

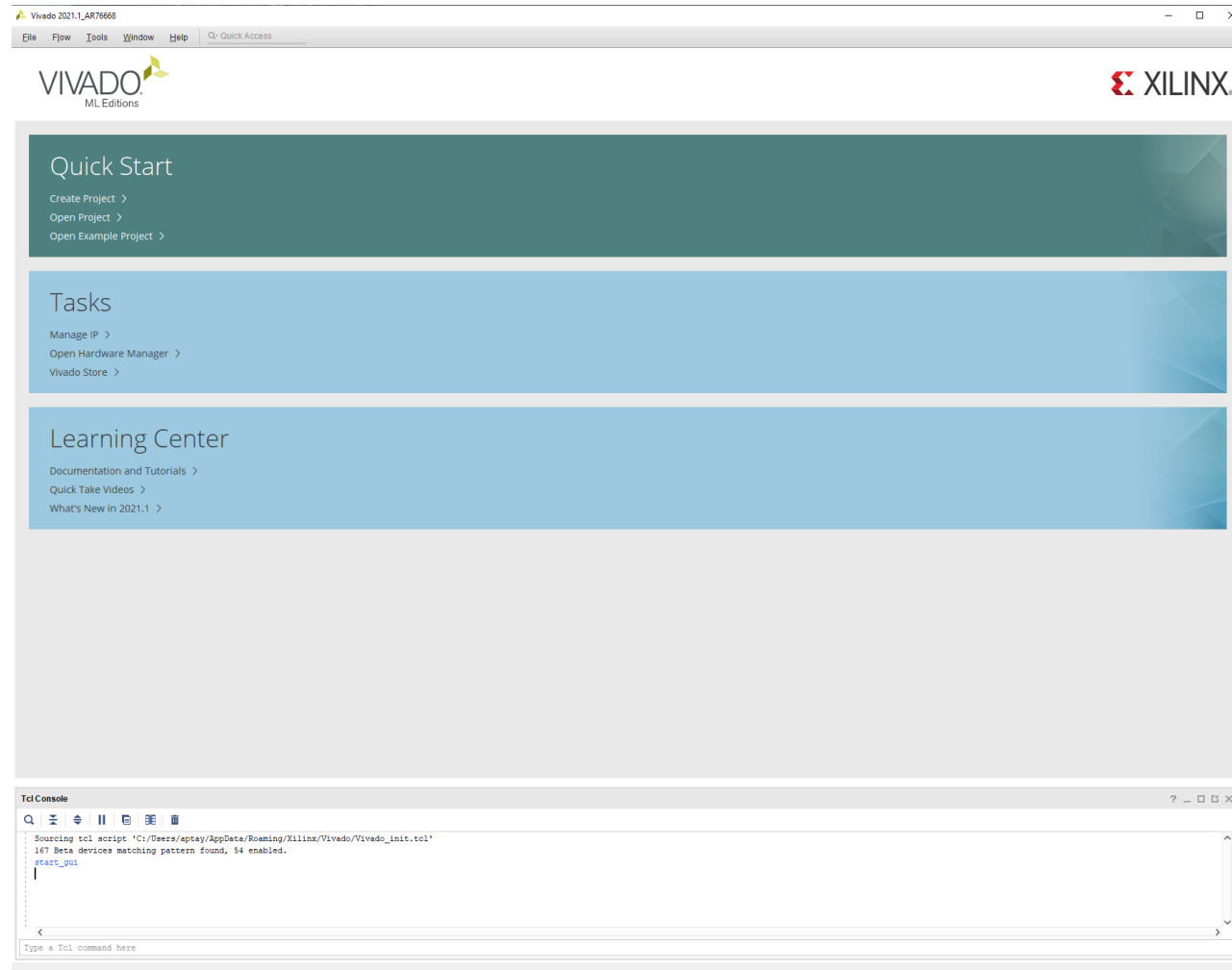


Mastering MicroBlaze

Lab Book

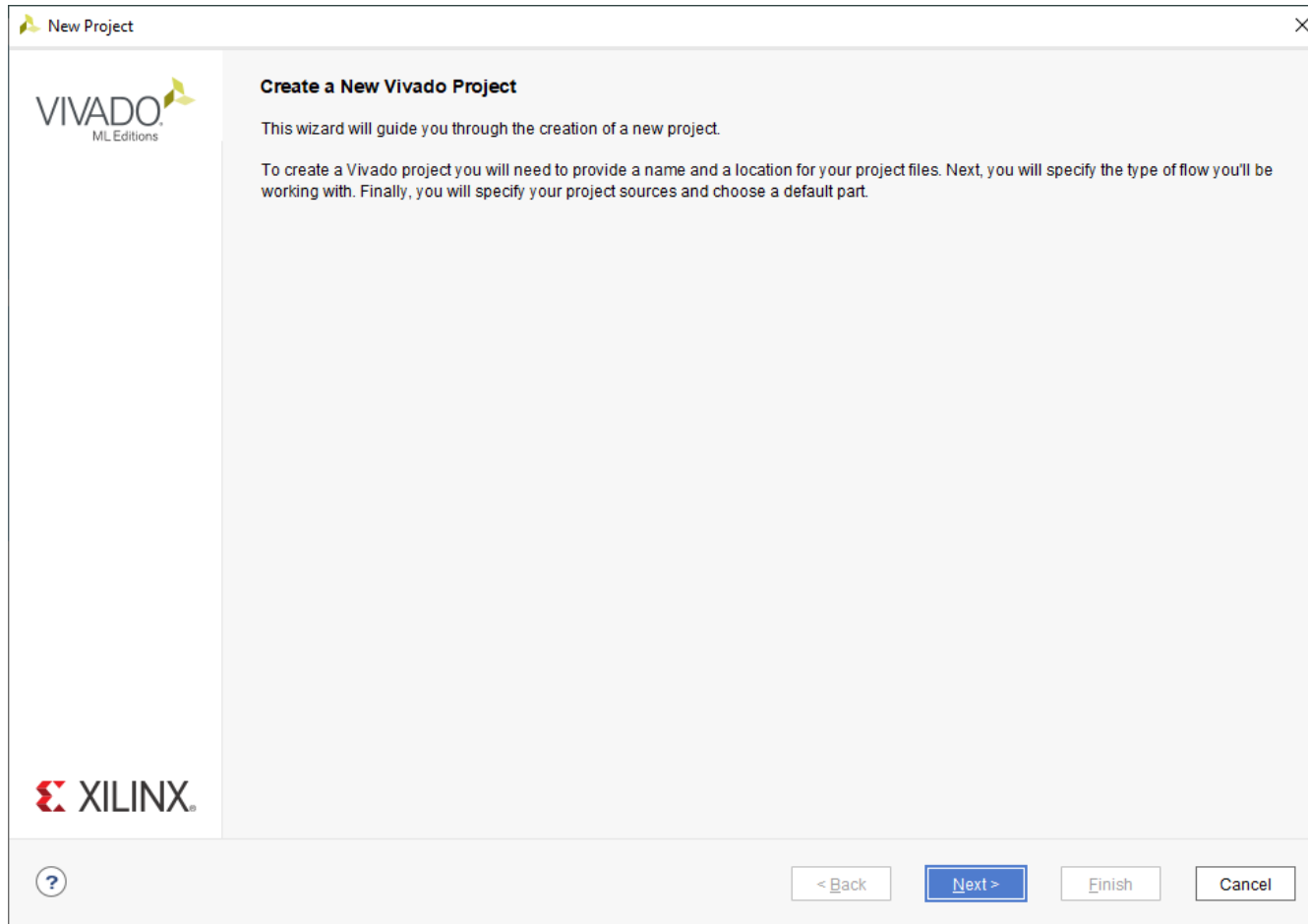
Lab: Mastering MicroBlaze

Open Vivado and Create a new project



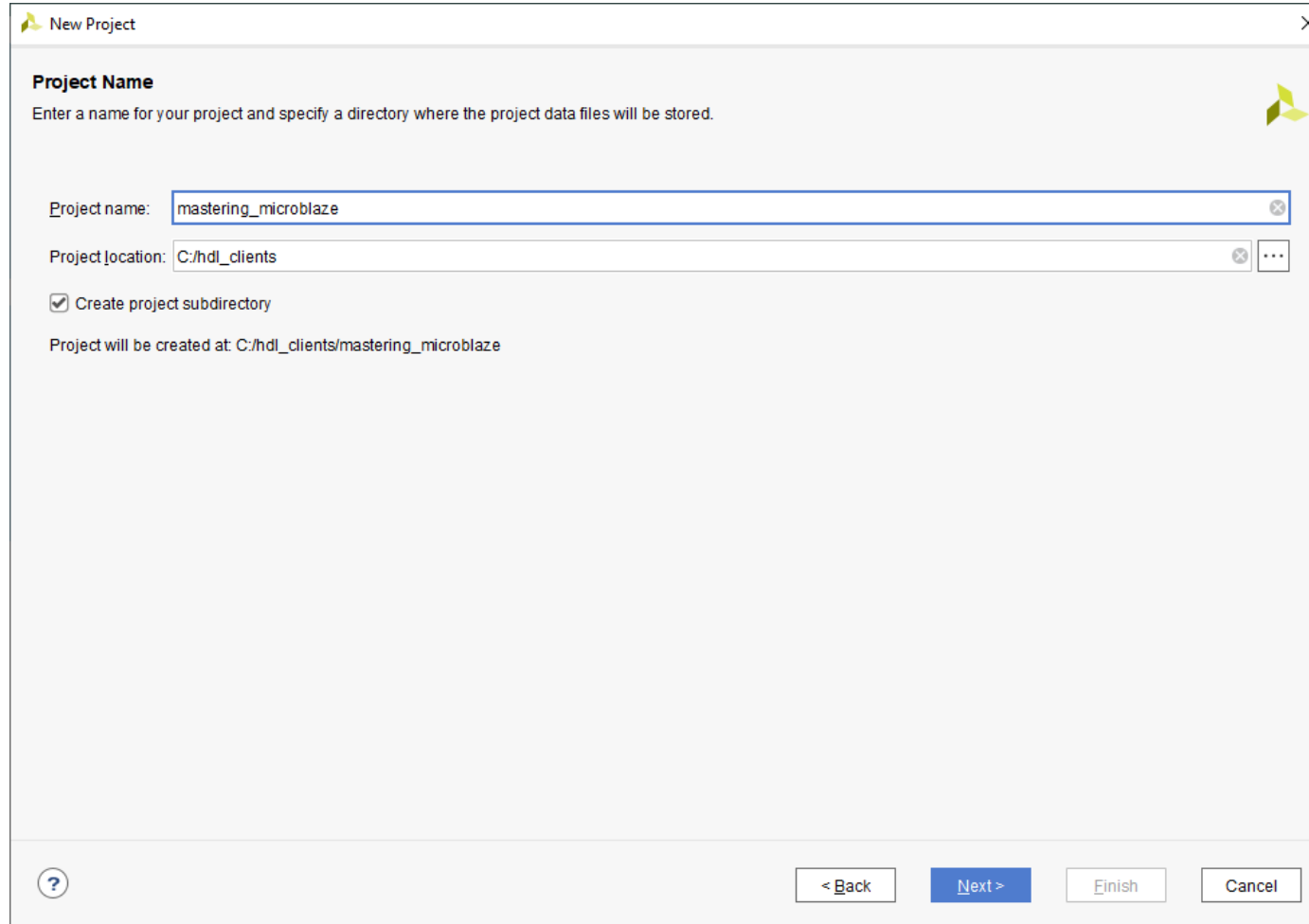
Lab: Mastering MicroBlaze

Click Next



Lab: Mastering MicroBlaze

Enter a location and project name



The image shows a 'New Project' dialog box with a title bar containing a yellow cube icon and a close button. The main area has a 'Project Name' section with a subtitle 'Enter a name for your project and specify a directory where the project data files will be stored.' and a yellow cube icon. Below this are two text input fields: 'Project name:' containing 'mastering_microblaze' and 'Project location:' containing 'C:/hdl_clients'. The 'Project location' field has a small 'x' icon and a browse button ('...'). A checkbox labeled 'Create project subdirectory' is checked. Below the checkbox, it says 'Project will be created at: C:/hdl_clients/mastering_microblaze'. At the bottom, there is a help button ('?'), a '< Back' button, a blue 'Next >' button, an 'Finish' button, and a 'Cancel' button.

New Project

Project Name
Enter a name for your project and specify a directory where the project data files will be stored.

Project name: mastering_microblaze

Project location: C:/hdl_clients

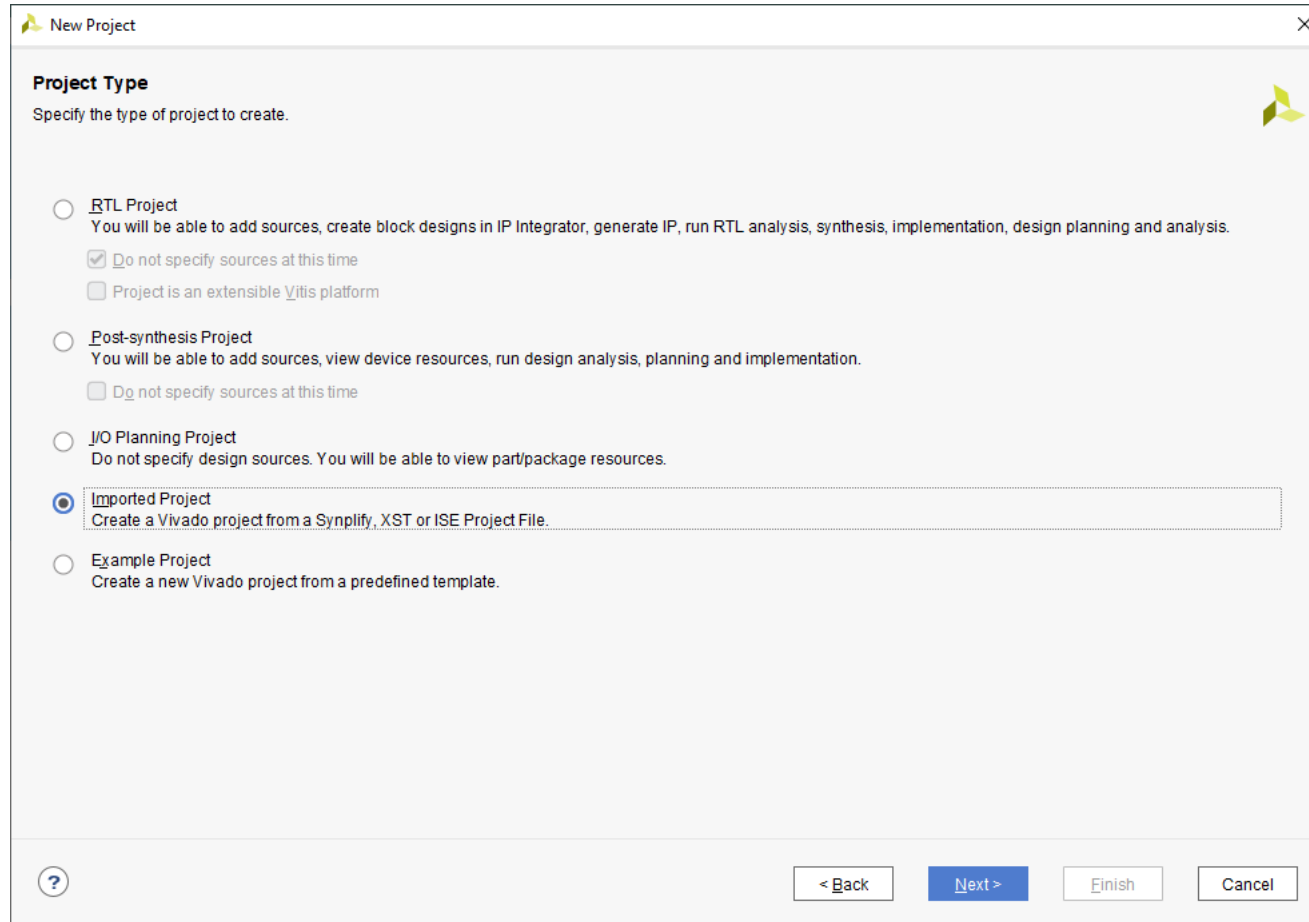
☒ Create project subdirectory

Project will be created at: C:/hdl_clients/mastering_microblaze

? < Back Next > Finish Cancel

Lab: Mastering MicroBlaze

Select project type as RTL Project and check Do Not Specify Sources at this time.



The image shows a 'New Project' dialog box with the following content:

New Project

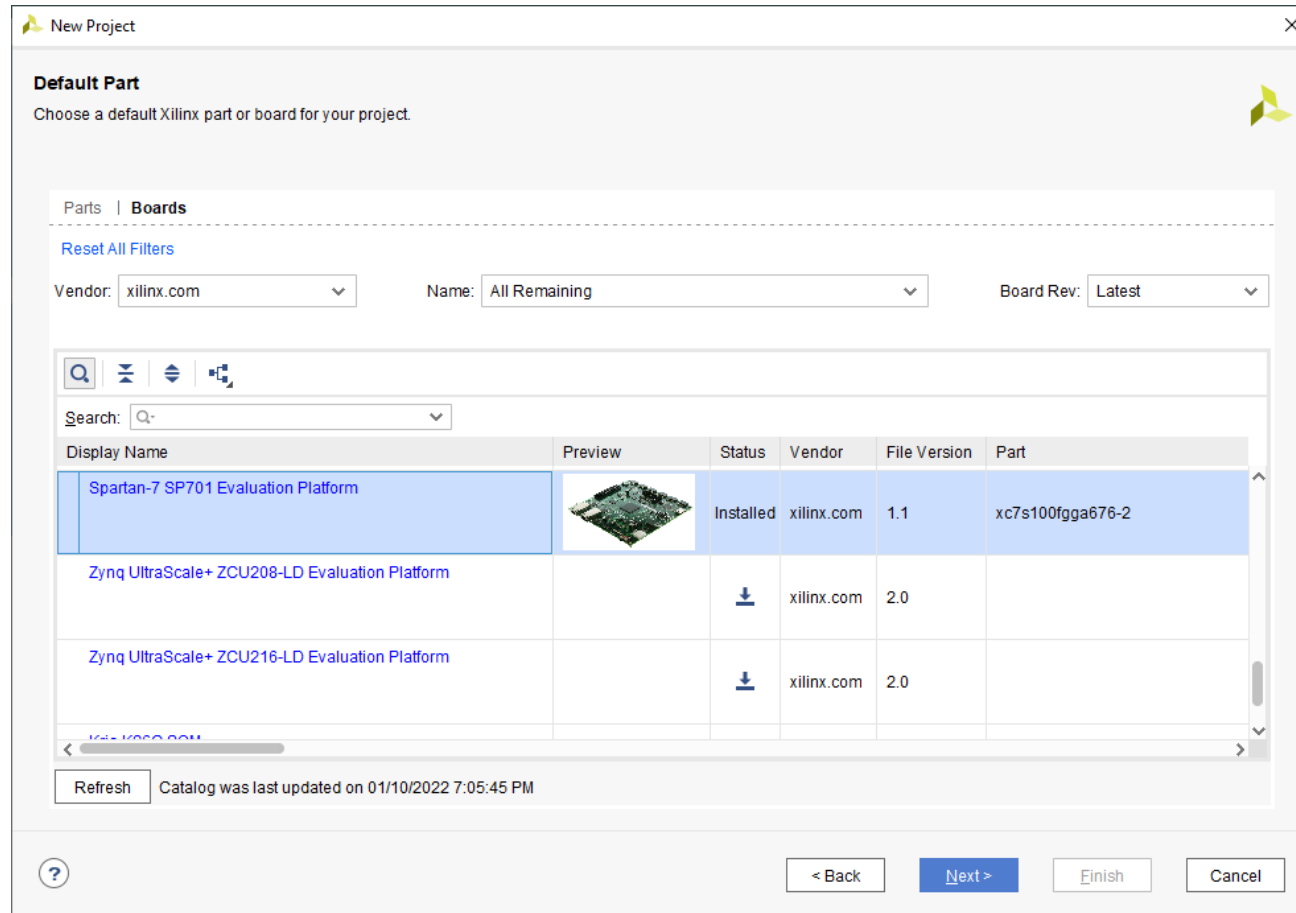
Project Type
Specify the type of project to create.

- ☐ **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
- ☐ **Post-synthesis Project**
You will be able to add sources, view device resources, run design analysis, planning and implementation.
☐ Do not specify sources at this time
- ☐ **I/O Planning Project**
Do not specify design sources. You will be able to view part/package resources.
- ☒ **Imported Project**
Create a Vivado project from a Synplify, XST or ISE Project File.
- ☐ **Example Project**
Create a new Vivado project from a predefined template.

Buttons at the bottom: ? < Back Next > Finish Cancel

Lab: Mastering MicroBlaze

Select your development board – I will be using the SP701. But ANY 7 Series / UltraScale (+) board with a LED connected to the IO should work.



New Project




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

Parts | **Boards**

[Reset All Filters](#)

Vendor: Name: Board Rev:

Search:

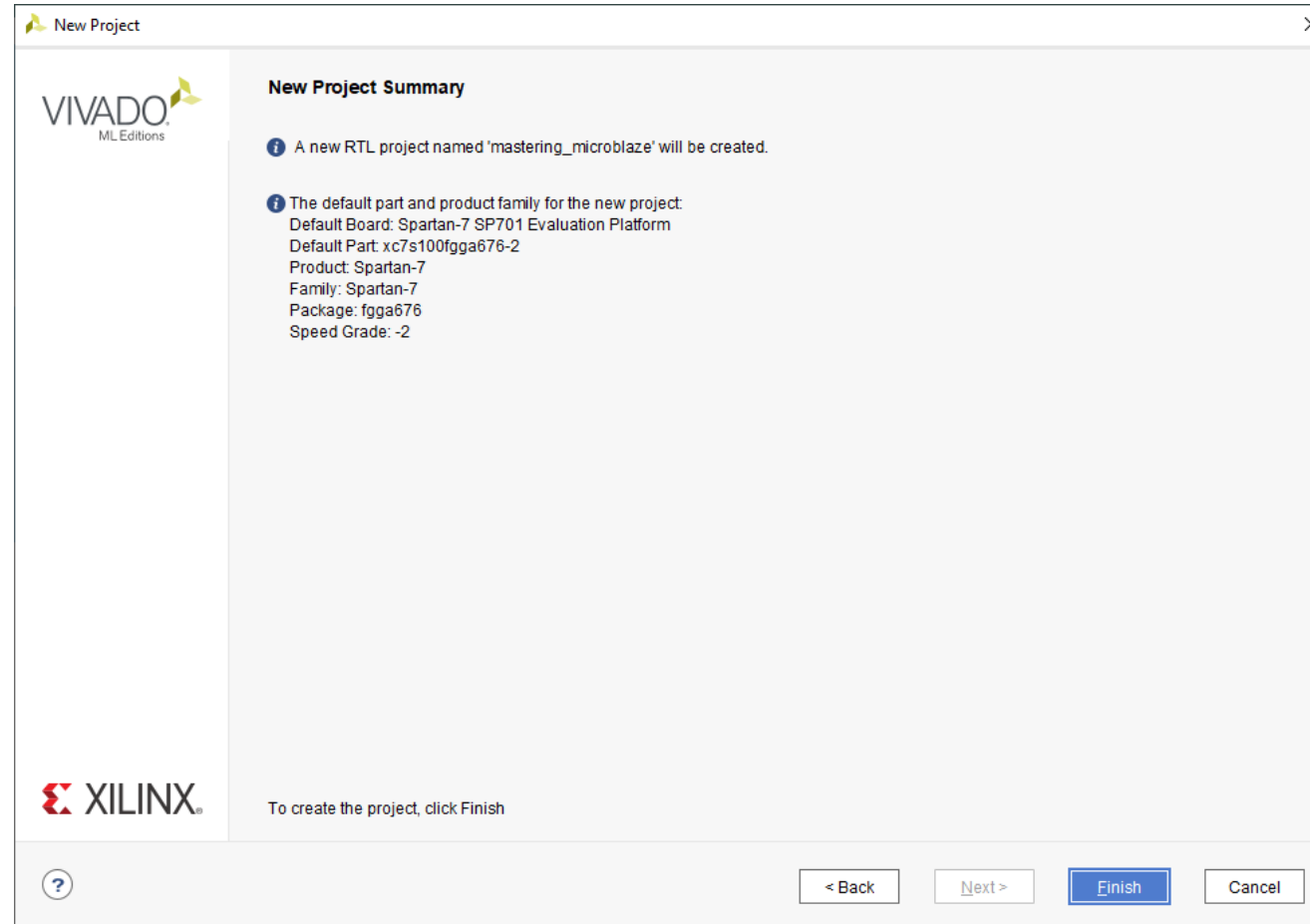
Display Name	Preview	Status	Vendor	File Version	Part
Spartan-7 SP701 Evaluation Platform		Installed	xilinx.com	1.1	xc7s100fpga676-2
Zynq UltraScale+ ZCU208-LD Evaluation Platform			xilinx.com	2.0	
Zynq UltraScale+ ZCU216-LD Evaluation Platform			xilinx.com	2.0	

[Refresh](#) Catalog was last updated on 01/10/2022 7:05:45 PM

[?<](#) [< Back](#) [Next >](#) [Finish](#) [Cancel](#)

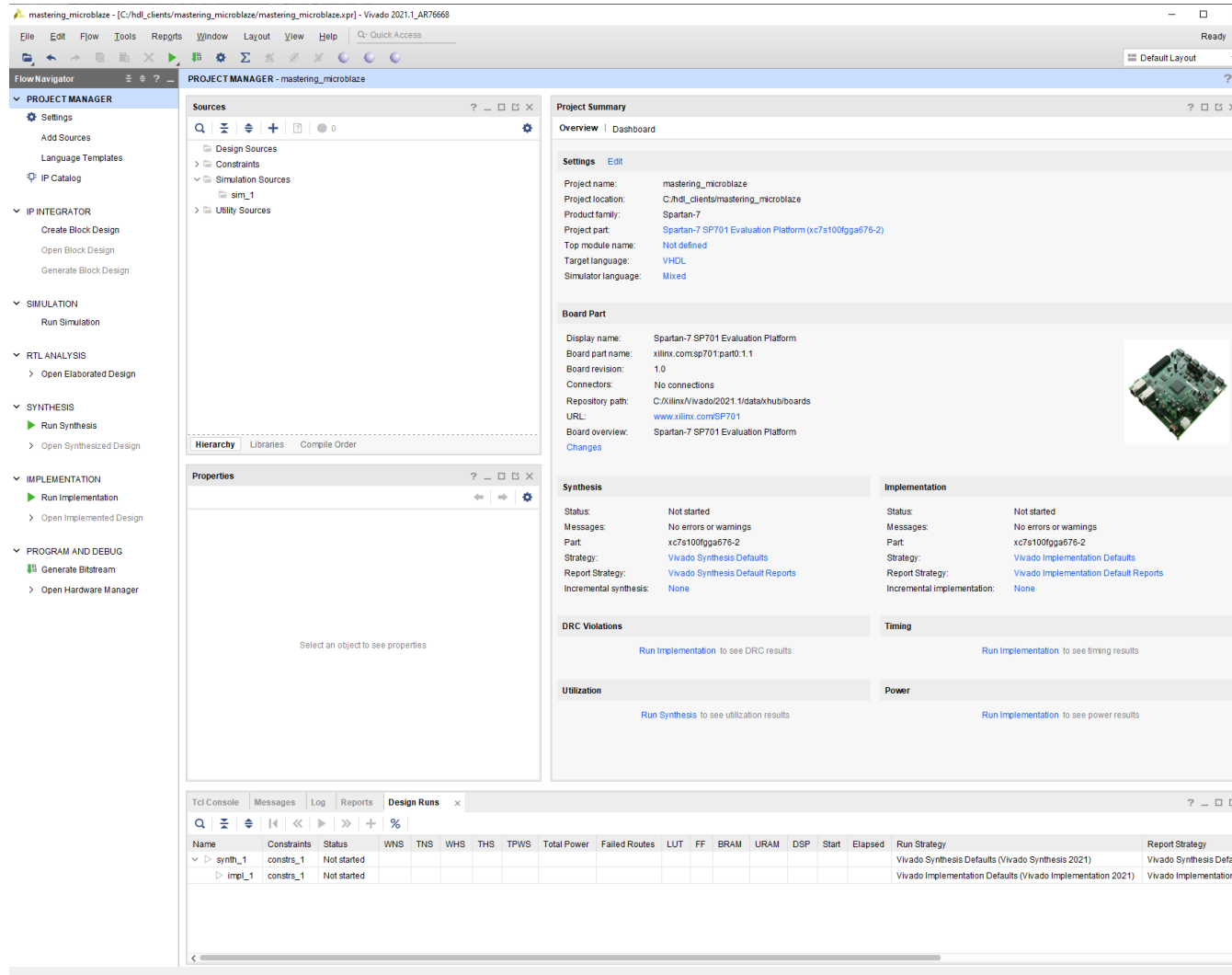
Lab: Mastering MicroBlaze

Click Finish



Lab: Mastering MicroBlaze

The open project should look as below



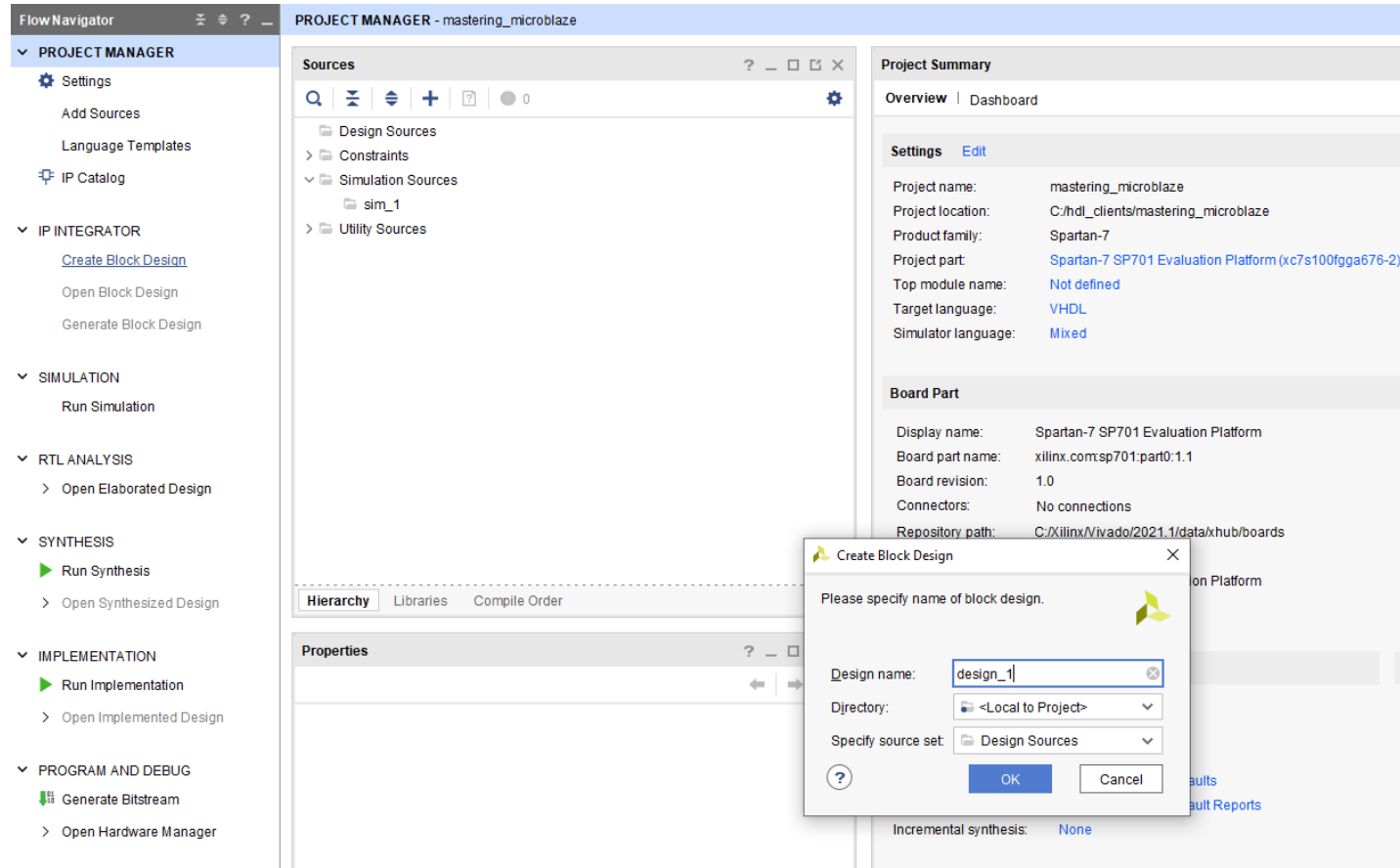
The screenshot displays the Vivado IDE interface for the 'mastering_microblaze' project. The interface is divided into several panels:

- Flow Navigator:** Shows the project workflow, including Settings, Add Sources, Language Templates, IP Catalog, IP INTEGRATOR, SIMULATION, RTL ANALYSIS, SYNTHESIS, IMPLEMENTATION, and PROGRAM AND DEBUG.
- PROJECT MANAGER - mastering_microblaze:**
 - Sources:** Lists Design Sources, Constraints, Simulation Sources (sim_1), and Utility Sources.
 - Properties:** A panel for viewing the properties of selected objects.
- Project Summary:**
 - Overview | Dashboard:**
 - Settings:** Project name: mastering_microblaze, Project location: C:/hdl_clients/mastering_microblaze, Product family: Spartan-7, Project part: Spartan-7 SP701 Evaluation Platform (xc7s100fpga676-2), Top module name: Not defined, Target language: VHDL, Simulator language: Mixed.
 - Board Part:** Display name: Spartan-7 SP701 Evaluation Platform, Board part name: xilinx.com:sp701:part0:1.1, Board revision: 1.0, Connectors: No connections, Repository path: C:/Xilinx/Vivado/2021.1/data/hub/boards, URL: www.xilinx.com/SP701, Board overview: Spartan-7 SP701 Evaluation Platform.
 - Synthesis:** Status: Not started, Messages: No errors or warnings, Part: xc7s100fpga676-2, Strategy: Vivado Synthesis Defaults, Report Strategy: Vivado Synthesis Default Reports, Incremental synthesis: None.
 - Implementation:** Status: Not started, Messages: No errors or warnings, Part: xc7s100fpga676-2, Strategy: Vivado Implementation Defaults, Report Strategy: Vivado Implementation Default Reports, Incremental implementation: None.
 - DRC Violations:** Run Implementation to see DRC results.
 - Timing:** Run Implementation to see timing results.
 - Utilization:** Run Synthesis to see utilization results.
 - Power:** Run Implementation to see power results.
- Design Runs:** A table showing the status of various design runs.

Name	Constraints	Status	WNS	TNS	WHIS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAM	URAM	DSP	Start	Elapsed	Run Strategy	Report Strategy
synth_1	constrs_1	Not started															Vivado Synthesis Defaults (Vivado Synthesis 2021)	Vivado Synthesis Default
impl_1	constrs_1	Not started															Vivado Implementation Defaults (Vivado Implementation 2021)	Vivado Implementation D

Lab: Mastering MicroBlaze

Click Create Block Design and leave the name unchanged



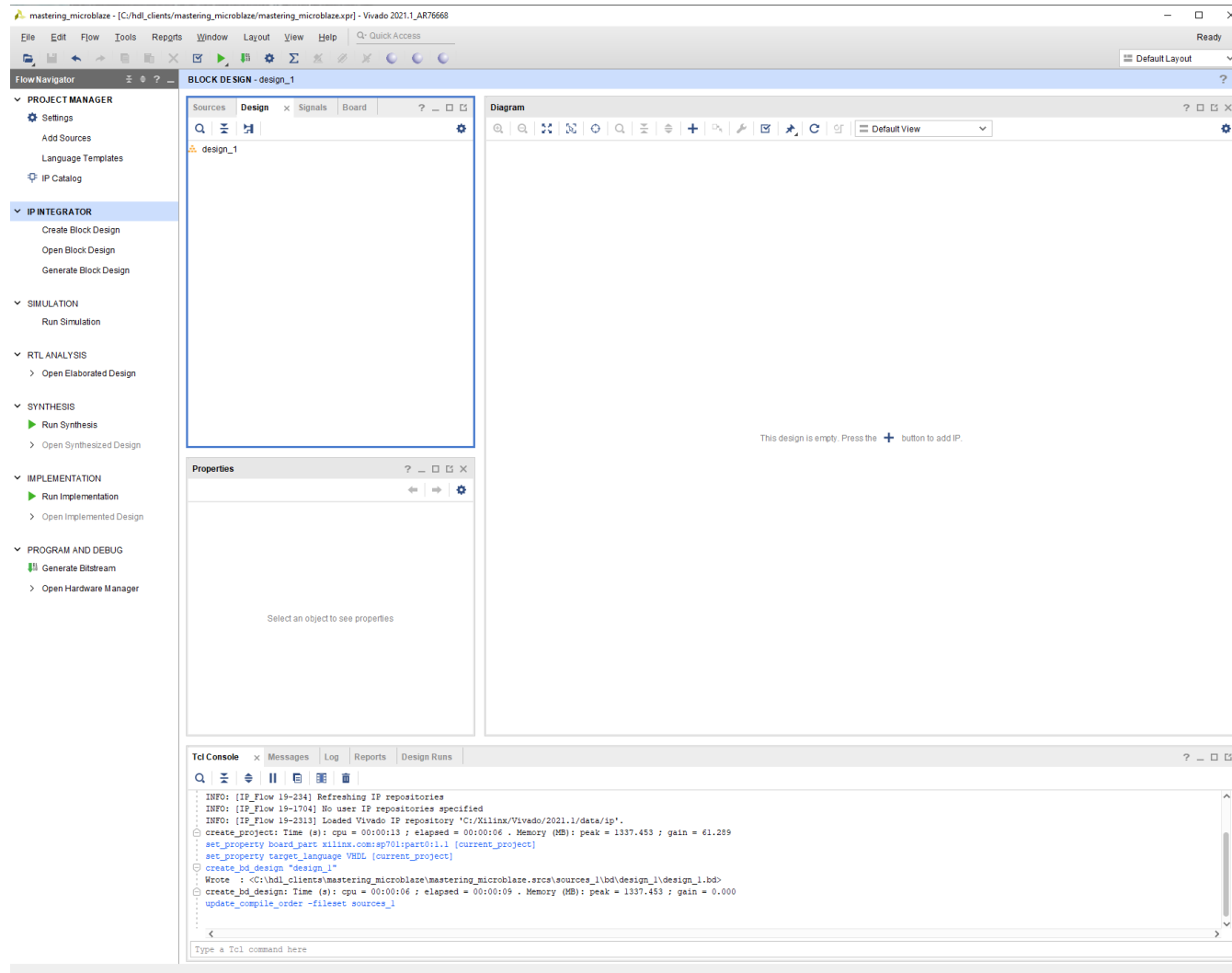
The screenshot displays the Xilinx IDE Project Manager interface for a project named 'mastering_microblaze'. The left sidebar shows the 'Flow Navigator' with various project management options. The main area is divided into 'Sources' and 'Properties' tabs. A 'Create Block Design' dialog box is open in the foreground, prompting the user to specify the name of the block design. The dialog box contains the following fields:

- Design name:** design_1
- Directory:** <Local to Project>
- Specify source set:** Design Sources

The dialog box also includes 'OK' and 'Cancel' buttons. The background interface shows the 'Project Summary' panel on the right, which provides an overview of the project settings, including the project name, location, product family (Spartan-7), and target language (VHDL).

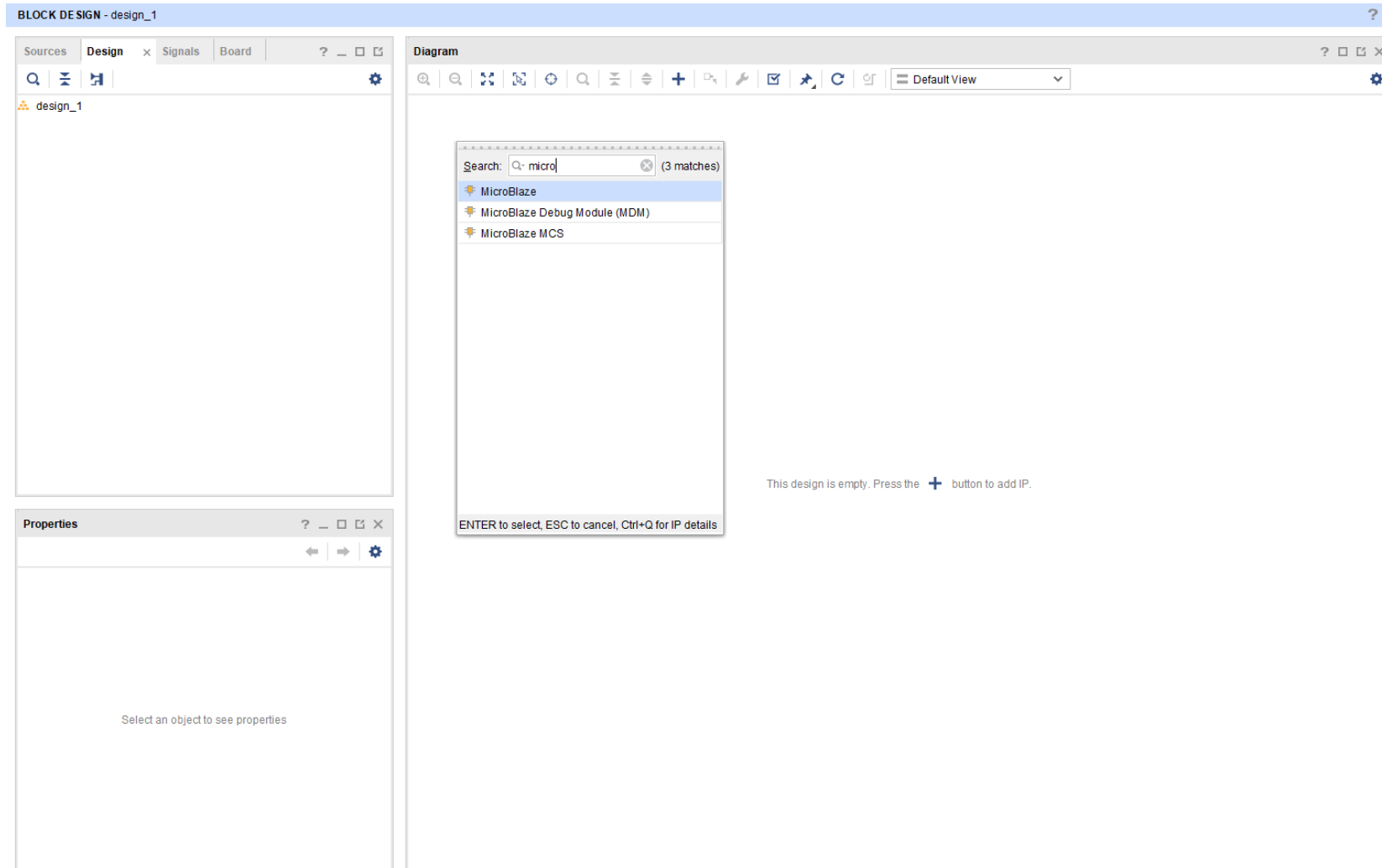
Lab: Mastering MicroBlaze

The Block Diagram should look as below



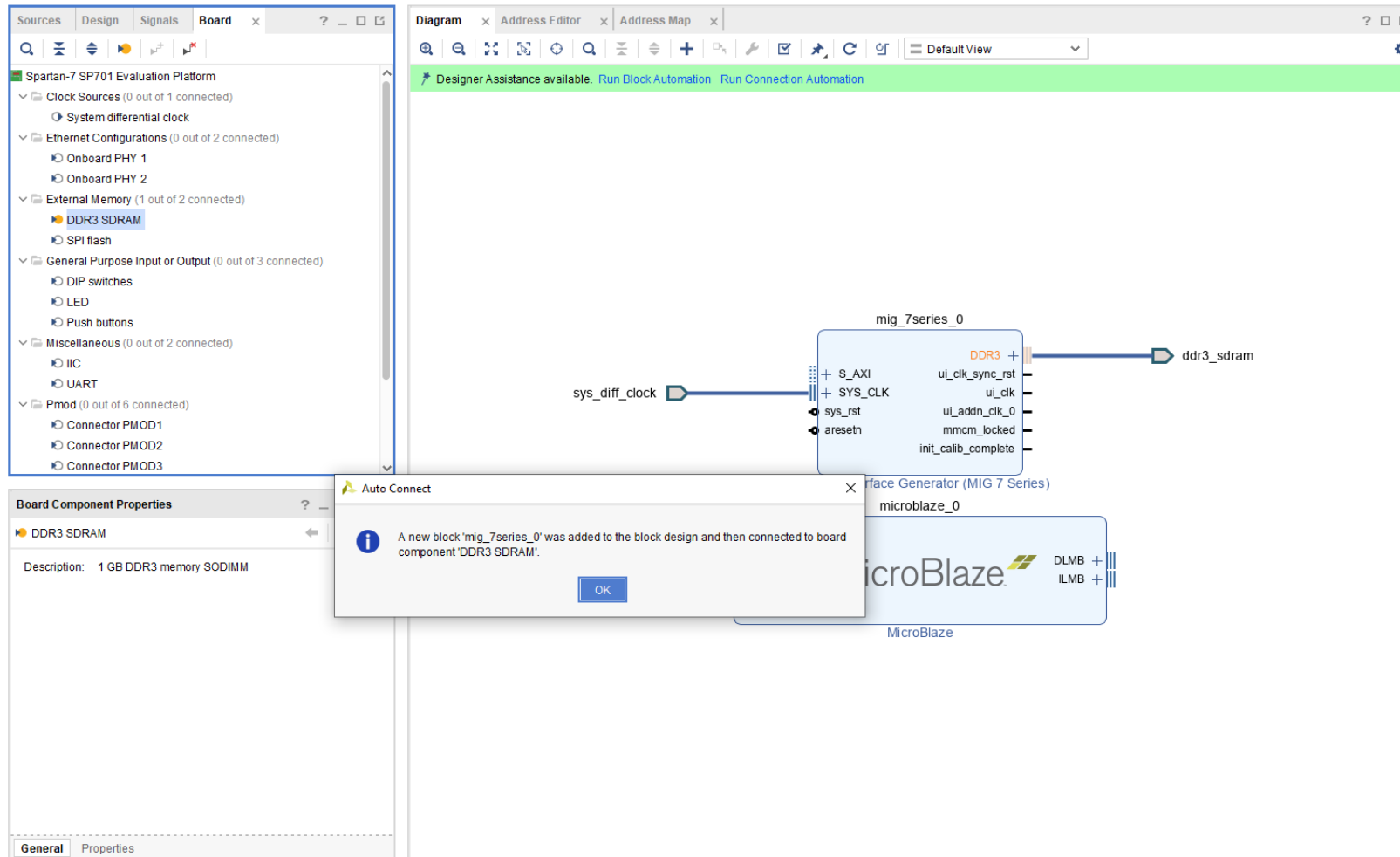
Lab: Mastering MicroBlaze

Click on the + in the block diagram and type in MicroBlaze



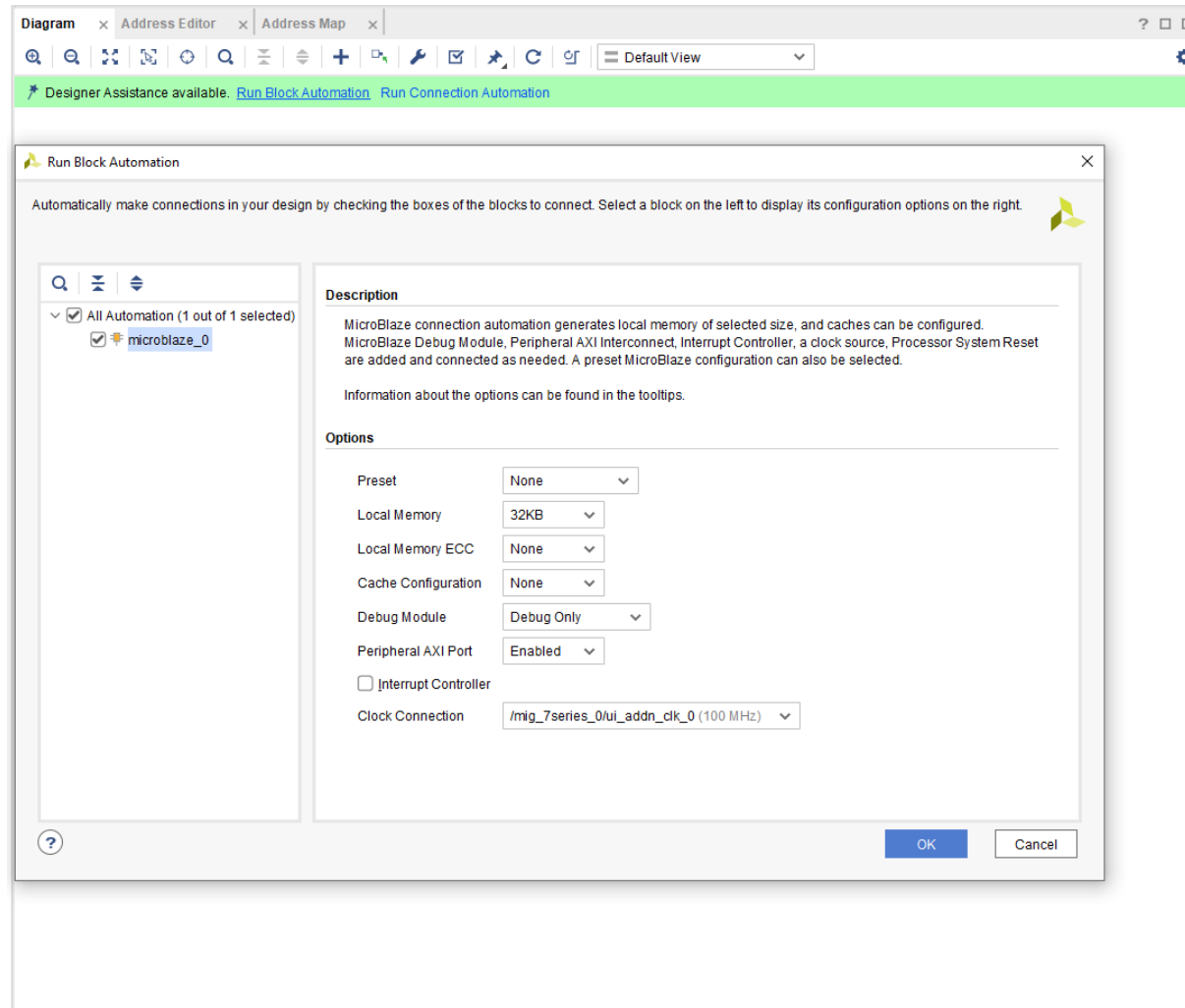
Lab: Mastering MicroBlaze

From the board tab select and drag the DDR3 onto the block diagram



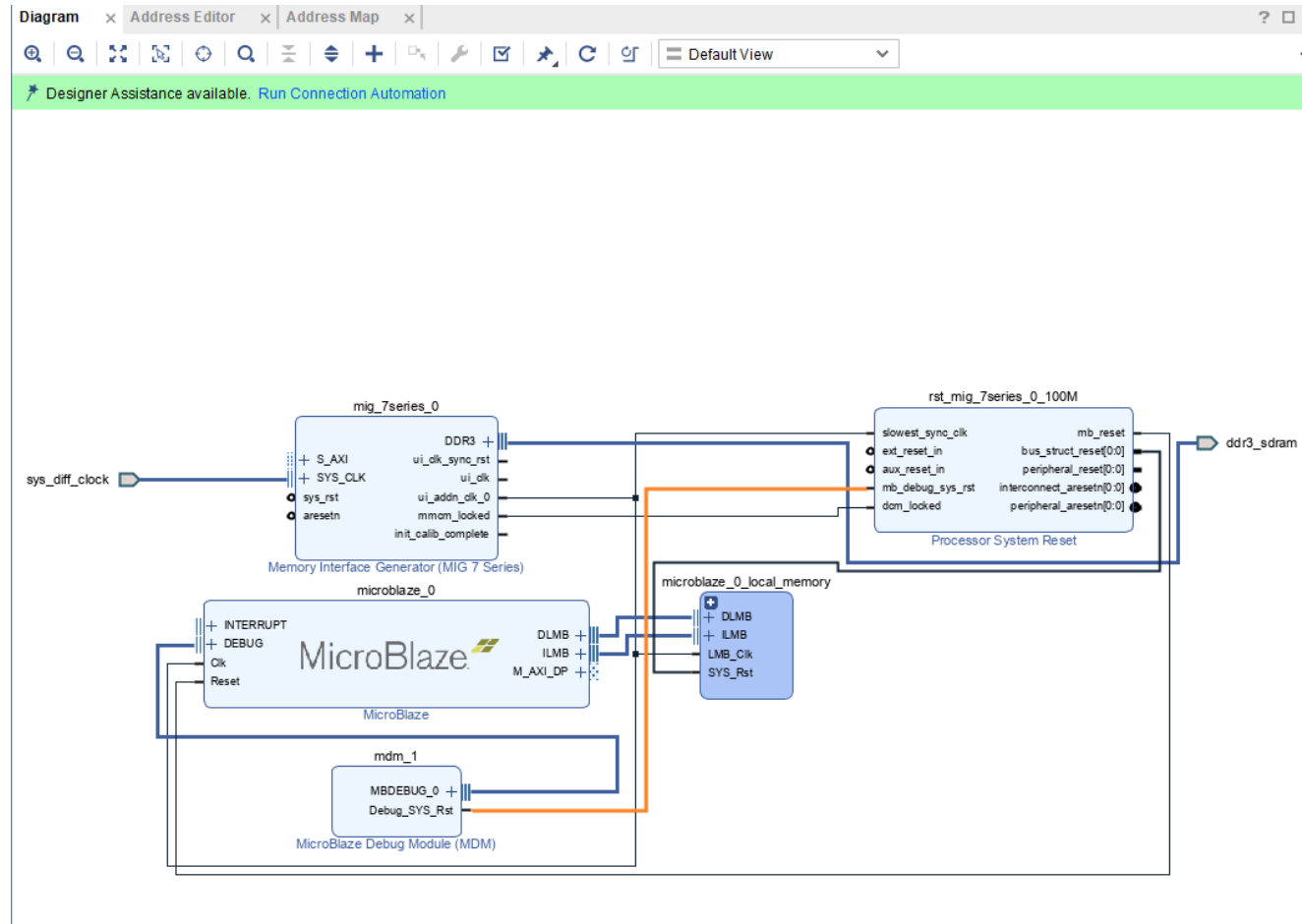
Lab: Mastering MicroBlaze

Run the Block Automation and configure 32KB of local memory and no cache



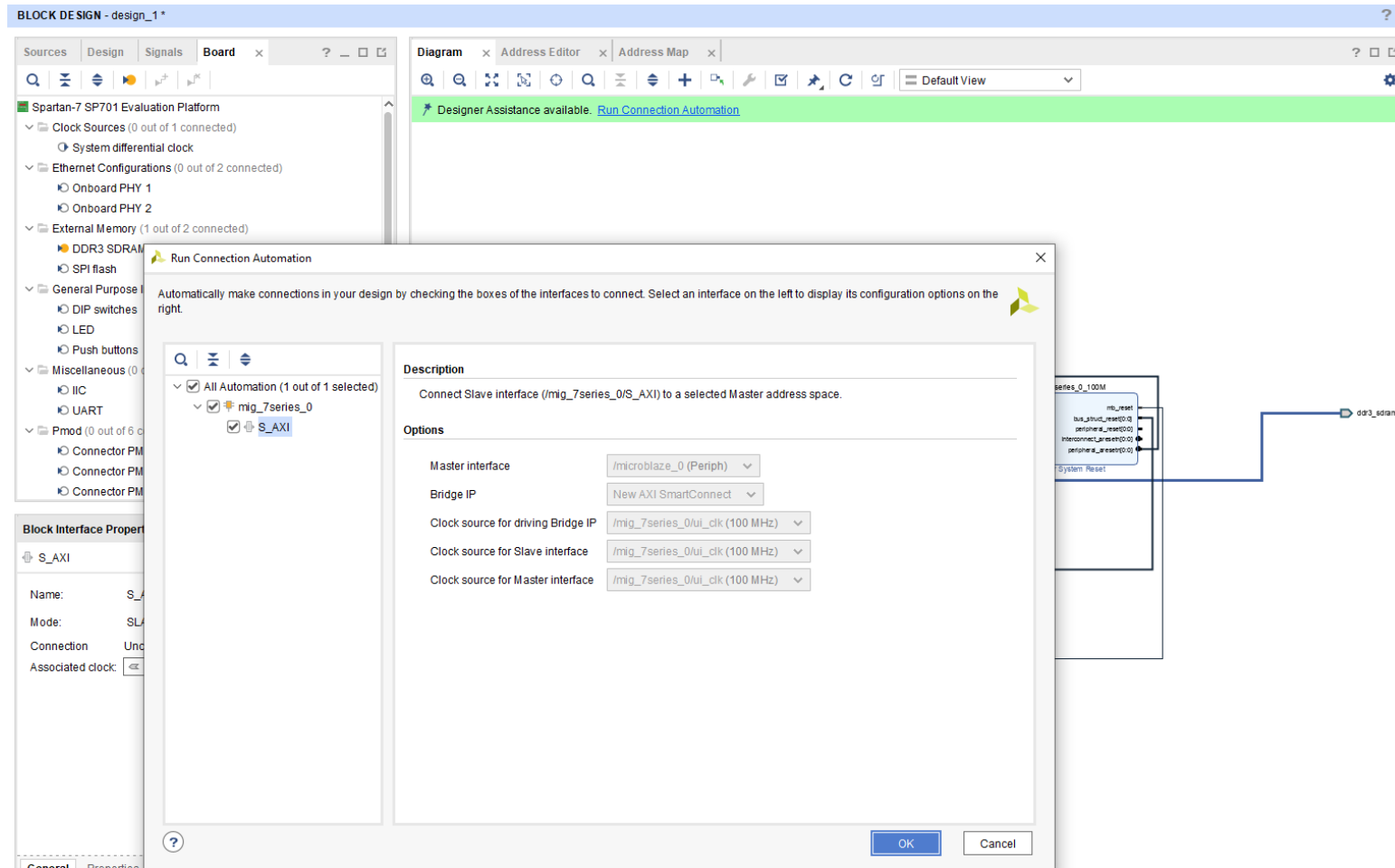
Lab: Mastering MicroBlaze

The result of the block automation should be as below



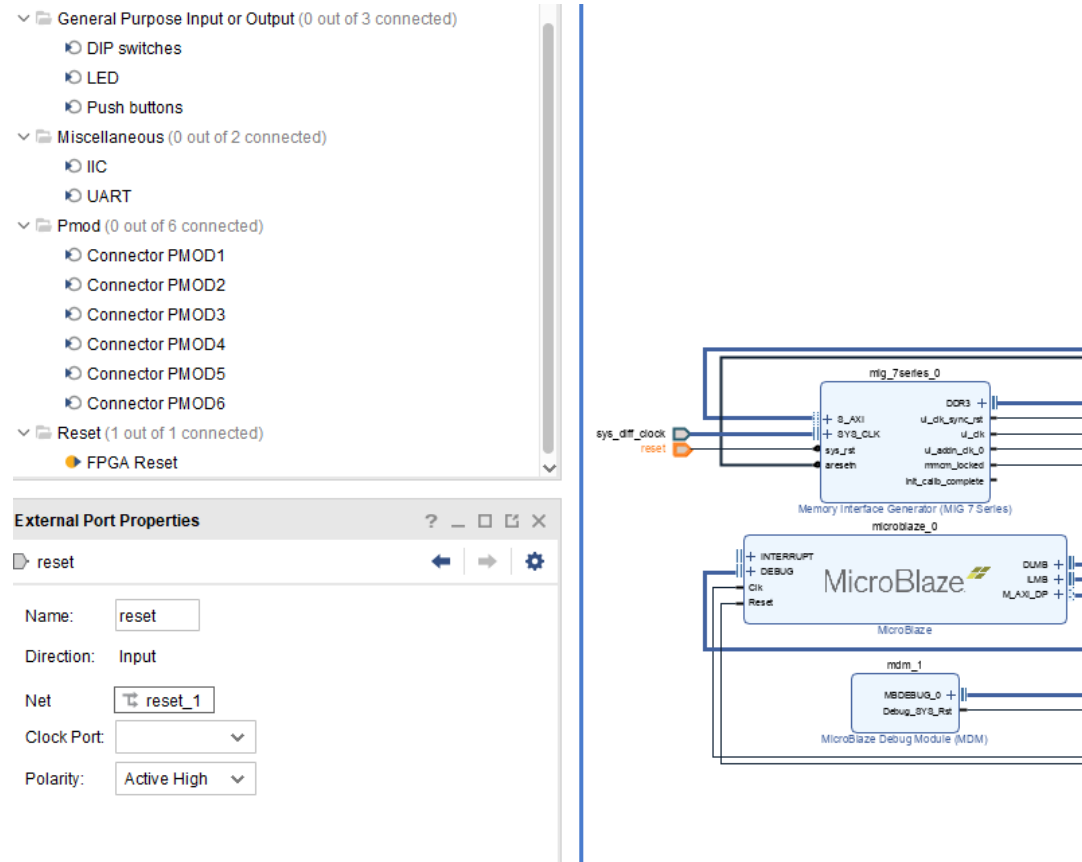
Lab: Mastering MicroBlaze

Run the Connection Automation to connect in the DDR3



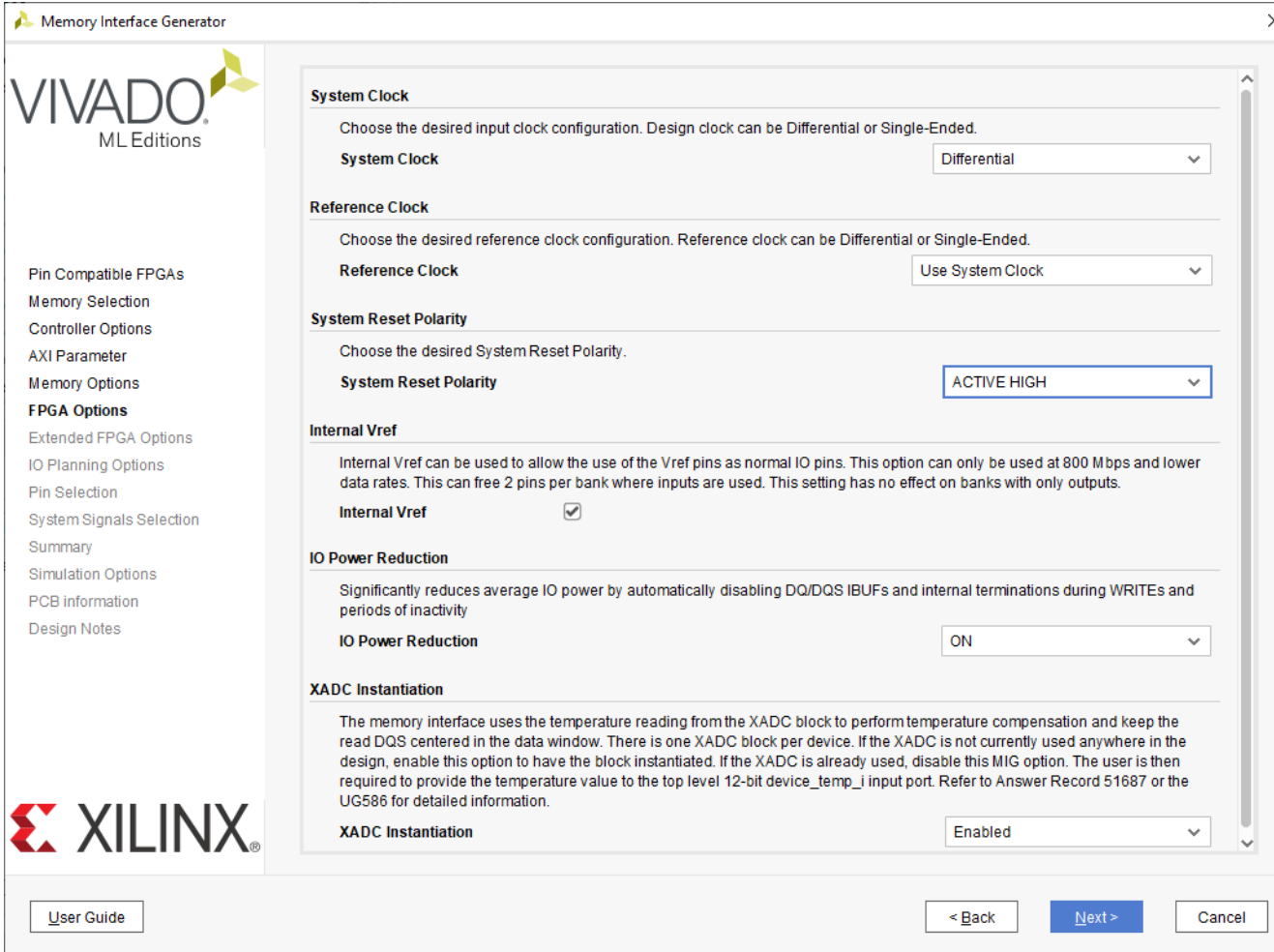
Lab: Mastering MicroBlaze

Drag and drop the reset port onto the diagram, select the port properties to determine its polarity



Lab: Mastering MicroBlaze

Re Customize the MIG and set the System Reset Polarity for Active High



Memory Interface Generator

VIVADO
ML Editions

Pin Compatible FPGAs
Memory Selection
Controller Options
AXI Parameter
Memory Options
FPGA Options
Extended FPGA Options
IO Planning Options
Pin Selection
System Signals Selection
Summary
Simulation Options
PCB Information
Design Notes

System Clock
Choose the desired input clock configuration. Design clock can be Differential or Single-Ended.
System Clock Differential

Reference Clock
Choose the desired reference clock configuration. Reference clock can be Differential or Single-Ended.
Reference Clock Use System Clock

System Reset Polarity
Choose the desired System Reset Polarity.
System Reset Polarity ACTIVE HIGH

Internal Vref
Internal Vref can be used to allow the use of the Vref pins as normal IO pins. This option can only be used at 800 Mbps and lower data rates. This can free 2 pins per bank where inputs are used. This setting has no effect on banks with only outputs.
Internal Vref ☒

IO Power Reduction
Significantly reduces average IO power by automatically disabling DQ/DQS IBUFs and internal terminations during WRITES and periods of inactivity.
IO Power Reduction ON

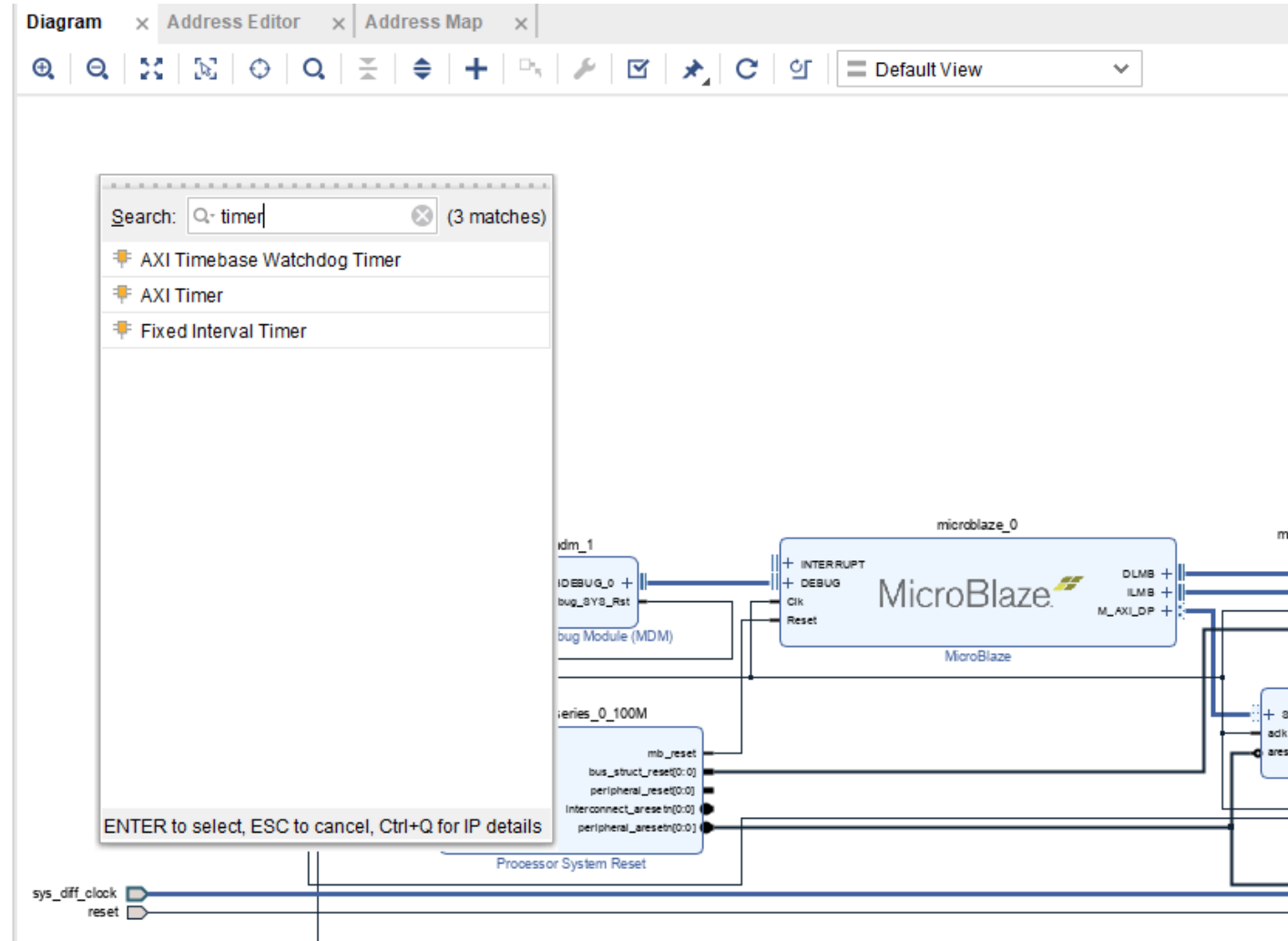
XADC Instantiation
The memory interface uses the temperature reading from the XADC block to perform temperature compensation and keep the read DQS centered in the data window. There is one XADC block per device. If the XADC is not currently used anywhere in the design, enable this option to have the block instantiated. If the XADC is already used, disable this MIG option. The user is then required to provide the temperature value to the top level 12-bit device_temp_i input port. Refer to Answer Record 51687 or the UG586 for detailed information.
XADC Instantiation Enabled

[User Guide](#) [< Back](#) [Next >](#) [Cancel](#)

XILINX

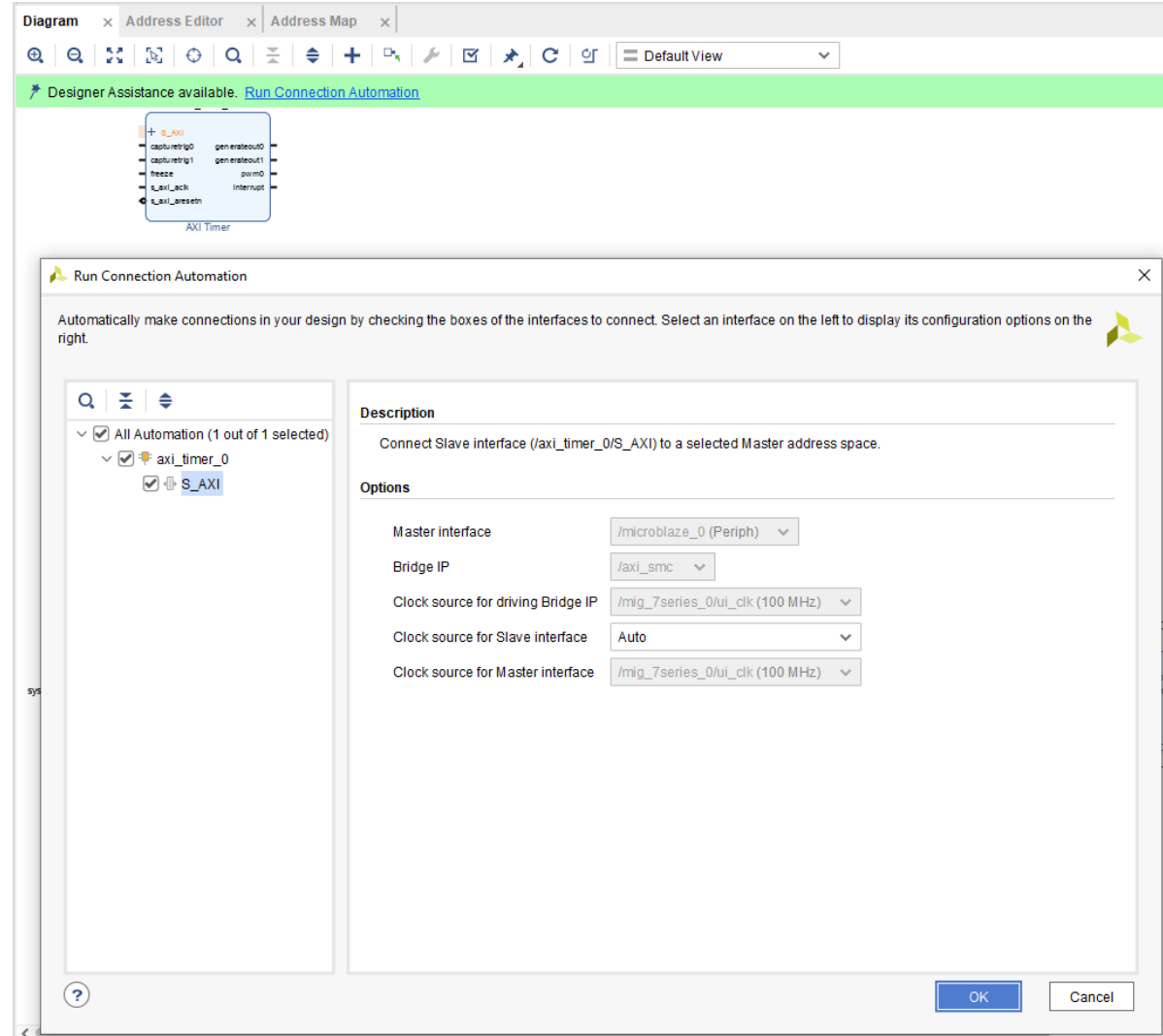
Lab: Mastering MicroBlaze

Add in a AXI Timer



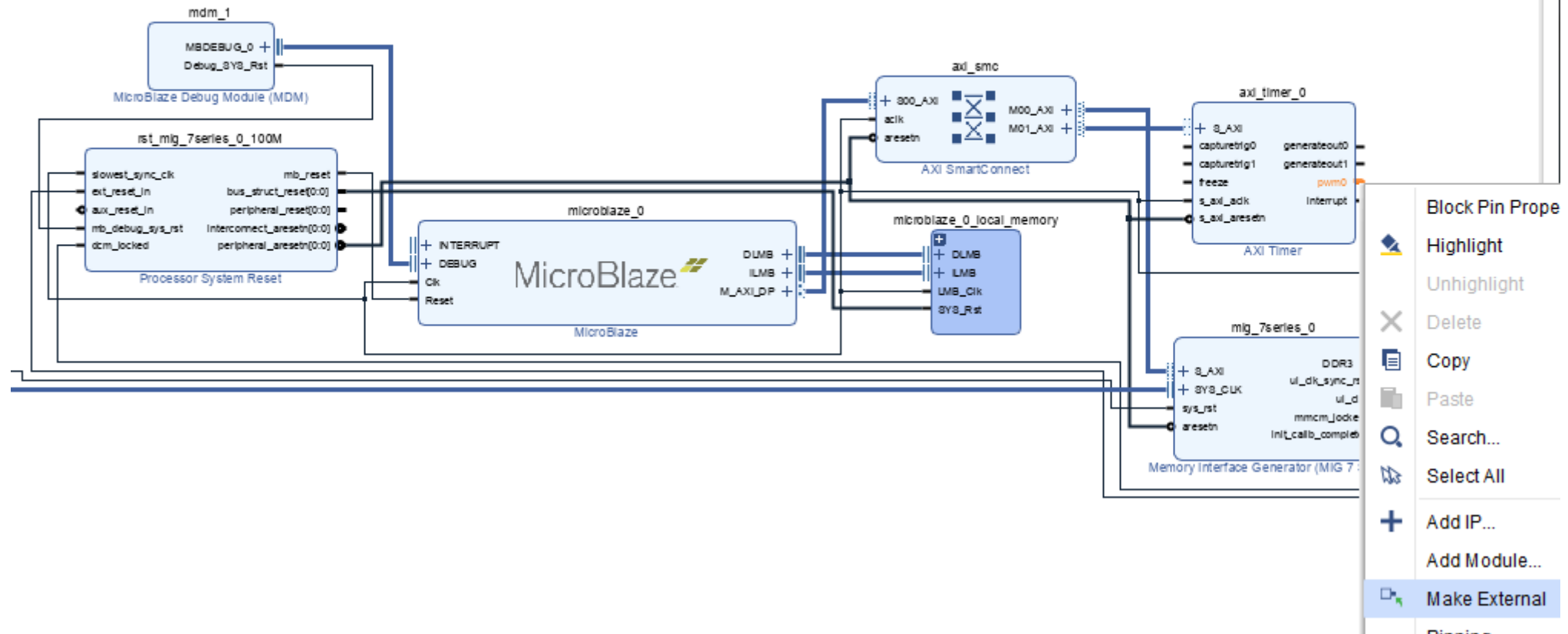
Lab: Mastering MicroBlaze

Run the Connection Automation



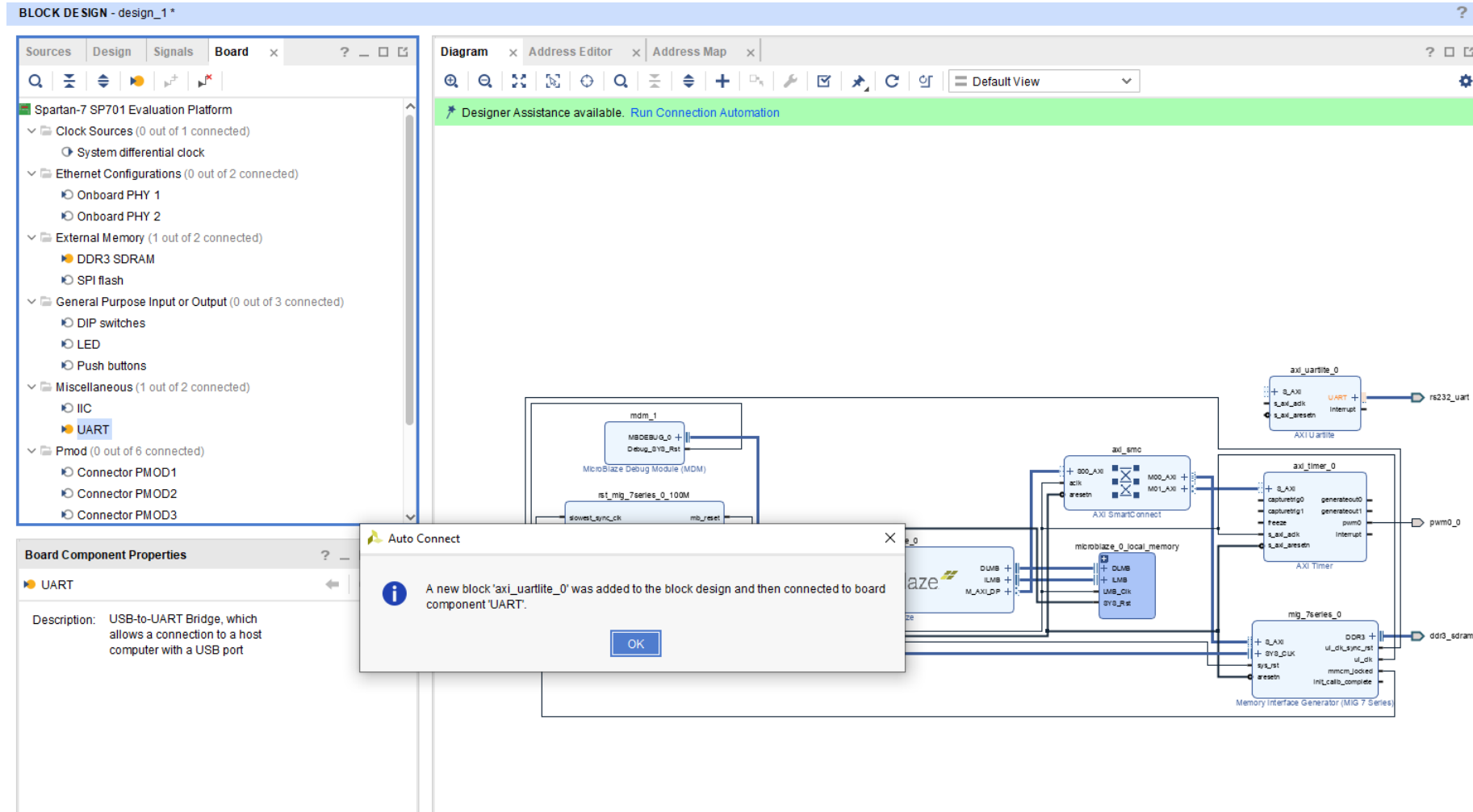
Lab: Mastering MicroBlaze

Select the PWM Pin and make it external



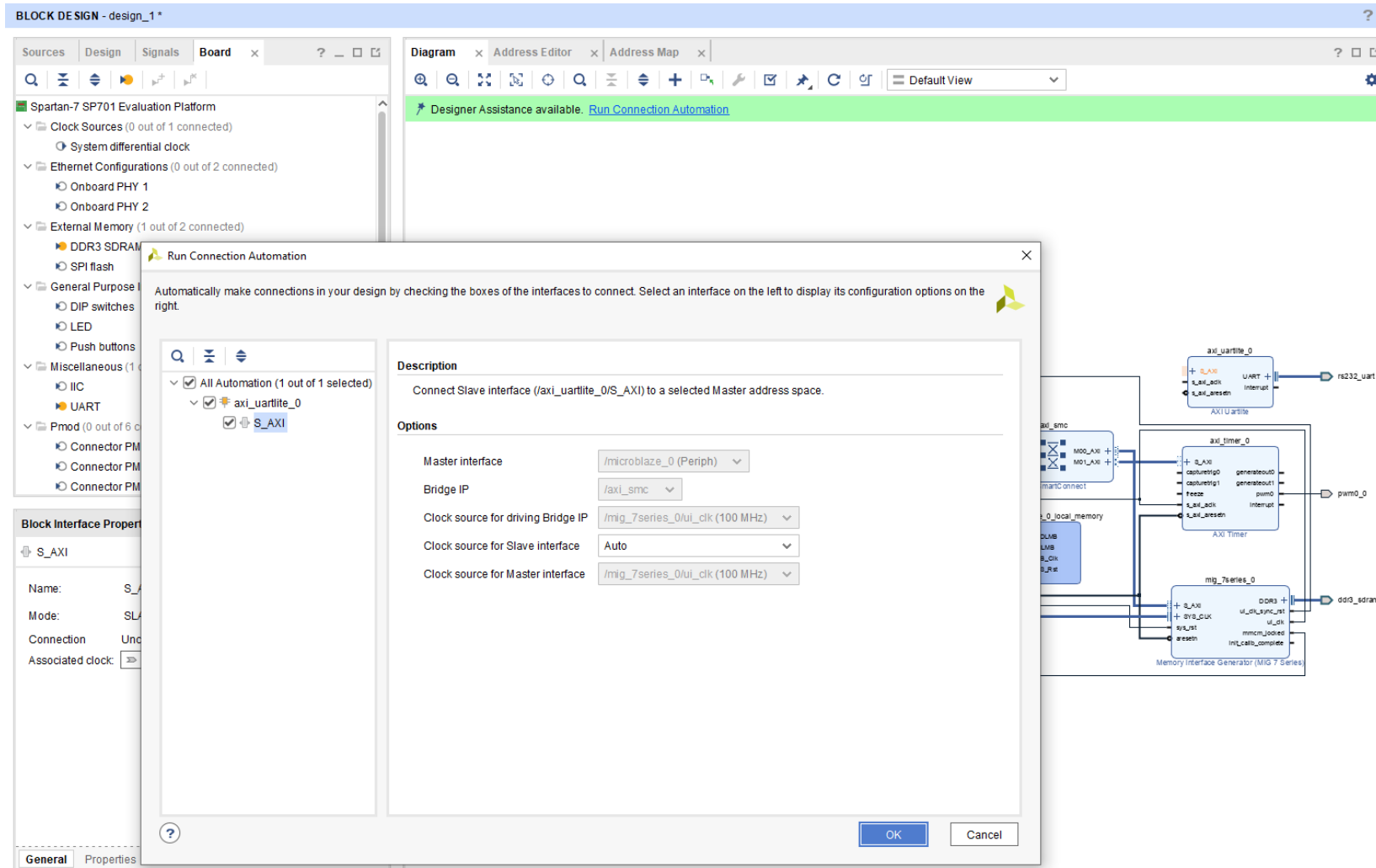
Lab: Mastering MicroBlaze

From the board menu drag and drop the UART onto the diagram



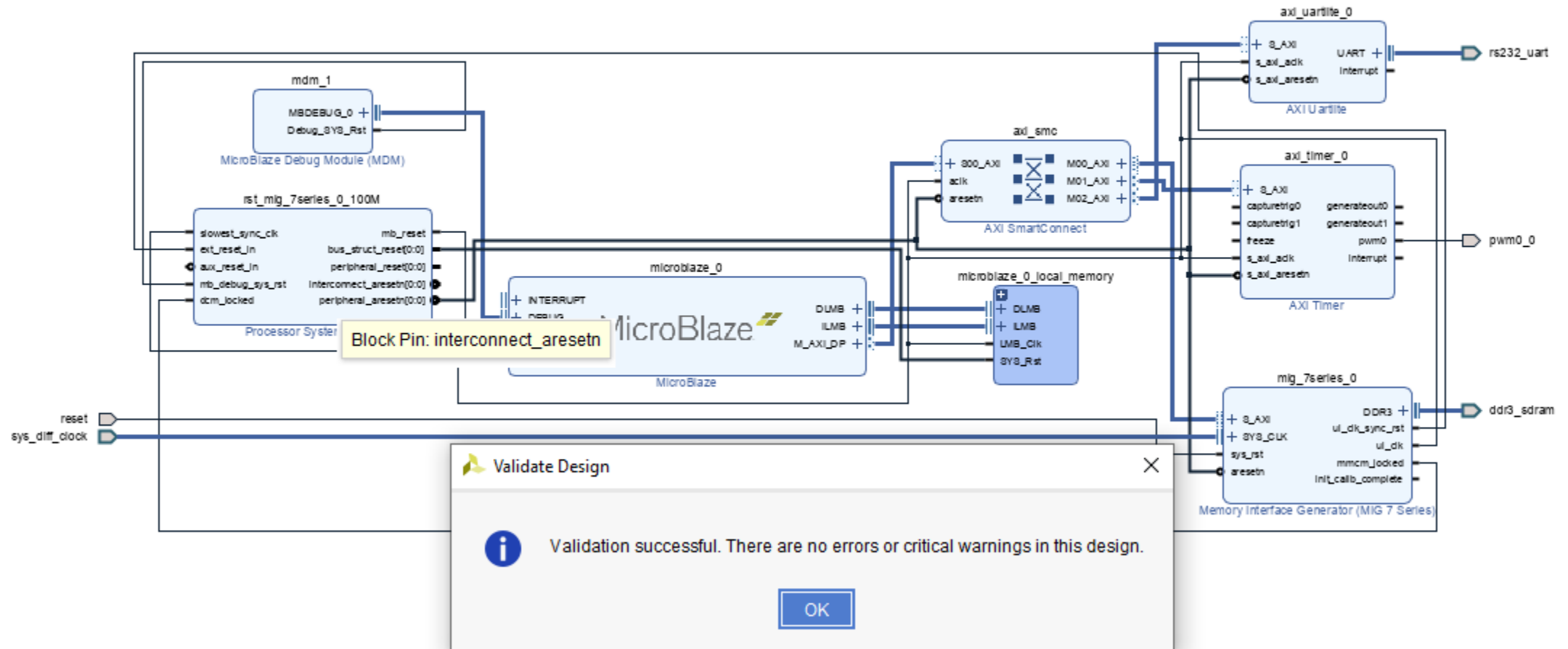
Lab: Mastering MicroBlaze

Run the connection automation



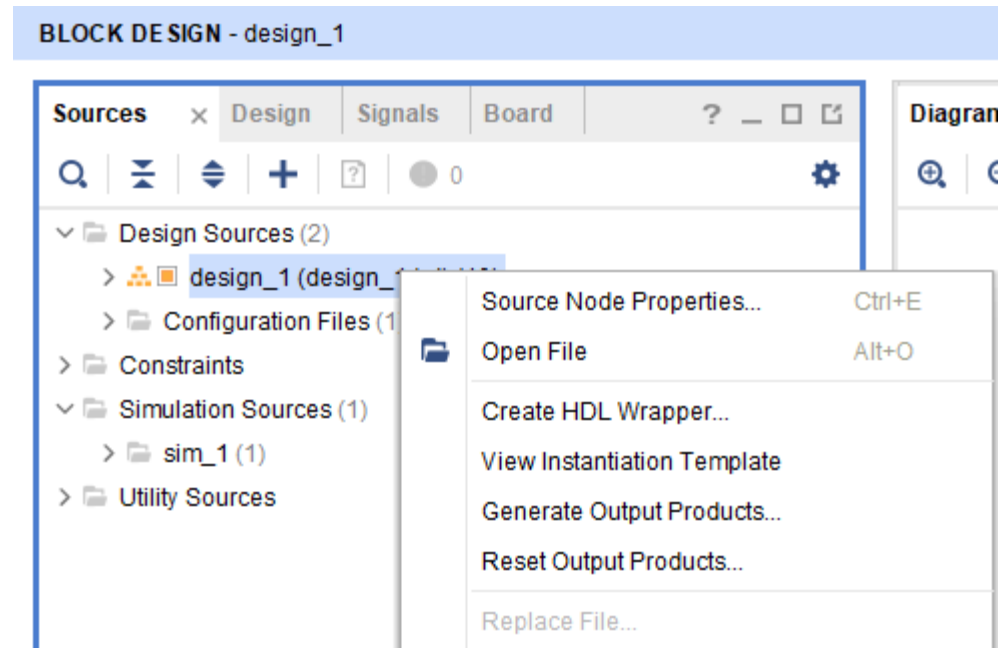
Lab: Mastering MicroBlaze

Validate the design = No errors or critical warnings should appear



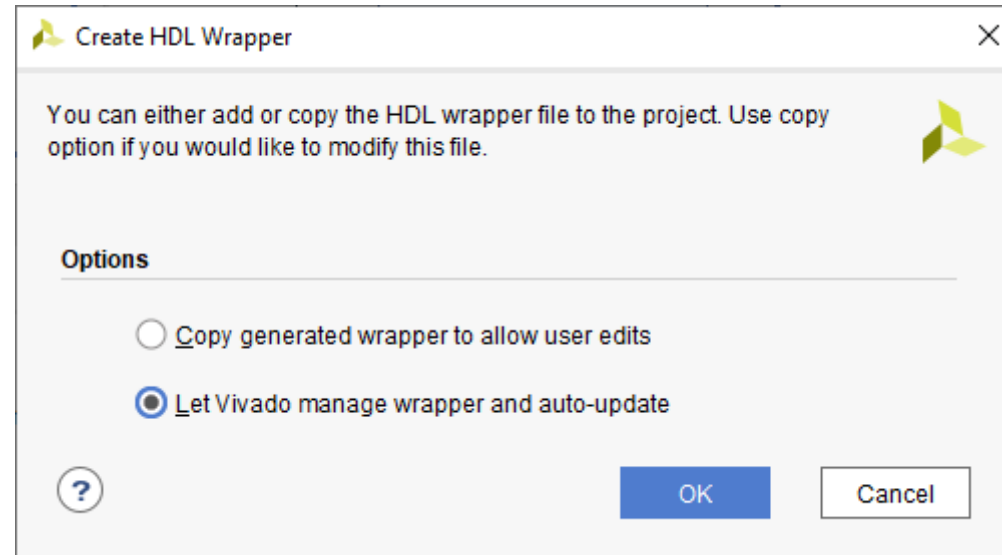
Lab: Mastering MicroBlaze

Create an HDL Wrapper



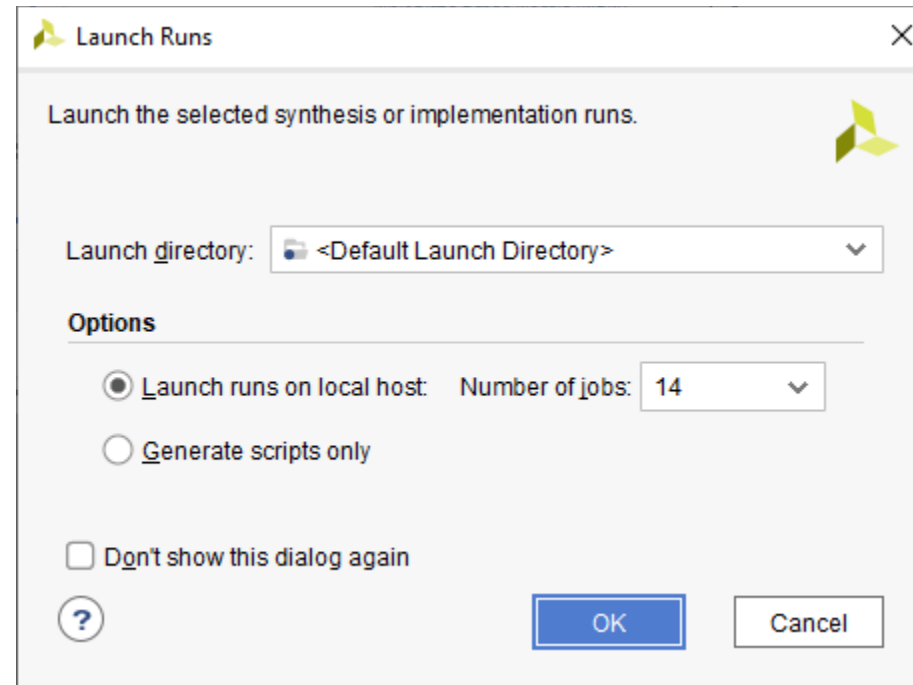
Lab: Mastering MicroBlaze

Let Vivado manage the wrapper



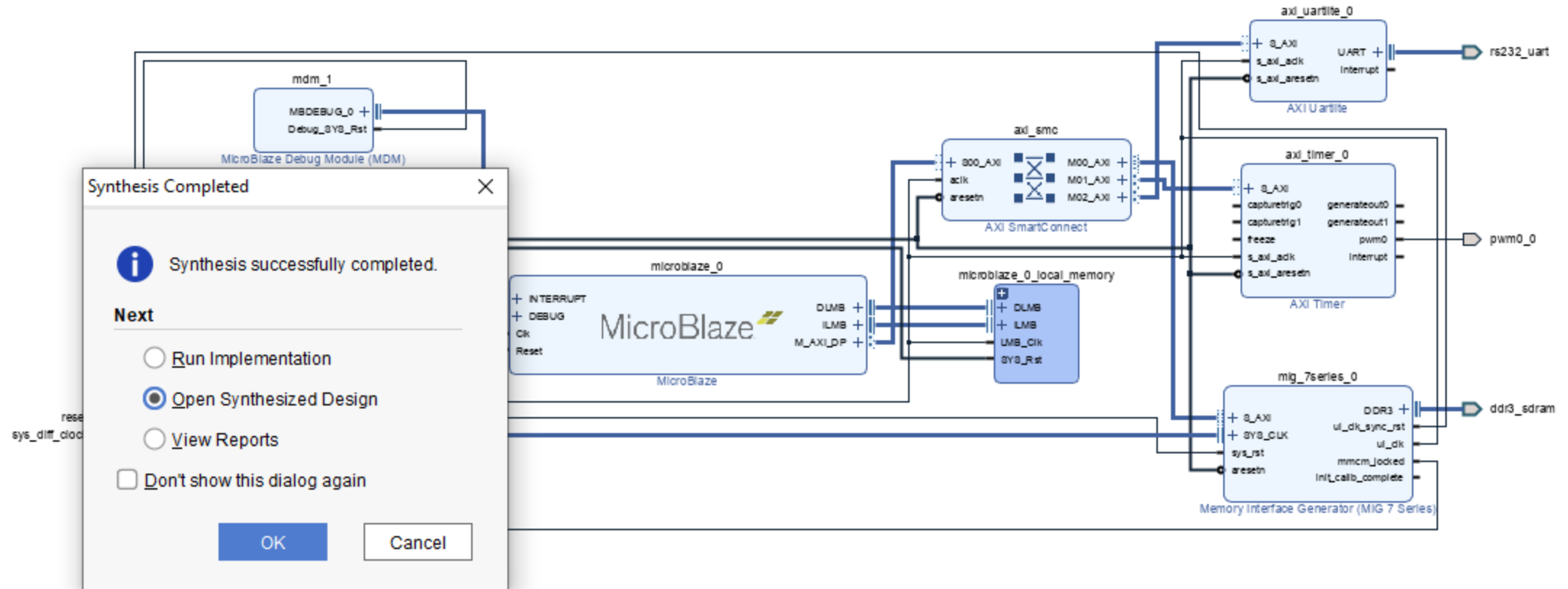
Lab: Mastering MicroBlaze

Synthesize the design



Lab: Mastering MicroBlaze

Once synthesis completes open the synthesized view



Lab: Mastering MicroBlaze

In the IO Ports view assign the PWM output to a pin connected to the LED

Tcl ConsoleMessagesLogReportsDesign RunsPackage PinsI/O Ports

Q

≡

⬆

⬇

+

⬆

?

—

□

⌵

NameDirectionNeg Diff PairPackage PinFixedBankI/O StdVccoVrefDrive StrengthSlew TypePull TypeOff-Chip TerminationIN_TERM

▼

🔍 All ports (52)

>

🔍 ddr3_sdram_54576 (48)

(Multiple)

☒

34

(Multiple)*

1.350

(Multiple)

FAST

NONE

FP_VTT_50

(Multiple)*

>

🔍 RST.RESET_54576 (1)

IN

☒

13

LVC MOS18

1.800

NONE

NONE

>

🔍 sys_diff_clock_54576 (2)

IN

☒

33

LVDS_25*

2.500

NONE

NONE

▼

🔍 Scalar ports (1)

🔍

pwm0_0

OUT

J25

☒

15

LVC MOS33*

3.300

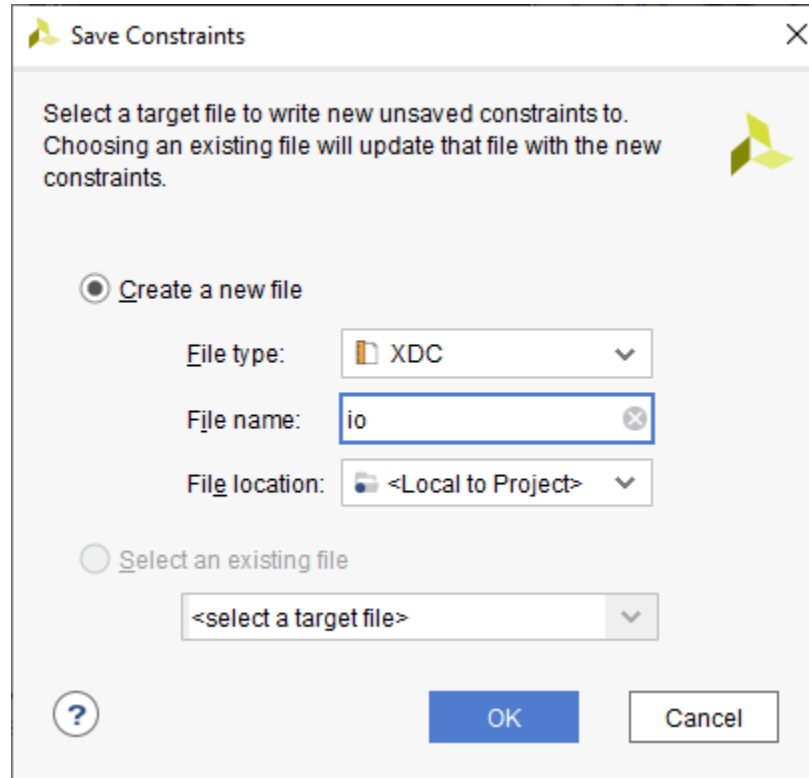
12

NONE

FP_VTT_50

Lab: Mastering MicroBlaze

Save the constraints



The image shows a 'Save Constraints' dialog box with a title bar containing a yellow triangle icon and a close button. The main text reads: 'Select a target file to write new unsaved constraints to. Choosing an existing file will update that file with the new constraints.' Below this, there are two radio button options. The first option, 'Create a new file', is selected. It has three sub-fields: 'File type:' with a dropdown menu showing 'XDC', 'File name:' with a text box containing 'io', and 'File location:' with a dropdown menu showing '<Local to Project>'. The second option, 'Select an existing file', is unselected and has a dropdown menu showing '<select a target file>'. At the bottom left is a help icon (question mark in a circle). At the bottom right are 'OK' and 'Cancel' buttons.

Save Constraints

Select a target file to write new unsaved constraints to.
Choosing an existing file will update that file with the new constraints.

☒ Create a new file

File type: XDC

File name: io

File location: <Local to Project>

☐ Select an existing file

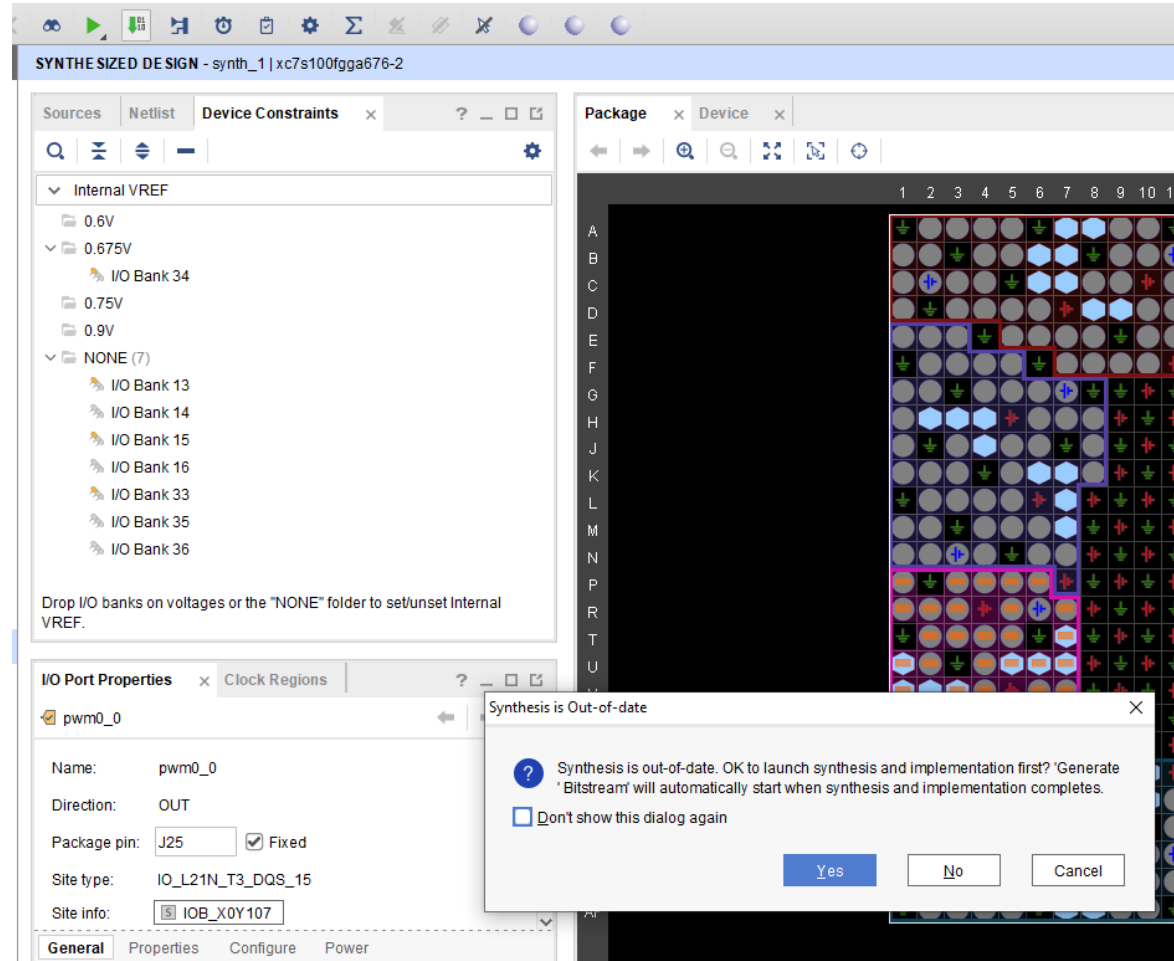
<select a target file>

?

OK Cancel

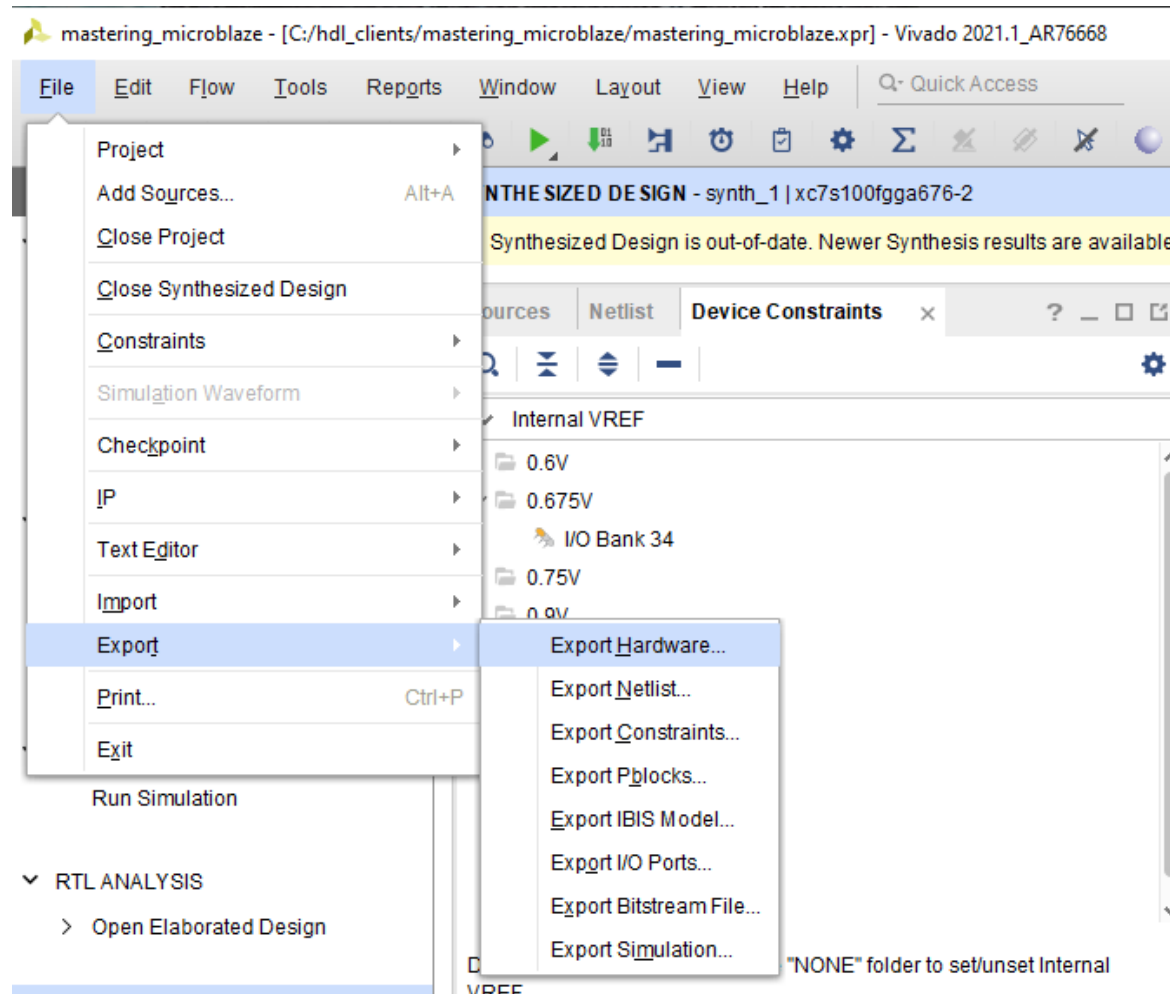
Lab: Mastering MicroBlaze

Generate the bitstream – if you see an out-of-date warning click yes



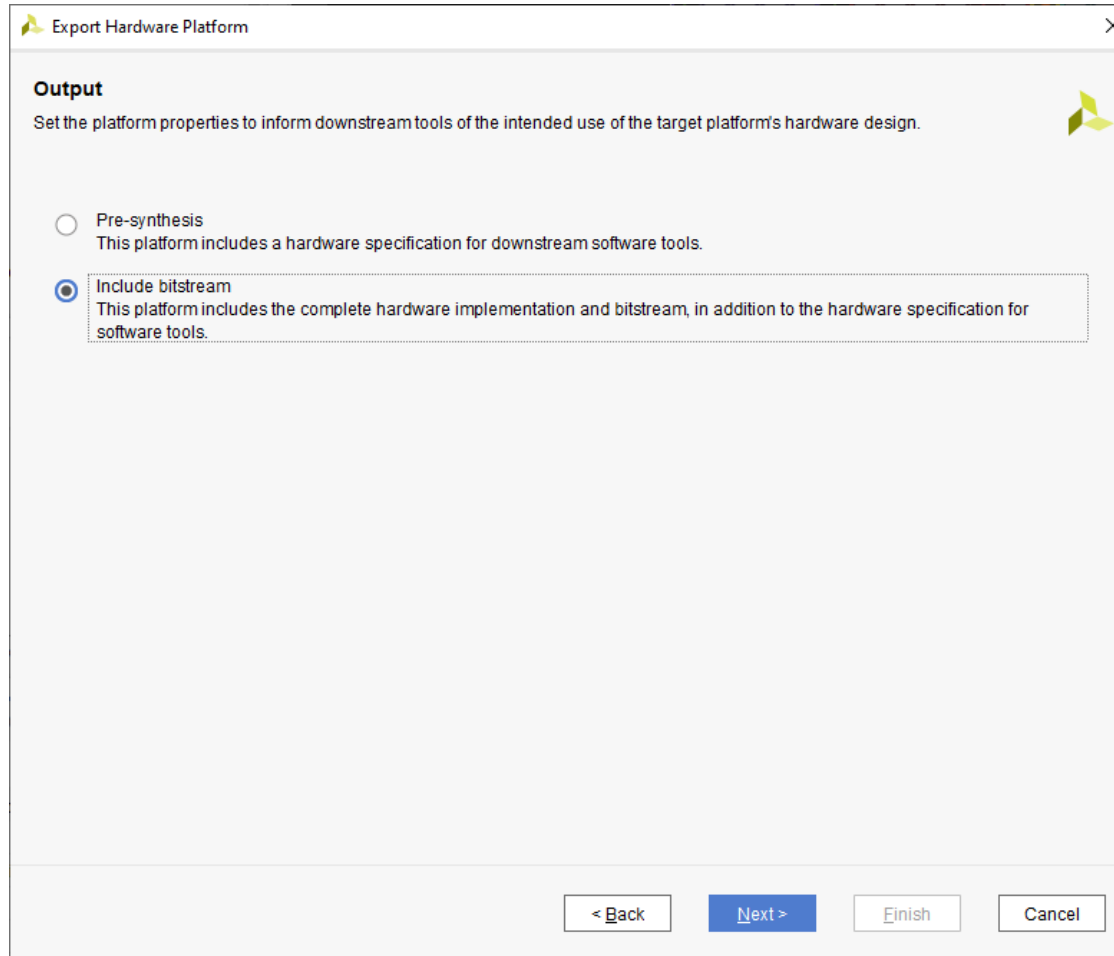
Lab: Mastering MicroBlaze

Once the bit stream is completed, export the hardware definition to Vitis



Lab: Mastering MicroBlaze

Export the Hardware with the bitstream



Export Hardware Platform

Output
Set the platform properties to inform downstream tools of the intended use of the target platform's hardware design.

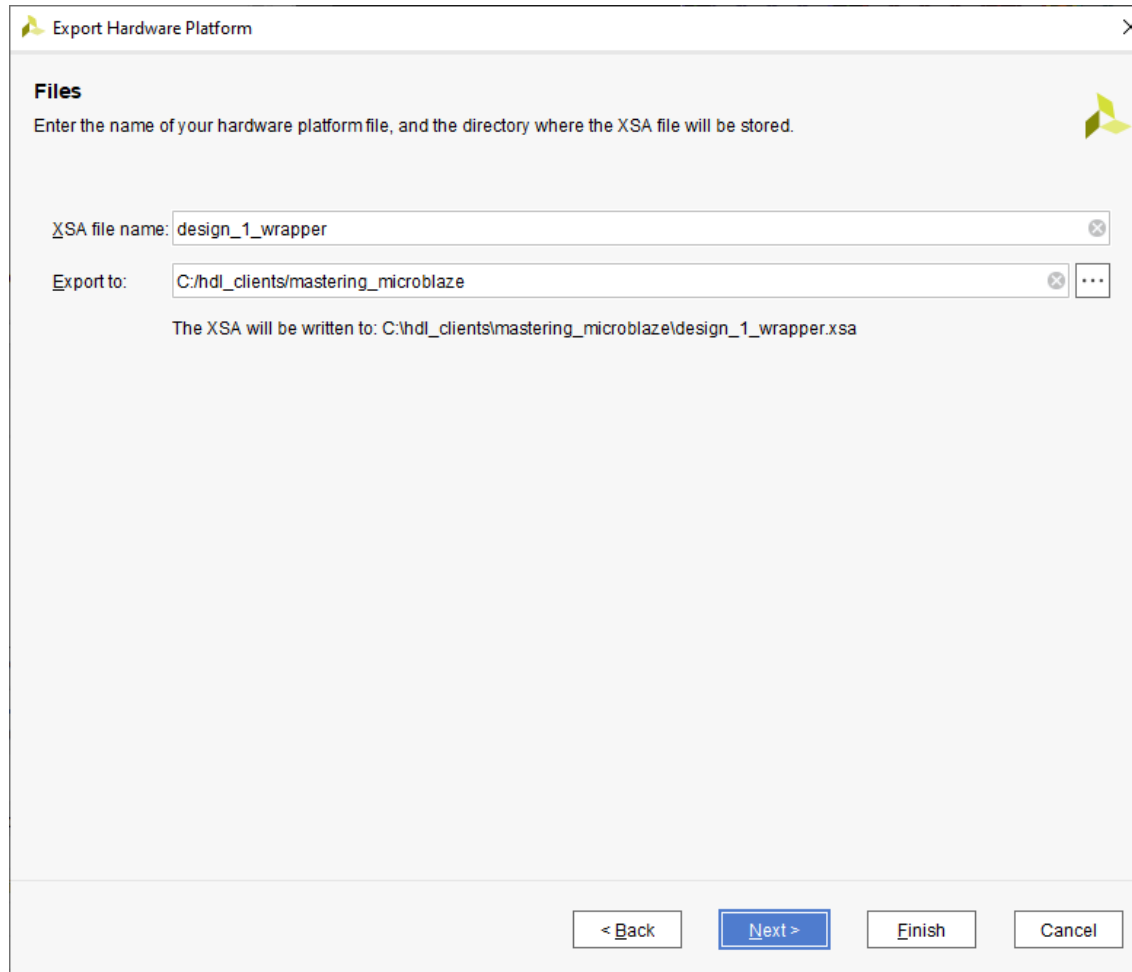
☐ Pre-synthesis
This platform includes a hardware specification for downstream software tools.

☒ Include bitstream
This platform includes the complete hardware implementation and bitstream, in addition to the hardware specification for software tools.

< Back Next > Finish Cancel

Lab: Mastering MicroBlaze

Leave the file name and location unchanged



The image shows a screenshot of the 'Export Hardware Platform' dialog box. The dialog has a title bar with a yellow gear icon and the text 'Export Hardware Platform'. Below the title bar, there is a section titled 'Files' with a yellow gear icon. The text 'Enter the name of your hardware platform file, and the directory where the XSA file will be stored.' is displayed. There are two input fields: 'XSA file name:' with the value 'design_1_wrapper' and 'Export to:' with the value 'C:/hdl_clients/mastering_microblaze'. Below these fields, it says 'The XSA will be written to: C:/hdl_clients/mastering_microblaze/design_1_wrapper.xsa'. At the bottom, there are four buttons: '< Back', 'Next >', 'Finish', and 'Cancel'.

Export Hardware Platform

Files
Enter the name of your hardware platform file, and the directory where the XSA file will be stored.

XSA file name: design_1_wrapper

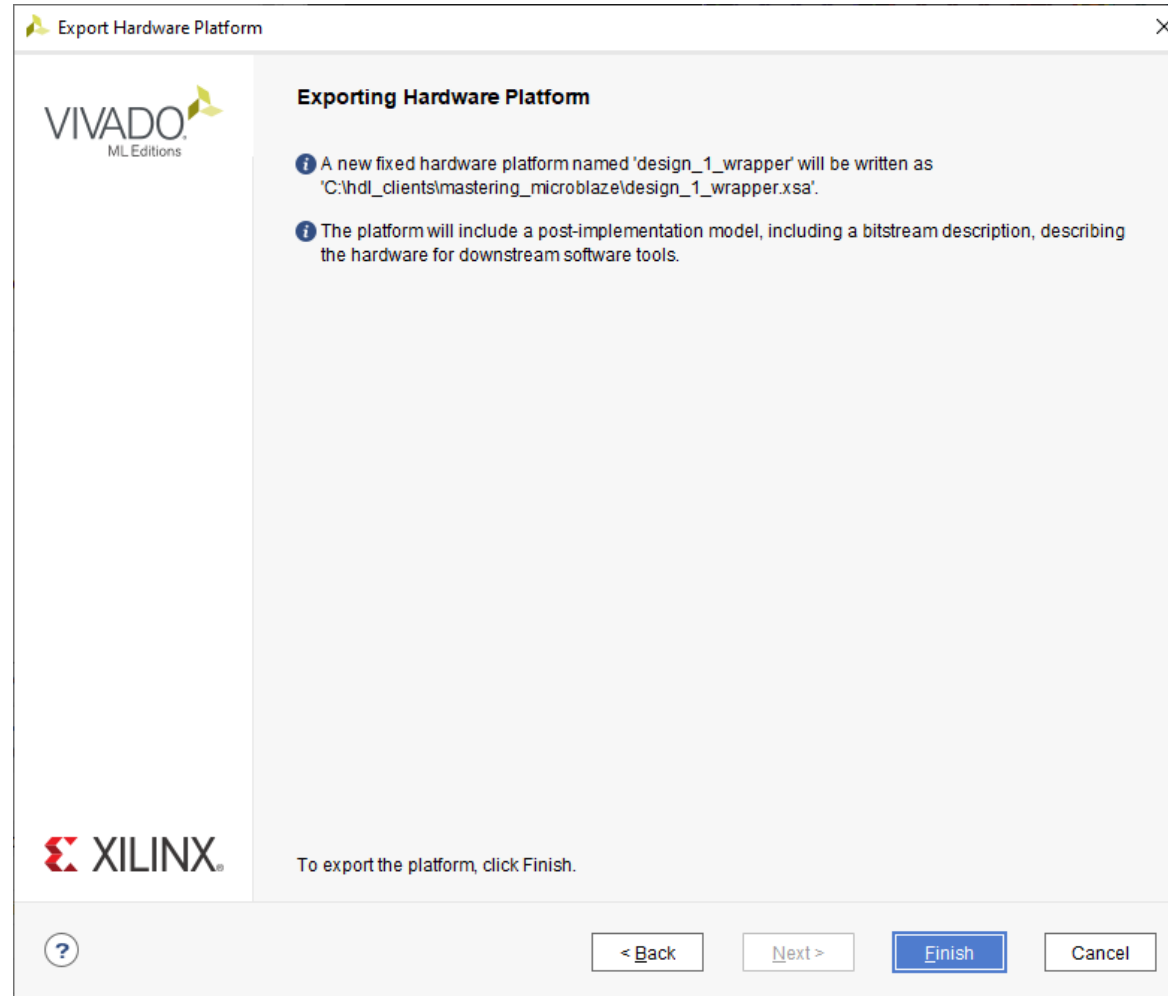
Export to: C:/hdl_clients/mastering_microblaze

The XSA will be written to: C:/hdl_clients/mastering_microblaze/design_1_wrapper.xsa

< Back Next > Finish Cancel

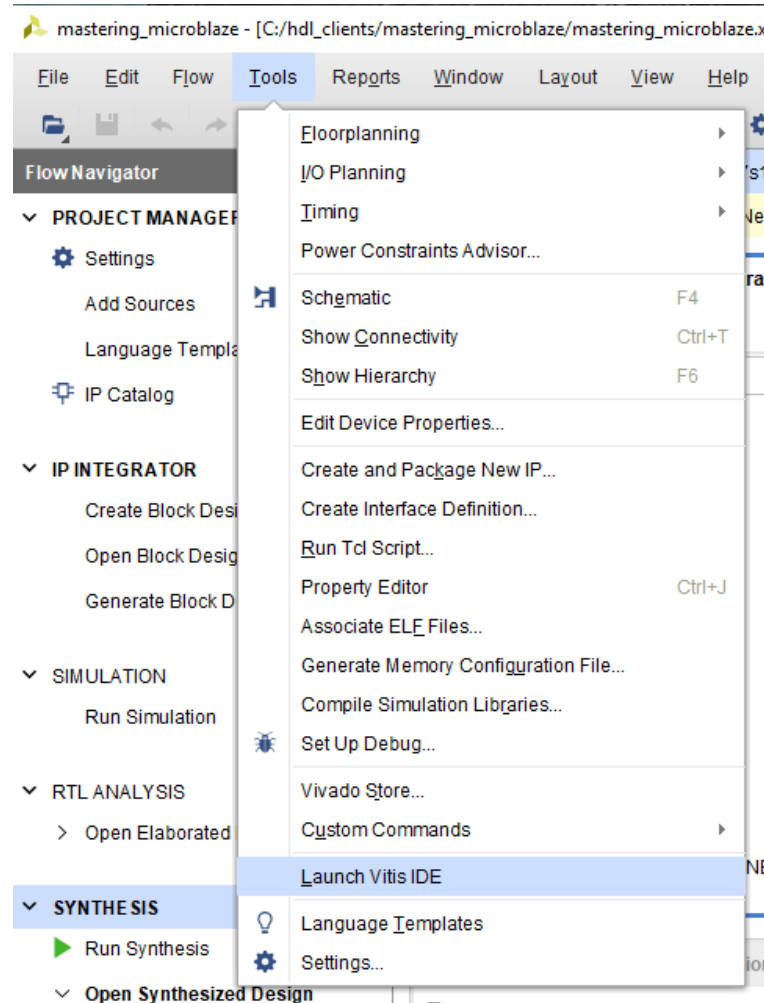
Lab: Mastering MicroBlaze

Click Finish



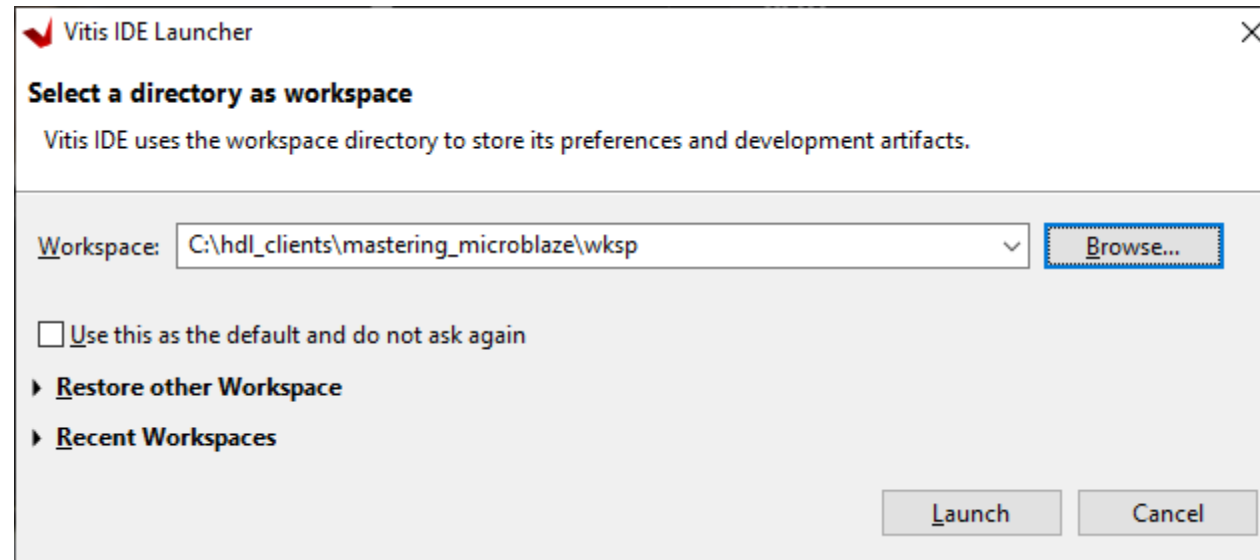
Lab: Mastering MicroBlaze

From the tools menu select Launch Vitis IDE



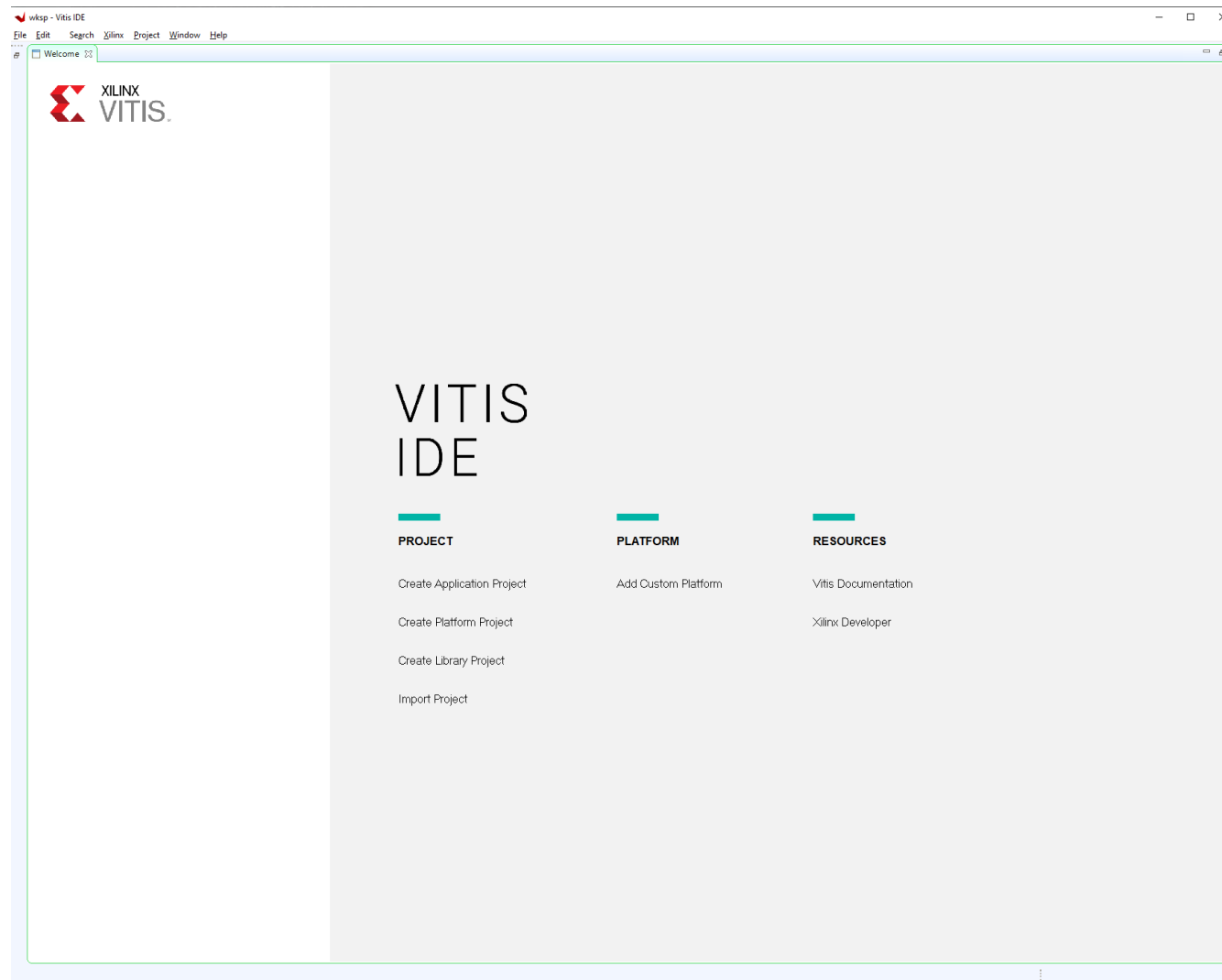
Lab: Mastering MicroBlaze

Select the workspace – this is where all the SW files and projects will be contained



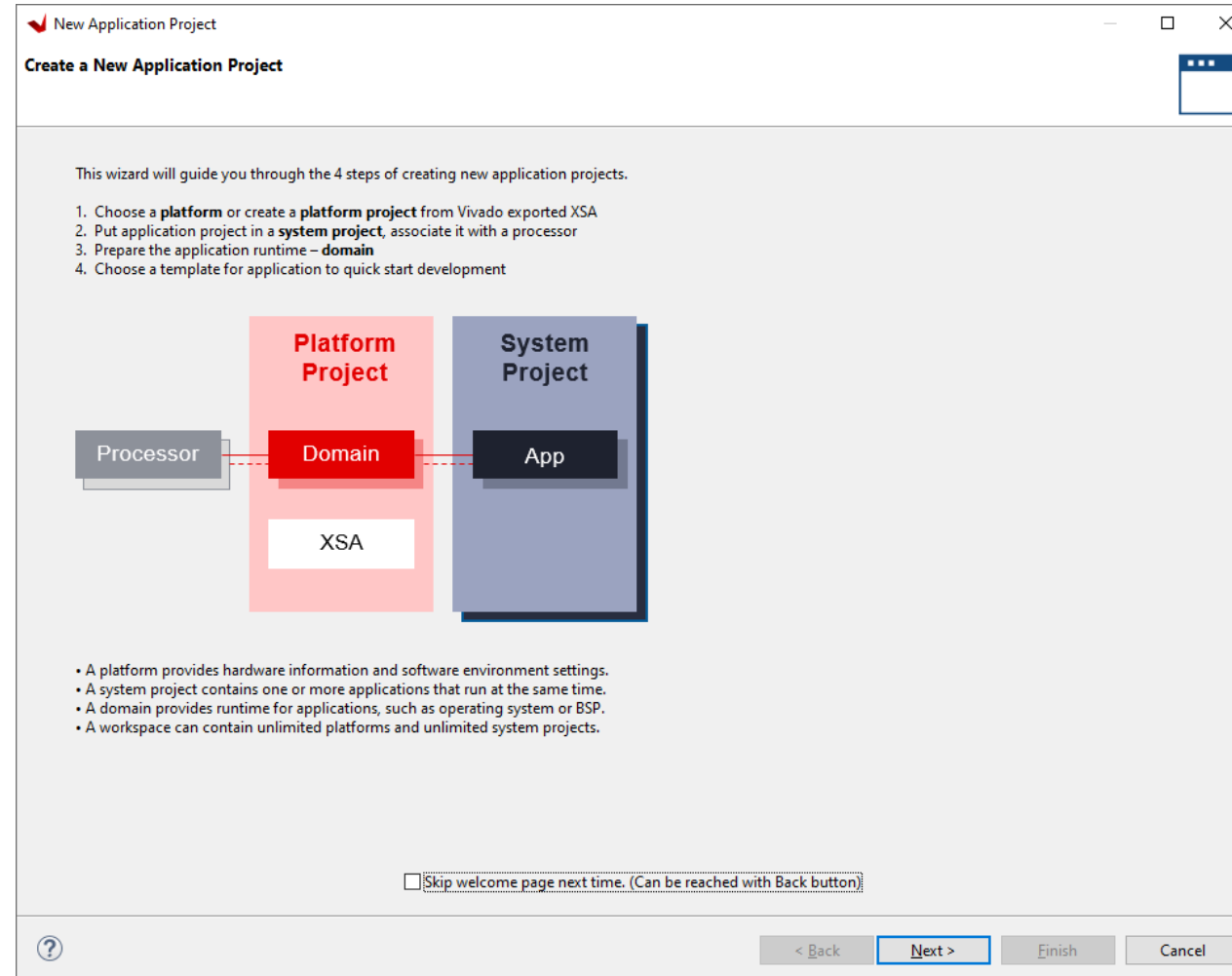
Lab: Mastering MicroBlaze

Once Vitis is Open, from the menu screen select Create-Application Project



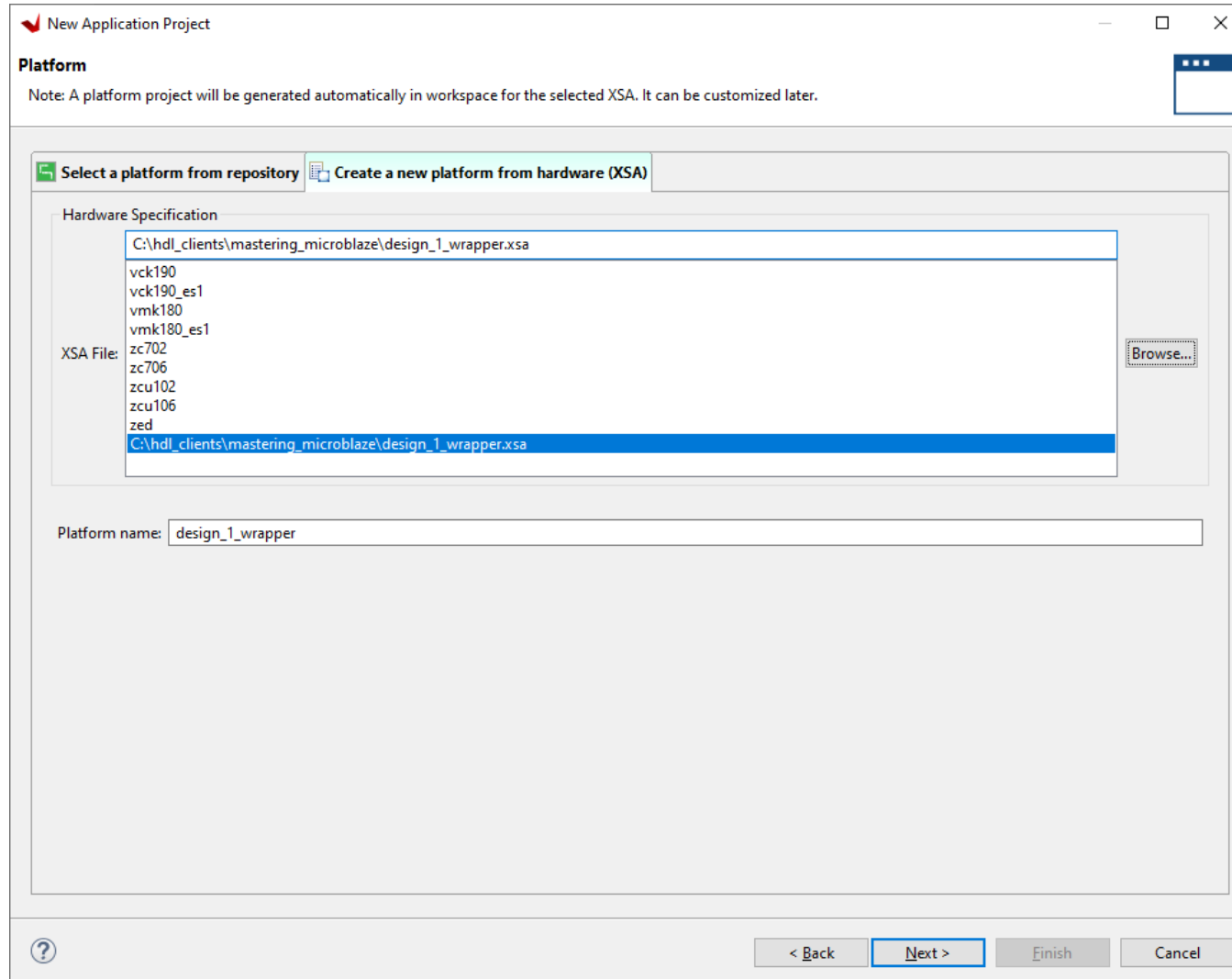
Lab: Mastering MicroBlaze

Click Next



Lab: Mastering MicroBlaze

Select the XSA just exported from Vivado



New Application Project

Platform

Note: A platform project will be generated automatically in workspace for the selected XSA. It can be customized later.

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

Hardware Specification

XSA File:

- C:\hdl_clients\mastering_microblaze\design_1_wrapper.xsa
- vck190
- vck190_es1
- vmk180
- vmk180_es1
- zc702
- zc706
- zcu102
- zcu106
- zed
- C:\hdl_clients\mastering_microblaze\design_1_wrapper.xsa

Browse...

Platform name: design_1_wrapper

< Back Next > Finish Cancel

Enter a project name

New Application Project

Application Project Details

Specify the application project name and its system project properties

Application project name:

application

System Project

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

Select a system project

+ Create new...

System project details

System project name: application_system

Target processor

Select target processor for the Application project.

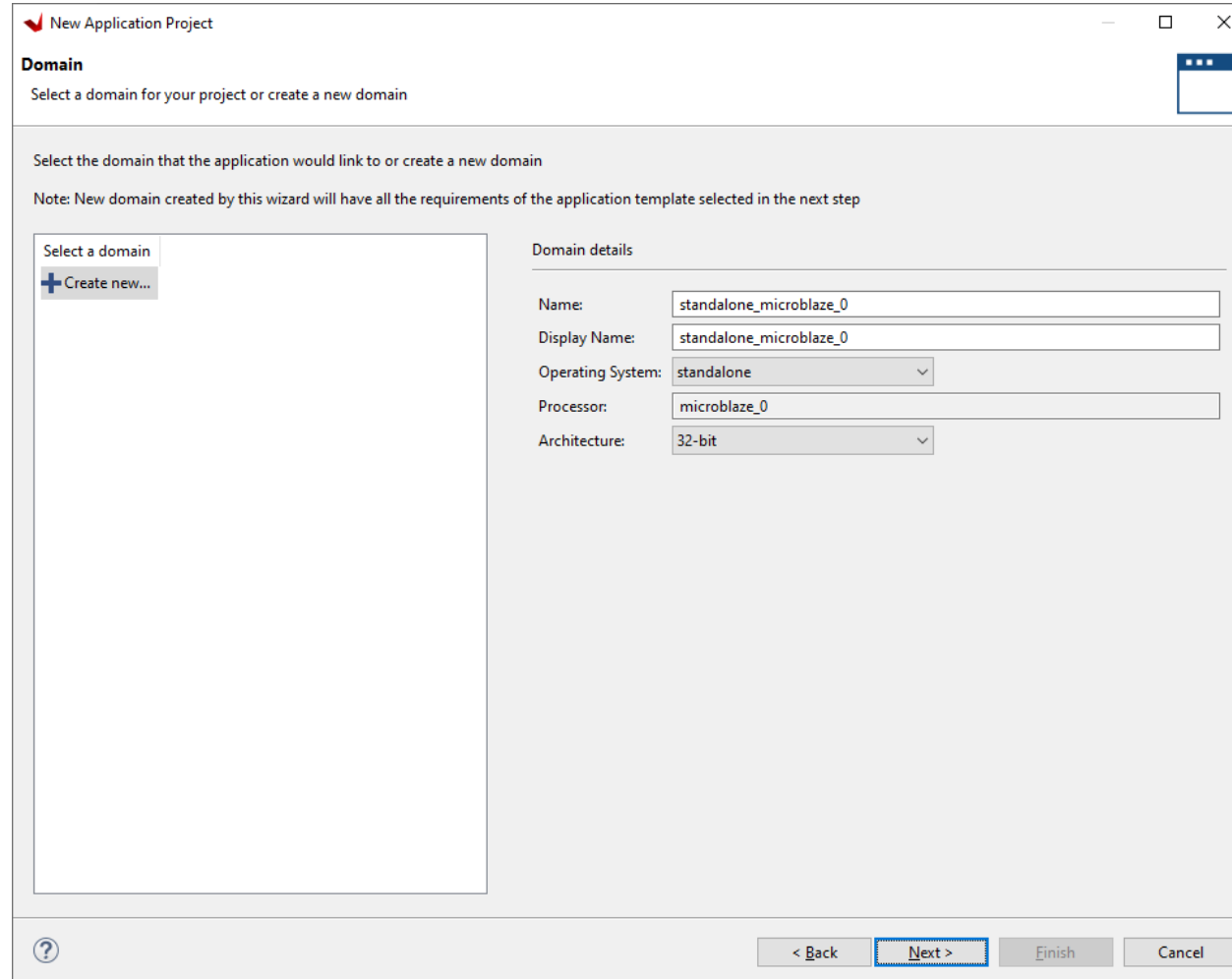
Processor	Associated applications	
microblaze_0	application	

Show all processors in the hardware specification

? < Back Next > Finish Cancel

Lab: Mastering MicroBlaze

Leave the domain unchanged



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

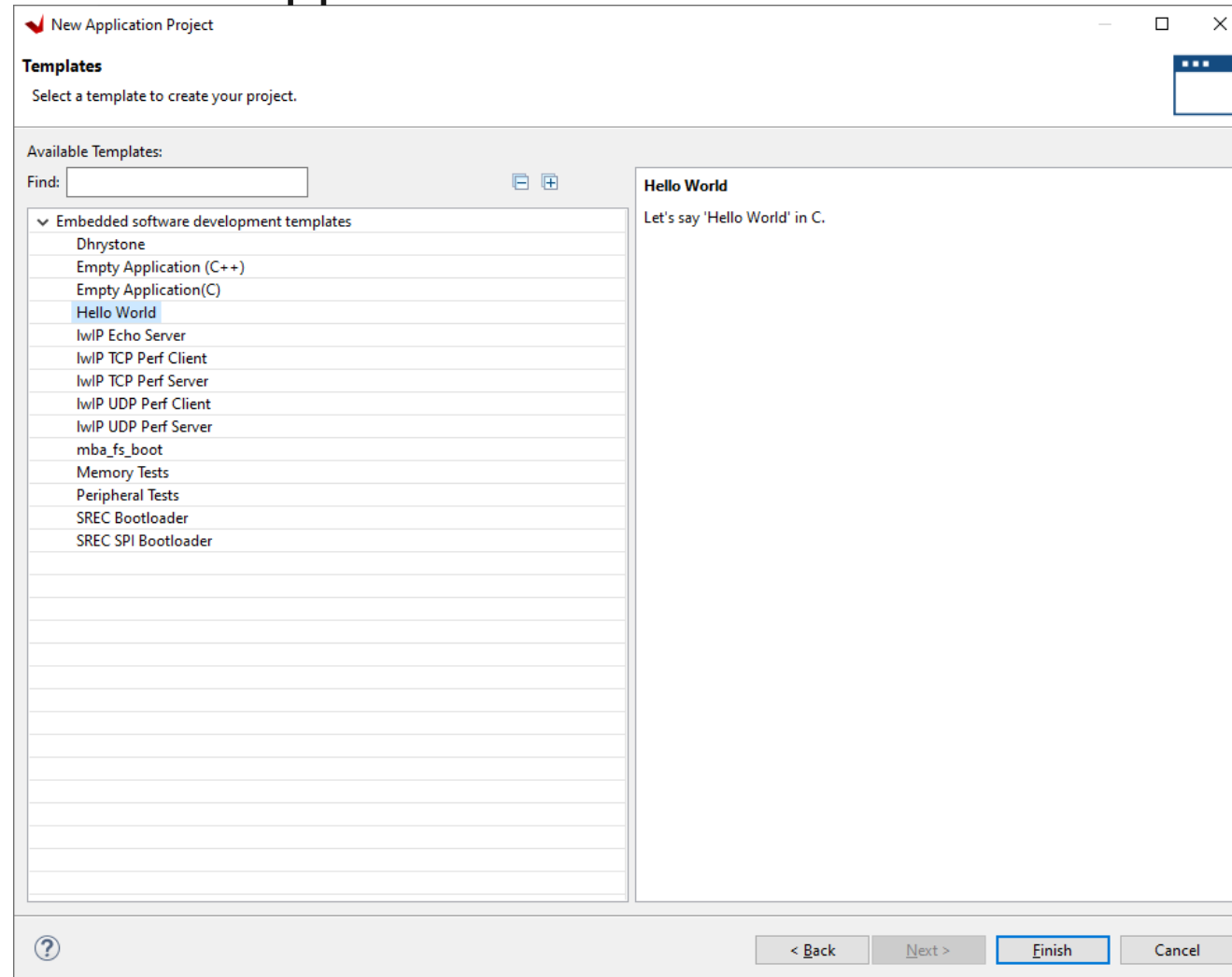
Processor: microblaze_0

Architecture: 32-bit

? < Back Next > Finish Cancel

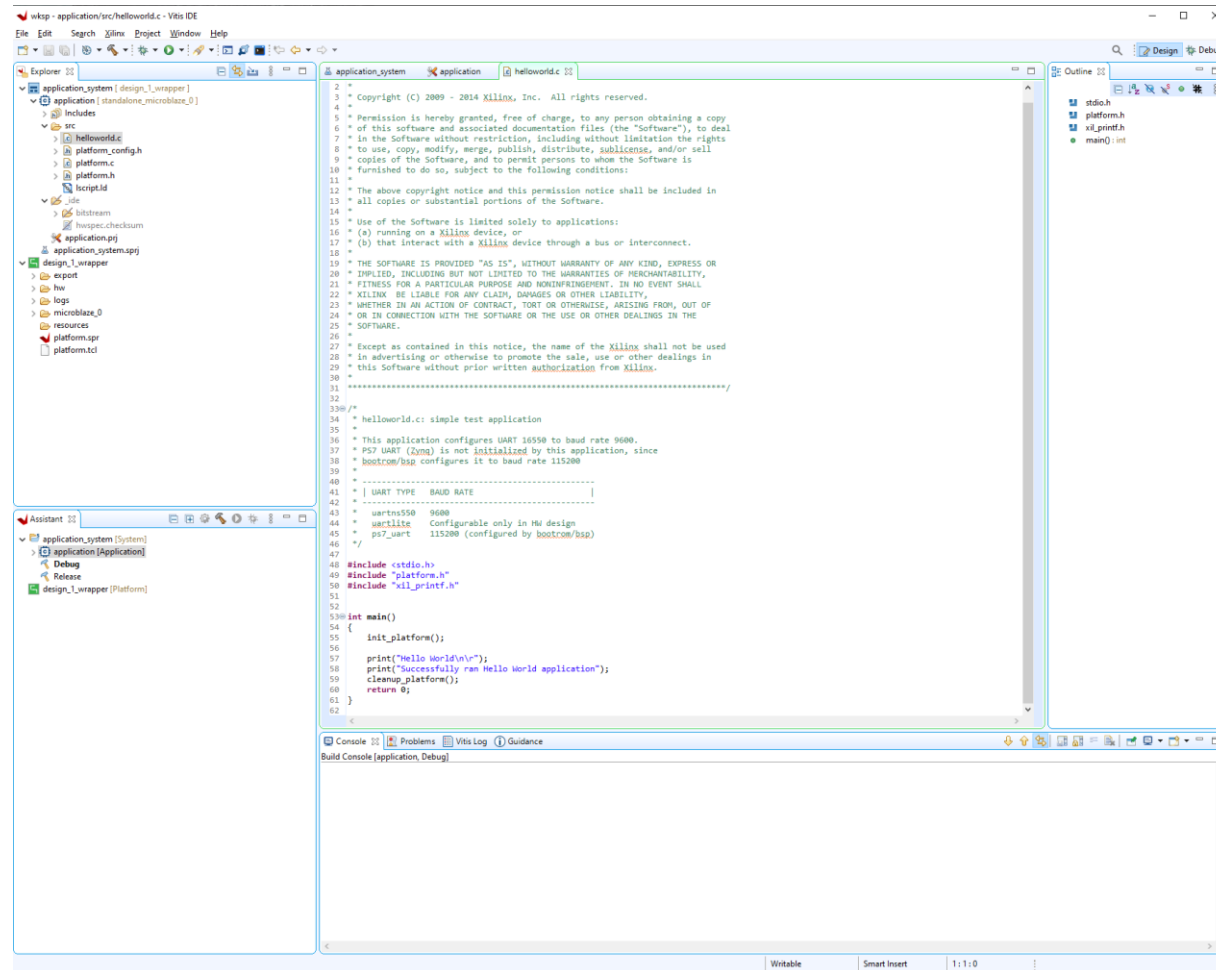
Lab: Mastering MicroBlaze

Select the Hello World Application



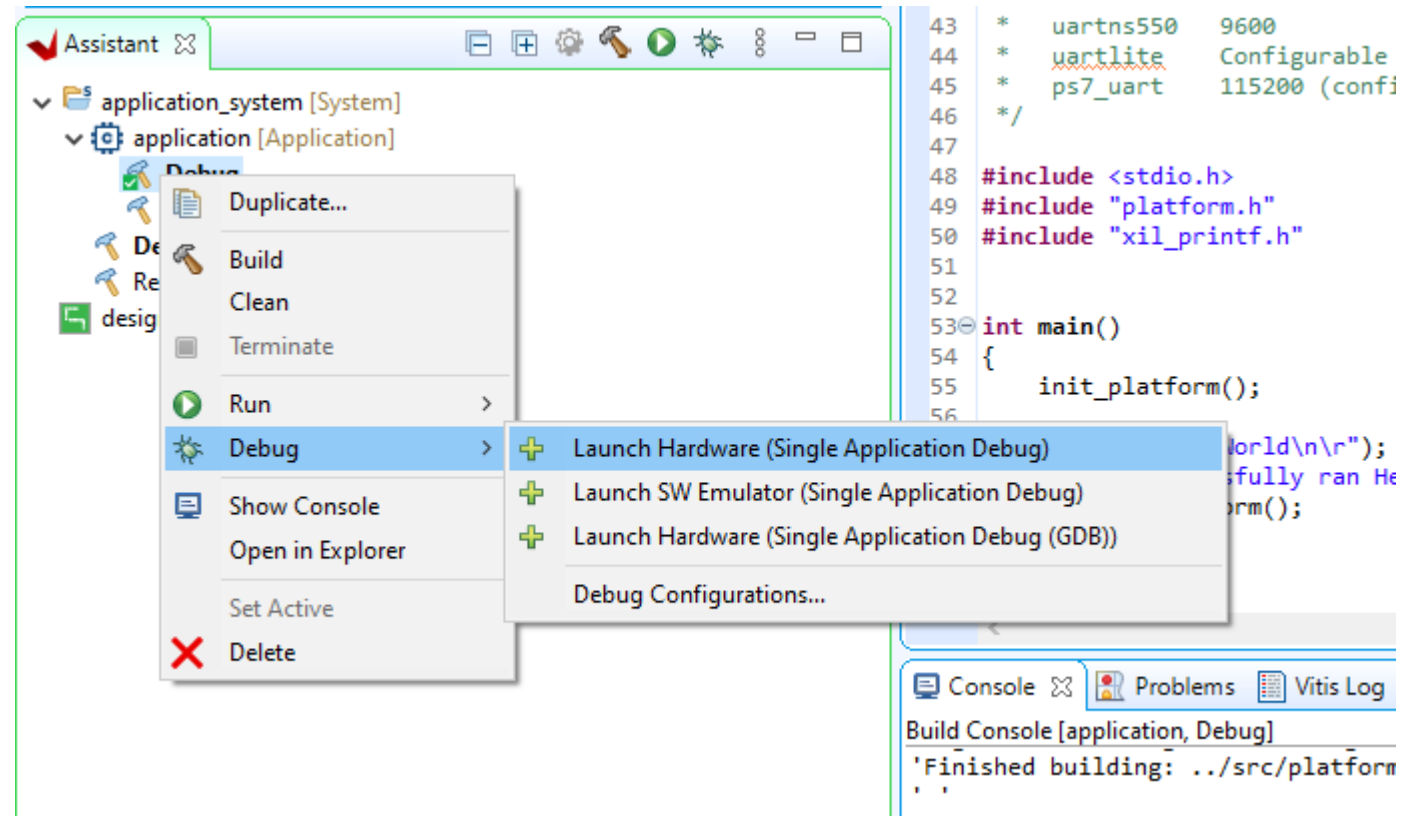
Lab: Mastering MicroBlaze

This will create a new application which prints hello world, we will use this to double check the correctness of the Vivado design. Click the Hammer to build the App



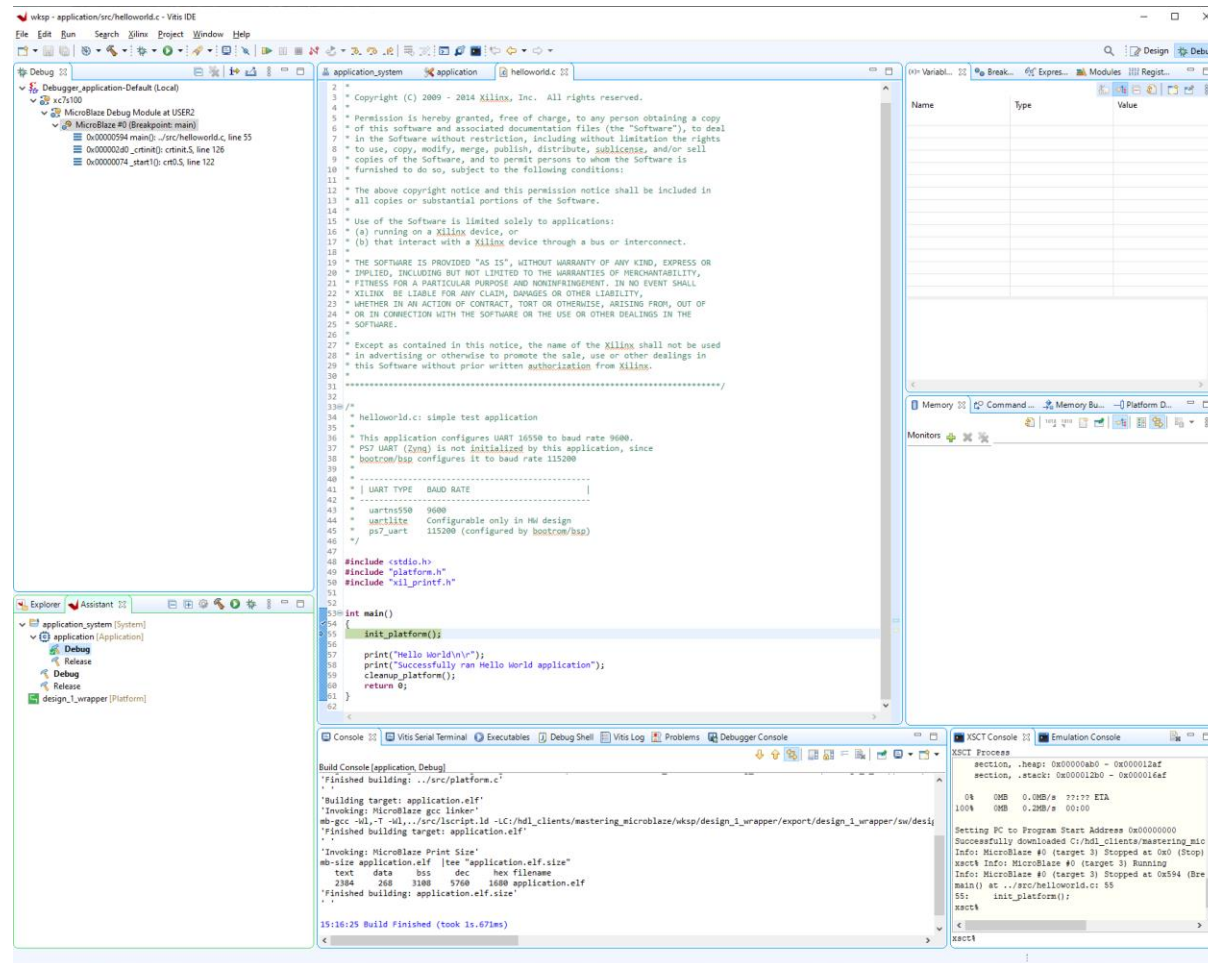
Lab: Mastering MicroBlaze

In the Assistant window select Debug, Launch on Hardware



Lab: Mastering MicroBlaze

This will program the FPGA and download the application, halting it for execution at the first line.



Lab: Mastering MicroBlaze

Single step through the code and observe the output in a terminal window

```
45  *   ps7_uart    115200 (configured by bootrom/bsp)
46  */
47
48  #include <stdio.h>
49  #include "platform.h"
50  #include "xil_printf.h"
51
52
53  int main()
54  {
55      init_platform();
56
57      print("Hello World\n\r");
58      print("Successfully ran Hello World application");
59      cleanup_platform();
60      return 0;
61  }
62
```

Console Vitis Serial Terminal Executables Debug Shell Vitis Log

Build Console [application, Debug]

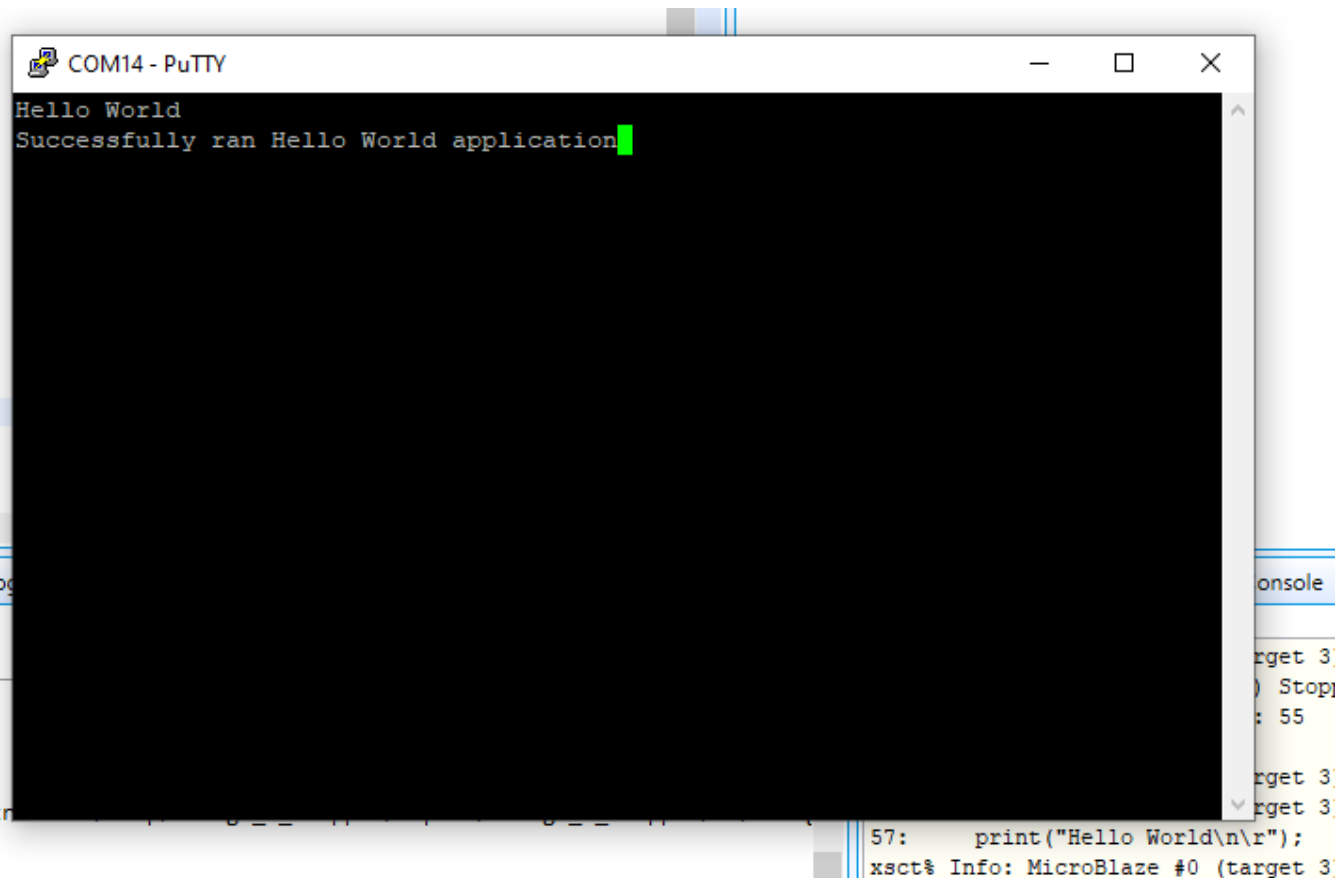
'Finished building: ../src/platform.c'

'Building target: application.elf'

'Invoking: MicroBlaze gcc linker'

mb-gcc -Wl,-T -Wl,.../src/lscript.ld -LC:/hdl_clients/mastering_micro

'Finished building target: application.elf'



```
COM14 - PuTTY
Hello World
Successfully ran Hello World application
```

```
57:      print("Hello World\n\r");
xsct% Info: MicroBlaze #0 (target 3)
```

Lab: Mastering MicroBlaze

Replace the code in the hello world application with that in the Git Repo, rebuild the code and download it to the development board.

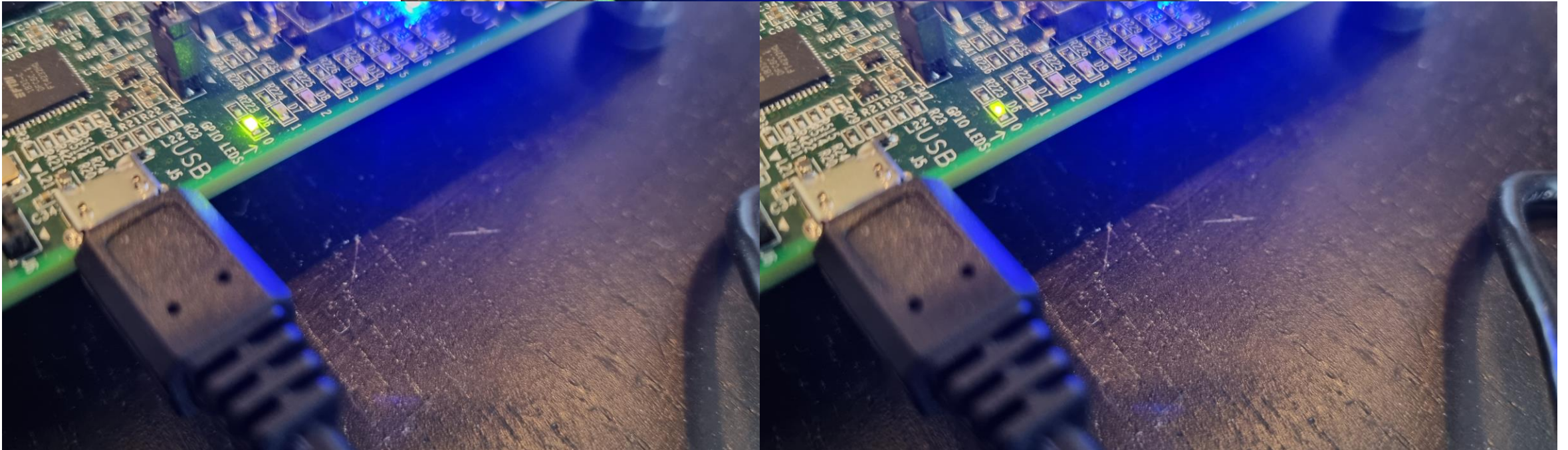
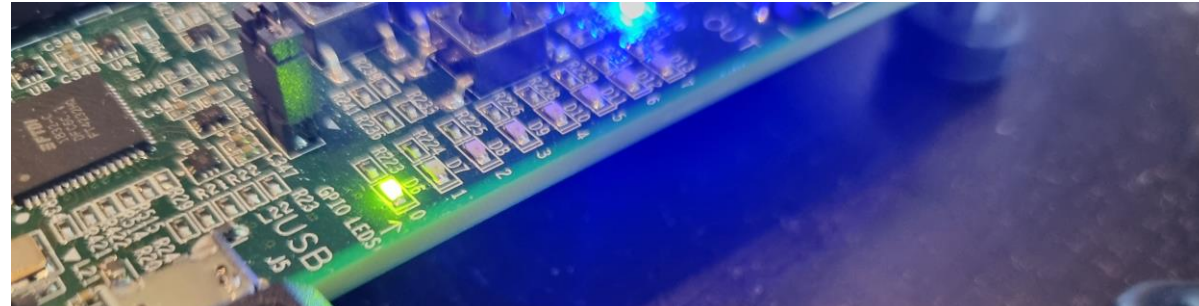
```
65 int main()
66 {
67     u8 DutyCycle;
68     u8 NoOfCycles;
69     u8 Div;
70     u8 Value;
71     u32 Period;
72     u32 HighTime;
73     u64 WaitCount;
74     int Status;
75     float DivF;
76
77     init_platform();
78
79     print("Mastering MicroBlaze\n\r");
80
81
82     XTmrCtr_Initialize(&TimerCounterInst, TMRCTR_DEVICE_ID);
83
84     XTmrCtr_PwmDisable(&TimerCounterInst);
85
86     Div = DUTYCYCLE_DIVISOR;
87     Period = PWM_PERIOD;
88     HighTime = PWM_PERIOD * (Div/10) ;
89
90     DutyCycle = XTmrCtr_PwmConfigure(&TimerCounterInst, Period, HighTime);
91
92     xil_printf("PWM Configured for Duty Cycle = %d\r\n", DutyCycle);
93
94     /* Enable PWM */
95     XTmrCtr_PwmEnable(&TimerCounterInst);
96
97     while(1){
98         print("Select a Brightness between 0 and 9\n\r");
99
100         /* Read an input value from the console. */
101         Value = inbyte();
102         Div = Value - 0x30;
103         DivF = (float) Div /10;
104         HighTime = PWM_PERIOD * DivF;
105
106         XTmrCtr_PwmDisable(&TimerCounterInst);
107         DutyCycle = XTmrCtr_PwmConfigure(&TimerCounterInst, Period, HighTime);
108         XTmrCtr_PwmEnable(&TimerCounterInst);
109     }
110
111     cleanup_platform();
112     return 0;
113 }
114
```

Lab: Mastering MicroBlaze

In a terminal window change the PWM to the LED and see the intensity change

[illegible]

Lab: Mastering MicroBlaze





ADIUVO

ENGINEERING AND TRAINING, LTD.

www.adiuvoengineering.com



adam@adiuvoengineering.com