Zack Fravel
010646947
CSCE 4914
Lab 4 Report

Synthesis Process

The synthesis process was relatively simple. After logging onto the VLSI server and using the tcsh and initrc commands, I was able to use dc_shell to synthesize my designs. Using the scripts provided in the lab .tgz, I created a template I could use for each design where the only change needed to be made was the $topname. Below is an image of the template script.

```
set_app_var search_path ". $::env(PDK_DIR)/osu_soc/lib/files $synopsys_root/libraries/syn/"
set_app_var link_library "* gscl45nm.db"
set_app_var target_library "gscl45nm.db"
set_app_var symbol_library "generic.sdb"

define_design_lib work -path ./work

set topname <name goes here>
set verilogs [glob $topname.v]

set myPeriod_ns 0.5
set myClk clk

analyze -format verilog  $verilogs

elaborate $topname

current_design $topname

link

create_clock -period $myPeriod_ns -name $myClk

compile_ultra

redirect $topname.timing.dc.rpt {report_timing}
redirect $topname.area.dc.rpt {report_area}
redirect $topname.power.dc.rpt {report_power}

write_sdc $topname.sdc
write -format verilog -hierarchy -output $topname.syn.v

quit
```

This script first properly sets the proper library variables and search paths for design vision, creates a clock signal, synthesizes the design in the FreePDK45 Library and writes out timing, area, and power analysis to three separate files.

The next step in the synthesis process was to create three of these scripts with $topname Adder64, Adder128, and FPGen. The FPGen dc.tcl script is slightly different, as shown later in the report. Once the script is properly written, I just navigate to the folder with the desired .v design and run the command `dc_shell -f dc.tcl` to perform synthesis as well as run area, timing, and power analysis.

Comparison between Adder64 and Adder128

The Adder64.v was provided, the only modification needed to be made to create a 128-bit adder was to duplicate the design with the name Adder128 and change both the size of `A`, `B`, and `sum` to `[127 : 0]` instead of `[63:0]`. In each of these comparisons, the Adder64 design will be on the left, and the Adder128 on the right.

Area

```
****************************************
Report : area
Design : Adder64
Version: M-2016.12-SP5
Date   : Sun Nov  5 13:08:42 2017
****************************************

Library(s) Used:

gscl45nm (File: /tools/library/FreePDK45/osu_soc/lib/files/gscl45nm.db)

Number of ports:                          194
Number of nets:                           257
Number of cells:                           64
Number of combinational cells:             64
Number of sequential cells:                 0
Number of macros/black boxes:               0
Number of buf/inv:                          0
Number of references:                       1

Combinational area:               570.668823
Buf/Inv area:                       0.000000
Noncombinational area:              0.000000
Macro/Black Box area:               0.000000
Net Interconnect area:      undefined  (No wire

Total cell area:                  570.668823
Total area:                       undefined
1
```

```
****************************************
Report : area
Design : Adder128
Version: M-2016.12-SP5
Date   : Sun Nov  5 13:09:31 2017
****************************************

Library(s) Used:

gscl45nm (File: /tools/library/FreePDK45/osu_soc/lib/files/gscl

Number of ports:                          386
Number of nets:                           513
Number of cells:                          128
Number of combinational cells:            128
Number of sequential cells:                 0
Number of macros/black boxes:               0
Number of buf/inv:                          0
Number of references:                       1

Combinational area:              1141.337646
Buf/Inv area:                       0.000000
Noncombinational area:              0.000000
Macro/Black Box area:               0.000000
Net Interconnect area:      undefined  (No wire load specified)

Total cell area:                 1141.337646
Total area:                      undefined
1
```

Timing

The timing comparisons have been shortened to save space on paper, however I have included the full reports in the “resources” folder. The final timing is circled in red at the bottom, again Adder128 on the right, Adder64 on the left.

```
Information: Updating design information... (UID-85)

****************************************
Report : timing
        -path full
        -delay max
        -max_paths 1
Design : Adder64
Version: M-2016.12-SP5
Date   : Sun Nov  5 13:08:42 2017
****************************************

Operating Conditions: typical   Library: gscl45nm
Wire Load Model Mode: top

  Startpoint: cin (input port)
  Endpoint: sum[63] (output port)
  Path Group: (none)
  Path Type: max

  Point                                     Incr       Path
  -------------------------------------------------------------
  input external delay                      0.00       0.00 f
  cin (in)                                  0.00       0.00 f
  intadd_0/U65/YC (FAX1)                    0.08       0.08 f
  intadd_0/U64/YC (FAX1)                    0.08       0.17 f

                              •
                              •
                              •

  intadd_0/U5/YC (FAX1)                     0.08       4.97 f
  intadd_0/U4/YC (FAX1)                     0.08       5.06 f
  intadd_0/U3/YC (FAX1)                     0.08       5.14 f
  intadd_0/U2/YS (FAX1)                     0.07       5.21 f
  sum[63] (out)                             0.00       5.21 f
  data arrival time                                    5.21
  -------------------------------------------------------------
  (Path is unconstrained)

1
```

```
Information: Updating design information... (UID-85)

****************************************
Report : timing
        -path full
        -delay max
        -max_paths 1
Design : Adder128
Version: M-2016.12-SP5
Date   : Sun Nov  5 13:09:31 2017
****************************************

Operating Conditions: typical   Library: gscl45nm
Wire Load Model Mode: top

  Startpoint: cin (input port)
  Endpoint: sum[127] (output port)
  Path Group: (none)
  Path Type: max

  Point                                     Incr       Path
  -------------------------------------------------------------
  input external delay                      0.00       0.00 f
  cin (in)                                  0.00       0.00 f
  intadd_0/U129/YC (FAX1)                   0.08       0.08 f
  intadd_0/U128/YC (FAX1)                   0.08       0.17 f
  intadd_0/U127/YC (FAX1)                   0.08       0.25 f

                              •
                              •
                              •

  intadd_0/U5/YC (FAX1)                     0.08      10.19 f
  intadd_0/U4/YC (FAX1)                     0.08      10.27 f
  intadd_0/U3/YC (FAX1)                     0.08      10.35 f
  intadd_0/U2/YS (FAX1)                     0.07      10.43 f
  sum[127] (out)                            0.00      10.43 f
  data arrival time                                   10.43
  -------------------------------------------------------------
  (Path is unconstrained)
```

Power

For the power analysis, they're shown one at a time, starting with Adder64 and following with Adder128.

```
****************************************
Report : power
        -analysis_effort low
Design : Adder64
Version: M-2016.12-SP5
Date   : Sun Nov  5 13:08:42 2017
****************************************


Library(s) Used:

    gscl45nm (File: /tools/library/FreePDK45/osu_soc/lib/files/gscl45nm.db)


Operating Conditions: typical   Library: gscl45nm
Wire Load Model Mode: top


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


  Cell Internal Power  = 484.0232 uW   (81%)
  Net Switching Power  = 109.9643 uW   (19%)
                         ---------
Total Dynamic Power    = 593.9875 uW  (100%)

Cell Leakage Power     =   2.5095 uW

Information: report_power power group summary does not include estimated clock tree power. (PWR-789)

                 Internal         Switching           Leakage            Total
Power Group      Power            Power               Power              Power   (   %    )  Attrs
--------------------------------------------------------------------------------------------------
io_pad             0.0000           0.0000            0.0000             0.0000  (   0.00%)
memory             0.0000           0.0000            0.0000             0.0000  (   0.00%)
black_box          0.0000           0.0000            0.0000             0.0000  (   0.00%)
clock_network      0.0000           0.0000            0.0000             0.0000  (   0.00%)
register           0.0000           0.0000            0.0000             0.0000  (   0.00%)
sequential         0.0000           0.0000            0.0000             0.0000  (   0.00%)
combinational      0.4840           0.1100        2.5095e+03             0.5965  ( 100.00%)
--------------------------------------------------------------------------------------------------
Total              0.4840 mW        0.1100 mW     2.5095e+03 nW         0.5965 mW
1
```

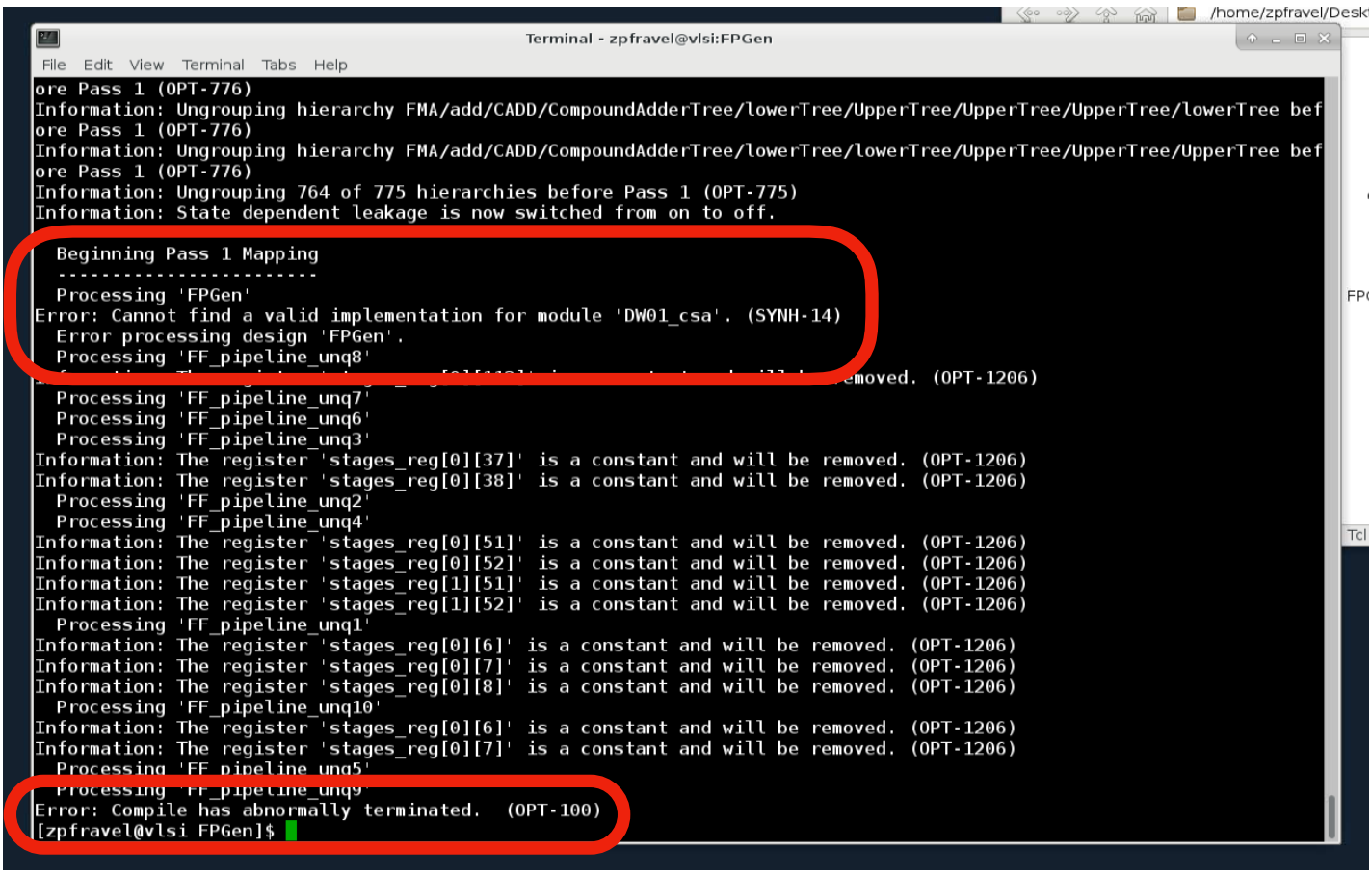
```
****************************************
Report : power
        -analysis_effort low
Design : Adder128
Version: M-2016.12-SP5
Date   : Sun Nov  5 13:09:31 2017
****************************************


Library(s) Used:

    gscl45nm (File: /tools/library/FreePDK45/osu_soc/lib/files/gscl45nm.db)


Operating Conditions: typical   Library: gscl45nm
Wire Load Model Mode: top


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


  Cell Internal Power  = 973.3128 uW   (81%)
  Net Switching Power  = 223.3008 uW   (19%)
                         ---------
Total Dynamic Power    =   1.1966 mW  (100%)

Cell Leakage Power     =   5.0189 uW

Information: report_power power group summary does not include estimated clock tree power. (PWR-789)

                 Internal         Switching           Leakage            Total
Power Group      Power            Power               Power              Power   (   %    )  Attrs
--------------------------------------------------------------------------------------------------
io_pad             0.0000           0.0000              0.0000            0.0000  (   0.00%)
memory             0.0000           0.0000              0.0000            0.0000  (   0.00%)
black_box          0.0000           0.0000              0.0000            0.0000  (   0.00%)
clock_network      0.0000           0.0000              0.0000            0.0000  (   0.00%)
register           0.0000           0.0000              0.0000            0.0000  (   0.00%)
sequential         0.0000           0.0000              0.0000            0.0000  (   0.00%)
combinational      0.9733           0.2233          5.0189e+03            1.2016  ( 100.00%)
--------------------------------------------------------------------------------------------------
Total              0.9733 mW        0.2233 mW       5.0189e+03 nW         1.2016 mW
1
```

FPGen Summary

The process for synthesizing the FPGen module is slightly different since it is laid out as a systemverilog hierarchical design. This means the script needed to have some modifications made to it. First of all, there needed to be changed references from verilog to systemverilog on the filename, as well as we needed to add another link library to design vision. On the beginning of the next page I have added a screenshot of the modified `dc.tcl` file.

```
set_app_var search_path ". $::env(PDK_DIR)/osu_soc/lib/files $synopsys_root/libraries/syn/"
set_app_var link_library "* gscl45nm.db" "dw_foundation.sldb"
set_app_var target_library "gscl45nm.db"
set_app_var symbol_library "generic.sdb"

define_design_lib work -path ./work

set topname FPGen
set systemverilogs [glob verilog/*{.sv,.v}]

set myPeriod_ns 0.5
set myClk clk

analyze -format sverilog  $systemverilogs

elaborate $topname

current_design $topname

link

create_clock -period $myPeriod_ns clk

compile_ultra

redirect $topname.timing.dc.rpt {report_timing}
redirect $topname.area.dc.rpt {report_area}
redirect $topname.power.dc.rpt {report_power}

write_sdc $topname.sdc
write -format verilog -hierarchy -output $topname.syn.v

quit
```

Using the script above, I was able to get the synthesis process started after some troubleshooting. However, I ended up still running into an issue during the synthesis process, so I was unable to ever get a successful area, timing, and power report because of the following error I was unable to get a resolution for.



Conclusions

While I was able to get an effective comparison between the two different adders, unfortunately I was not able to perform the full synthesis of the FPGen module. I have included all relevant files (verilog, gate level netlists, scripts, and reports) in the zip file in the “resources” folder. I have also included all files related to the failed FPGen synthesis. I have also included images of the full synthesis process for each module in their respective folders.