Design of a Reconfigurable Multi-Precision Floating-Point Computation Unit for Dot Product of 4D Vectors

Handout: 2024/11/12

Due: 3 weeks later

1. (Multi-Precision Dot-Product)
   Given two 4-dimensional vectors $x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$ and $y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix}$ where the vector components x_i, y_i could be IEEE 754 single-precision or half-precision floating-point format, design a computation unit that performs the dot product (inner product) DP4 of the two vectors, i.e., calculate
   $$z = x_1 \times y_1 + x_2 \times y_2 + x_3 \times y_3 + x_4 \times y_4$$
   where the output z is also in IEEE 754 ***single-precision or half-precision*** floating-point format. The user can specify the format of operation, either single-precision or half-precision. Note that in case of half-precision operation, you can turn off the unused circuits to reduce power. For half-precision mode, use the 16-bit MSB part of a 32-bit single-precision data.
2. (Sub-Word Multiplication)
   Since your design needs to consider both single-precision and half-precision data formats, it is suggested that you use the half-precision components to construct those for single-precision. For example, if your single-precision datapath needs 24*24 unsigned multiplications for Q1.23 format, you can construct it using four 12*12 multiplications for mantissa of Q1.10 format because half-precision datapath only needs 10 fractional bits.
3. (Prime-Time for Clock Gating)
   Design your dual-precision floating-point DP4 such that in the half-precision mode, the unnecessary components are turned off using clock-gating to reduce dynamic power. Note that power report from Synopsys Design Compiler (DC) (static timing analysis) will not give the accurate power for half-precision inputs where clock-gated circuits are turned off because users do not need to provide input patterns to DC and thus DC does know how many inputs are half-precision. Use PrimeTime to measure the power in the half-precision mode. Assume that half of the inputs are of 16-bit half-precision format.
4. (Non-Pipelined and Pipelined Implementations)
   You can first design a non-pipelined version. Afterward, analyze the delays of various components and then insert pipelined registers at proper positions in order to reduce the cycle time with balanced delay in each pipelined stage. The following is a choice of 4-stage pipelining mechanism:
   - 1st stage: unpack, multiplication, alignment
   - 2nd stage: multi-operand addition, conversion to sign-magnitude
   - 3rd stage: normalization
   - 4th stage: rounding, pack
5. (References)
   The following references list some related papers. You can adopt some of the concepts in your designs. Explicitly mention what special design methods you have adopted in this homework. For example, you might consider using the leading one anticipator to reduce the critical path delay.

6. You have several weeks to do this entire homework. But you need to submit intermediate reports every week. The weekly reports show your progress in each week. The complete report dues as shown in the handout.
7. The report should clearly explain your design and how you verify your design. The following lists some of the information suggested to be included in your final report.
   (a) Problems encountered and how you solve the problems
   (b) Your special designs. These special designs will give you extra bonus scores.
   (c) Both RTL and gate-level simulations and your verification strategy. In other words, how can you make sure that the testbenches can completely verify the correctness of your designs?
   (d) Profiling of area and delay after the logic synthesis of Synopsys Design Compiler. Use two different constraints: area-optimization and delay optimization for logic synthesis. Area profiling is using a circle graph to show the area percentage for the key hardware components. Delay profiling is to identify the critical path, the delay, and the distribution (along the key hardware components) of the critical path delay. Also, give the report of power for the synthesis results.
   (e) Discussion of your observations for the synthesis results (area, delay, and power).

References:

1. D. Kim and L.-S. Kim, "A Floating-Point Unit for 4D Vector Inner Product with Reduced Latency", *IEEE Trans. Computers*, vol. 58, no. 7, pp. 890-901, July 2009.
2. J. Sohn and E. E. Swartzlander , Jr., "A Fused Floating-Point Four-Term Dot Product,", *IEEE Trans. Circuits and Systems-I*, vol. 63, no. 3, pp. 370-378, Mar. 2016.