McGill University
Department of Electrical and Computer Engineering
Computer Architecture Laboratory (ECSE 487 – Winter 2009)

Assignment 1

Due January 26, 2009, 2:30 pm.

- This assignment is to be done individually. Submission is done both by WebCT Vista (myCourses) and in the assignment box in Trottier. The submission is not complete without both hardcopy and electronic copy and the time of the latest will be the submission time. Submit your report in PDF format.
- Please read the entire handout carefully before starting on the assignment to get an idea of the big picture.

Introduction

The goals of this assignment are:

1. To develop a moderately complex hardware design using both behavioral and structural models in VHDL.
2. To become familiar with the simulation and synthesis software design flow.
3. To learn how to develop a VHDL testbench for verification of your design.

A VHDL *behavioral* model specifies a module's function using VHDL programming constructs such as "if-then-else" clauses, etc. It does not specify any internal structure. A VHDL *structural* model specifies a module's internal structure by identifying all internal modules or units (call these "sub-modules") and defining how they are interconnected using the "port map" construct. It does not specify any internal behavior.

The sub-modules in a structural description are distinct entities and they may be specified either structurally or behaviorally (or both). This assignment will illustrate these concepts.

Design Problem

We consider the design of a shifter. Shifters are needed in many places such as in integer ALUs to implement shift instructions, in floating point units, etc... While a first implementation approach could be a bidirectional shift register with parallel load, often one employs combinatorial shifters known as *barrel shifters* which have a delay proportional to the $\log_2$ of the word size. We want a shift unit (Figure 1) that can perform

the following operations needed to implement instructions for an 8-bit computer: shift arithmetic right (`S = "11"`), shift logical right (`S = "10"`), shift logical left (`S = "01"`), and rotate (`S = "00"`), all from 0 to 7 places (`N = [000, ..., 111]`).

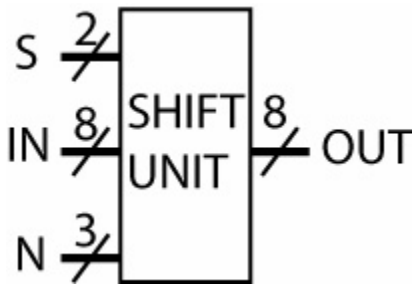

Fig. 1

You will develop two versions of this shift unit, one which is purely combinatorial and the other which is pipelined for speedup. To see how a barrel shifter is constructed, we start from a 2-bit word and design a circuit to conditionally swap the bits as per Figure 2 which can be viewed as rotating one place. For a 4-bit word (Figure 3) we can do shifts by any amount ("rotate" in this example) in one direction by combinations of one and two place shifts (as in binary numeration: `N="N1N0"` determines the number of places shifted). So if `N` is the word size, then we can combinatorially shift by any number of places with $\log_2(N)$ layers.

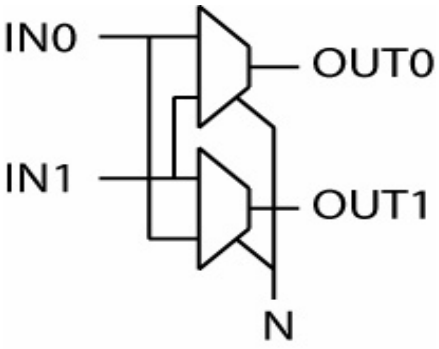

Fig. 2

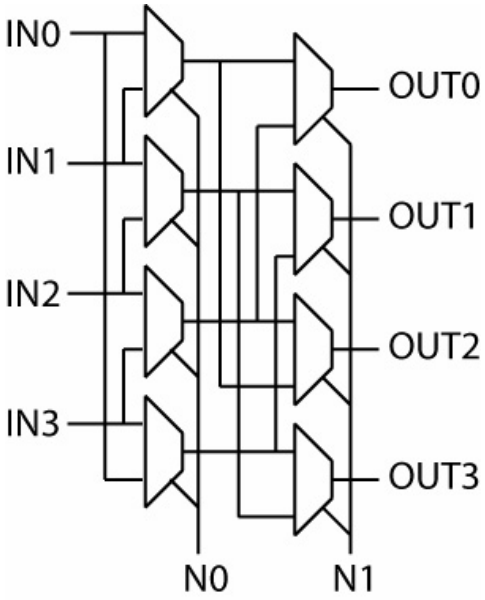

Fig. 3

This scheme can be expanded and adjusted to provide the four shift functions required by the shift unit of Figure 1. Once the design is complete, the shift unit will have a layered structure which lends itself to pipelining. So we will also develop a pipelined version of the shift unit as shown in Figure 4.

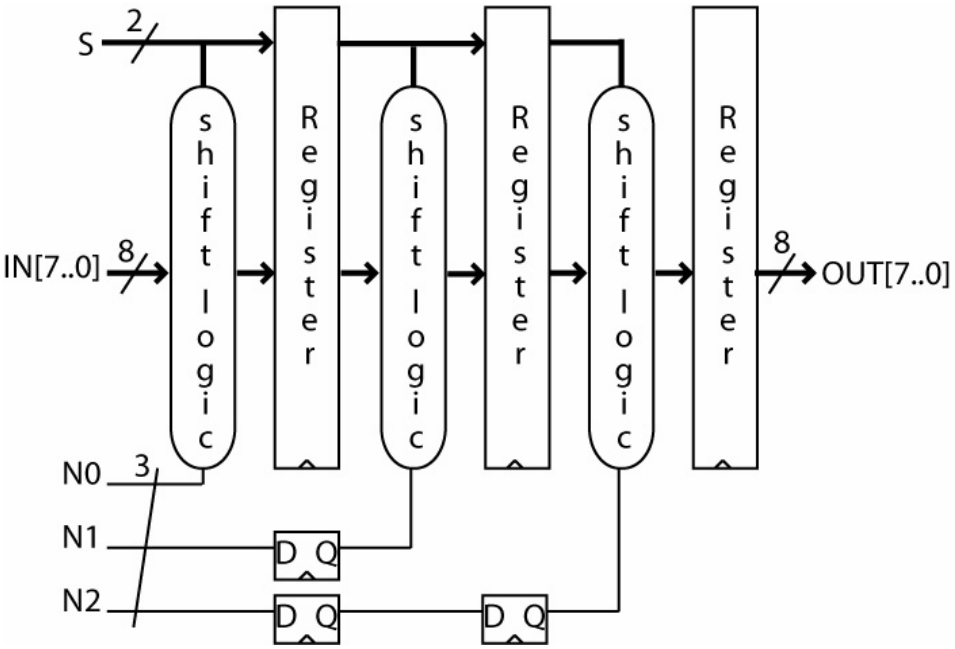

Fig. 4

Assignment

Behaviorally, designs can be specified in terms of input-output properties and signals or variables can be used to represent registers. Structural models are a description of a circuit in terms of combinatorial components and storage elements. You can refer to the course textbook Sections 5.3 and 5.4. You are to write the descriptions in synthesizable VHDL (therefore you can't use the "`after`" statement). Please refer to Appendix A of the course textbook for details on synthesizable VHDL.

Part 1: Design and test an 8-bit 4-function barrel shifter. Note that there are tradeoffs to provide for left and right shifts (hint one way to do that is to reverse the order of the input and output bits, but there are others).

Design the shifter using:

a) A purely behavioral model
b) A structural model. Justify which sub-elements you described behaviorally in this model and the reasons why.

Part 2: Design a pipelined version of the barrel shifter. The diagram in Fig. 4 has three stages which is probably optimal in the case of a single function. However, you should select the design and number of stages to maximize throughput in all four functions. Select a modeling style (behavioral or structural) that makes sense for your design, but be sure to justify your choice in your report.

Verification

An important part of the design process is verification of your design. In fact, it often takes just as long to verify a design as the design itself. Verification should not be an afterthought, but instead an integral part of the design process. To automatically verify your designs, develop VHDL *testbenches* that test your designs (both Part 1 and Part 2). It is easy to see that this shifter can be *exhaustively* tested with every possible input vector and configuration of the circuit (note that in general this is ***not*** true and a big part of testing is choosing a small subset of input vectors that cover the important cases). However, the number of input vectors is still in the thousands (I'll leave it to you to calculate exactly how many) and therefore you will want to write a computer program (C/Java) to automatically generate the input vectors and expected output. The testbench will read these from a file, apply the input vectors to your circuit and compare the result of your circuit with the expected results and then report if they are not equal. Note that the testbench itself is just for simulation and will not be synthesized, therefore it does not need to be restricted to synthesizable VHDL.

Do not underestimate the effort required to do the verification (but also you will recognize how useful, in fact indispensable this procedure is! **You probably should start developing a testbench right away.** This will be used in debugging your circuits as you develop them.

Synthesis

You have to synthesize your design without the testbench. Discuss the following aspects of your design:

- Perform several synthesis runs. Optimize for maximum speed and then for minimum area. Discuss the difference between the two implementations such as:
    - Maximum clock speed and the critical path
    - Number of logic cells used and related it to the VHDL code and the schematic view from the synthesizer

In your report, give a detailed discussion about the results of the synthesis of both approaches, including the used logic cells, look-up tables and the latency of your circuit. Identify in your design the part that takes the most space and the longest path of the design. Discuss the improvement you will have to make if you need to make your design smaller, or to make it run faster.

Design Tips

- Start small. Don't try to implement the entire circuit at once. Develop a little piece and test it. When it is working, then add some more functionality a bit at a time. There really is no other way to get it working.
- Use the up-front effort required to develop a testbench and automatic test vector generation to your advantage! Typically, once I define the functional behavior of my design and port maps of my circuit, I start working on the testbench. Once it is done, it can be used throughout the design and test cycle. You can start with a testbench and a few critical test vectors you come up with by hand to stress the part of the design you are working on. Later, you can use the testbench to run through the automatically generated test vectors and use it to automatically report compliance.
- "If you can't draw it won't work". ALWAYS draw schematics (RTL level, not gate level) of the circuits you are trying to model.

Laboratory Report

The report should consist of the following sections:

1. A front page (course number, lab title, student name, ID, email, date).
2. Lab description (state the design problems in your own wording).
3. For each part:
    1. Description of the methodology followed (including any figures that help document your design).
    2. Documented VHDL source code for all entities.
    3. Simulation results (traces) to show that your design operates correctly. Please include documented vector files if they are used. Make sure your simulations sufficiently cover all significant cases. Each simulation should be legible and briefly described.
    4. Testbench including the VHDL and the software code you used to generate test vectors.
    5. Synthesis for maximum speed and minimum area.
    6. Summary of resources and performance achieved in term of throughput and latency.
    7. Part 2 only: Discussion of how well your design meets its speedup objective.
4. Comment on the differences and usefulness between behavioral and structural descriptions. Suggest when each type of description is useful. Comment on the advantages of having both types of descriptions in VHDL, i.e. why did the designers of VHDL support both descriptions? Are there any drawbacks of having both descriptions? Are there situations where both descriptions might be useful?

Figures you provide in your report should be drawn by you, and not scanned or cut and pasted from other sources. If you essentially reproduce a figure from another source, even if you re-draw it yourself, remember to cite the source.