Introduction to Zynq Hardware
Lab 5
Adding a PL Peripheral

SPEEDWAY

June 2017
Version 11

AVNET
Reach Further™

Lab 5 Overview

In this lab, we will expand our block design by extending our memory space with a PL-based Block RAM (BRAM). The BRAM can be used to buffer data going between the PS and PL.

Lab 5 Objectives

When you have completed Lab 5, you will know how to do the following:

- Add a BRAM from the IP Catalog
- Connect AXI peripherals to the Zynq PS.

Experiment 1: Add the PL BRAM from the IP Catalog

This experiment shows how to add the PL BRAM IP from the IP Catalog.

Experiment 1 General Instruction:

Open Vivado Project and Block Design. Add Block Memory to the block design.

Experiment 1 Step-by-Step Instructions:

1. <Optional> If you did not complete Lab 3 or wish to start with a clean copy, delete the `ZynqDesign and SDK_Workspace` folders in the `ZynqHW/2017_1` folder. Then unzip **`Solutions\ZynqHW_Lab3_Solution.zip`** to the `2017_1` folder. If you have 7-zip installed, you can do this by right-clicking and dragging **`ZynqHW_Lab3_Solution.zip`** to the `2017_1` folder. Select **7-Zip → Extract Here**.

   Perhaps you are wondering why we are starting with the solution for Lab 3 rather than the solution for Lab 4. Well the reason is that Lab 4 activities did not result in a modification to the hardware platform from Lab 3 but acted rather as a demonstration that TCL can be used to manipulate a project directly.

2. Launch Vivado.

3. **Open** the project: **C:\Speedway\ZynqHW\2017_1\ZynqDesign\ZynqDesign.xpr**

4. **Open** the Block Design, *Z_system.bd*

5. With the block design open, click anywhere in the white space of the diagram and right-click then select **Add IP…** Alternatively select the Add IP button on the vertical shortcut bar.

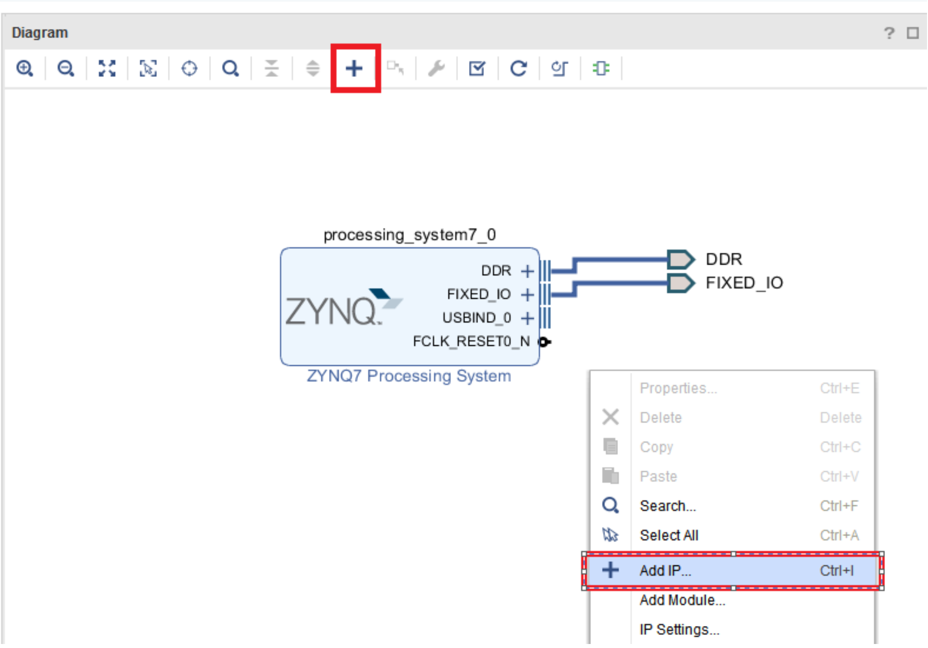
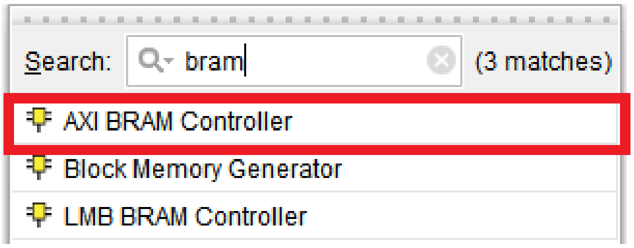


Figure 1 - Add IP to Block Design

6. The IP catalog will open. Add a Block RAM Controller by entering **bram** in the search field. Double-click **AXI BRAM Controller** to add it to the system.



Search: bram (3 matches)

Figure 2 - Add AXI BRAM Controller

7. Double-click the **AXI BRAM Controller**. It may be helpful to hit the Regenerate Layout Button on the vertical shortcut bar, ↻, to see this view.

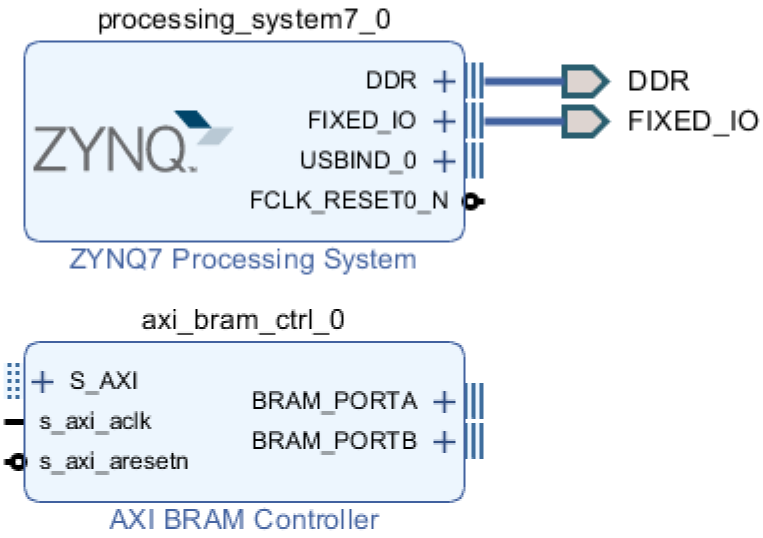
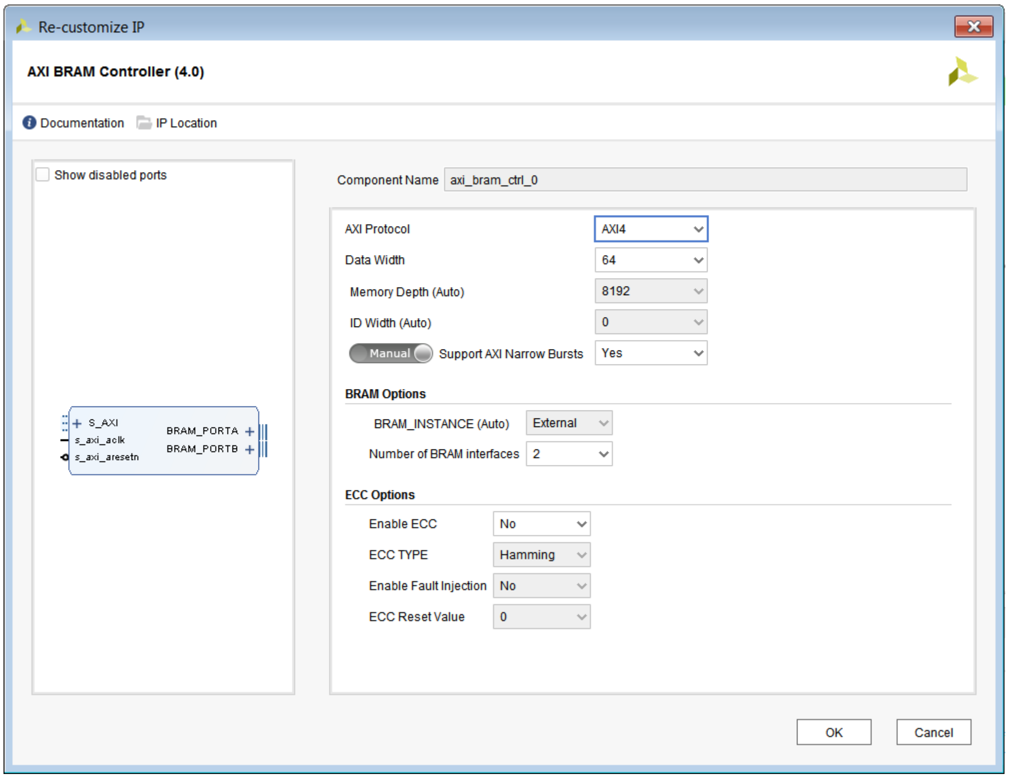


Figure 3 - Customize IP with Double-Click

8. Increase the *Data Width* to **64** bits.

9. Change the *Support for AXI Narrow Bursts* to **Manual** and set to **Yes**. Click **OK**.



Re-customize IP

AXI BRAM Controller (4.0)

Documentation IP Location

□ Show disabled ports

Component Name: axi_bram_ctrl_0

AXI Protocol: AXI4

Data Width: 64

Memory Depth (Auto): 8192

ID Width (Auto): 0

Manual Support AXI Narrow Bursts: Yes

BRAM Options

BRAM_INSTANCE (Auto): External

Number of BRAM interfaces: 2

ECC Options

Enable ECC: No

ECC TYPE: Hamming

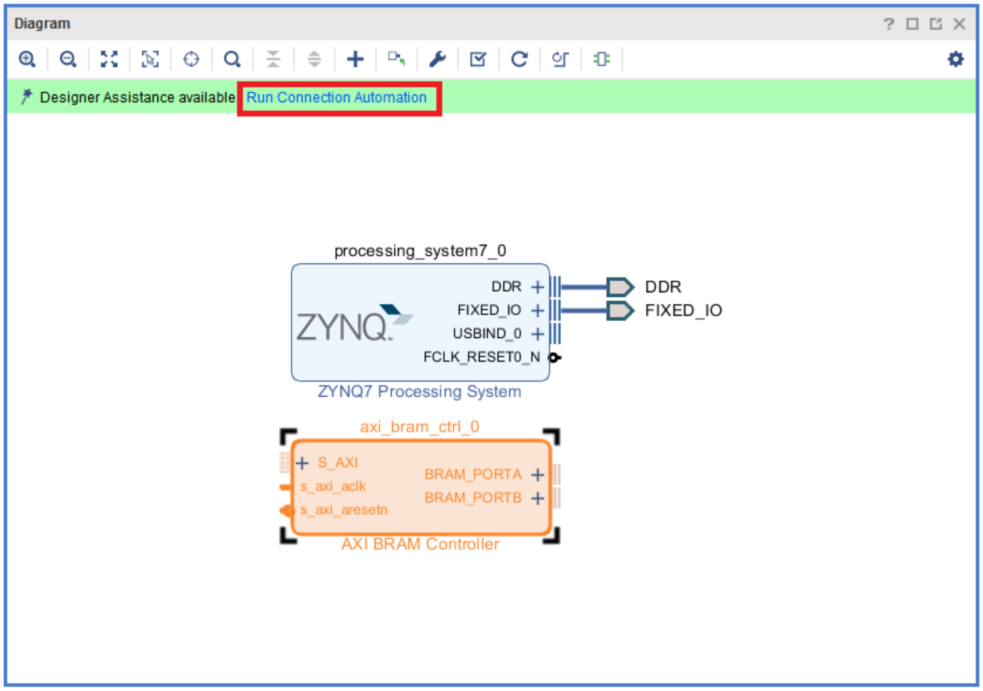
Enable Fault Injection: No

ECC Reset Value: 0

OK Cancel

Figure 4 - BRAM Controller Options

10. See the green bar at the top of the screen that advises you that *Designer Assistance available*. Click on **Run Connection Automation**.

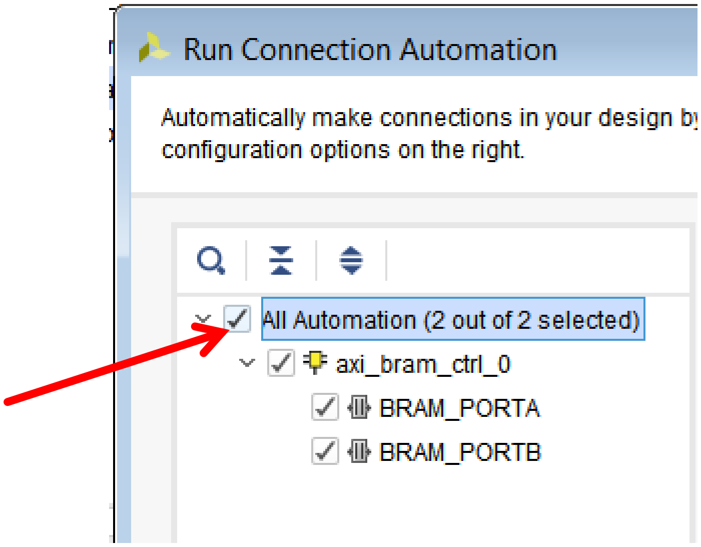


Diagram

Designer Assistance available Run Connection Automation

Figure 5 – Designer Assistance Available

11. Check the box for **All Automation**. Then click **OK**.



Run Connection Automation

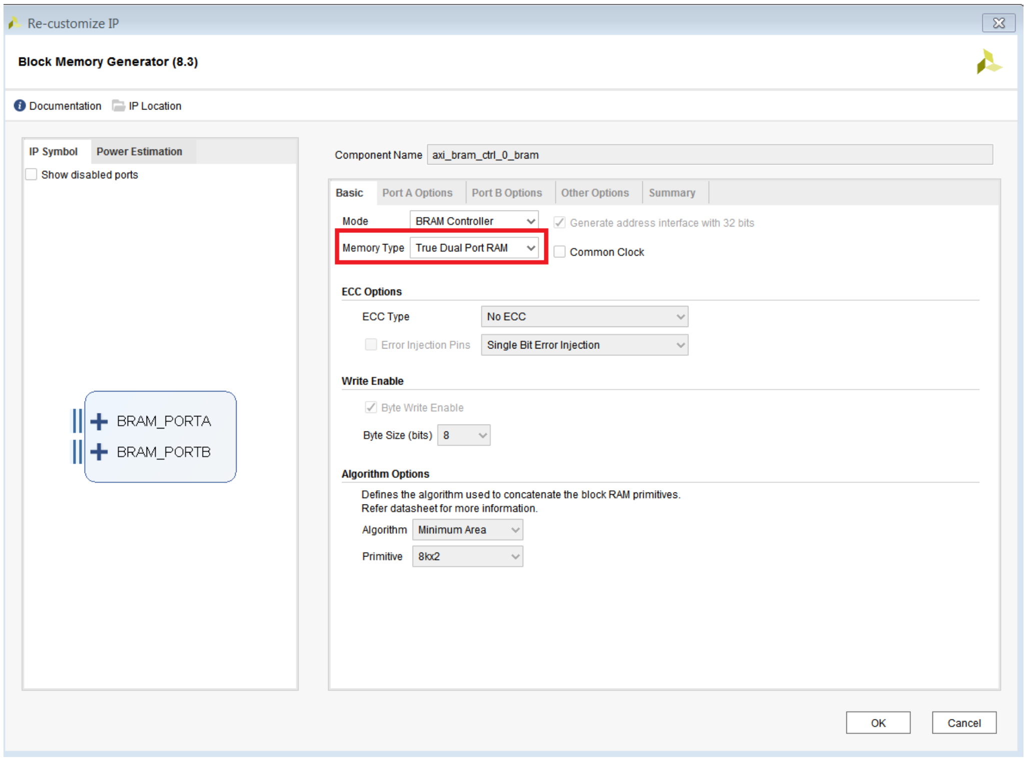
Automatically make connections in your design b
configuration options on the right.

☑ All Automation (2 out of 2 selected)
☑ axi_bram_ctrl_0
☑ BRAM_PORTA
☑ BRAM_PORTB

Figure 6 – Designer Assistance Check

12. See that the Assistant has automatically added and connected a *Block Memory Generator* IP block. Double-click the **axi_bram_ctrl_0_bram** to see how the tool set up the IP.

13. Notice that **True Dual Port RAM** from the *Memory Type* drop-down list has been set. Click **OK**.



Re-customize IP

Block Memory Generator (8.3)

Documentation IP Location

□ Show disabled ports

Component Name: axi_bram_ctrl_0_bram

Mode: BRAM Controller
☑ Generate address interface with 32 bits
Memory Type: True Dual Port RAM
□ Common Clock

ECC Options
ECC Type: No ECC
□ Error Injection Pins: Single Bit Error Injection

Write Enable
☑ Byte Write Enable
Byte Size (bits): 8

Algorithm Options
Defines the algorithm used to concatenate the block RAM primitives.
Refer datasheet for more information.
Algorithm: Minimum Area
Primitive: 8kx2

Figure 7 - Block Memory Generator Set to True Dual Port

Experiment 2: Connect AXI Interconnect and Build

This experiment shows how to add an AXI Interconnect Block from the IP Catalog.

Experiment 2 General Instruction:

Enable a Master AXI Interconnect on the PS. Enable a fabric clock from the PS. Connect all IP to the PS.

Experiment 2 Step-by-Step Instructions:

If you examine the BRAM Controller closely, you will notice an **S_AXI** port in the upper left hand corner. The BRAM Controller expects to be a Slave on an AXI Interconnect. However, in order to do that, we will first need a Master. Since we now have logic in the PL, we will also need a clock accessible to that logic.

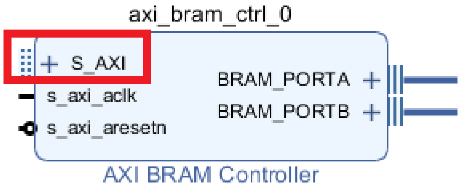


Figure 8 – Slave AXI Port on AXI BRAM Controller

1. **Double-click** the Zynq PS (processing_system7_0) block.

2. At the bottom of the screen click on green box labeled, **32b GP AXI Master Ports**.

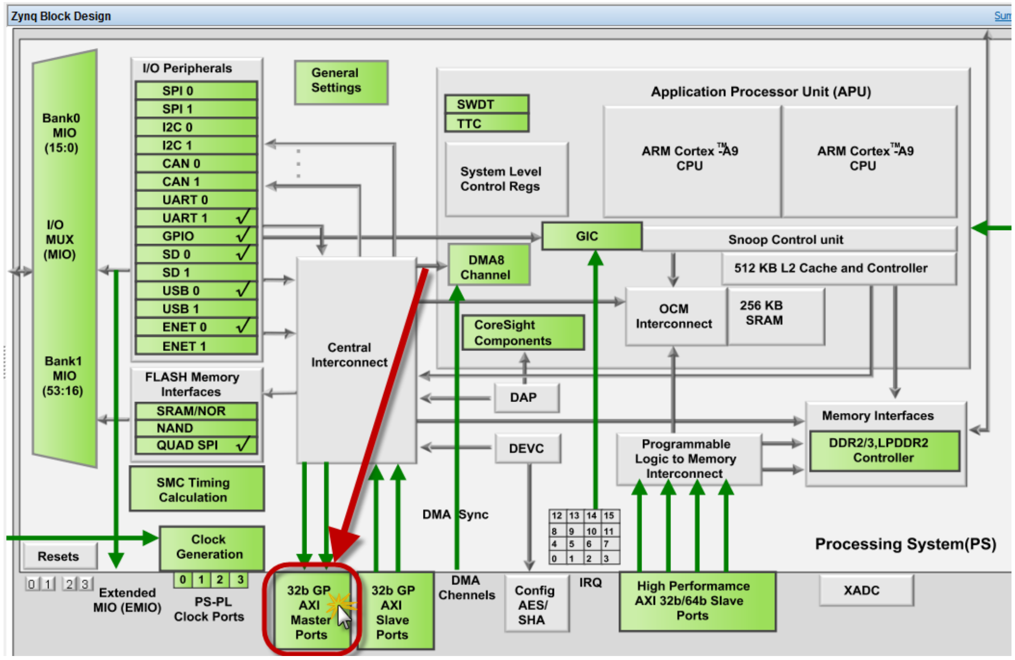
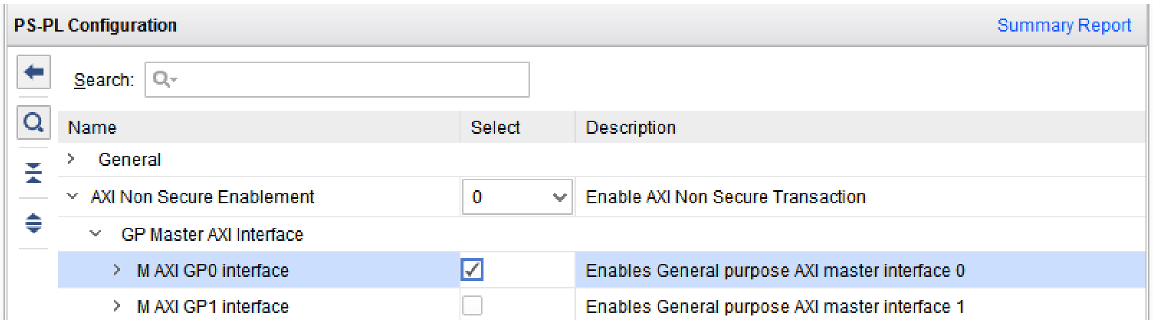


Figure 9 - Customize PS

3. Check the box to select **M AXI GP0 Interface.**



PS-PL Configuration

Summary Report

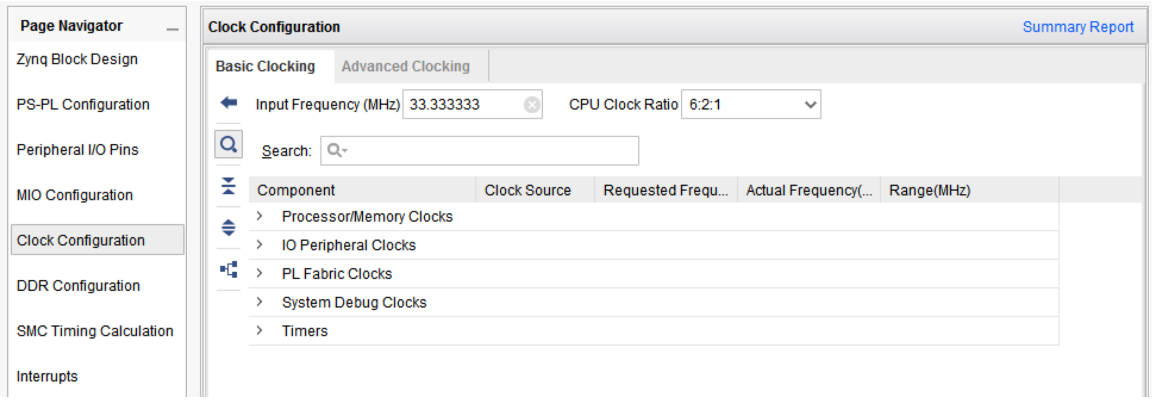
Search:

Name	Select	Description
› General		
⌄ AXI Non Secure Enablement	0	Enable AXI Non Secure Transaction
⌄ GP Master AXI Interface		
› M AXI GP0 interface	☑	Enables General purpose AXI master interface 0
› M AXI GP1 interface	☐	Enables General purpose AXI master interface 1

Figure 10 - Enable General Purpose Master AXI Interface

That takes care of our AXI interface that we were missing. Next we need to export a clock from the PS.

4. Select **Clock Configuration** from the Page Navigator. And **expand all** clocks.



Clock Configuration

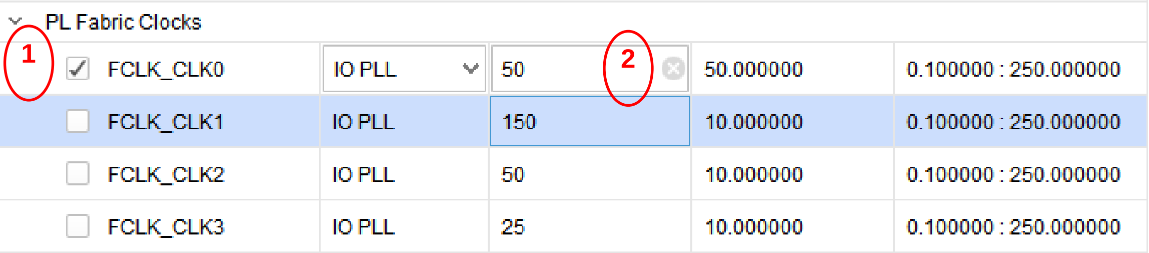
Input Frequency (MHz) 33.333333 CPU Clock Ratio 6:2:1

Search:

Component	Clock Source	Requested Frequ...	Actual Frequency(...	Range(MHz)
› Processor/Memory Clocks				
› IO Peripheral Clocks				
› PL Fabric Clocks				
› System Debug Clocks				
› Timers				

Figure 11 - Clock Configuration

5. Enable Fabric Clock 0 by checking the box next to *FCLK_CLK0*. Set it to **50MHz**.



⌄ PL Fabric Clocks

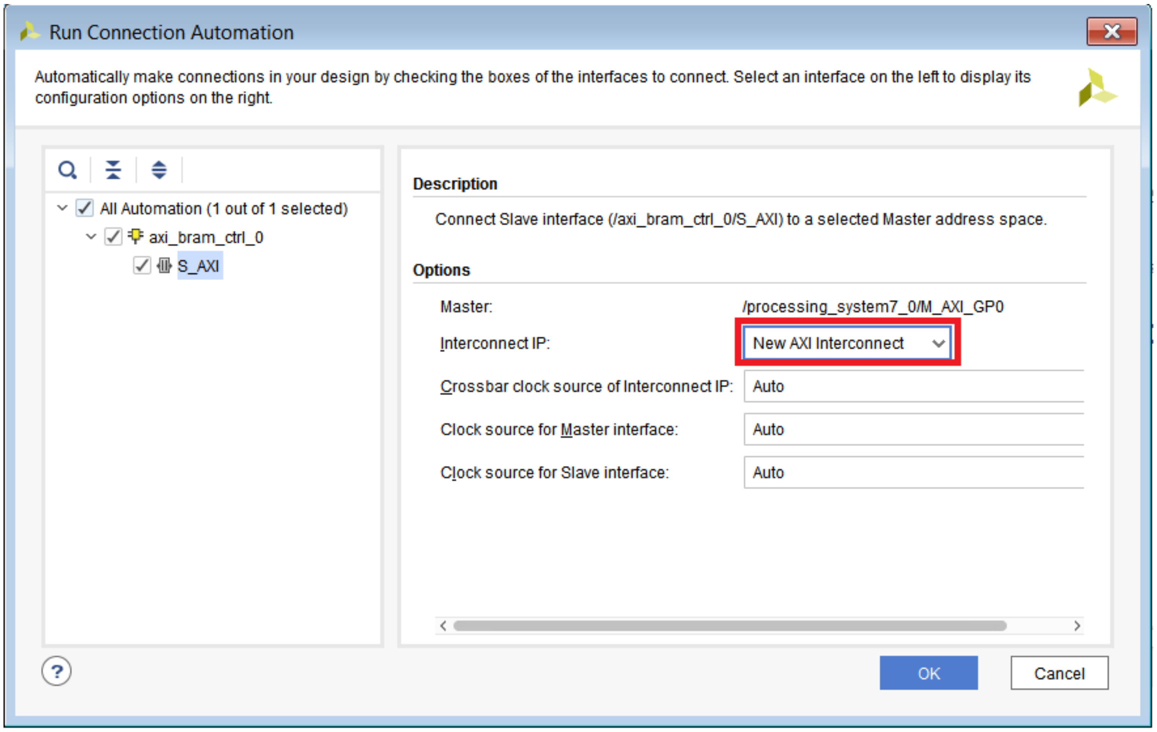
☑ FCLK_CLK0	IO PLL	50	50.000000	0.100000 : 250.000000
□ FCLK_CLK1	IO PLL	150	10.000000	0.100000 : 250.000000
□ FCLK_CLK2	IO PLL	50	10.000000	0.100000 : 250.000000
□ FCLK_CLK3	IO PLL	25	10.000000	0.100000 : 250.000000

Figure 12 - Enable Fabric Clock 0

6. Click **OK** to exit the PS IP configuration.

7. Now you can see that the PS7 block now has **M_AXI_GP0** and **FCLK_CLK0** connections. Also, the *Designer Assistance* is back! Click **Run Connection Automation**.

8. Make sure **All Automation** is checked, also make sure that Interconnect IP type is “New AXI Interconnect”, then click **OK**.



Run Connection Automation

Automatically make connections in your design by checking the boxes of the interfaces to connect. Select an interface on the left to display its configuration options on the right.

- ☑ All Automation (1 out of 1 selected)
  - ☑ axi_bram_ctrl_0
    - ☑ S_AXI

Description

Connect Slave interface (/axi_bram_ctrl_0/S_AXI) to a selected Master address space.

Options

Master:	/processing_system7_0/M_AXI_GP0
Interconnect IP:	New AXI Interconnect
Crossbar clock source of Interconnect IP:	Auto
Clock source for Master interface:	Auto
Clock source for Slave interface:	Auto

Figure 13 – Connection Automation for BRAM Controller S_AXI

9. When complete, regenerate the layout ↻. You should see something similar to below.

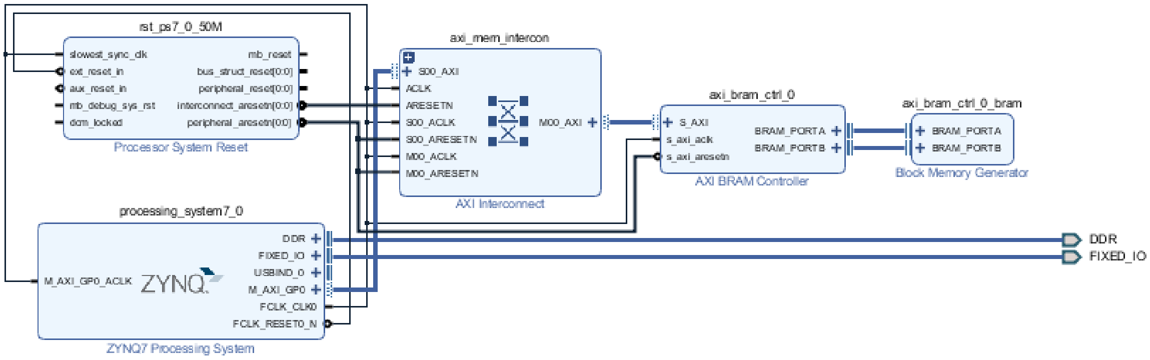


Figure 14 – Block Design with AXI BRAM Controller and AXI Interconnect Connections

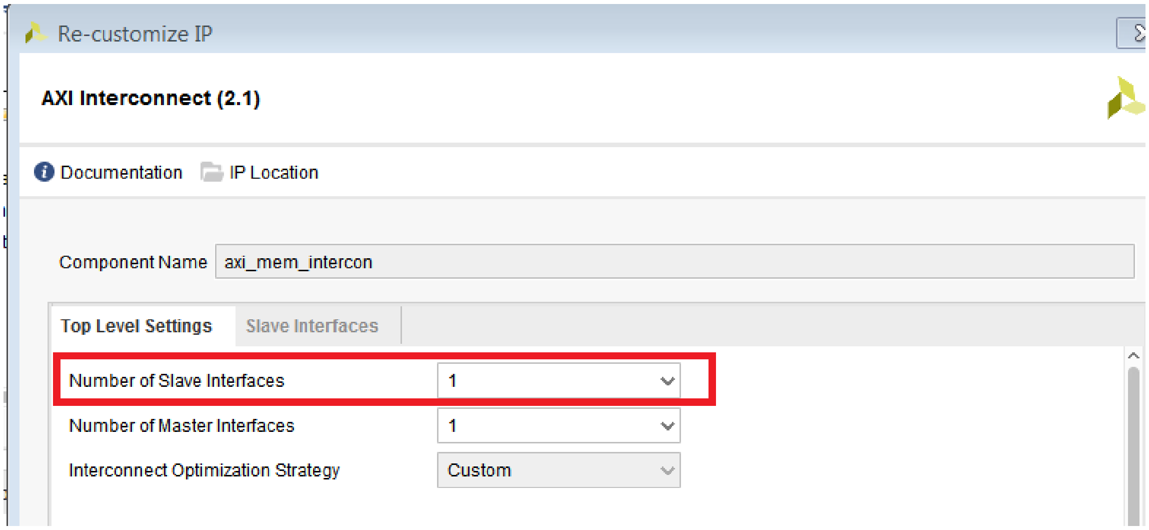
The Designer Assistant has added the following IP:

- AXI Interconnect
- Processor System Reset

The Designer Assistant has also made the following connections:

- BRAM Controller S_AXI to AXI Interconnect M_AXI
- AXI Interconnect S_AXI to PS7 M_AXI
- PS7's FCLK_CLK0 in 6 locations
- PS7's FCLK_RESET0_N to the Processor System Reset
- Processor System Reset then resets the AXI Interconnect and BRAM Controller

10. You can double-click and view the AXI Interconnect and see the number of Master interfaces is **one**. That makes sense since we only had 1 slave. If you add more slave IP, then you can simply increase this. Click **OK**.



Re-customize IP

AXI Interconnect (2.1)

Component Name axi_mem_intercon

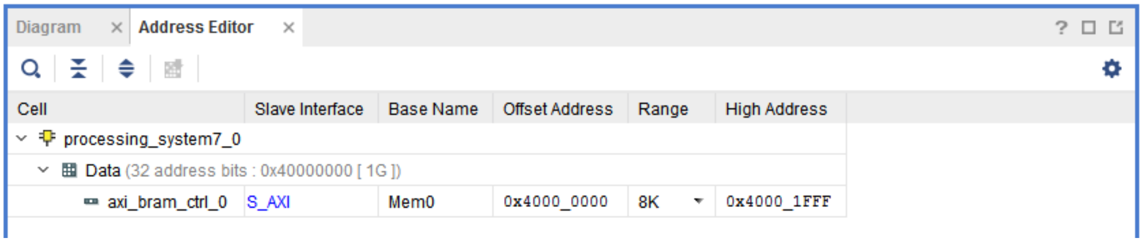
Number of Slave Interfaces 1

Number of Master Interfaces 1

Interconnect Optimization Strategy Custom

Figure 15 - Customize AXI Interconnect

11. We're not quite done yet. We need to view the address space for the BRAM on the AXI interface. To access the Address Editor tab click **Window → Address Editor** at the top of the screen. Then click **expand all** in the **Address Editor** Tab. The Designer Assistant has already assigned the address. If you do this manually, you can use the Auto Assign Address button at the bottom of the shortcut list (which is now grayed out). Here you can adjust the Range if you'd like to have more or less memory. You can also change the Offset Address if there is a particular address you want. For now, we will leave this at 8K located at 0x4000_0000.



Cell	Slave Interface	Base Name	Offset Address	Range	High Address
processing_system7_0					
Data (32 address bits : 0x40000000 [1G])					
axi_bram_ctrl_0	S_AXI	Mem0	0x4000_0000	8K	0x4000_1FFF

Figure 16 - BRAM Address Assignment

12. Switch back to the *Diagram view* by selecting the **Diagram** tab.

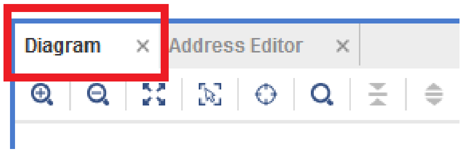


Figure 17 - Diagram Tab

13. Next, we need to validate the design. Vivado has a robust design rule checker for Vivado IP Integrator block designs. Select **Validate Design**. Click **OK** on Validation Successful window.

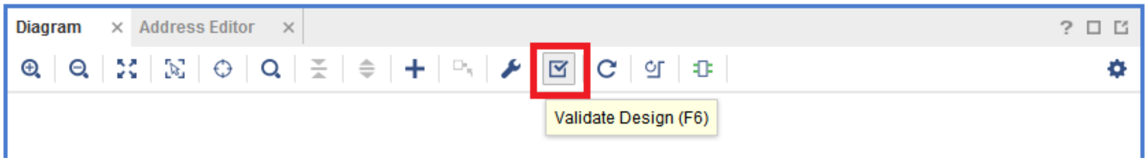
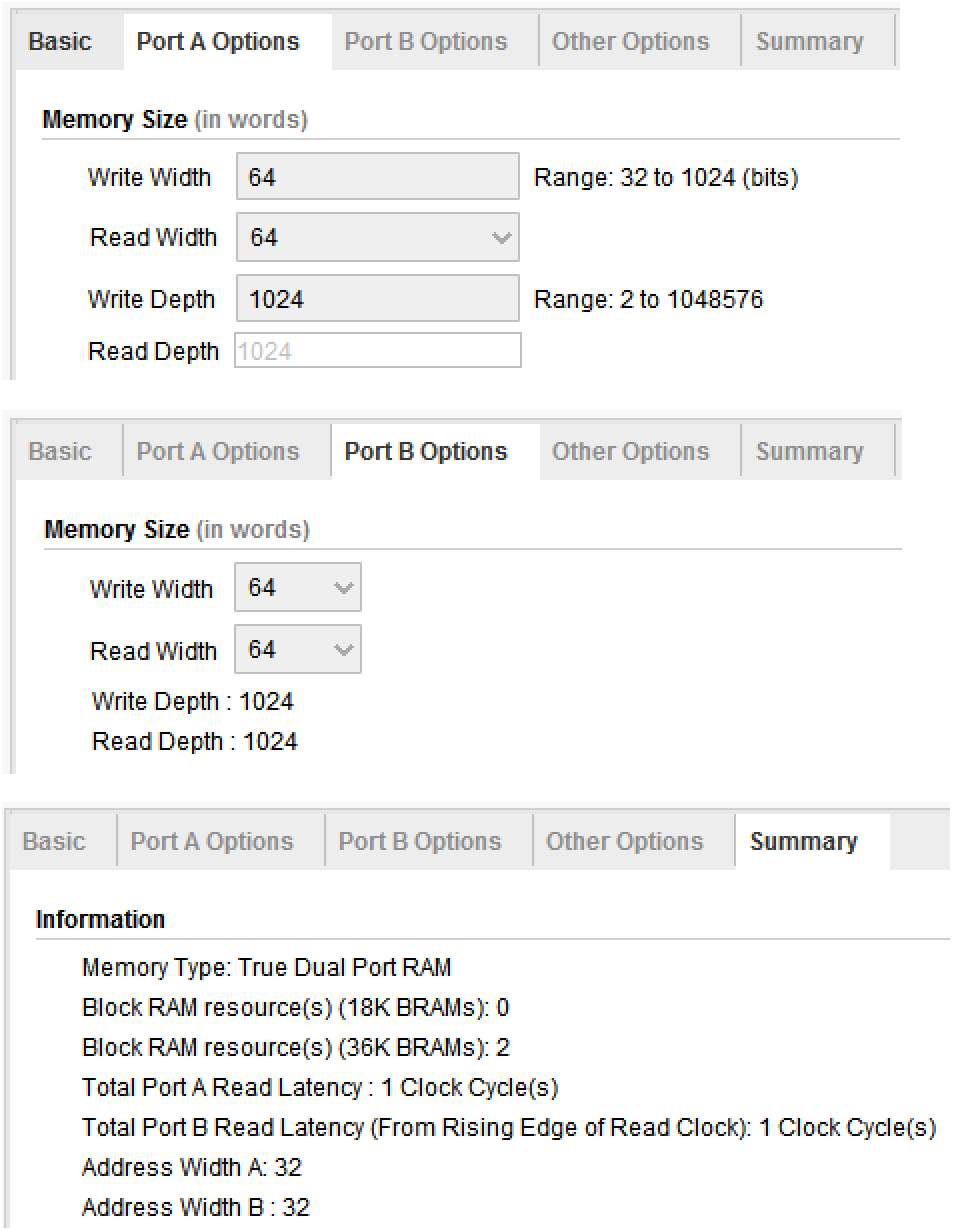


Figure 18 - Validate Design

14. View the Block Memory Generator IP, **axi_bram_ctrl_0_bram**, by double-clicking on the IP.

15. Click on the **Port A and B Option** tabs to see the data widths and depth. Also click on the **Summary** tab to see how many BRAM resources will be used by this IP. **Note**: The width and depth parameters are inherited from the BRAM Controller IP and updated when the design was validated.



Memory Size (in words)

Write Width 64 Range: 32 to 1024 (bits)

Read Width 64

Write Depth 1024 Range: 2 to 1048576

Read Depth 1024

Memory Size (in words)

Write Width 64

Read Width 64

Write Depth : 1024

Read Depth : 1024

Information

Memory Type: True Dual Port RAM

Block RAM resource(s) (18K BRAMs): 0

Block RAM resource(s) (36K BRAMs): 2

Total Port A Read Latency : 1 Clock Cycle(s)

Total Port B Read Latency (From Rising Edge of Read Clock): 1 Clock Cycle(s)

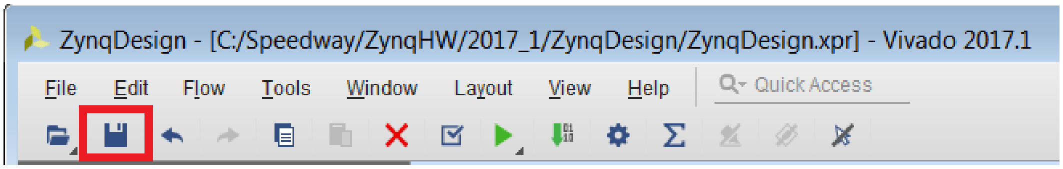
Address Width A: 32

Address Width B : 32

Figure 19 - BRAM Options

16. Click **OK** to accept the settings.

17. Save the block design by typing **save_bd_design** in the TCL Console or by clicking the save block design button from the top shortcut bar.



ZynqDesign - [C:/Speedway/ZynqHW/2017_1/ZynqDesign/ZynqDesign.xpr] - Vivado 2017.1

Figure 20 - Save Block Design

18. Reset and then generate the Output Products by right-clicking on the **Z_system_i** under *Sources*.

19. Guide Vivado 2017.1 to insert global clock buffer to the FCLK0 by giving command **set_property CONFIG.PCW_FCLK_CLK0_BUF TRUE [get_bd_cells /processing_system7_0]** to the TCL-window. This will improve the timing considerably if Vivado has not done this automatically.

20. Select **Generate Bitstream** from the Flow Navigator. Click **Yes** to launch synthesis and implementation. Then Select OK on the Launch Runs window. This will take a few minutes. Answer the questions below while waiting.

Questions:

Answer the following questions:

- *How many BRAM's are consumed by the Block Memory Generator?*

____

- *What is the base address of the BRAM?   Why is it mapped here?*

____

- *If more IP peripherals were connected, where would they connect?*

____

Exploring Further

If you have more time and would like to investigate more…

- Explore what other IP is available in the catalog. Note: if you add it to your design, make sure to remove it before proceeding to the next lab.

This concludes Lab 5.

Revision History

Date	Version	Revision
6 Nov 13	02	Initial Draft
19 Nov 13	03	Pilot updates
5 Nov 14	04	Updated for Vivado 2014.3
5 Jan 15	05	Updated for Vivado 2014.4
05 Mar 15	06	Finalize for Vivado 2014.4
16 Mar 15	07	Minor edits for release
Oct 15	08	Updated to Vivado 2015.2
July 2016	09	Updated to Vivado 2016.2
May 2017	10	Updated to Vivado 2017.1
June 2017	11	Updated to Vivado 2017.1 for MiniZed + Rebranding

Resources

www.minized.org

www.microzed.org

www.picozed.org

www.zedboard.org

www.xilinx.com/zynq

www.xilinx.com/sdk

www.xilinx.com/vivado

Answers

Experiment 2

- *How many BRAM's are consumed by the Block Memory Generator?*

2

- *What is the base address of the BRAM?   Why is it mapped here?*

0x40000000, because this is starting address space for M_AXI_GP0 (refer to the Zynq All Programmable SoC User Guide).

- *If more IP peripherals were connected, where would they connect?*

Add Master AXI ports to the AXI Interconnect Block for additional slaves to connect into.