EELE 367 - Introduction to Logic Circuits

Lab 9(a) – Ripple Counter and the Character Displays

Objective

The objective of this lab is to gain experience designing and using sequential storage devices in VHDL. You will create a D-flip-flop model and then use it to build a 38-bit ripple counter. The slower bits of the counter will be used to drive the LEDs and HEX character displays on the DE0-CV FPGA board.

Outcomes

After completing this lab you should be able to:

- Create a model of a D-flip-flop storage device in VHDL.
- Create a 38-bit ripple counter out of the D-flip-flop subsystem.
- Use the slower bits of the counter to drive the various LED and HEX I/O on the DE0-CV board.

Deliverables

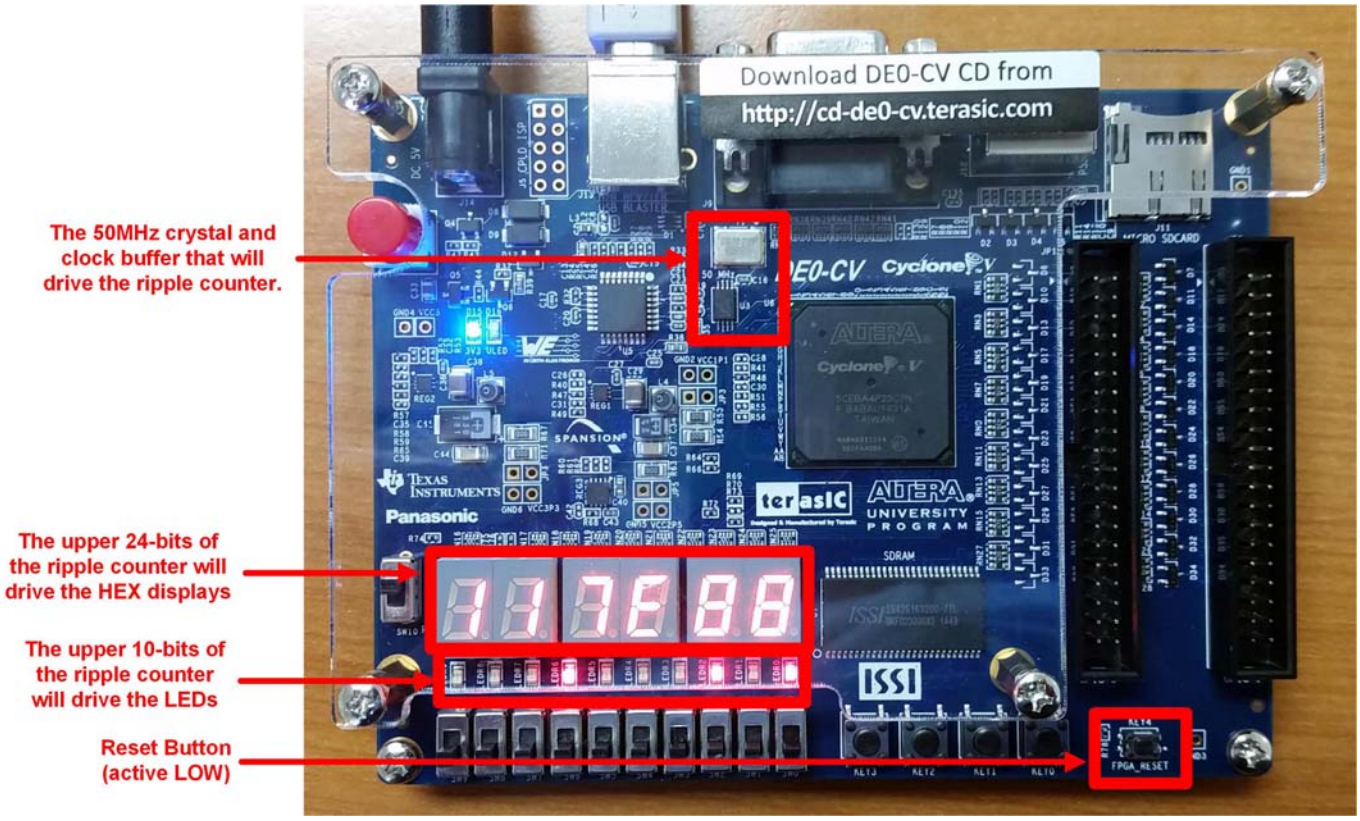
The deliverable(s) for this lab are as follows:

- Demonstrate a 38-bit ripple counter made of D-flip-flops where the slower bits drive the HEX displays and LEDs on the DE0-CV FPGA board (90%).
- Upload your top.vhd file to the lab DropBox (10%).

Lab Work & Demonstration

Part 1 – System Design

You are going to build a 38-bit ripple counter out of D-flip-flops. You will first create a model of a rising edge triggered D-flip-flop in VHDL (dflipflop.vhd) and then use it in your top.vhd to create the counter. The counter will be clocked from the 50MHz clock on the DE0-CV board. The reset for the counter will come from the active LOW Reset push button. The counter needs to be 38-bits wide in order to provide some bits that toggle slow enough to be seen with the human eye when observed on the DE0-CV LEDs. The upper 24-bits of the counter will be used to drive the 6x HEX character displays on the DE0-CV board. The upper 10-bits of the counter will be used to drive the 10x LEDs on the DE0-CV board. Once active, you will see the character displays and the LEDs free running. The following figure shows the I/O that will be used in this lab.



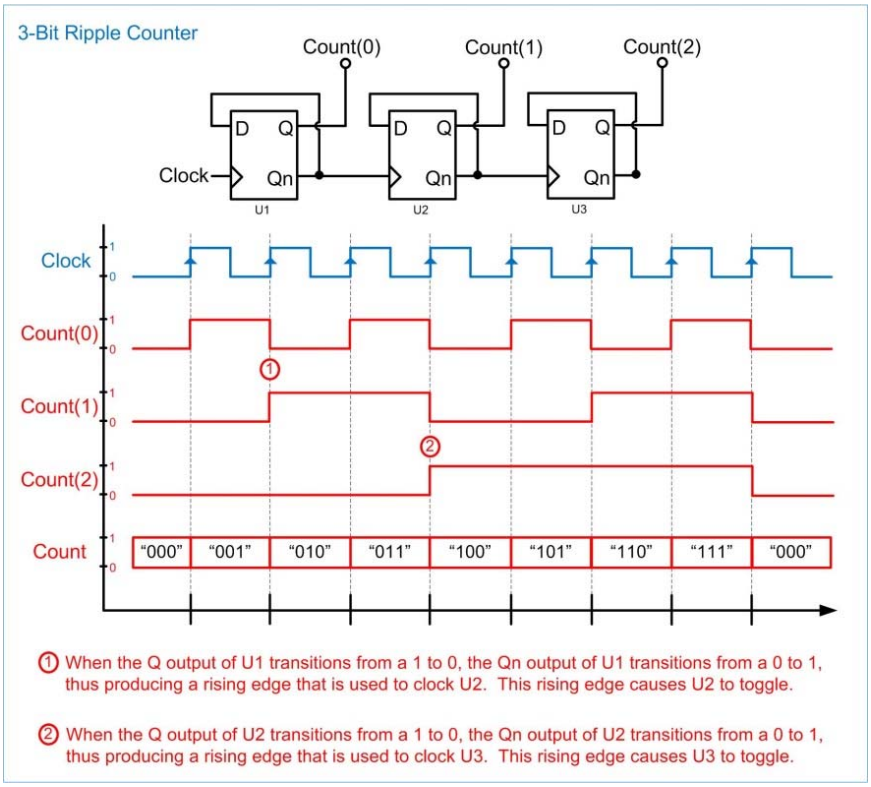
A) Create new Quartus project by copying a prior lab. Name it "Lab9a_Ripple_Counter_and_Character_Displays".

B) Create a model for a rising edge D-flip-flop. Call it "dflipflop.vhd". The model should have an active low reset.

C) Modify your top.vhd file entity to support the additional I/O that will be used in this lab. Use the entity in the following figure. Notice that you will need to add the *Clock* and *Reset* signals in addition to widening the LEDR vector to 10-bits.

```
1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3
4  entity top is
5    port (Clock : in std_logic;
6          Reset : in std_logic;
7          SW    : in  std_logic_vector(3 downto 0);
8          LEDR  : out std_logic_vector (9 downto 0);
9          HEX0  : out std_logic_vector (6 downto 0);
10         HEX1  : out std_logic_vector (6 downto 0);
11         HEX2  : out std_logic_vector (6 downto 0);
12         HEX3  : out std_logic_vector (6 downto 0);
13         HEX4  : out std_logic_vector (6 downto 0);
14         HEX5  : out std_logic_vector (6 downto 0));
15 end entity;
16
```

D) Create the 38-bit ripple counter by declaring and then instantiating your dflipflop.vhd model. Recall that a ripple counter is made using the following topology. You will need to instantiate your D-flip-flop 38 times.



You will need to create internal signals to connect to the D, Q, and Qn ports of the D-flip-flops. Consider creating 38-bit signal vectors called **CNT** and **CNTn**.

E) Connect the most significant 24-bits of the counter to your char_decoder.vhd components to drive the 6x HEX displays. You'll connect the bits in groups of four. For example, CNT(37 downto 34) will drive HEX5, CNT(33 downto 30) will drive HEX4, etc.

F) Connect the most significant 10-bits of your counter directly to the 10x LEDs on the DE0-CV board.

G) Find the pin assignments for the Clock, Reset, and additional LEDR I/O used in this lab using the DE0-CV User's Guide. Add these to your pin_assignments.csv document and then import into Quartus.

H) Compile your design. Fix any Errors.

I) Download and test your design. Fix any Errors.

G) Demonstrate your design.

DEMO – Once checked off, upload your top.vhd file only to the lab DropBox.

Lab Grading

Demo – Proper Operation of your Counter Circuit	_____ **/ 90**
Review of your top.vhd file	_____ **/ 10** (uploaded to DropBox)
Total	_____ **/ 100**

Post Lab Survey (Just for your own knowledge. This is not graded)

1) Can you create a D-flip-flop model in VHDL?
2) Can you use your D-flip-flop model to build a ripple counter that is clocked off of the 50MHz clock on the DE0-CV board and reset using the “FPGA_Reset” button?
3) Can you observe the slower bits of the ripple counter using the HEX displays and LEDs on the DE0-CV board?