

Name: \_\_\_\_\_ ID: \_\_\_\_\_

Start Date: Monday, June 10, 2019

Due Date: Tuesday, June 11, 2019

## Software and Hardware Co-Design with Zybo, Spring 2019 HUST

### Lab #8 Interrupt-Driven Ping-Pong Game on Zybo

This is an individual lab. Each student must perform it on his or her own and demonstrate this lab to obtain credit for it. Late lab submission will be accepted with a grade reduction of 20% for each day that it is late.

#### 1 Zynq Book Sources

You will need programs `interrupt_counter_2B.c` and `interrupt_counter_2D.c` from the zynq book sources. The files are available from Zynq Book Sources link at <http://www.zynqbook.com/download-tuts.html>. They are also available from the instructor.

#### 2 Objectives

- 1) *Perform The Zynq Book Tutorial 2, Next Steps in Zynq SoC Design.*
- 2) *Modify Lab #7 LED Ping-Pong Game to use Input change interrupt and Timer interrupt. Input change interrupt service will handle four push buttons. Timer interrupt service will handle LED display. The main program will handle terminal display only after initialization.*

#### 3 Demonstration and Lab Memorandum by the Deadline

1. Demonstrate your interrupt-driven ping-pong game on your Zybo board.
2. Submit a pdf copy of a lab memorandum that includes the following.
  - *A memo header to include: To:, From:, Date:, Re, etc. Follow some standard memo style to write your memo.*
  - *A short paragraph to describe this lab and what you have done and included in this memo.*
  - *A number of typical screen captures of the Zynq Book Tutorial 2 to show you have completed it.*
  - *A copy of your final source code. Your source code should have a header to include your name, date and brief description of your code. It should be well commented and have no magic number nor magic port.*

#### 4 The Zynq Book Tutorial 2, Next Steps in Zynq SoC Design

Go through this tutorial to create two Vivado projects: `zynq_interrupts_exercise2B` for `interrupt_counter_2B.c` and `zynq_interrupts_exercise2D` for `interrupt_counter_2D.c`. Fix any errors or abnormal behaviors if you encounter them.

Get some screen shots of both parts as evidence of your completion of this tutorial.

#### 5 Requirements for the Interrupt-Driven Ping-Pong Game

The external behavior of the game is mainly the same as in Lab #6 except for reset and restart buttons. The game will reset as soon as the reset button is pressed. The game will restart as soon as the restart button is pressed no matter if the game is over.

- 1) *Momentarily depressing BTN2 should reset the score for the left player to 00. It should also reset the score for the right player to 00. The score should be displayed in the SDK serial port window on your laptop PC.*

- 2) *Momentarily depressing BTN1 should serve a “ball”; that is, a single lit LED should appear at the left end of the line of four LEDs (LD3) and the ball (the lit LED) should begin moving from the left end (LD3) of the line of LEDs toward the right end (LD0) at a variable speed (that is selected by the binary number entered on the slide switches).*
- 3) *When the ball (the lit LED) reaches the rightmost position (LD0), the right pushbutton (BTN0), which may be regarded as the paddle of the right player, should be depressed (it may NOT be pushed ahead of this time!) in order to hit the ball back in the other direction. If the right pushbutton is not pressed during this time, or if it happens to have been pressed too early before the ball arrives at LD0, or too late after the ball has traveled past LD0, the ball should “fall off” of the end of the line of the 4 LEDs (the ball vanishes... no LEDs remain lit), and the serve terminates. If this happens, play terminates, and one point should be added to the left player’s score.*
- 4) *Assuming the ball was correctly hit back toward the left, then the left push button (BTN3) must be pressed when the ball eventually arrives back at the leftmost LED position (LD3), in order to send it back the other way. If the left push button is not pressed at this crucial time while the ball is at LD3, then the right player’s score is incremented.*
- 5) *Once the serve has terminated, and a point has been awarded to either the right or left player, the players must wait for the serve button (BTN1) to be momentarily depressed once again. Then the ball must be served in the other direction (the direction of the serve should alternate with each press of the center button).*
- 6) *The speed of play (difficulty of the game) should be adjustable in 16 steps using the 4 on-board DIP switches. A setting of 15 should be very slow, and a setting of 0 should be quite fast, though still be (barely) playable by an expert!*

## 6 The Architecture of the Interrupt Driven LED Ping-Pong Game

This game is composed of three major subroutines: main(), push button interrupt handler and timer interrupt handler. The main program will set up interrupts and initialize all ports and timer. The main program will then print on the terminal of the game status and results. The two interrupt handlers will deal with push buttons and timer interrupts.

Push button interrupt handler will decide which button is pressed and set game status and result accordingly.

Timer interrupt handler will update LED display.

Global status and result variables are used to communicate among the three major subroutines.

## 7 References

1. Private Timer device drivers for Cortex A9, scutimer v2\_0, Xilinx.
2. General purpose I/O (XGpio) device drivers, gpio v4\_0, Xilinx.
3. How to Use Interrupts on the Zynq SoC, Xcell Journal 87
4. xil\_exception.c, standalone\_v4\_2, from  
C:\Xilinx\SDK\2014.4\data\embeddedsd\lib\bsp\standalone\_v4\_2\src\cortexa9
5. Chapter 3: Interrupt Controller, Cortex™-A9 MPCore, Revision: r2p2, Technical Reference Manual
6. The\_Zynq\_Book\_Tutorials, v1.2 - September 2014, University of Strathclyde, Glasgow, Scotland, UK