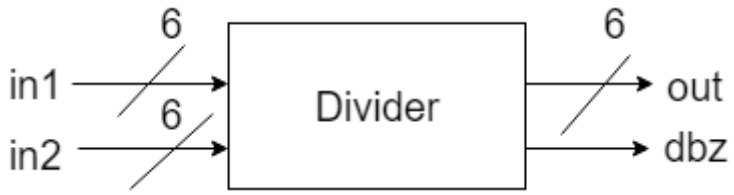
2018 Digital IC Design
Homework 2: Divider

1. Introduction

The divider is a device that can be used to perform division. It can be usually classified into signed divider or unsigned divider. Please implement a 6-bit unsigned divider. The method of implement divider is not constraint, but you can not use verilog build-in divider directly. As you encounter divided-by-zero, you need to set signal dbz to 1

2. Design Specifications

2.1 Block Overview



2.2 I/O Interface

Signal Name	I/O	width	Description
in1	Input	6	
in2	Input	6	
out	Output	6	
dbz	Output	1	You need to set 1 as you encounter divided-by-zero

2.3 File Description

File Name	Description
div.v	RTL code for using Verilog
div_tb.v	Test bench for verifying design
cycloneii_atoms.v	Simulation library for gate-level simulation

3. Scoring

3.1 Functional Simulation (pre-sim) **[70%]**

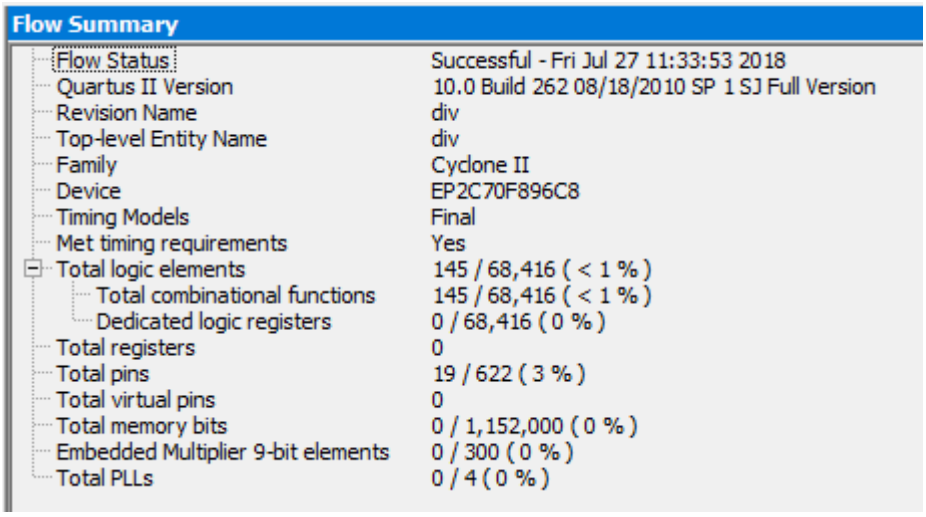
All of the result should be generated correctly, and you will get the following message in ModleSim simulation.

```
#         4090 data is correct
#         4091 data is correct
#         4092 data is correct
#         4093 data is correct
#         4094 data is correct
#         4095 data is correct
#         4096 data is correct
# --------------------PASS--------------------
# All data have been generated successfully!
# ** Note: $stop    : C:/Users/DIC/Desktop/2018_DIC/201
#    Time: 409610 ns  Iteration: 0  Instance: /div_tb
```

3.2 Gate-Level Simulation (post-sim) **[30%]**

3.2.1 Synthesis

Your code should be synthesizable. After synthesizing in Quartus, a file named *div.vo* will be obtained.



Device : Cyclone II EP2C70F896C8

3.2.2 Simulation

All of the result should be generated correctly using *div.vo*, and you will get the following message in ModleSim simulation.

```
#         4089 data is correct
#         4090 data is correct
#         4091 data is correct
#         4092 data is correct
#         4093 data is correct
#         4094 data is correct
#         4095 data is correct
#         4096 data is correct
# --------------------PASS--------------------
# All data have been generated successfully!
# ** Note: $stop    : C:/Users/DIC/Desktop/2018_DIC/2018
#    Time: 143370 ns  Iteration: 0  Instance: /div_tb
```

3.3 Performance **[0%]**

The performance is scored by the logic elements you used and the simulation time in post-sim. The scoring equation is *(Total logic elements + total memory bit+ 9*embedded multiplier 9-bit element) × (longest gate-level simulation time in ns)*. (The smaller the better).

3.4 Note:

Like HW1, you can modify your clock cycle in post-sim

4. Submission

4.1 Submitted files

You should classified your files into three directories and compressed to .zip format. The naming rule is **HW2_*studentID_name_version*.zip**. The vision is v1 for the first submission, and v2, v3… for the revisions.

	RTL category
*.v	All of your verilog RTL code
	Gate-Level category
*.vo	Gate-Level netlist generated by Quartus
	Documentary category
*.pdf	The report file of your design (in pdf).

4.2 Report file

Please follow the spec of report. You are asked to describe how the circuit is designed as detailed as possible, and the flow summary result and minimum CYCLE in post-sim are necessary.

4.3 Please submit your .zip file to folder HW2 in the ftp site.

Deadline: 2018-11-02 23:55

ftp : 140.116.245.92
Usermame : ic_design
Password : ic_design

5. If you have any problem, please contact by the TA by email :

Weiting84610@gmail.com 陳威廷
davidshih0908@gmail.com 施信宇