Computer Assignment 5

ECGR 2181 - Fall 2015 – Version 2 (minor typo fixes)

Assignment Overview

In this assignment you will use Vivado 15.2 Webpack to write HDL, simulate (optional) and program an FPGA. In this CAD, you will be creating a guessing game. One CAD partner will input a number using the switches, set it with btnR, then the other CAD partner will guess the number and test to see if it's true with btnL. When btnL is pressed and the values match a LED lights up and if they values don't match the LED is turned off. The LED stays high/low until you test it again. You'll need use the debounce.vhd for the buttons (provided for Lab 4).

Background

You need to declare the debounce component, then instantiate it like in the previous lab. In this lab you'll only use 8 of the switches. **Simulation for this lab isn't required** since you have to demo it, it's only to help you debug the code.

Drivers for the FPGA Board

The FPGA board requires some drivers for it to be recognized by Vivado. You need to download and install Digilent's Adept System software which can be located at https://www.digilentinc.com/Products/Detail.cfm?NavPath=2,66,828&Prod=ADEPT2

Getting started

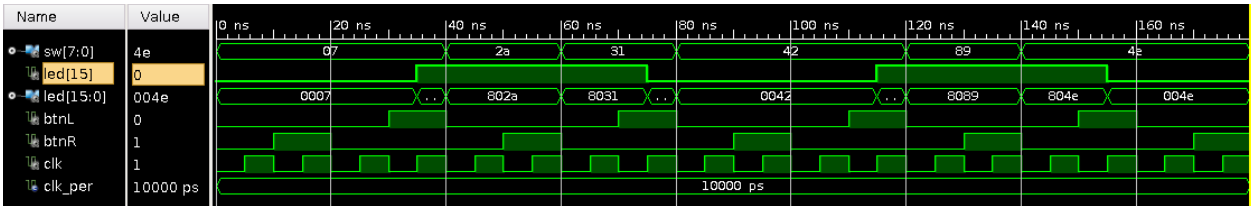
1. Create a directory to store all of the assignment's files.
2. Open Vivado 2015.2 and create a new project called **computer_assignment_3**.
3. It will be an RTL project
4. Add the debounce.vhd from Dr. conrad's website..
5. Add the constraints/xdc file.
   a) DL https://www.digilentinc.com/Data/Products/BASYS3/basys3_master.zip .
   b) Extract the zip file.
   c) Add the extracted file as a constraints/xdc file.
6. The Basis3 uses an **XC7A35TCPG236C-1** FPGA
   - Artix 35T
   - Speed grade -1
   - CPG236 packaging
   - C temp grade

Task

This task is to write and simple game where one person sets a number/value and the other person tries to guess it.

1. Edit the constraints file and uncomment the lines corresponding with:
    - sw (7 down to 0)
    - btnR
    - btnL
    - clk
    - led (15 downto 0)
2. Create the top.vhd file:
    a) Under "Flow Navigator" click "Add sources."
    b) Select "Create or add design sources."
    c) Create top.vhd and then click "Finish."
    d) Create inputs:
        - sw ~ (7 downto 0)
        - btnR
        - btnL
        - clk
    e) Create outputs:
        - led ~ (15 downto 0)
3. Once you have your vhdl source file:
    - Add two debouncers, one for each button.
    - Connect the sw to led(7 downto 0).
    - Create the logic needed to store your sw value when you set it using btnR.
    - Create the logic needed to test the stored and sw values and set the led(15) accordingly using btnL.
4. Make sure it's syntax error free and can be synthesized.
5. *Add the top_tb.vhd file:*
    a) Under "Flow Navigator" click "Add sources."
    b) Select "Create or add simulation sources."
    c) Add top_tb.vhd.
    d) Make sure simulation set is "sim_1."
    e) Click "Finish.
6. *Comment out the debounces and connect the btrR and btrL signals to their corresponding signals,*
7. *Select the right simulation set.*
    a) Under "Flow Navigator" click "Simulation Settings."
    b) Under "Simulation set" input "sim_1."
    c) Under "Simulation top module name" input "top_tb."
    d) Click "Ok."
8. *Run the simulation by and check your results. The expected output is on the last page of the assignment.*
9. *Edit top.vhd and uncomment your debouncers back in.*

10. Create the file you need to load onto the FPGA (the bitstream). Click “Generate Bitstream” under the “Flow Navigator.”
11. Program the FPGAG board
    a) “Hardware Manager” in the Flow Navigator.
    b) “Open Target” and then “Auto Connect.”
    c) “Program Device” and then “xc7a35t_0.”
    d) The defaults of the pop up are ok, just press program.



Name	Value
sw[7:0]	4e
led[15]	0
led[15:0]	004e
btnL	0
btnR	1
clk	1
clk_per	10000 ps

Figure 2: Expected Waveform of Simulation

Turn in

The write up is very simple, just include things like: the basic design process, anything you had trouble with, an explanation of anything that doesn’t work, etc. It isn’t meant to be extensive, but a brief overview. **Include top.vhd, or any code you wrote in the Moodle turn in**.

Make sure everything is in a single PDF on Moodle. You can join PDFs with Adobe Acrobat if you have multiple PDFs. Name the assignment CAD5-xxxxxx-yyyyyy, where xxxxxx is the last name of one CAD partner and yyyyyy is the last name of the other CAD partner. Turn in only one pdf under one of the CAD partner’s Moodle account.

Grading

Write up	/8 Points
Printout and correctness of code	/12 Points
Demonstration of FPGA	/20 Points
	Total of 40 Points

Demonstration

The demonstration of the code is simple. Brian will be the TA you need to demonstrate the VHDL running on the FPGA to. Fill out the demonstration sheet, print it, and fill out the self-review before the TA demo. You need to be able to set and test numbers with the switches correctly and led(15) lights up when a match is found (after testing it).

Demonstration Grading:

Set the number correctly	/8 points
Test the number correctly	/8 points
All LEDs are connected to the correct signals	/4 points
	/Total of 10 point