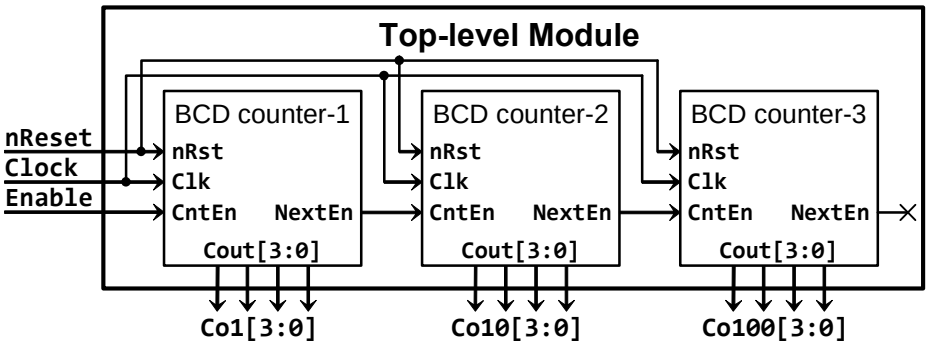
EE342 - Digital System Design

Laboratory Experiment - 2
BCD Counter

Preliminary Work

A Binary Coded Decimal (BCD) counter increments up to **9** and it goes back to **0** after **9**. Synchronous BCD counters can be connected together to form a three digit counter as shown below. All counter modules are synchronously triggered by the same clock. The **`NextEn`** output of a BCD counter is connected to the **`CntEn`** input of the next counter. Counter-2 is enabled when Counter-1 output is **9**. Similarly, Counter-3 is enabled when both Counter-1 and Counter-2 outputs are **9**.



1. Write a synchronous BCD counter module that has the following inputs and outputs.

`nRst:`	active-low asynchronous reset input that clears counter output
`Clk:`	clock input triggering at rising edge
`CntEn:`	count enable input; **0**=> stop, **1**=> count
`Cout[3:0]:`	4-bit counter output
`NextEn:`	output to enable the next digit counter. set to **1** when **`Cout[3:0]`** are **`4'd9`**, and set to **0** otherwise.

2. Create a project directory for Experiment 2 and set up a new project in Quartus II using the **File->New Project Wizard...** menu item following the instructions given in EE342_Lab_QuartusIntro.pdf. Compile and debug the BCD counter module you wrote.

Procedure

1. Write a top-level module that instantiates three BCD counter modules and counts up to **999**. In the top-level module connect the **`NextEn`** outputs of the counter modules to the **`CntEn`** inputs of the higher order counter modules. The top-level module inputs and outputs are:

`nReset:`	active-low asynchronous reset input that clears all counter outputs
`Clock:`	clock input triggering at rising edge
`Enable:`	counter enable input; **0**=> stop, **1**=> count
`Co1[3:0]:`	4-bit counter output corresponding to 1's decimal digit.
`Co10[3:0]:`	4-bit counter output corresponding to 10's decimal digit.
`Co100[3:0]:`	4-bit counter output corresponding to 100's decimal digit.

2. Add the top-level module file to the project (**Project->Add/Remove Files in Project...**). Assign the new module as the top-level entity of the project. Select the **Assignments->Settings...** menu item, and enter the name of your top-level module in the "**General**" category. Compile and debug the project.

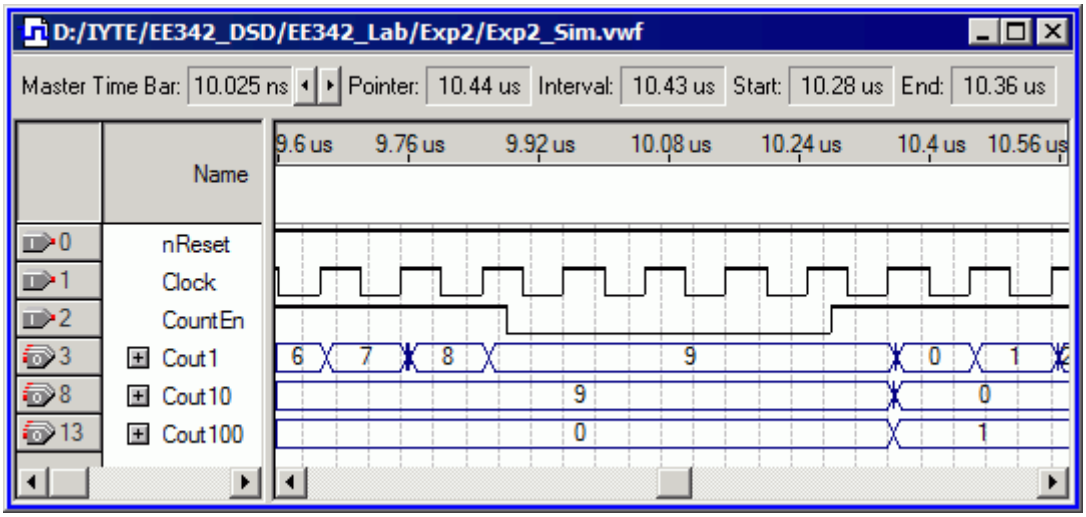
3. Prepare a simulation waveform file as follows.

- Create a new waveform file.
- Set "**End Time**" of the waveform file to **20 µs**.
- Include all input/output pins in the waveform display.
- Generate a **10 MHz** clock as the **`Clock`** input and set other inputs to enable the counter.
- Select decimal display for the counter outputs. Right-click on the waveform label, select **Properties**, and choose "**Unsigned Decimal**" as the **Radix**.
- Save the waveform file.
- Open Simulation tool, and select the saved waveform file as the simulation input.

4. Start simulation and check if your code produces the correct outputs.
Check the count sequence after "**009**".
Check the count sequence after "**099**".

Modify either BCD counter module or the top-level module to obtain the correct count sequence, if necessary.

5. Set the **count enable** input to "**0**" for a few clock cycles right after the counter outputs become "**099**" as shown below. Check if all counter modules stop as they should while the **count enable** input is "**0**".



Modify your code to obtain the correct count sequence, if necessary.

6. Check the circuit output at a higher frequency as follows.

- Restore the **count enable** input to "**1**".
- Set "**End Time**" of the waveform file to **1 µs**.
- Set frequency of the **`Clock`** input to **250 MHz** (4 ns clock period).
- Save the waveform file and run simulation again.

Check the counter outputs and provide an explanation for any deviation from expected counter results.

7. As a home exercise (not part of the experiment) upgrade the BCD counter module to an up/down counter with an additional up/down control input. Note that the **`NextEn`** output should be set to **1** when **`Cout[3:0]`** are **`4'd0`** when the counter is decrementing.