EEDG/CE 6370
Design and Analysis of Reconfigurable Systems
Homework 3
FPGA Soft IPs - Multipliers

1. Laboratory Objectives

- Understand how to use the IPs included in Quartus Prime's IP generator.
- Prototype the design on an FPGA board using switches as inputs and LEDs to visualize the output.

2. Summary

In this lab you will learn to use Quartus's IP generator and configure a given IP based on different specifications.

3. Pre-lab

- Review the lecture slides that covers soft IPs

4. Tool Requirements

- Quartus Prime
- Questa simulator
- DE1-SoC board

5. System Overview

Figure 1 shows how the multiplier function will be implemented on the DE1-SoC board. Out of the 10 switches, 8 will be used as inputs (dataa and datab). The result of the multiplication will be displayed on 8 of the 10 LEDs.

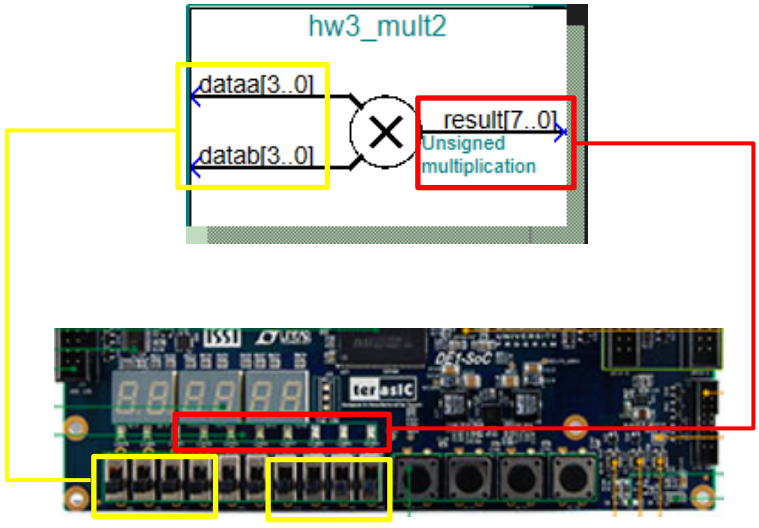
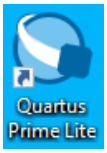


Figure 1 DE1-SoC board peripherals overview used to enter and display result of the multiplication

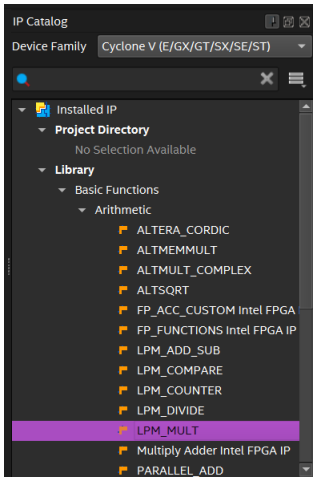
6. Design using IP Generator

- Open Quartus Prime 

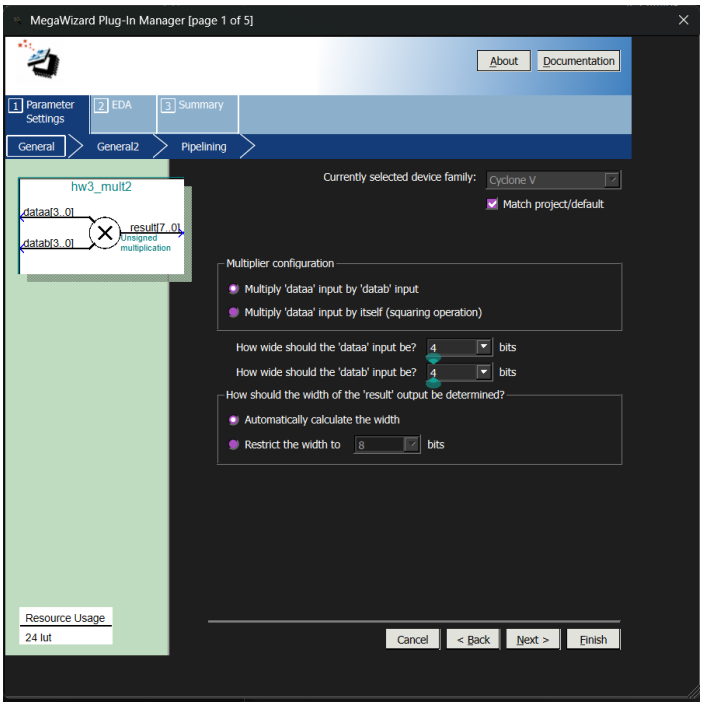
In case of Windows run the tools always as administrator (right click in icon)

- Create a new project.
  File ➔ New Project Wizard → hwmult
- Click next → empty project →
- Click next (do not add any files)
- Select Cyclone V 5CSEMA5F31C6 device.
- Click next and finish.

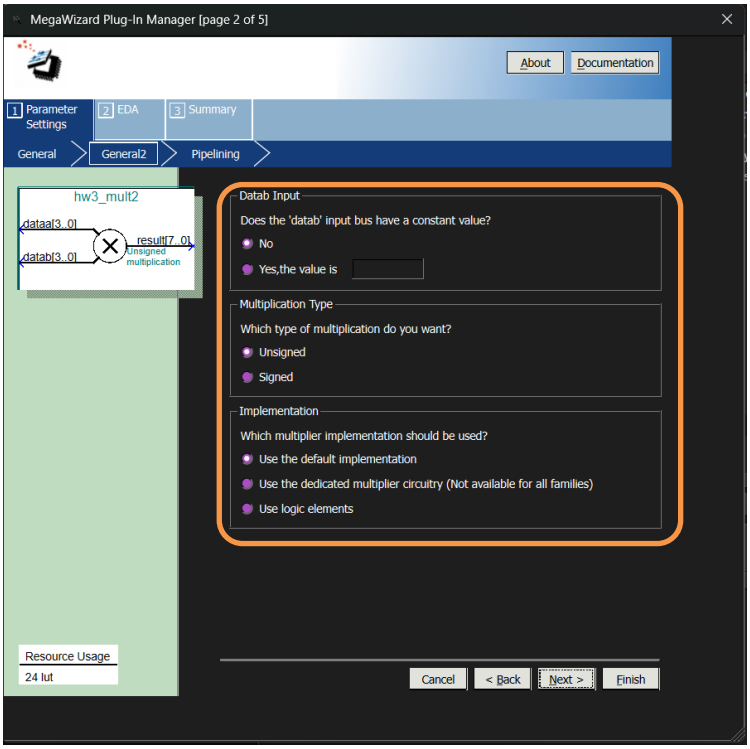
- Open the IP core generator (right pane of Quartus main window)→ Select LPM_MULT from the Basic Functions→Arithmetic this will open a window to name the IP. **Note:** LPM stands for Library of Parametrized Modules)



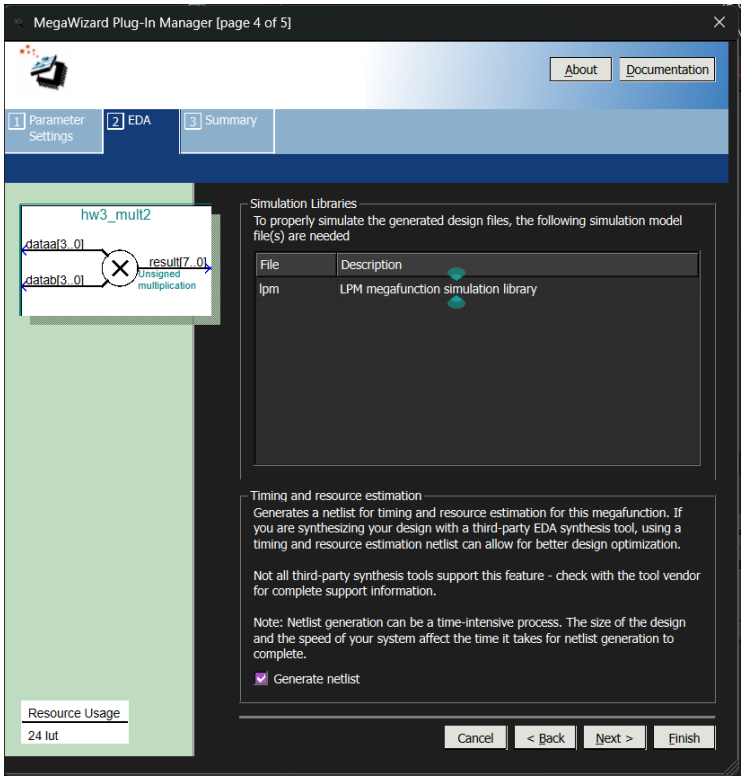
- Name the IP → hwmult
- Configure the IP as shown below: 4 bits for the input dataa, and 4 bits for input datab. The result of two 4-bit numbers will be an 8-bit number as shown.



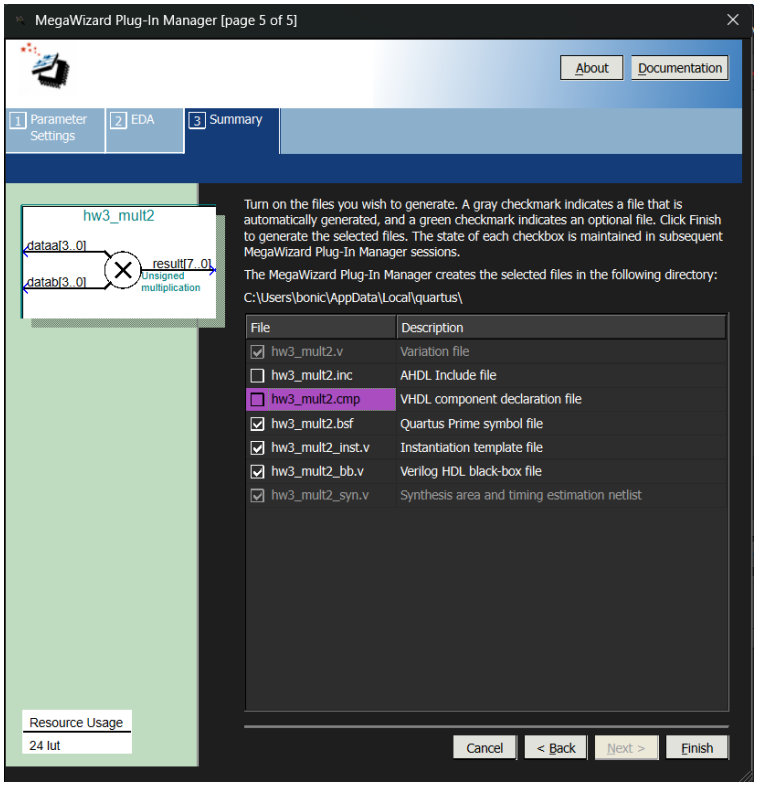
- On the next screens leave all the default options as shown below:



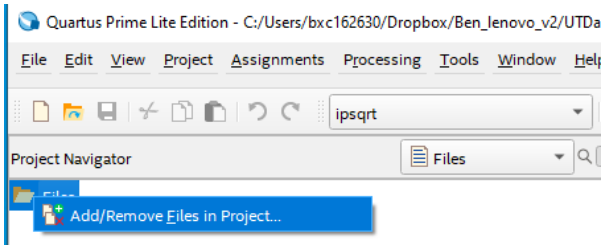
- Select “Generate netlist”. This will create the simulation model for the multiplier that you are creating.



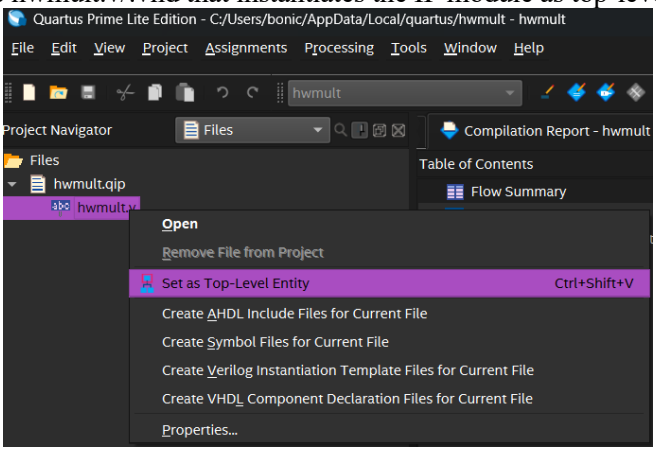
- Continue selecting the project files to be generated in order to instantiate the multiplier in the rest of the project files.



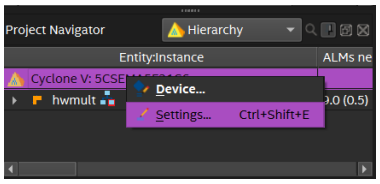
- The IP generator will store the new Verilog/VHDL files in hwmult folder/synthesis and hwmult/synthesis/submodules
- Add all of the files to the project

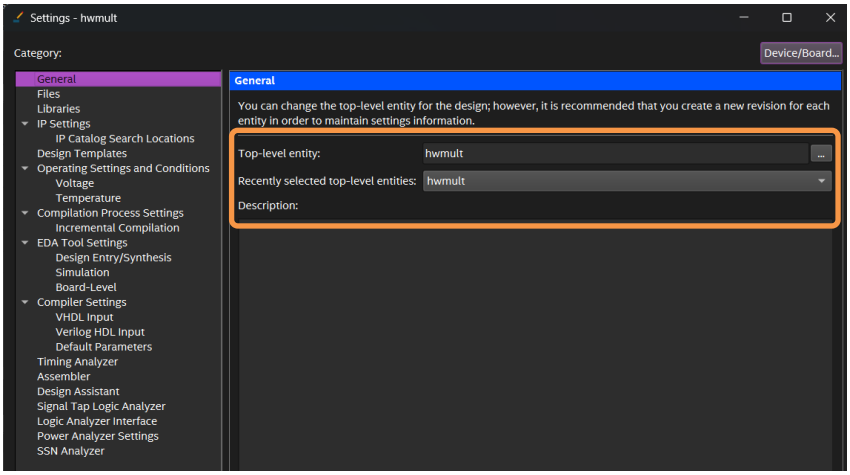


- Set the main file hwmult.v/.vhd that instantiates the IP module as top-level entity:



- If the file name does not match the entity name, set the top-level entity in Settings→General->top-level entity.





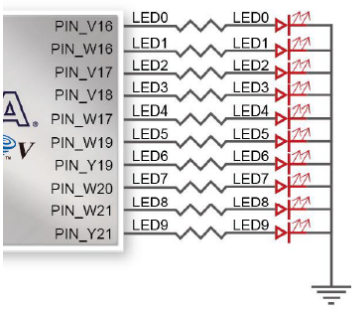
- Create a new SDC file where you specify the FPGA clock running at 50MHz (File→New→SDC file)

```
create_clock -name "clk" -period 20.000ns [get_ports {clk}]
derive_pll_clocks
derive_clock_uncertainty
```

- Connect the top entity IOs to the DE1-SoC board switches, keys and LEDs through the Pin Planner (assignment→Pin Planner) based on the datasheet given by Terasic (see appendix)

1. Connect SW9 to SW6 to dataa and SW3 to SW0 to datab

Connected results(7..0) to LED7 to LED0.



Note: Be careful with the order to the Pins and MSB to LSB bits.

- Synthesize and generate the bitstream
- Connected the DE1-SoC board→ Auto-detect→ Select new .sof file and program FPGA.
- Check that the design works.

UT Dallas Honor Code

UT Dallas values academic integrity. Therefore, all students must understand the meaning and consequences of cheating, plagiarism and other academic offences under the Code of Student Conduct and Disciplinary Procedures.
Group assignments must be completed solely by the members of the group. Cross-group work is not allowed. Moreover, similar assignments have been offered before at UT Dallas and other universities. Any use of information from previous assignments is prohibited. The tutorials are to be taken individually. Failure to respect this rule constitutes dishonesty and is a direct violation of the University Honor Code.

Appendix

Table 3-6 Pin Assignment of Slide Switches

Signal Name	*FPGA Pin No.*	*Description*	*I/O Standard*
SW[0]	PIN_AB12	Slide Switch[0]	3.3V
SW[1]	PIN_AC12	Slide Switch[1]	3.3V
SW[2]	PIN_AF9	Slide Switch[2]	3.3V
SW[3]	PIN_AF10	Slide Switch[3]	3.3V
SW[4]	PIN_AD11	Slide Switch[4]	3.3V
SW[5]	PIN_AD12	Slide Switch[5]	3.3V
SW[6]	PIN_AE11	Slide Switch[6]	3.3V
SW[7]	PIN_AC9	Slide Switch[7]	3.3V
SW[8]	PIN_AD10	Slide Switch[8]	3.3V
SW[9]	PIN_AE12	Slide Switch[9]	3.3V

Table 3-8 Pin Assignment of LEDs

Signal Name	*FPGA Pin No.*	*Description*	*I/O Standard*
LEDR[0]	PIN_V16	LED [0]	3.3V
LEDR[1]	PIN_W16	LED [1]	3.3V
LEDR[2]	PIN_V17	LED [2]	3.3V
LEDR[3]	PIN_V18	LED [3]	3.3V
LEDR[4]	PIN_W17	LED [4]	3.3V
LEDR[5]	PIN_W19	LED [5]	3.3V
LEDR[6]	PIN_Y19	LED [6]	3.3V
LEDR[7]	PIN_W20	LED [7]	3.3V
LEDR[8]	PIN_W21	LED [8]	3.3V
LEDR[9]	PIN_Y21	LED [9]	3.3V

[END]