Introduction to Xilinx Vivado IP Integrator

Introduction

In this course, you will be using a block design to describe how the processor components connect to each other. For this purpose, you should be familiar with Vivado's IP Integrator feature. The IP Integrator assists the users to understand how to achieve greater design productivity by rapidly creating and reusing subsystem level IP with Vivado and IP Integrator. As a result, tasks that you need to learn are:

   ➢ Creating a block design and adding IPs.
   ➢ Create a VHDL module and add it to the block design.
   ➢ Extract a subset of bus bits using the Slice IP.
   ➢ Concatenate signals using the Concat IP.
   ➢ Set signals to a specific constant using the Constant IP.
   ➢ Create HDL wrapper for your design.
   ➢ Instantiate your wrapper in test-bench and simulate it.

This lab simply begins with a 1-bit full adder (described in VHDL) and uses this block to construct a 2-bit adder using Vivado IP Integrator. The numbers to be added are A and B (each 2-bit unsigned integers) and the result (sum) is a three-bit unsigned integer.

Lab Objective

The objective of this lab is to create a 2-bit adder from a 1-bit full adder. This process should familiarize students with the IP Integrator tool to be used in next lab projects.

Design Entry and Simulation Steps

1- In Vivado, create a new project (File → New Project ...). Select a location and name the project *lab1*. Click Next in the Add Sources and Add Constrains pages. These can be added later.

2- Add a new VHDL source to your project. In the Sources window, Click the Add Sources button → Add or create design source → Create File. Name the file *FA → Finish*.

Replace the content of the file you created with the full adder code from Beachboard. The file name will be FA.vhd. Make sure to save the file.

3- Now create a block design using the IP Integrator tool as shown below. Name the block design a*dder_2bit*.
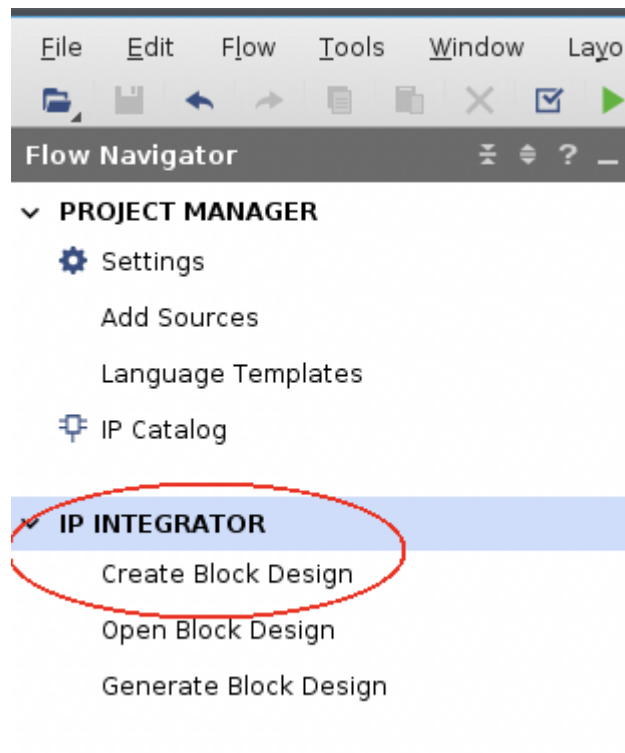
4- Add two instances of the full adder to the block design. In the Sources window, right-click on the source file FA.vhd and click "Add Module to Block Design".
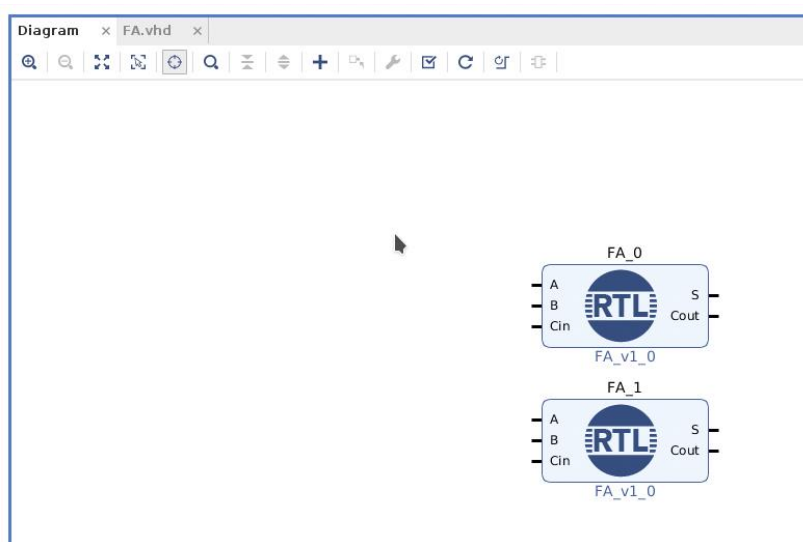
5- Now add ports for input and output. These will serve as ports for the block design. Right-click on the block design background and click "Create Port…". In this lab, we need two input ports.
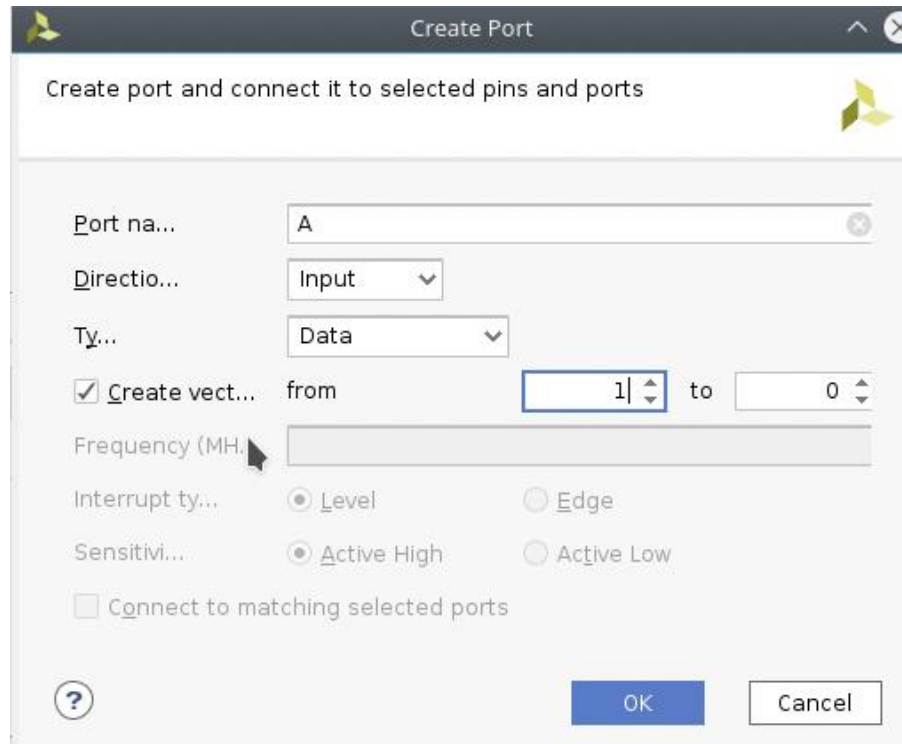
A: a 2-bit data input

B: a 2-bit data input

We need one output port:

Sum: a 3-bit data output.

Here is an example for setting up port A.

Follow this example to create all ports needed.

6- Add 4 slice IP's (in the Diagram window, click Add IP). The Slice IP can be used to extract a subset of bits from a bus. For example, we need to get bit0 and bit1 separately from port A which is a two-bit input.
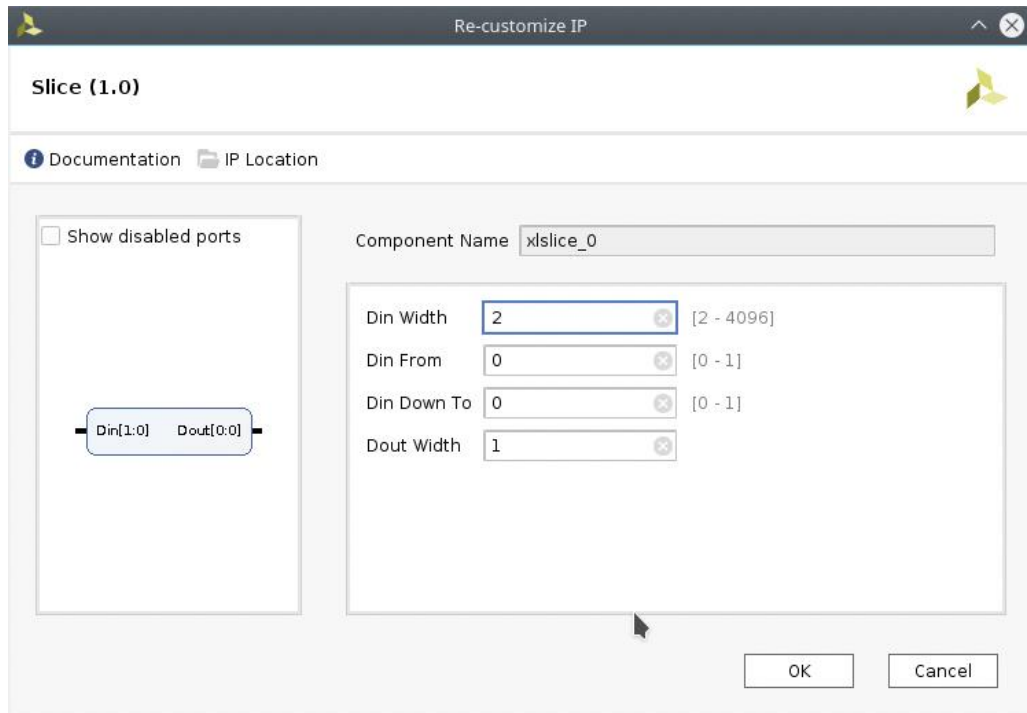
To customize the IP, double-click on it. Now, set the following parameters:

Din Width==> The size of the bus to be split. In this case the bus is 2 bits wide.

Din from ==> Subset starts from this value.

Din Down To => Subset ends at this value.

The following figures show how to extract bit 0 from the 2-bit input.

Follow the previous example to extract individual bits from input A and B. Use one slice IP for each bit. The bits will be connected to the adder as follows.

A(0) →port A in FA_0

A(1) → port A in FA_1

B(0) →port B in FA_0

B(1) → port B in FA_1

7- Use a Constant IP to provide the Cin for FA_0. The value of the constant should be zero. To customize the IP, double-click on it. Then set the *Const Width* and *Const Value*. In this case, they will be 1 and 0 respectively.

8- Use a Concat IP to concatenate the S output from the two full-adder along with the Cout from FA_1. The Concat IP, combines signals into a wider signal. Customize the IP to increase its number of ports to 3. Also connect the cout output of FA_0 to Cin in FA_1. Your 2-bit adder should look as follows:
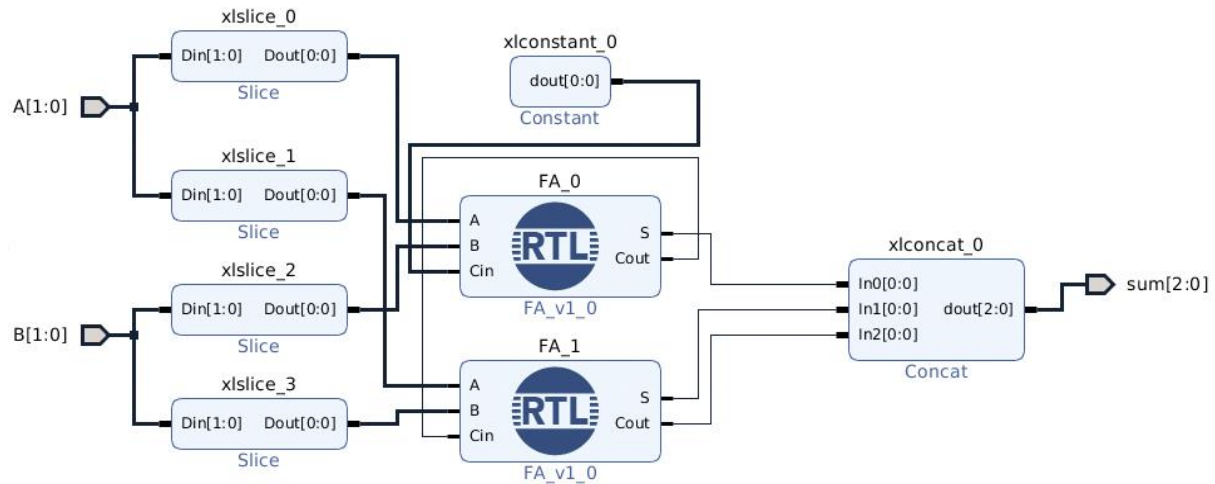
9- Now we are ready to create a wrapper for the block design. Right-click on the block design in the Sources window (the yellow icon) and select "Generate HDL Wrapper" and also click on "Generate Output Products" accept the default options. Notice that an HDL wrapper is created and listed in the Sources window.

10- Download the test-bench from BeachBoard (adder_tb.vhd). Create a test-bench source and replace its content with the code in the file you downloaded. Make sure that you understand how the wrapper is instantiated in the test-bench.

11- Now run the behavioral simulation by clicking on "Run Simulation" in the Flow Navigator section and observe the waveform result.
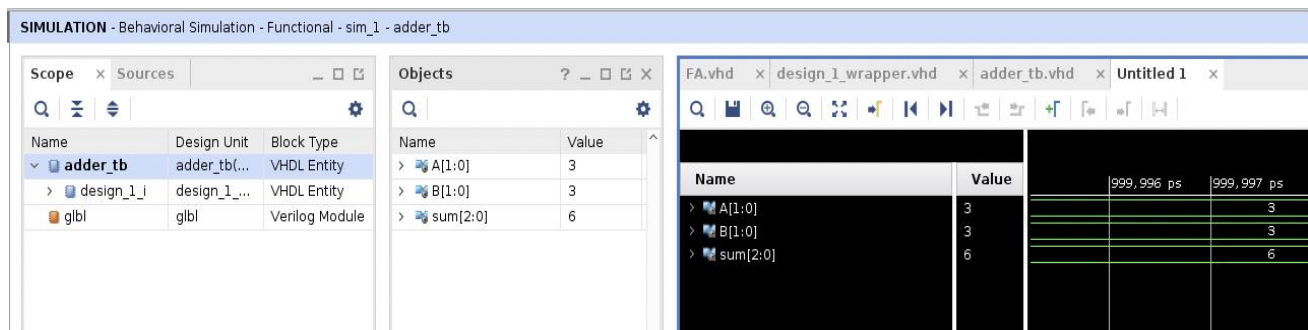
Lab Deliverables

Make sure to complete all the tasks above to get familiar with the Vivado tool and IP Integrator. You need to submit a brief report explaining what you learned from this lab and include the snapshot of your design and captured waveform. One submission per group is enough. **The due date for submitting the required files is July 15th (07/15/2021), 11:59 PM.**