ECE 414 Computer Organization and Design

EXPERIMENT 4. DIVIDER

Steps	Grade
1	/1.0
2.a	/2.0
2.b	/2.0
3.a	/1.0
3.b	/2.0
3.c	/1.0
4	/1.0
Total	/10

Submitted by:
on:
to : Prof. Onur Tigli

Objective

In this experiment, you are required to apply datapath and control unit design techniques in Verilog HDL to design and optimize a division algorithm. Coding, verification, simulation, synthesis and timing analysis will be carried out using Xilinx ISE tools. Verified designs will be implemented on Digilent's NEXYS-4 FPGA boards for demonstration of correct operation.

Procedure

Given two unsigned n-bit numbers A and B, design a divider circuit which produces two n-bit outputs Q and R, where Q is the quotient (A/B) and R is the remainder.

This operation can be executed by shifting the digits in A to the left, one digit at a time, into a shift register R. After each shift operation, compare R with B. If R ≥ B, a 1 is placed in the appropriate bit position in the quotient and B is subtracted from R. Otherwise, 0 bit is placed in the quotient. Following pseudo-code describes the division algorithm:

```
R=0;
for i = 0 to n-1
        Left-shift [R|A]
        if R≥B then
                q_i = 1;
                R= R-B;
        else
                q_i = 0;
        end if;
 end for
```

1. The datapath section of your design should contain the following components

- n-bit shift registers for A, R, and Q
- n-bit register for B
- subtractor for R-B
- counter to count down the number of bits

You can utilize modules you designed in previous experiments for building your datapath. Design this datapath circuit as a separate module in Verilog by using instances of the listed components.

2. The control circuit
   a. Derive an ASM chart that lists down the states and the operations that take place in your divider design. Define your "status" – output from datapath/input to controller- and "control" – output from controller/input to datapath – signals.
   b. Design this controller circuit as a separate module in Verilog. Run a functional verification that shows all the inputs and outputs of the controller working correctly.

3. Divider
   a. Design a top-level Verilog module that connects the datapath and the controller.
   b. Develop a testbench that runs the following divisions: 100/20; 55/7. Verify the correct functional operation by presenting snapshots of sample waveforms.
   c. Using the Xilinx ISE Synthesis tool, synthesize the design and obtain device utilization summary. Obtain the RTL and technology schematics showing the block level and register level implementation of your design.

4. Generate the programming file and download your design to NEXYS-4 board. Verify the operation of your design by applying input combinations using the switches/push buttons and observing the outputs on the LEDs or seven segments.

References

[1]: Brown, S., Vranesic, Z., "Fundamentals of Digital Logic with VHDL Design," McGraw Hill, 2000.