CSE4210 – Architecture and Hardware for DSP

Project Task 3

FIR Filter Implementation

Introduction

Several techniques were introduced to improve the performance of DSP system including pipelining and parallel process. In this task, you will use the filter designed in Task 2 and explore the architecture level trade-offs in implementing the FIR filter in order to achieve the best performance, e.g. highest operation speed.

The design should be coded in Verilog, simulated and synthesized using Altera Quartus II software, and mapped to Altera FPGA Cyclone II EP2C35F672C6 (DE-2 Development Board). Based on synthesis results, the performance of the filter will be evaluated in terms of speed (delay), area, and power.

You will use Altera Quartus II software for simulation and Altera DE-2 Development Board as the target device. The Quartus software is available in LAS 3057. You can borrow the Altera DE-2 Development Board from the Lab Monitors at LAS 1006 during lab time (11:30am – 1:30pm). You need to return the board once the lab is over.

Tasks

1. Determine the filter architecture

Assume the filter input is restricted to a range between -1 to 1 and represented in 8 bits. Further assume that the delay components are implemented by D-flip-flop or FIFO which has negligible propagation delays compared to adders (you design in Task1). Note that the results from Task1, i.e. speed, area, power of different adders, form the base for your design trade-off. Given that the filter coefficients are presented by 8 fractional bits (total of 9 bits). Your team is required to evaluate the performance of 4 filters, i.e. (1) the baseline design (filter without pipelining and parallel), (2) pipelined design, (3) 2-level parallel design, (4) combined pipelining and 2-level parallel processing in one design.

Important note:
You should only use techniques covered in Chapters 1 to 6 to improve the performance of the filter.

You are required to predict the performance of these filters, and to identify which design is the best in terms of speed. Draw the block diagrams of all your designs.

2. Filter implementation and verification

Develop Verilog codes for your filters.

Develop test benches for the functional verification of these filters. Using Quartus II to perform your simulation and synthesis.

   a. Perform functional simulation to verify your designs.
   b. Map your design to Altera DE-2 development board, i.e. to select the target FPGA device that matches the DE-2 board.
   c. Perform timing analysis on your designs, considering the worst case. Record delays for each filter.
   d. From synthesis results, record area (gate counts) and power.

Report

Your report should include the following sections.

1. Introduction: A brief introduction of the theory about the project. A brief description of software tools used.
2. Task Management: A brief description on how your team is organized in performing this task, the role of each team member, the contributions of each team member.
3. Design Procedures: Detailed descriptions of the design procedures including architecture trade-off, and performance prediction.
4. Filter Implementation: Detailed description of your HDL design, simulation setup, simulation results, verification, and comparison figures or tables.
5. Discussion: Does your prediction match the implemented filter? Discuss the pros and cons of techniques used in improving the performance.
6. Conclusion
7. Appendix including Verilog code, test benches, and compilation reports.

Resource

1. Tutorials and labs on Quartus II are available at:
http://www.altera.com/education/univ/materials/digital_logic/labs/unv-labs.html

2. FIR filter design using Matlab:
http://www.mathworks.com/help/dsp/examples/designing-low-pass-fir-filters.html