Homework Assignment 5

(Due Mar. 13th at the beginning of the class)

0. Preparation
    - Download the following file into your working directory.
        - o wget http://www.eecs.wsu.edu/~ee434/Homework/hw05.tar.gz
    - Unzip it.
        - o tar xvzf hw05.tar.gz
    - Source synopsys.sh
        - o source synopsys.sh

1. [Synthesis and Analysis, **10 points**]
    - In this problem, we will synthesize and analyze a four-bit adder.
    - Open add4.v and see the source code.
    - Open add4.tcl and see the script.
    - Run design compiler.
        - o *design_vision –no_gui*
    - Run the following script.
        - o design_vision> *source add4.tcl*
    - It will compile the source code and synthesize a four-bit adder.
    - Let's analyze the circuit.
    - Run the following command to analyze area.
        - o design_vision> *report_area*

```
Number of ports:              14
Number of nets:               30
Number of cells:              21
Number of references:          5

Combinational area:       24.738000
Noncombinational area:     0.000000
Net Interconnect area:    undefined  (Wire load has zero net area)

Total cell area:          24.738000
Total area:               undefined
```

    - There are 14 ports (A:four bits, B: four bits, Cin, S: five bits, 4+4+1+5=14).
    - There are 30 nets. Open add4_mapped.v (netlist) and count the number of nets. There are 16 internal nets (wire n3, n4, …) and 14 input/output nets (primary input and output nets). 16+14=30.
    - There are 21 cells (= instances = standard cells). Open add4_mapped.v and count the number of instances.

- # references is # types of the standard cells used. Open add4_mapped.v and count the number of the types of the standard cells used. INV_X1, AOI22_X1, OR2_X1, XOR2_X1, OAI21_X1.
- The total area of the combinational cells is 24.738um^2.
- The total area of the non-combinational cells (FFs, latches, fill cells, etc.) is 0.
- Run the following command to analyze timing.
    - design_vision> *report_timing*

```
Startpoint: B[0] (input port)
Endpoint: S[4] (output port)
Path Group: (none)
Path Type: max

Des/Clust/Port     Wire Load Model       Library
------------------------------------------------
myAdd4             5K_hvratio_1_1        NangateOpenCellLibrary

Point                                    Incr       Path
-----------------------------------------------------------
input external delay                     0.00       0.00 f
B[0] (in)                                0.00       0.00 f
U19/ZN (OAI21_X1)                        0.04       0.04 r
U18/ZN (OAI21_X1)                        0.04       0.09 f
U15/ZN (OR2_X1)                          0.06       0.15 f
U14/ZN (AOI22_X1)                        0.05       0.21 r
U13/ZN (INV_X1)                          0.03       0.24 f
U10/ZN (OR2_X1)                          0.06       0.30 f
U9/ZN (AOI22_X1)                         0.05       0.35 r
U8/ZN (INV_X1)                           0.03       0.38 f
U5/ZN (OR2_X1)                           0.06       0.44 f
U4/ZN (AOI22_X1)                         0.05       0.50 r
U3/ZN (INV_X1)                           0.02       0.52 f
S[4] (out)                               0.00       0.52 f
data arrival time                                   0.52
-----------------------------------------------------------
(Path is unconstrained)
```

- It shows a worst path (there could be multiple worst paths).
- The start point of the worst path shown above is B[0].
- The end point of the worst path shown above is S[4] (this is actually the carry-out output port).
- The delay of the worst path is 0.52ns (520ps).
- The path is “unconstrained”, i.e., it doesn’t have any timing constraint.
- Run the following command to analyze power.
    - design_vision> *report_power*

```
Global Operating Voltage = 1.1
Power-specific unit information :
    Voltage Units = 1V
    Capacitance Units = 1.000000ff
    Time Units = 1ns
    Dynamic Power Units = 1uW    (derived from V,C,T units)
    Leakage Power Units = 1nW


  Cell Internal Power  =   9.3815 uW   (67%)
  Net Switching Power  =   4.7099 uW   (33%)
                         ---------
Total Dynamic Power    =  14.0914 uW  (100%)

Cell Leakage Power     = 580.0209 nW
```

- "Cell Internal Power" is the power consumed inside cells.
- "Net Switching Power" is the power consumed to drive nets.
- [**Submit**]
    - Create a netlist for a four-bit ripple-carry adder. Use only the following standard cell to implement it.
    - Standard cell: FA_X1
        - Primary inputs: A, B, CI
        - Primary outputs: S, CO
    - Load the netlist into Design Compiler (modify add4.tcl. Do not run "compile –exact_map" and "write –format …").
    - Show area, timing, and power.

2. [Optimization, **20 points**]
    - In this problem, we will optimize the HDL code and compare area, timing, and power.
    - Run Design Compiler and source "add4_read.tcl" to read in the HDL code.
    - Run the following command to synthesize and optimize the code.
        - design_vision> *set_max_delay -from {A* B* Cin} -to {S*} 0.5*
        - This sets a max. delay (0.5ns) from any input pin to any output pin.
    - Compile.
        - design_vision> *compile*
    - Get total area, worst path delay (**data arrival time**), cell internal power, net switching power, cell leakage power, and total power (= internal power + switching power + leakage power).
    - Change the max. delay constraint from 0.5ns to 0.49ns and re-compile it.
        - design_vision> *set_max_delay -from {A* B* Cin} -to {S*} 0.49*
        - design_vision> *compile*
    - Get the area, delay, and power numbers again.

- [**Submit**] Get {area, delay, internal power, switching power, leakage power, total power} for each max. delay constraint (d_{MAX}) and fill in the following table.

d_{MAX} (ns)	Area (um^2)	Worst path delay (ps)	Cell internal power (uW)	Net switching power (uW)	Cell leakage power (uW)	Total power (uW)
0.5						
0.49						
0.48						
…						
0.xx		VIOLATED				

3. [Design, Synthesis, and Optimization, **30 points**]
    - Write a Verilog code for a 32-bit adder.
        - Primary inputs: [31:0] A, [31:0] B, Cin
        - Primary outputs: [32:0] S
    - Synthesize and time it.
    - Use "set_max_delay –from {A* B* Cin} –to {S*} XX" to set up timing constraints.
    - Minimize the longest-path delay (but you should not violate the timing constraint).
    - [**Submit**] Area, worst-path delay, cell internal power (PI), net switching power (PS), cell leakage power (PL), and total power (PI+PS+PL).