CMPEN 270 Laboratory 7 ***2020***

In this lab you will work on:

1. *Dataflow*
2. *Behavioral Design*
3. *Modular design*
4. *Module Integration*

Problem Description:

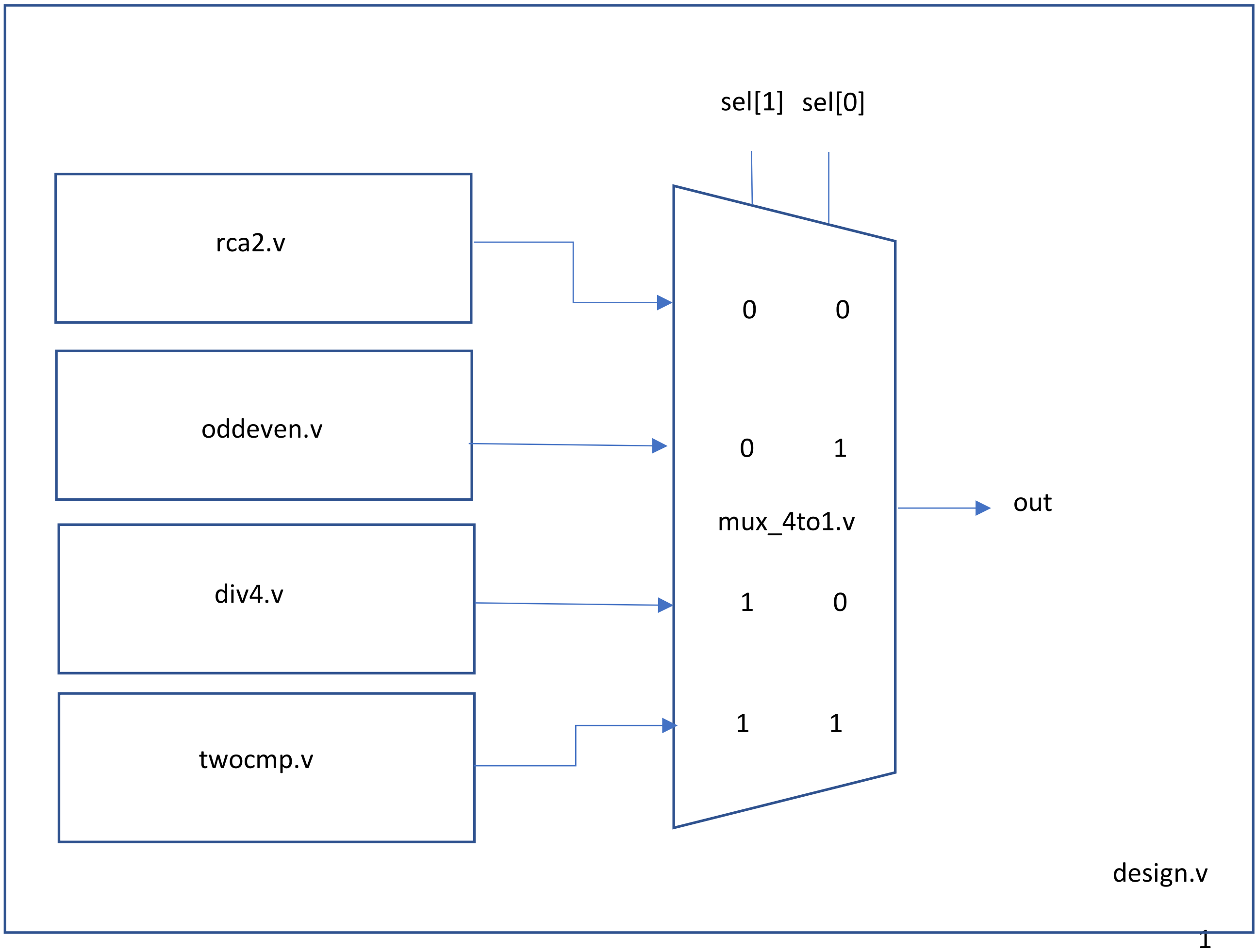
In the last lab you were introduced to the Verilog HDL and its versatility. In the previous video lecture, you were introduced to the behavioral modeling. In this lab we will focus on the dataflow and behavioral modeling, and module integration. For this purpose, we will solve a near real-life problem description.

A customer wants you to design the following system:

Assume that you have got an 4bit number as an input. Design a 4:1 multiplexer which outputs...

- the sum of the two halves of the 4bit input using a 2bit ripple carry adder when the selector = 0
- whether the 4bit input is odd or even when the selector = 1
- the input divided by four when the selector = 2
- 2's complement of the input when the selector = 3

Your Project should follow the following diagram:



Given this is going as a prototype in a control block, the customer asks it in the following format:

EDA Playground link with starter files: https://www.edaplayground.com/x/DhmJ

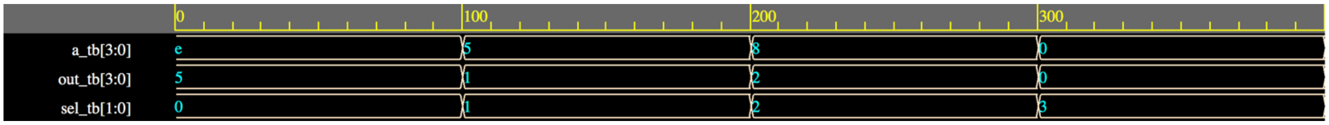
1. A module to find if a 4bit number is odd or even. Name this module as **oddeven.v**. Module oddeven.v takes a 4bit number as input and gives a 4bit output. The output is 0001 if the number is odd and 0000 if it is even.
2. A 2bit ripple carry adder. Name this module **rca2.v**. This uses TWO 1bit half adders to form a 2bit ripple carry adder. Remember that the sum of two 2bit numbers might generate a 3bit number. In case you are running behind, you can design a simple 2bit adder. In that case call your module **ADD2.v**. Just a note that the individual with a RCA will get more points than an individual with ADD2.
3. A divided by 4 module. Call this module **div4.v**. This module takes a 4bit number as an input and gives a fourth of the number as an output (4bit).
4. A 2's complement module. Name this module **twocmp.v**. This takes a 4bit number as input and gives the 2's complement of the number as output.
5. A 4:1 multiplexer module. Name this module **mux_4to1.v**. This takes four 4bit inputs as the input to the multiplexer , one 2bit input as the selector for the multiplexer, and one 4bit output as the output of the multiplexer.
6. The top-level module for this project will be called **design.v**. This takes an 4bit number as an input, a 2 bit number as a selector input, and gives a 4bit number as an output according to the problem statement description.
7. It must be clear by now that the design.v will contain all these other modules inside.

Example input/Output:

1. In: a = 14 = 4’b 1110, sel = 0 = 2'b00; out = 5
2. In: a = 5 = 4’b 0101, sel = 1 = 2'b01; out = 1

Solution Waveform:

You will need to check your solution using the waveforms for this lab. There will be a video posted to help explain creating, using, and reading the waveforms in EDA Playground. Below is the solution waveform.


0
100
200
300
a_tb[3:0]
e
5
8
0
out_tb[3:0]
5
1
2
0
sel_tb[1:0]
0
1
2
3