

- 1) Designing and analyzing synchronous counters.
- 2) Understanding the concepts of state machine design.

Parts 1A, 1B and 2A

- 

B) Create a schematic of Fig 9-1.

- C) Compile and simulate. Check your simulation results make sure your circuit is working according to your state table of part A.

(For simulation: assign 2ns period to the input clk and run for 20ns.)

- D) Copy the seven segment decoder design from lab5 or lab6 into your working directory. Then use **Add Source from project menu** to add the seven segment decoder source to your project. Next, make a schematic symbol of the seven segment and add it to your schematic of Fig. 9. Connect the output of the flip-flops to the 3 lower input bits of the seven segment decoder. Connect the MSB input of 7-segment decoder to GND. Add markers to the outputs of the 7-segment decoder. Save and compile the schematic.

- E) Assign the pin numbers in the I/O planning step. The clock input should be connected to a push button on the board. After generating the .ucf file, the following statement needs to be added to the .ucf file to avoid error messages.

```
NET "XXXXXX" CLOCK_DEDICATED_ROUTE = "FALSE";
```

XXXXXX should be replaced by the clock assignment given your schematic file.

- F) Download and test it on the board.

- 2) A) Write a VHDL description code for a self starter 4 bit sequence counter that counts up/down odd numbers. All unused states should go to state 0001. Use **UD** input as your up/down controller. If **UD = 1** count up and if **UD = 0** count down.

Turn in a hard copy as prelab.

- B) Compile and simulate your code.

For simulation: Assign 2ns period to the clock input and run for 30ns.

- C) In a new VHDL source file include a copy of the counter and a copy of 7-segment decoder (lab 7 VHDL) as components. Connect the outputs of the counter to inputs of the 7-segment decoder via signals.

( Make sure the 7-segment decoder source file is in your working directory and it is added to your current project.) Compile the new file. No simulation is required.

- D) Assign the pin numbers in the I/O planning step. The clock input should be connected to a push button on the board. After generating the .ucf file, the following statement needs to be added to the .ucf file to avoid error messages.

```
NET "XXXXXX" CLOCK_DEDICATED_ROUTE = "FALSE";
```

XXXXXX should be replaced by the clock assignment given your VHDL file.

- E) Download and test it on the board.

**The schematic diagrams and the simulation results of part 1 should be included in the report. For part 2 all final VHDL files and the simulation results should be included.**