

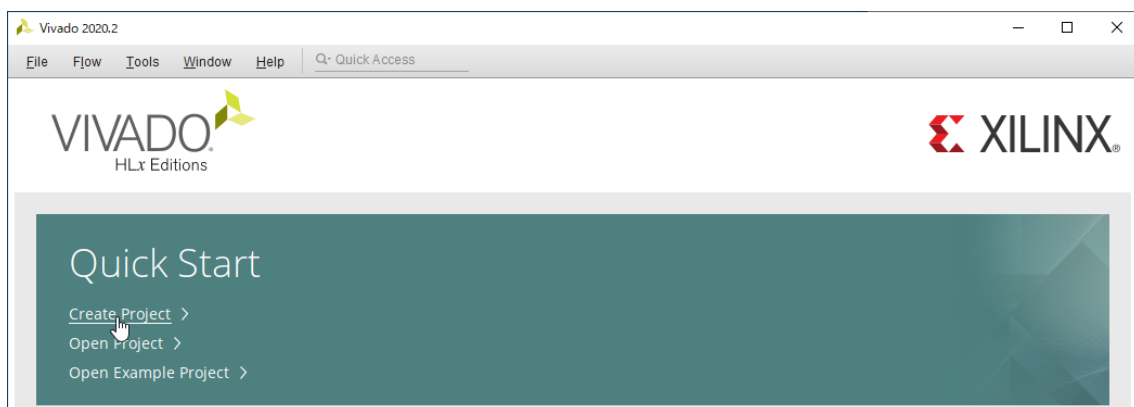
Vivado Design Suite Tutorial

Introduction

This tutorial guides you through the design flow using Xilinx Vivado software to create a simple digital circuit using Verilog HDL. You will go through the typical design flow targeting the Artix-7 based BASYS3 board.

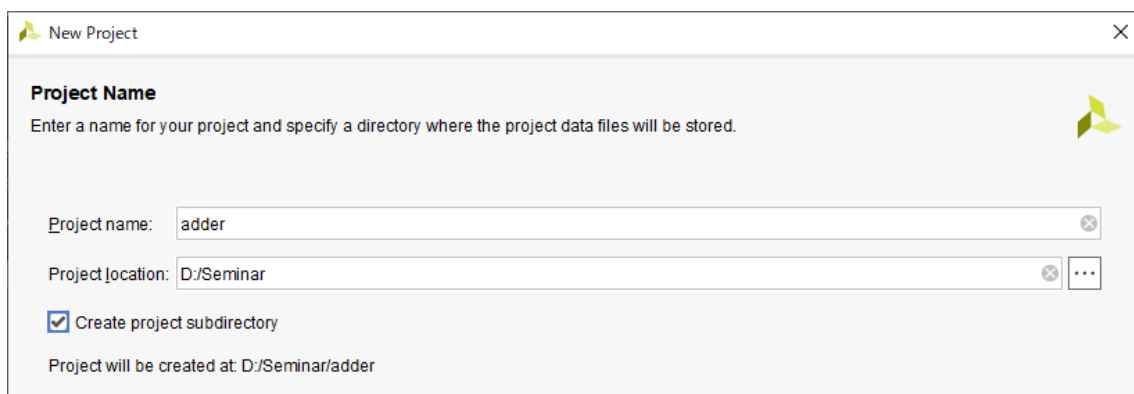
1. Creating a Vivado Project

1.1) Open up Vivado 2020.2 (not Vivado HLS) and click **Create Project** to start the wizard.

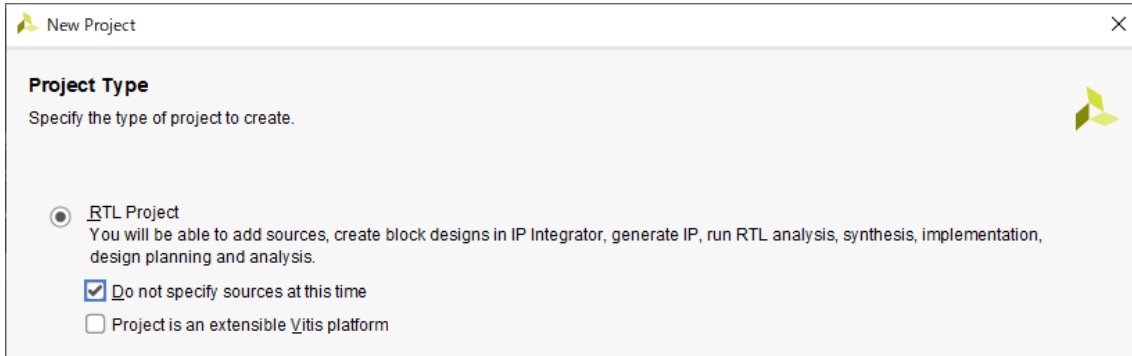


1.2) You will see the “Create a New Vivado Project” dialog box. Click **Next**.

1.3) Enter **adder** in the “Project name” field. Click the browse button of the “Project location” field, browse to your project saving directory, and click **Select**. Make sure that the “Create Project Subdirectory” box is checked. Click **Next**.



1.4) Select **RTL Project** option in the “Project Type” form and check the “Do not specify sources at this time” box. Click **Next**.



New Project

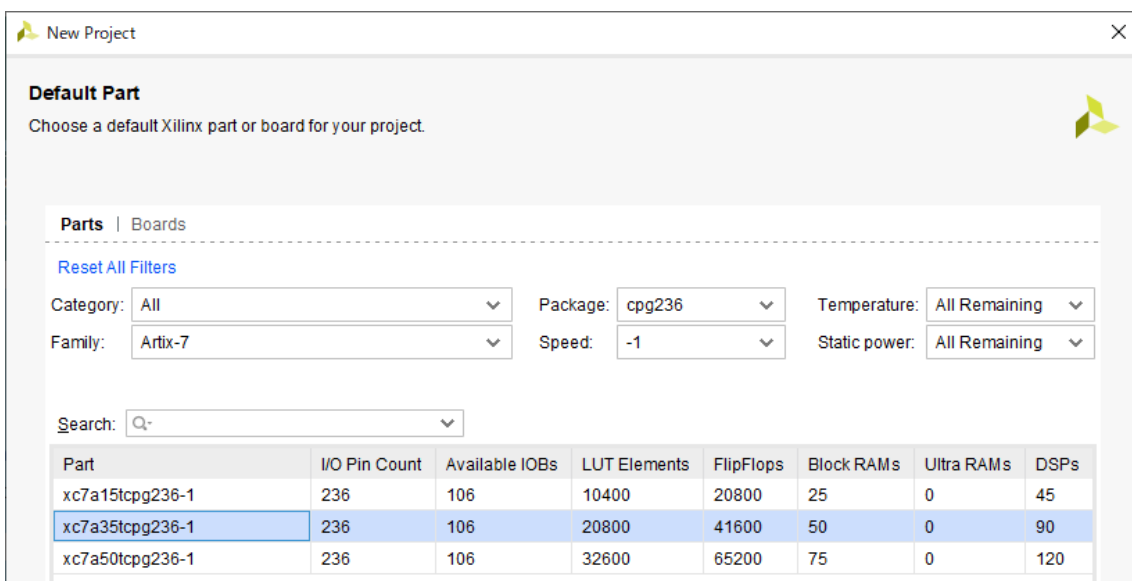
Project Type
Specify the type of project to create.

☒ **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

1.5) In the “Default Part” form, using the **Parts** option and various drop-down fields of the filter section, select the **xc7a35tcpg236-1** part. Click **Next**.



New Project

Default Part
Choose a default Xilinx part or board for your project.

Parts | Boards

[Reset All Filters](#)

Category: All Package: cpg236 Temperature: All Remaining

Family: Artix-7 Speed: -1 Static power: All Remaining

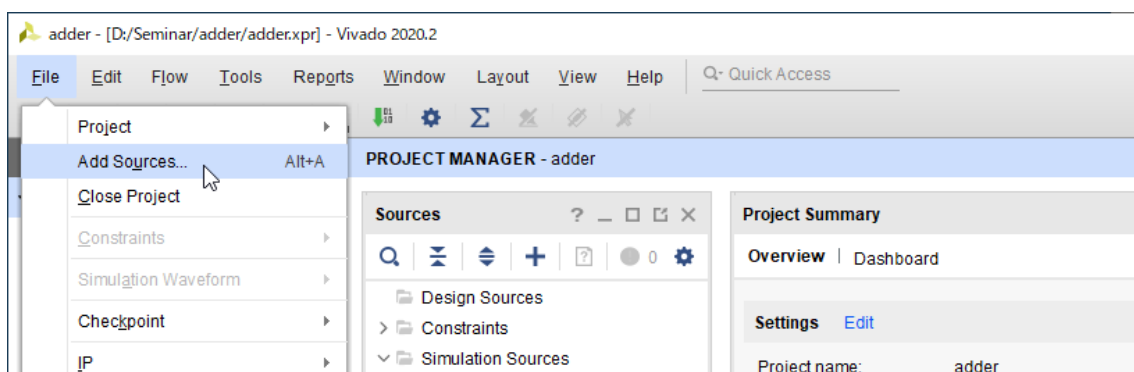
Search: Q-

Part	I/O Pin Count	Available IOBs	LUT Elements	FlipFlops	Block RAMs	Ultra RAMs	DSPs
xc7a15tcpg236-1	236	106	10400	20800	25	0	45
xc7a35tcpg236-1	236	106	20800	41600	50	0	90
xc7a50tcpg236-1	236	106	32600	65200	75	0	120

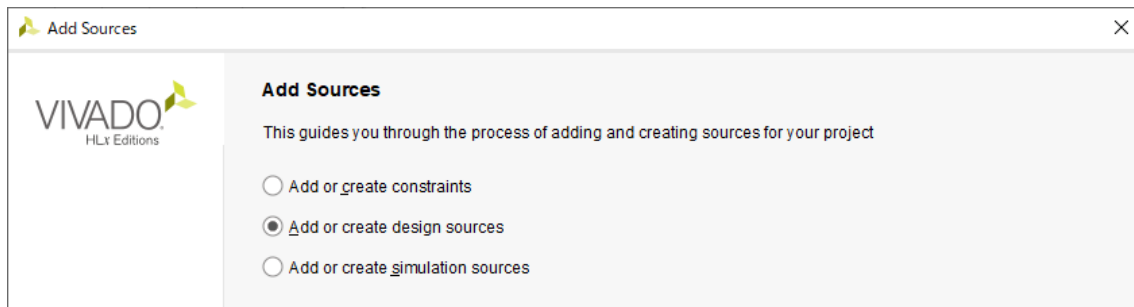
1.6) You will see “New Project Summary” form. Click **Finish** to create the project.

2. Creating a Source File

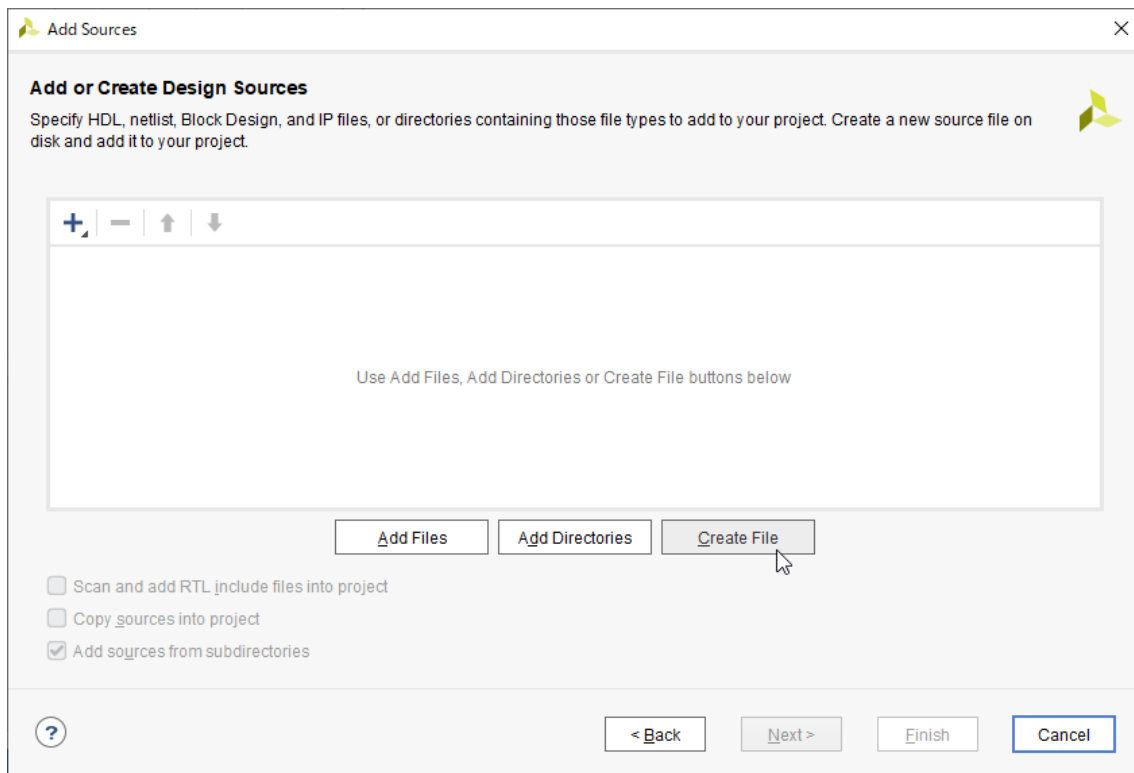
2.1) Select **Add Sources** from the File menu.



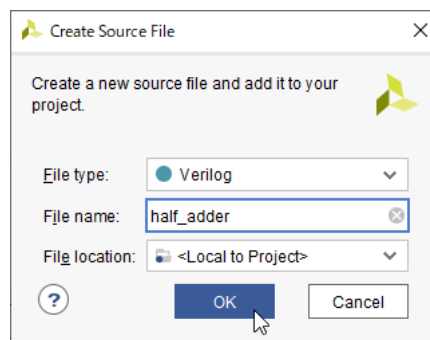
2.2) In the “Add Sources” form, select **Add or create design sources**. Click **Next**.



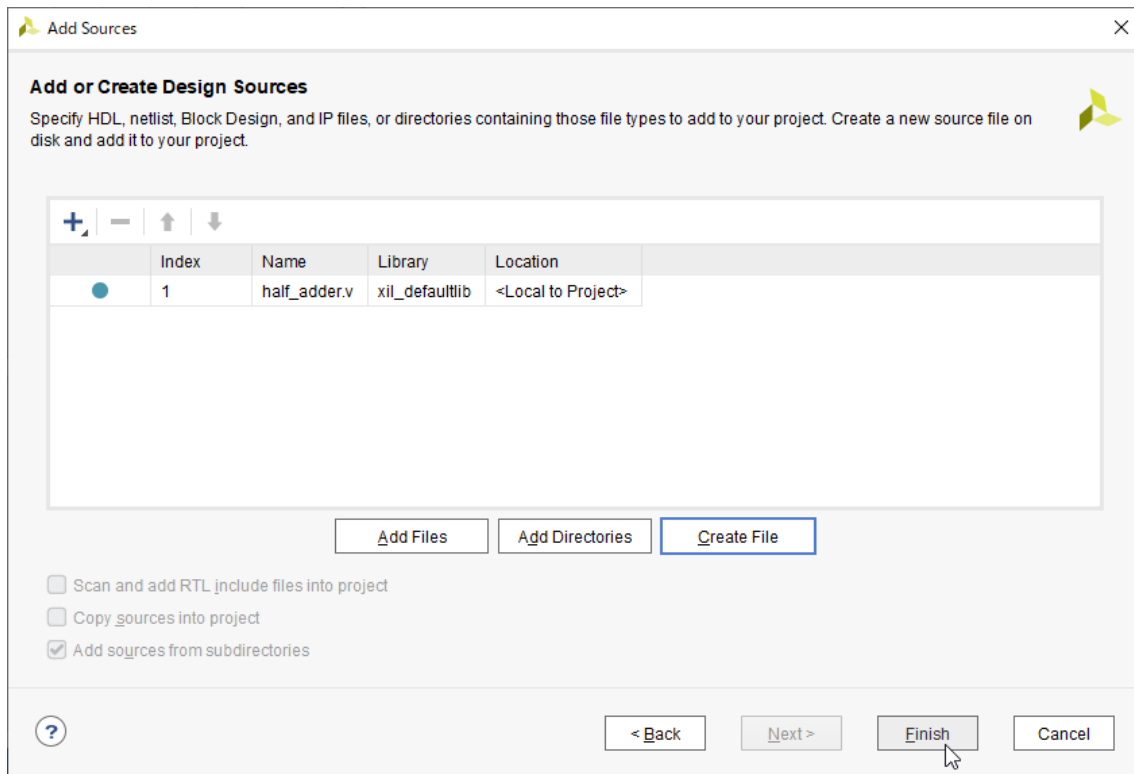
2.3) In the “Add or Create Design Sources” form, click **Create File** button.



2.4) In the “Create Source File” form, select **Verilog** in the “File type” field. Enter **half_adder** in the “File name” field. Click **OK**.



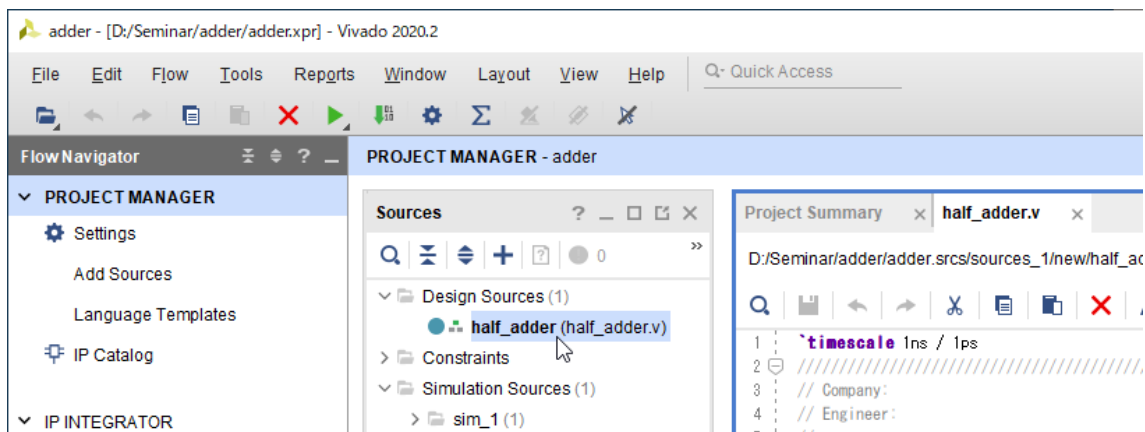
2.5) You will see the created file in the “Add or Create Design Sources” form. Click **Finish**.



2.6) Skip the “Module Definition” form by clicking **OK**.

2.7) You will see the “Define Module” dialog box. Click **Yes**.

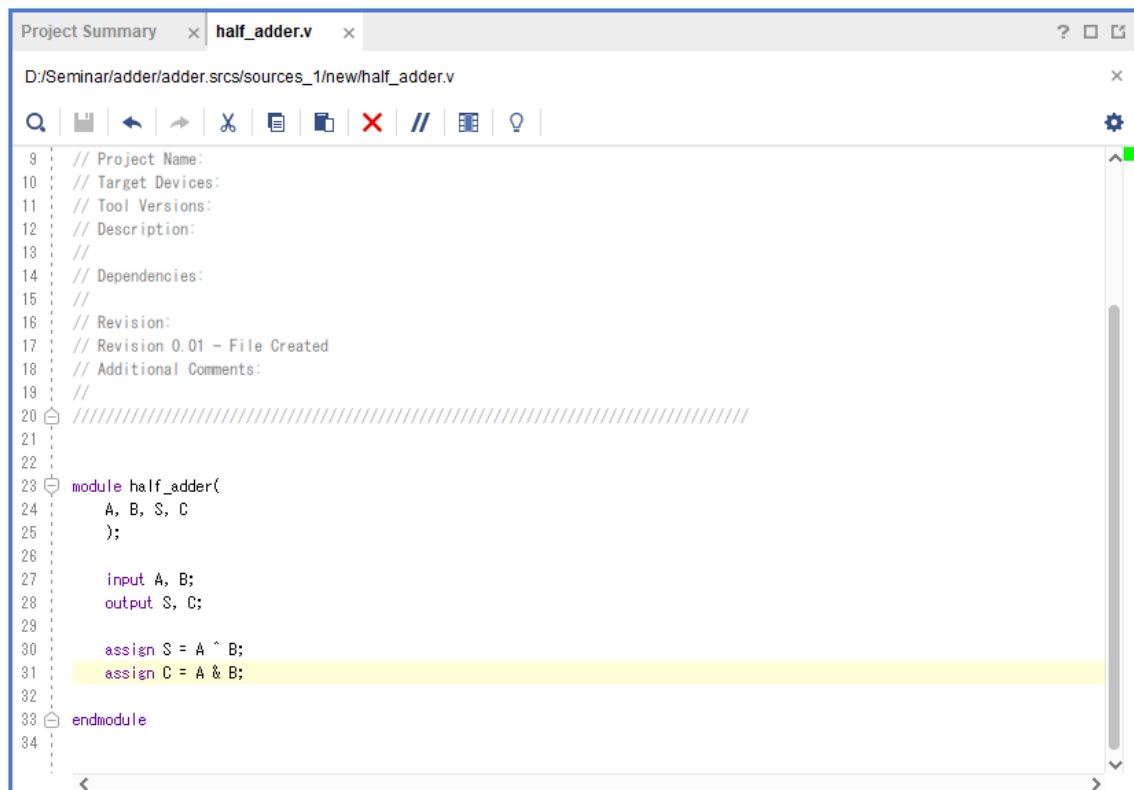
2.8) The created file will be added to the “Design Sources” folder in the “Sources” pane of the “PROJECT MANAGER”.



3. Verilog Coding

3.1) Double click on the **half_adder.v** file in the “Sources” pane to open it.

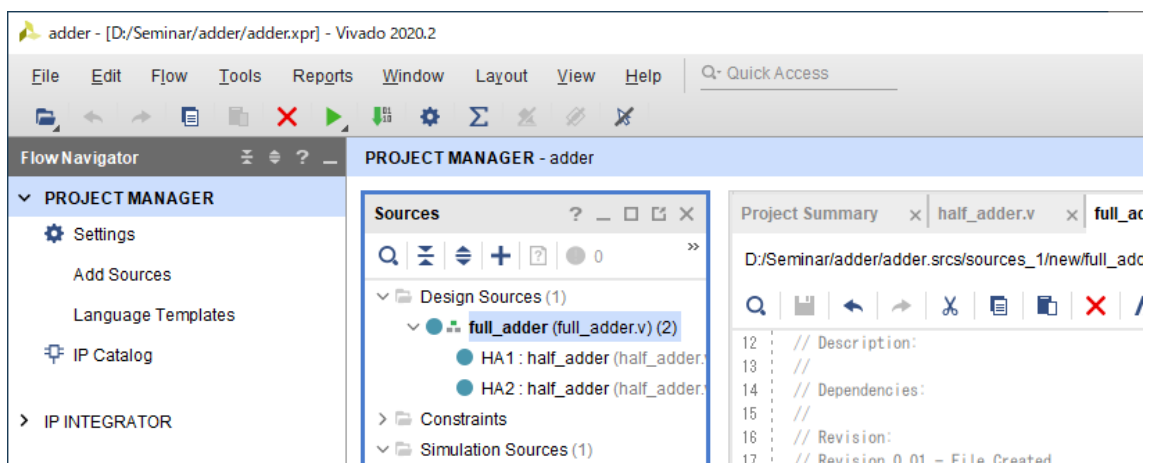
3.2) Write the half adder Verilog code into the text editor.



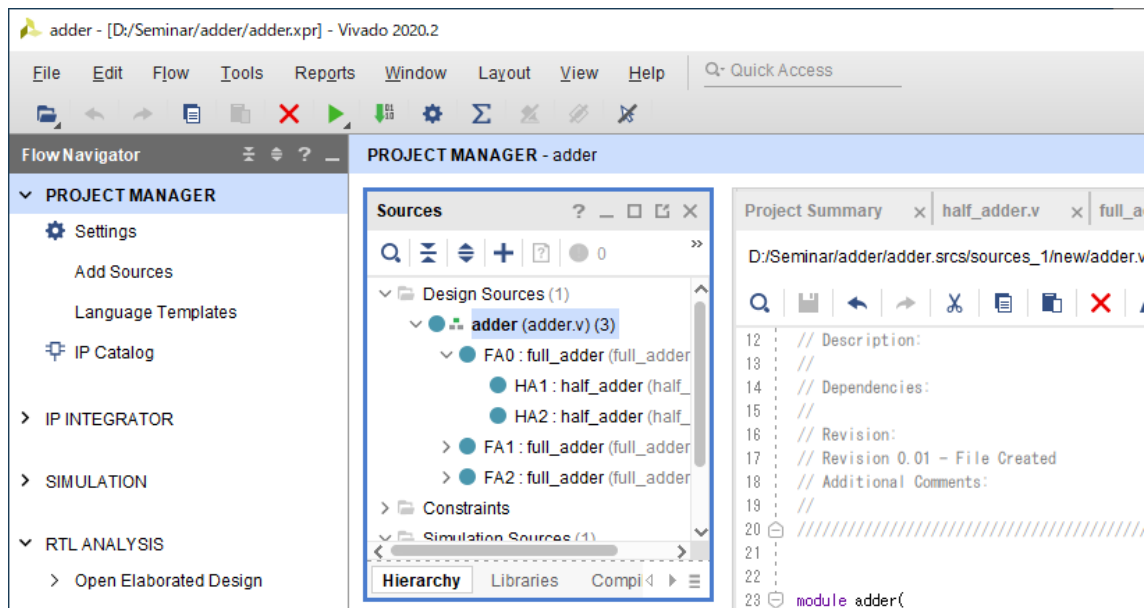
3.3) Ctrl + S to save the file.

3.4) Create and add the **full_adder.v** file into the project and write the full adder Verilog code.

3.5) Vivado automatically updates the hierarchy of the modules and instances and detects the top module of the project. The top module defines the hierarchy of the design for synthesis and implementation.

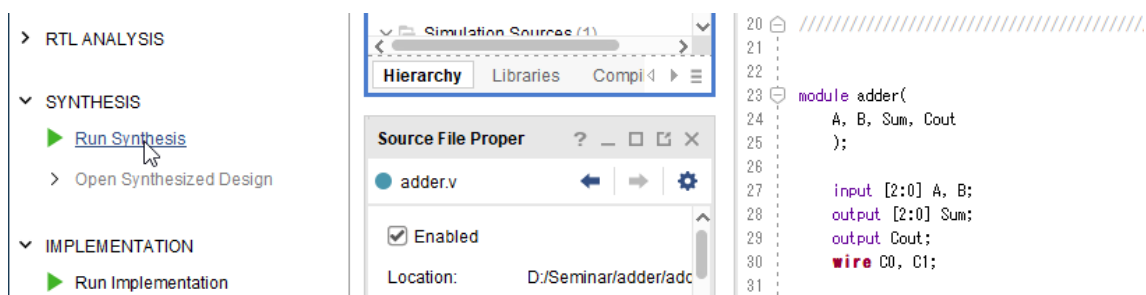


3.6) Create and add the **adder.v** file into the project and write the 3-bit adder Verilog code. It will be the top module of the project.

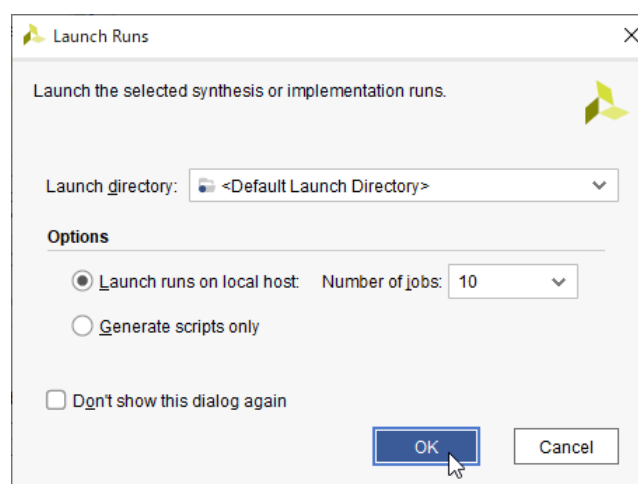


4. Synthesis

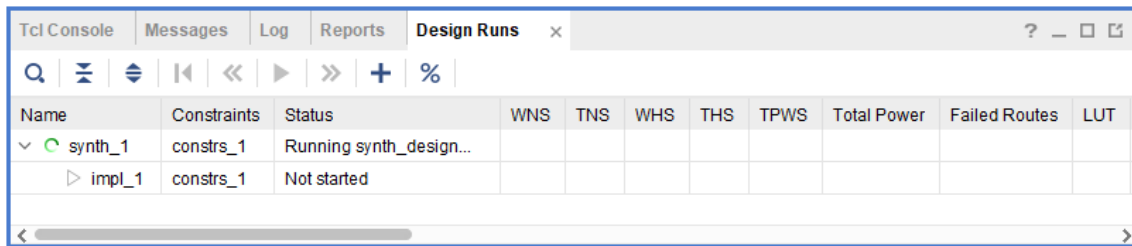
4.1) Click **Run Synthesis** in the “Flow Navigator”.



4.2) You will see the “Launch Runs” form. Click **OK**.



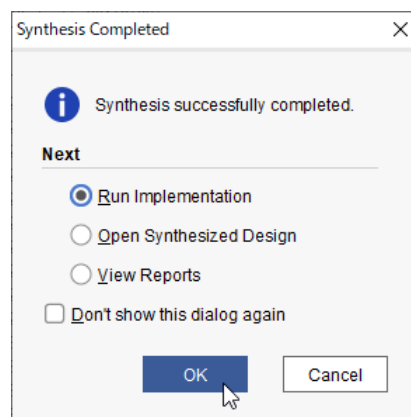
4.3) You can see the run status and information in the “Design Runs” tab.



Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT
synth_1	constrs_1	Running synth_design...								
impl_1	constrs_1	Not started								

5. Implementation

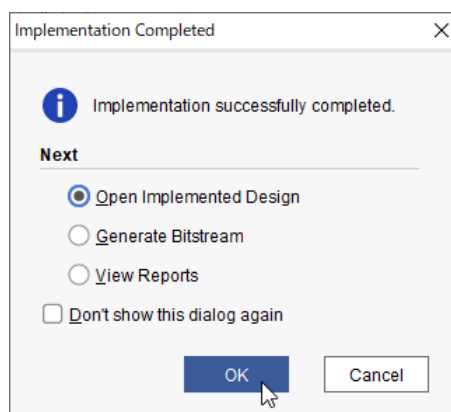
5.1) Upon completion of the synthesis process, you will see the “Synthesis Completed” dialogue. Select **Run Implementation** and click **OK**.



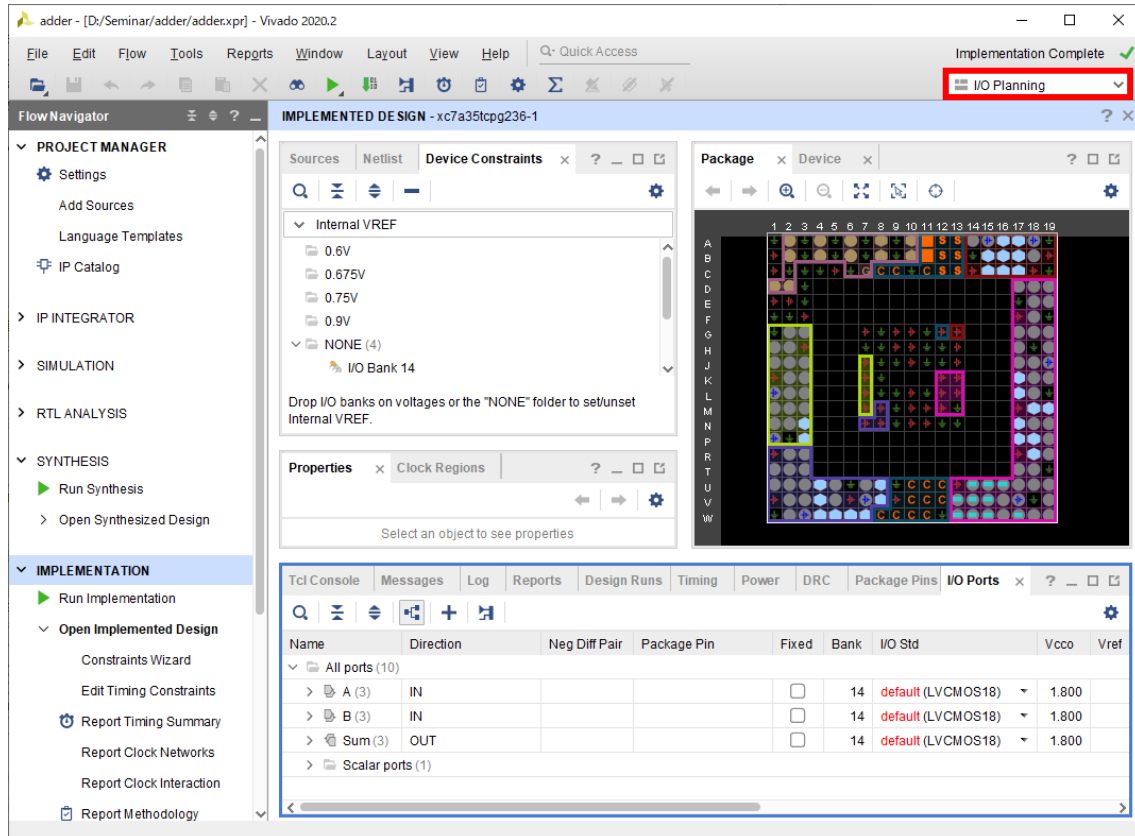
5.2) You will see the “Launch Runs” form. Click **OK**.

6. Pin Assignment

6.1) Upon completion of the implementation process, you will see the “Implementation Completed” dialogue. Select **Open Implemented Design** and click **OK**.

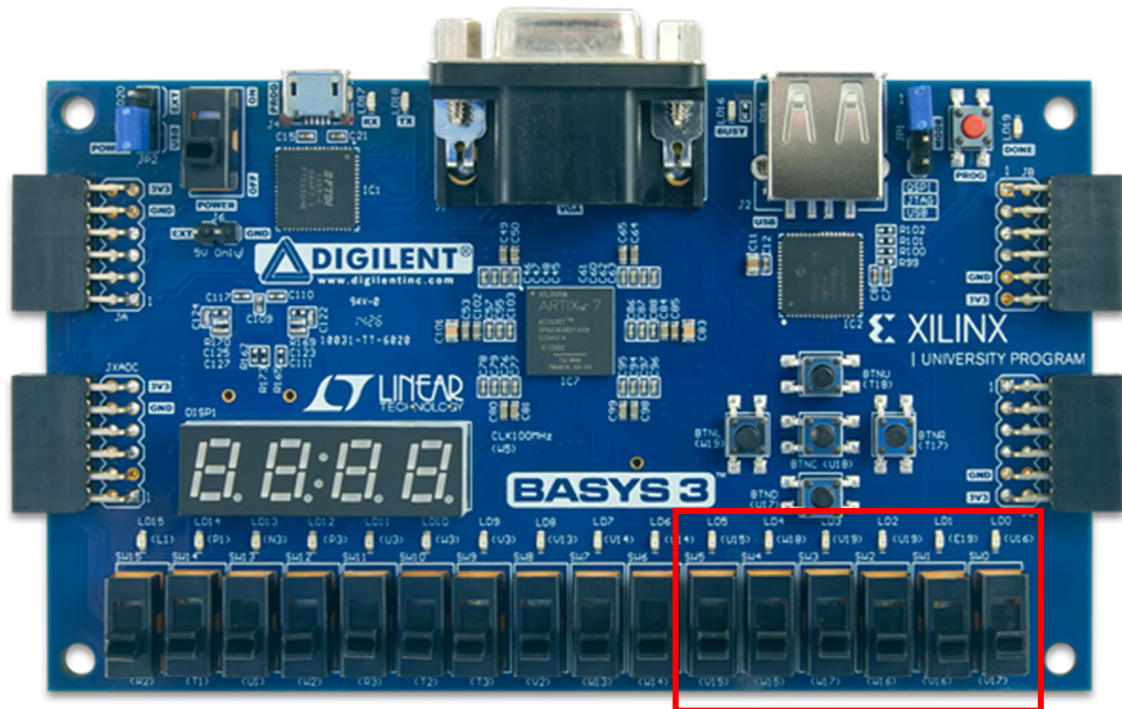


6.2) To launch the I/O Planning view layout, select **I/O planning** in the **Layout** menu.

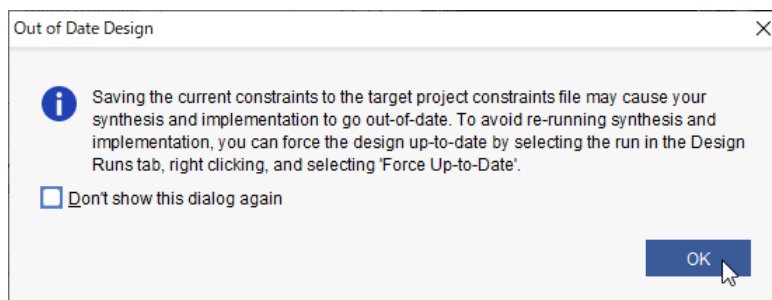


6.3) In the "I/O Ports" tab, set the **Package Pin** and **I/O Std** for the BASYS3 board.

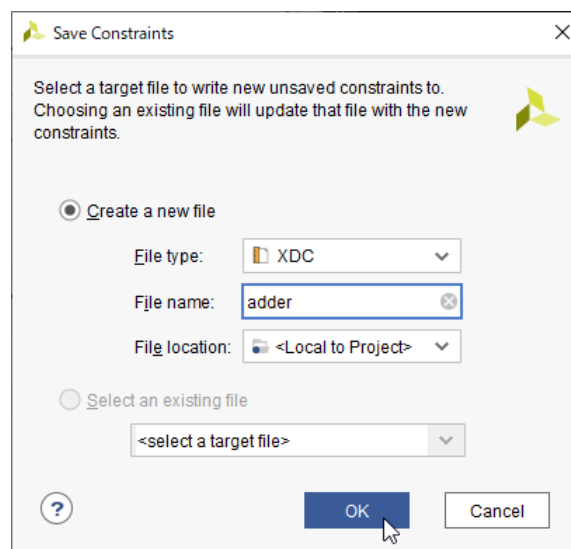
Tcl Console	Messages	Log	Reports	Design Runs	Timing	Power	DRC	Package Pins	I/O Ports	x
<div><div><div><div><div></div><div></div><div></div><div></div></div><div><div></div><div></div><div></div><div></div></div><div><div></div><div></div><div></div><div></div></div><div><div></div><div></div><div></div><div></div></div></div><div><div></div><div></div><div></div><div></div></div><div><div></div><div></div><div></div><div></div></div><div><div></div><div></div><div></div><div></div></div></div></div>										
Name	Direction	Neg Diff Pair	Package Pin	Fixed	Bank	I/O Std	Vcco			
All ports (10)										
A (3)	IN			<input checked="" type="checkbox"/>	14	LVCMOS33*	3.300			
A[2]	IN		W16	<input checked="" type="checkbox"/>	14	LVCMOS33*	3.300			
A[1]	IN		V16	<input checked="" type="checkbox"/>	14	LVCMOS33*	3.300			
A[0]	IN		V17	<input checked="" type="checkbox"/>	14	LVCMOS33*	3.300			
B (3)	IN			<input checked="" type="checkbox"/>	14	LVCMOS33*	3.300			
B[2]	IN		V15	<input checked="" type="checkbox"/>	14	LVCMOS33*	3.300			
B[1]	IN		W15	<input checked="" type="checkbox"/>	14	LVCMOS33*	3.300			
B[0]	IN		W17	<input checked="" type="checkbox"/>	14	LVCMOS33*	3.300			
Sum (3)	OUT			<input checked="" type="checkbox"/>	14	LVCMOS33*	3.300			
Sum[2]	OUT		U19	<input checked="" type="checkbox"/>	14	LVCMOS33*	3.300			
Sum[1]	OUT		E19	<input checked="" type="checkbox"/>	14	LVCMOS33*	3.300			
Sum[0]	OUT		U16	<input checked="" type="checkbox"/>	14	LVCMOS33*	3.300			
Scalar ports (1)										
Cout	OUT		V19	<input checked="" type="checkbox"/>	14	LVCMOS33*	3.300			



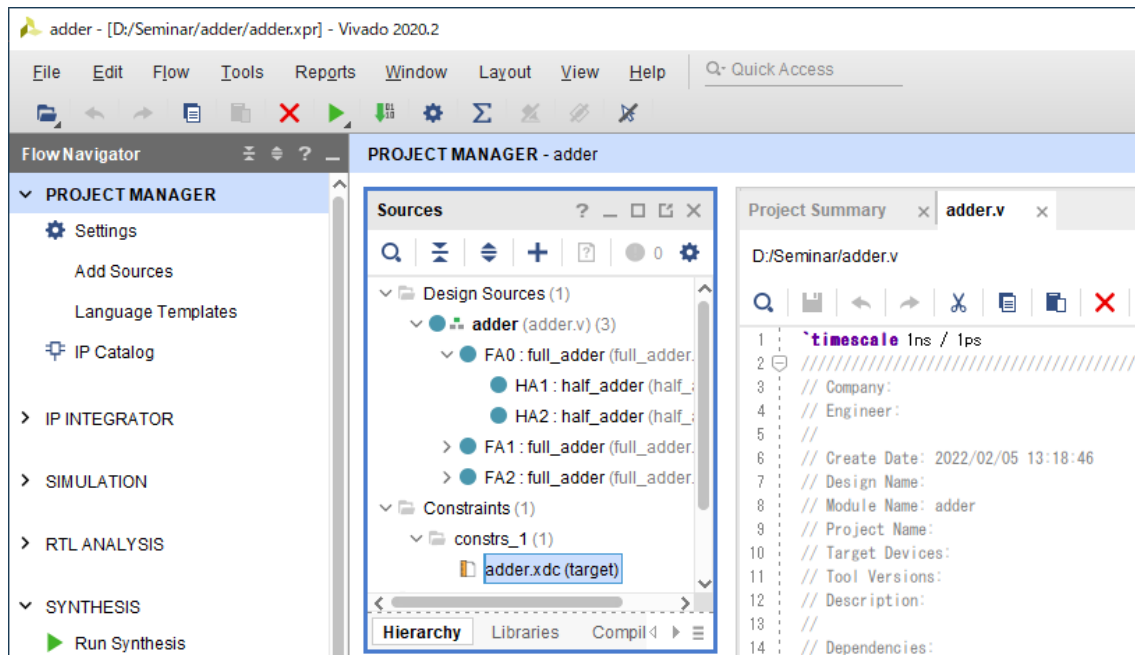
6.3) **Ctrl + S** to save the pin assignment. You will see the “Out of Date Design” dialog. Click **OK**.



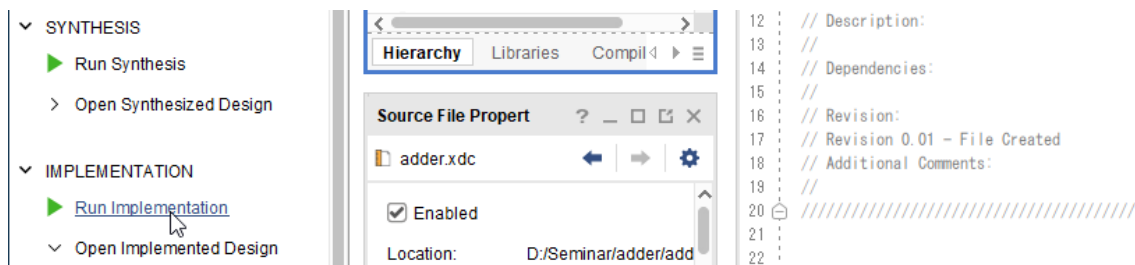
6.4) You will see the “Save Constraints” form. Enter **adder** in the “File name” field and click **OK**.



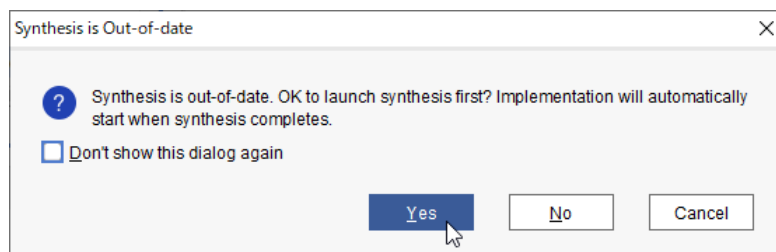
6.5) The constraint file **adder.xdc** will be added to the “Constraints” folder in the “Sources” pane of the “PROJECT MANAGER”.



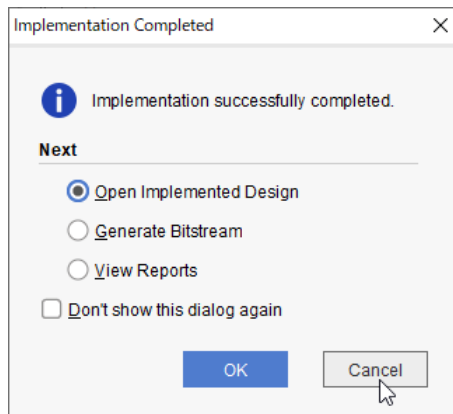
6.6) Click **Run Implementation** in the “Flow Navigator” to rearrange the layout.



6.7) You will see the “Synthesis is Out-of-date” dialogue. Click **Yes**.

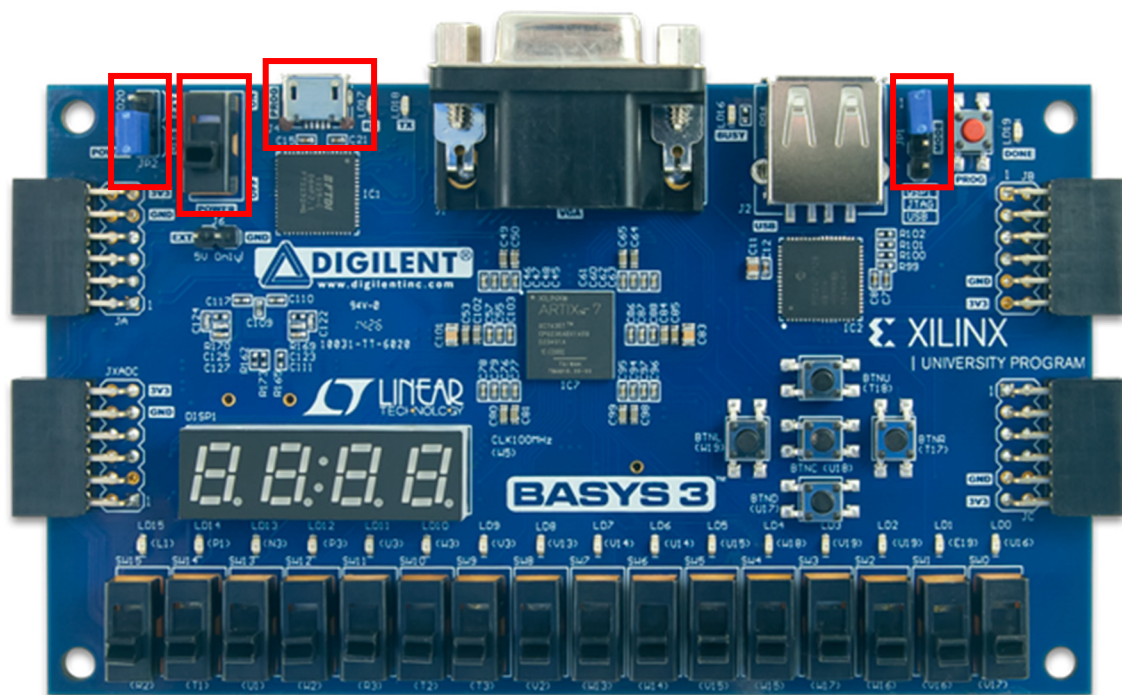


6.8) Upon completion of the implementation process, you will see the “Implementation Completed” dialogue. Click **Cancel** to ignore.



7. Programming FPGA

7.1) Make sure that **JP2** of the BASYS3 board is configured to use **USB** as power source. Furthermore, make sure **JP1** is configured to the **QSPI** mode. Insert a micro USB cable to **J4** and connect it to your PC. Finally, switch **SW16** to the **ON** position. You will see the demo design running.

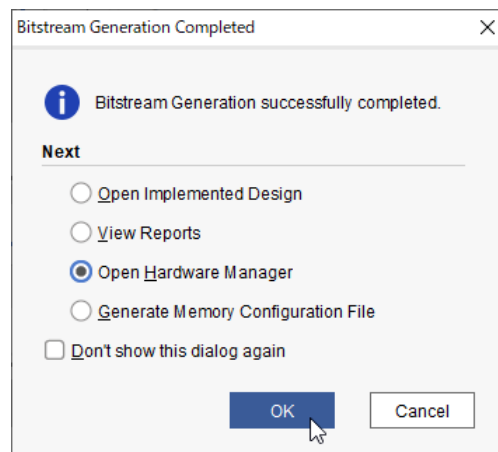


7.1) Click **Generating Bitstream** in the “PROGRAM AND DEBUG” section of the “Flow Navigator”.

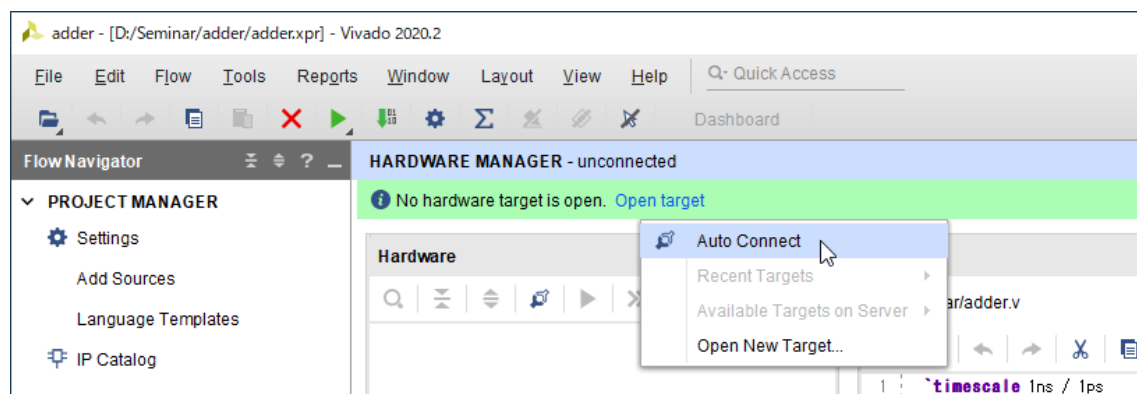


7.2) You will see the “Launch Runs” form. Click **OK**.

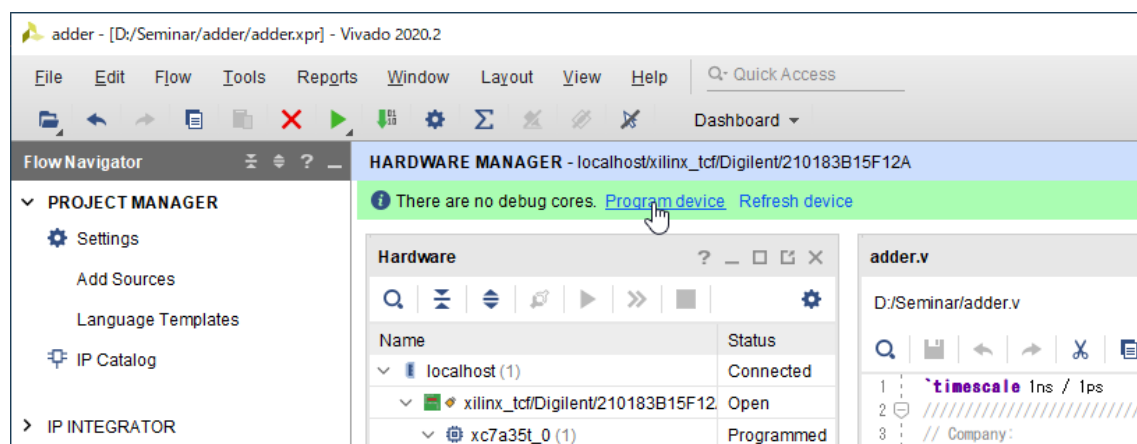
7.3) Upon completion of the bitstream generation, you will see the “Bitstream Generation Completed” dialogue. Select **Open Hardware Manager** and click **OK**.



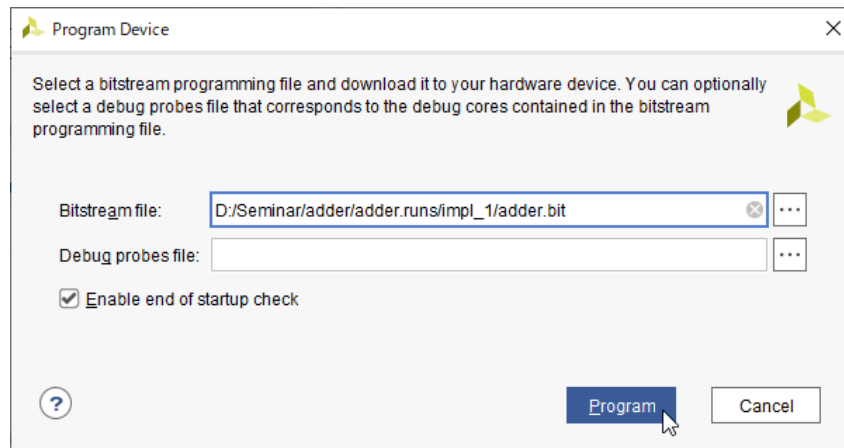
7.4) Click **Open target** in the “HARDWARE MANAGER” and select **Auto Connect**.



7.5) Click **Program device**.



7.6) You will see the “Program Device” form. Click **Program**.



7.7) The BASYS3 is now running your 3-bit adder!

7.8) After testing your design, press **PROG** button to refresh the device.

