Transition Guide from Altera Quartus to Xilinx Vivado

Introduction to Xilinx and Vivado

Xilinx (now part of AMD) is a leading company in the field of programmable logic devices, best known for its development of **FPGAs (Field Programmable Gate Arrays)**, **SoCs (System on Chips)**, and **adaptive compute acceleration platforms (ACAPs)**. Founded in 1984 and now part of AMD (since 2022), Xilinx has been a pioneer in reconfigurable computing.

To learn more:

https://en.wikipedia.org/wiki/Xilinx

Vivado Design Suite is Xilinx's flagship development environment for FPGA and SoC (System-on-Chip) designs. Unlike Quartus, which is used for Intel (formerly Altera) FPGAs, Vivado offers a more advanced and integrated workflow for synthesis, simulation, implementation, and debugging.

To learn more:

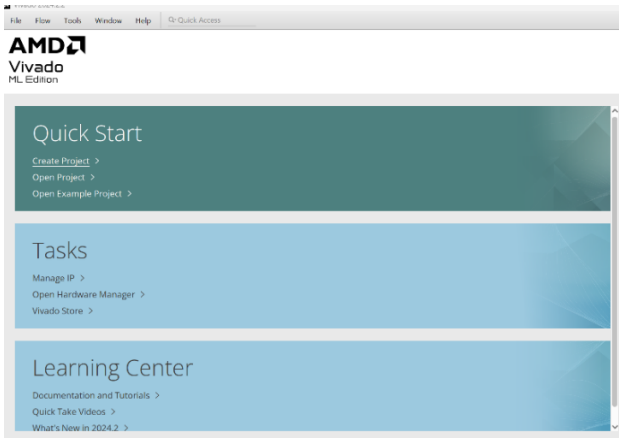
https://docs.amd.com/v/u/en-US/dh0019-vivado-implementation-hub

Vivado Features

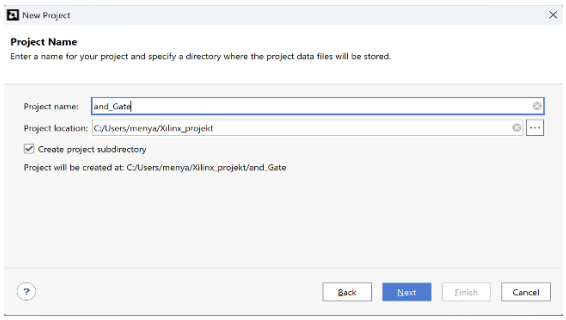
- HDL Design: Supports VHDL and Verilog
- Block Design (IP Integrator): Graphical design with prebuilt IP blocks
- Synthesis and Implementation: Converts RTL code into hardware netlists
- Simulation: Integrated simulator for functional verification
- Bitstream Generation: Creates the file used to program the FPGA
- Hardware Debugging: Includes tools like Integrated Logic Analyzer (ILA)

Getting Started with Vivado

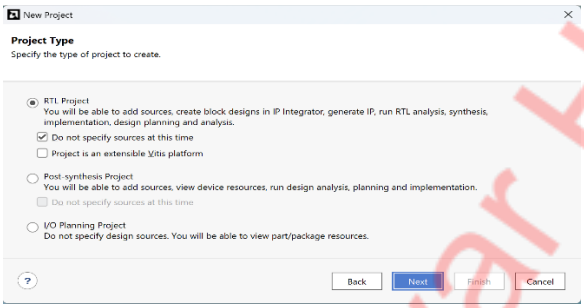
1. **Open Vivado**
   Click on Create New Project



2. **Enter project name and location**
   Project names are case-sensitive and should not contain space or special characters (e.g., ! @ # $). Use letters, numbers, and underscores only (e.g., led_blinker_01).

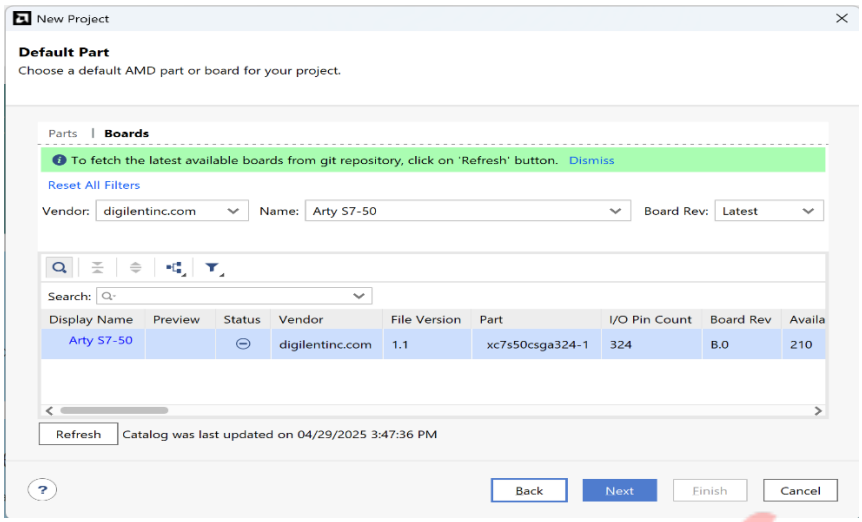


3. **Choose RTL Project**
   Use this for custom digital logic design (not just use prebuilt IPs).

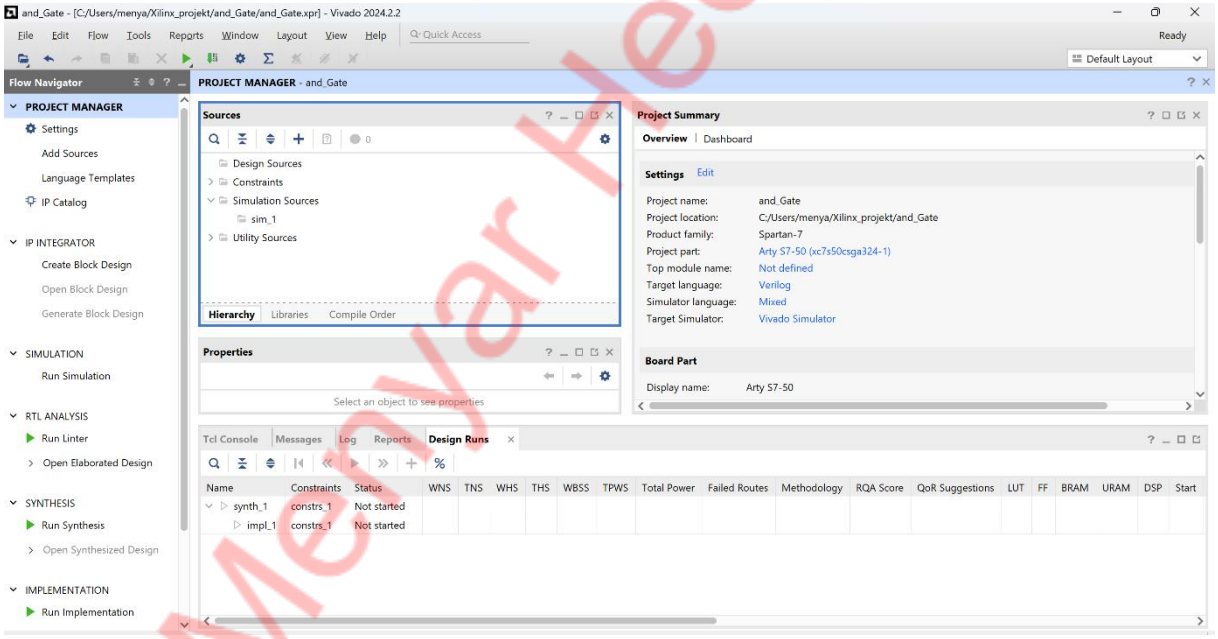


4. **Select the correct FPGA board**

   Click on the → "**Boards**" tab to choose the development board.
   Then click → "**Refresh**" to update the list.
   From the Vendor dropdown, select → "**Digilentinc.com**".
   Find and select "**Arty S7-50**", then click on → "**Install**" next to the board name (if needed).
   Finally, click "**Next**", then "**Finish**" to complete project setup.



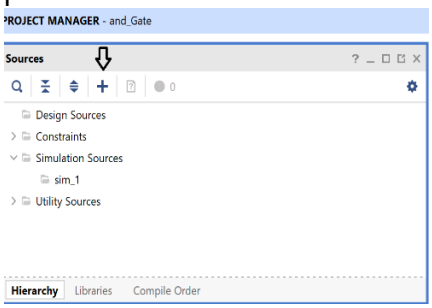
After creating a new project, Vivado opens the **main interface**, which includes the following key sections:



- **Flow Navigator (left side):**
  A step-by-step guide to the FPGA design flow, including design entry, simulation, synthesis, implementation, and bitstream generation.
- **Project Manager (center):**
  Shows design sources (HDL files), constraints, simulation files, and project hierarchy.
- **Project Summary (right side):**
  Displays information about the project settings, selected board/part, top module, languages used, and simulator type.
- **Properties & Design Runs (bottom):**
  Displays properties of selected files and shows the progress of synthesis and implementation runs.
  This layout helps you manage your design files, analyze, simulate, and implement your digital logic design efficiently.

5. **Add sources (VHDL/Verilog files)**

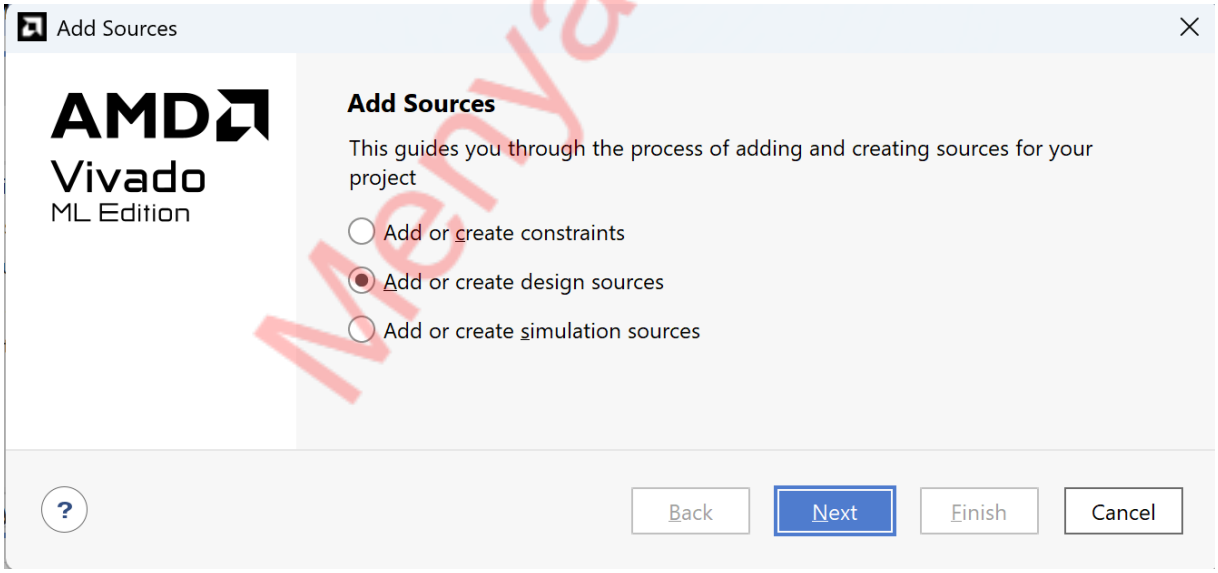
   First, click on the plus sign in the" **Sources**" box to add new files as shown in the picture below.



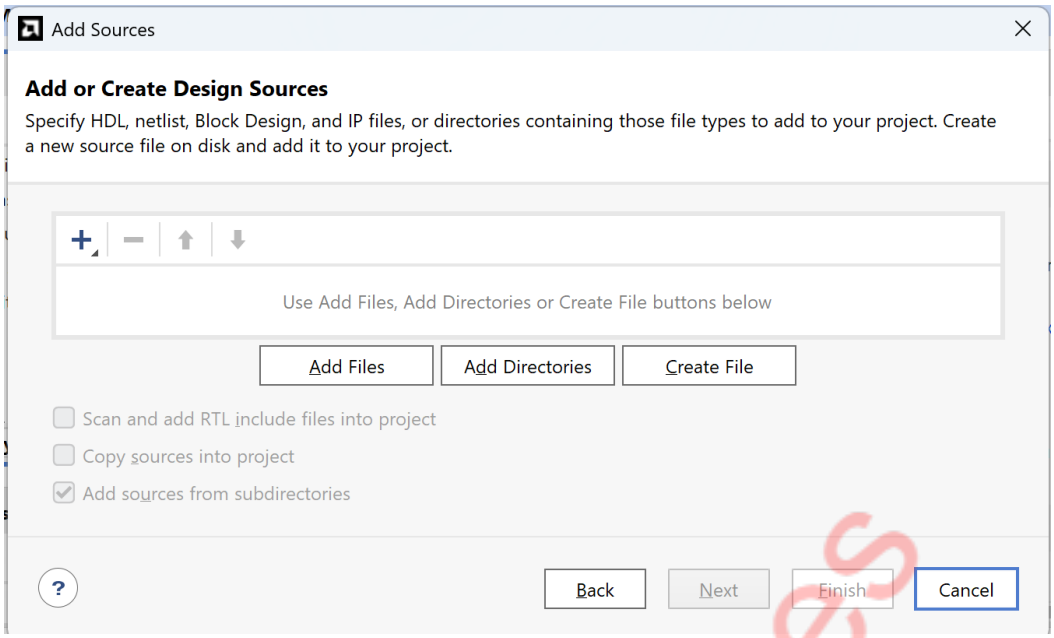
In next step, you can add or create different types of source files for your project:

- **Add or create design sources:** Add or write your main hardware design files (VHDL or Verilog). These describe the actual logic of your circuit (e.g., counters, state machines).
- **Add or create simulation sources:** Add testbench files used only for simulation to verify the functionality of your design. These are not part of the final hardware.
- **Add or create constraints**: Add constraint files (e.g., .xdc) to define FPGA pin assignments and timing requirements, like assigning a clock to a specific pin**.**

(You can choose to add files now or skip and add them later from within Vivado).



For now, we will choose **Add or create design sources** which is equivalent to add new file, VHDL template in Quartus.



This window lets you add your main VHDL or Verilog files to the project.

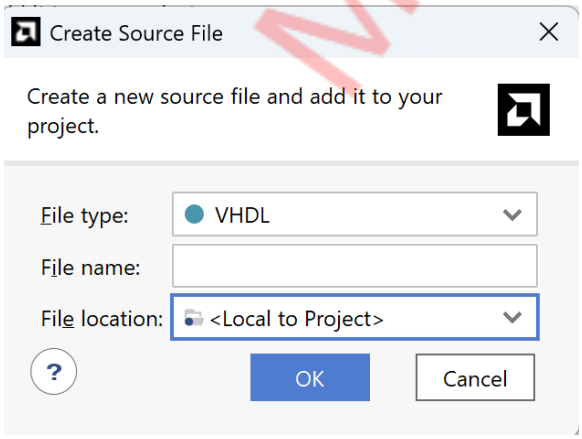
You have three main options:

- **Add Files**: Browse and add existing VHDL/Verilog files from your computer.
- **Add Directories**: Add an entire folder containing source files.
- **Create File**: Create a new source file directly in Vivado.

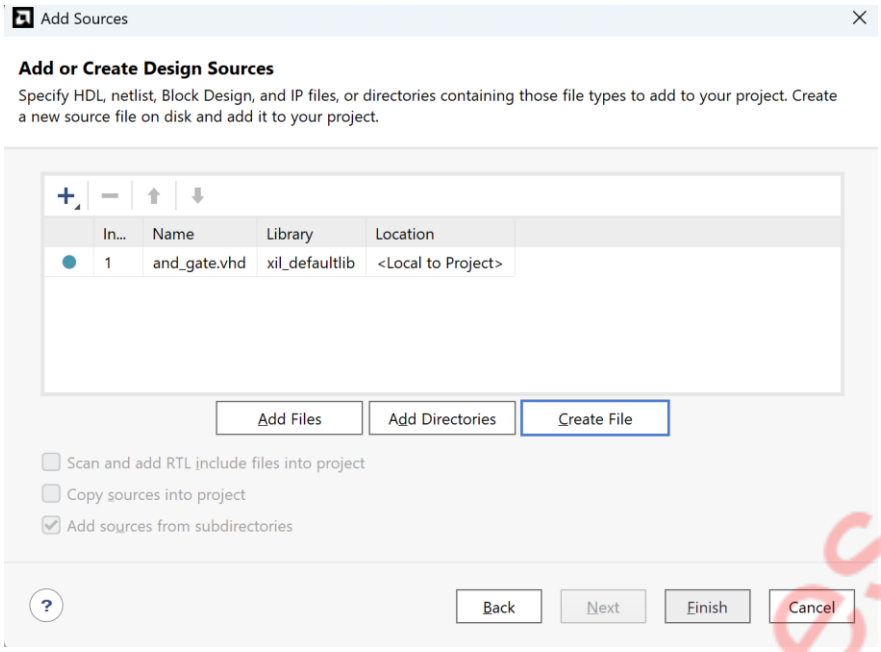
You can also choose to:

- Include files from subdirectories (checked by default).
- (Optional) Copy the files into the project directory.

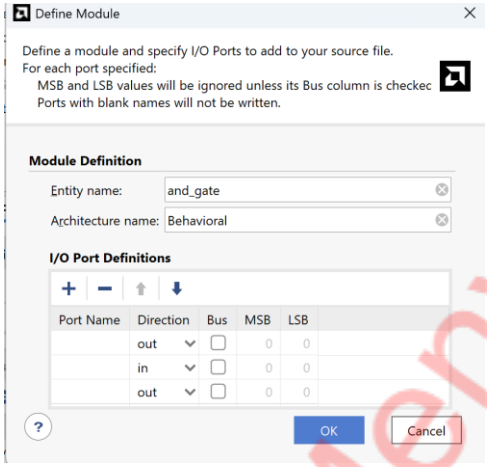
After selecting or creating your files, click **Next** to specify the file name, location and hardware description languages you want to use, in our case we will choose VHDL.



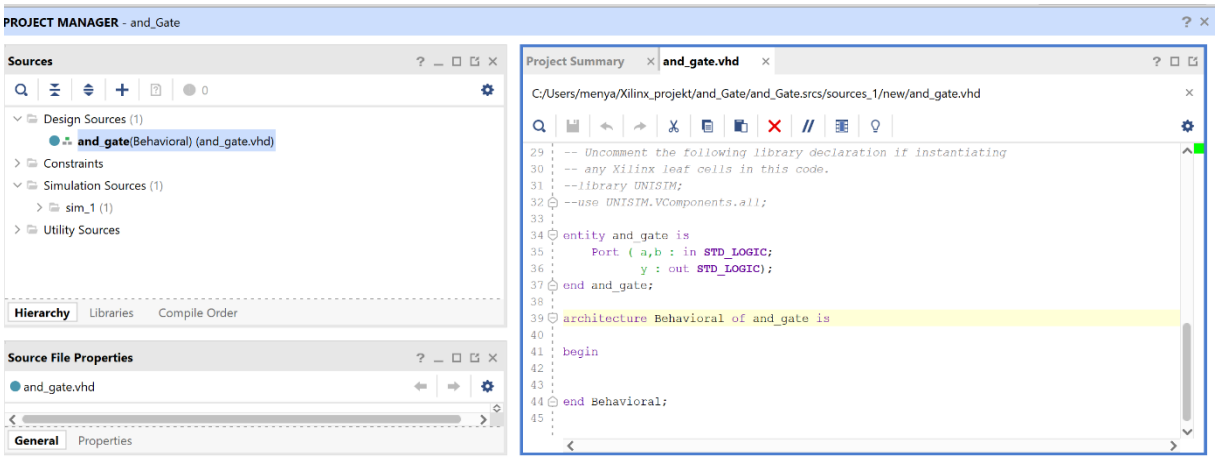
Now you just need to click on **Finish**.



In the next step you can get a ready-made template for your design If you want to. All you have to do is specify the name of the **Entity**, **Architecture**, and **Port**.

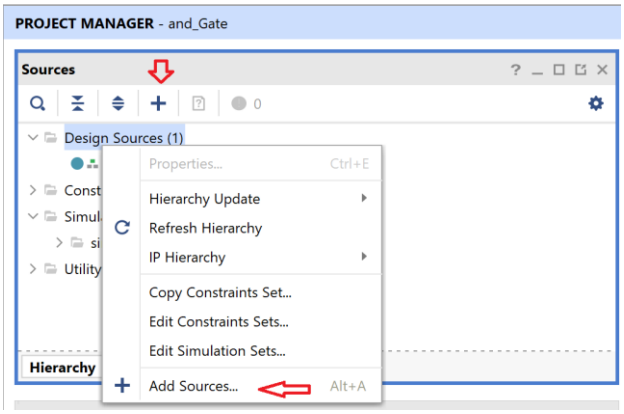


Click on **OK** and Vivado will automatically create to you the VHDL template**.** The new file will be in **Design Sources** as a top-level file (you can change that later if you want to).



Now you can easily add your own code or define the inputs and outputs in the ready-made template.

To add new Sources to your file, you can directly click on the plus sign **(+),** or right-click on **Design Sources** and choose Add Resources from the menu**.**



Congratulations! You are now ready to start working in Vivado — creating a new project, organizing your design files, and exploring the tool

In this demo video, we will go through all the previous steps

https://youtu.be/kYnRCnBfnFg

Key Differences Between Quartus and Vivado

Feature	Quartus (Intel/Altera)	Vivado (Xilinx)
Project Management	Project Navigator	Project Manager
Simulation	ModelSim (often external)	Built-in XSIM (Vivado Simulator)
Debugging	SignalTap Logic Analyzer	Integrated Logic Analyzer (ILA)
Virtual Input/Output	ISSP	Virtual I/O (VIO)
Timing Analysis	TimeQuest Timing Analyzer	Vivado Timing Analyzer
FPGA Configuration	USB-Blaster / Programmer	Vivado Hardware Manager
Project Structure	.qpf (Project File)	.xpr (Vivado Project File)
Constraints	.qsf (Quartus Settings)	.xdc (Xilinx Design Constraints)