ECE 3829: Advanced Digital System Design with FPGAs
A Term 2020

Lab 2- VGA Display
Lab signoff and Report due Sunday September 20th 4pm

Create a display on a VGA monitor using the Basys3 board.

The lab involves the use of the Xilinx Core Generator (for the MMCM), existing IP (the Digilent VGA controller), as well as combinational and sequential logic elements. There are multiple parts to this project. To be successful will require a good design and debugging approach. Make simpler projects that you can test and debug separately, and then combine them together.

This lab (and report) is to be completed individually – it will be worth 35% of your course lab grade.

Lab Signoff Deadline: during a lab section (during the week of September 14th). The TAs will use the signoff sheet to record what you have working. Submit your Verilog listings to Canvas at the same time as lab signoff. The lab report is due Sunday September 20th at 4pm (no late reports accepted).

Description

Part 1: Seven Segment Display:

- Modify the simple seven segment display from lab 1 to create a seven_seg module that can display a value from "0000" to "FFFF" on the four seven segment displays. The input to the module should be a 16-bit wide bus, with four bits used to indicate the value to be displayed on each of the seven segments. You will also need a clock to cycle through the four digits.
- Make this a separate module – you will use this module in this and later projects.
- Test this out by using the 16 slide-switches to enter various numbers (0000 to FFFF).

Part 2: VGA display

- Use a MMCM to create a 25MHz clock required for the sequential logic in this lab.
    - (see the MMCM tutorial for how to add this IP to your design).
    - Note: only connect the 100MHz FPGA clock to the MMCM (nothing else)
    - Add a period constraint to your XDC to match the Basys3 board 100MHz clock frequency.

- Create a VGA display using the VGA controller provided by Digilent (just the 640 by 480 version) – see information at end of this doc.
- Use the slide-switches to select and display one of the following patterns

- Complete green display
- Vertical bars of alternating red and yellow colors with each horizontal bar 16 pixels wide
- A black screen with a large blue block 64 pixels wide by 64 pixels high in the top right corner of the screen
- A black screen with a white block 32 pixels wide by 32 pixels high in the top left corner of the screen

(These should be relatively easy once you start working with the VGA controller provided by Digilent – do not forget to include the 'blank' signal)

Part 3: Moving block

- Assume the white block starts at position (0,0)
- Create a counter 'x' that counts from 0 to 19 at 1Hz rate
- Create a second counter 'y' that counts from 0 to 14 at 1Hz rate
- Use these two counters to move the white block one-block to the right (as 'x' increments and one-block down (as 'y' increments).
- You can 'wrap-around' when you reach the edge of the screen or just stop.

Extra credit

Up to 10% lab bonus points for any good improvements or enhancements to your design (must demo on board **and** describe well in your report). For example, print ECE3829 on the VGA monitor or make a Pong game!

Reference Material

Read the *Seven Segment and VGA Port* section in the Nexys3 Reference Manual.

Notes:

Your final design should combine parts 1, 2 and 3.

Demo your system during one of the lab sessions before the deadline. Have your Verilog source files ready so they can be uploaded to Canvas immediately after the demo (don't forget to includes name, description, and comments).

Write a report including: an introduction, a description of your design including good block diagrams showing how you implemented the design, a section describing how many flip-flops your design used and why. Include part of the synthesis file that displays warnings – copy and explain these. Include a conclusion describing any problems or issues you had, and lessons learned. Include your signoff sheet and your source files in an appendix.

Grading Guidelines

- [50 pts] Implementation
  - [50 pts] Design works on board and meets requirements
- [20 pts] Source Code – Verilog in Appendix
  - Code style and comments (well-commented and tab-indented code!)
  - Use of *case* vs. *if*, spaghetti code vs. structured, etc.
  - Recognizable implementation of "standard" elements (state machines, counters, shift registers, clock dividers, decoders)
  - Good modular design
  - No latches or other synthesis problems
- [30 pts] Lab Report
  - [5 pts] Brief Introduction / Problem Statement
  - [15 pts] General overview of approach to solution and description (include Block and State Diagrams with descriptions) and oscilloscope pictures
  - [5 pts] FPGA resource usage (# flip-flops with explanation) and listing and explanation of warning messages (don't copy all the Xilinx reports – just the relevant sections)
  - [5 pts] Conclusions
    - Problems faced in implementation
    - Solutions used to solve problems
    - Lessons learned from the project
    - Suggestions for further improvements and extensions
- [10 pts] Extra points
  - Possible extra points for good additional features or capabilities (need to demo on board and include description in report)

ECE 3829: Lab 2 sign-off sheet

Name: _____

Part 1 (Seven Segment Display)

The seven segment display works (0000 to FFFF) _____

Part 2 (VGA)

The VGA displays a green screen _____

Vertical bars (red and yellow, 16-pixels) _____

Blue block (top right corner 64-pixels) _____

White block (top left corner 32-pixels) _____

Part 3 (Moving block)

The white block moves right and down every one second

_____

All combined

All parts are combined into one project _____

Extra Credit (describe)

_____

_____