

## Computer Assignment 2

### ECGR 2181 - Fall 2015

#### Assignment Overview

In this assignment you will use Vivado 15.2 Webpack to write and simulate a hex to seven segment display convert. You'll also need to turn in a single PDF to Moodle (instructions below).

#### Background

A seven segment display is a way to display a decimal or hex digit. There are various ways to label the segments of the display or encoding the digit, but below is what we're using for this computer assignment.

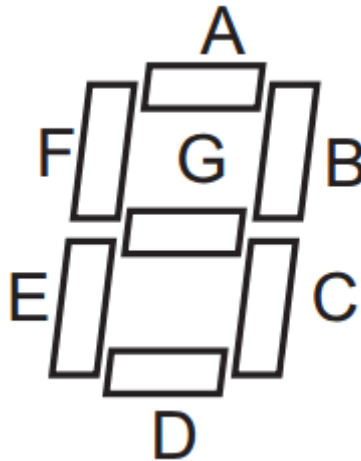


Figure 1: Seven Segment Display Layout

source <http://www.lucky-light.com/LED%20Displays/Four%20Digit%20Display/KW4-281.pdf>  
(seven segment display on the Basys3)

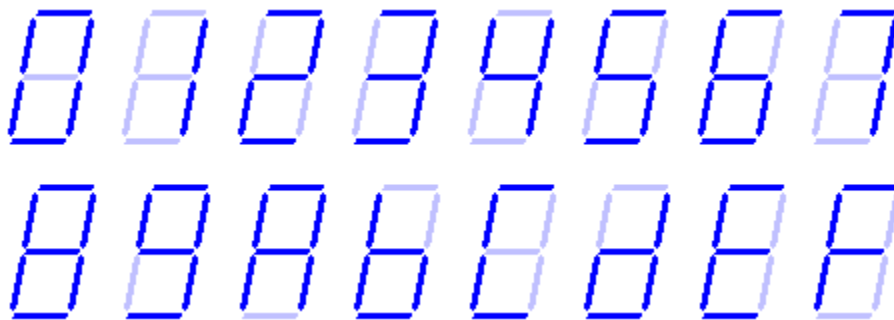


Figure 2: Hex Encoding

source [http://www.faculty.etsu.edu/tarnoff/labs2150/dig\\_apps/sevenseg.htm](http://www.faculty.etsu.edu/tarnoff/labs2150/dig_apps/sevenseg.htm)

in(3)	in(2)	in(1)	in(0)	A	B	C	D	E	F	G
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	1	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	0	1	1
1	0	1	0	1	1	1	0	1	1	1
1	0	1	1	0	0	1	1	1	1	1
1	1	0	0	1	0	0	1	1	1	0
1	1	0	1	0	1	1	1	1	0	1
1	1	1	0	1	0	0	1	1	1	1
1	1	1	1	1	0	0	0	1	1	1

Table 1: Truth Table of the Encoder

## 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\_2**.
3. It will be an RTL project
4. No sources to input. Although if the target and simulation languages aren't set to vhdl, change them to vhdl.
5. No existing IP.
6. No constraints file.
7. The Basis3 uses an **XC7A35T-1CPG236C** FPGA
  - o Artix 35T
  - o Speed grade 1
  - o CPG236 packaging
  - o C temp grade

## Task

This task is to implement a encoder which input is a 1 4-bit hex signal and the output is 7 1-bit signals for each display segment (A-G).

1. Create the encoder.vhd file:
  - a. Under "Flow Navigator" click "Add sources."
  - b. Select "Create or add design sources."
  - c. Create task1.vhd and then click "Finish."
  - d. Create an input named "hex\_in" which is 4 bits wide.
  - e. Create 7 outputs named "A" through "G."
2. Once you have your vhd source file, edit it to implement the encoder.
  - A when/else or if/else statement might be helpful
  - A internal signal of the outputs grouped together as a bus might help clean up the code
3. Make sure it's syntax error free and can be synthesized.
4. Add the encoder\_tb.vhd file:
  - a. Under "Flow Navigator" click "Add sources."
  - b. Select "Create or add simulation sources."
  - c. Add encoder\_tb.vhd.
  - d. Make sure simulation set is "sim\_1."
  - e. Click "Finish."
5. 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 "encoder1\_tb."
  - d. Click "Ok."
6. Run the simulation by and check your results. The expected output is on the last page of the assignment.
7. Vivado doesn't include a way to print test bench results so you will have to take a clear screenshot of at least one iteration of the circuit.

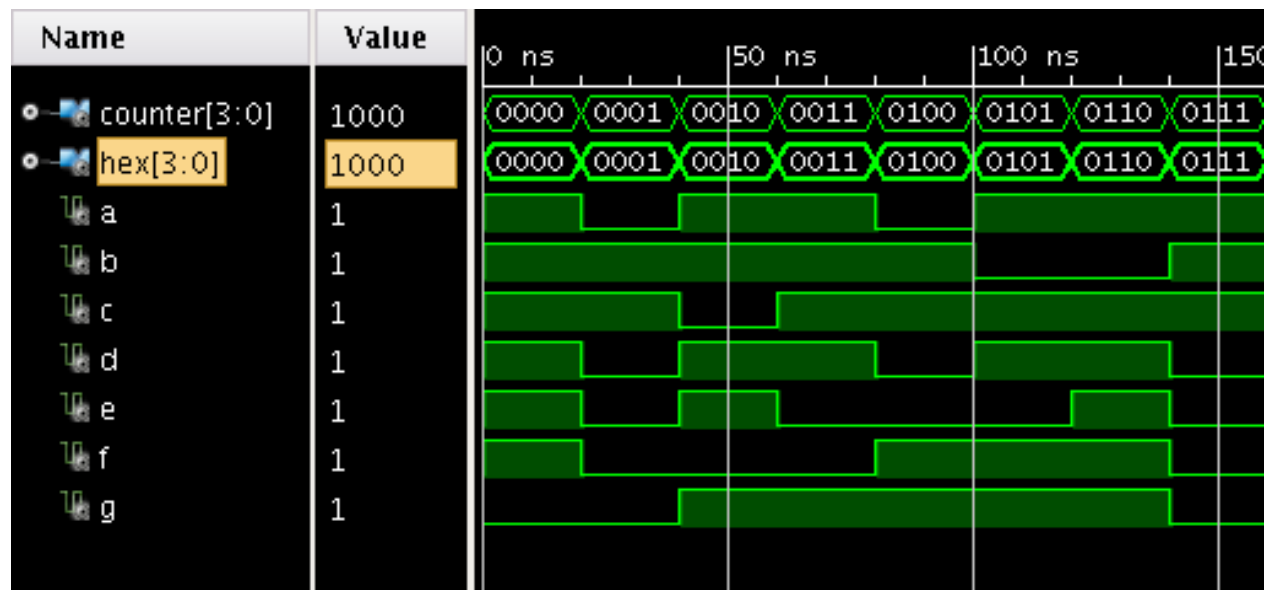
**Turn in**

The write up is very simple, just include things like: the basic design process, anything you had trouble with, an explanation of anything doesn't work, etc. It isn't meant to be extension, but a brief overview.

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 CAD2-xxxxxx-yyyyyy, where xxxxxx is the last name of one lab partner and yyyyyy is the last name of the other lab partner. Turn in only one pdf under one of the lab partner's Moodle account. Make sure everything is in a single PDF on moodle. You can join PDFs with Adobe Acrobat if you have multiple PDFs.

**Grading**

Write up	/4 Points
Printout of encoder.vhd	/8 Points
Printout of encoder_tb simulation results	/8 Points
	/Total of 20 Points



Sample of the Expected Waveform