Lab 5
VGA Display. Brick-breaker game.

Design and implement a digital circuit capable of displaying predefined patterns on the screen of a VGA monitor, and provide the basic components for a brick-breaker game. Your circuit must generate all control and data signals driving the VGA output of the BASYS2 board. Please refer to the introductory lab slides for information about the meaning and timing of VGA control signals. Use the clock frequency of 25 MHz (default frequency for BSYS2 board is 50 MHz), and the resolution as close as possible to 640x480 pixels. This Lab can be done in teams or individually, but teams are required to finish more tasks.

Task 1: Display Ball (in Blue color) and Background (in Cyan color)
(for individuals: required, 1 points; for teams: required, 0.75 points)

Design a circuit to display a ball in blue ball in the center of the screen. The background of the screen is in cyan color. The ball is 20 pixels in diameter.

Task 2: Display Pedal bar (in Red color)
(for individuals: required, 1 points; for teams: required, 0.75 points)

Extend Task 1 by adding a pedal bar (in red color). Adjust the width of pedal to be equal to 5 pixels (in x-axis) and height to be 60 pixels (in y-axis).
Task 3: Display Boundary (in Green color)  
(for individuals: required, 1 points; for teams: required, 0.75 points)  
Extend Task 2 by adding a boundary (in green color). Adjust the boundary thickness to be equal to 3 pixels, length to be 634 pixels (in x-axis) and height to be 474 pixels (in y-axis).

Task 4: Ball and Pedal bar movements  
(for individuals: required, 1 points; for teams: required, 0.75 points)
Extend Task 3 by adding the capability to move the ball automatically to any point inside the boundary. The ball should bounce back when it hit the boundary or pedal bar and must not exit the boundary. The movement of pedal bar is controlled by using switch 0 (switching the toggle switch to high state represents upward movement and switching it to low state represent downward movement). Pedal bar can only move up and down and must not cross top and down ends of boundary. The white arrows depict movement of the ball; they do not need to be drawn on the screen.

---

**Task 5: Ball speed and Pedal bar sizes**  
*for individuals: required, 1 points; for teams: required, 1 points*

Extend Task 4 to allow the capability of adjusting the ball speed and pedal bar sizes. Switches 1-2 select the step size to change the balls speed (in both horizontal and vertical direction). One step is equal to 4 pixels, which allows the user to change the step size between 4 and 16 pixels in both x and y-axis (switch 1-2 position 00 means step size = 4, 01 means step size = 8 and 10 means step size = 12). Switch 3-4 select the pedal bar sizes (switch 3-4 position 00 means 100% of pedal bar size i.e. default option, 01 means 50% of pedal bar size and 10 means 25% of pedal bar size).
**Task 6: Displaying Bricks**
(for individuals: **required, 1 points**; for teams: **required, 1 points**)

Extend Task 5 by adding a 12 bricks, 4 bricks each in three vertical columns (in magenta color). Adjust the width of each brick to be equal to 5 pixels (in x-axis) and height as per your requirement (see figure below).

**Task 7: Bricks disappearing when hit by a ball**
(for individuals: **bonus, 2 points**; for teams: **required, 1 points**)

Extend Task 6 by adding the capability to break the bricks i.e. when ball hits any of the brick, it should disappear.
Task 8: Display score on Seven segment display
(for teams only: bonus, 2 points)

Extend Task 7 by adding a capability to display the score on seven segment display. Whenever the ball hits any of the bricks, count on seven segment display should increment and display the total score. When all of the bricks are broken, it should pause the game and DONE should be displayed on the display.

For the all tasks, determine the following properties:

1. Number of CLB slices
2. Minimum clock period after synthesis [ns]
3. Maximum clock frequency after synthesis [MHz]
4. Minimum clock period after implementation [ns]
5. Maximum clock frequency after implementation [MHz]

As a part of the design process:

1. Draw an optimized block diagram of your color, sync generator, ball and pedal bar fill and movement, bricks fill, boundary fill, counter circuits and top level unit.
2. Translate your block diagram (and the rest of the circuits) into RTL VHDL.
3. Write a testbench capable of verifying the functionality of the overall circuit.
4. Synthesize, implement, and load your design on the Basys board to verify functionality.
In the lab report include:

1. Optimized block diagrams (scanned version of the hand-drawn hardcopy submitted using Blackboard in the pdf or MS Word format (.).

2. VHDL source codes for:
   a. VGA Sync generator
   b. Ball fill and movement
   c. Pedal bar fill and movement
   d. Bricks fill
   e. Boundary fill
   f. Counter circuit
   g. Clock divider
   h. Color generator
   i. Top level unit

3. User Constraint File (UCF)

<table>
<thead>
<tr>
<th>Activity</th>
<th>Tuesday Section</th>
<th>Wednesday Section</th>
<th>Thursday Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hands-on Session and Introduction to</td>
<td>03/20/2012</td>
<td>03/21/2012</td>
<td>03/22/2012</td>
</tr>
<tr>
<td>the Experiment</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Demonstration and Deliverables Due</td>
<td>04/3/2012</td>
<td>04/4/2012</td>
<td>04/5/2012</td>
</tr>
<tr>
<td>(Schedule B)**</td>
<td>04/10/2012</td>
<td>04/11/2012</td>
<td>04/12/2012</td>
</tr>
</tbody>
</table>

**Students following Schedule B forfeit their right to all points for Lab 7: PicoBlaze & Serial Communication (a total of 6 points).**