

Creating Multi Processor Solutions

Course Workbook

Table of Contents

About this Workbook	<u>Page 3</u>
Pre-Lab: Workshop Pre-requisites	<u>Page 4</u>
Lab 2: Creating Multi Processor Solutions	<u>Page 7</u>

About this Workbook

The contents of this workbook are created by Aduvo Engineering & Training, Ltd.

If you have any questions about the contents, or need assistance, please contact Adam Taylor at adam@aduvoengineering.com.

Pre-Lab

Creating Multi Processor Solutions

Required Hardware

ArtyZ7-20

Usb cable

Downloads and Installations

Step 1 – Download and install the following at least 1 day prior to the workshop. This may take a significant amount of time and drive space.

Watch the video available [here](#) to show how to configure the installation

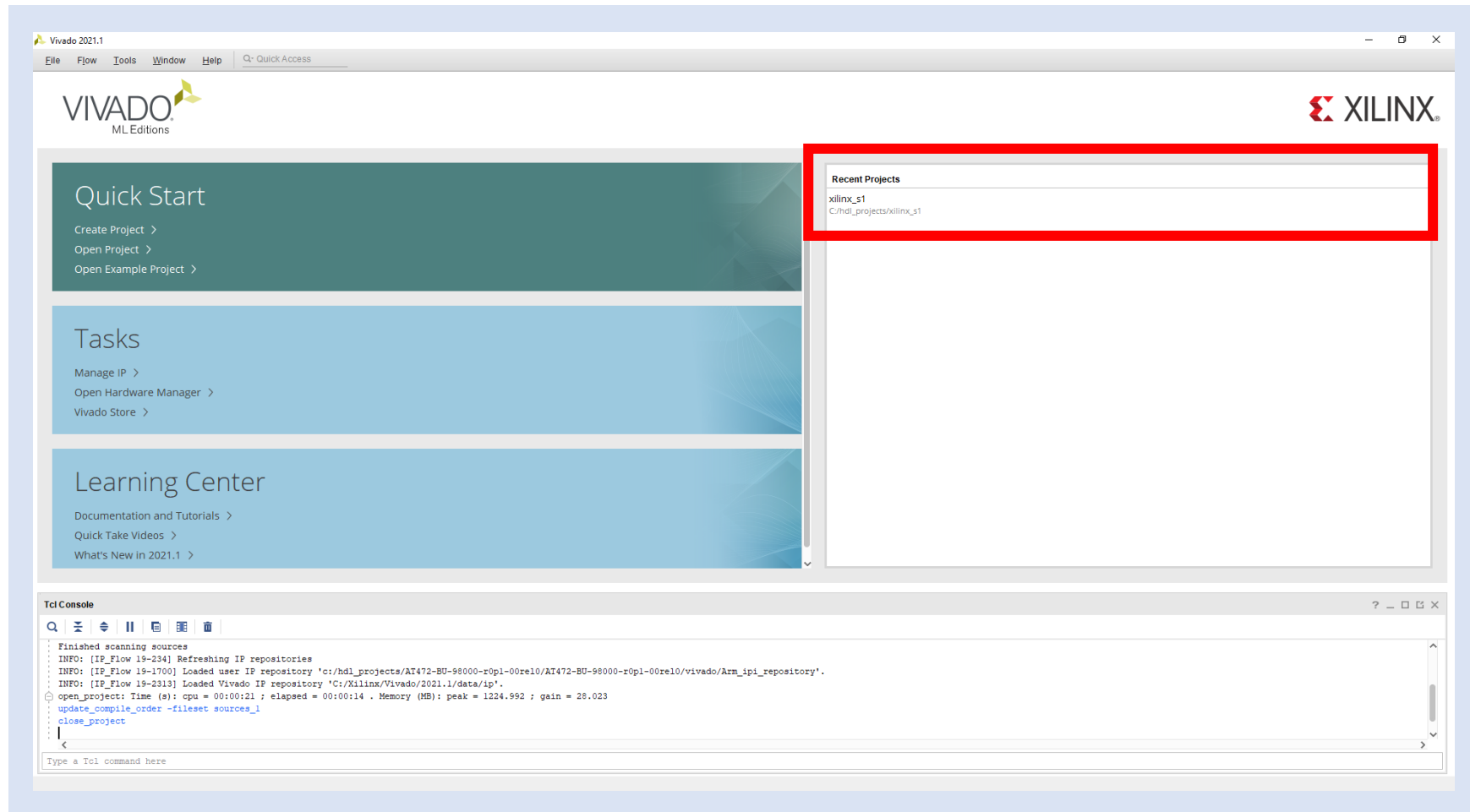
Vitis 2021.1	Download
--------------	--------------------------

Lab 2

Project creation & Flow

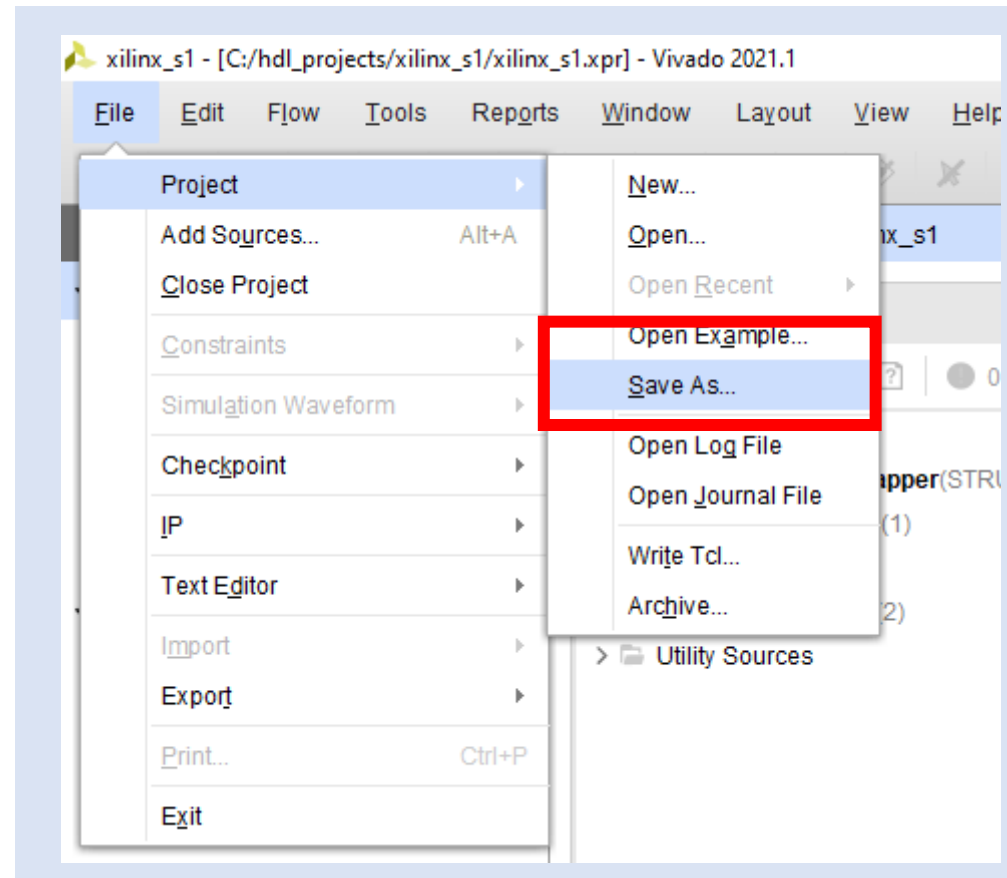
Lab 2: Understanding Vitis Project creation & Flow

Step 1 – Open Vivado 2021.1 – Open the project from session one



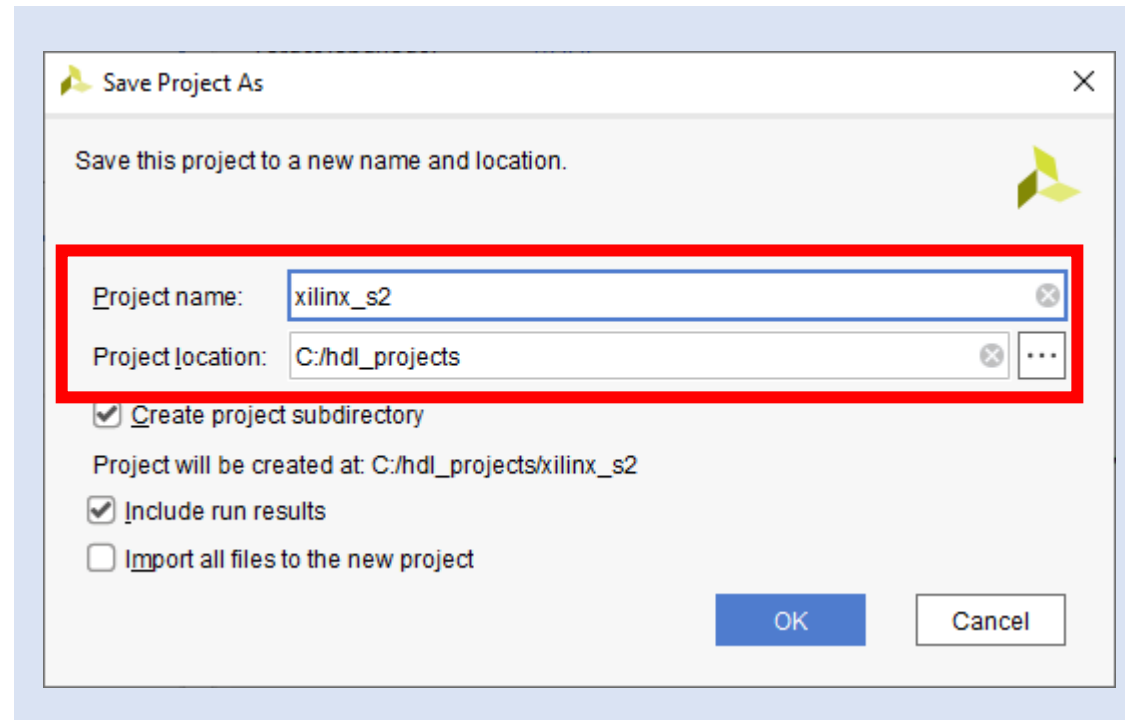
Lab 2: Understanding Vitis Project creation & Flow

Step 2 – From File->Project select save as and name it session two



Lab 2: Understanding Vitis Project creation & Flow

Step 3 – From File->Project select save as and name it session two



Lab 2: Understanding Vitis Project creation & Flow

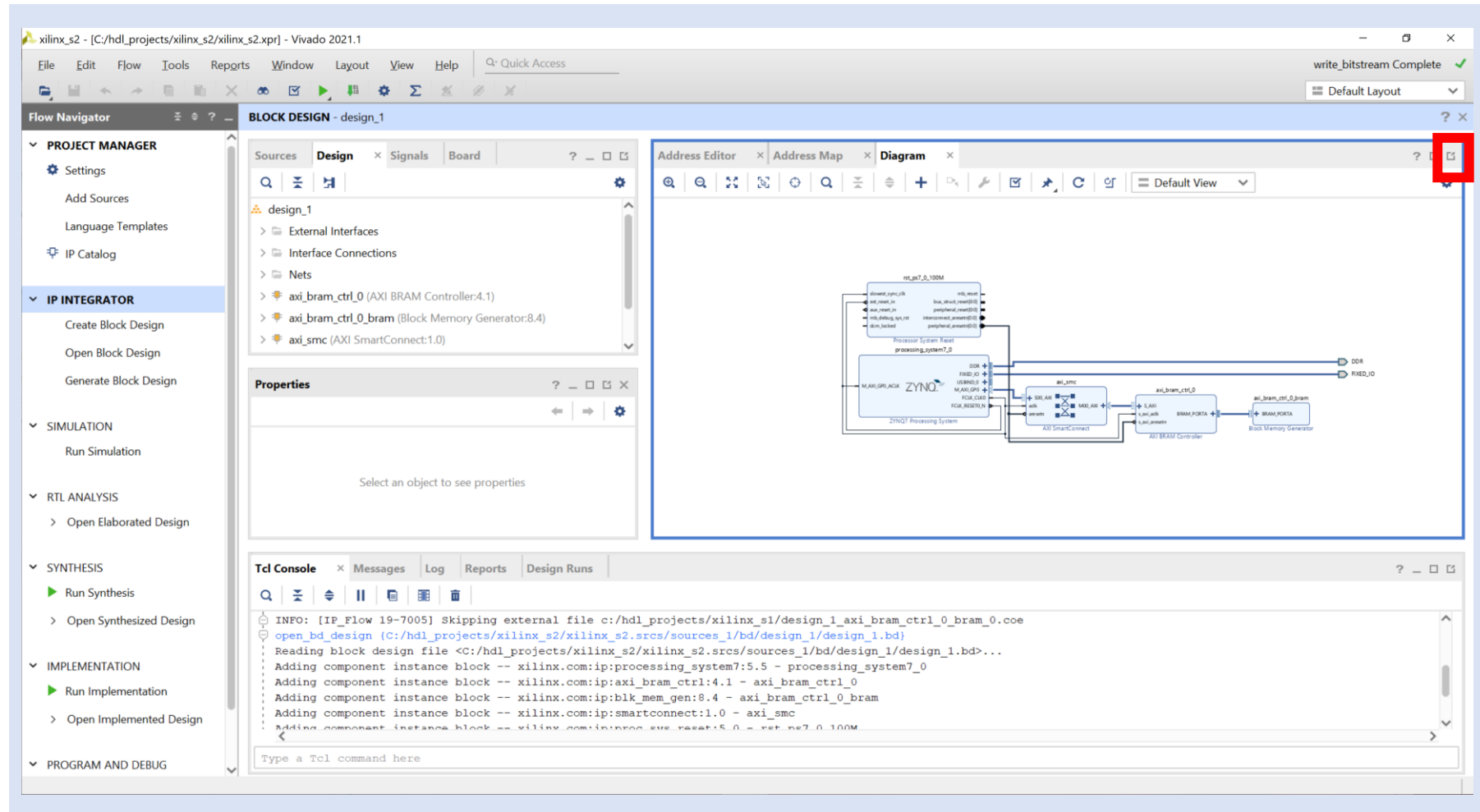
Step 4 – Open the Block Diagram

The screenshot displays the Vivado 2021.1 Project Manager interface. The left sidebar contains a 'PROJECT MANAGER' section with a red box highlighting the 'Open Block Design' option. The main area shows the 'Sources' pane with 'design_1_wrapper' selected. The 'Project Summary' pane on the right provides details about the project, including the board part (Arty Z7-20) and the status of the synthesis and implementation runs. The bottom pane shows the 'Design Runs' table with columns for Name, Constraints, Status, and various resource metrics.

Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAM	URAM	DSP	Start	Elapsed	Run Strategy
✓ synth_1 (active)	constrs_1	synth_design Complete!								0	0	0.0	0	0	11/4/21, 10:14 AM	00:00:56	Vivado Synthesis Defaults (Vivado Synthesis 2021)
✓ impl_1	constrs_1	write_bitstream Complete!	1.257	0.000	0.048	0.000	0.000	1.415	0	1489	1182	7.5	0	0	11/4/21, 10:15 AM	00:02:12	Vivado Implementation Defaults (Vivado Implementation 2021)
Out-of-Context Module Runs																	
> design_1		Submodule Runs Complete													11/3/21, 2:46 PM	00:04:27	

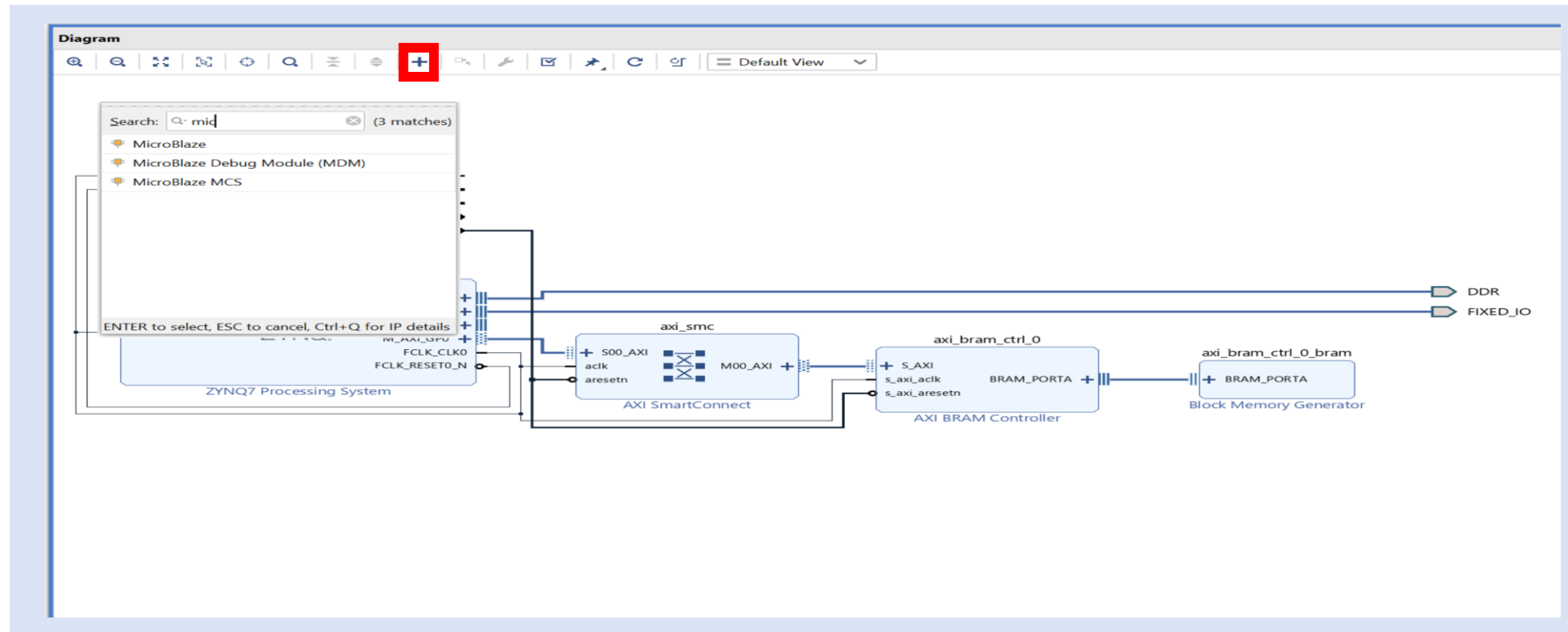
Lab 2: Understanding Vitis Project creation & Flow

Step 5 – Un dock the block diagram, and maximize it



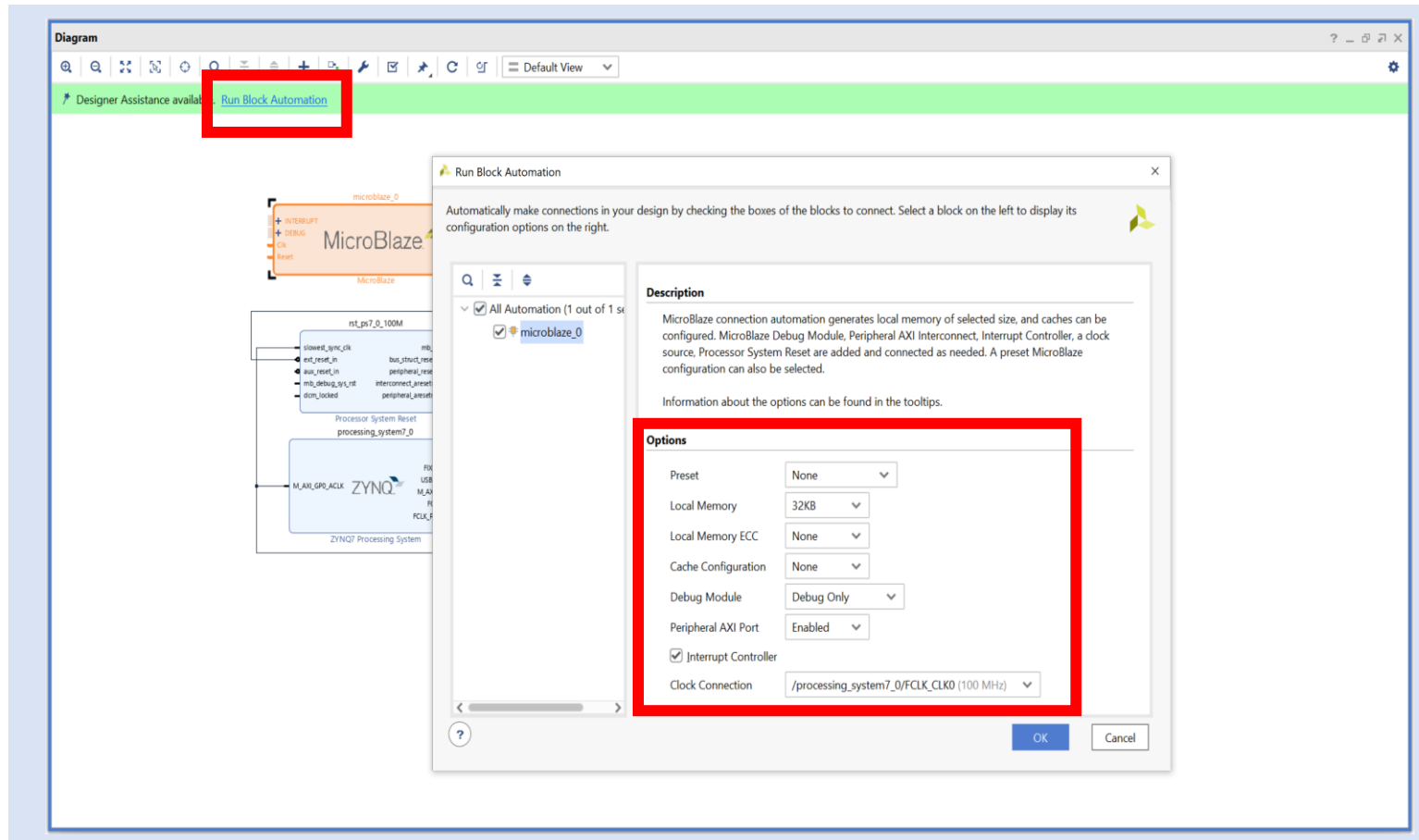
Lab 2: Understanding Vitis Project creation & Flow

Step 6 – Click on + and in the search bar, type microblaze – double click on the MicroBlaze IP to add it to IP Integrator



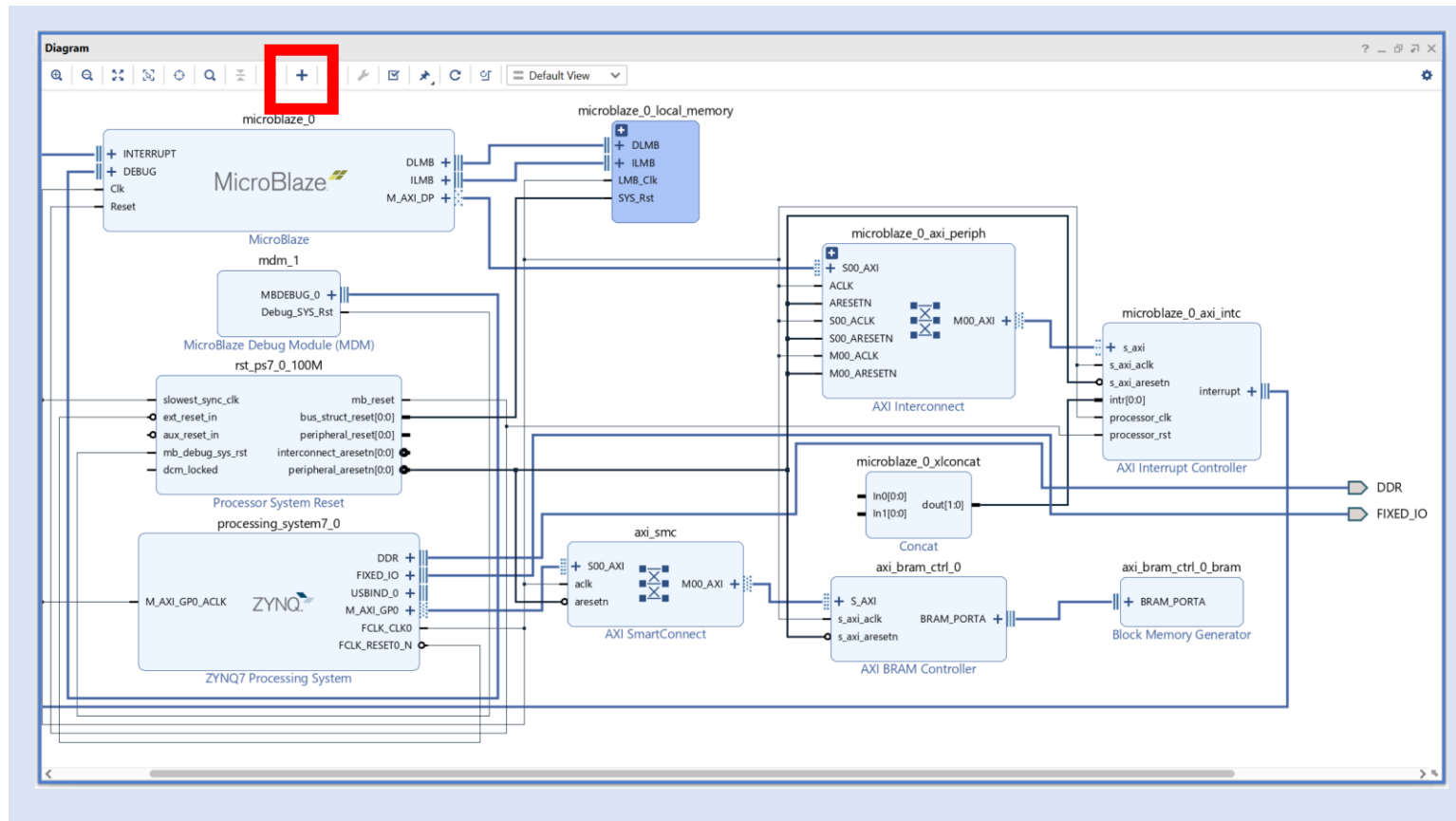
Lab 2: Understanding Vitis Project creation & Flow

Step 7 – Click on Run Block Automation (this will help you configure the MicroBlaze), Select Local Memory 32KB, No ECC, No Cache, Debug only, AXI enabled, check Interrupt, Clock 100 MHz click OK



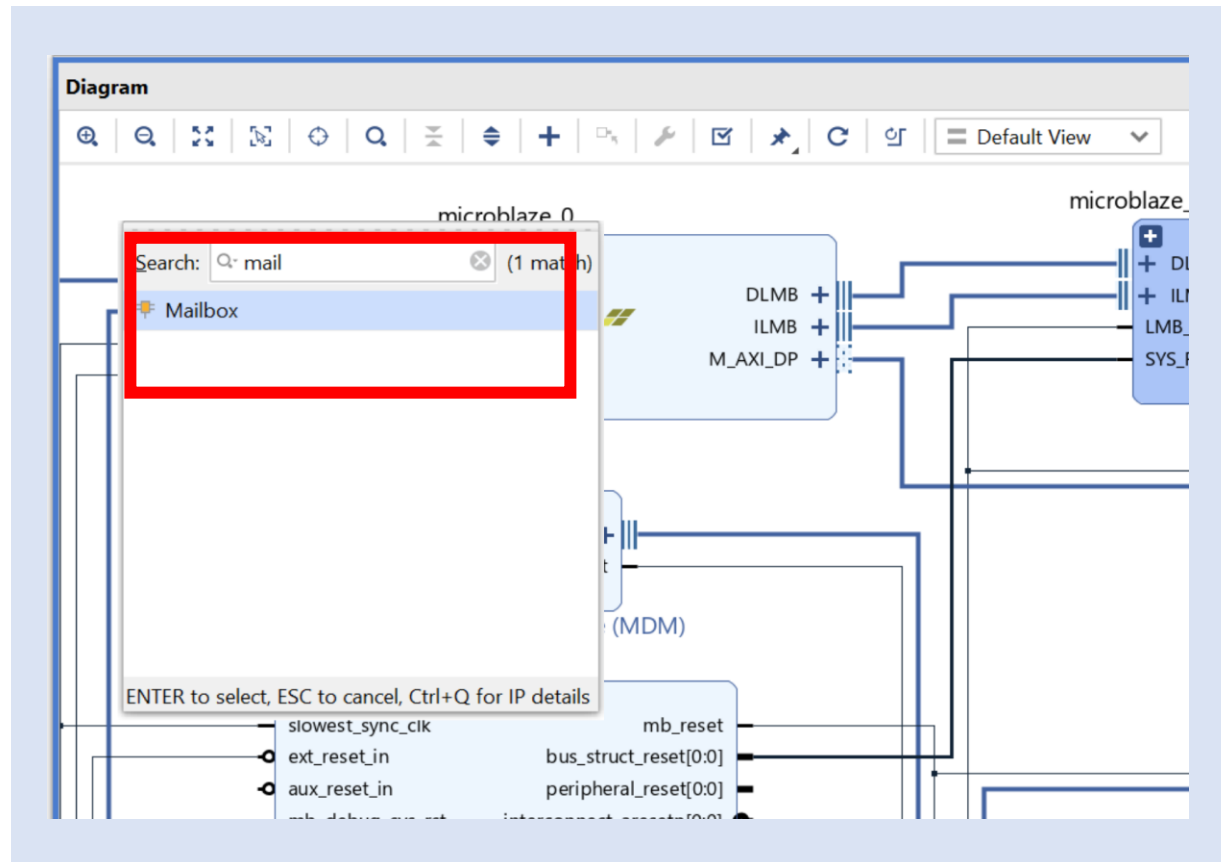
Lab 2: Understanding Vitis Project creation & Flow

Step 8 – Once the automation finishes, click on the + symbol



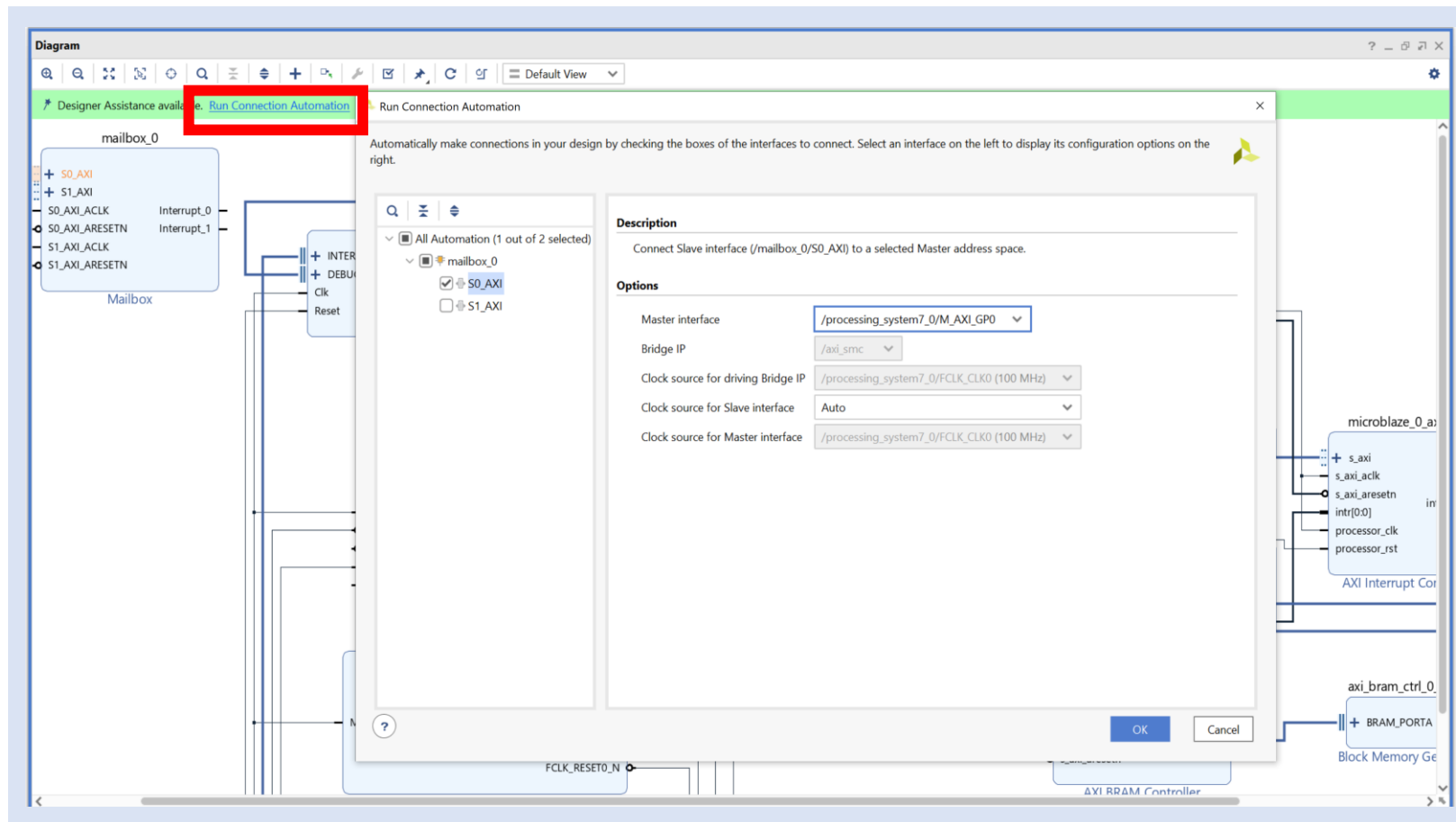
Lab 2: Understanding Vitis Project creation & Flow

Step 9 – Type in Mailbox, double click and add it to the diagram



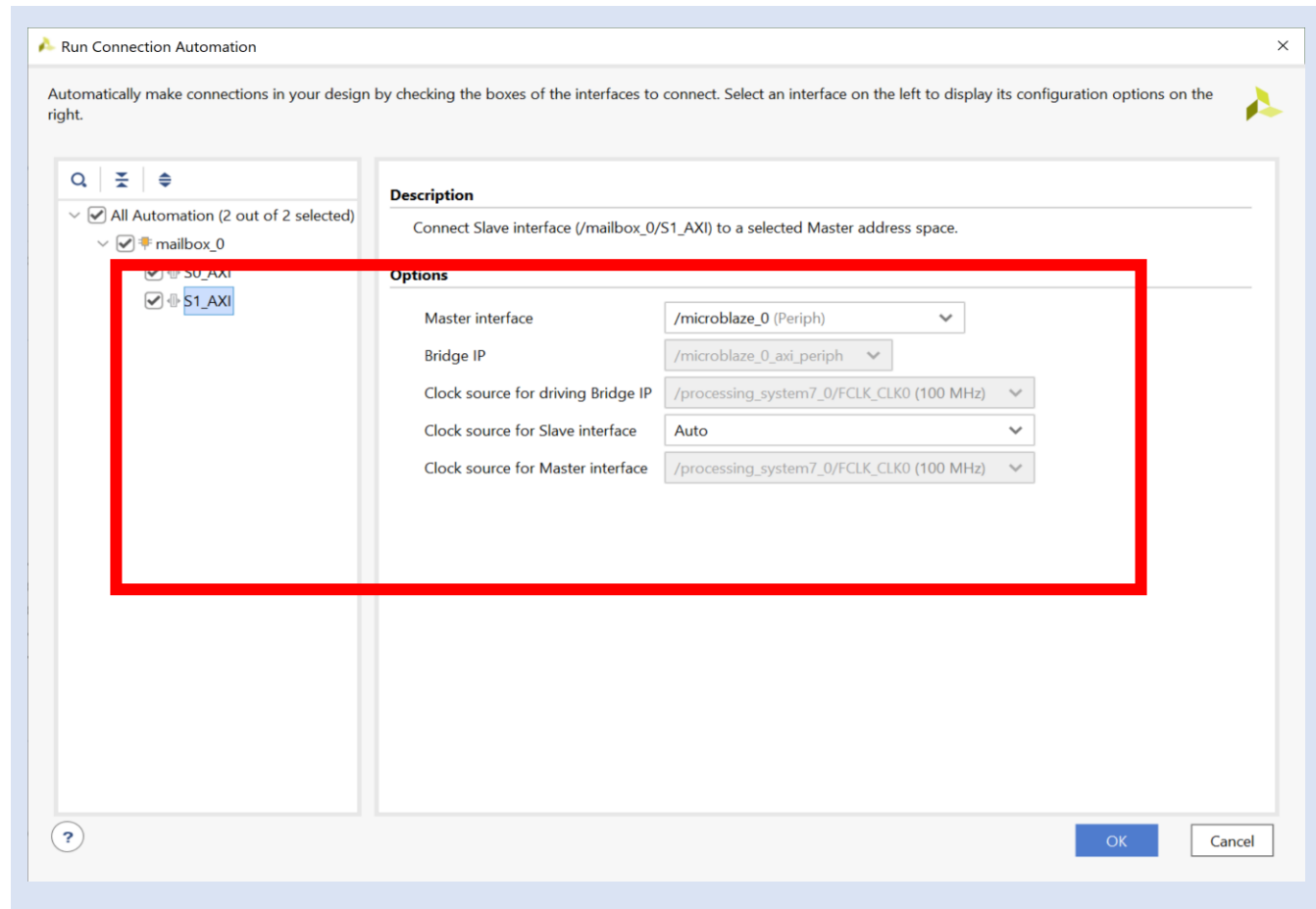
Lab 2: Understanding Vitis Project creation & Flow

Step 10 – Click on Run Connection Automation, Check Mailbox. For S0_AXI set master interface to M_AXI_GP0 on the Processing System



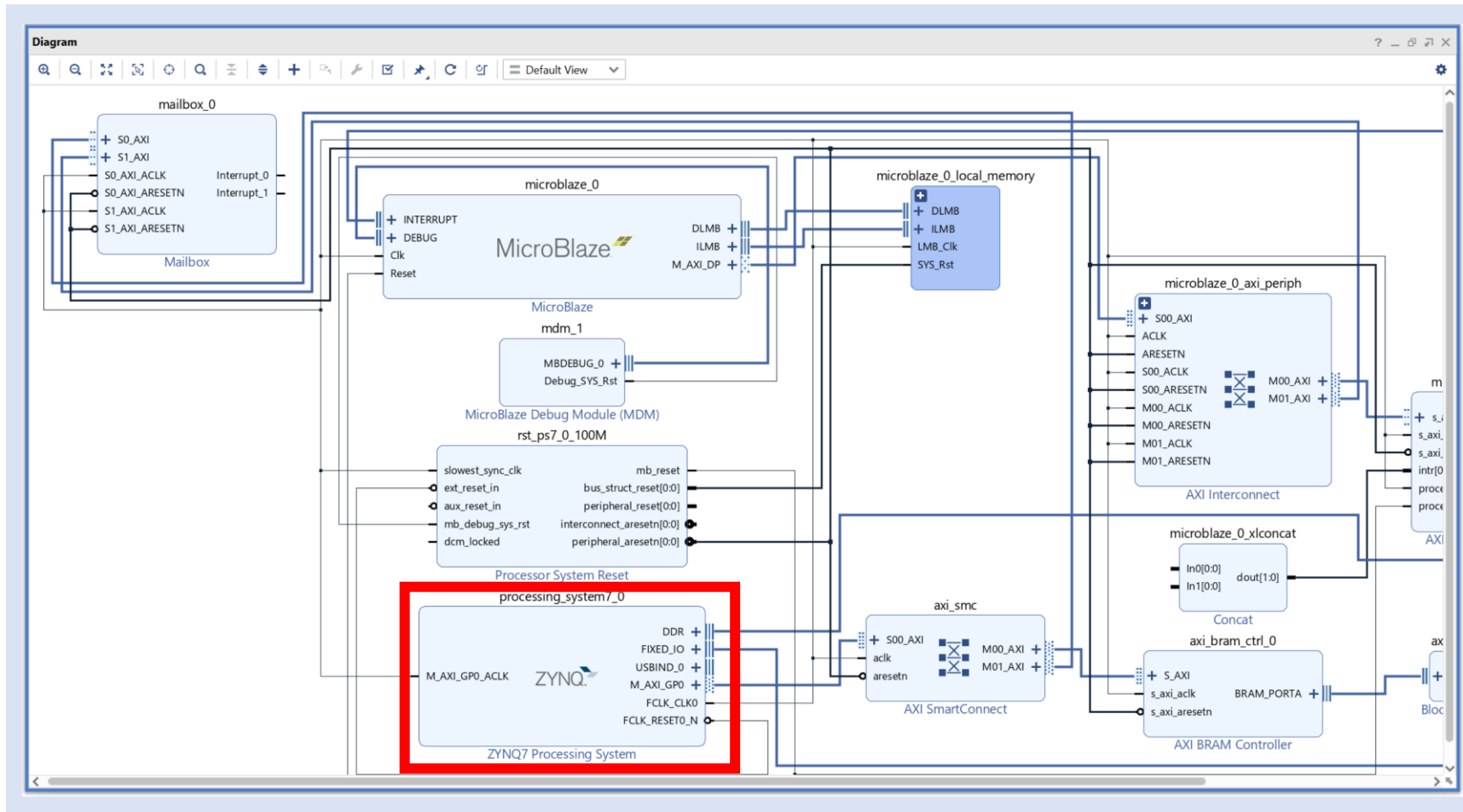
Lab 2: Understanding Vitis Project creation & Flow

Step 11 – Click on S1_AXI and select the master interface to be the MicroBlaze system.



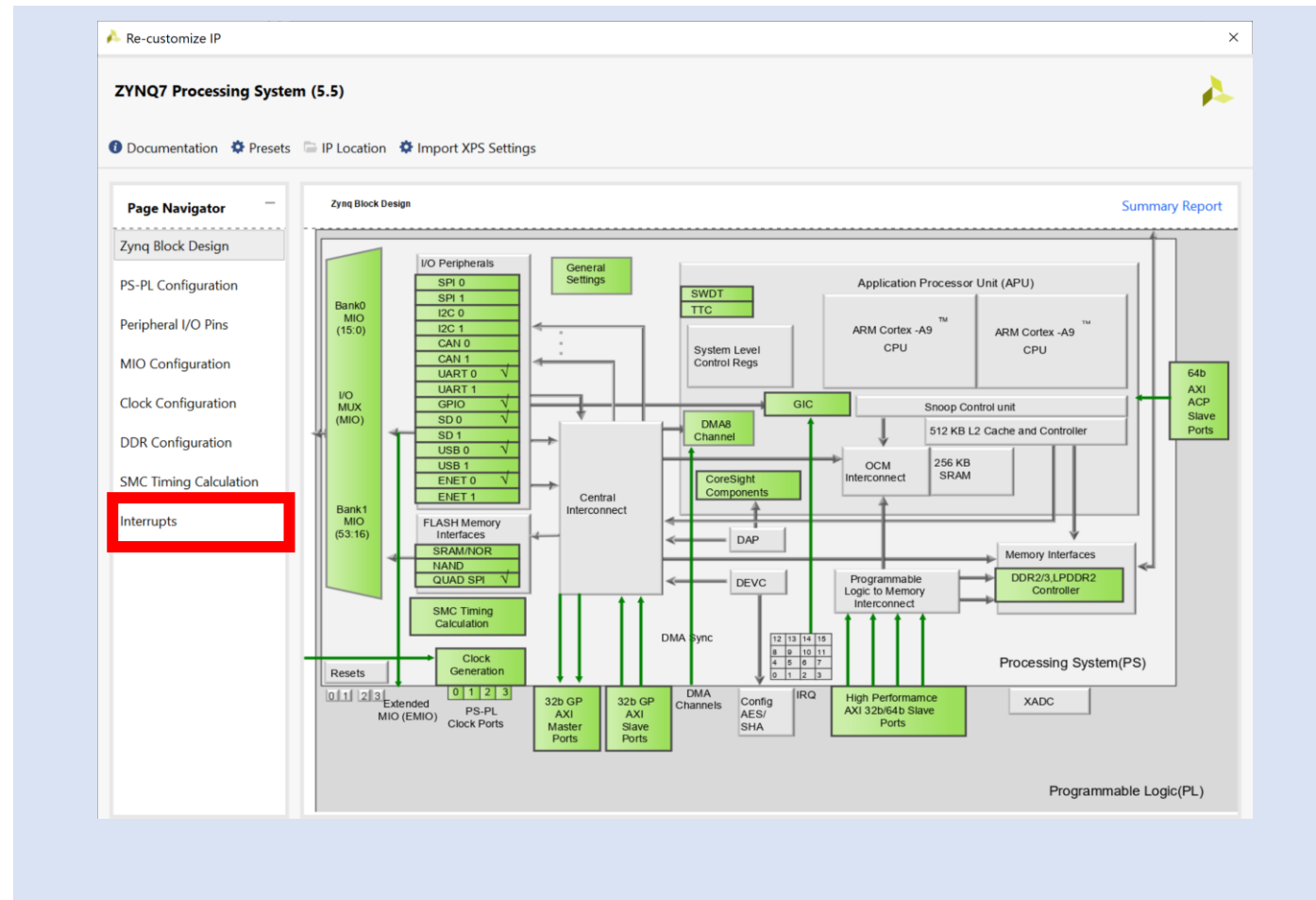
Lab 2: Understanding Vitis Project creation & Flow

Step 12 – Click the processing system



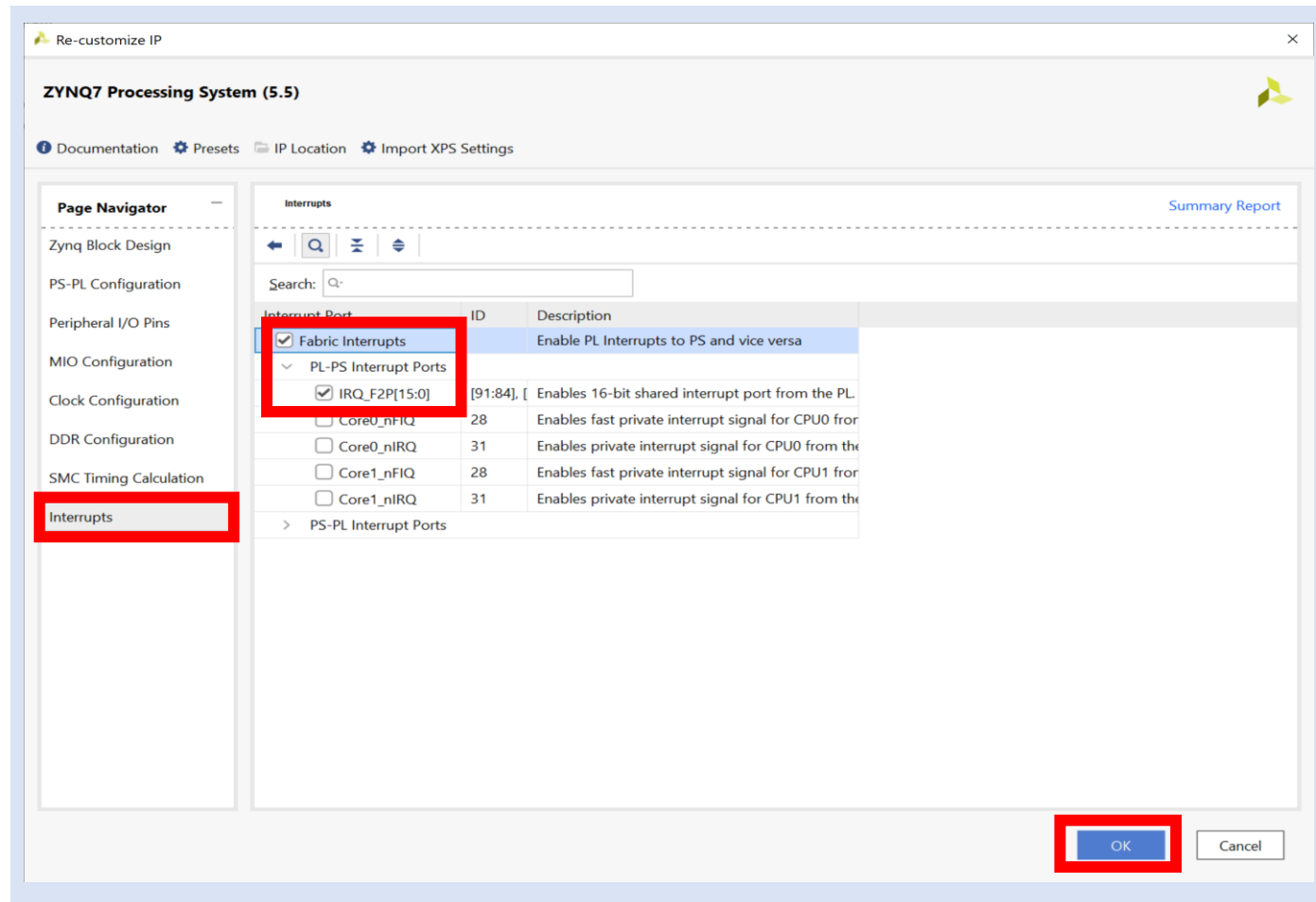
Lab 2: Understanding Vitis Project creation & Flow

Step 13 – Click interrupts



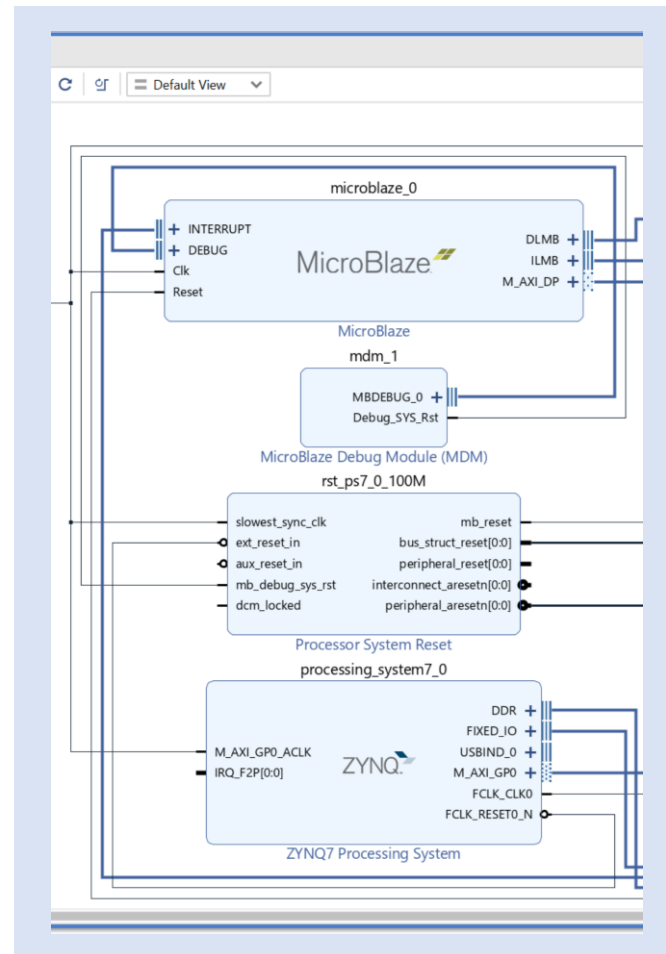
Lab 2: Understanding Vitis Project creation & Flow

Step 14 – Click interrupts, select Fabric Interrupts, IRQ_R2P and click OK



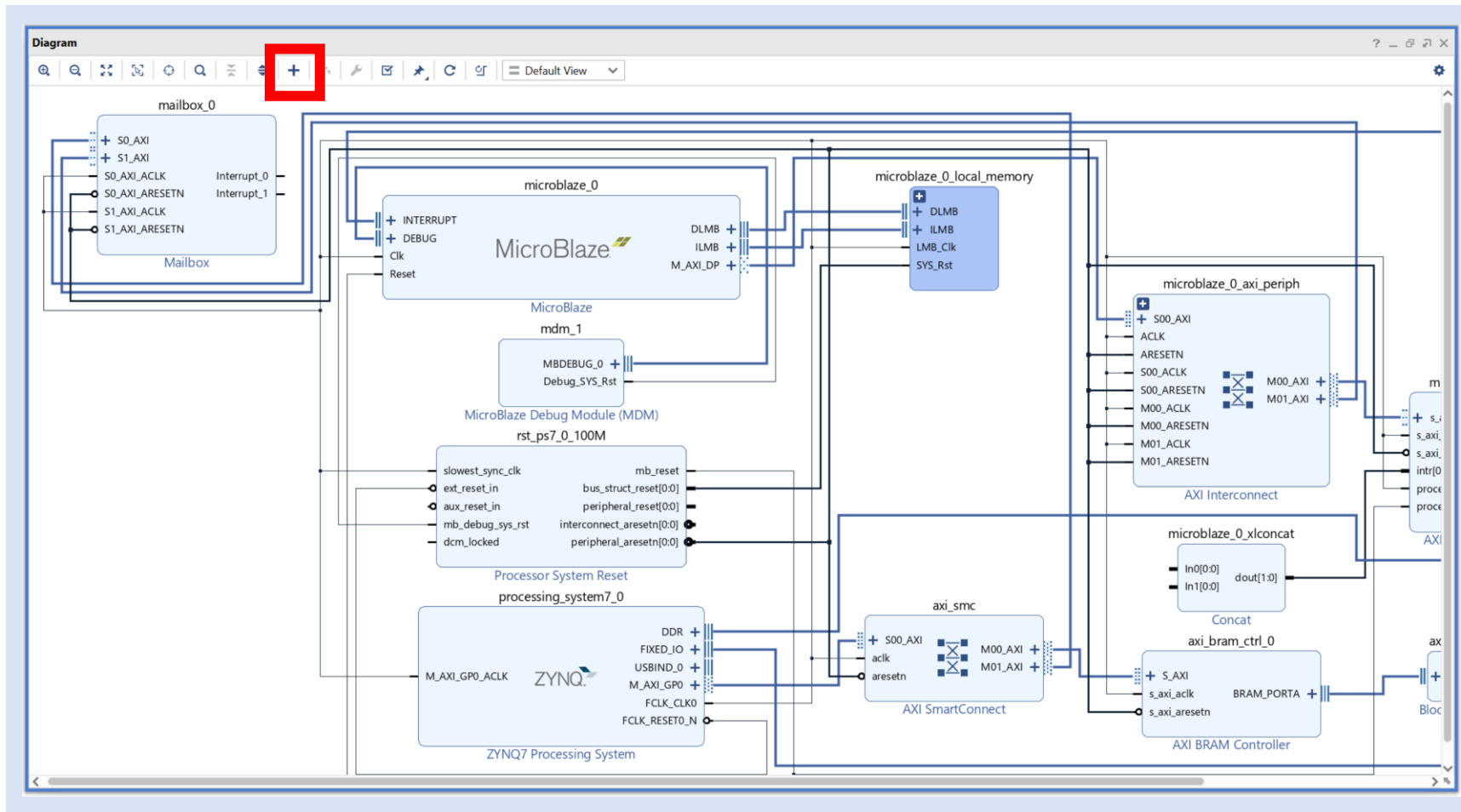
Lab 2: Understanding Vitis Project creation & Flow

Step 15 – This will update the processing system block to include IRQ_F2P interrupt



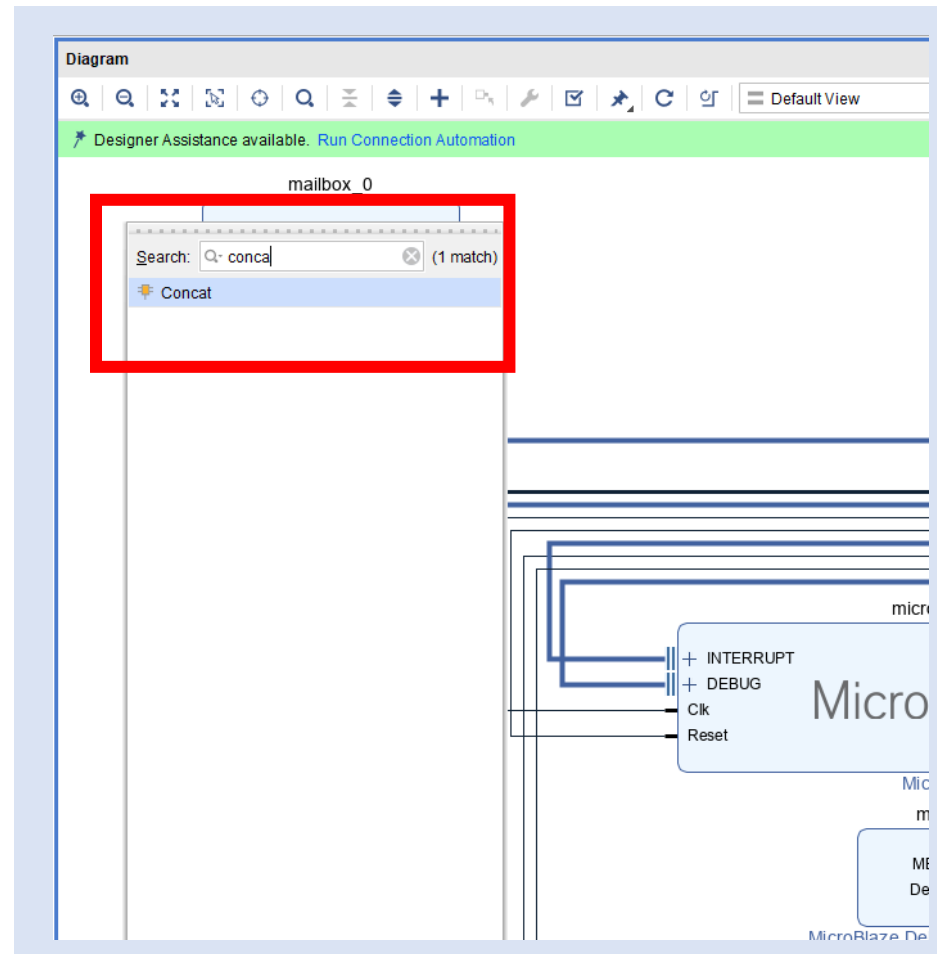
Lab 2: Understanding Vitis Project creation & Flow

Step 16 – Click +



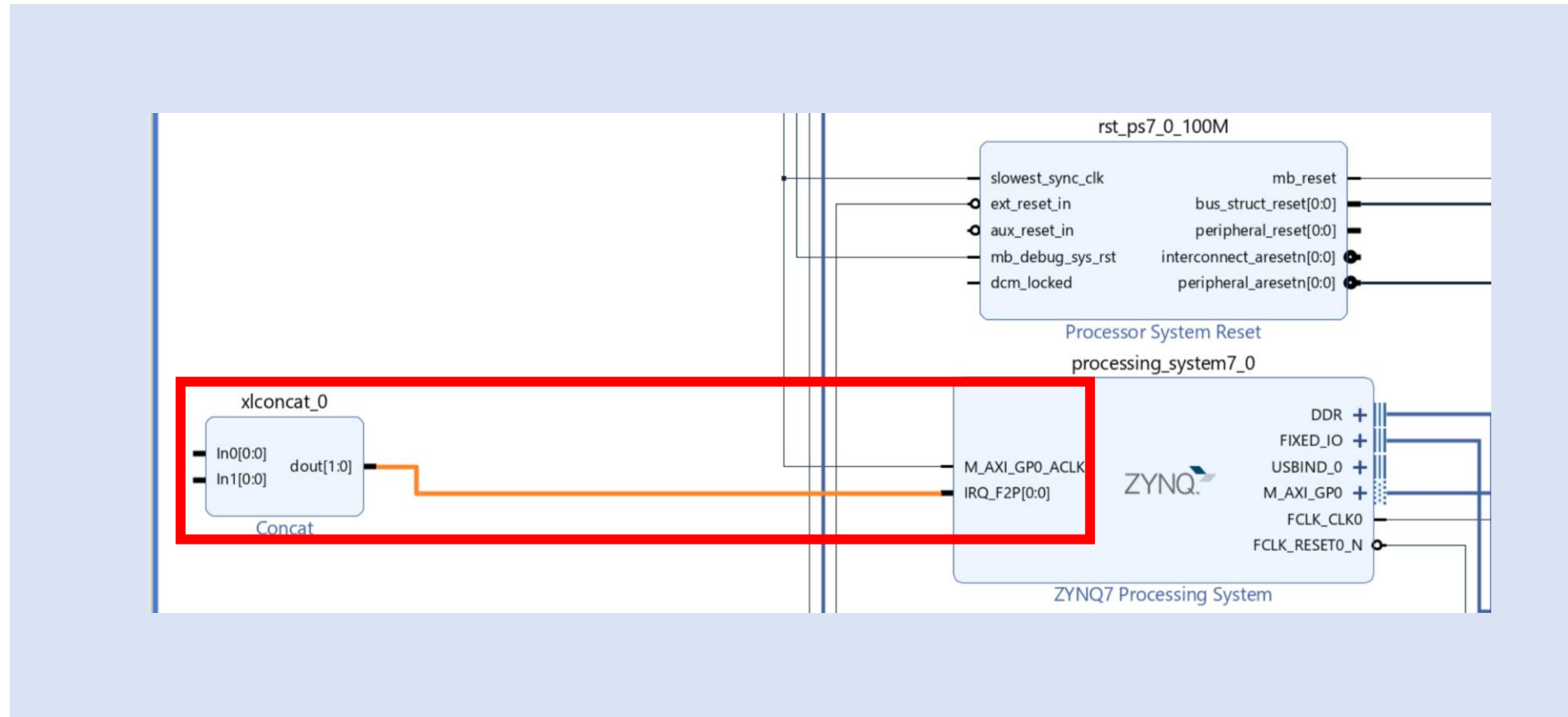
Lab 2: Understanding Vitis Project creation & Flow

Step 17 – Type Concat and double click to add in the concatenation block



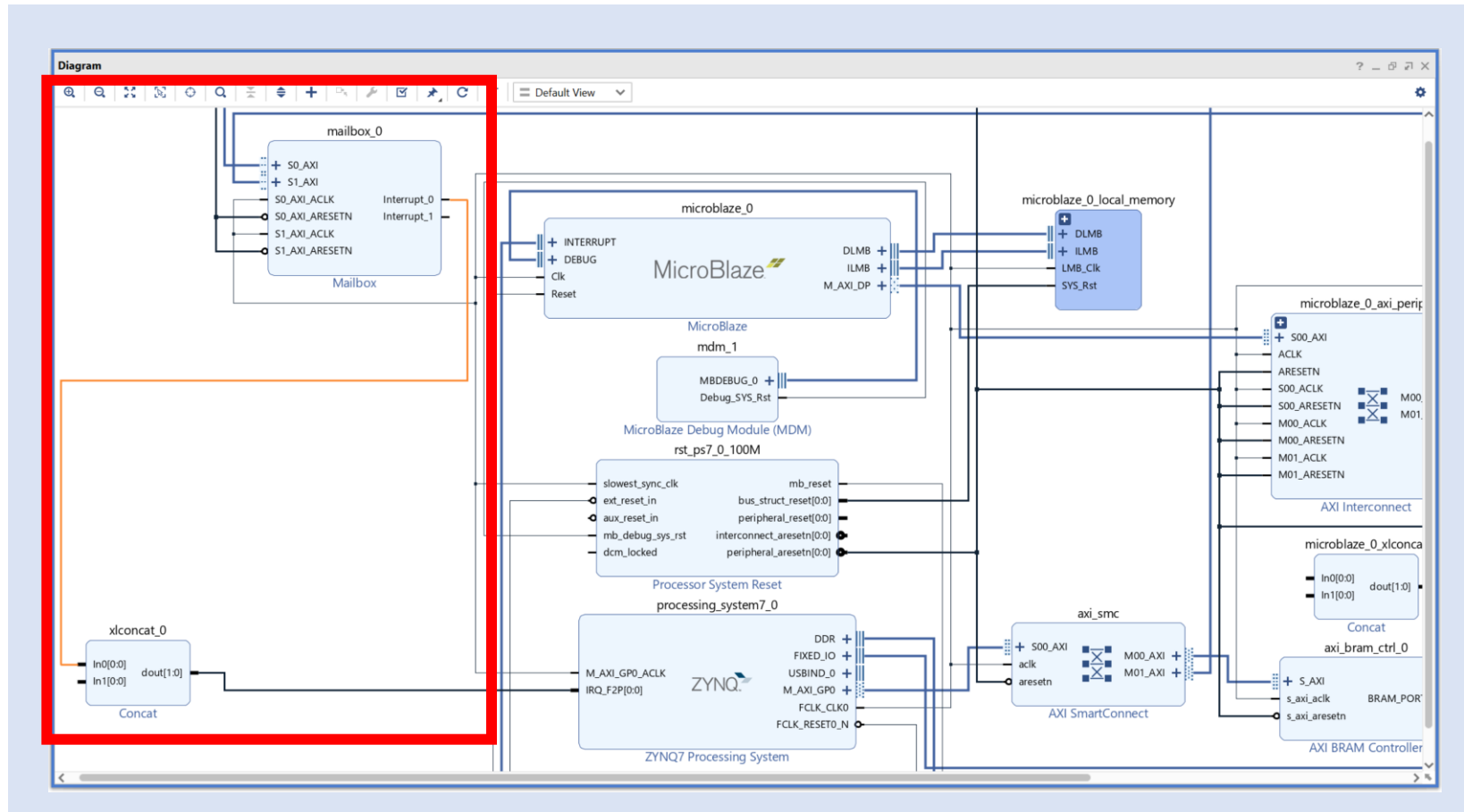
Lab 2: Understanding Vitis Project creation & Flow

Step 18 – Connect the Concat Block to the IRQ_F2P input



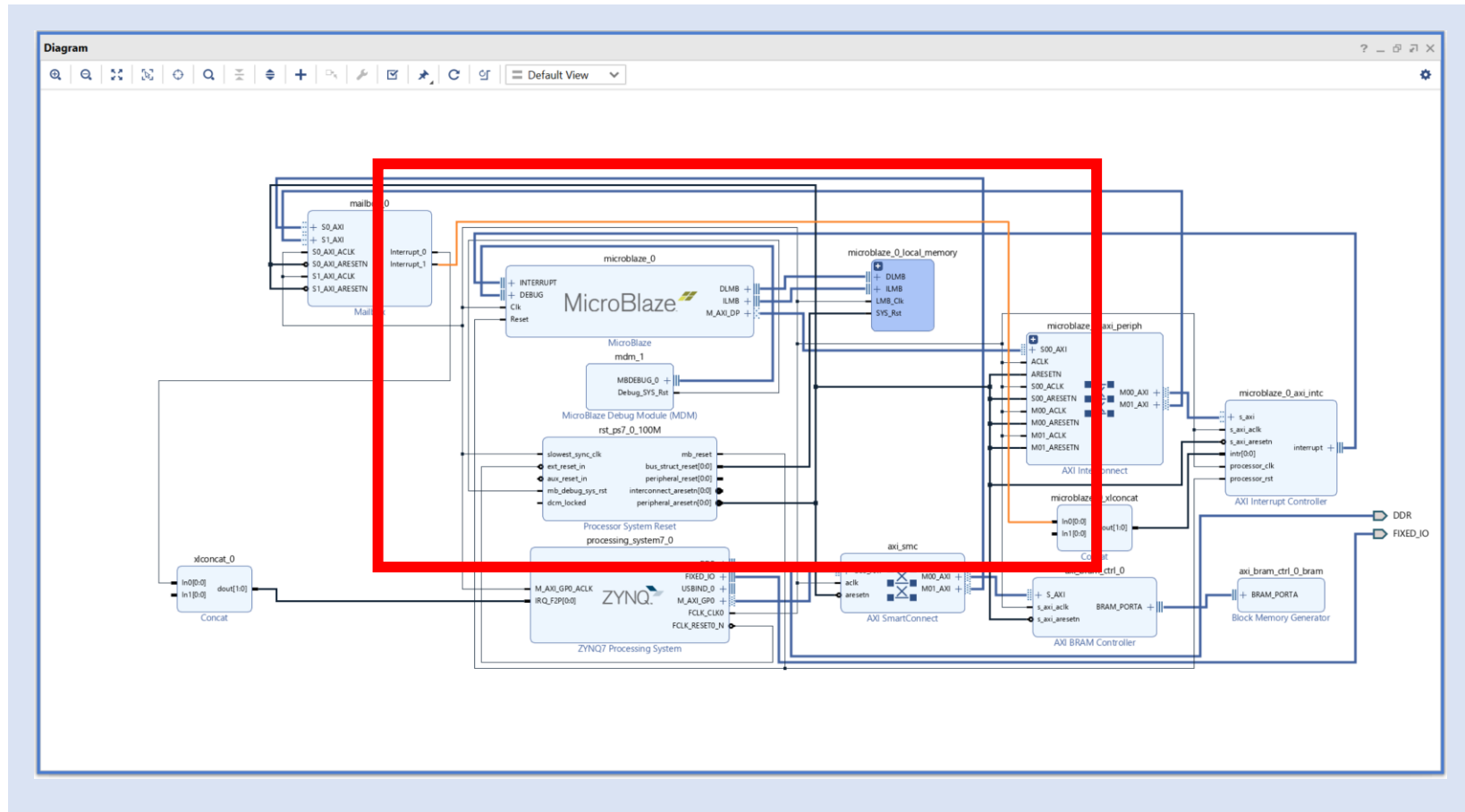
Lab 2: Understanding Vitis Project creation & Flow

Step 19 – Connect the interrupt one output from the mailbox to the concat block just added.



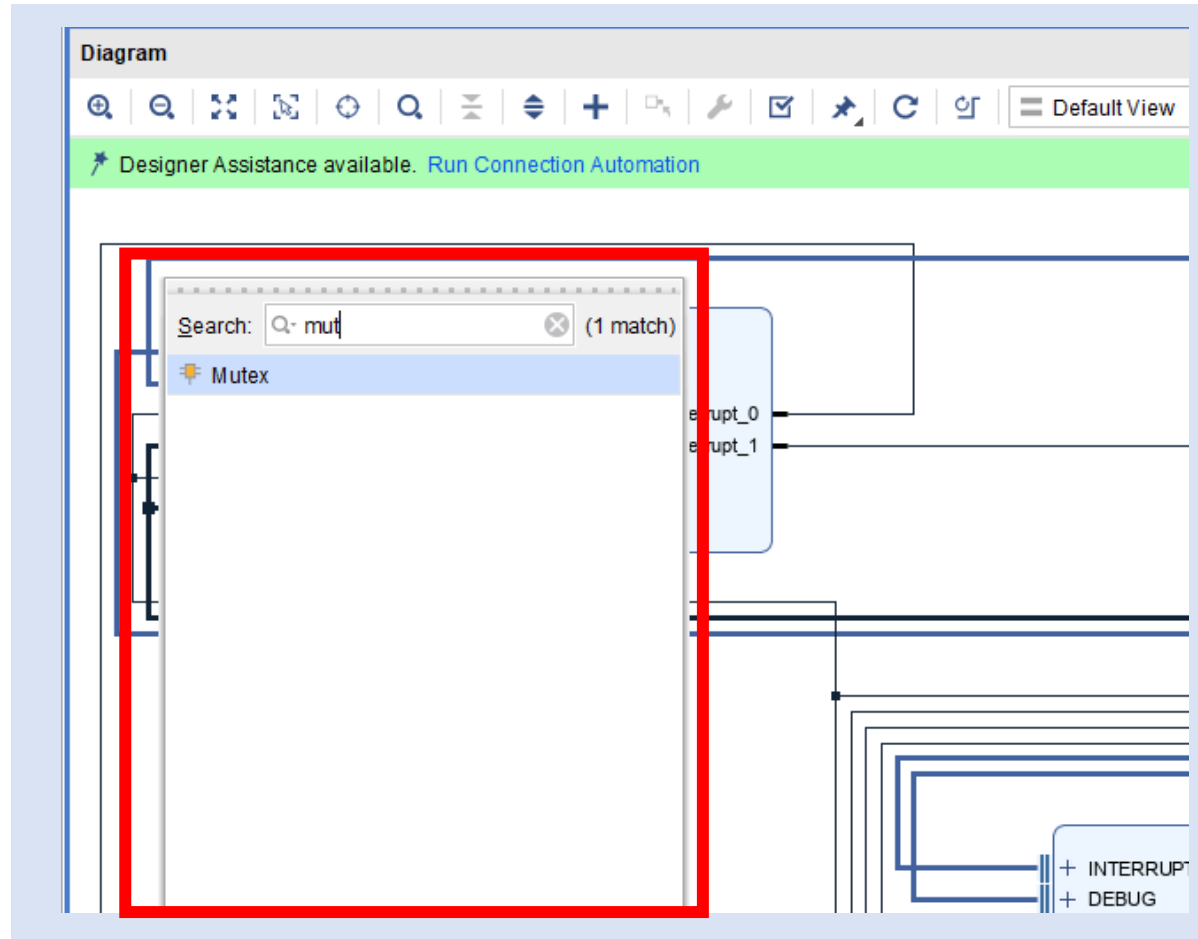
Lab 2: Understanding Vitis Project creation & Flow

Step 20 – Connect the interrupt two output from the mailbox to the concat block for the MicroBlaze.



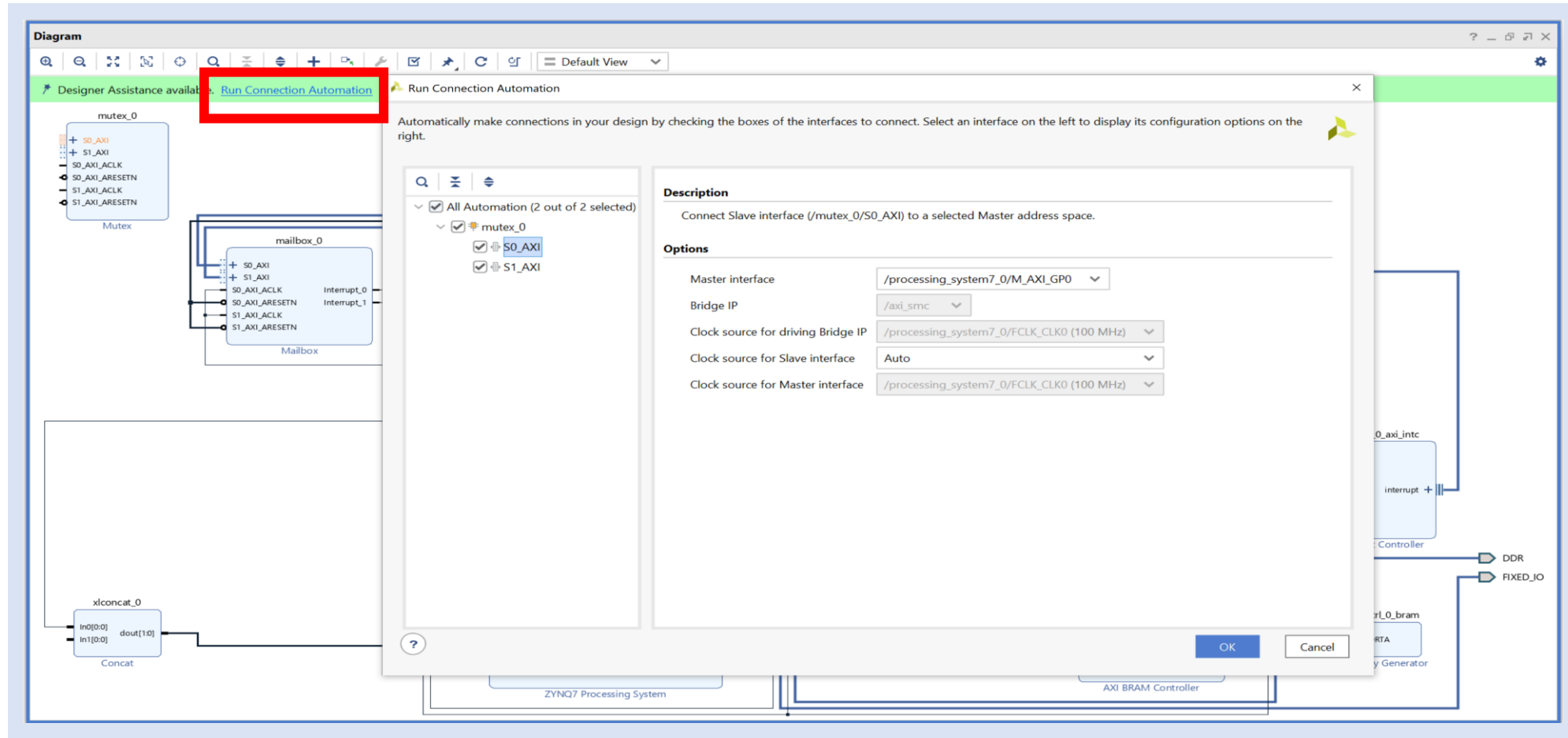
Lab 2: Understanding Vitis Project creation & Flow

Step 21 – Click on + and type Mutex, double click to add it in to IP integrator.



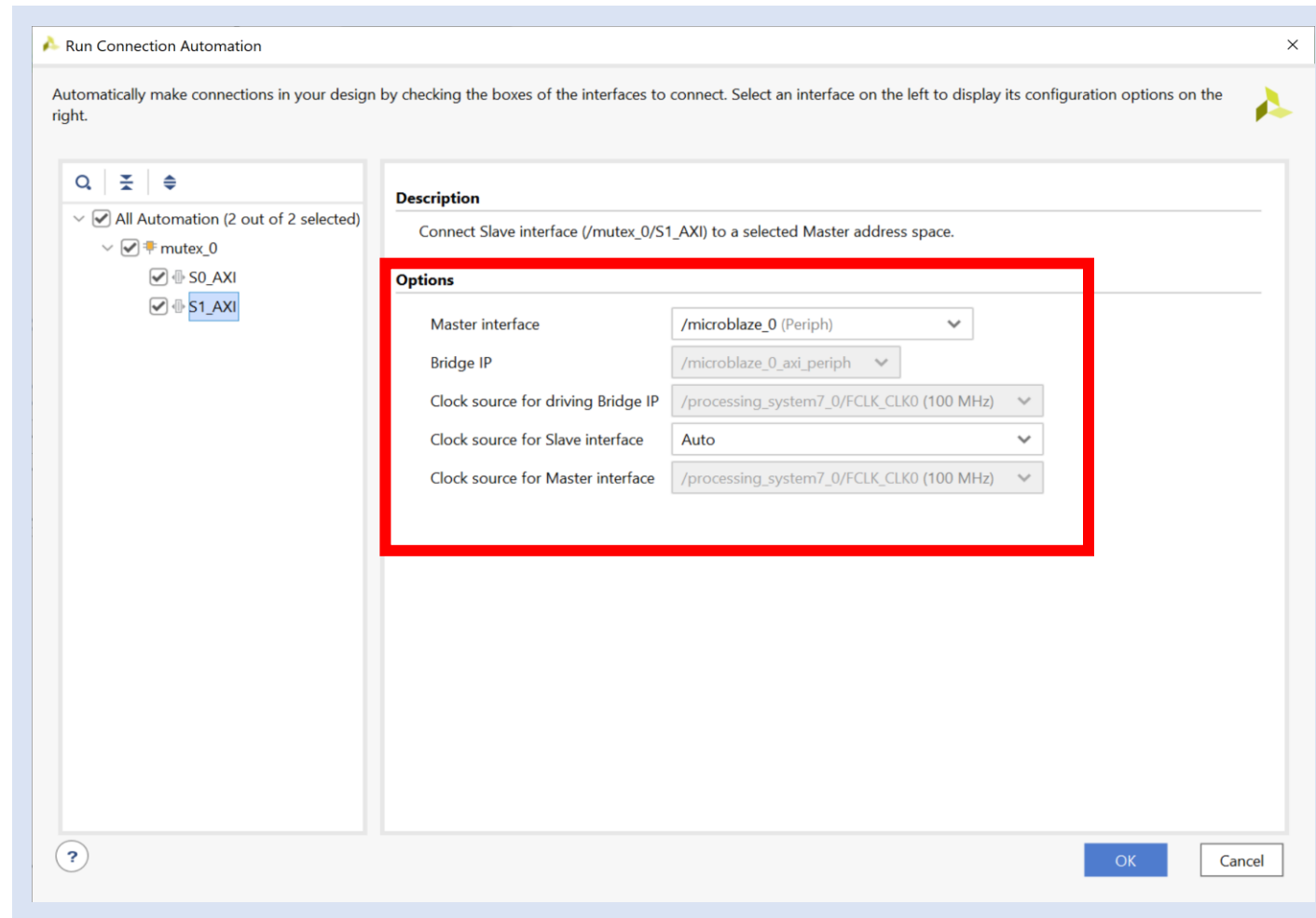
Lab 2: Understanding Vitis Project creation & Flow

Step 22 – Run the block automation, check the Mutex and for S0_AXI select the M_AXI_GPO.



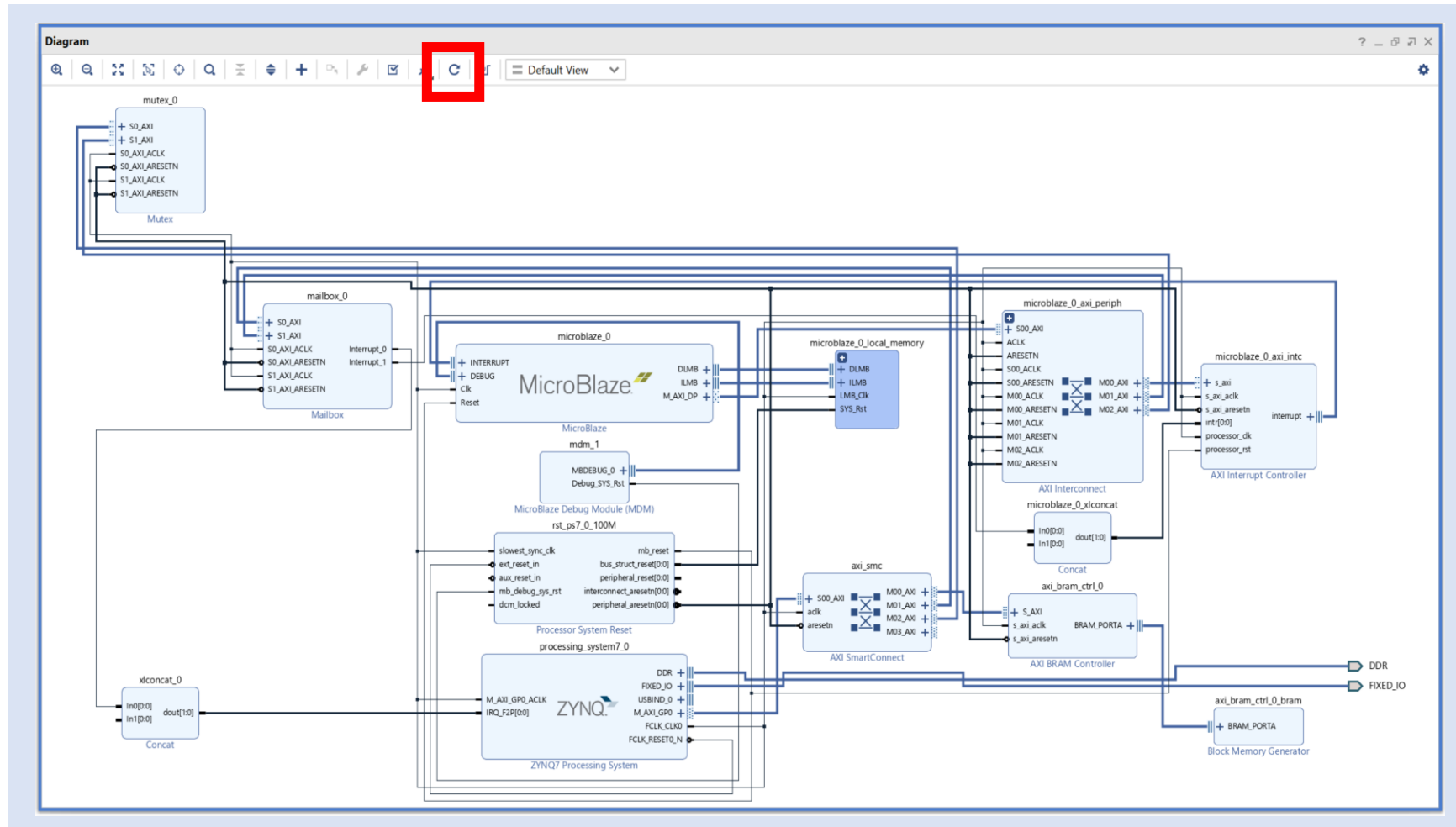
Lab 2: Understanding Vitis Project creation & Flow

Step 23 – Select MicroBlaze for the Master Interface on S1 AXI.



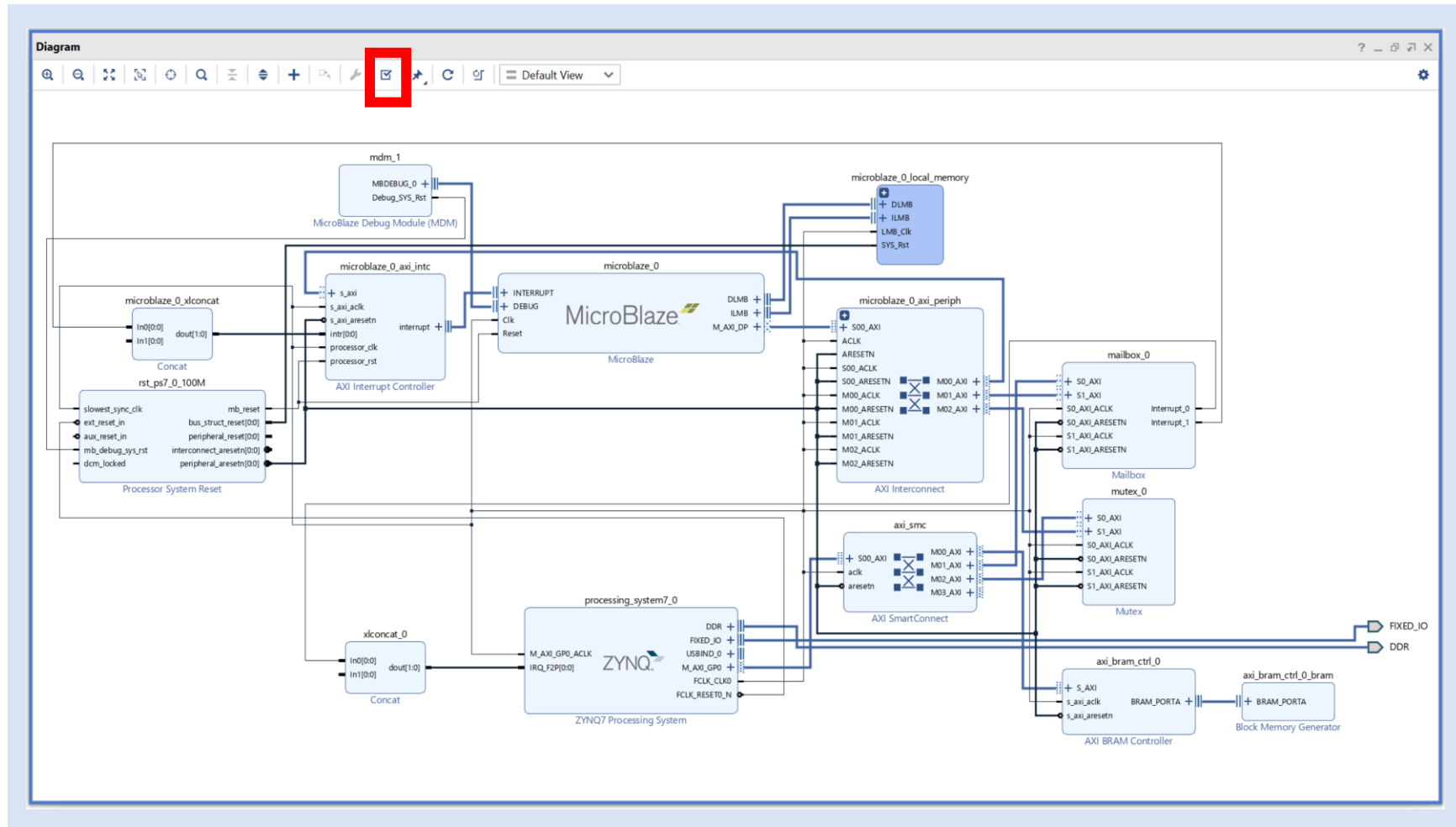
Lab 2: Understanding Vitis Project creation & Flow

Step 24 – Hit the regenerate layout button to create the layout as below



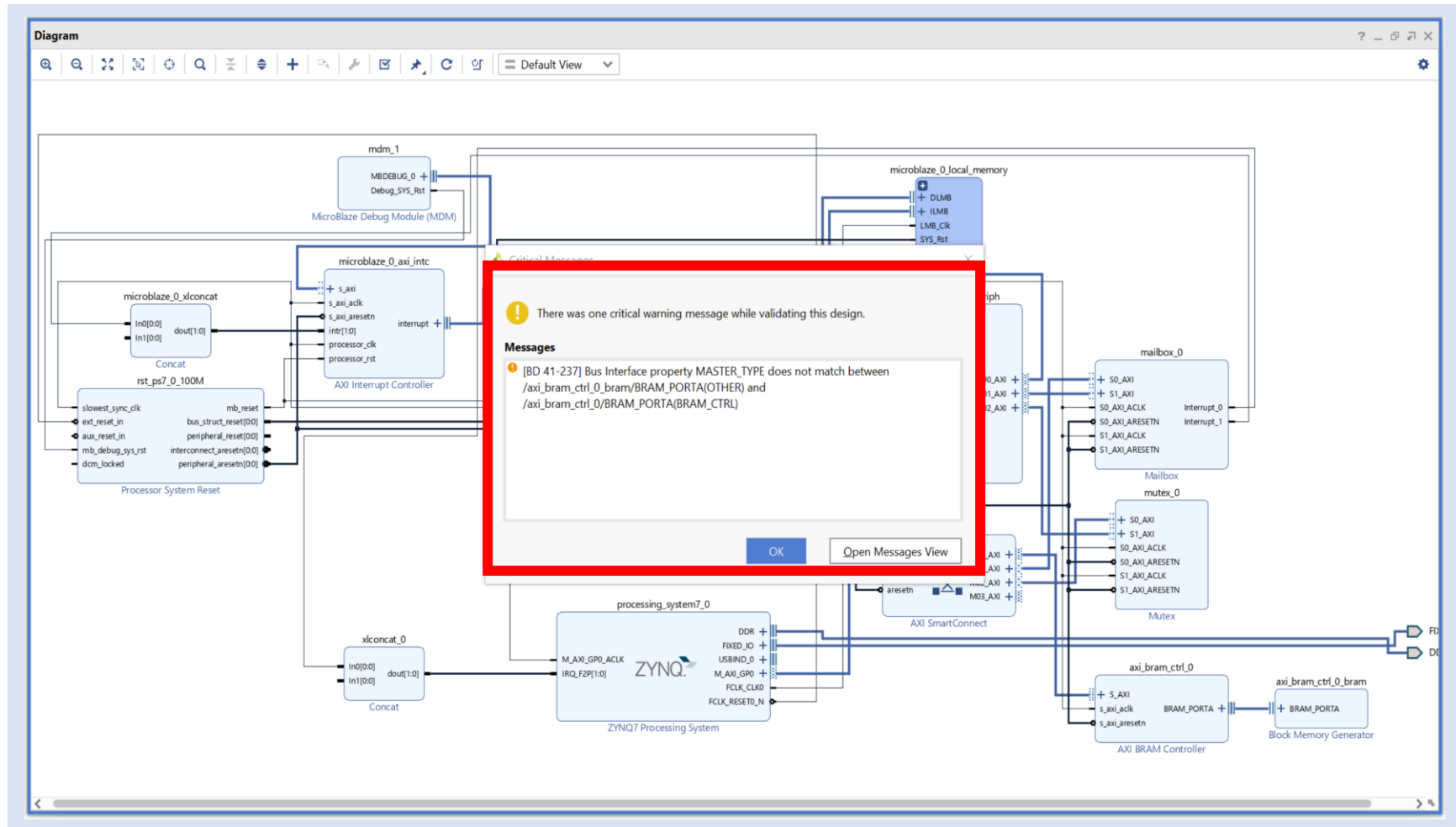
Lab 2: Understanding Vitis Project creation & Flow

Step 25 – Validate the design



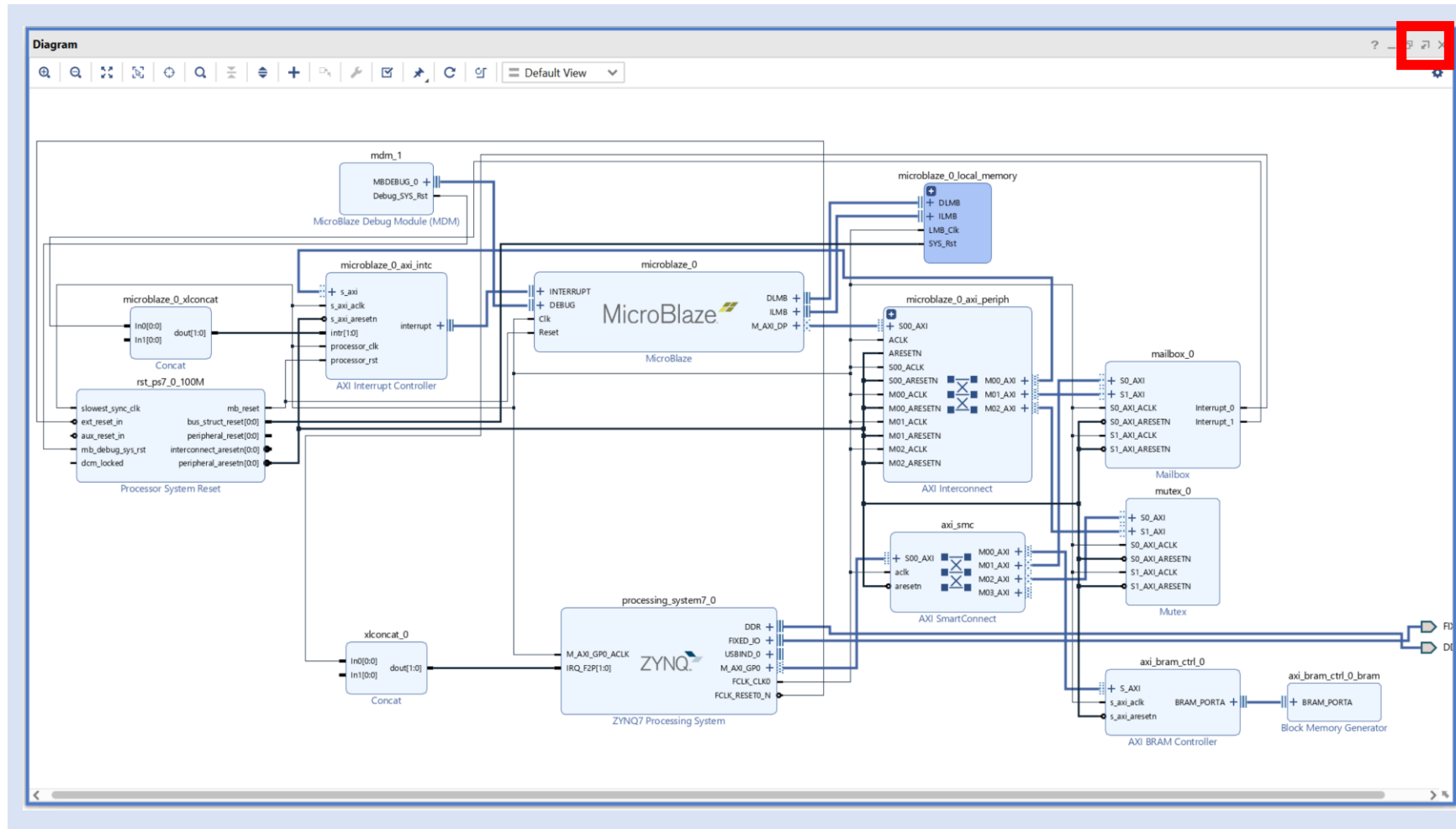
Lab 2: Understanding Vitis Project creation & Flow

Step 26 – The only warning messages should be those seen before



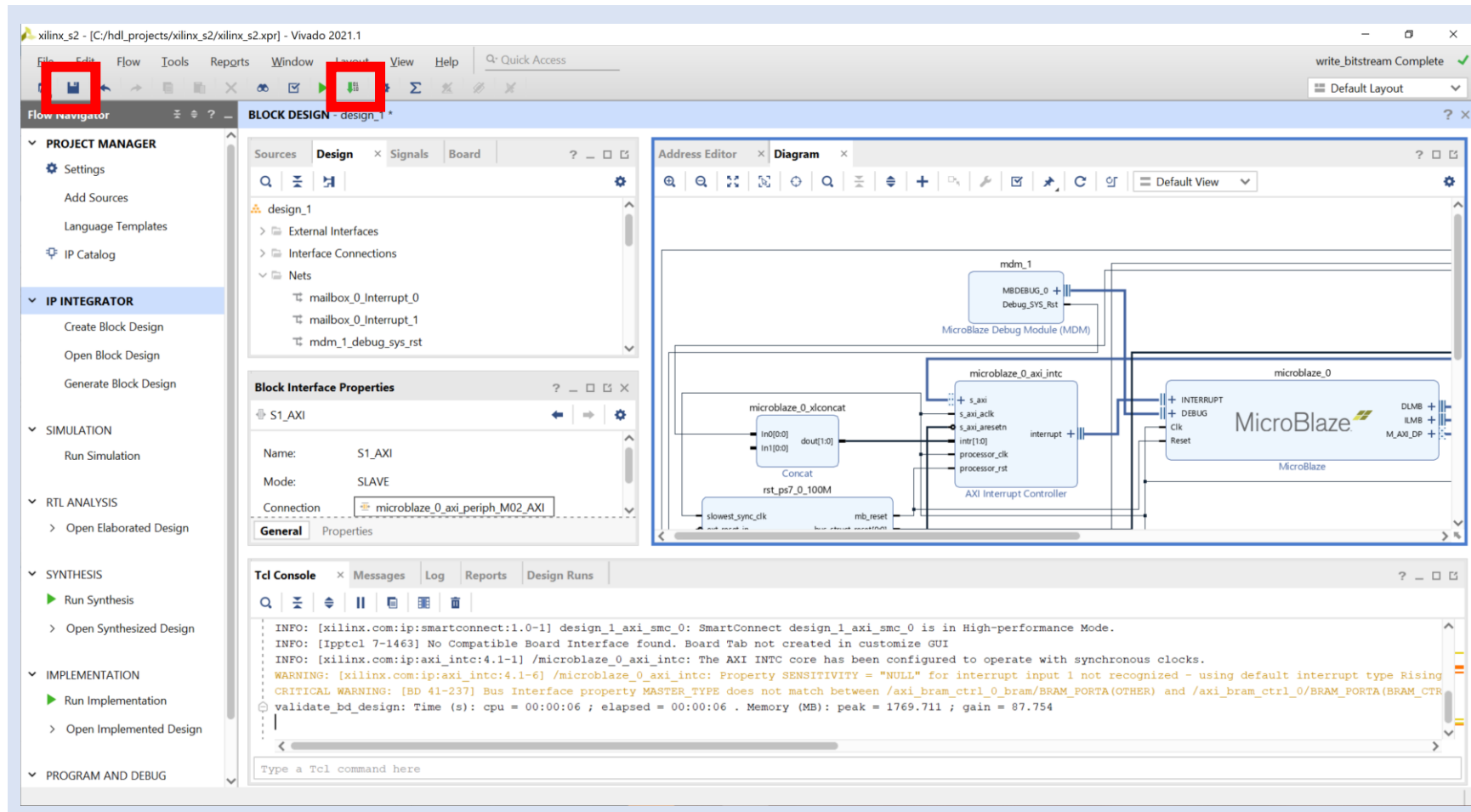
Lab 2: Understanding Vitis Project creation & Flow

Step 27 – Re Dock the IP Integrator window



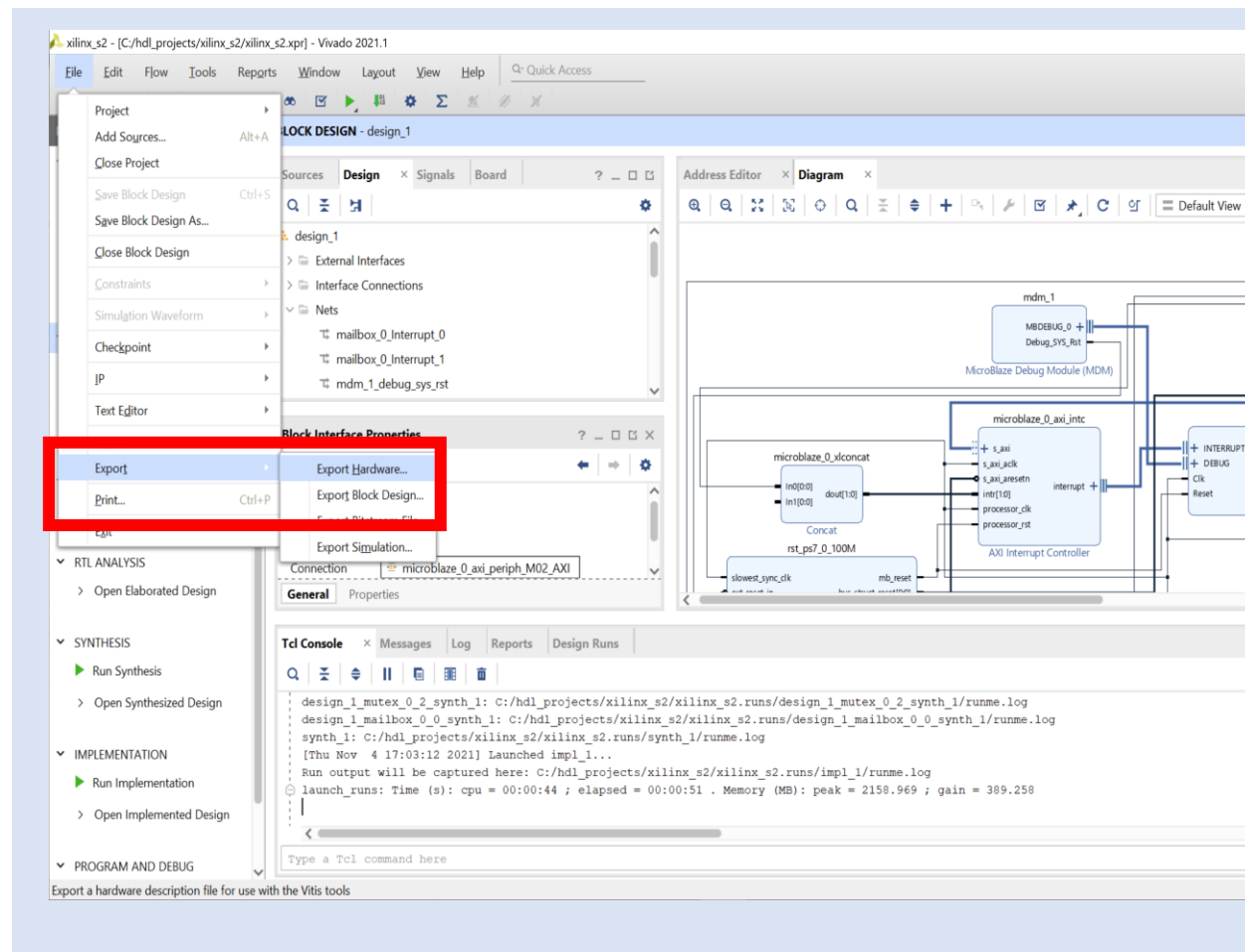
Lab 2: Understanding Vitis Project creation & Flow

Step 28 – Save the design and then re generate the bit stream



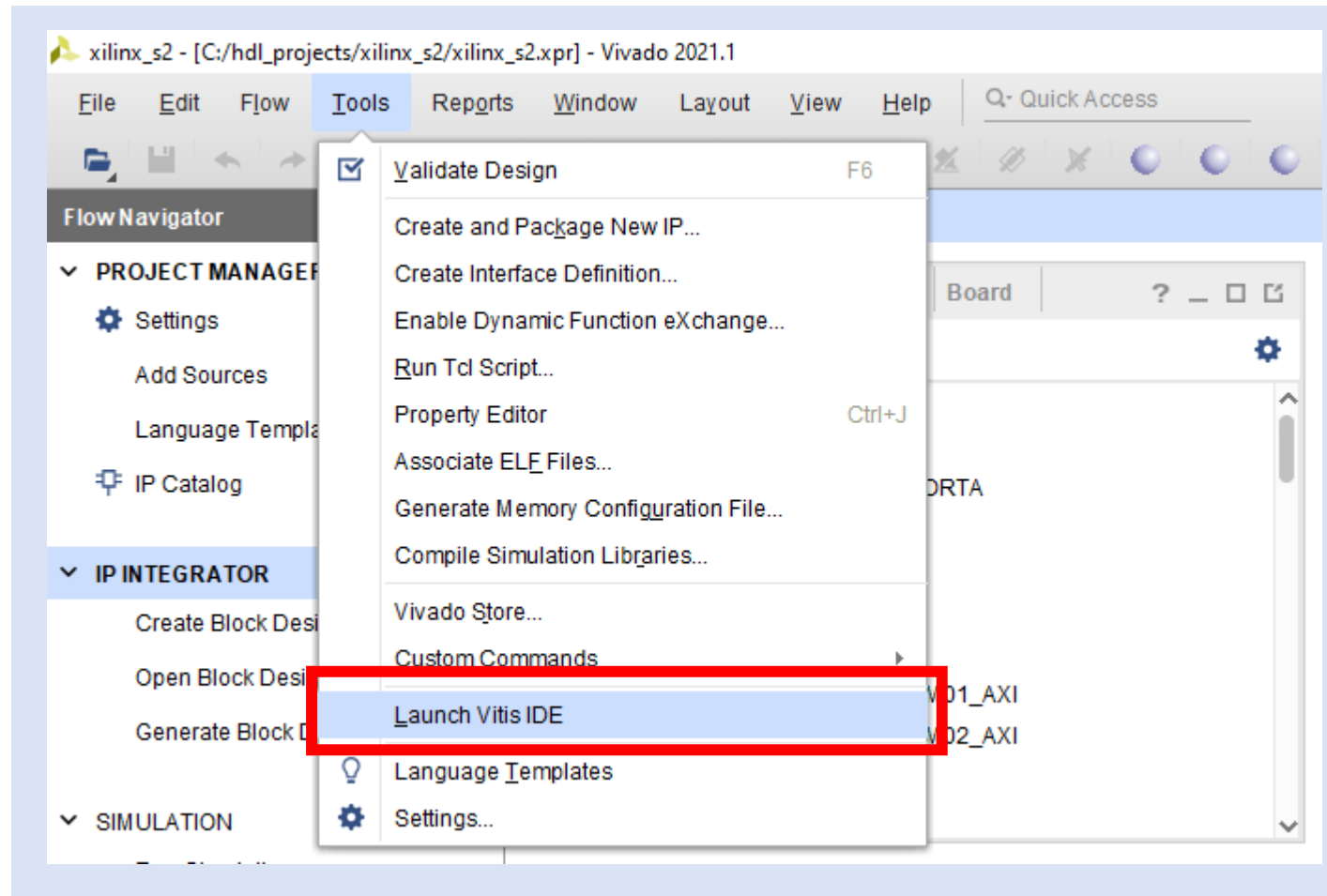
Lab 2: Understanding Vitis Project creation & Flow

Step 29 – Once the bitstream is complete, export the bit stream with the bit file to the project directory



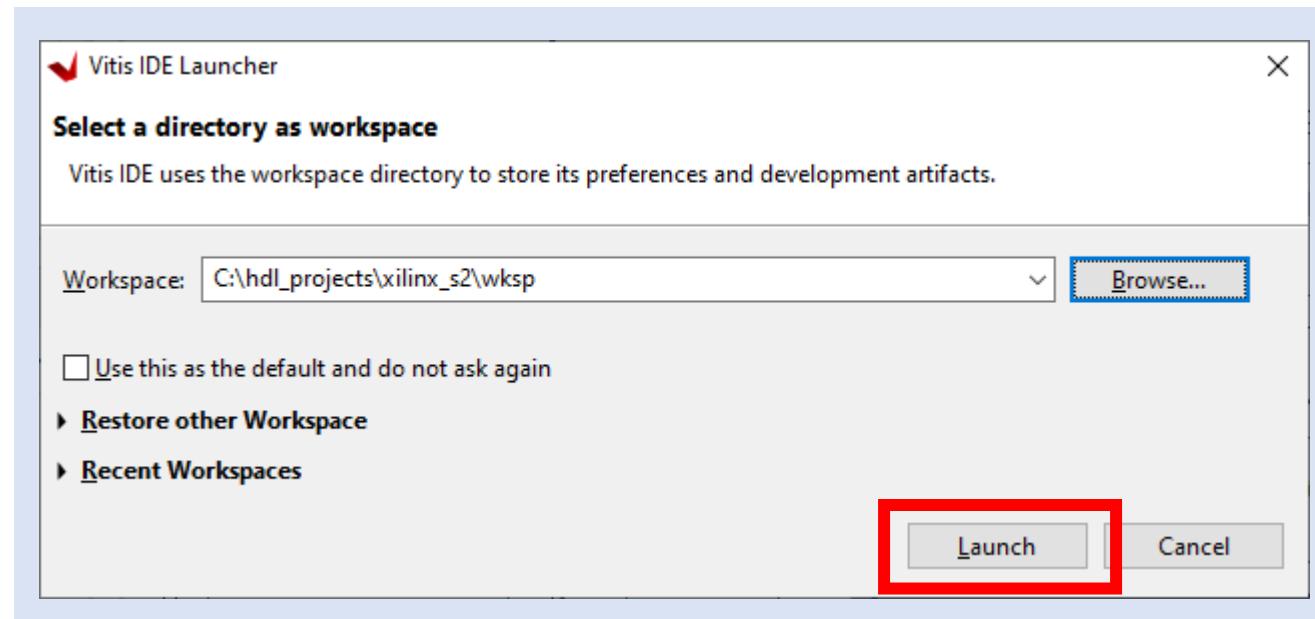
Lab 2: Understanding Vitis Project creation & Flow

Step 30 – From Tools select launch Vitis IDE



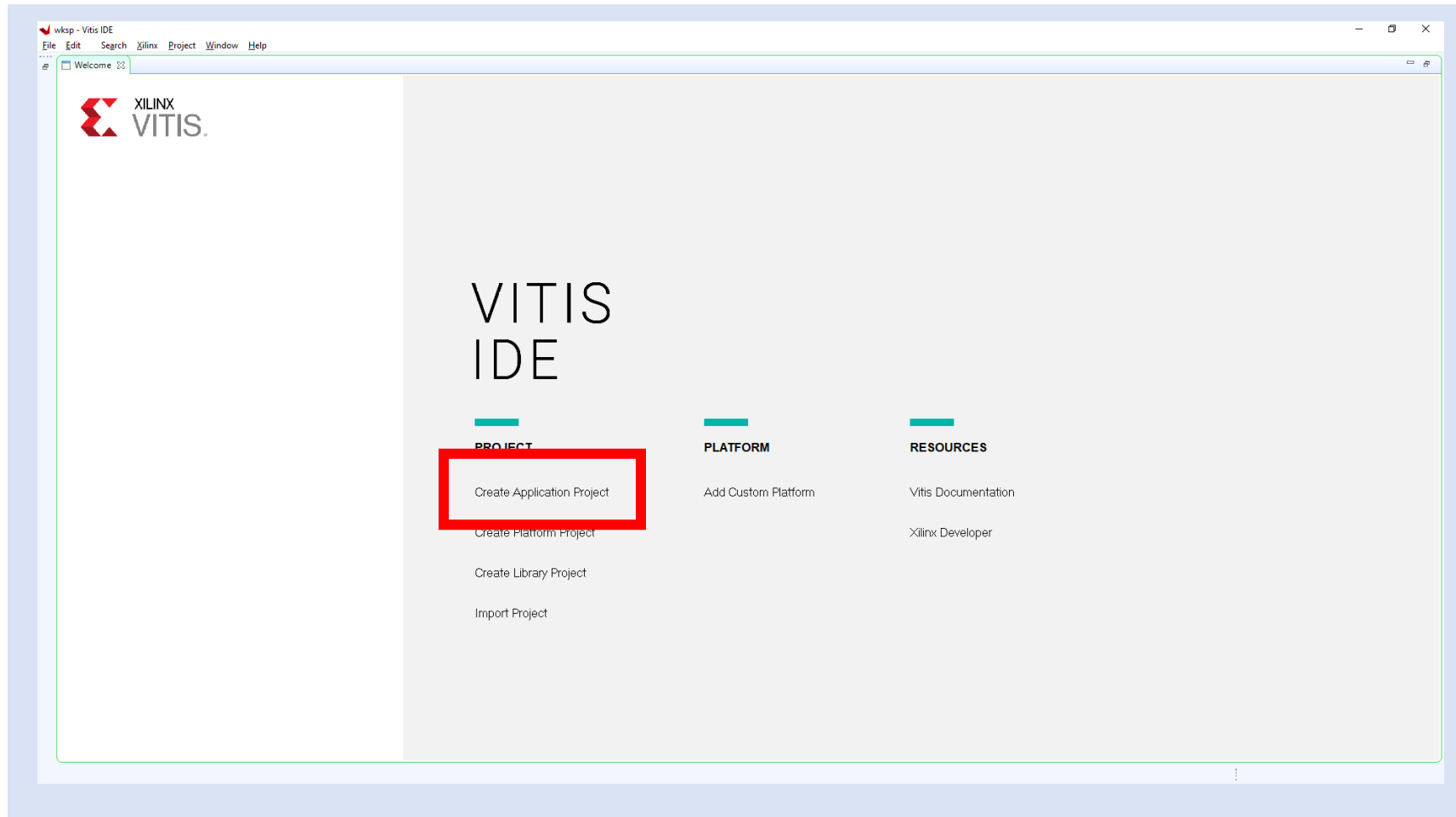
Lab 2: Understanding Vitis Project creation & Flow

Step 31 – From Tools select launch Vitis IDE, create a new workspace inside the project directory.



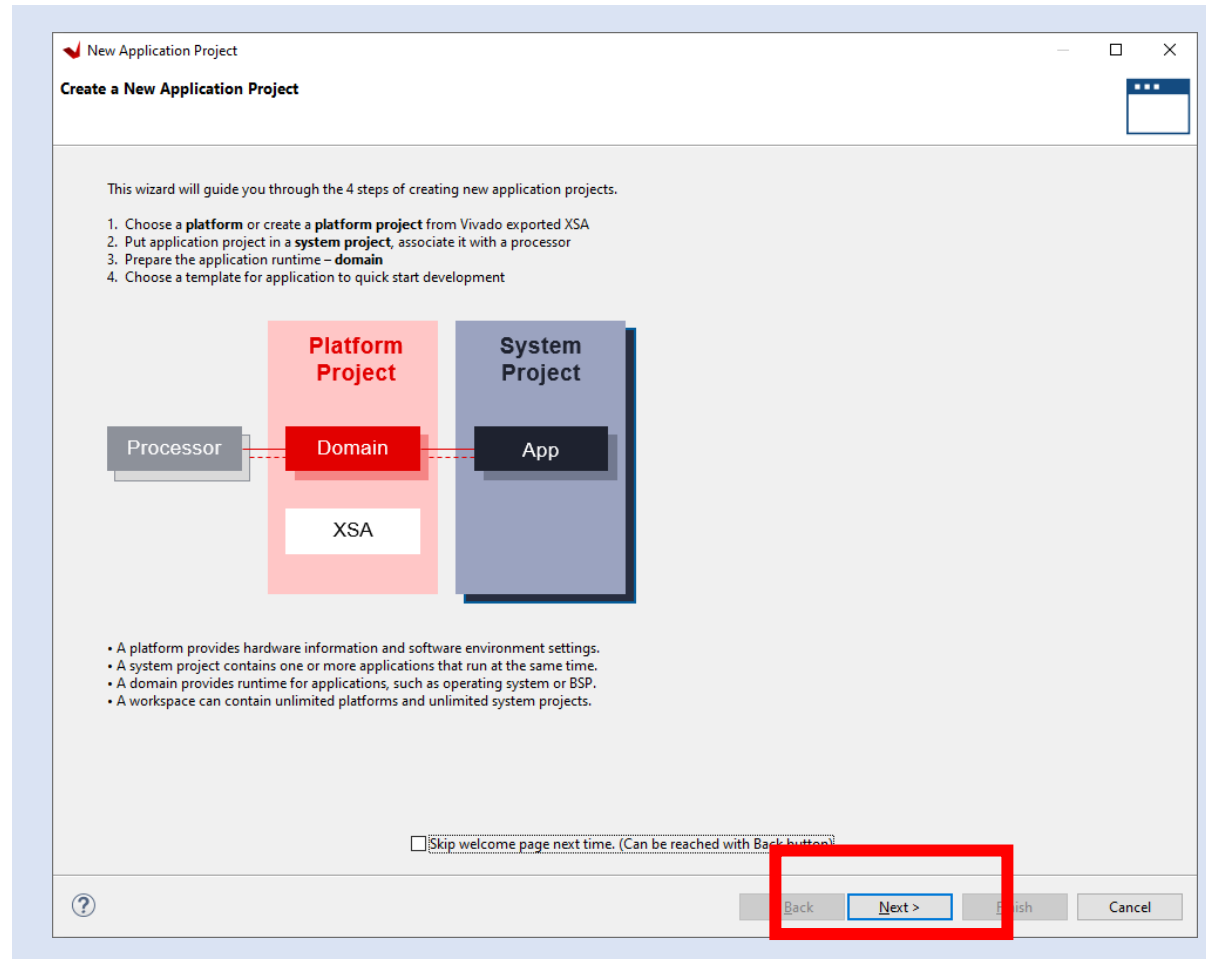
Lab 2: Understanding Vitis Project creation & Flow

Step 32 – Select Create Application Project



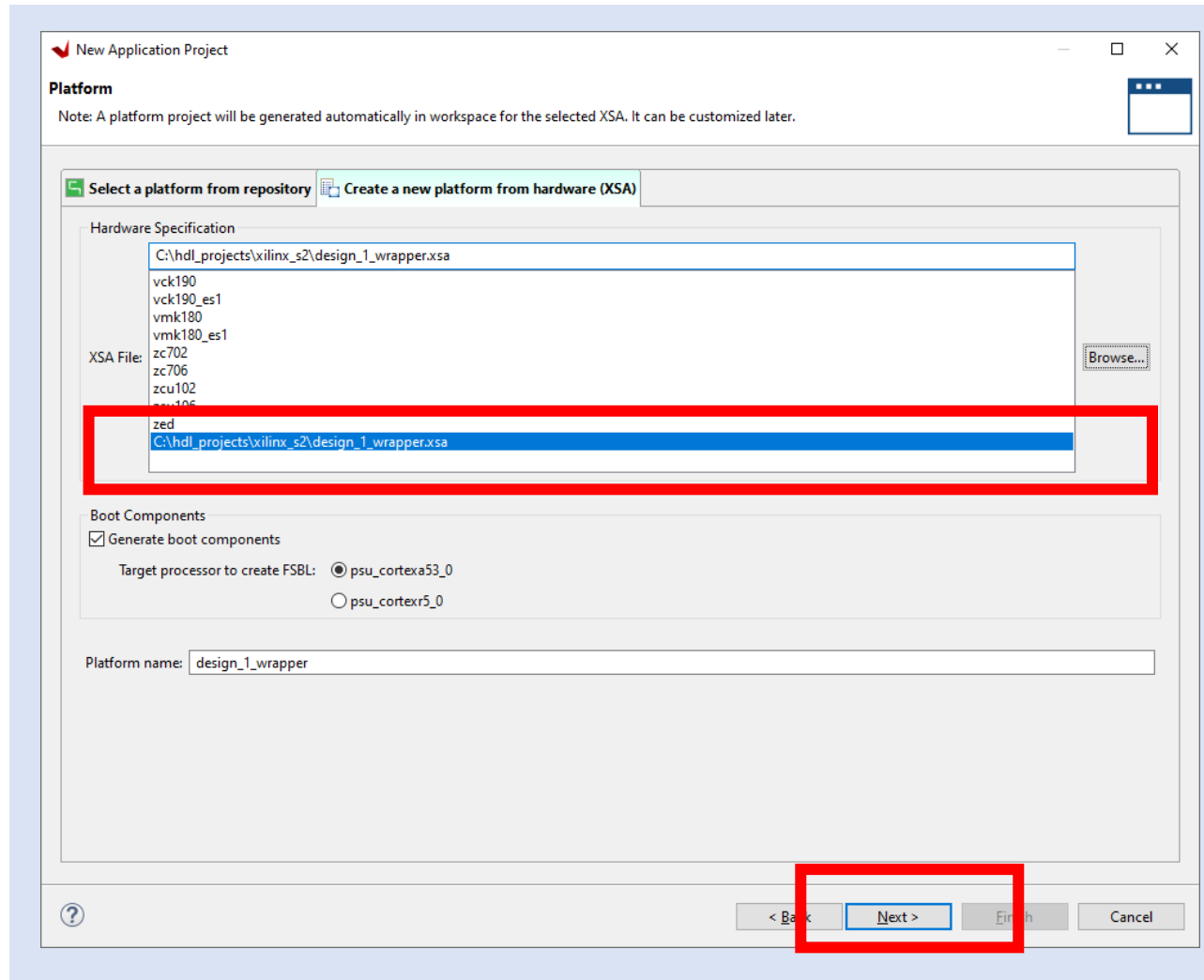
Lab 2: Understanding Vitis Project creation & Flow

Step 33 – Click next



Lab 2: Understanding Vitis Project creation & Flow

Step 34 – Click Select Create a new platform from hardware, browse to the new project directory – select the XSA previously exported



Lab 2: Understanding Vitis Project creation & Flow

Step 35 – Enter a project name and click next – This is for the MicroBlaze

New Application Project

Application Project Details
Specify the application project name and its system project properties

Application project name:

System Project
Create a new system project for the application or select an existing one from the workspace *i*

Select a system project
[+ Create new...](#)

System project details

System project name:

Target processor

Select target processor for the Application project.

Processor	Associated applications
microblaze_0	IPC_MB
ps7_cortexa9_0	
ps7_cortexa9_1	
ps7_cortexa9 SMP	

Show all processors in the hardware specification ☒ *i*

?

Lab 2: Understanding Vitis Project creation & Flow

Step 36 – Click on Next

New Application Project

Domain
Select a domain for your project or create a new domain

Select the domain that the application would link to or create a new domain
Note: New domain created by this wizard will have all the requirements of the application template selected in the next step

Select a domain
+ Create new...

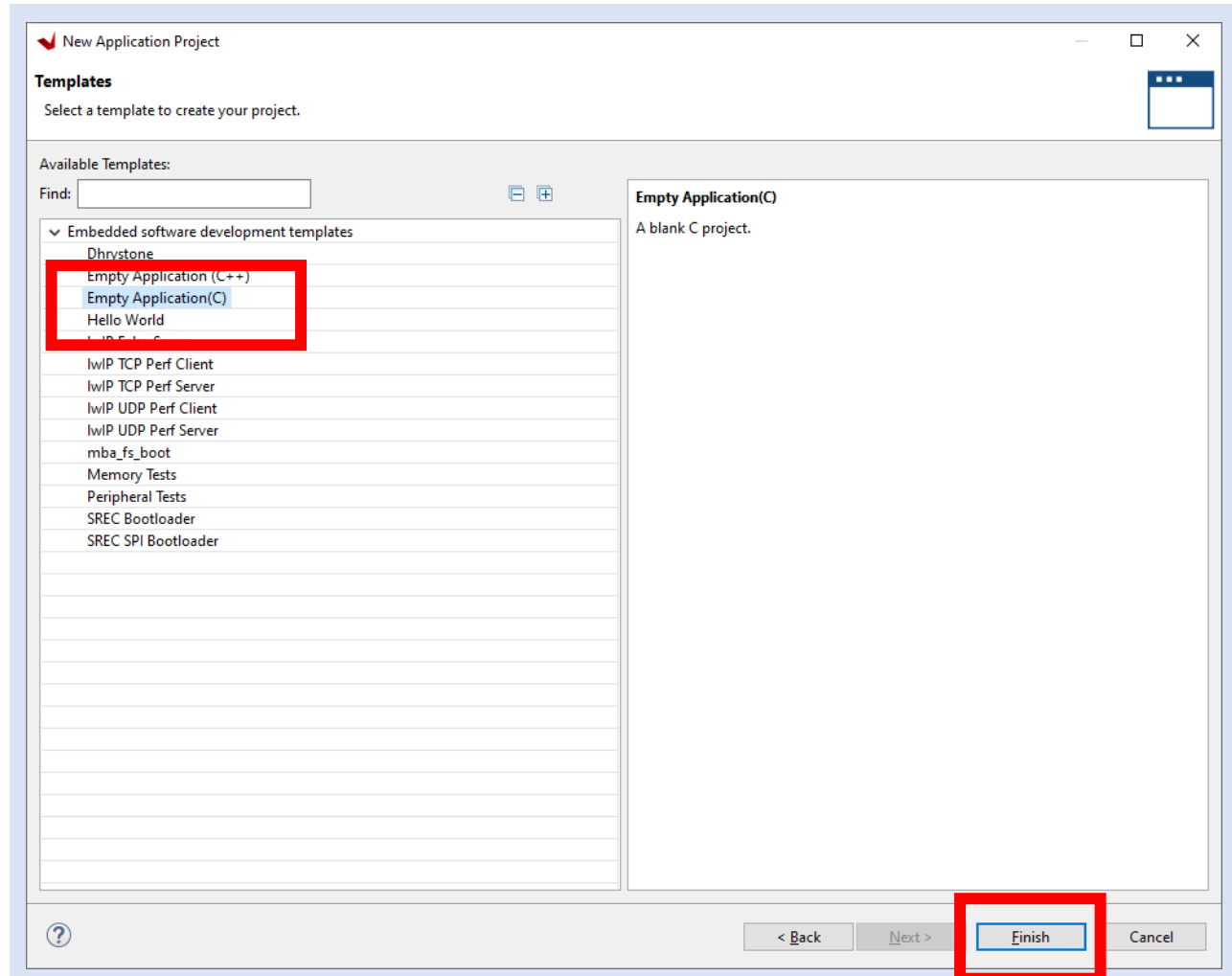
Domain details

Name: standalone_microblaze_0
Display Name: standalone_microblaze_0
Operating System: standalone
Processor: microblaze_0
Architecture: 32-bit

< Back Next > Finish Cancel

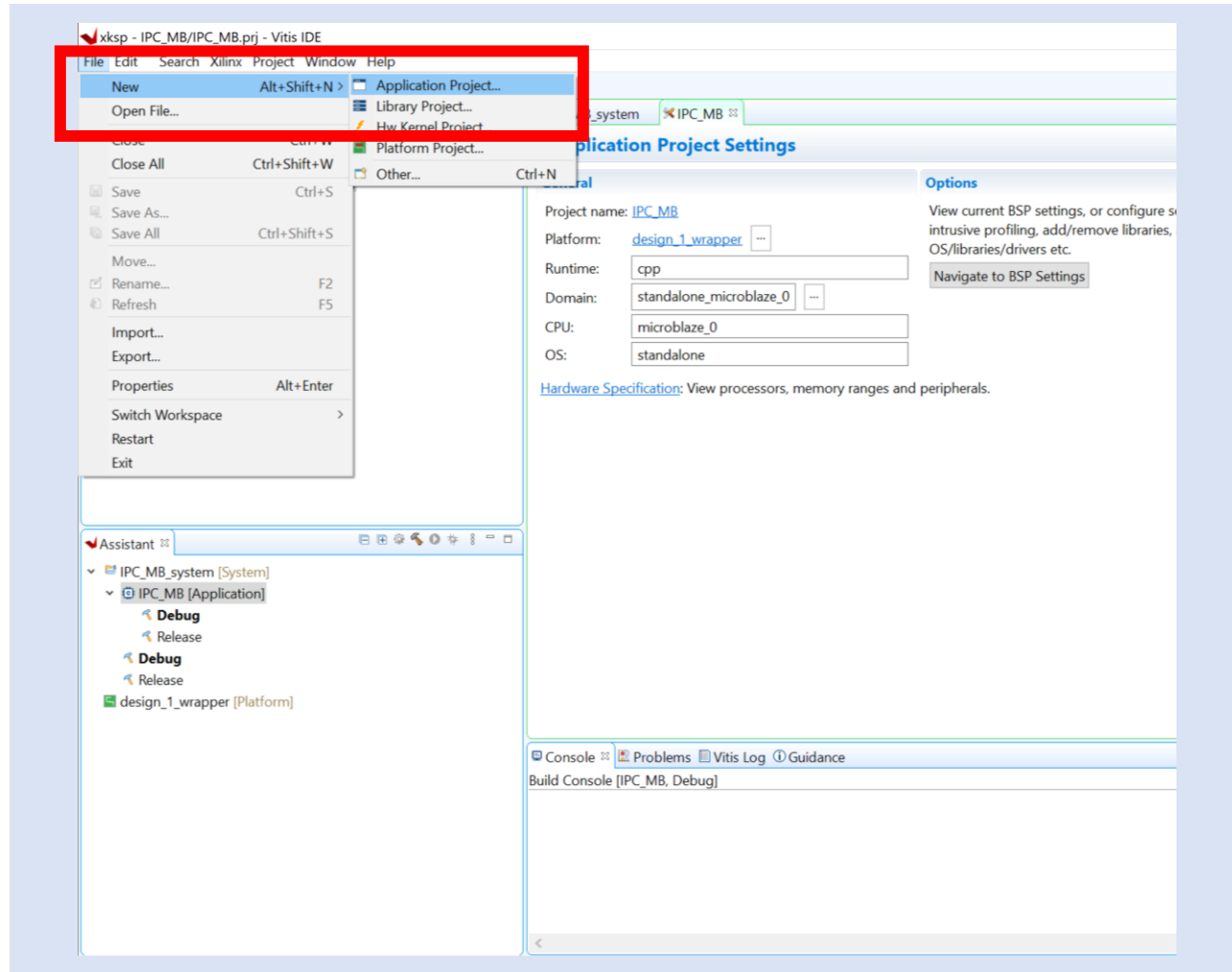
Lab 2: Understanding Vitis Project creation & Flow

Step 37 – Select Empty Application, click finish and project will be created



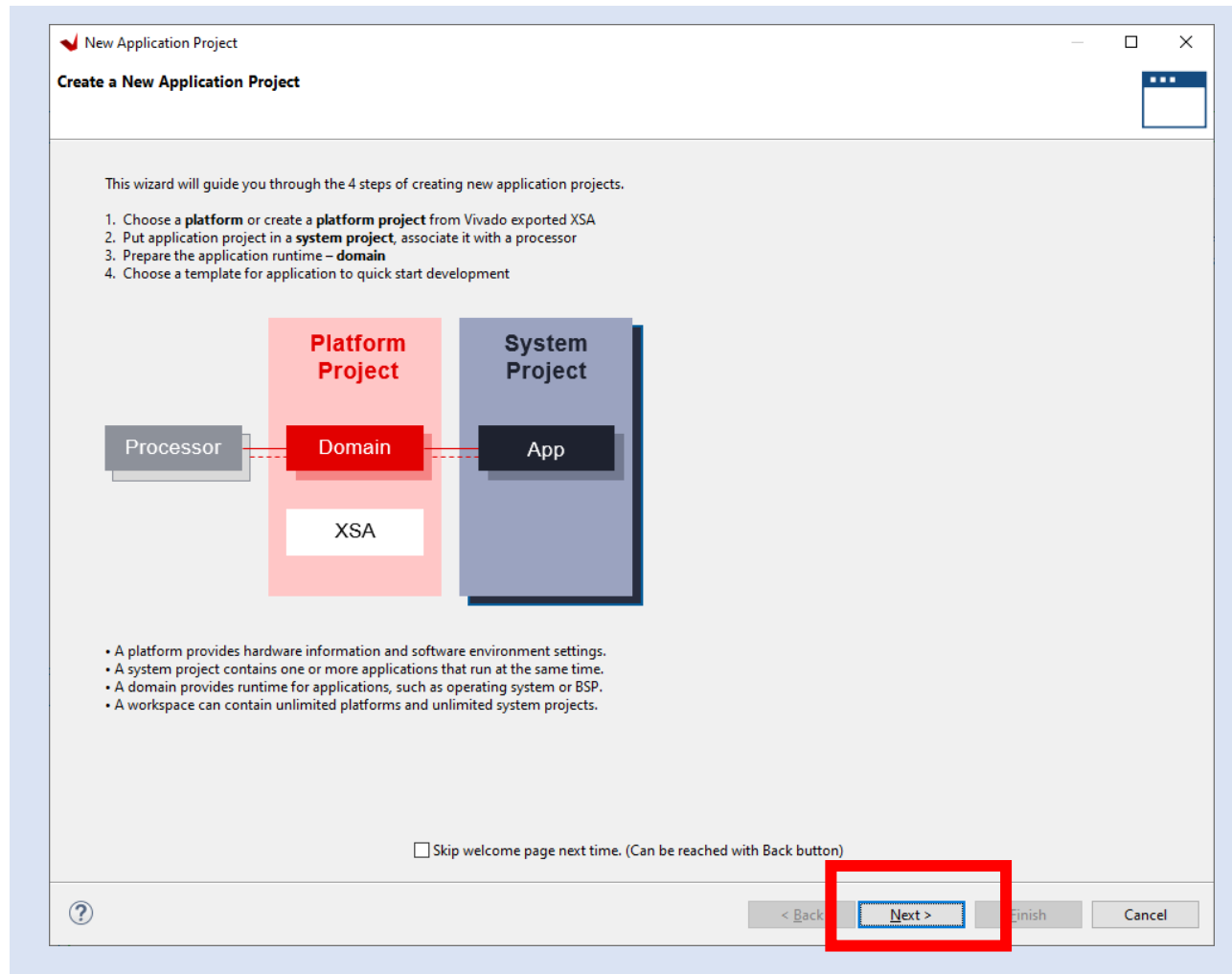
Lab 2: Understanding Vitis Project creation & Flow

Step 38 – From File select new application project



Lab 2: Understanding Vitis Project creation & Flow

Step 39 – Click next



Lab 2: Understanding Vitis Project creation & Flow

Step 40 – Select the current platform and click next

New Application Project

Platform

Choose a platform for your project. You can also create an application from XSA through the 'Create a new platform from hardware (XSA)' tab.

Select a platform from repository **Create a new platform from hardware (XSA)**

Find:

+ Add ⚙ Manage

Name	Board	Flow	Vendor	Path
design_1_wrapper [custom arty-z7...		Embedded SW D...	xilinx	C:\hdl_projects\xilinx_s2\vksp\design_1_wrapper\export\design_1_wrapper\design_1_wrapper.xp

Platform Info

General Info

Name: design_1_wrapper

Part: xc7z020clg400-1

Family: zynq

Description: design_1_wrapper

Acceleration Resources

The selected platform does not have application acceleration capabilities

Domain Details

Domains

Domain name	Details
standalone_microblaze_0	CPU: microblaze_0C

< Back **Next >** Finish Cancel

Lab 2: Understanding Vitis Project creation & Flow

Step 41 – Select the IPC System, ensure show all processors is selected, select processor ps7_cortexa9_0 and enter a name IPC_ARM. Then click next

New Application Project

Application Project Details

Specify the application project name and its system project properties

Application project name:

System Project

Create a new system project for the application or select an existing one from the workspace

Select a system project

IPC_MB_system

Create new...

System project details

System project name:

Select target processor for the Application project.

Processor	Associated applications
microblaze_0	IPC_MB
ps7_cortexa9_0	IPC_ARM
ps7_cortexa9_1	
ps7_cortexa9 SMP	

Show all processors in the hardware specification ☒

< Back Next > Finish Cancel

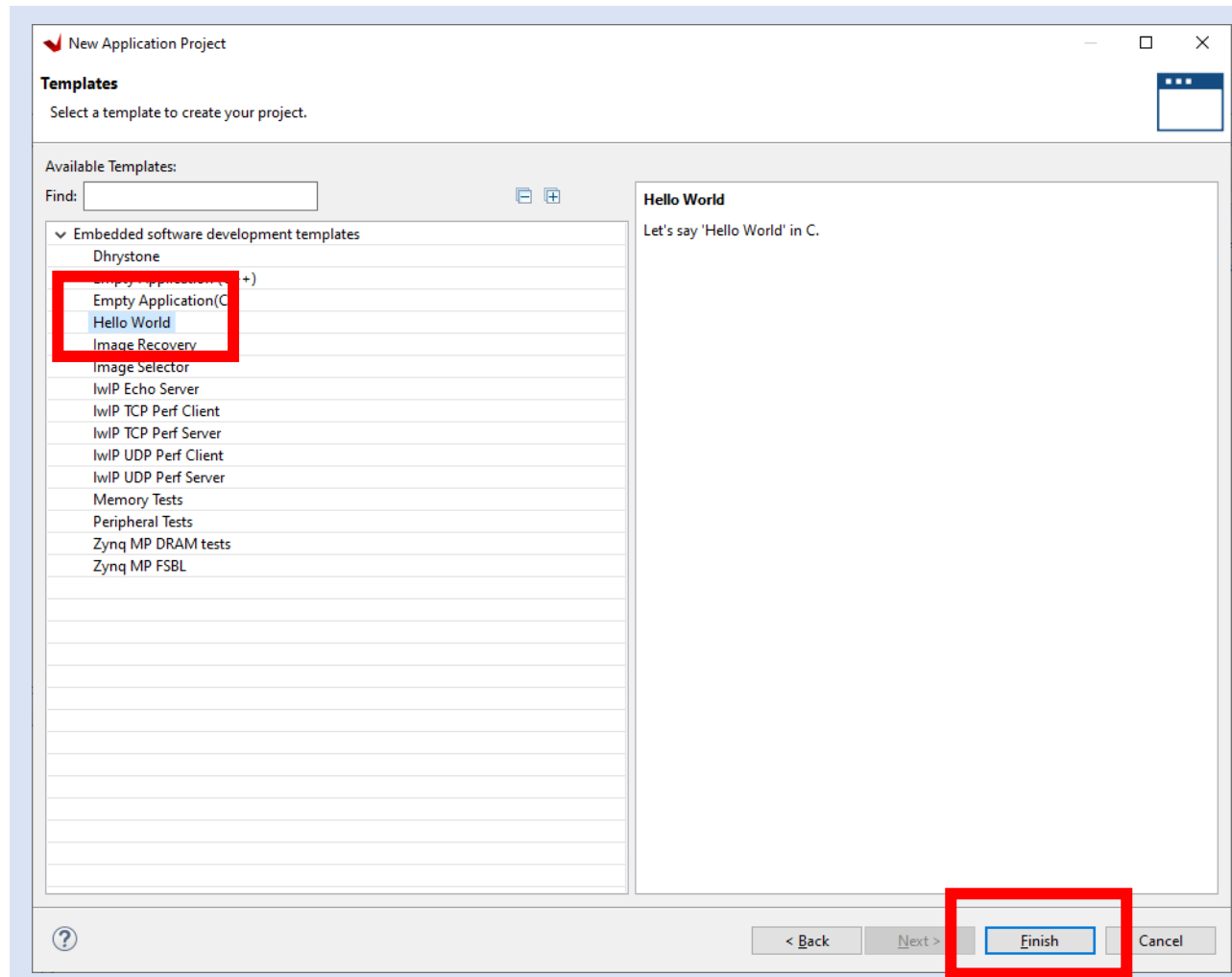
Lab 2: Understanding Vitis Project creation & Flow

Step 42 – Click on Next

The screenshot shows the 'New Application Project' wizard in Vitis. The title bar says 'New Application Project'. The main heading is 'Domain'. Below it, the text says 'Select a domain for your project or create a new domain'. There is a sub-section titled 'Select the domain that the application would link to or create a new domain' with a note: 'Note: New domain created by this wizard will have all the requirements of the application template selected in the next step'. On the left, there is a list box with 'Select a domain' and a '+ Create new...' button. On the right, under 'Domain details', there are fields for Name, Display Name, Operating System, Processor, and Architecture. The values are: Name: standalone_ps7_cortexa9_0, Display Name: standalone_ps7_cortexa9_0, Operating System: standalone, Processor: ps7_cortexa9_0, and Architecture: 32-bit. At the bottom, there are buttons for '< Back', 'Next >', 'Finish', and 'Cancel'. The 'Next >' button is highlighted with a red box.

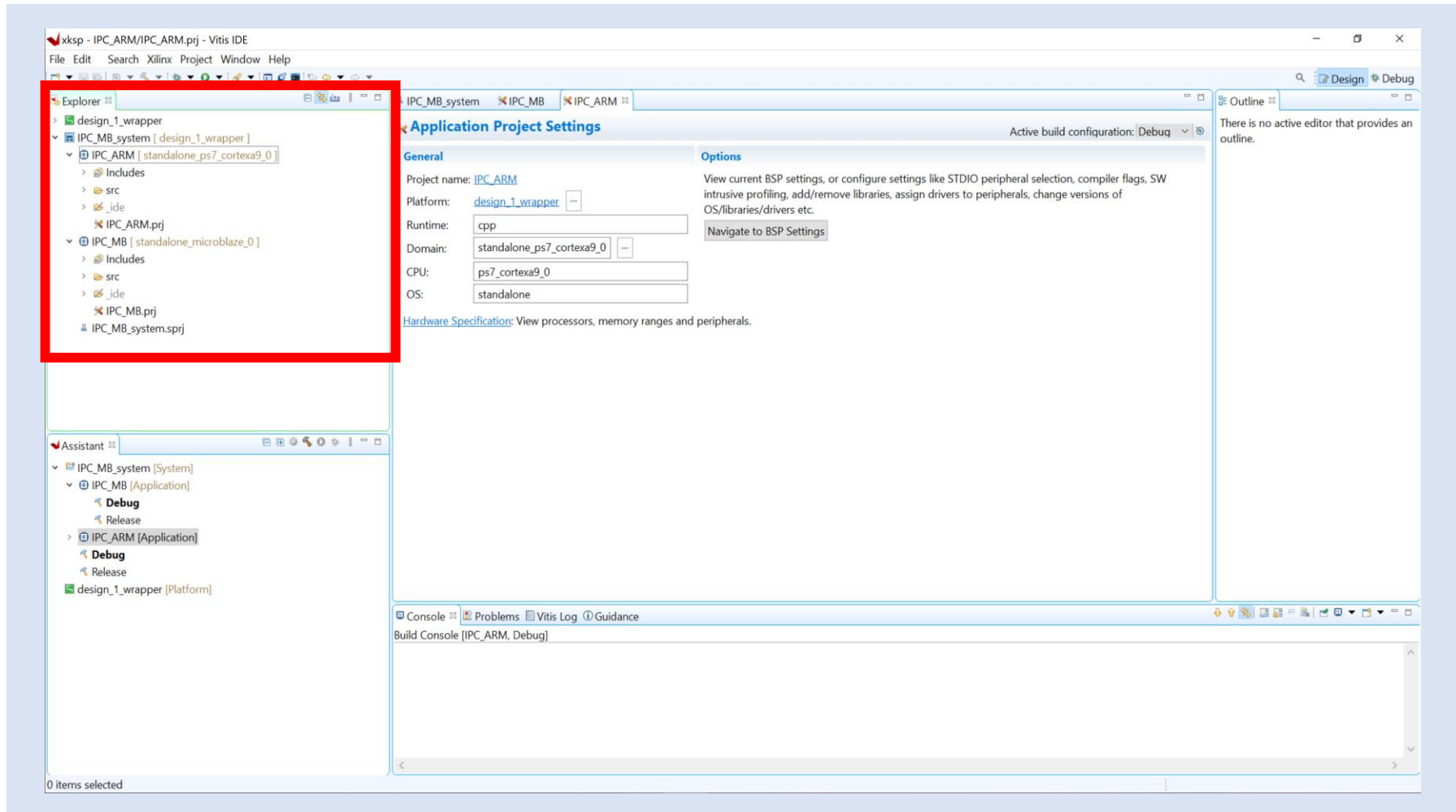
Lab 2: Understanding Vitis Project creation & Flow

Step 43 – Select the Hello World template and click finish



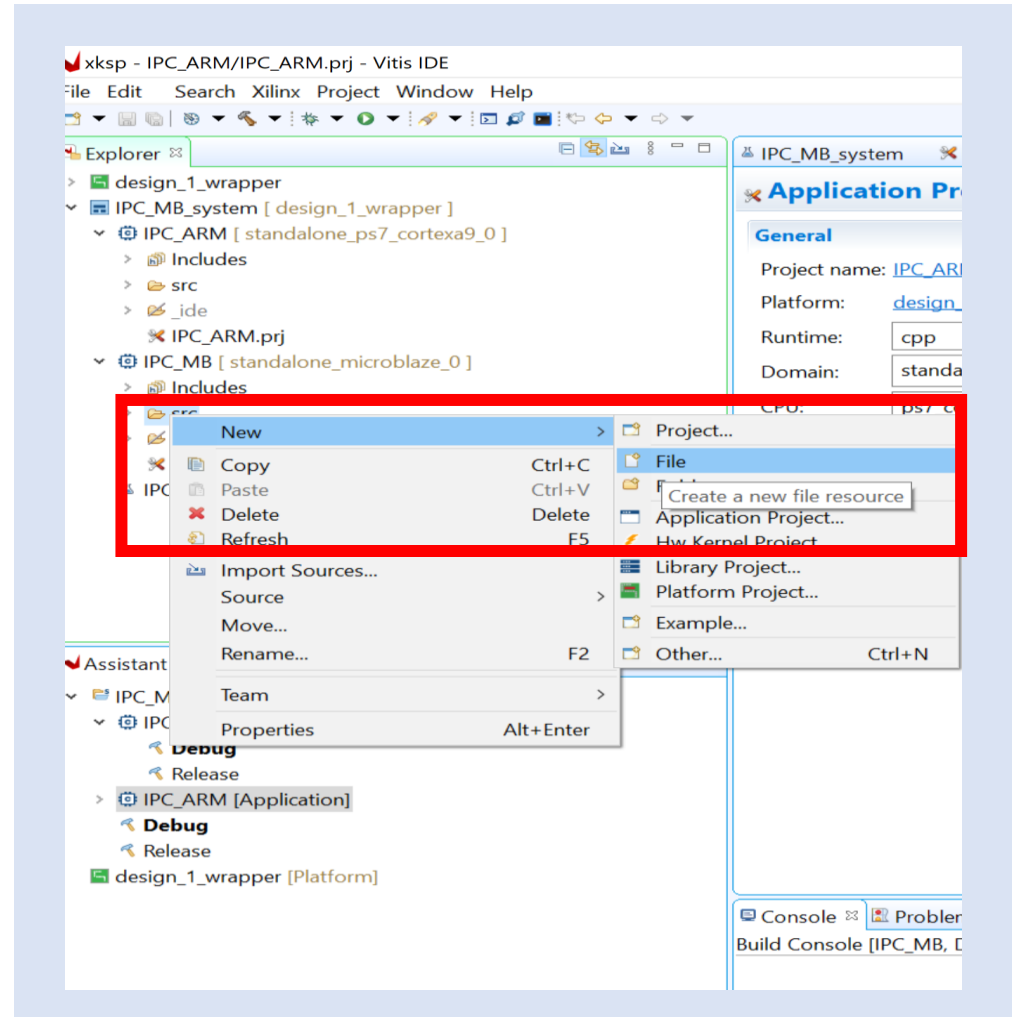
Lab 2: Understanding Vitis Project creation & Flow

Step 44 – You should now see two application projects along with the platform



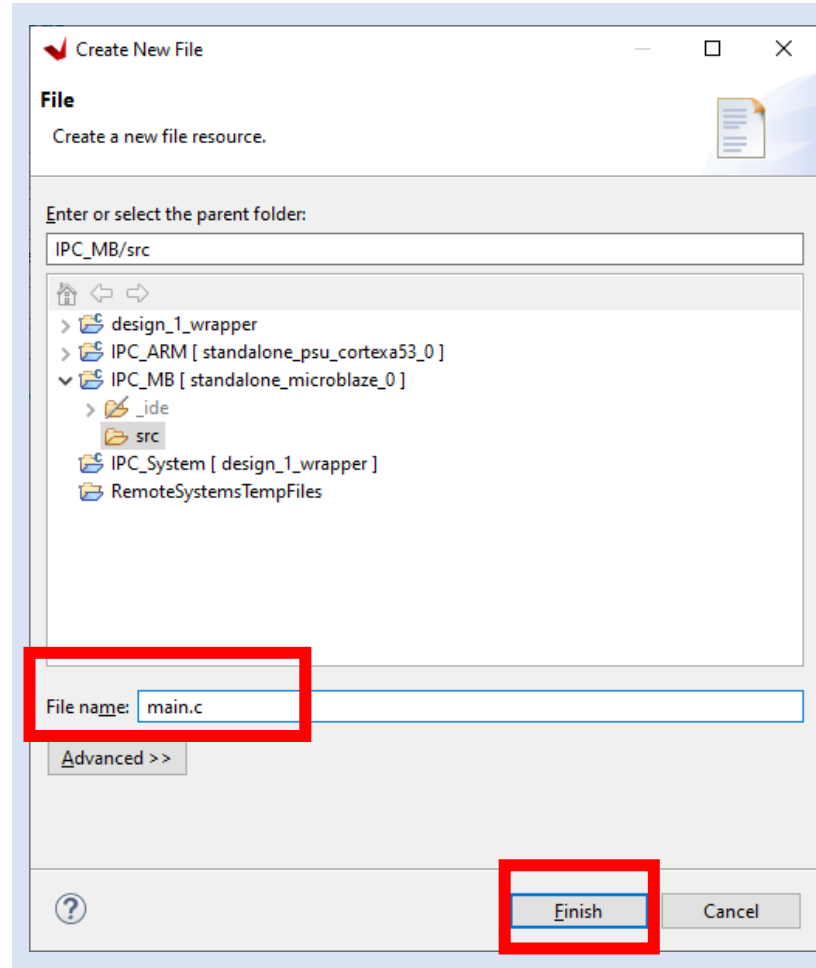
Lab 2: Understanding Vitis Project creation & Flow

Step 45 – Click on the MB project SRC directory, right click and select new File.



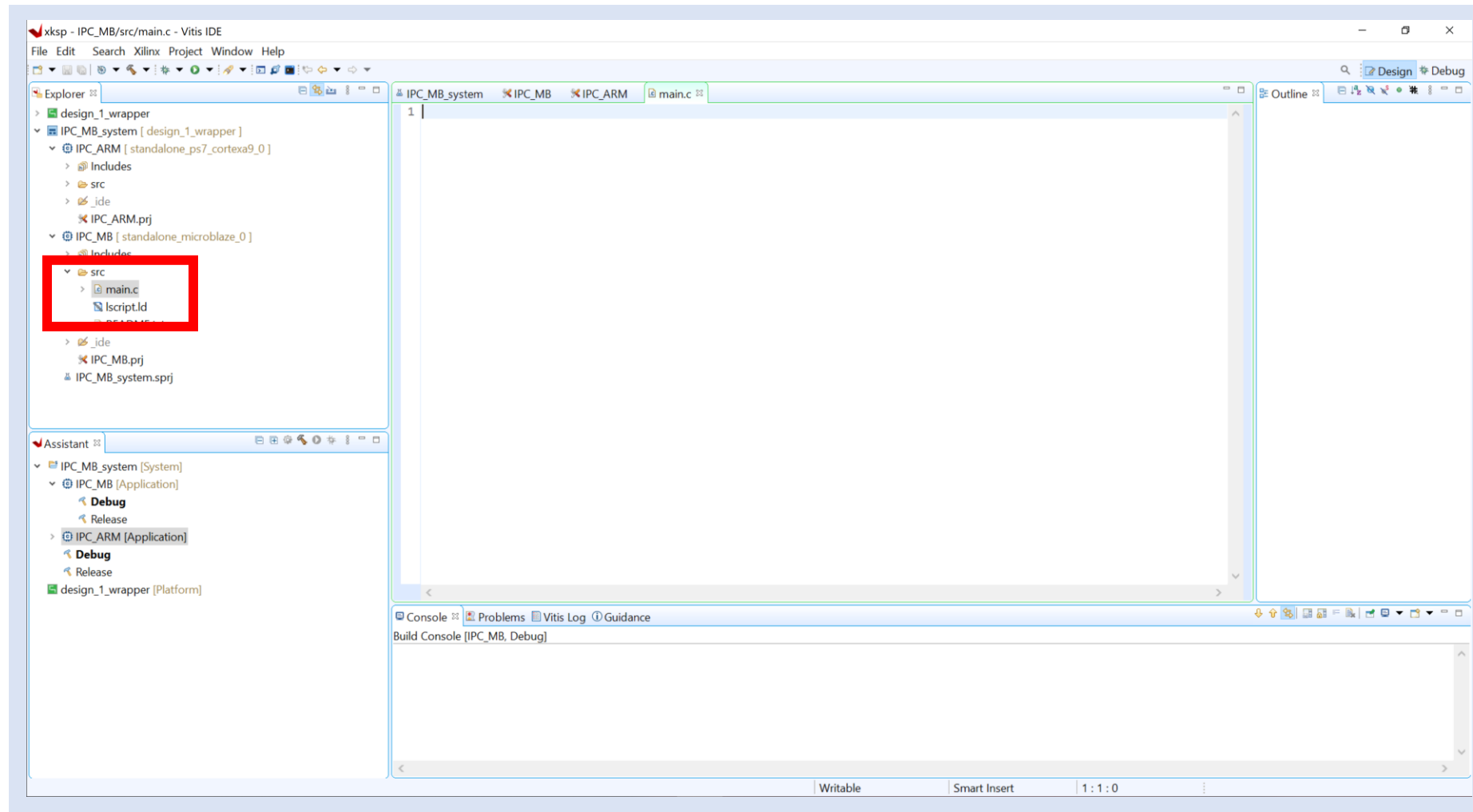
Lab 2: Understanding Vitis Project creation & Flow

Step 46 – Enter a name for the file then click finish



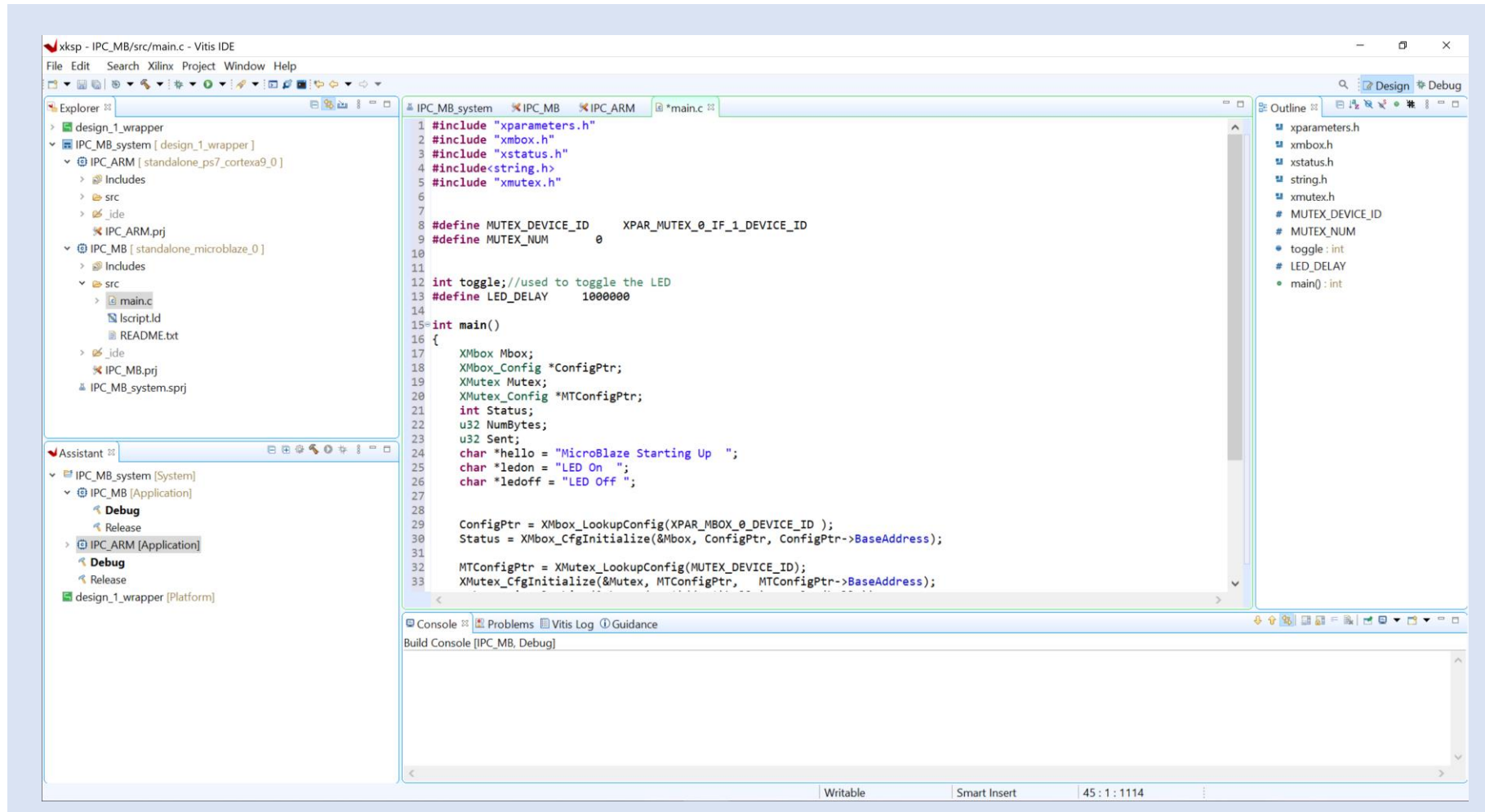
Lab 2: Understanding Vitis Project creation & Flow

Step 47 – Open the newly created file



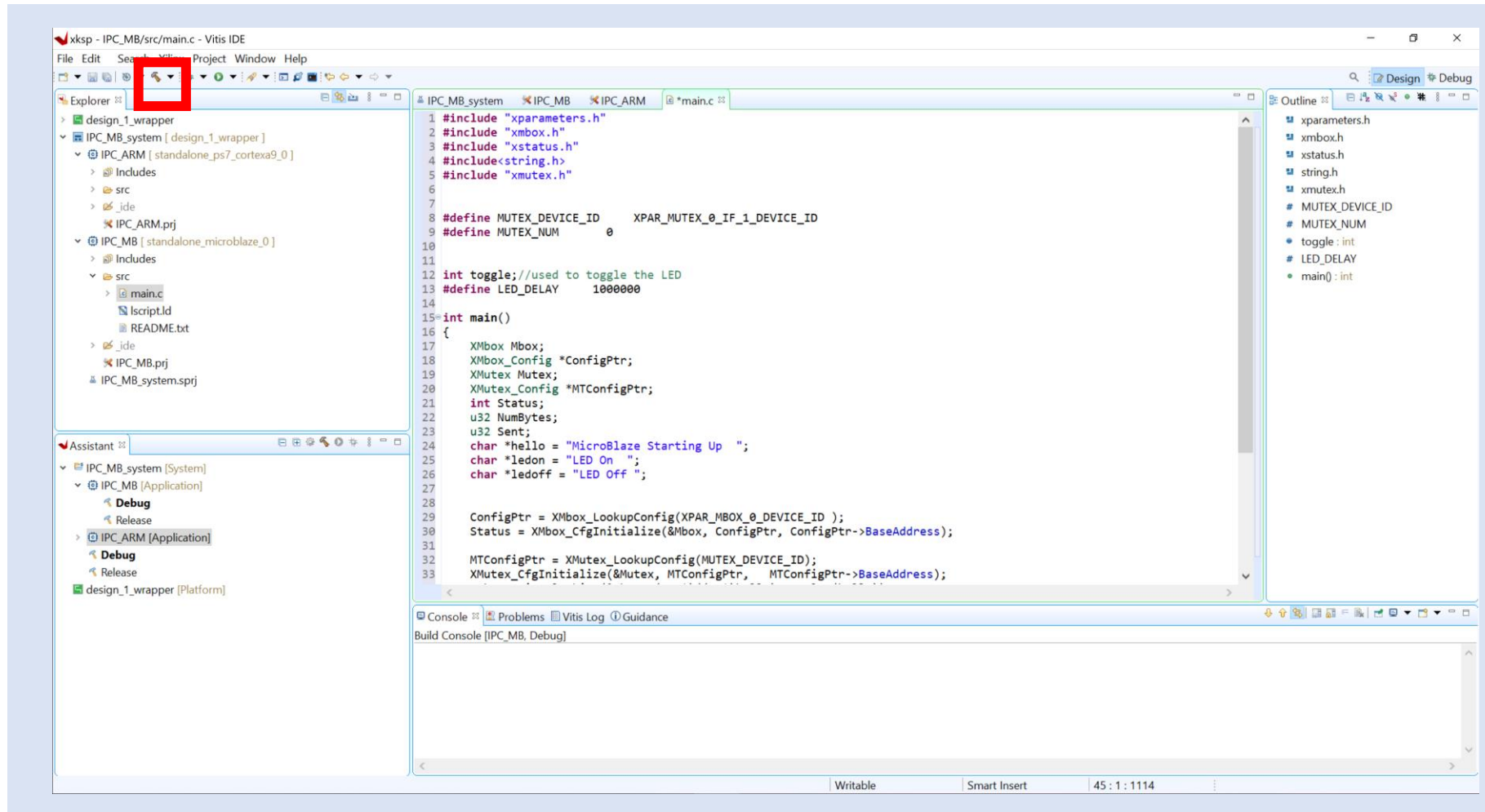
Lab 2: Understanding Vitis Project creation & Flow

Step 48 – Copy the main.c code into the empty file from the Github repo session two lab and save



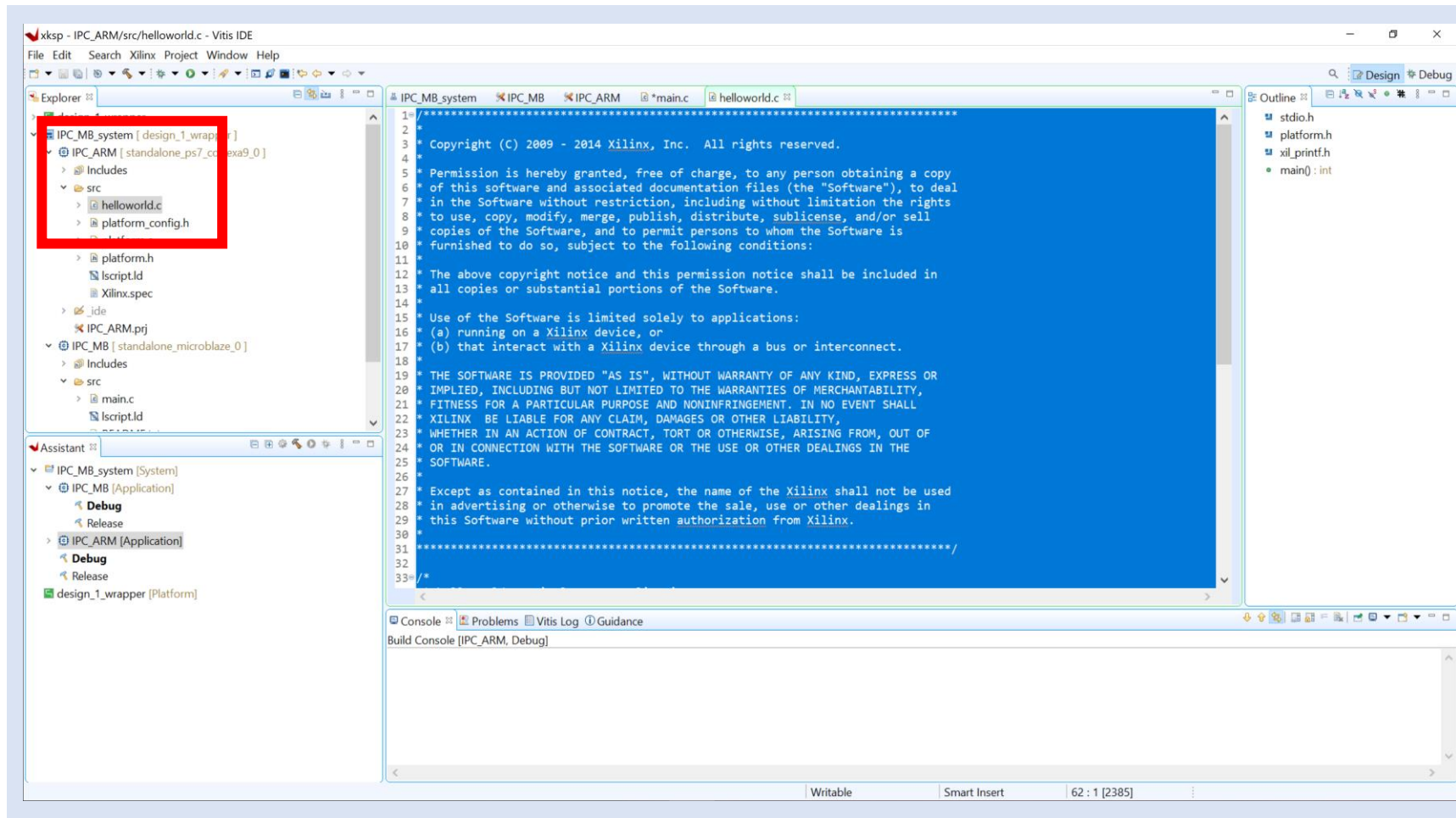
Lab 2: Understanding Vitis Project creation & Flow

Step 49 – Click the hammer to build the code



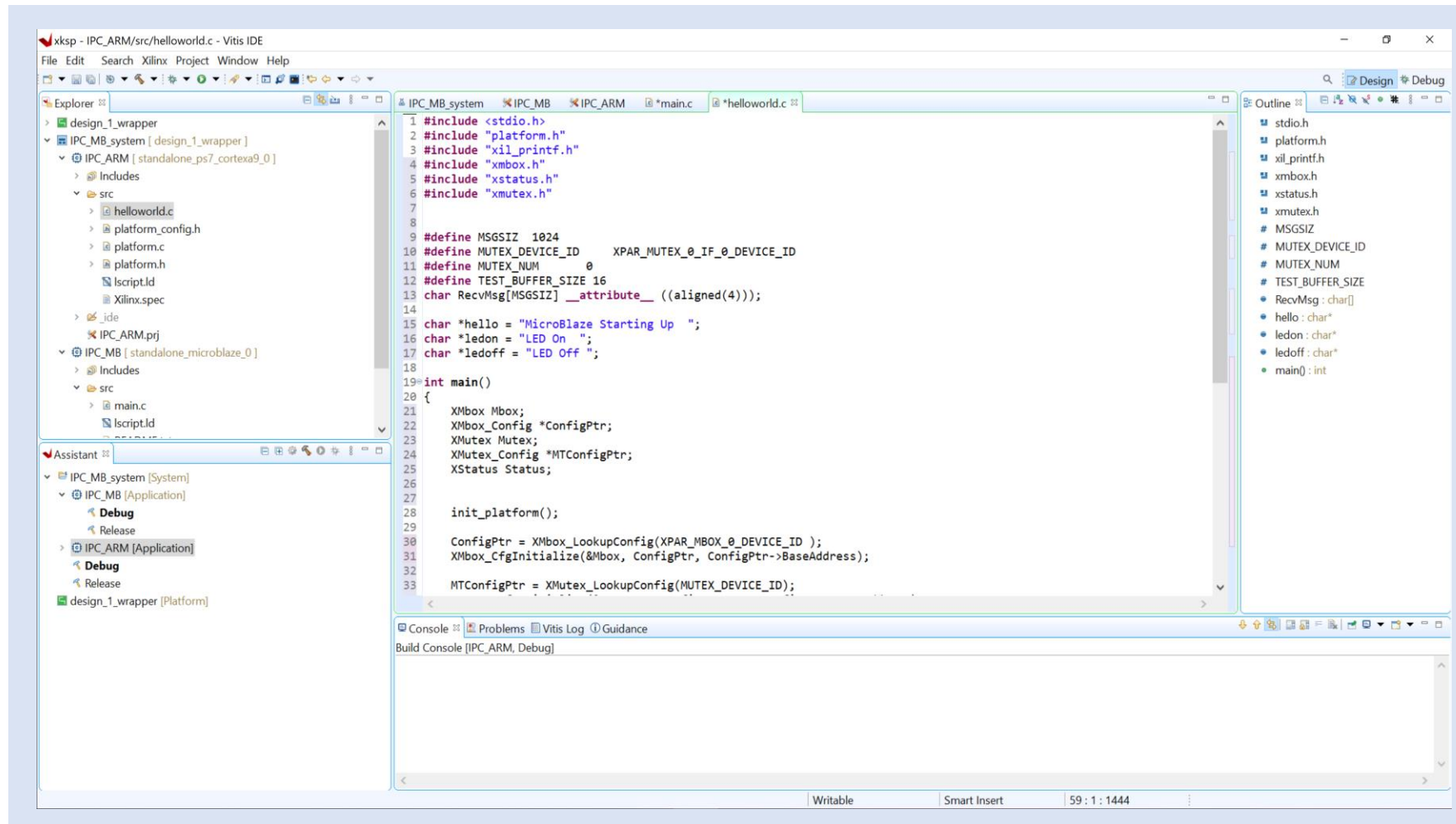
Lab 2: Understanding Vitis Project creation & Flow

Step 50 – Open the IPC_ARM hello world file and select and delete the entire contents



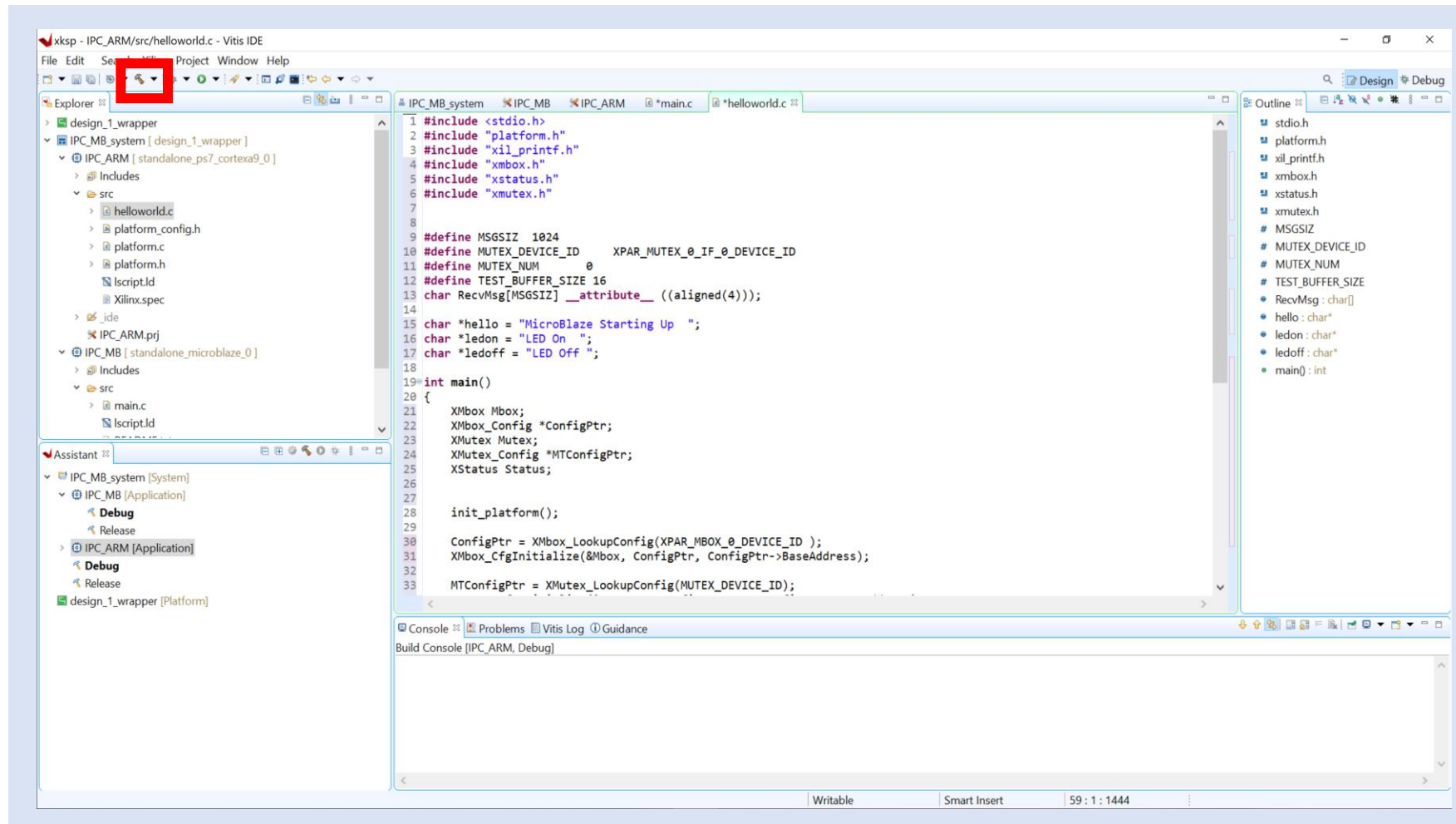
Lab 2: Understanding Vitis Project creation & Flow

Step 51 – Copy across the Zynq Application code from the Gitlab



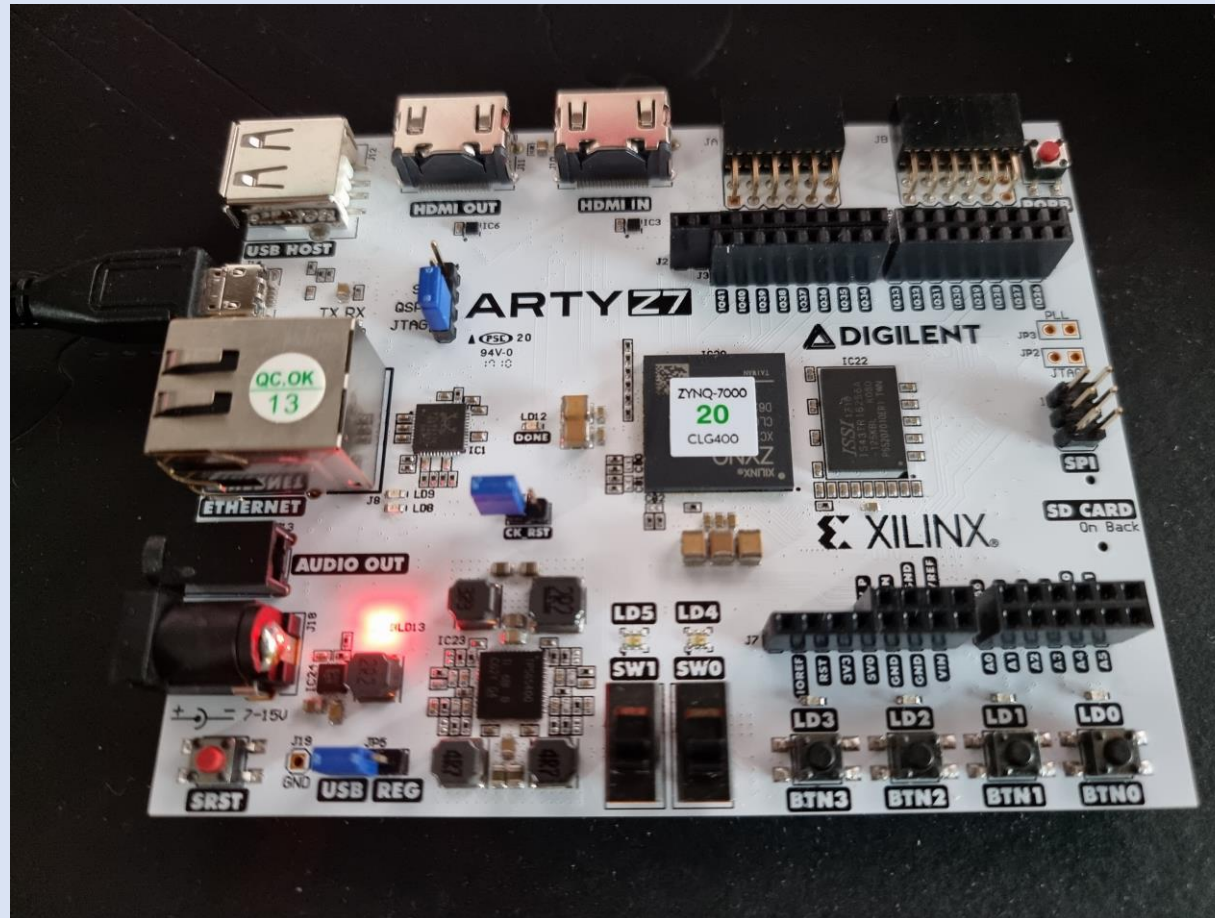
Lab 2: Understanding Vitis Project creation & Flow

Step 52 – Click the hammer to build the code



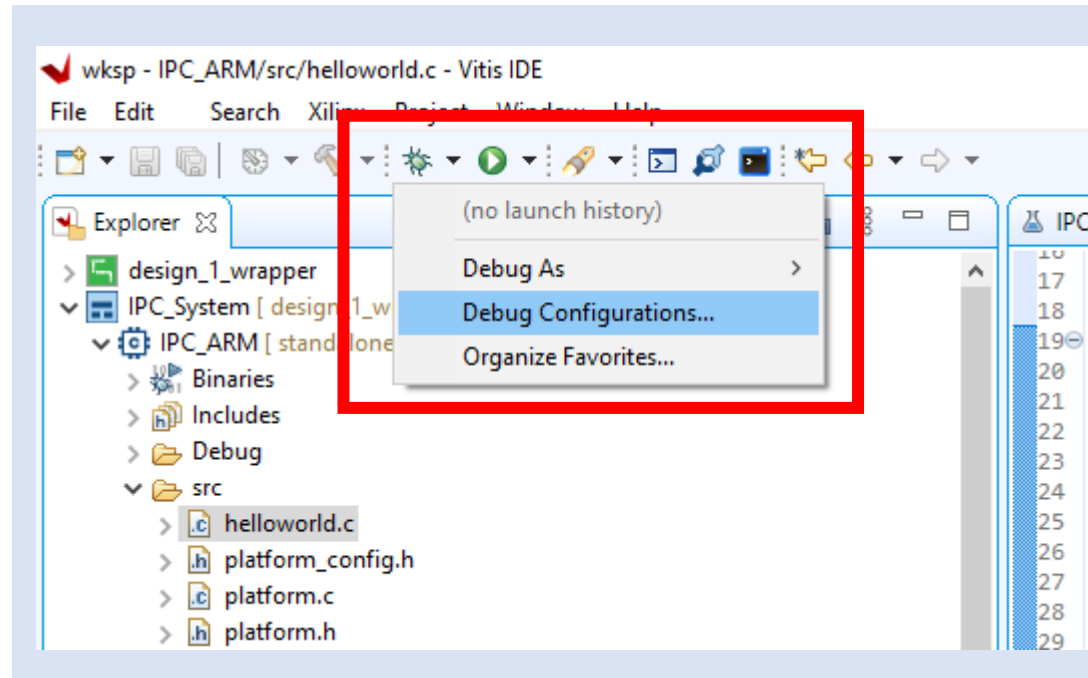
Lab 2: Understanding Vitis Project creation & Flow

Step 53 – Connect the USB to the ArtyZ7, Check the boot Mode is JTAG. Power on the board.



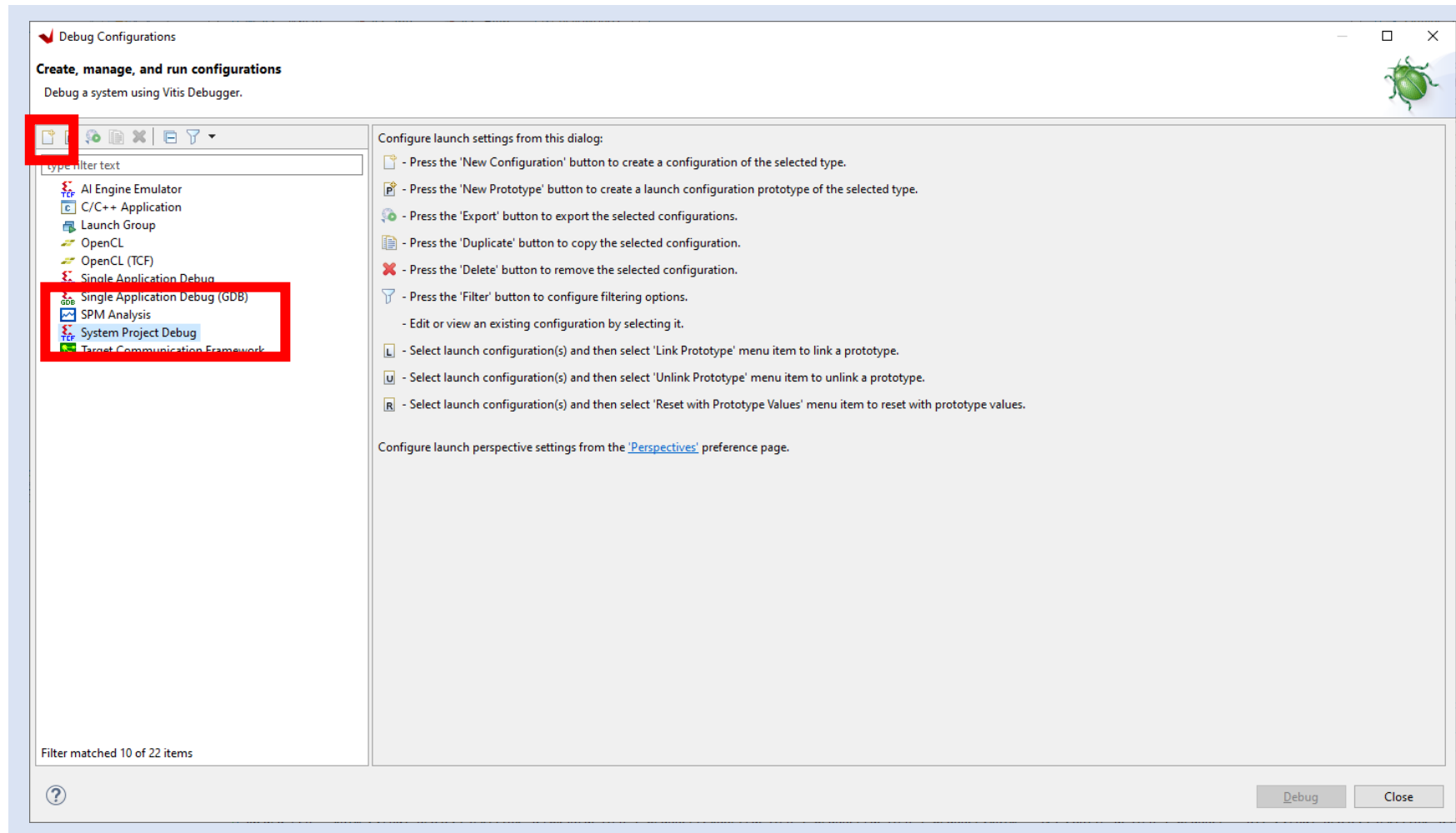
Lab 2: Understanding Vitis Project creation & Flow

Step 54 – From the debug icon, click on the arrow and select debug configuration



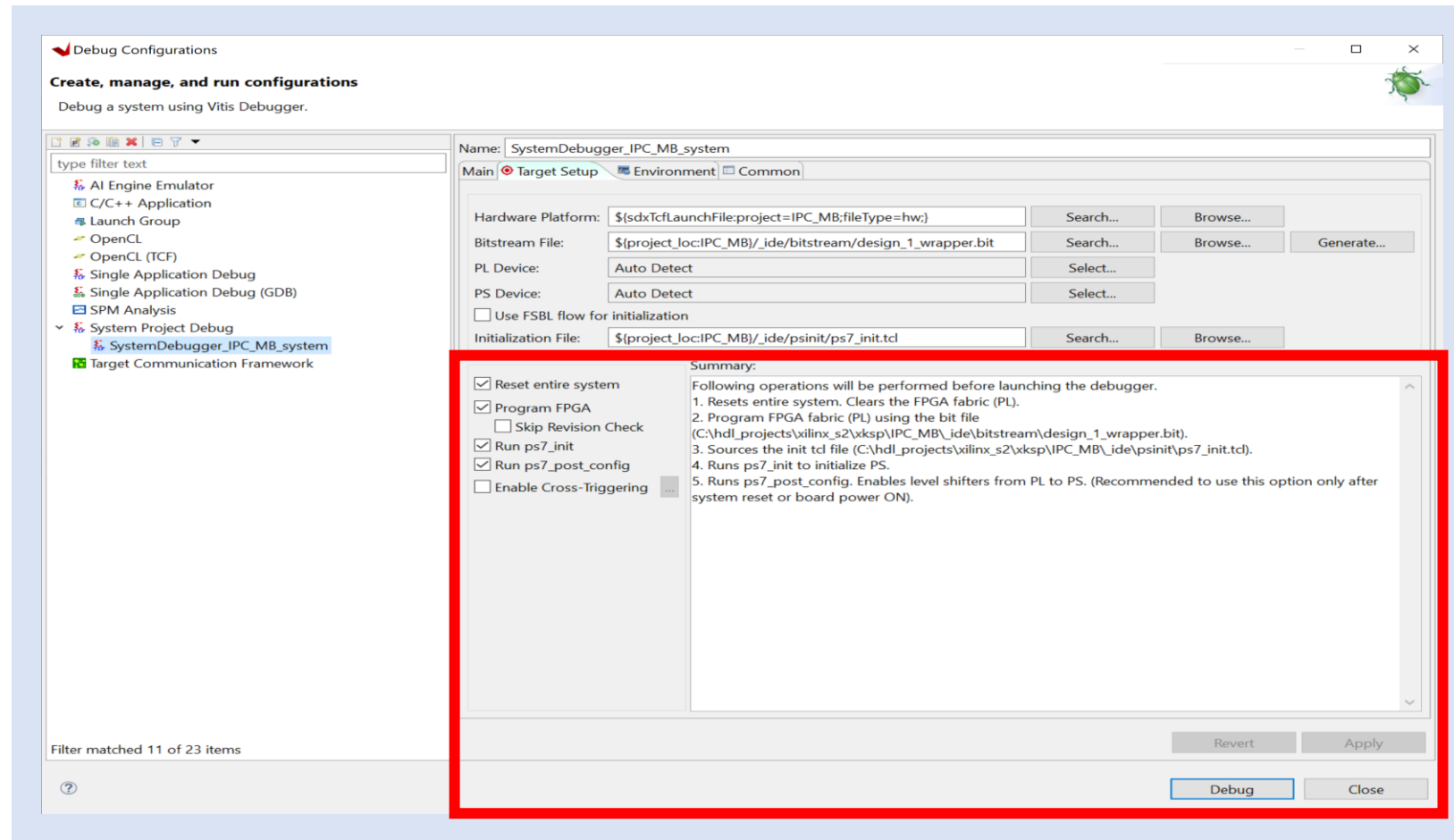
Lab 2: Understanding Vitis Project creation & Flow

Step 55 – Select System Project Debug and new



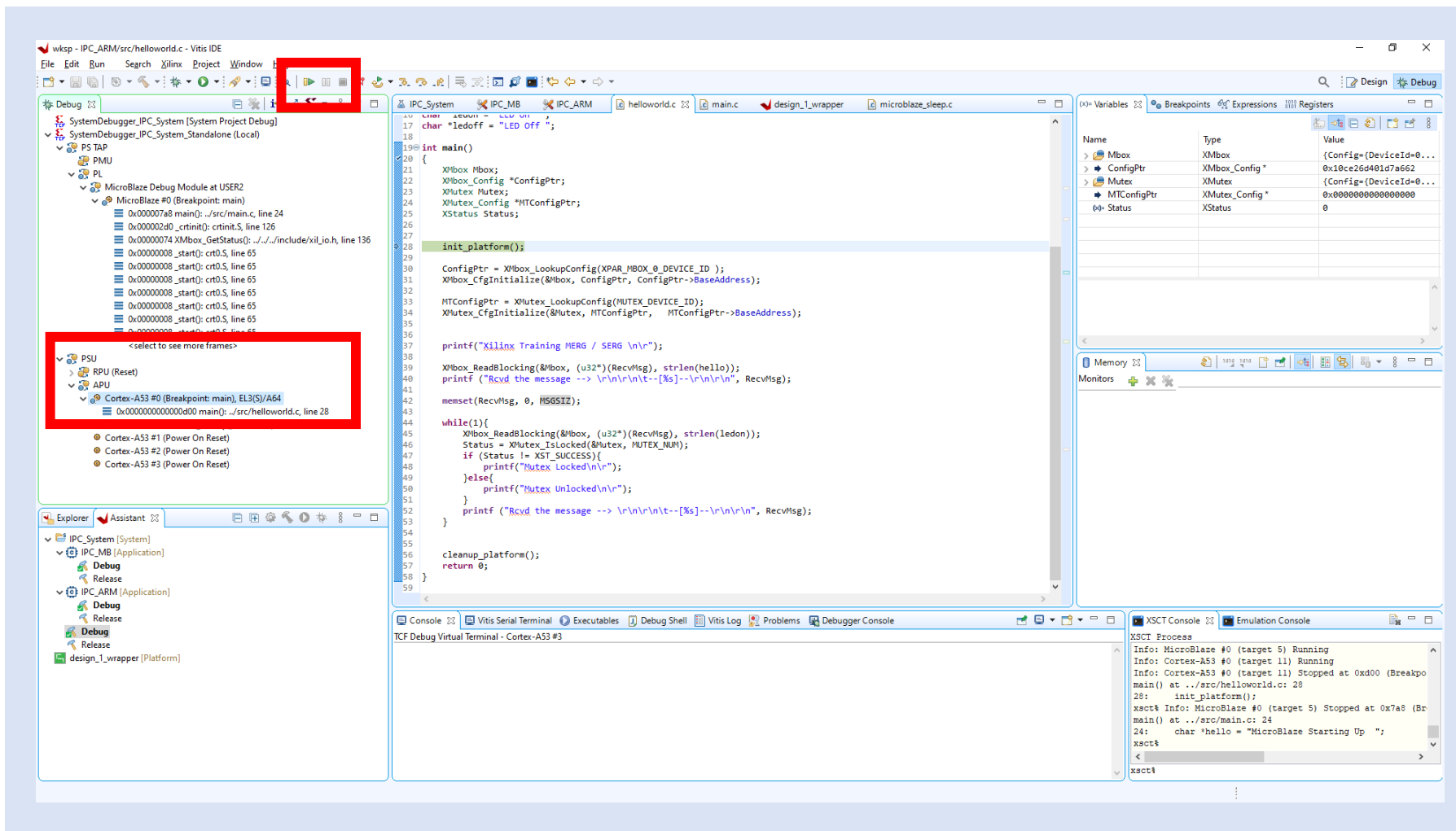
Lab 2: Understanding Vitis Project creation & Flow

Step 56 – Click on Target Set up – This should be auto populated. Notice the PS is configured, followed by the PL before both applications are downloaded. Click Debug



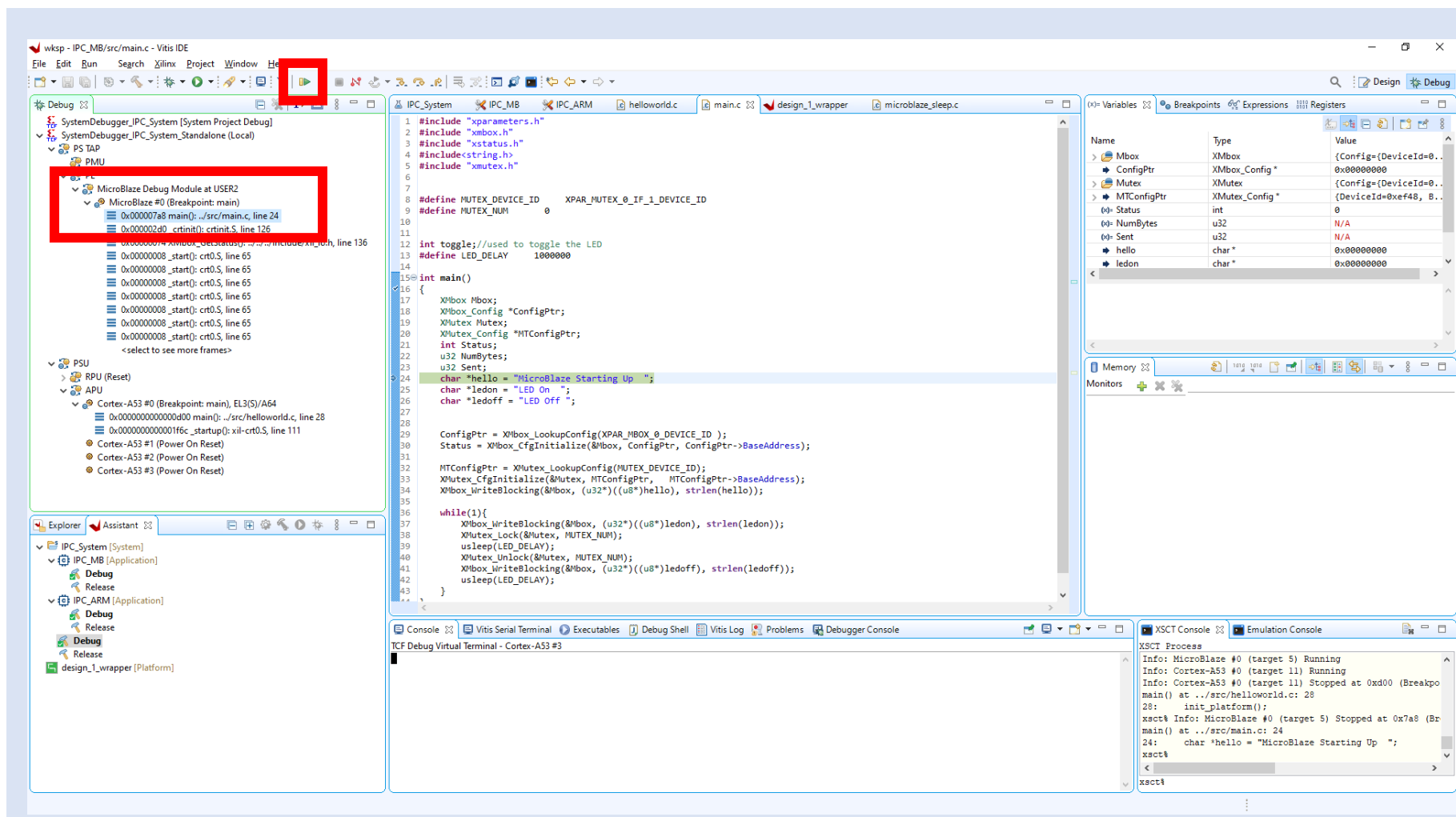
Lab 2: Understanding Vitis Project creation & Flow

Step 57 – Both applications will download and pause execution, click on the APU and hit run to start it



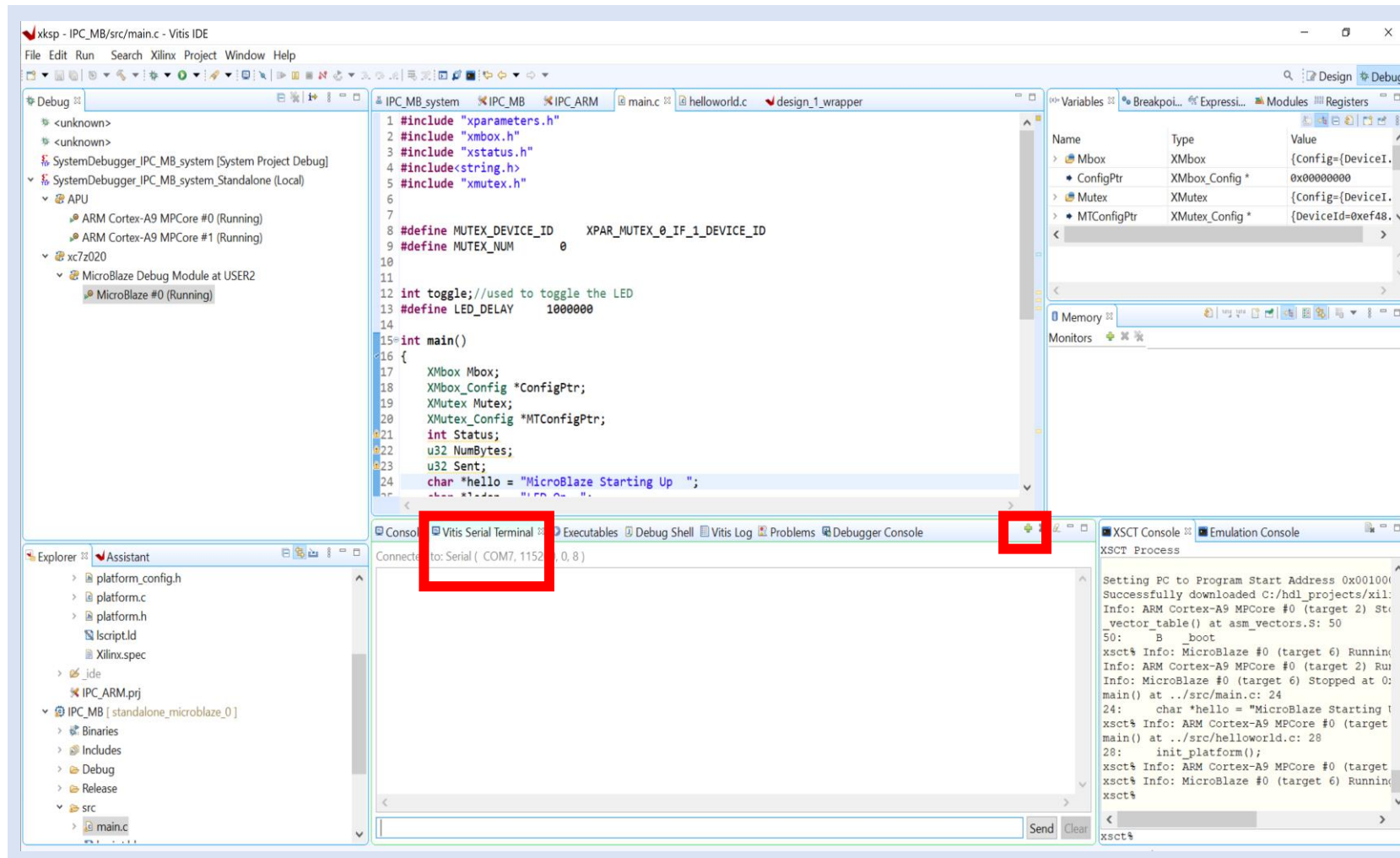
Lab 2: Understanding Vitis Project creation & Flow

Step 58 – Select the MB application and start that executing



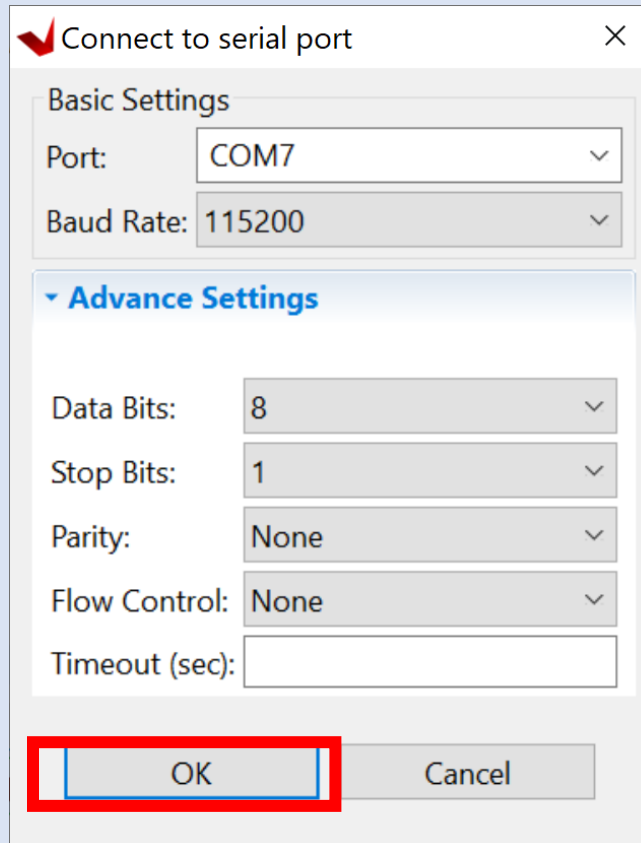
Lab 2: Understanding Vitis Project creation & Flow

Step 59 – Click the Vitis Serial Terminal window and click +



Lab 2: Understanding Vitis Project creation & Flow

Step 60 – Check the settings below and click ok



Connect to serial port

Basic Settings

Port: COM7

Baud Rate: 115200

Advance Settings

Data Bits: 8

Stop Bits: 1

Parity: None

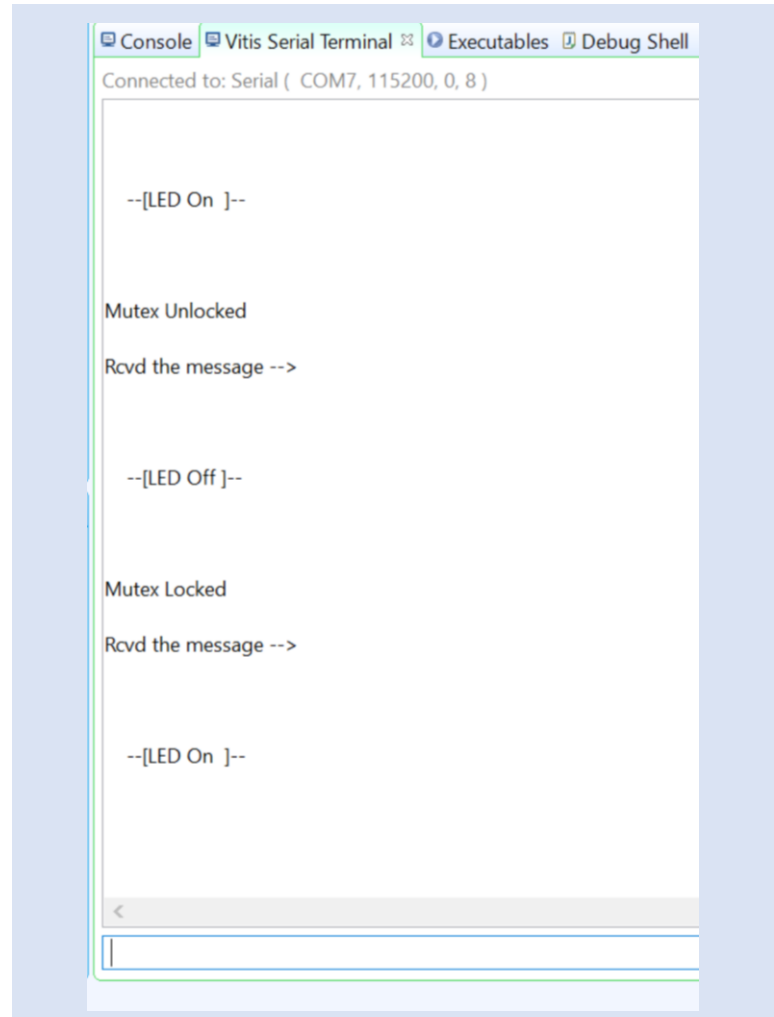
Flow Control: None

Timeout (sec):

OK Cancel

Lab 2: Understanding Vitis Project creation & Flow

Step 61 – The following results will be observed



```
Console Vitis Serial Terminal Executables Debug Shell
Connected to: Serial ( COM7, 115200, 0, 8 )

--[LED On ]--

Mutex Unlocked
Rcvd the message -->

--[LED Off ]--

Mutex Locked
Rcvd the message -->

--[LED On ]--
```