Laurice Sattouf

Choi Tim Anthony Young

Dimitri Garcia

ECE4304- Lab 4

3/03/2021

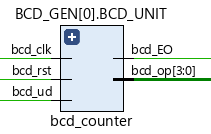
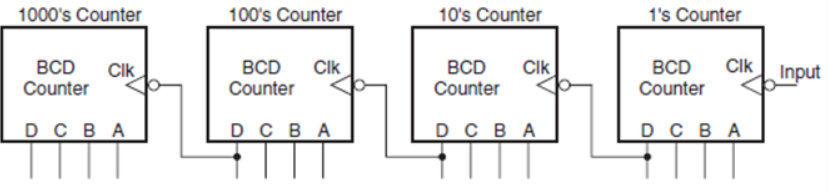
Professor M. Aly

**Cascaded BCD Counter**

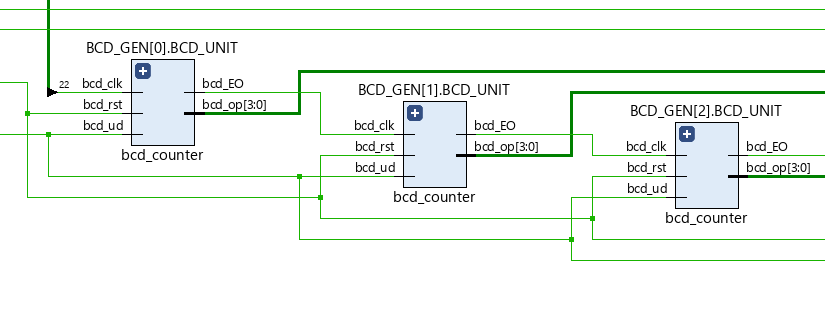
**Purpose:**

1. Design a BCD counter counting up and down based on the value specified on an input switch.
2. Use all the displays available (8 units) of the BCD to display the results.
3. Design the implementation with two different reset buttons so that we could reset the counter or the displays separately.
4. Define all possible corner cases using a testbench textio.

**Procedure:**

1. Design the component of BCD counter and make it generic so that we could create all 8 unites of the counter (16 digit total) using the original component.
2. Apply the coding techniques to develop the cascaded counter as follow:

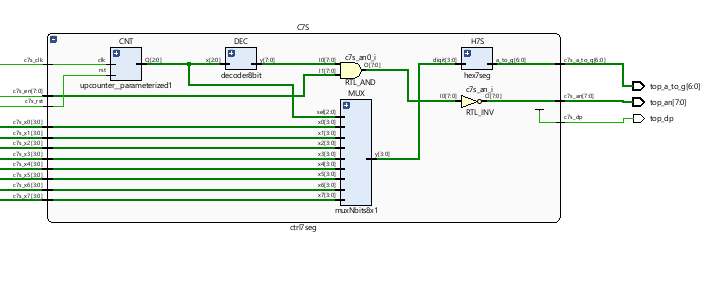
The arrangement works as follows. Initially, all four counters are in the all 0s state. The counter representing the decimal digit of 1’s place is clocked by the pulsed signal that needs to be counted.

The successive units are clocked by the MSB of the immediately previous counter stage. The first nine pulses take 1’s place counter to 1001. The tenth pulse resets it to 0000, and ‘1’ to ‘0’ transition at the MSB of 1’s place counter clocks 10’s place counter. 10’s place counter gets clocked on every tenth input clock pulse.

The bcd\_EO was used as an enable to the BCD sequence we have in our implementation.

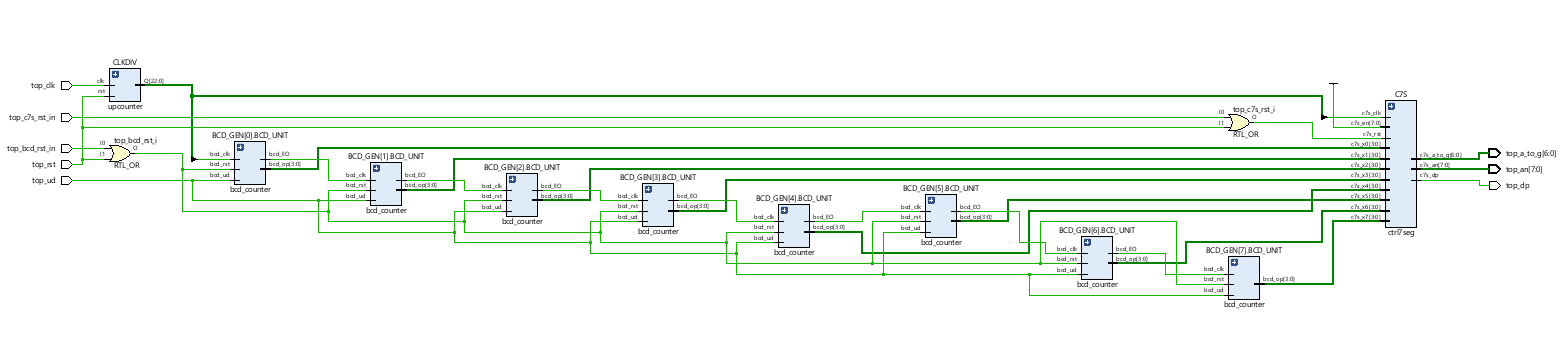
1. Use the 7-segment display which contains the following:
   1. Clock divider.
   2. The decoder.
   3. The Multiplexer.
   4. BCD to segments units.

The following schematic demonstrate the above:



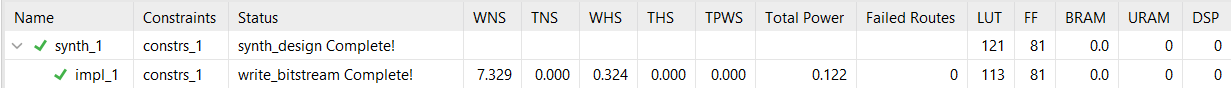
**Schematic:**

Wrapping all the component created, we end up having the following cascaded BCD counter:



**Figure 3: RTL Schematic using Vivado**

**Power and Resources Used:**



113 lookup tables and 0.122 total power used to implement this design using Vivado. We had two different codes, and this was the most efficient with the least amount of power and lookup tables.

**Difficulties:**

This lab was easy to implement since all the syntax that needed are clear and basics.

**Work Contribution:**

* We had a meeting to brainstorm and explain the main idea of the lab, and we created the schematic so that everyone would work individually to achieve the most optimized design.
* We had a zoom meeting to choose the design with less power consuming, then we were able to demo our implemented design and cover all the corner cases.
* Documentation and reports were evenly distributed, and it covered all the steps of our successfully implemented design.