CSIE DIC Lab 2023

Final Project – Floating Point Number Multiplier

Due on: 17:00, Jun. 07, 2023

Before you start to work on this project, please read the following description, including system I/O ports, specifications, and requirements, carefully. Following the given instructions, and don't hesitate to contact the TAs if you find any ambiguity in the note or project description.

Floating Point Numbers:

Based on **IEEE 754** standard, the double-precision numbers are stored in 64 bits: 1 for the sign, 11 for the exponent, and 52 for the fraction. An exponent is an unsigned number represented using the bias method with a bias of 1023. The fraction represents a number less than 1, but the significant of the floating-point number is 1 plus the fraction part. In other words, if ***s*** stands for the sign bit, ***e*** is the biased exponent and ***f*** is the value of the fraction field, the number being represented is

$$(-1)^s \times 1.f \times 2^{e-1023}$$

It is worthwhile to take notice that when the biased exponent is 2047, a zero fraction field represents **infinity**, and a nonzero fraction field represents **NaN** (Not a Number). When the biased exponent and the fraction field are both 0, then the number represented is exact **0**.

For example, the following 64-bit word represents for $-1.25 \times 2^2 = -5.0$.

{1'b1, 11'b100_0000_0001, 52'h4_0000_0000_0000 }

Project Description:

The topic of this project is to design a pipeline multiplier for floating-point numbers. The architecture specifications and system requirements are given as follows:

1. Input Ports: **CLK, RESET, ENABLE, DATA_IN[7:0]**
2. Output Ports: **DATA_OUT[7:0], READY**
3. It is **active-high synchronous reset** architecture.
4. The rounding mode is **"round to nearest"**. In this mode, the representable value nearest to the infinitely precise result should be delivered.
5. **You should try to maximize the operation frequency** of this pipeline floating multiplier.
6. The output latency after data are input should be smaller than **60** clock cycles.
7. The data are input when **ENABLE** is "high". If **A[63:0]** and **B[63:0]** are two

double-precision numbers, it takes 16 clock cycles to input them to the multiplier. Also, for each double-precision number, the lower bytes are inputted first. The detailed input timing diagram is shown in Fig. 1.

8. As you start to output, the **READY** should be activated to ″high″. It takes 8 cycles to output the result. The lower bytes should be outputted first. In Fig. 1, **Z[63:0]** means the output result of your floating multiplier. This result is compared with the corrected result **C[63:0]** to check if your design is correct.
9. Your design must operate at **150 MHz** without any timing violation.
10. **You should prepare one report for TA in PDF or WORD format. You are required to describe the architecture/algorithm of your design as detailed as possible. Also, timing/area/power information is also required to be listed in the report.**
11. **Illustrate your design and how you can prove your design is correct or not.**
12. In this final project, you should prepare your test pattern, but you can find the example design and example test pattern from TA's directory:
    % tar xzvf ~dicta/final_project.tar.gz

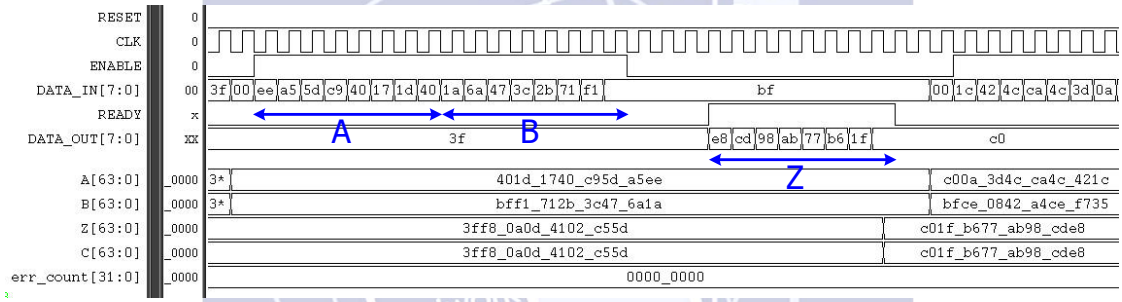

Fig. 1: Referenced timing diagram.

Demo:

1. **Demo1:** You must complete a floating-point multiplier design. RTL simulation of the floating-point multiplier should be correct. Besides, your design must be a synthesizable design.
2. **Demo2:** Gate-level simulation of the floating-point multiplier should be correct. Besides, your design must operate at **150MHz** without any timing violation in gate-level simulation.
3. **Demo3:** You must complete the Auto Placement and Routing (APR) of this final project and pass the Calibre DRC/LVS check. Besides, your design must operate at **150MHz** without any timing violation in post-layout gate-level simulation.
4. **Demo4:** Final Project report.

Grade:

Demo1: Correct RTL code (25%), Compliance with IEEE-754 (5%)

Demo2: Correct Gate-level simulation (25%)

Demo3: Complete APR and post-layout gate-level simulation and without error in Calibre DRC/LVS report (25%)

Demo4: Final Project Report (20%)

Note: **You should try to maximize the operation frequency** of this floating multiplier. The grade in Demo2, Demo3, and Demo4 will depend on the operation frequency of your design.

Note: The final project demo must be finished before **17:00, Jun. 07, 2023**.