University of California, Davis
Department of Electrical and Computer Engineering

EEC180B DIGITAL SYSTEMS II Spring 2006

LAB 4: Graphics Display Design

Objective: In this lab you will design, simulate, synthesize and download a Pong-type video game. Your design will be implemented and tested on the Xilinx Spartan-3 board interfaced to a VGA monitor. One objective of this lab is to give you an understanding of the basic principles of VGA video display generation.

Pre-lab (50 points)

This lab is based on Ch. 9 and particularly Section 9.14 of the text Rapid Prototyping of Digital Systems by Hamblen and Furman. Study that chapter carefully before attempting this lab. You should also complete the Synplicity/ModelSim/Xilinx Tutorial II before attempting this lab since it covers many of the issues you will face in this lab. Complete the following tasks before starting on the lab:

1. Do the exercises specified in Tutorial II and have your work verified by your TA.

2. Create a video graphics display with three lines as follows:
   - A one-pixel-wide vertical red line down the center of the monitor
   - A one-pixel-wide horizontal blue line across the center of the monitor
   - A one-pixel-wide diagonal green line from the top left to the bottom right of the monitor.

   All three lines must intersect at a single point in the middle of the VGA monitor. Use the vga_sync.vhd module in your design. Demonstrate your design to your TA.

System Specifications

In this lab, you will extend the bouncing ball example given in Section 9.14 of the text by Hamblen. You will add the following enhancements to the bouncing ball design:

1. Allow the ball to move in both the X and Y directions and to bounce off all four walls. The X and Y components of the ball's motion must always be non-zero. The ball should bounce off a wall such that the angle of incidence equals the angle of reflection.

2. Display a small paddle near the right side of the VGA monitor and display the four walls at the edges of the monitor. Use two pushbutton inputs (BTN1 and BTN0) to move the paddle up or down. Pressing one of the pushbuttons will move the paddle up and pressing the other will move the paddle down. If neither pushbutton is pressed, the paddle will remain stationary.

3. Keep count of the number of successive hits without a miss and display the count on the

two right-most seven-segment displays. If you miss the ball and it hits the right wall, you must clear the count of successive hits and start over. Display the highest number of hits (i.e. the high score) since the last reset on the two left-most seven-segment displays. The number of hits and the high score can be displayed in hexadecimal.

4. The paddle only hits the ball when the ball is moving right, not left. When the ball bounces off the right wall (after a miss), the paddle must not change the ball's direction if the ball hits it as it is moving left. After a miss, you can either start the game over with some type of "serve" or you can just let the ball bounce off the right wall and continue playing with the hit counter reset to zero.

5. Use slide switches SW1 and SW0 on the Xilinx board to set the speed of the ball. There should be 4 different speed levels, from slow to fast. These switches should change the speed of the ball at any point during the game.

6. Display the game using the following colors:
   Background – Black
   Walls – Green or Red
   Paddle – Blue
   Ball – White or Yellow

Lab Requirements

1. Write a complete VHDL model for the Pong game.

2. Simulate the design using ModelSim. You should trace at least one or two lines of video output so that you can see part of a wall displayed. You will not be able to do a complete functional simulation since it would require far too much CPU time and memory. However, you should verify that at least the top one or two lines display correctly. Since you cannot simulate the design completely, you should be extra thorough in checking and carefully analyzing your code to catch any logical errors. **Demonstrate your simulation to your TA. (60 points)**

3. Synthesize your Pong design using Synplicity's Amplify software.

4. Run the Xilinx ISE software on the EDIF (.edf) output file. Make sure to check the report (.rpt) file to verify that the pin assignments were made according to the attributes.

5. Download and test your design on a Xilinx Spartan-3 board that is interfaced to a VGA monitor. You may need to do several iterations of synthesis, compile and download since you cannot simulate your complete design. **Have your TA assign a score and sign a verification sheet when your design works correctly in the Xilinx board. (60 points)**

Lab Report (30 points)

For your lab report, include the following:

- Lab Cover Sheet with signed TA verification for a working design.

- Complete VHDL source code for your design and your simulation test bench.
- Simulation waveforms showing part of one row of video display
- Schematic of your Pong design printed from Amplify.
- Answer the following question:
    1. Based on your VHDL code, how many cycles of the 25 MHz system clock are required for one complete video frame? Remember that a complete video frame includes some off-screen time. Show your calculations!