HDL Project 3

Josh Belany
Julian Duarte
Nick Schaecher

Basic Description:

For our project we created a simplified version of the game Pong. If one is not familiar with Pong, it is one of the earliest arcade games ever created. It was published in 1972 by the company Atari. The game consists of two “paddles” on opposite sides of a black screen and a virtual “ball”. Originally pong was created as a two-player game with each player controlling a paddle which could be moved up or down the screen. If a given player was able to position their paddle in the path of the oncoming ball in time, the ball would bounce back across the screen at an appropriate angle for which it had “hit” the paddle. The ball was also capable of bouncing on the top and bottom of the screen. If either player could not position their paddle to hit the ball back, then the player who last hit the ball received a point. This continued until either player reached a score of 11 and had thus won.

The mechanics are the same for our version of Pong and the original except that our version is only single player, a computer is used to emulate a second player. The goal of our game is for the player to achieve a score of 11. The player will compete against the computer which is also attempting to achieve a score of 11. The player will notice during each contested point that the ball picks up speed after being hit across the screen. This feature adds difficulty for the player and the computer. In order to begin the game, the player must flip switch 15 on the basys 3 xilinx board to high. You can also pause the game by switching this switch back to low. The reset switch which is switch 0 will reset the score and the position of the ball if the player desires to perform this operation. In order to move the paddle figure for the player up and down the player must press the top and bottom buttons respectively. In the event either side achieves a score of 11 the game will end. In order to restart the game, the player needs to flip the reset switch high and then back to low. The player will notice that the background surrounding the game interface changes colors. This has no dependence or effect on the game play but rather is a feature to improve the aesthetic of the display.

HDL Project 3

Josh Belany
Julian Duarte
Nick Schaecher

Pong Technical Description:

Basys 3 User Interface Inventory

Basys pin (physical manifestation) : Pong utilization

W5 (clock)	:	CLK_100MHz, used to synchronize processing
V17 (switch)	:	Reset, used to set all non-clock states to their default
R2 (switch)	:	start, used to control the game progression
W7 (LED)	:	LED driver for segment 0
W6 (LED)	:	LED driver for segment 1
U8 (LED)	:	LED driver for segment 2
V8 (LED)	:	LED driver for segment 3
U5 (LED)	:	LED driver for segment 4
V5 (LED)	:	LED driver for segment 5
U7 (LED)	:	LED driver for segment 6
U2 (anode)	:	7-segment LED selection line, anode 0
U4 (anode)	:	7-segment LED selection line, anode 1
V4 (anode)	:	7-segment LED selection line, anode 2
W4 (anode)	:	7-segment LED selection line, anode 3
T18 (button)	:	Control signal for input up
U17 (button)	:	Control signal for input down

VGAStart

VGAStart is the top-level module of our VGA control structure. This module links the VGA control modules to the Pong game modules as well as the Pong game modules to the counter and 7 segment display modules.

VGALLDriver

This module is meant to drive a 800x600@72Hz display. In order to do this the module keeps track of the x and y position of the pixel being driven and outputs the given color for the pixel being driven.

VGAClient

This module computes the desired color at a given pixel. In this module the player and computer paddle colors and size as well as the ball color and size are defined. The paddles are defined to be 2 by 28 pixels in size and white in color. While the ball is represented with a diamond which is 5 pixels at its widest. The size of the play area is also defined in this module which we chose to be 512 by 256 pixels. The surrounding area is defined to be the input from RNDColor.

RNDCOLOR

This module sets a 12bit register based on a base value and a shift value. This module is responsible for the different colors that flash on the border of our play area.

Pong

This is the top-level module for our implementation of Pong. This module connects the Player, Computer, CLK_Divider, and Ball modules together. It also passes the status flags and ball clock into the counter module.

Player

This module controls the player paddle position and determines the next position of the player paddle depending on the state of the up or down buttons. This module also ensured the paddle does not leave the given bounds of the play area. In the case that reset is enabled this module resets the position of the player paddle to the default position.

Counter

This module is used to keep track of the game score and the win conditions. This module ensures that the score does not exceed 11 and once a score of 11 is reached the win flag is set. It also allows us to display the game score in decimal format.

Computer

This module controls the computer paddle position and determines the next position of the computer paddle depending on the position of the ball. This module also ensures the paddle does not leave the play area. In the case that reset is enabled this module resets the position of the computer paddle to the default position.

CLK_Divider

This module is used to set the movement speed of the ball and paddles. Initially both paddles and the ball refresh their positions at a rate of 30Hz. The ball and player paddles utilize different counters to define their clocks. This was done to allow us to increase the travel speed of the ball without increasing the refresh rate of the paddles. The paddle speeds never change.

Ball

This module tracks the position of the ball, the definition of what happens when a player scores, how the ball should bounce based on its position and direction of movement. This module takes the position of the computer and player paddles as inputs and compares their positions with the ball position. If the position of a given paddle is not in the path of the ball and the ball reaches the edge of the play area, the proper score flags are set. In the case that a paddle is in the appropriate position the ball will bounce.

HDL Project 3 Josh Belany
Julian Duarte
Nick Schaecher

Hex27Seg

The module Hex27Seg is used as a lookup table that converts a 4bit hexadecimal number into a 7bit binary number which will drive the 7-segment LED displays. The 7-segment LED displays are active low and take a serial input.

Mux4Machine

The module Mux4Machine was provided by Dr. Wright. We added a selection line to turn on the decimal point on the 7 Segment display so that we could differentiate the computer's score from the player’s score.