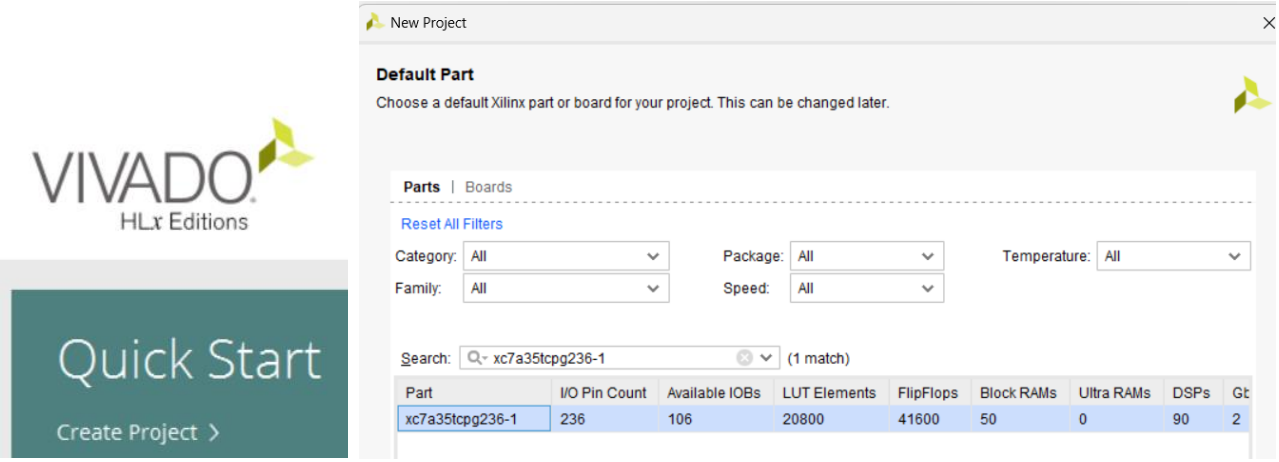
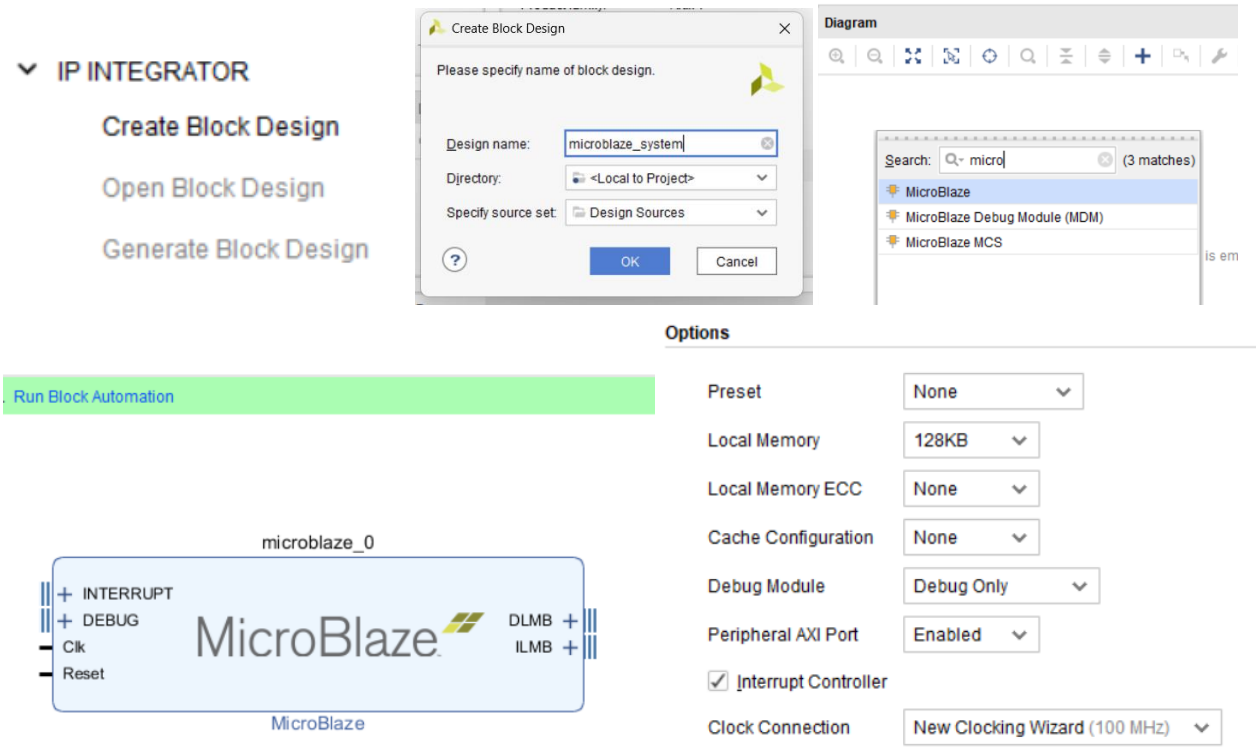
This guide will create a custom IP core for microblaze to handle all the basic IO interfaces on Basys3 board. Moreover, it will handle the switching signals for seven segment display.

This guide has been inspired from “Building Custom AXI IP” guide from Xilinx
https://www.xilinx.com/content/dam/xilinx/training/embedded/embd/Building-Custom-AXI-IP.pdf

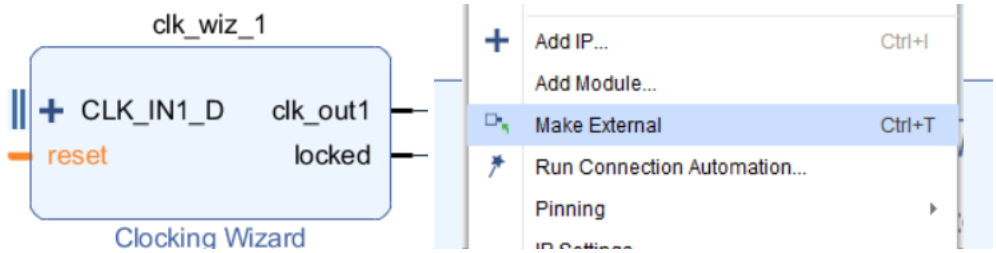
1- Create a new Vivado project. Instead of using board file, select the part number i.e. “xc7a35tcpg236-1”.



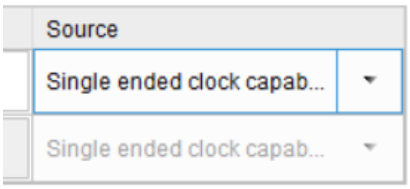
2- Create a minimal microblaze system.



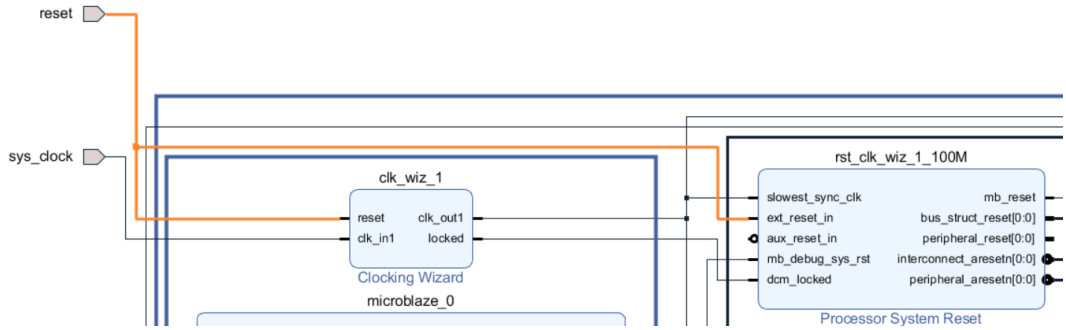
Run Connection automation. Right click “reset” pin of clocking wizard. Make it external.



Similarly, make clk_in1_d external as well after setting it ‘single ended’.

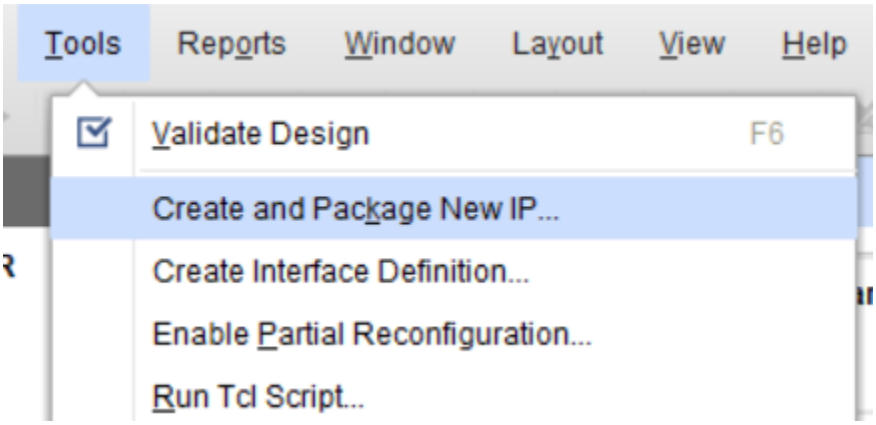


Make the following reset connection manually.



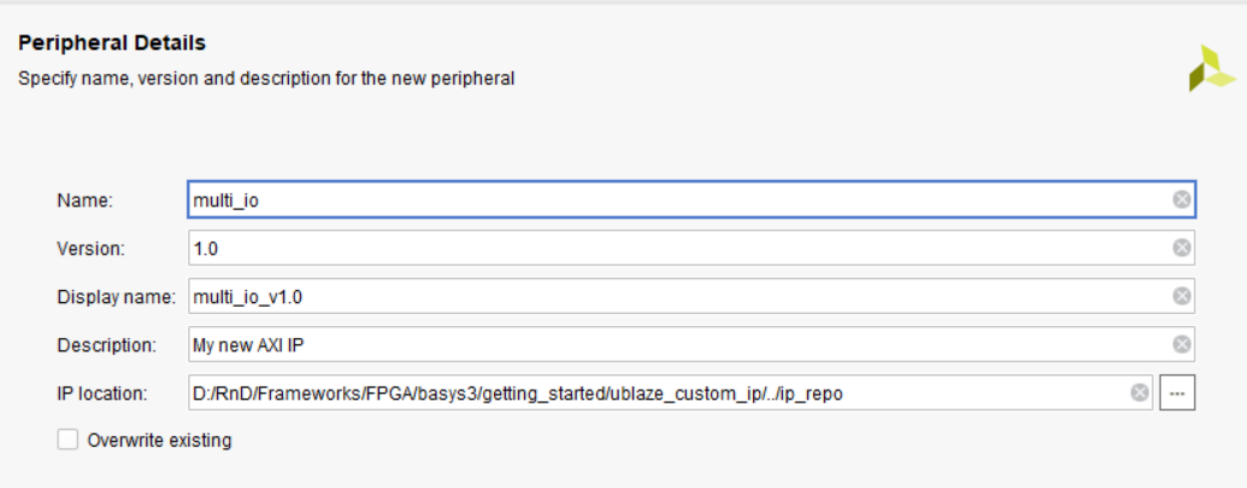
3- Add UARTlite IP. Run connection automation.

4- Select “Create and Package New IP” from tools.

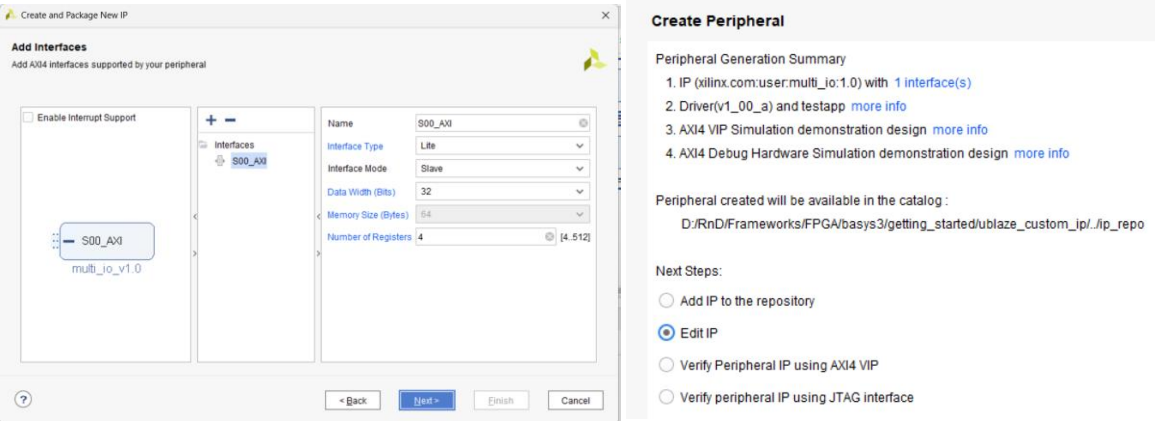


Select create a new AXI4 peripheral.

Give it a name e.g. “multi_io”

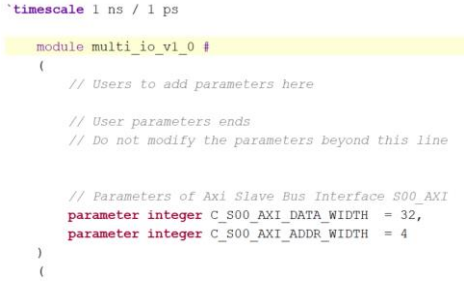


5- Leave the default connections intact. In the next screen, select “edit IP”



This will open a new Vivado project for IP customization.

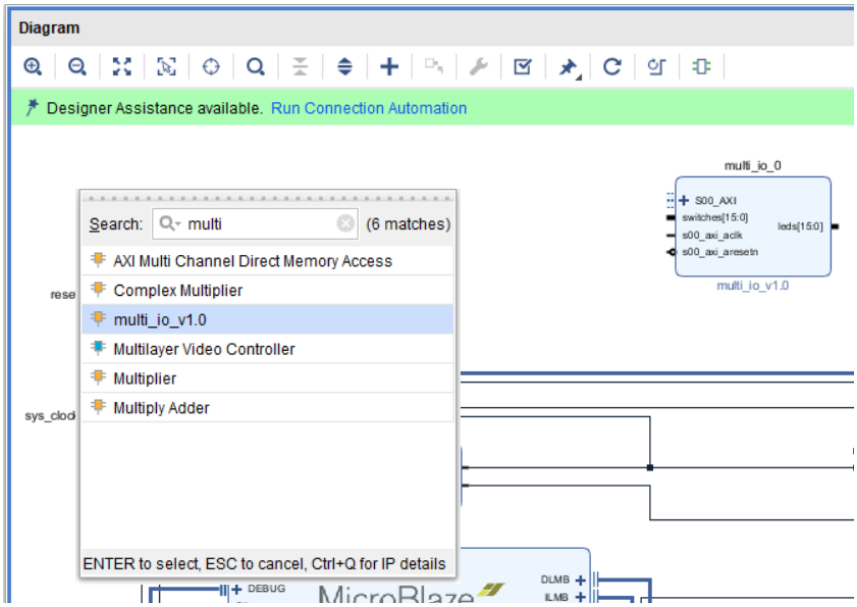
6- Open the Verilog file for the newly generated IP.



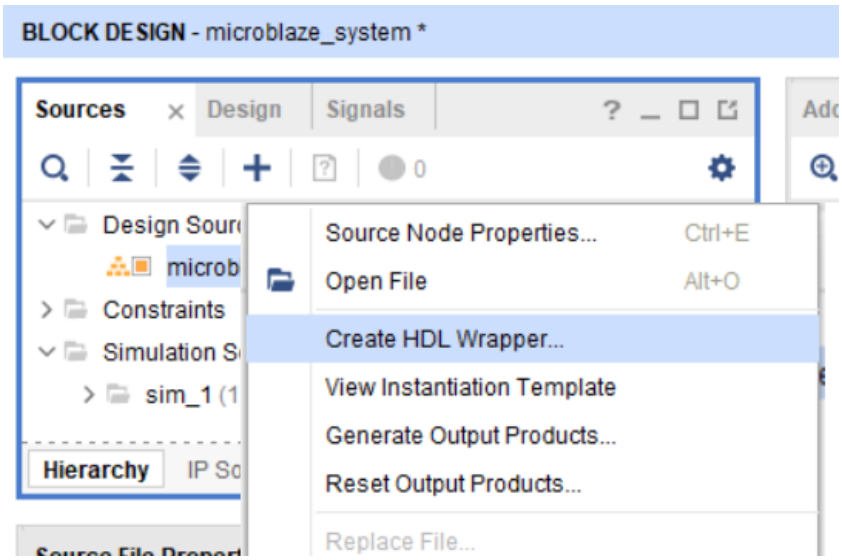
There would be two files.

7- Add ports for leds, dip switches, push buttons and seven segment display. Write Verilog code for the desired functionality. Set other IP-related parameters if needed in the components.xml. See the “Building Custom AXI IP” guide from Xilinx for details. Close the IP project.

8- Add the custom IP in microblaze board file. Make interfaces external.



9- Create HDL wrapper for the system.



10- Add the constraint file for Basys3 and manually add all the signals. Generate Bitstream and write code in SDK.