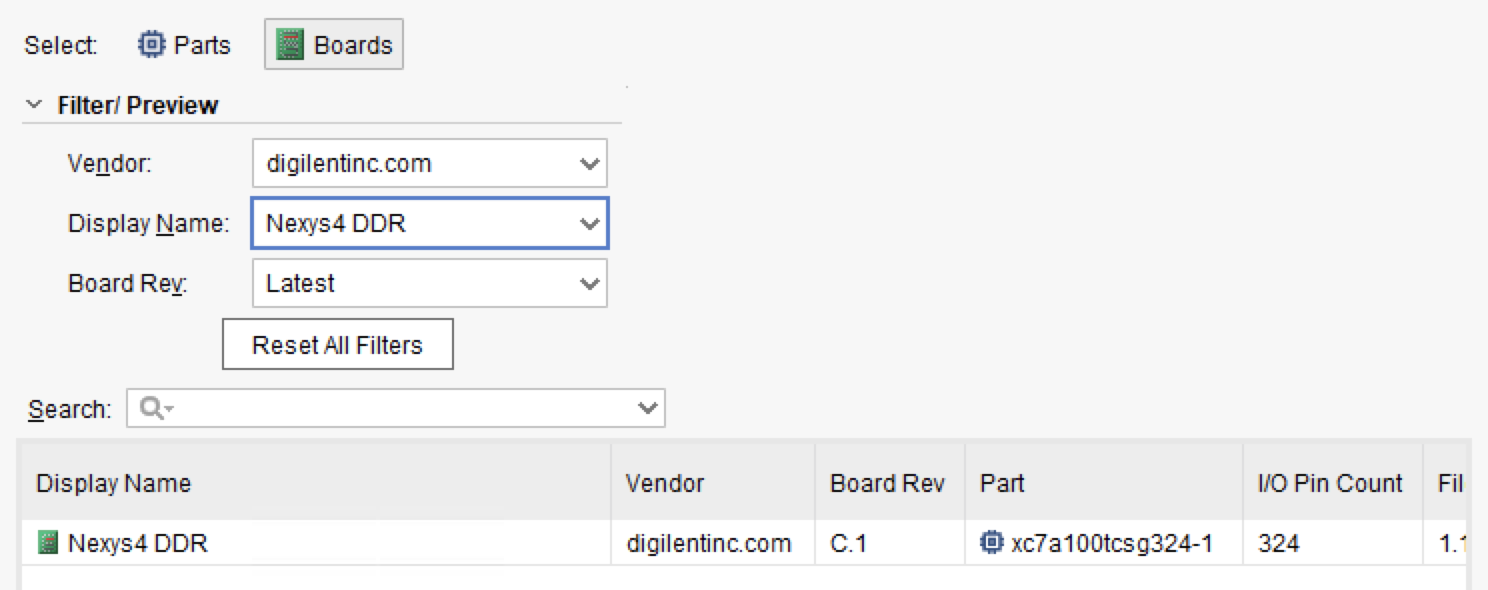
1. **Select by Parts:**

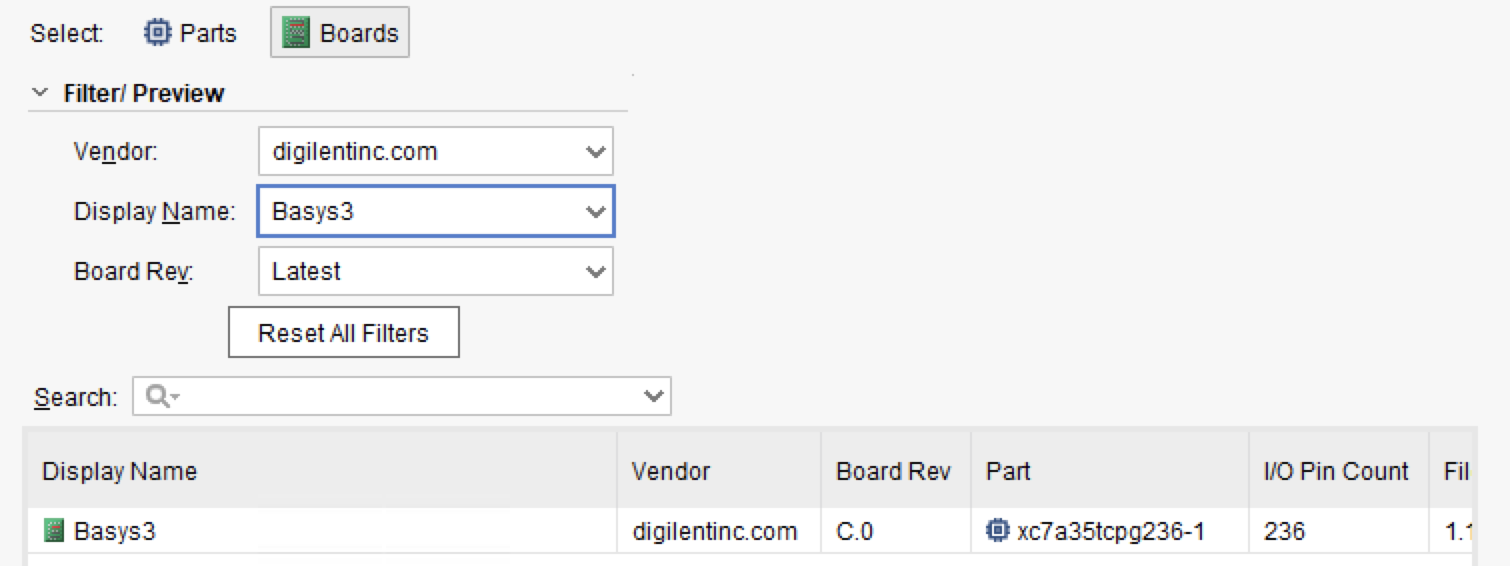
Search and Choose parts:

**XC7A35TCPG236-1 for Basys3, or XC7A100TCSG324-1 for Nexys4 DDR.**

****

1. **Select by Boards:**
2. According to the instruction from [Digilent](https://reference.digilentinc.com/learn/software/tutorials/vivado-board-files/start) (digilentinc.com) website, download Digilent Vivado boards file from [github.com](https://github.com/Digilent/vivado-boards).
3. Copy folders in “new/board\_files” into “*vivado install location*/Vivado/*version*/data/boards/board\_files”. For detail, please refer to the instruction on [Digilent website](https://reference.digilentinc.com/reference/software/vivado/board-files?redirect=1).
4. You will find board file for **Basys3** or **Nexys4 DDR:**





**Finish Creating Project:**

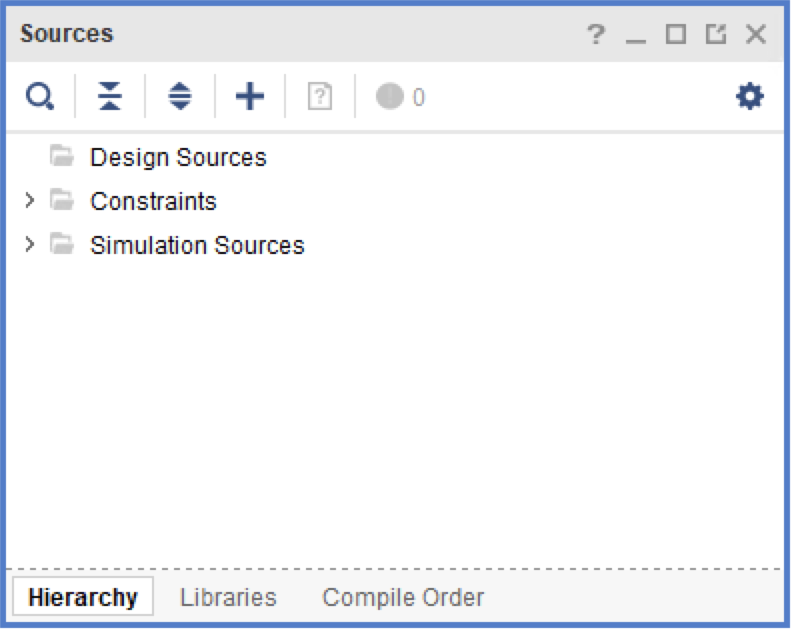
Then a new project is created.

1. **Design and Simulation**
2. **IP Integrator Design**

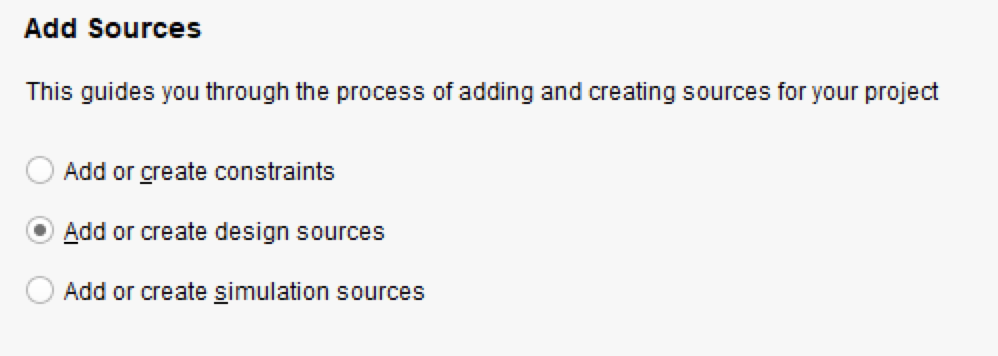
Please refer to [Vivado Design Hub – Using IP Integrator](https://www.xilinx.com/support/documentation-navigation/design-hubs/dh0009-vivado-using-ip-integrator-hub.html) and [Vivado Design Suite Tutrial: Designing IP Subsystems Using IP Integrator](https://www.xilinx.com/support/documentation/sw_manuals/xilinx2017_2/ug995-vivado-ip-subsystems-tutorial.pdf).

1. **Verilog Programmatically Design**

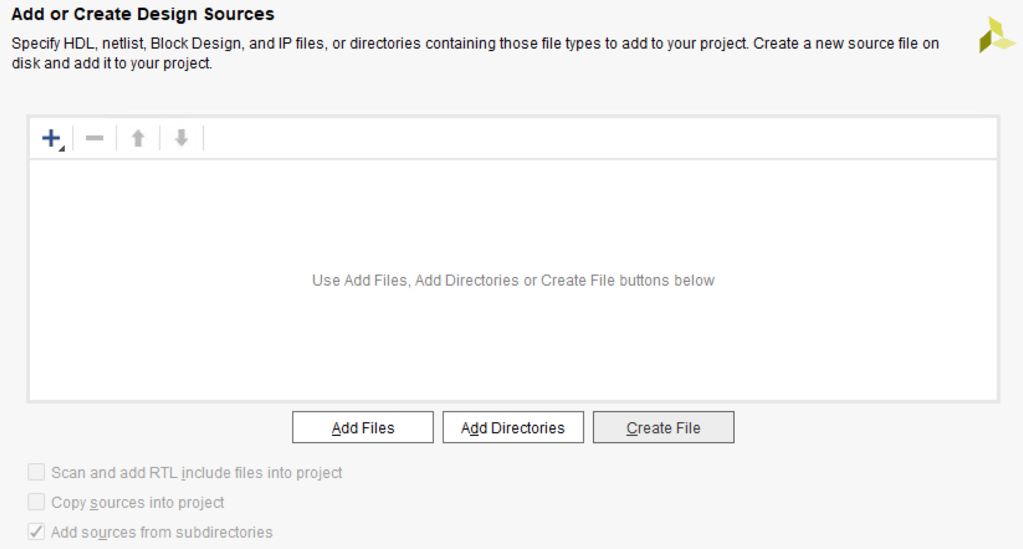
Find source windows and click “+” to add source.



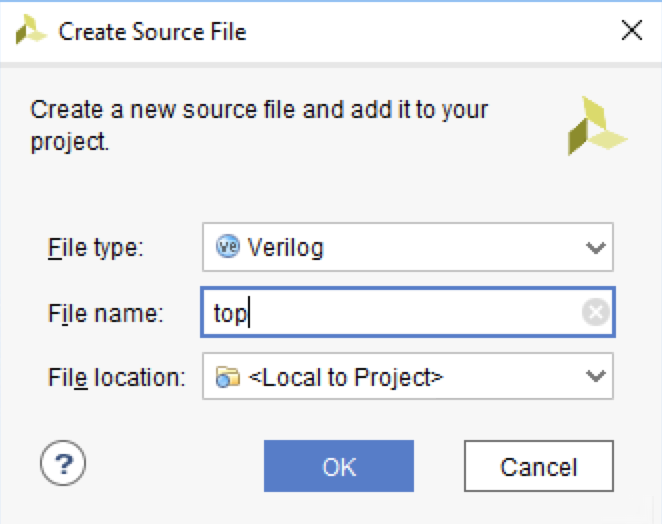
Choose “Add or create design sources”.



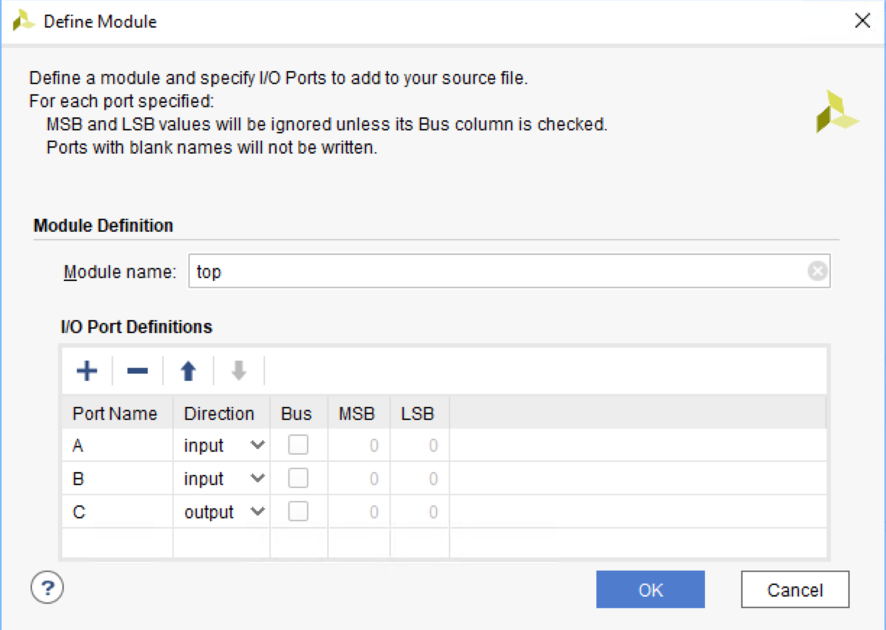
Click Create Files



Choose File type to be Verilog, enter File name.



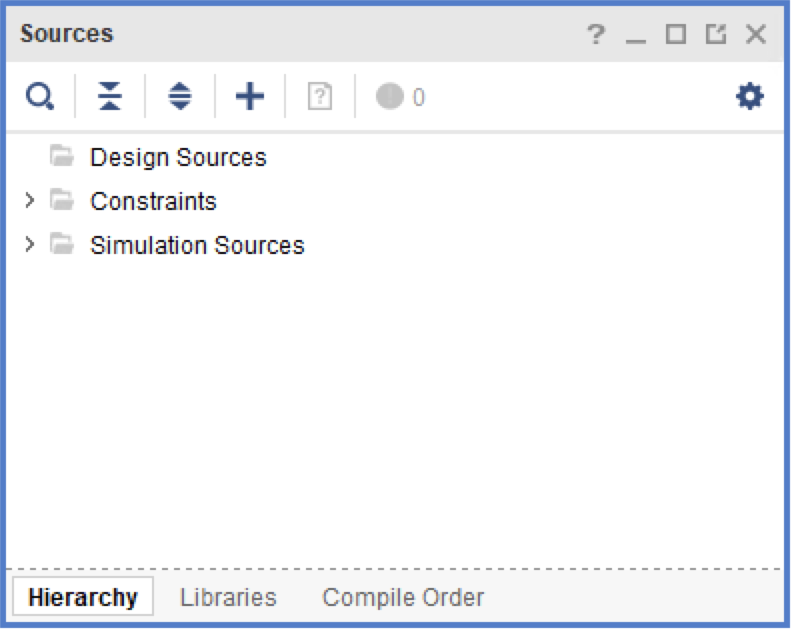
Click “OK” and “Finish”, then a “Define Module” window will pop up. You can skip this step, or you can also name your module ports here.



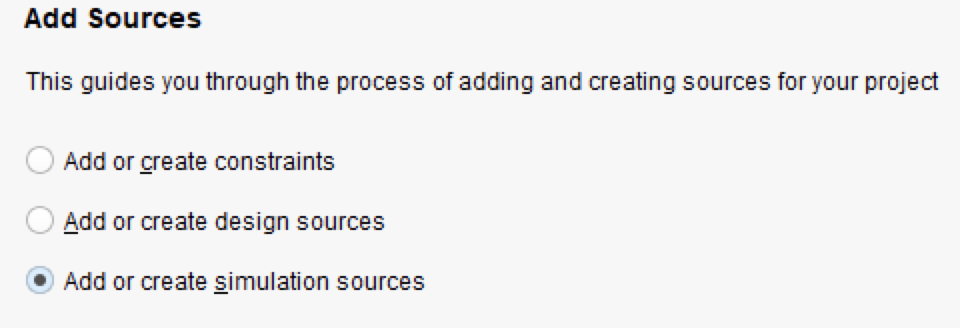
Then you can code in your Verilog file to implement your design.

1. Create Simulation

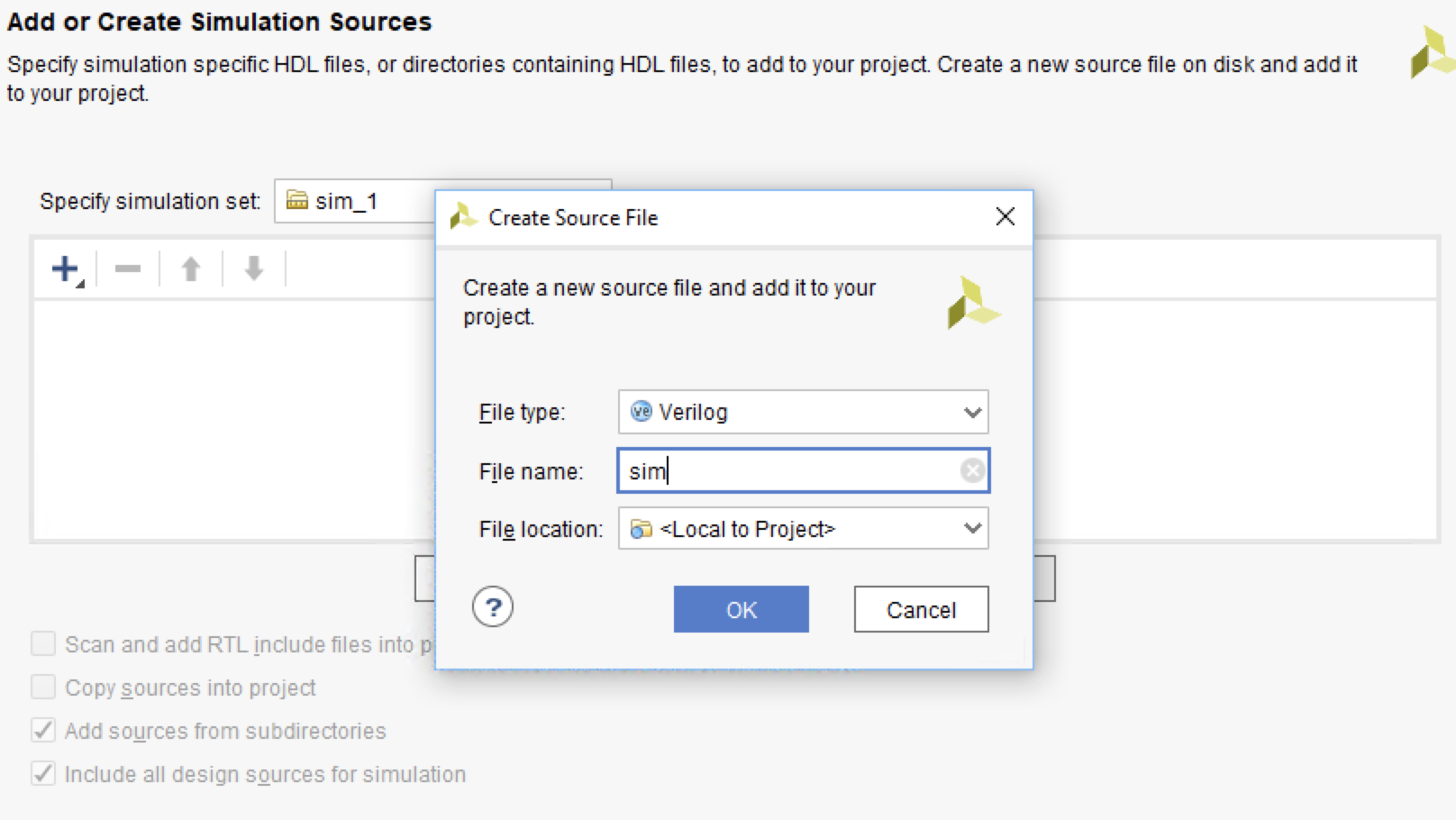
Find source windows and click “+” to add source.



Choose “Add or create design sources”.



Click “Create File”, Name file and choose location.



Click OK and Finish.

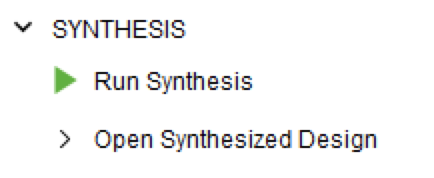
Open the simulation Verilog file in Source Hierarchy to write your simulation.



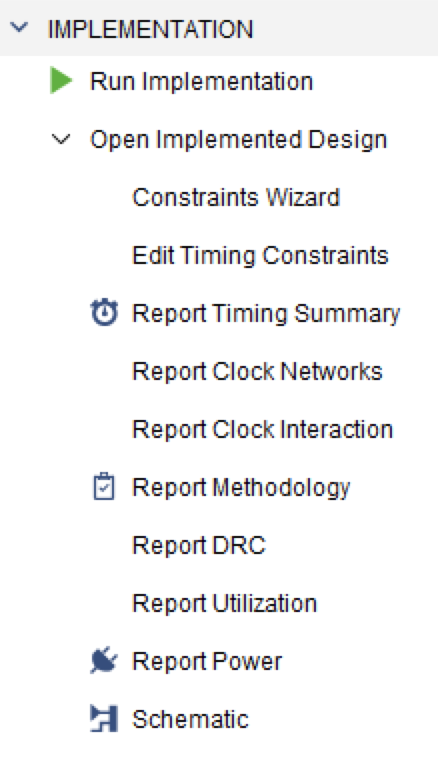
Choose corresponding simulation file and run simulation by clicking “Run Simulation” in the left **Flow Navigator** bar and get the result.

1. **Implementation on FPGA Board**
2. Generate .bit file

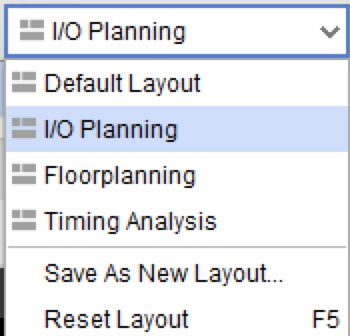
Run Synthesis in Flow Navigator with default setting.



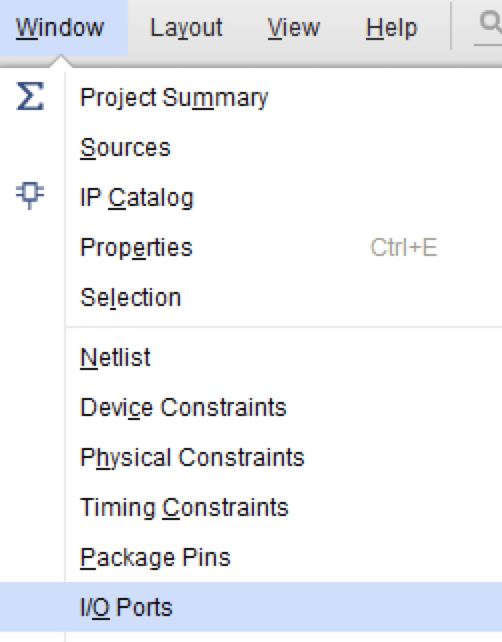
Click “**Run Implementation**” in Flow Navigator.

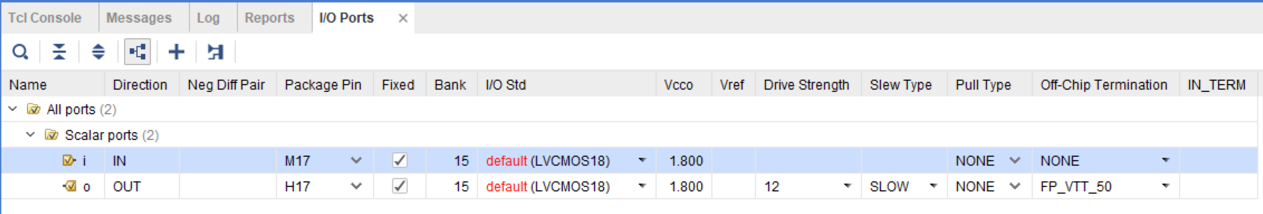


Choose “**I/O Planning**” of panel view in upper right drop down menu.



Select “**I/O Port**” tab in the bottom window. You can open this window in “Window->I/O Port”:





Then you can assign input and output pin number according to board ***Reference Manual*** to “**Package Pin**” column.

Save the change and rerun **Synthesis** and **Implementation.**

Connect your board to your computer. And click **Generate Bitstream**. Allow device auto connect.

