HO CHI MINH UNIVERSITY OF TECHNOLOGY

FACULITY OF COMPUTER SCIENCE AND ENGINEERING

DEPARTMENT OF MECHATRONICS

LOGIC DESIGN WITH HDL LAB

STUDENT REPORT

Class: CC01		Group: 04
Student's name:	Trịnh Sơn Lâm	1852502
	Nguyễn Trọng Tính	1752545
	Lê Bá Thành	1852739
	Hoàng Nhật Quang	1852691

Lecturer Ph.D Phạm Quốc Cường

APRIL 1, 2022

WEEK 3 Sequential Logic Circuit

Exercise:

Exercise 1

Clock Frequency Divider

a. Police Siren: Design a circuit that generate a 1 Hz output signal using Verilog HDL. This signal is connected to 2 RGB LEDs (1 displays the blue color, 1 display the red color) on Arty-Z7 FPGA Board to make it blink interleave with each other (turn on for 0.5s - turn off for 0.5s). Know that the input clock frequency is 125 MHz.

Write test benches to simulate the circuits in a

We have file code is **1hz_rgbLed.v**

And the we have test bench for 1hz_rgbLed is **1hz_rgbLed_tb.v**

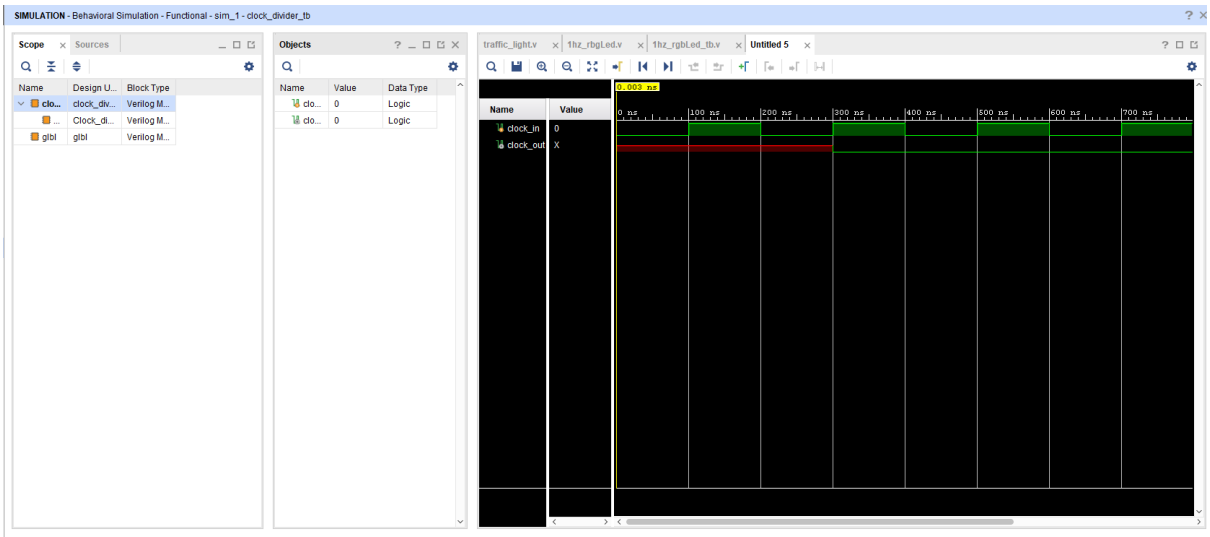

Figure 1: The waveform of test bench for RGB LED from 1Hz

Test the circuits on FPGA board using LEDs and RGB LED.

We have some changes in constraint as follows:

```
## Clock Signal
set_property -dict { PACKAGE_PIN H16    IOSTANDARD LVCMOS33 } [get_ports { clk }]; #IO_L13P_T2_MRCC_35 Sch=SYSCLK
#create_clock -add -name sys_clk_pin -period 8.00 -waveform {0 4} [get_ports { clk }];#set

## Switches
#set_property -dict { PACKAGE_PIN M20    IOSTANDARD LVCMOS33 } [get_ports { in[1] }]; #IO_L7N_T1_AD2N_35 Sch=SW0
#set_property -dict { PACKAGE_PIN M19    IOSTANDARD LVCMOS33 } [get_ports { in[2] }]; #IO_L7P_T1_AD2P_35 Sch=SW1

## RGB LEDs
set_property -dict { PACKAGE_PIN L15    IOSTANDARD LVCMOS33 } [get_ports { c_out[0] }]; #IO_L22N_T3_AD7P_35 Sch=LED4_B
#set_property -dict { PACKAGE_PIN G17    IOSTANDARD LVCMOS33 } [get_ports { empty }]; #IO_L16P_T2_35 Sch=LED4_G
##set_property -dict { PACKAGE_PIN N15    IOSTANDARD LVCMOS33 } [get_ports { full }]; #IO_L21P_T3_DQS_AD14P_35 Sch=LED4_R
#set_property -dict { PACKAGE_PIN G14    IOSTANDARD LVCMOS33 } [get_ports { led5_b }]; #IO_0_35 Sch=LED5_B
#set_property -dict { PACKAGE_PIN L14    IOSTANDARD LVCMOS33 } [get_ports { led5_g }]; #IO_L22P_T3_AD7P_35 Sch=LED5_G
set_property -dict { PACKAGE_PIN M15    IOSTANDARD LVCMOS33 } [get_ports { c_out[1] }]; #IO_L23N_T3_35 Sch=LED5_R

## LEDs
```

Figure 2: Assign pin for Arty-Z27 before running for FPGA

Then we have the generated bitstream file is **Clock_divider.bit**

b. Addition: Crossroad Traffic Light: Design a pair of traffic lights for a crossroad intersection

Write test benches to simulate the circuits in a

We have file code is **traffic_light.v**

And the we have test bench for traffic light is **traffic_light_tb.v**

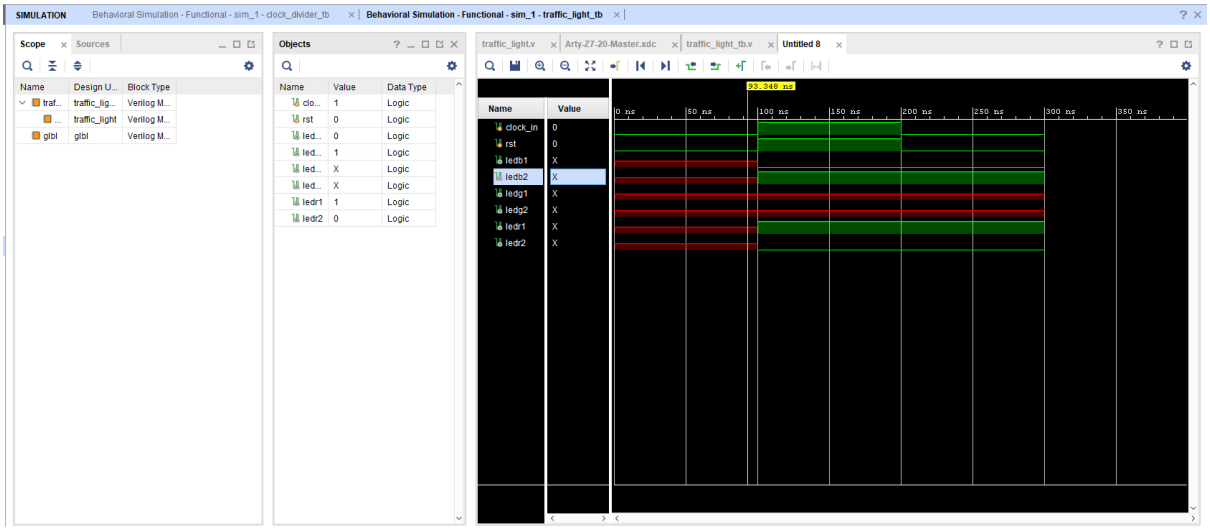

Figure 3: The waveform of test bench for traffic light

We have the generated bitstream file is **traffic_light.bit**

Exercise 2

Edge Detection circuit.

a. Design a Rising Edge Detection circuit. This circuit will use at least 2 flip-flops. The behavior of the circuit is similar to the waveform in Figure 1. Assume that the in signal's HIGH levels last equal to or longer than a clock cycle.

The output is active HIGH in 1 cycle of clock when a rising edge occurs in input signal. Delay is within 0-2 clock cycles.

Write RTL code and test benches to simulate the circuit.

We have file code is **edge_detector.v**

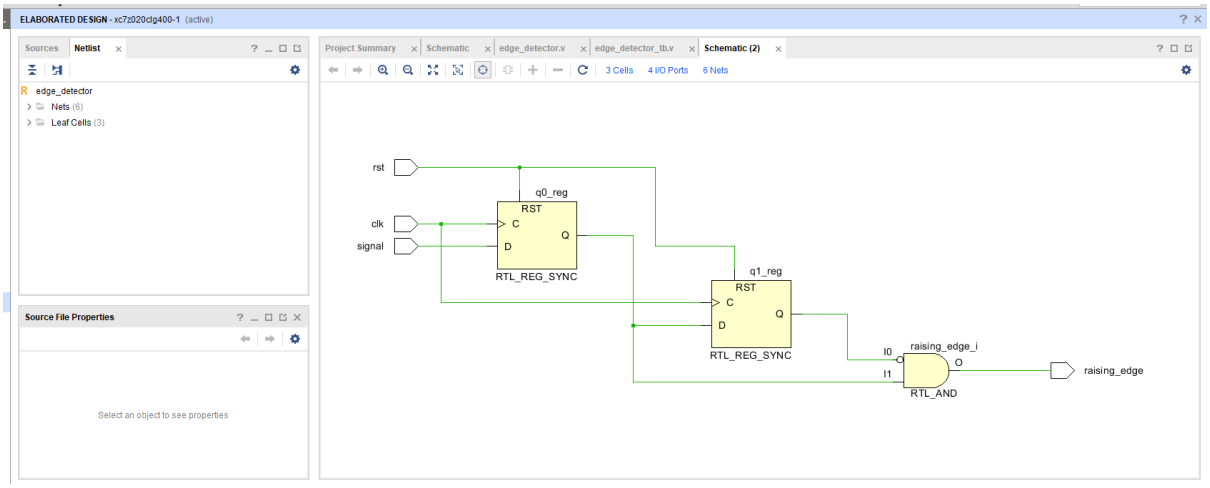

Figure 4: Edge dectector in RTL schemetic

And the we have test bench for edge dectector is **edge_detector_tb.v**

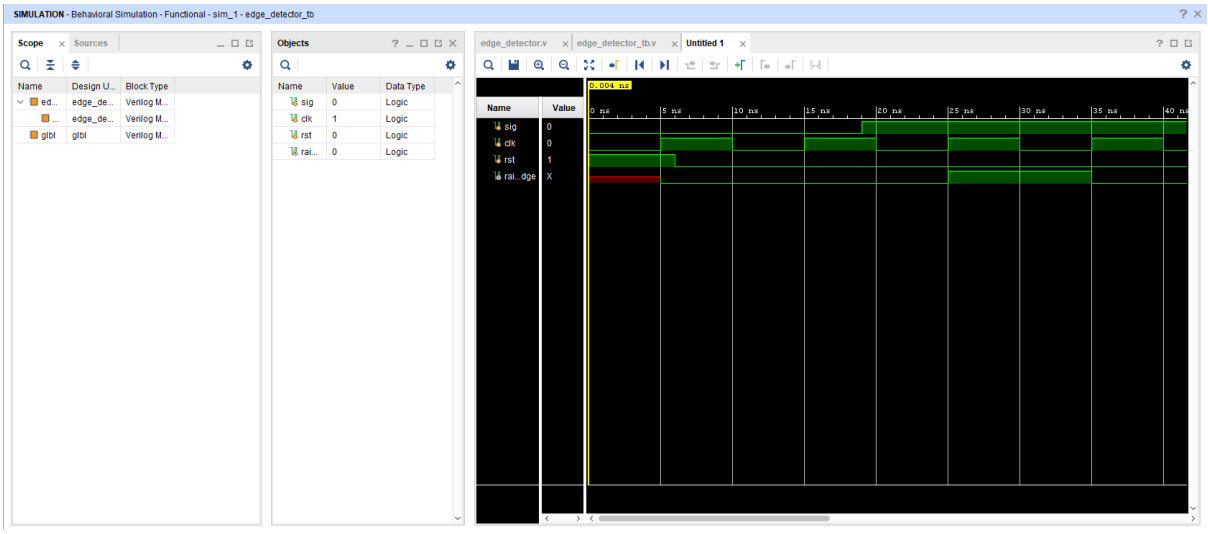

Figure 5: The waveform of test bench for edge dectector

b. Write a 4-bit binary counter that counts up 1 unit when a button is pushed. Use the edge detection circuit to generate an enable signal for the counter when pushing the button. Test the design on FPGA board

We have file code is **four_bit_binary_counter.v**

And then we have the generated bitstream file is **four_bit_binary_counter.bit**

Exercise 3

Change mode String bit LED circuit. Use Verilog HDL to model a state machine for a circuit that changes display mode of a bit string.

In initial, LEDs show the default bit string 0011 which is performed by a reset signal. And buttons in board will set the display mode as follow:

• Button 0: Mode Reset: Show the default 4-bit string on LEDs.

• Button 1: Mode Shift Left Ring : Shift 4-bit string to left in a ring every 1s.

• Button 2: Mode Shift Right Ring: Shift 4-bit string to right in a ring every 1s.

• Button 3: Pause: Pause the current shifting string.

. Write a test bench to simulate the circuit and test the circuit on the Arty-Z7 board.

We have file code is **led_shift_test.v**

And the we have test bench for traffic light is **led_shift_tb.v**

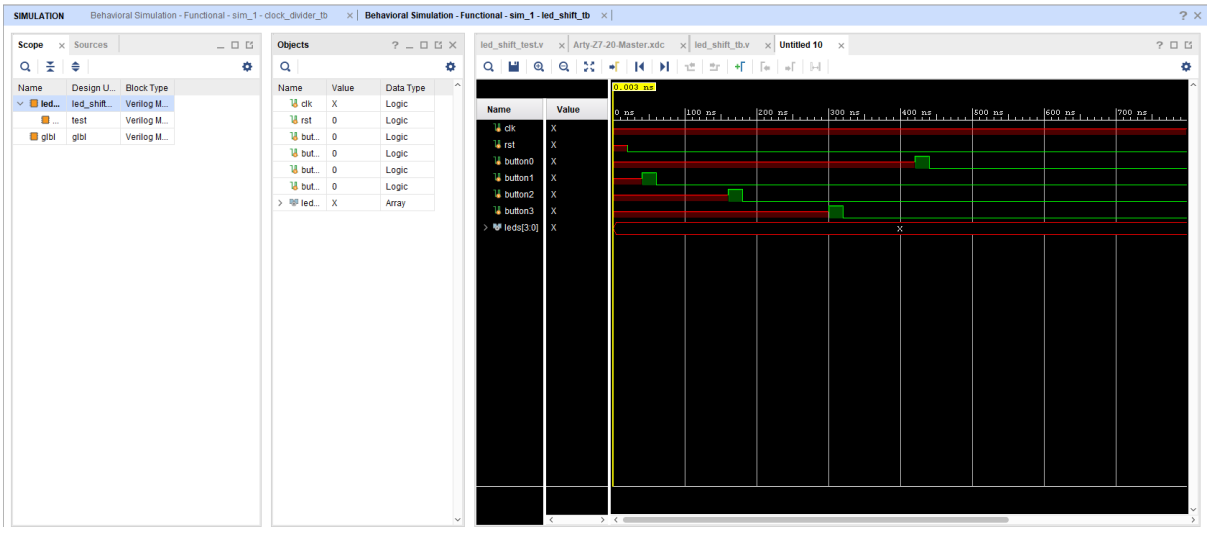

Figure 6: The waveform of test bench for led shift

We have some changes in constraits as follows:

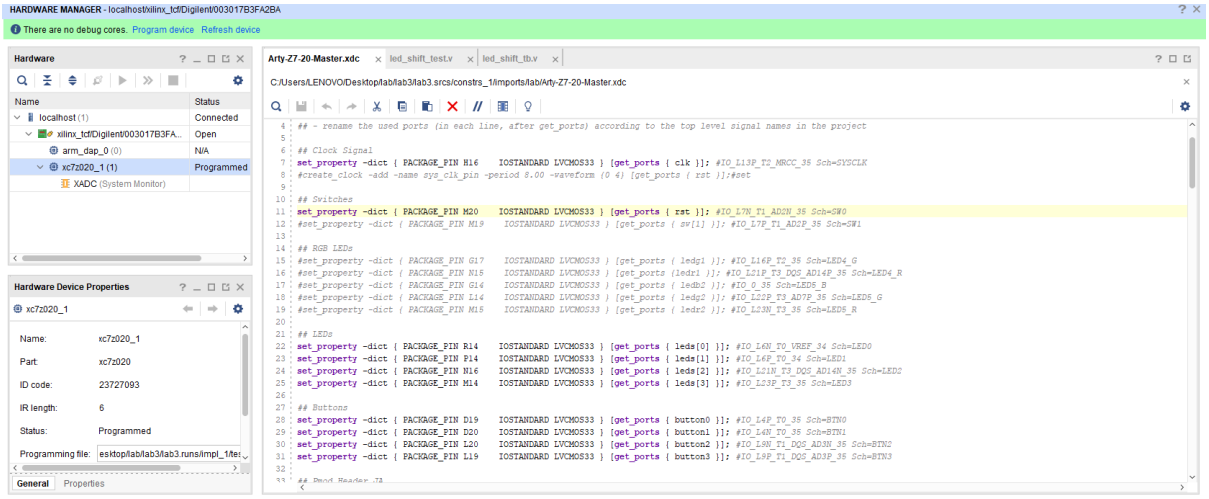

Figure 7: Assign pin for Arty-Z27 before running for FPGA

We have the generated bitstream file is **test.bit**