Lab 2: More Combinational Circuits
CDA 4203L
TA: Andrew White
Sept 8, Sept 10

Overview
In this lab, we will be creating a Priority Encoder.

Logic Design Refresher:

Priority Encoder
A priority encoder accepts data from its inputs and provides a binary representation on the outputs. A priority is assigned to each input so that when two or more inputs are simultaneously active, the input with the highest priority is represented on the output, where signal N-1 on an encoder with N inputs has the highest priority. For example, on an 8-bit priority encoder, if inputs input0 and input6 are high, then the 3-bit output will be "110."

VHDL Review:
IF/ELSIF/ELSE Conditional Statement (taken from Lab1):
It is good design practice to always have a full conditional branch – meaning always have an else part.

ex.

```
if clear = '1' then
        Q <= '0';
elsif clock = '1' then
        Q <= D;
else
        -- Do something else here if you want!
end if;
```

Lab Procedure:
This lab is to be completed in behavioral VHDL (use a process again). Implementation using conditional and/or case statements is easier then with Karnaugh maps, boolean equations, and simple gates.

The Priority Encoder:
Create an eight input (three bit output) priority encoder. Use a IF/ELSIF/ELSE conditional statement for implementation, as it implies priority through the order that the conditions are placed. For the output, use a linear array of three bits for representation:

```
std_logic_vector(2 downto 0)
```

This is done by specifying the MSB as 2 and the LSB as 0 for the output signal port

when creating the new VHDL source file with the Xilinx tools. To assign a value to it, use double quote characters to surround the value you are assigning in the signal assignment statement:

array_of_8_bits <= “01011100”;

Inputs can either be in the form of eight individual inputs of type *std_logic*, or an array of 8 bits as a *std_logic_vector*. **NOTE:** Specifying the inputs as an array of 8 bits will make simulation in Modelsim simpler when assigning values using force statements. The format of the Modelsim **force** statement for an 8-bit array of type *std_logic_vector* with 0ns delay is:

force *array_of_8_bits “010111000” 0ns*

If no inputs are asserted, then assign the output high impedance values (a *Z* for each element or “bit”)**.**

Follow the same procedure as the first lab to create the project and source file within Xilinx ISE tools. Once the encoder has been created, synthesize it to verify proper syntax. Refer to lab1 for the proper procedure for creating a project, adding a new VHDL Source File, and synthesizing a design.*

Simulate the Priority Encoder using Modelsim. All of the inputs will need to have a stimulus applied using the *force* command, this includes an initial value at the start of the simulation. DO NOT USE THE REPEAT OPTION (*-repeat after_some_delay*). Exhaustive testing is not necessary for the design, but adequate testing is required: multiple encoder input combinations (some high, some low) to demonstrate proper function of circuit. Apply **at least** five different input combinations.

To apply multiple values to each input, the values of the inputs can either be specified for the entire simulation in one *force* command using the “*logic_value after_delay”* pairs, all comma separated, or by specifying the values in individual *force* commands and running the simulation with the *run* command for some period before changing the input value with another *force* command and running again:

1. set initial input values with 0ns delay
2. run simulation for period
3. set next input values with 0ns delay
4. run simulation for period
5. set next input values with 0ns delay
6. run simulation for period
   ... until circuit has been adequately tested

Remember, the syntax for the force command is:

force [**-repeat** *after_period*] *signal_name logic_value after_delay* [, *logic_value after_delay, ...*]

Lab Report

For the Lab report, create a cover sheet with your Name, the Lab Number of current

lab, Course Name, and what Day and Time you attend lab (ex. MW 3:30 – 4:45). This format for cover sheet will be used for all following labs. Print a copy of your VHDL source file and your waveform from simulation. On the waveform, indicate at each test input combination transition, which input has priority and what the output value is in decimal. This can be done by marking with a highlighter or by clearly indicating with a pencil. This is primarily for you to become familiar with reading the waveforms and verifying the functionality of your own circuit. Submit the cover sheet, VHDL source file, and the waveform from Modelsim to the TA by the end of the second lab period for this lab.

*Course web page:
http://www.csee.usf.edu/~smohanty/teaching/ComputerSystemDesignFall2003/