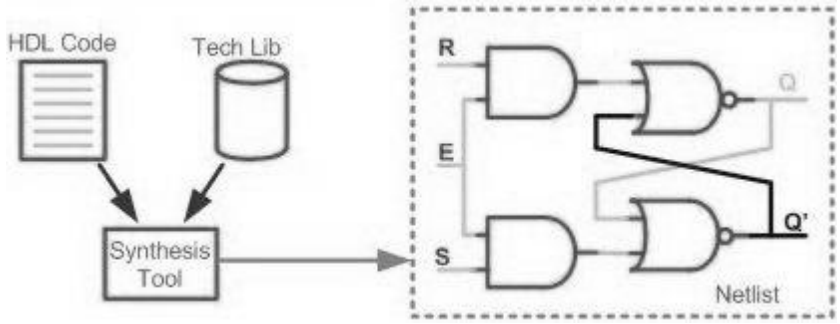
Tutorial for Verilog Synthesis Lab (Part 2)

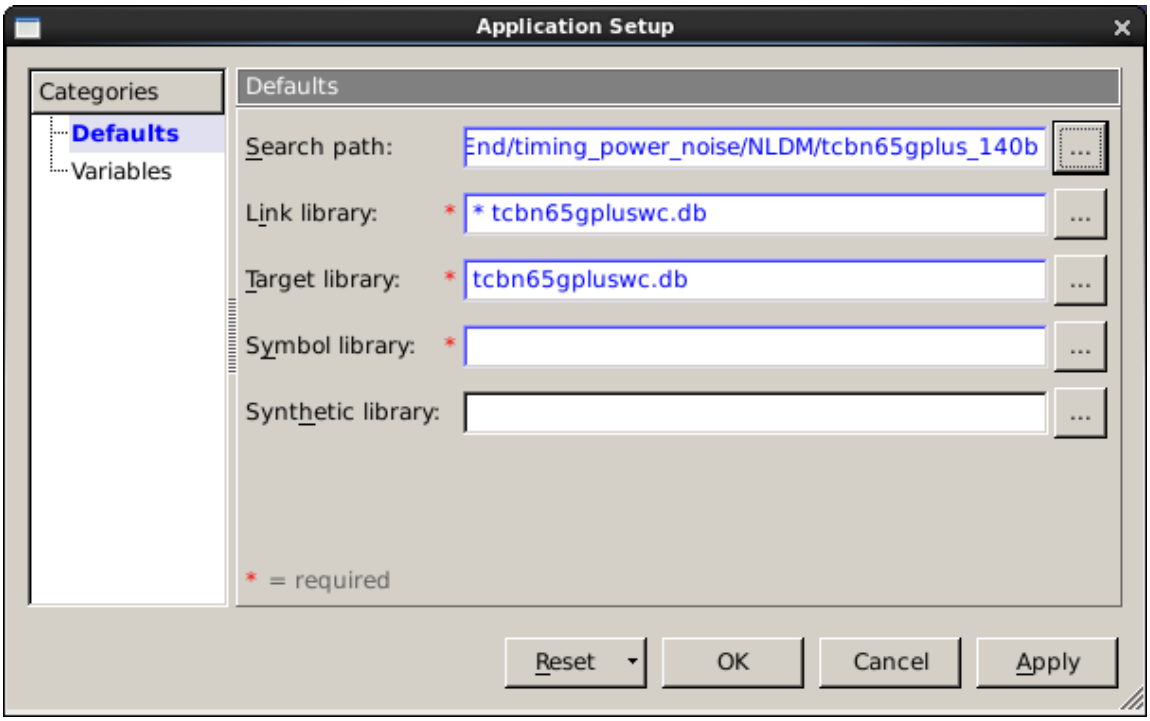
Before you synthesize your code, you must absolutely make sure that your verilog code is working properly. You will waste your time if you synthesize bad code!

A synthesizer takes a high-level design file (HDL code) and produces a gate level representation of the design using a technology library. The same HDL code can potentially be represented at the gate level in multiple ways. Using a synthesis tool, we will try to get an optimized representation of our HDL code.

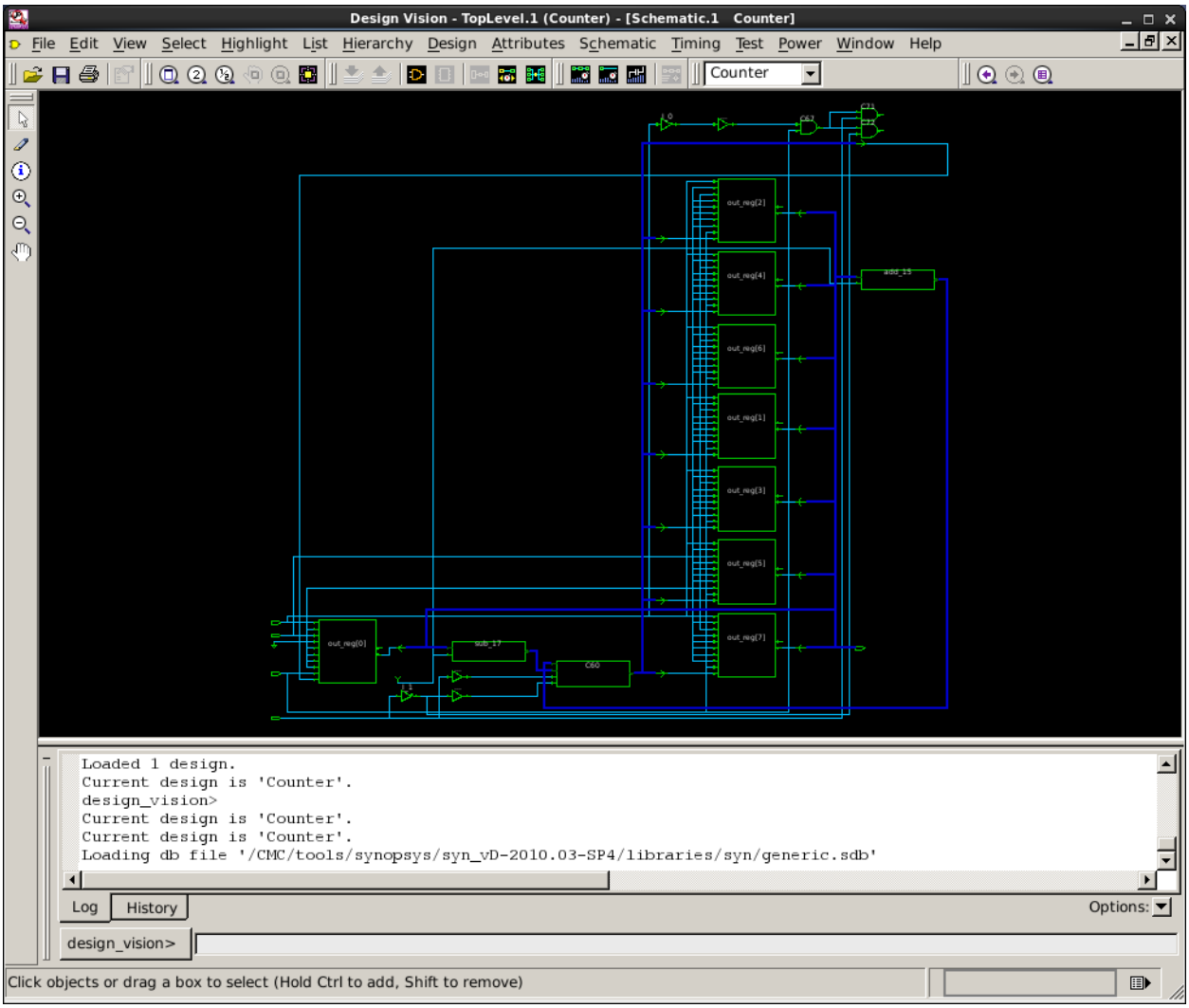


In this tutorial, we will use the simple counter that we encoded and simulated in Part 1. To complete the lab requirement, you must synthesize the serial multiplier that you have written, tested and verified.

1. For synthesis, we will use Design Vision from Synopsys. To initiate Design Vision, In a Terminal, go to your lab2 directory (i.e. **cd /elec4708/lab2**) and type "**design_vision**". You should see a blank window of Design Vision. You may explore the menus to familiarize yourself with the tool.
2. Click **File -> Setup -> Defaults** to setup the default environment.
3. In the "Search path" field, add the following (leave the existing paths):
   `/CMC/kits/tsmc_65nm_libs/tcbn65gplus_200a/TSMCHOME/digital/Front_End/timing_power_noise/NLDM/tcbn65gplus_140b`
4. In the "Link library" field replace "* your library.db" with the following
   `* tcbn65gpluswc.db`
5. In the "Target library" field, replace "your_library.db" with:
   `tcbn65gpluswc.db`
6. In the "Symbol library" field, remove "your_library.db" and keep the field blank. The window should now match the following figure. Click Apply to save your setup and then close the window.



7. Choose **File -> Read**, and then browse for the verilog file. Some messages will appear in the Log window.
8. Choose **Schematic -> New Design Schematic View** and the schematic of the Verilog module should appear.



9. We have to compile the design to get an optimal synthesized implementation. For the simple counter that we have, we will only add capacitive loading at the output port and compile it. We will use a capacitive load of 0.1 pF. To do this, enter the following into the command input filed at the bottom of the Design Vision window:

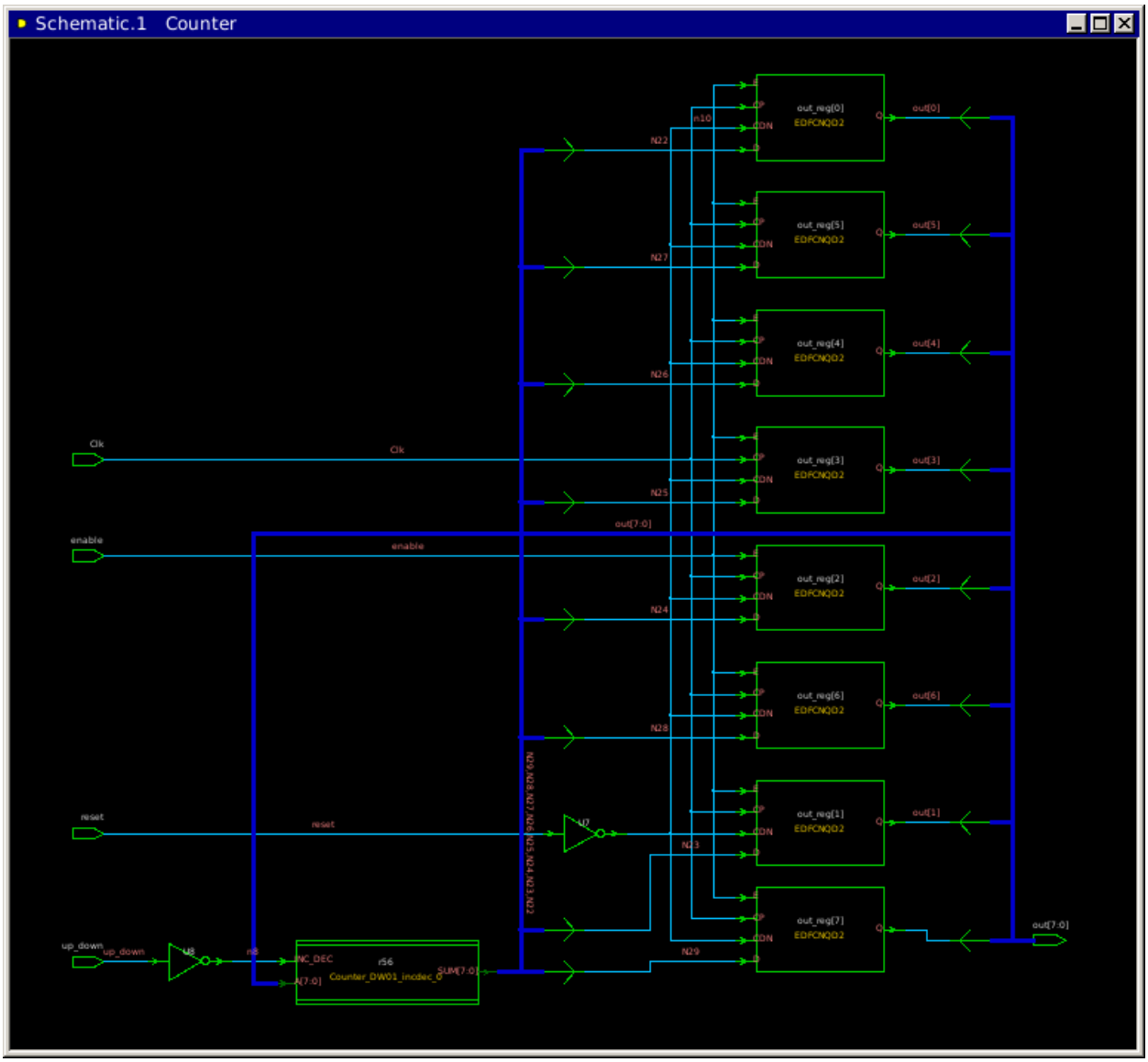
   ```
   set_load 0.1 [all_outputs]
   ```

   Alternatively, the Symbol level view is convenient for applying attributes and constraints to a design. Select the output pin (port) first, then click Attribute->Operating Environment->Load, enter 0.2 in capacitive load field (do not put a unit as no character is permitted, you must use the default unit of pF). Click Apply, and then Cancel. Deselect the output pin by clicking any blank place on the schematic.

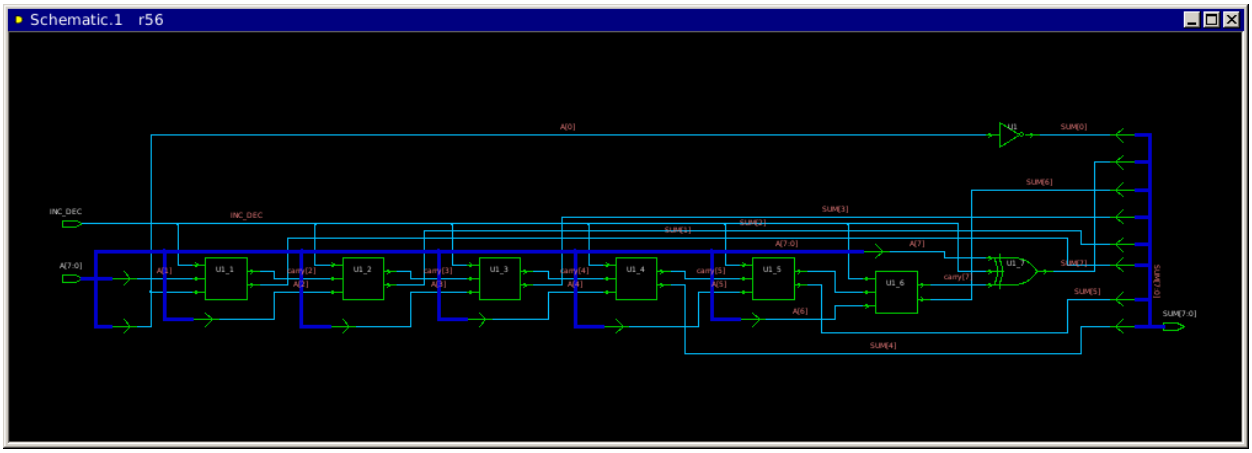
10. Enter “compile” in the command input field at the bottom of the Design Vision window. Wait until the synthesis is done. Read through all the information produced in the command window.

```
  Beginning Pass 1 Mapping
  ------------------------
  Processing 'Counter'

  Updating timing information
Information: Updating design information... (UID-85)
Information: Design 'Counter' has no optimization constraints set. (OPT-108)

  Beginning Implementation Selection
  ----------------------------------
  Processing 'Counter_DW01_incdec_0'

  Beginning Mapping Optimizations  (Medium effort)
  --------------------------------
  Structuring 'Counter'
  Mapping 'Counter'

   ELAPSED            WORST NEG TOTAL NEG  DESIGN
    TIME      AREA      SLACK     SLACK   RULE COST         ENDPOINT
  --------- --------- --------- --------- --------- --------------------------
    0:00:05     160.6      0.00       0.0       1.4
    0:00:05     160.6      0.00       0.0       1.4
    0:00:05     160.6      0.00       0.0       1.4
    0:00:05     160.6      0.00       0.0       1.4
    0:00:05     160.6      0.00       0.0       1.4
    0:00:05     149.8      0.00       0.0       1.3
    0:00:05     149.8      0.00       0.0       1.3
    0:00:05     149.8      0.00       0.0       1.3
    0:00:05     149.8      0.00       0.0       1.3
    0:00:05     149.8      0.00       0.0       1.3
    0:00:05     157.3      0.00       0.0       0.2
    0:00:05     158.4      0.00       0.0       0.0
    0:00:05     158.4      0.00       0.0       0.0
    0:00:05     158.4      0.00       0.0       0.0
    0:00:05     158.4      0.00       0.0       0.0



  Beginning Delay Optimization Phase
  ----------------------------------

   ELAPSED            WORST NEG TOTAL NEG  DESIGN
    TIME      AREA      SLACK     SLACK   RULE COST         ENDPOINT
  --------- --------- --------- --------- --------- --------------------------
    0:00:05     158.4      0.00       0.0       0.0
    0:00:05     158.4      0.00       0.0       0.0
    0:00:06     158.4      0.00       0.0       0.0


  Beginning Area-Recovery Phase  (cleanup)
  ----------------------------

   ELAPSED            WORST NEG TOTAL NEG  DESIGN
    TIME      AREA      SLACK     SLACK   RULE COST         ENDPOINT
  --------- --------- --------- --------- --------- --------------------------
    0:00:06     158.4      0.00       0.0       0.0
    0:00:06     158.4      0.00       0.0       0.0
    0:00:06     158.4      0.00       0.0       0.0
    0:00:06     158.4      0.00       0.0       0.0
    0:00:06     158.4      0.00       0.0       0.0
    0:00:06     158.4      0.00       0.0       0.0
    0:00:06     158.4      0.00       0.0       0.0
    0:00:06     158.4      0.00       0.0       0.0
    0:00:06     158.4      0.00       0.0       0.0
    0:00:06     158.4      0.00       0.0       0.0
    0:00:06     158.4      0.00       0.0       0.0
Loading db file '/CMC/kits/tsmc_65nm_libs/tcbn65gplus_200a/TSMCHOME/digital/Front_End/timing_power_noise/NLDM/tcbn65gplus_140b/tcbn65gpluswc.db'

  Optimization Complete
  ---------------------
1
Current design is 'Counter'.
```

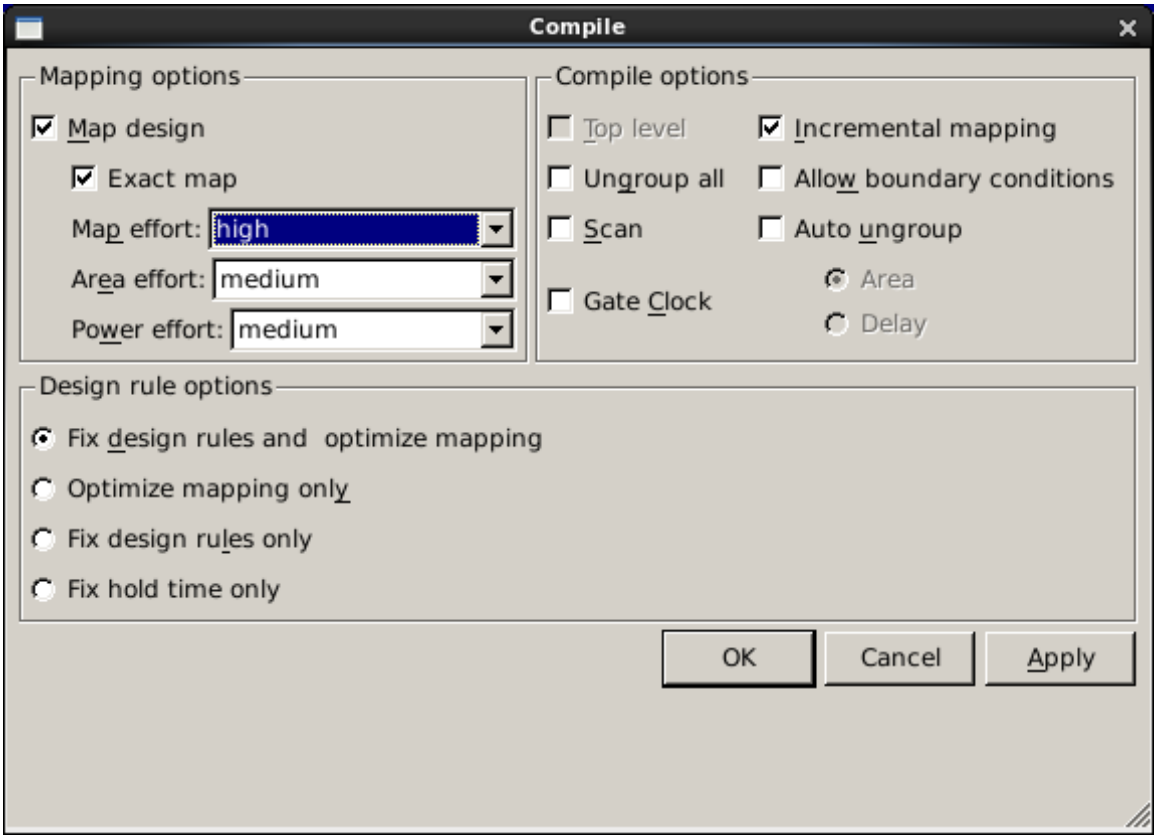
11. Choose **Schematic -> New Design Schematic View** again to show the new implementation schematic generated by the synthesizer. It has actually optimized your design automatically! The synthesizer optimizes a design based on constraints while keeping the functionality same.



In the above synthesized schematic, there is a box that is not a primitive digital element (in the bottom left). Double clicking this box will go into it and you can see primitives that compose this function.



12. Scroll through the command window and you will notice different steps of optimization and a lot of other information. By default, we have used medium effort for optimization. As you increase your effort, the synthesis time becomes longer.
13. Click **Design->Compile Design**. You can access some optimization criteria from here. When you click OK, the synthesizer will compile with the selected options. Since the counter used in this tutorial is relatively simple, changing the compiling efforts do not change the schematic very much. It may make a huge difference for your multiplier however.



Alternatively, you can enter “compile –map_effort high” in the command input field to synthesize with high effort. Type “compile -?” to check out available command line options.

14. To save your design in an unmapped db format, select **File -> Save**, and name your design as counter.ddc, choose DDC as the File Format. Select the Save all designs in hierarchy option. When a design is saved as a .ddc file, the design plus all attributes are saved. The equivalent command is: “write -format ddc -hierarchy –output counter.ddc”.
15. For your report, select the top level design, click **Design -> Report Design**. You can also produce other reports like timing, area, power, and so on. You can direct the output to a file for later reference. The equivalent input commands are: “report_area” and “report_timing”.
16. To exit Design Vision, select **File->Quit**. Click OK.