EE20021 Digital Systems Design: Project

U Martinez-Hernandez

December 2019

Introduction

In the laboratory sessions you will use the FPGA board and your VgaController module to create a PongGame project (see Figure 1). Download the lab files from Moodle before you start.

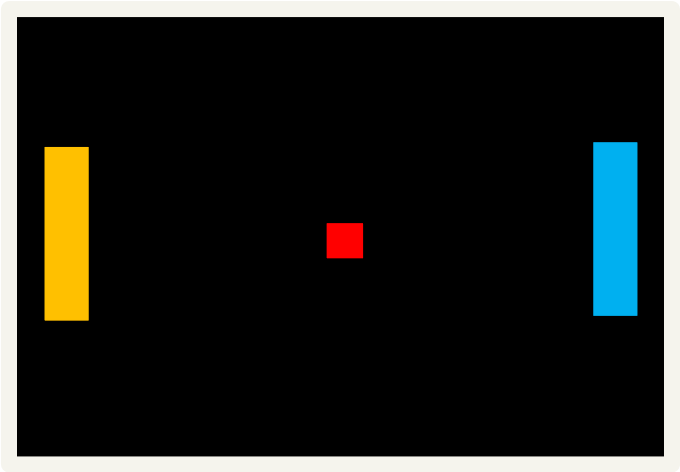

Figure 1 – Pong game

Specifications

The Pong game will be evaluated based on the following specifications:

Geometrical specifications

- Left side bar
  - Initial position [x, y]: [30, 225]
  - Size [height, width]: [150 pixels, 50 pixels]
  - Colour: Any colour different from the background, the right side bar and the ball
- Right side bar
  - Initial position [x, y]: [720, 225]
  - Size [height, width]: [150 pixels, 50 pixels]
  - Colour: Any colour different from the background, the left side bar and the ball

Figure 2 shows the geometric specifications for the Pong game.

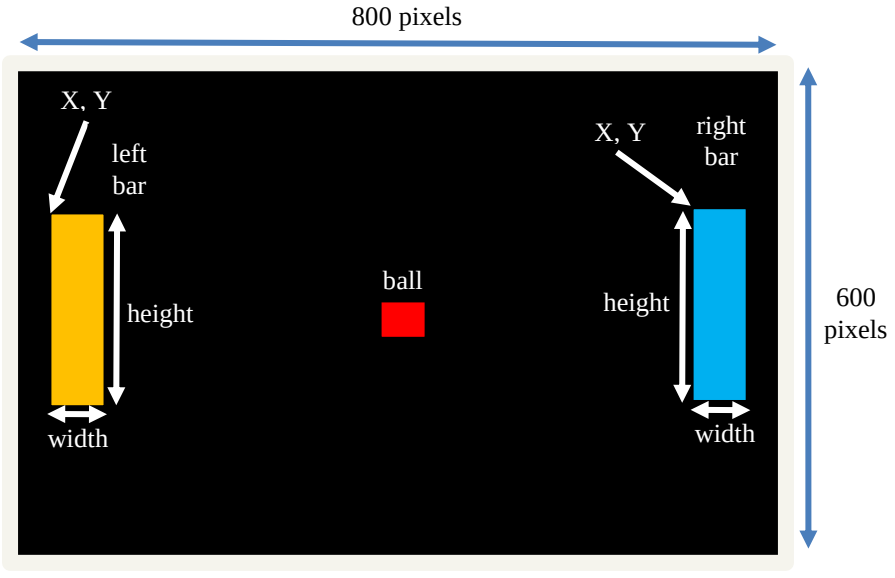

Figure 2 – Geometrical specifications

Movement of lateral bars controlled by push buttons

- The KEY[0] button moves the right bar down at the same speed as the ball travels, while the button is pressed
- The KEY[1] button moves the right bar up at the same speed as the ball travels, while the button is pressed
- The KEY[2] button moves the left bar down at the same speed as the ball travels, while the button is pressed
- The KEY[3] button moves the left bar up at the same speed as the ball travels, while the button is pressed
- Lateral bars should move only in the vertical limits of the visible area
  Figure 3 shows the specification for the movement of lateral bars

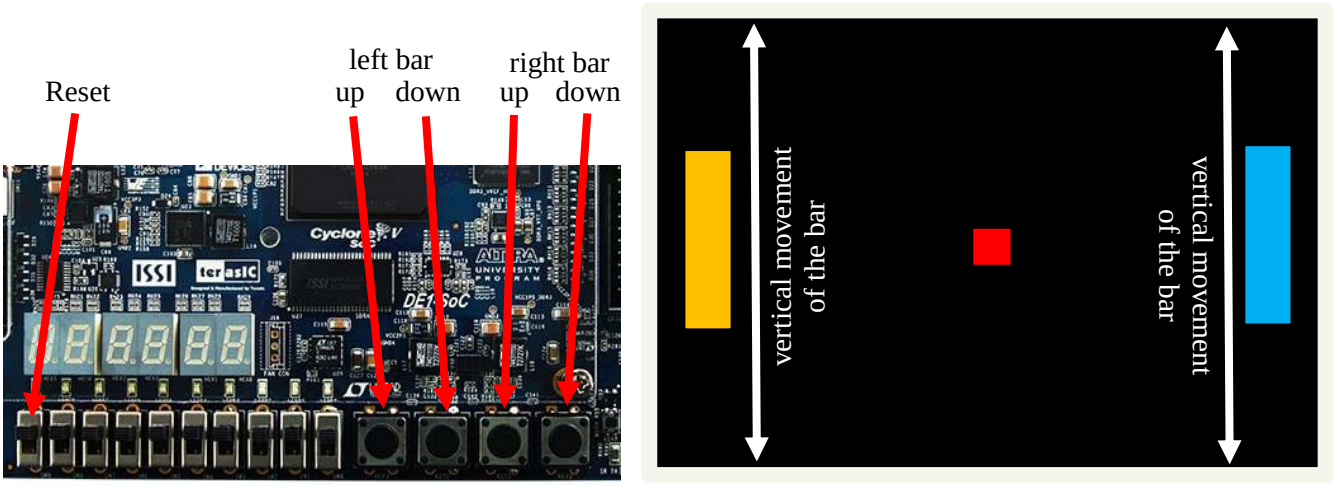

Figure 3 – Object movement using push buttons

Collision detection behaviour

- The ball bounces if a collision is detected with the top and bottom edges of the visible area
- The ball bounces if a collision is detected with the edges of the left, right, top and bottom sides of the bars
- The ball starts at the centre of the screen if a collision is detected with the left or right edges of the visible area
  Figure 4 shows examples of collisions with the lateral bars and edges of the screen.

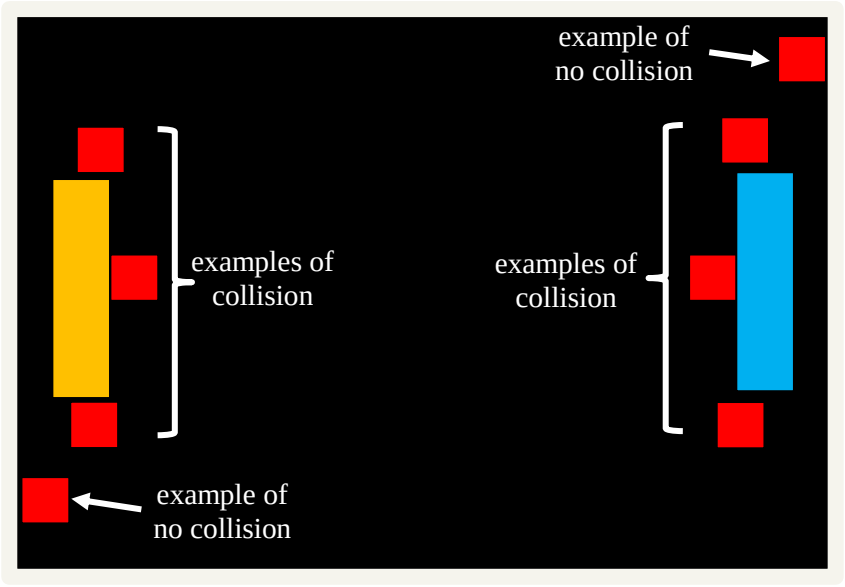

Figure 4 – Examples of collision events

Reset behaviour

- The Pong game restarts using the switch number 10 (SW[9]).

Comments and code

- Effective use of indentation, meaningful signal names and comments that aid understanding of the design.

Modules provided in Moodle for the development of the pong game.

- PongGame.sv
- ball.sv
- bar.sv
- slowClock.sv
- soc_system.sv
- **Your will also need to add your VgaController.sv module**

PongGame.sv is used to instantiate the ball and the bars at the left and right sides. In this module you also communicate all the modules, for example, you can send and receive signals between the ball.sv and square.sv modules to implement more functionality. Here you can also specify the colour for each of the objects and the background of the screen.

ball.sv is used to specify the dimensions and functionality of the ball moving in the screen. Here you can also specify the behaviour of the ball when a collision with the bars is detected or when the bar has crossed to the left or right sides of the visible area of the screen.

bar.sv is used to specify the dimensions and functionality of the bar at the left and right side of the screen. You can specify the behaviour of the bar according to the push buttons pressed by the user.

slowClock.sv is used to reduce the frequency (or speed) for printing the ball and bars in the screens and have the effect of movement. Otherwise, if we use the internal clock of 50MHz, you will not be able to see the movement effect of the objects in the screen.

NOTE: You will need to add your **VgaController.sv** module to the project.