# Lab 03: Synthesize, Implement and Generate Bistream with Vivado

# Objectives

After completing this lab, you will be able to:

- Synthesize and implement the design.

- Generate the bitstream.

- Configure Zynq Ultrascale+ using the generated bitstream and verify the functionality

# Steps

This lab is a continuation of lab 1. Open the project created in lab 1 and continue from there.

**Synthesize the Design and Analyze the Project Summary Output**

1. Click on **Run Synthesis** under the *SYNTHESIS* tasks of the *Flow Navigator* pane.

The synthesis process will be run on the lab1.v file (and all its hierarchical files if they exist). When the process is completed a *Synthesis Completed* dialog box with three options will be displayed.

2. Select the *Open Synthesized Design* option and click **OK** as we want to look at the synthesis output before progressing to the implementation stage.

Click **Yes** to close the elaborated design if the dialog box is displayed.

3. Select the **Project Summary** tab and understand the various windows.

If you don’t see the Project Summary tab then select **Window > Project Summary** or click the **Project Summary** icon .

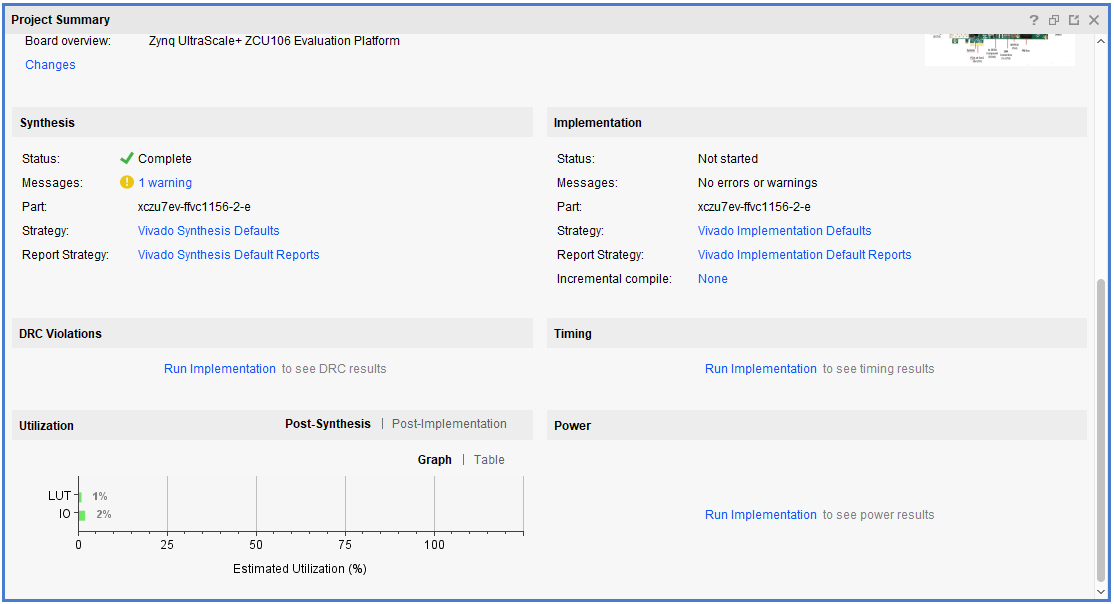


Figure 1 Project summary view

Click on the various links to see what information they provide and which allows you to change the synthesis settings.

4. Click on the **Table** tab in the **Project Summary** tab.

Notice that there are an estimated 3 LUTs and 8 IOs (4 input and 4 output) that are used.

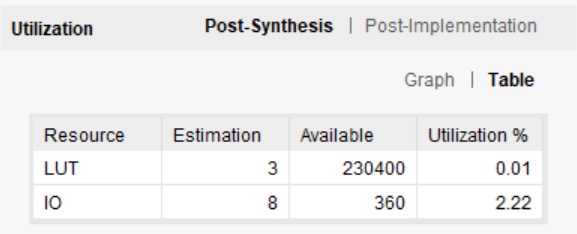


Figure 2 Resource utilization estimation symmary

5. In The *Flow Navigator*, under Synthesis (expand *Open Synthesized Design* if necessary), click on **Schematic** to view the synthesized design in a schematic view.

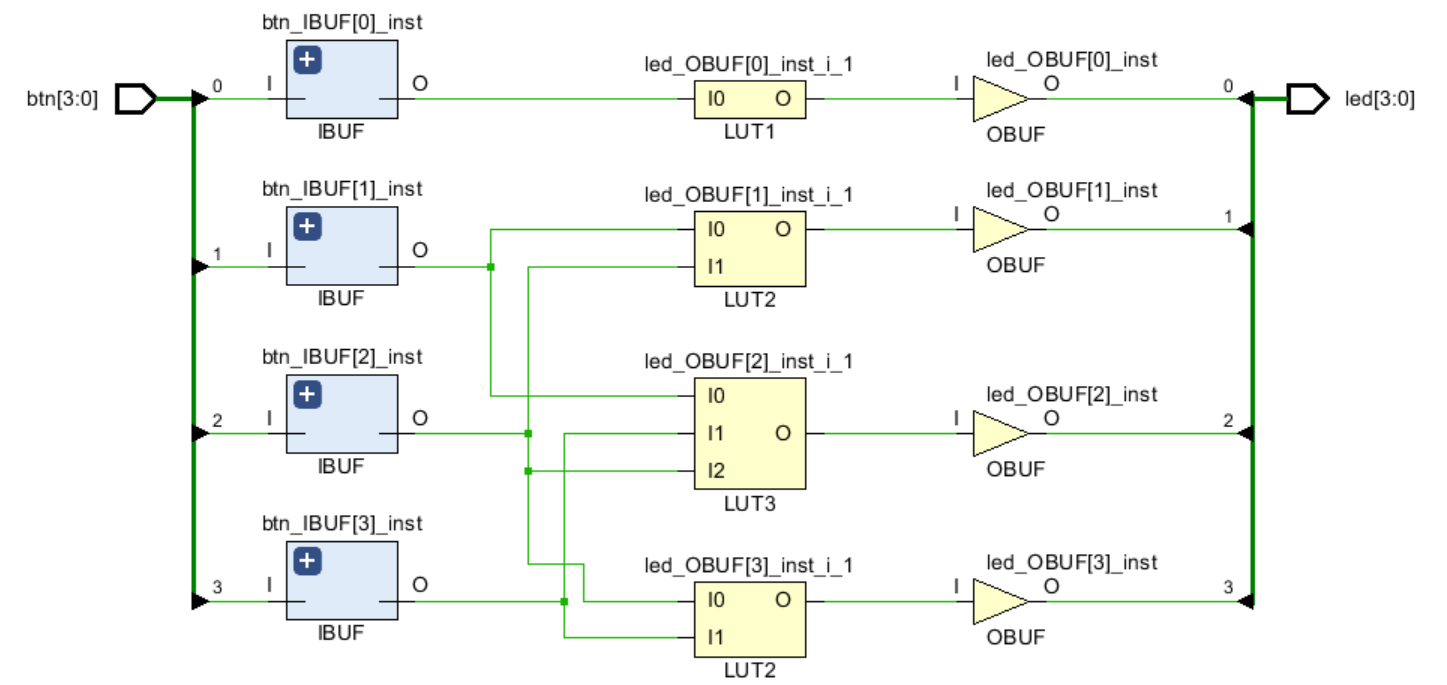


Figure 3 Synthesized design's schematic view

Notice that IBUFs and OBUFs are automatically instantiated (added) to the design as the input and output are buffered. The logical gates are implemented in LUTs (1 input is listed as LUT1, 2 input is listed as LUT2, and 3 input is listed as LUT3). Four gates in RTL analysis output are mapped onto 3 LUTs in the synthesized output.

**Implement the Design.**

1. Click on **Run Implementation** under the *Implementation* tasks of the *Flow Navigator* pane.

The implementation process will be run on the synthesized design. When the process is completed, an *Implementation Completed* dialog box with three options will be displayed. You can choose to use how many jobs you want to implement this design. In general, more jobs consume more computing resources and less runtime.

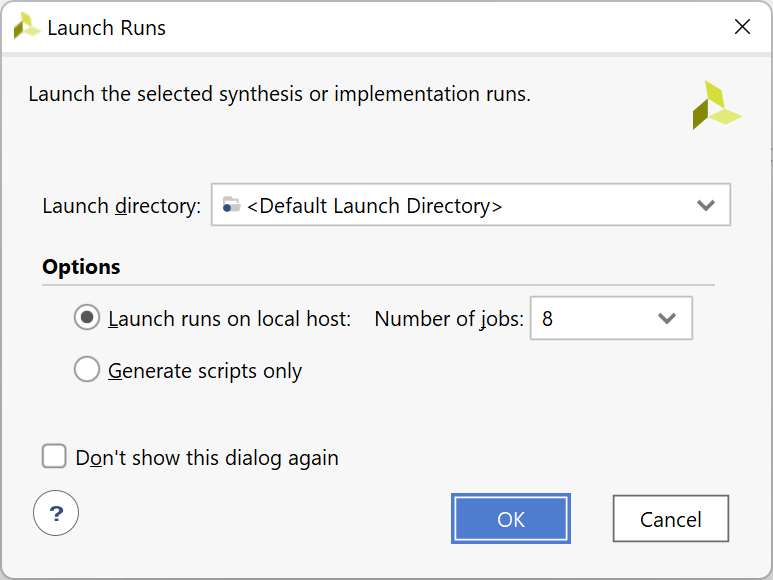


Figure 4 Select the threads (jobs) you want to use for implementation

2. Select **Open implemented design** and click **OK** as we want to look at the implemented design in a Device view tab.

3. Click Yes, if prompted, to close the synthesized design. The implemented design will be opened.

4. In the Netlist pane, select one of the nets (e.g. led\_OBUF[3]) and notice that the net displayed in the Device view tab (you may have to zoom in to see it).

5. If it is not selected, click the *Routing Resources* icon  to show routing resources.



Figure 5 Viewing implemented design

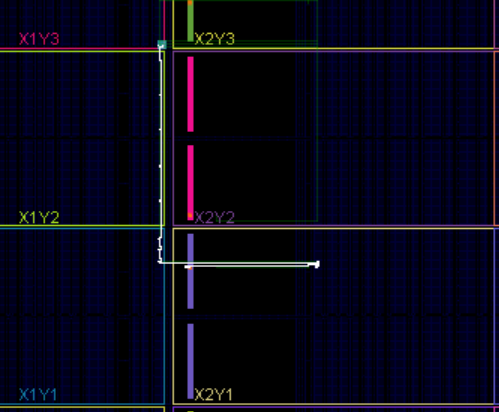


Figure 6 Selecting a net

6. Close the implemented design view by selecting **File > Close Implemented Design**, and select the **Project Summary** tab (you may have to change to the Default Layout view) and observe the results.

7. Select the **Post-Implementation** tab.

**Notice** that the actual resource utilization is 3 LUTs and 8 IOs. Also, it indicates that no timing constraints were defined for this design (since the design is combinational).

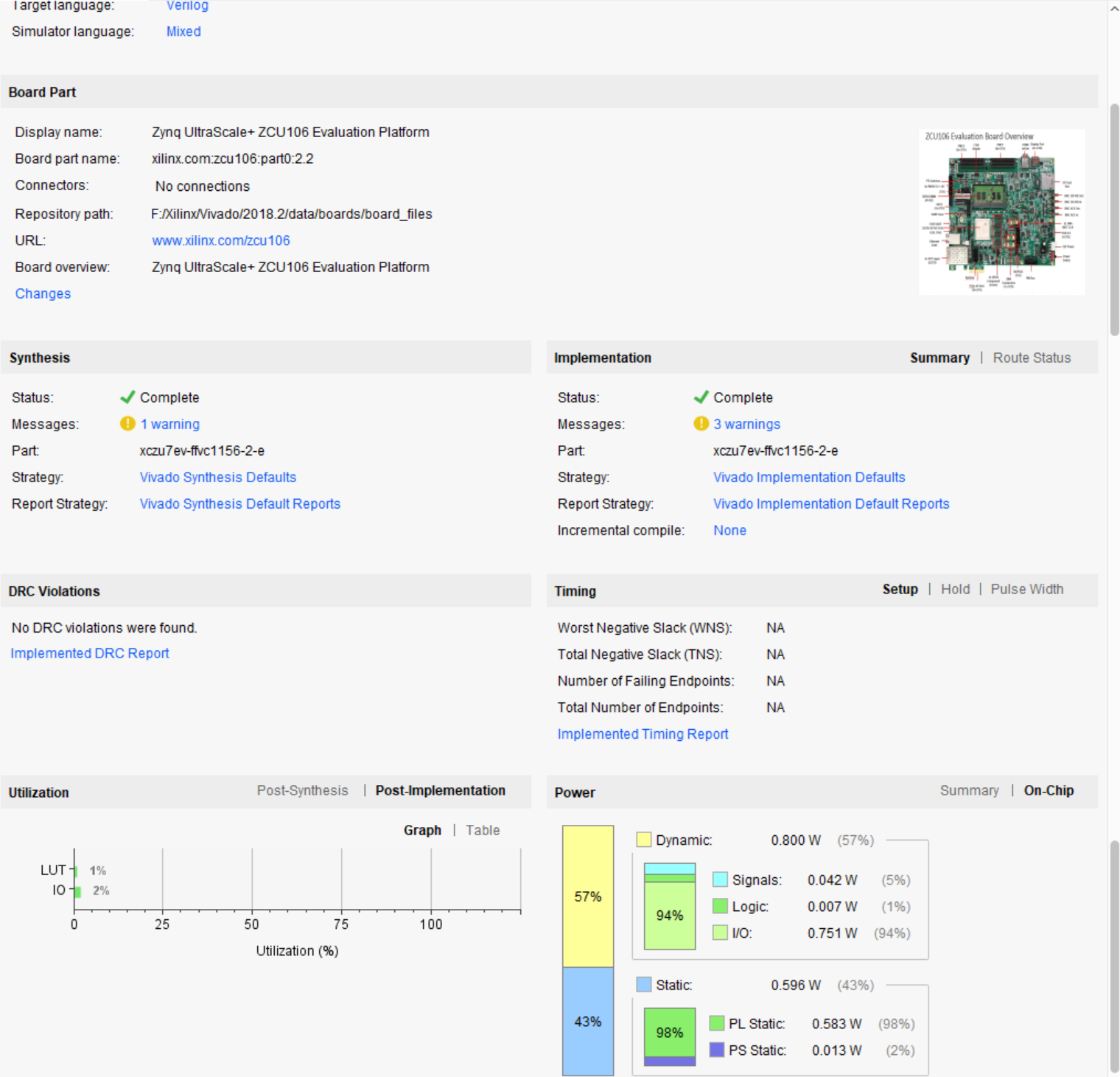


Figure 7 Implementation results for the ZCU106

8. In Vivado, select the Reports tab in the bottom panel (if not visible, click Window in the menu bar and select Reports), and double-click on the Utilization Report entry under the Place Design section. The report will be displayed in the auxiliary view pane showing resource utilization. Note that since the design is combinatorial, no registers are used.

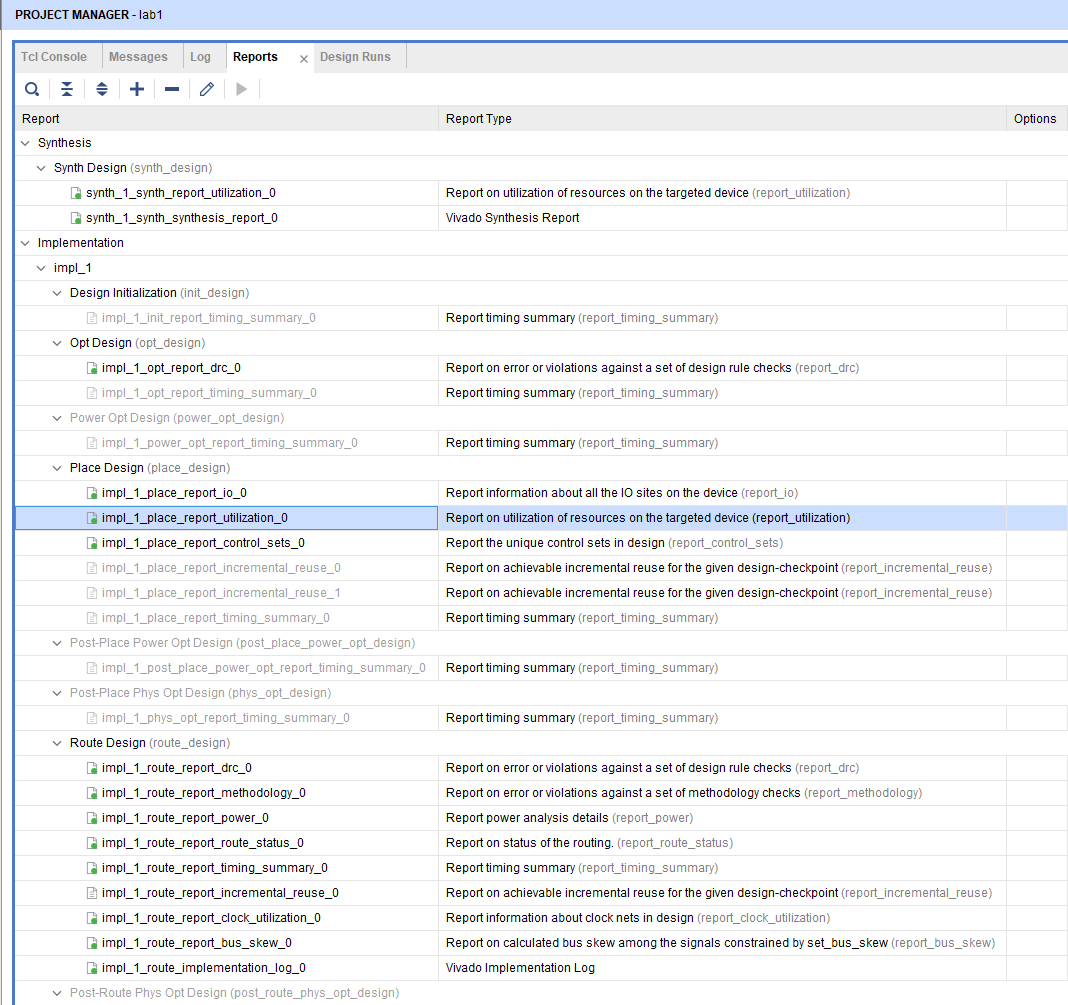


Figure 8 Available reports to view

**Perform Timing Simulation.**

1. Select **Run Simulation > Run Post-Implementation Timing Simulation** process under the *Simulation* tasks of the *Flow Navigator* pane.

The Vivado simulator will be launched using the implemented design and **lab1\_tb** as the top-level module.

2. Click on the **Zoom Fit** button to see the waveform window from 0 to 200 ns.

3. Right-click at 50 ns (where the btns input is set to 0000b) and select **Markers > Add Marker**.

4. Similarly, right-click and add a marker at around 58.000 ns where the **leds** changes.

5. You can also add a marker by clicking on the Add Marker button (). Click on the **Add Marker** button and left-click at around 60 ns where **e\_led** changes.

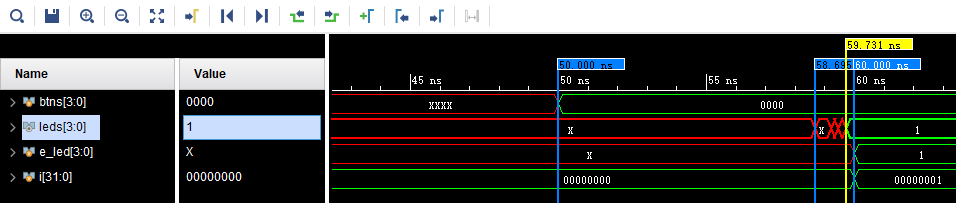


Figure 9 Timing simulation output

Notice that we monitored the expected led output at 10 ns after the input is changed (see the testbench) whereas the actual delay is about 8 to 9.7 ns (depending on the board).

6. Close the simulator by selecting **File > Close Simulation** without saving any changes.

**Generate the Bitstream and Verify Functionality.**

**Connect the board and power it ON. Generate the bitstream, open a hardware manager session, and program the FPGA.**

1. Make sure that the Micro-USB cable is connected to the USB JTAG (J2) connector on the ZCU106 board.

2. Connect the power supply to the ZCU106 (J52).

3. Power **ON** the board.

4. Click on the **Generate Bitstream** entry under the *PROGRAM AND DEBUG* tasks of the *Flow Navigator* pane.

The bitstream generation process will be run on the implemented design. When the process is completed a *Bitstream Generation Completed* dialog box with three options will be displayed.

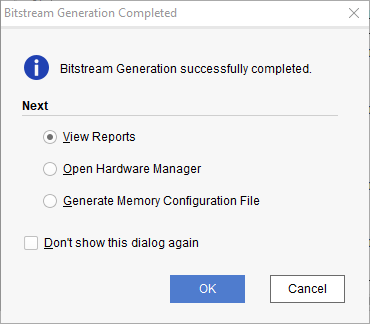


Figure 10 Bitsream generation

5. Select the *Open Hardware Manager* option and click **OK**.

The Hardware Manager window will open indicating “unconnected” status.

6. Click on the **Open target** link.



Figure 11 Opening new Hardware target

7. From the dropdown menu, click **Auto Connect**.

The Hardware Session status changes from Unconnected to the server name and the device is highlighted. Also notice that the Status indicates that it is not programmed.

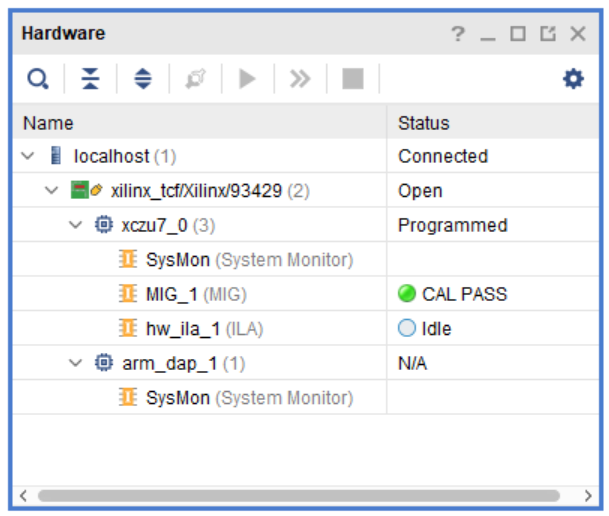


Figure 12 Opened hardare manager session

Select the device and verify that the lab1.bit is selected as the programming file in the General tab.

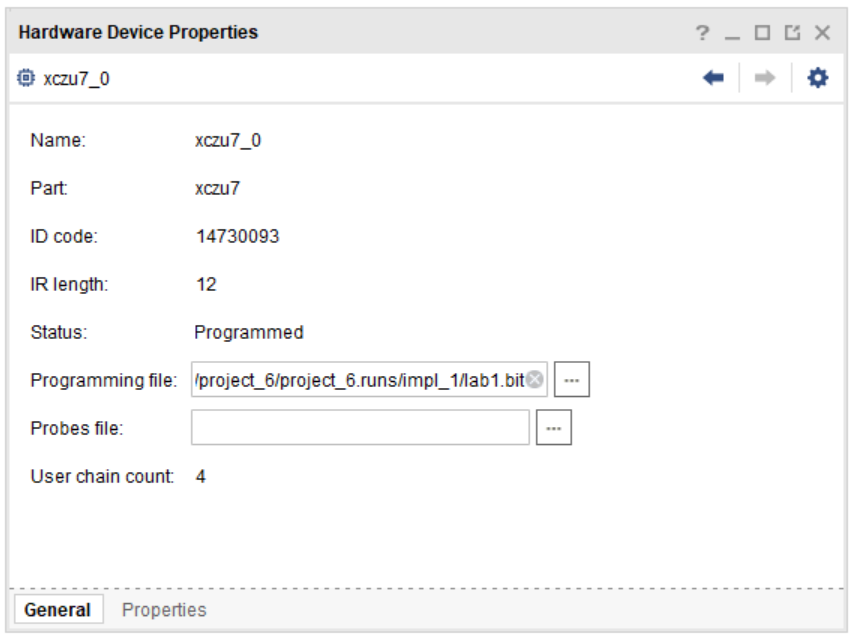


Figure 13 Programming File

8. Click on the *Program device* link in the green information bar to program the target FPGA device. Another way is to right click on the device and select *Program Device*.

9. Click **Program** to program the FPGA.

The DONE LED will lit when the device is programmed. You may see some other LEDs lit depending on switch positions.

10. Verify the functionality by flipping the switches and observing the output on the LEDs (Refer to the earlier logic diagram).

11. When satisfied, close the hardware manager session by selecting **File > Close Hardware Manager**.

12. Click **OK** to close the session.

13. Power **OFF** the board.

14. Close the **Vivado** program by selecting **File > Exit** and click **OK**.

# Exercises

1. Make some changes to the constraints file so that the input will come from the DIP switch (refer to the ZCU106 User Guide to find the correct pin number).

2. Try to implement and test on the kit the modified design in lab 1.