Adding Custom IP to the System

Introduction

This lab guides you through the process of adding a custom peripheral to a processor system. You will add an AXI4Lite interface peripheral.

Objectives

After completing this lab, you will be able to:

- Add a custom peripheral to your design
- Add pin location constraints
- Add block memory to the system

Procedure

This lab is separated into steps that consist of general overview statements that provide information on the detailed instructions that follow. Follow these detailed instructions to progress through the lab.

This lab comprises two primary steps: You will add a customized LED_IP and connect the IP in the design, and add the Block RAM (BRAM) Memory.

Design Description

You will extend the Lab 2 hardware design by adding an AXI peripheral (refer to LED_IP in **Figure 1**) to the system, and connecting it to the LEDs on the Zynq board you are using. You will connect the peripheral to the system and add pin location constraints to connect the LED display controller peripheral to the on-board LED display. Finally, you will add BRAM Controller and BRAM before generating the bitstream.

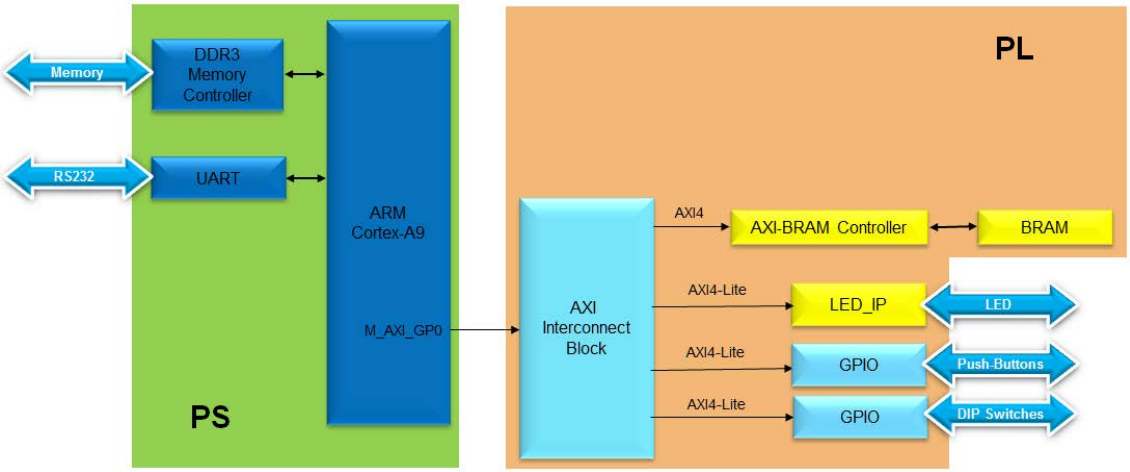

Figure 1. Design Updated from Previous Lab

General Flow for this Lab

In the instructions below;
{***sources***} refers to: EmbeddedSystem_labs\lab3
{***labs***} refers to : EmbeddedSystem_labs\lab3
{***labsolutions***} or for the Zybo refers to: EmbeddedSystem_labs\lab2

Modify the Project Settings Step 1

1-1. Open the lab2 project from the EmbeddedSystems_labs directory, and save the project as lab3. Set Project Settings to point to the created IP repository.

1-1-1. Unzip the customized LED_IP from the file **led_ip_1.0.zip** to **{labs}\led_ip_1.0**

1-1-2. Start **Vivado 2017.2** and open the saved **{labs}\lab2** project (Accept to update to current version of Vivado – Report IP status and "**Upgrade Selected**" IP cores and finally "**Generate Output Products**")

1-1-3. Select **File > Save Project As …** to open the *Save Project As* dialog box. Enter **lab3** as the project name. Make sure that the *Create Project Subdirectory* option is checked, the project directory path is {labs}\ and click **OK**.

This will create the lab3 directory and save the project and associated directory with lab3 name.

1-1-4. Click **Project Settings** in the *Flow Navigator* pane.

1-1-5. Select **IP->Repository** in the left pane of the *Project Settings* form.

1-1-6. Click on the **+** button, browse to **{labs}\led_ip_1.0** and click **Select**.

The led_ip_v1_0 IP will appear the **IP in the Selected Repository** window.

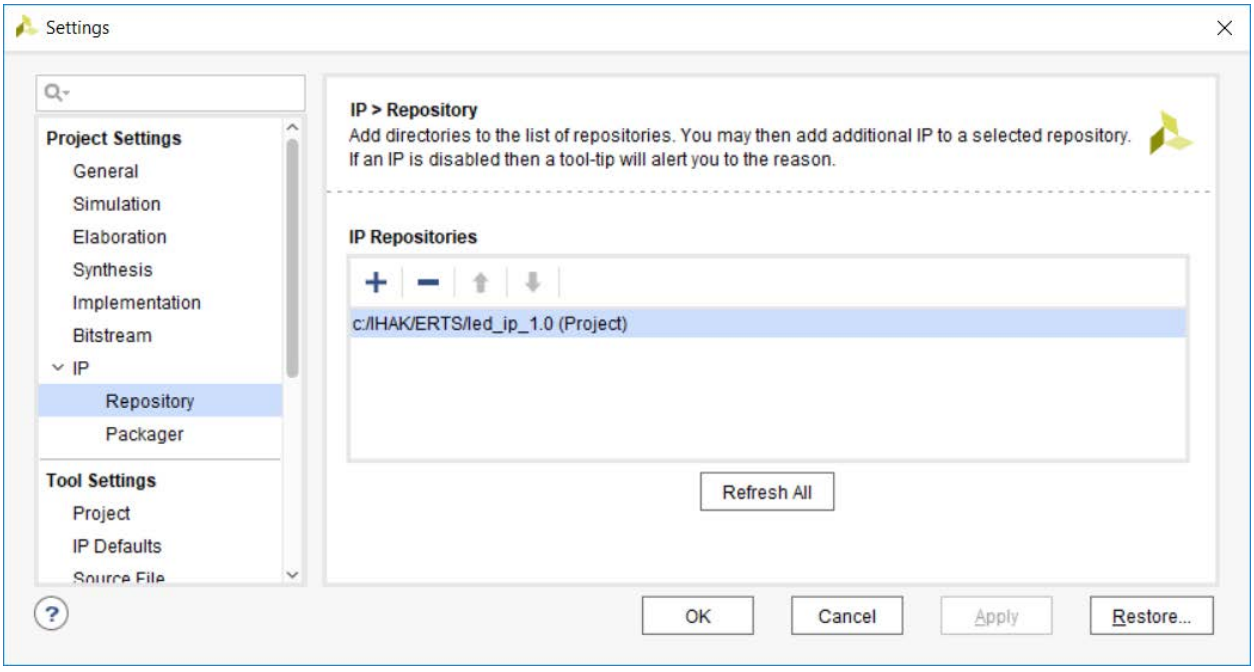

Figure 13. Specify IP Repository

1-1-7. Click **OK**.

Add the Custom IP, BRAM, and the Constraints Step 2

2-1. Add led_ip to the design and connect to the AXI4Lite interconnect in the IPI. Make internal and external port connections. Establish the LED port as external FPGA pins.

2-1-1. Click **Open Block Design** under **IP Integrator** in the Flow Navigator pane

2-1-2. Click the Add IP icon and search for **led_ip_v1_0** in the catalog by typing “led” in the search field.

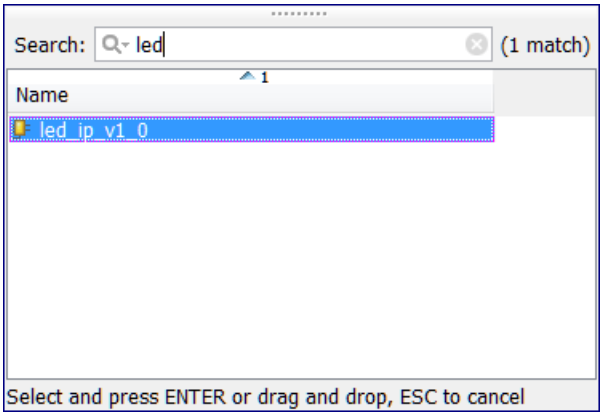

Figure 12. Searching for led_ip in the IP Catalog

2-1-3. Double-click **led_ip_v1_0** to add the core to the design.

2-1-4. Select the IP in the block diagram and change the instance name to **led_ip** in the properties view.

2-1-5. Double click the block to open the configuration properties

2-1-6. For the ZedBoard, leave the *Led Width* set to 8, or **for the Zybo, set the width to 4**

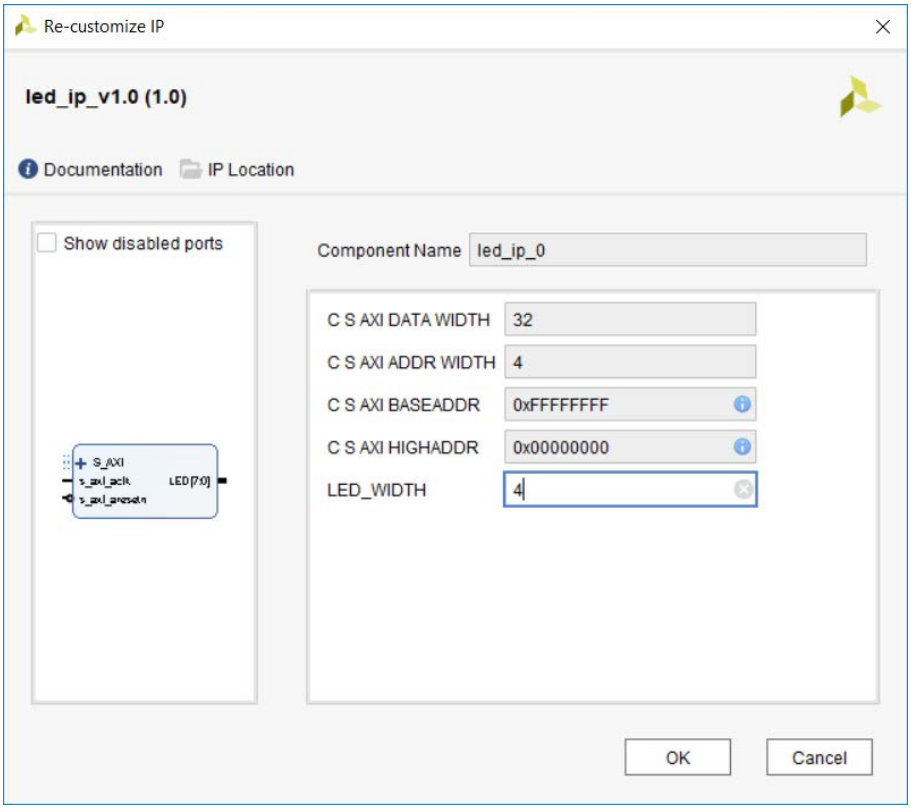

Figure 13. Configure the LED IP LED_WIDTH

2-1-7. Click on **Run Connection Automation**, select **/led_ip/S_AXI** and click **OK** to automatically make the connection from the AXI Interconnect to the IP.

Click the regenerate button ( ) to redraw the diagram.

2-1-8. Select the *LED* port on the led_ip instance (by clicking on its pin), right-click and select **Make External**.

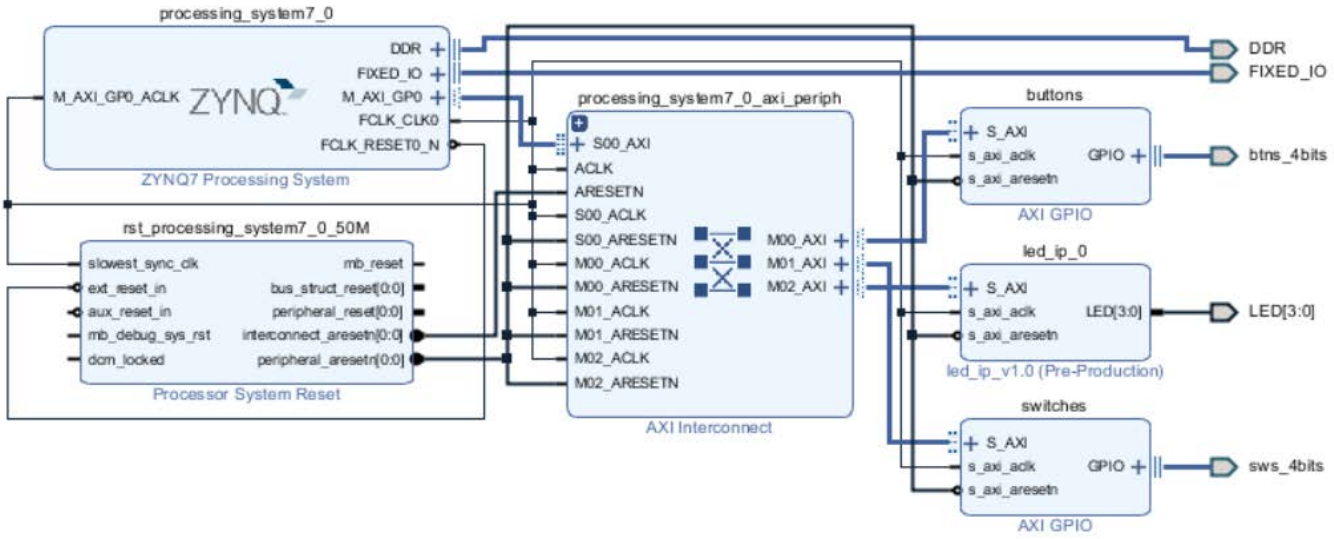

Figure 14. LED external port added and connected

2-1-9. Select the **Address Editor** tab and verify that an address has been assigned to *led_ip*.

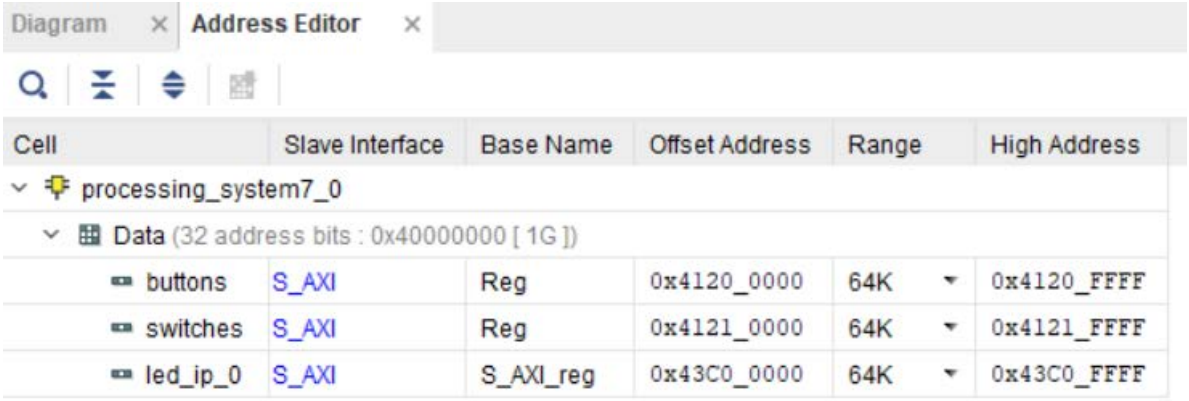

Figure 15. Address assigned for led_ip

2-1-10. Change the **AXI GPIO leds** to control **buttons** instead of leds as shown in figure 14.

2-1-11. Remove the external MIO btn pin from the ZYNQ processing system as illustrated in figure 14.

2-2. Add BRAM to the design

2-2-1. In the Block Diagram, click the Add IP icon and search for BRAM and add one instance of the **AXI BRAM Controller**

2-2-2. Run *Connection Automation* on **axi_bram_ctrl_0/S_AXI** and click **OK** when prompted to connect it to the **M_AXI_GP0 Master.**

2-2-3. Double click on the block to customize it and change the number of BRAM interfaces to 1 and click **OK**.

Notice that the AXI Protocol being used is AXI4 instead of AXI4Lite since BRAM can provide higher bandwidth and the controller can support burst transactions.

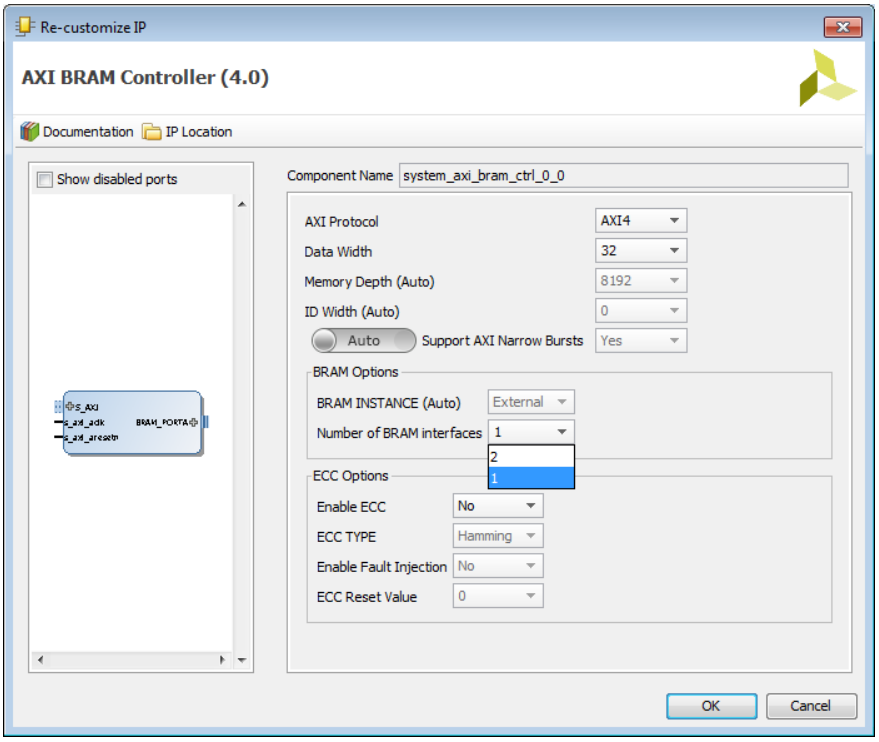

Figure 16. Customize BRAM controller

2-2-4. Click on *Run Connection Automation* to add and connect a **Block Memory Generator by selecting axi_bram_ctrl_0/BRAM_PORTA and click OK** (This could be added manually)

2-2-5. Validate the design to ensure there are no errors (F6), and click the regenerate button ( ) to redraw the diagram.

The design should look similar to the figure below.

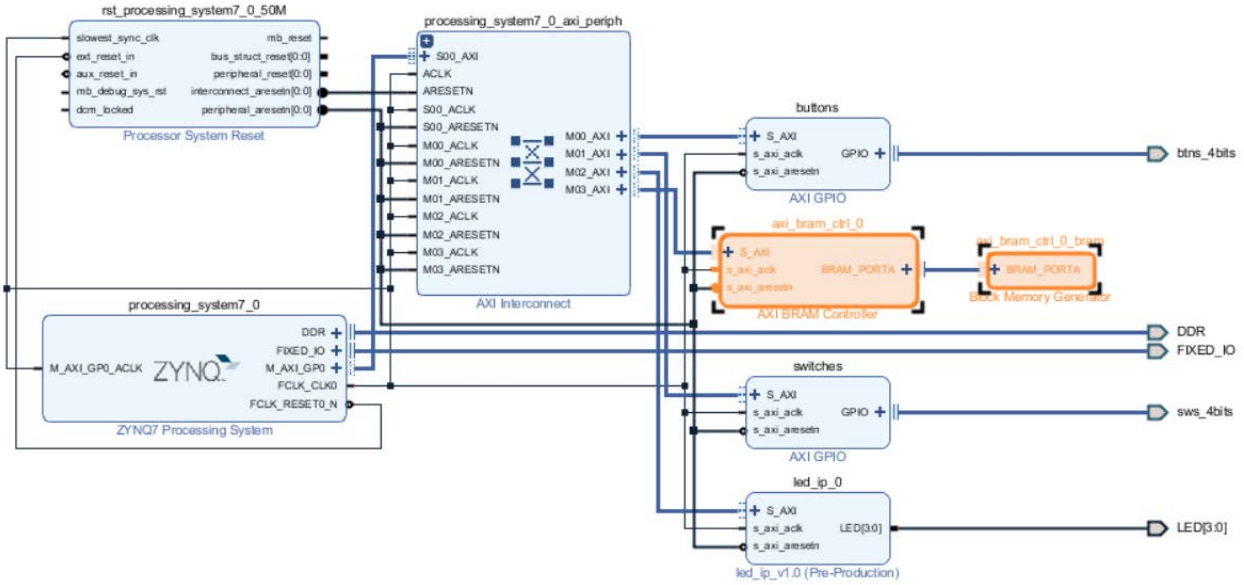

Figure 17. Completed Block Diagram

2-2-6. In the Address editor, increase the Range of the **axi_bram_ctrl_0** to **8K**

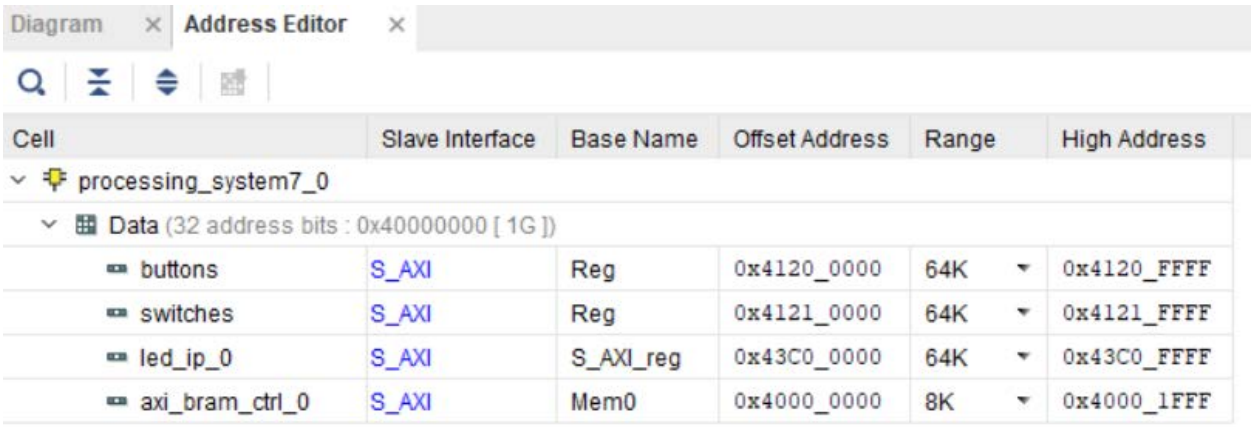

Figure 20. Adjusting memory size

2-2-7. Press **F6** to validate the design one last time.

2-3. Update the top-level wrapper and add the provided lab3_*.xdc constraints file.

2-3-1. Click **Add Sources** in the *Flow Navigator* pane, select **Add or Create Constraints**, and click **Next**.

2-3-2. Click the **Add Files** button, browse to the {**sources}\lab3** folder, select **lab3_zed.xdc** for the ZedBoard, or **lab3_Zybo.xdc** for the Zybo

2-3-3. Click **Finish** to add the file.

2-3-4. Expand Constraints folder in the *Sources* pane, and double click the **lab3_*.xdc** file entry to see its content. This file contains the pin locations and IO standards for the LEDs on the Zynq board. This information can usually be found in the manufacturer's datasheet for the board.

2-3-5. Right click on *system.bd* and select *Generate output products*

2-3-6. Click on **Generate Bitstream** and click **Yes** if prompted to save the Block Diagram, and click **Yes** again if prompted to launch synthesis and implementation. Click **Cancel** when prompted to *Open the Implemented Design*

Conclusion

Vivado was used to import a custom IP block into the IP library. The IP block was added to the system. Connection automation was run where available to speed up the design of the system by allowing Vivado to automatically make connections between IP. An additional BRAM was added to the design. Finally, pin location constraints were added to the design.