EE 465 Lab Report

Lab 5 – Synthesis Constraint

Written by: Anh Q. Ho, Aaron Pederson
Lab Section – Monday 04:10pm

ABSTRACT

Synthesis procedures are truly depends on how the architecture or system engineers set their constraints since they design and know when and what signals should go where. By setting constraints, students can emulate the real working environment for the design of this experiment. In this lab, students are to only cover the most important and common used constraints in the semiconductor industries.

INTRODUCTION

Due to the expensive cost of design and manufacture the chip, in this lab students are to make sure the chip will not fail in a real world situation after it is manufactured. Also, students are to discover how the constraint can affect the synthesis result, area, power consumption or generation and the timing delay of the chip.

Objectives

- Setting up synthesis constraints
- Circuit timing, area, and power reports

In this lab there are multiple steps students will need to follow in order to achieve the above objectives. Below is just a quick overview of the lab tasks. Please follow the lab manual to get all the information needed.

NOTE: All the commands for this lab are purposely to run the files name ALT_MULTADD, if the lab 1 verilog file name is different please change it to ALT_MULTADD or change all command with "ALT_MULTADD" to the module name.

It is a lot easier to change the Verilog file rather than changing every command.

> The Verilog module should be name AL_MULTADD and should be saved as "ALT_MULTADD.v"

1. Exploring the effects of various constraints
    a. Delete the line "read_sdc ./scripts/design.sdc" and everything below it in the run_synth.tcl file.
    b. Launch RTL compiler and run the script
    c. Run the following commands in the terminal to set time and load unit
    WARNING: there are no "spaces" after the colons in the first two lines of the commands.

```
dc::set_time_unit -picoseconds
dc::set_load_unit -picofarads
define_clock -period 400 -name clk [clock_ports] -rise 50 -fall 50
set_attribute clock_network_late_latency 1 clk
set_attribute clock_source_early_latency 1.5 clk
set_attribute clock_setup_uncertainty {100 50} clk
set_attribute slew {100 110 110 120}  [find / -clock clk]
report clocks
external_delay -clock clk -input 200 -name in_delay/designs/ALT_MULTADD/ ports_in/*
external_delay -clock clk -output 200 -name out_delay/designs/ALT_MULTADD/ ports_out/*
set_attribute external_driver [find [find / -libcell MUX2ND2] -libpin ZN] /designs/ALT_MULTADD/ports_in/*
set_attribute external_pin_cap 2 /designs/ALT_MULTADD/ports_out/*
set_attribute max_fanout 10 /designs/*
synthesize -to_mapped -effort high
```

d. Use the following commands to begin synthesis and to generate various reports

```
report area > ${OUTPUT_DIR}/area.rpt
report gates > ${OUTPUT_DIR}/gates.rpt
report timing > ${OUTPUT_DIR}/timing.rpt
report timing -lint > ${OUTPUT_DIR}/lint.rpt
report summary > ${OUTPUT_DIR}/summary.rpt
report power > ${OUTPUT_DIR}/power.rpt
```

e. The last few lines of commands are used to store results

```
write -mapped > ${OUTPUT_DIR}/jpeg_mapped.v
write_script > ${OUTPUT_DIR}/jpeg_mapped.g
write_sdc > ${OUTPUT_DIR}/jpeg_mapped.sdc
```

i. Study the timing report in the “run_dir” directory. Please find the “How_to_read_time_report.pdf” in the class website for guidance.

ii. Change one timing constraint value from Step c above, and re-run the synthesis and timing report.
NOTE: try to alter the constraint by at least a factor of 10, for easier notice in report adjustments.

iii. Compare and contrast the new report to the original.

iv. Explain the behavior of each constraint.

The document RTL_Compiler_Constraints_and_STA.pdf may help with the previous step.

2. Controlling area/power tradeoff.

RESULTS & DISCUSSION

The following graphs show the comparisons of the effects of different constraint changes on different results of the synthesis. We selected a few variables to enter in to an excel spreadsheet and graphed each one against the changed constraints.

Table 1 - Simulation #1 and 2 results.

run 1 (original synthesis)	
clock_network_late_latency	1
clock_source_early_latency	1.5
clock_setup_uncertainty	{100 50}
slew	{100 110 110 120}
power_optimization_effort	Default(medium)
Total Cells	3581
Total Area	7892
Leakage Power (nW)	79669.333
Dynamic Power (nW)	942023.152
Total Power (nW)	1021692.485
Timing slack (ps)	-1212

run 2(half slew constraint)	
clock_network_late_latency	1
clock_source_early_latency	1.5
clock_setup_uncertainty	{100 50}
slew	{50 55 55 60}
power_optimization_effort	Default(medium)
Total Cells	3467
Total Area	7546
Leakage Power (nW)	75073.107
Dynamic Power (nW)	915991.741
Total Power (nW)	991064.847
Timing slack (ps)	-1232

Table 2 - Simulation #3 & 4 results.

run 3(low power optimization)	
clock_network_late_latency	1
clock_source_early_latency	1.5
clock_setup_uncertainty	{100 50}
slew	{100 110 110 120}
power_optimization_effort	low
Total Cells	3581
Total Area	7892
Leakage Power (nW)	79669.333
Dynamic Power (nW)	942023.152
Total Power (nW)	1021692.485
Timing slack (ps)	-1212

run 4(high power optimization)	
clock_network_late_latency	1
clock_source_early_latency	1.5
clock_setup_uncertainty	{100 50}
slew	{100 110 110 120}
power_optimization_effort	high
Total Cells	3581
Total Area	7892
Leakage Power (nW)	79669.333
Dynamic Power (nW)	942023.152
Total Power (nW)	1021692.485
Timing slack (ps)	-1212

Table 3 - Simulation # 5 & 6 results.

run 5(decreased late latency)	
clock_network_late_latency	0.3
clock_source_early_latency	1.5
clock_setup_uncertainty	{100 50}
slew	{100 110 110 120}
power_optimization_effort	Default(medium)
Total Cells	3572
Total Area	7935
Leakage Power (nW)	79522.972
Dynamic Power (nW)	938112.303
Total Power (nW)	1017635.275
Timing slack (ps)	-1419

run 6(decreased early latency)	
clock_network_late_latency	1
clock_source_early_latency	0.2
clock_setup_uncertainty	{100 50}
slew	{100 110 110 120}
power_optimization_effort	Default(medium)
Total Cells	3587
Total Area	7839
Leakage Power (nW)	78262.534
Dynamic Power (nW)	942023.152
Total Power (nW)	1011252.585
Timing slack (ps)	-1413

Table 4 - Simulation #7 results.

run 7(decreased all constraints)	
clock_network_late_latency	0.2
clock_source_early_latency	0.2
clock_setup_uncertainty	{10 5}
slew	{1 1 1 1}
power_optimization_effort	Default(medium)
Total Cells	3625
Total Area	7916
Leakage Power (nW)	79852.398
Dynamic Power (nW)	965443.329
Total Power (nW)	1045295.727
Timing slack (ps)	-1316

From the above results; the following graphs are presented to create an easier illustration to see which constraints are better for the area, power and timing reports.

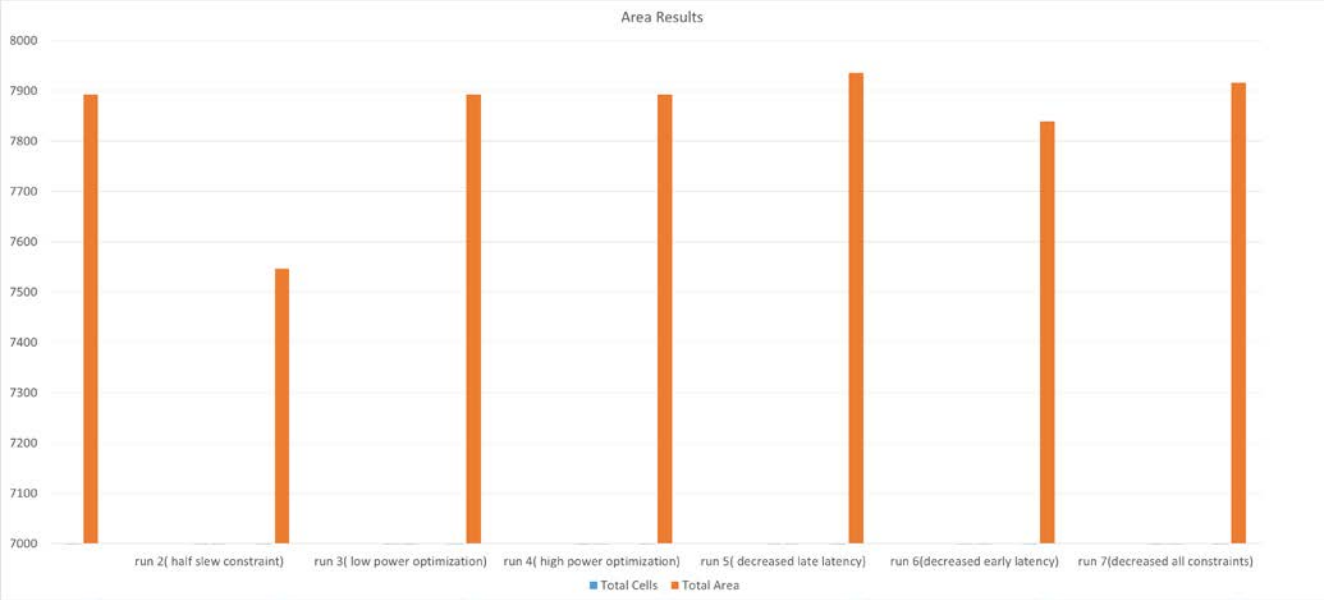

Figure 1- Area report; when the slew constraint was decreased, the area slightly decreases.

The second graph is the power. The power results are relatively consistent. The big changes are when the slew is decreased in run 2: the power decreases and in run 7 where all constraints are decreased: the power increases.

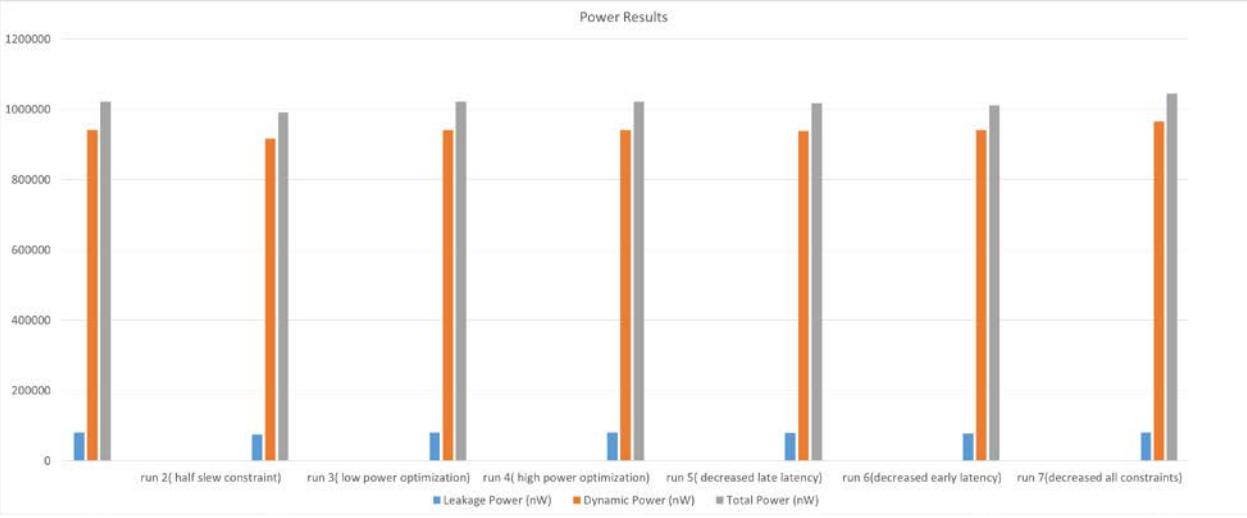

Figure 2 - Power report; best result was the original where there is no changed constraints and normal power optimization.

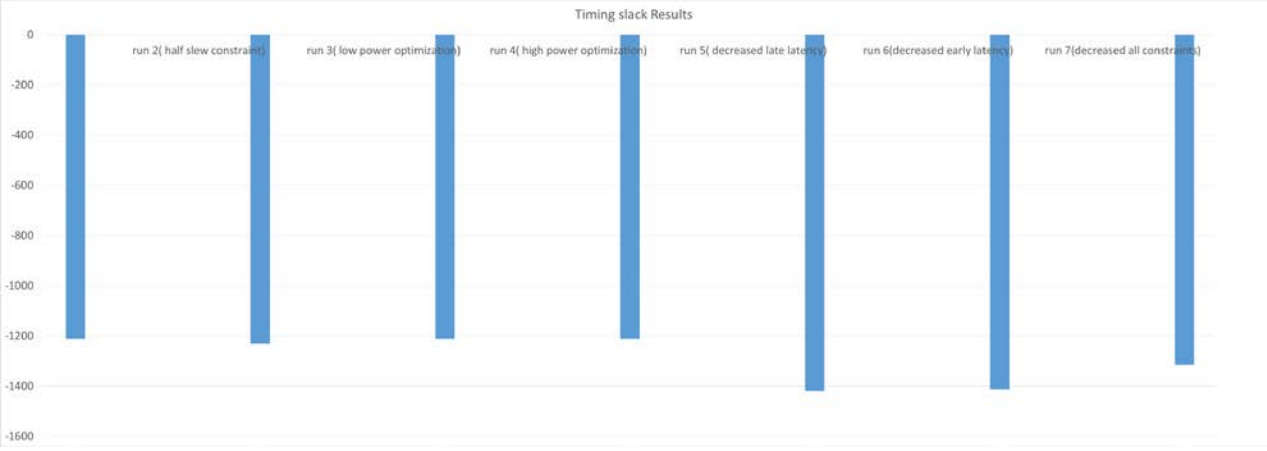

Figure 3 - Timing report.

CONCLUSION

The lab went fairly well for the group, the only concerns were the commands as mentioned before working on this lab (NOTE), the groups were using quite a bit of time trying to determine the problem since the lab manual instructed to save the file as “ALT_MULTADD” but not saying much about the naming of the module. Otherwise, the minor issues is how much to change on the timing constraints. At first, all changes were half of the original values, but not much were changed in the reports. Thus a change in a factor of ten, the alteration were well spotted.