Fall 2024

Selected Topics in Computer Design-CSE416s

Final Project

Adaptive Traffic Light Controller-Part B

Team #4

Members:

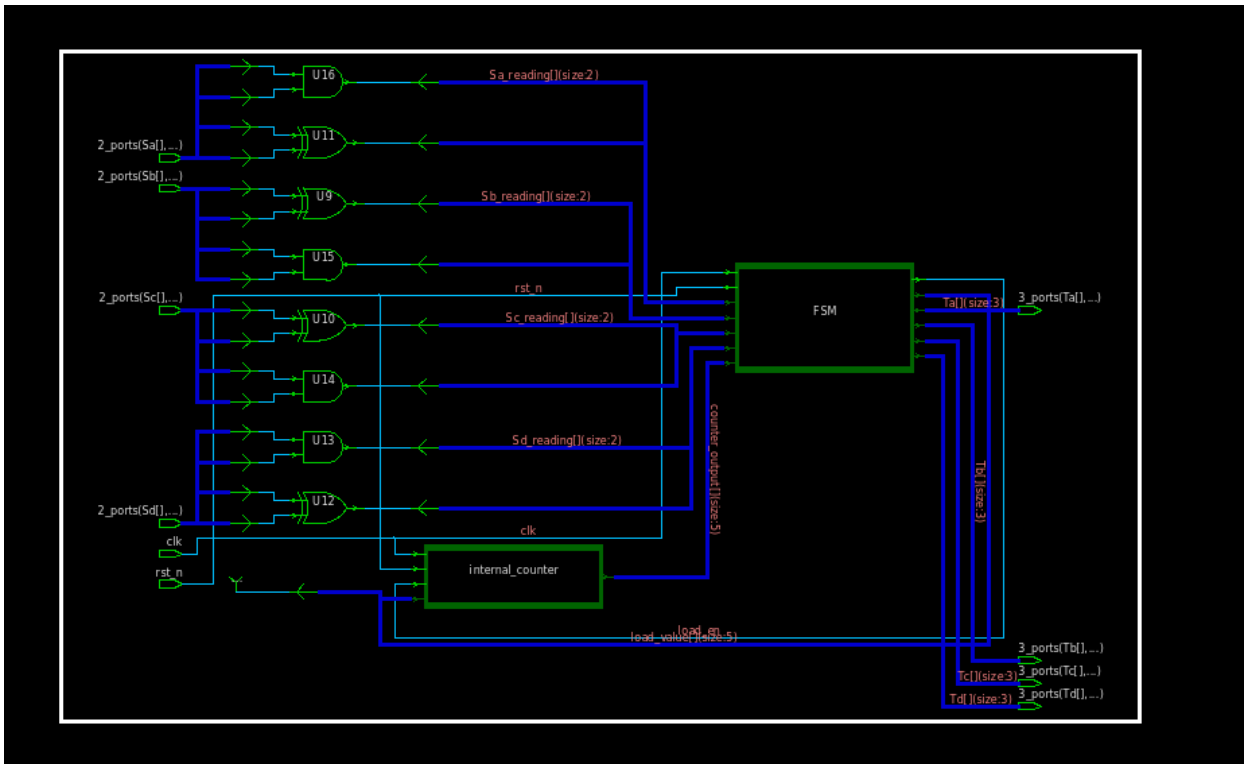
Name	Code	Department
Fares Khalaf Salman Sultan	2101371	CSE
Youssef Hany Elreweny	2101449	CSE
Ahmed Ashraf Ali	2100255	CSE
Youssef Ossama Sayed	2101240	CSE

After completing RTL verification and ensuring that our design meets the project requirements, the next step involves synthesizing the RTL code into a technology-mapped Gate Level (GL) netlist. The objective is to ensure there are no timing violations and to perform formal verification to confirm that the functionality remains correct after synthesis.

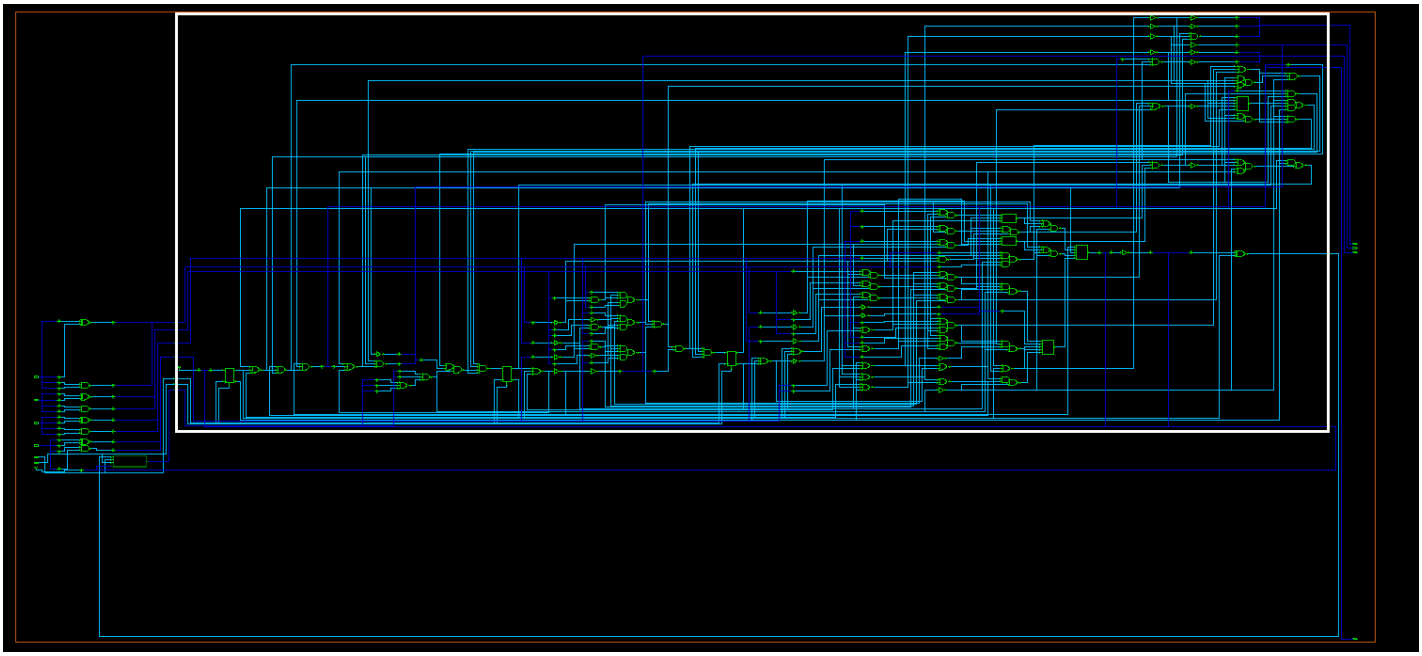
❖ Steps:

1. **Synthesis Script Implementation:**
   We created a synthesis script (syn_script.tcl) to define the RTL code and standard cells of the given Library for the synthesis tool, Design Compiler.
2. **Constraints Script Implementation:**
   A constraints script (cons.tcl) was developed to specify system clock properties, including clock period, uncertainty, and rising/falling edge times. The script also sets input and output path delays, driving cells, output load, and wire loads to enable the tool to perform Static Timing Analysis (STA) using these constraints.
3. **Netlist Generation and Validation:**
   After running the synthesis and constraints scripts, a GL netlist was generated. We ensured that the synthesis process was error-free and verified that no unintended components, such as latches, were present.
4. **Constraints Review:**
   We reviewed the constraints reports to confirm that all constraints were met and verified that there were no timing violations.
5. **Formal Verification:**
   Finally, a formal verification script (syn_fm_script.tcl) was implemented to compare the GL netlist with the original RTL code. Using the Formality tool, we confirmed that the system's functionality remained unchanged. The formal verification process was concluded by extracting and reviewing the verification reports.

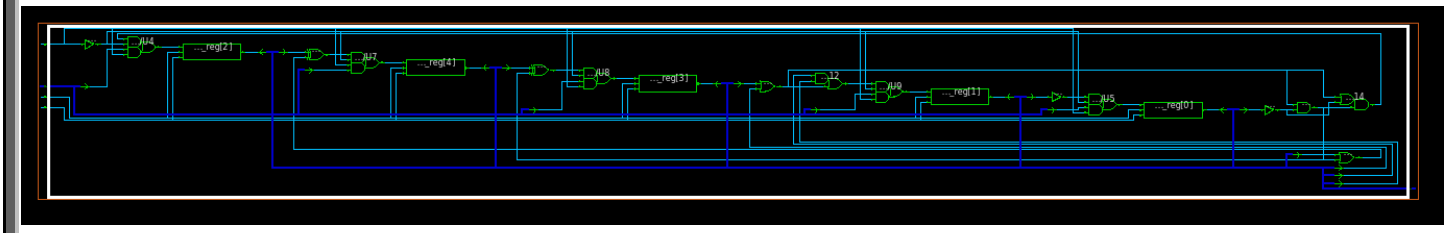
❖ Schematic :



TopModule

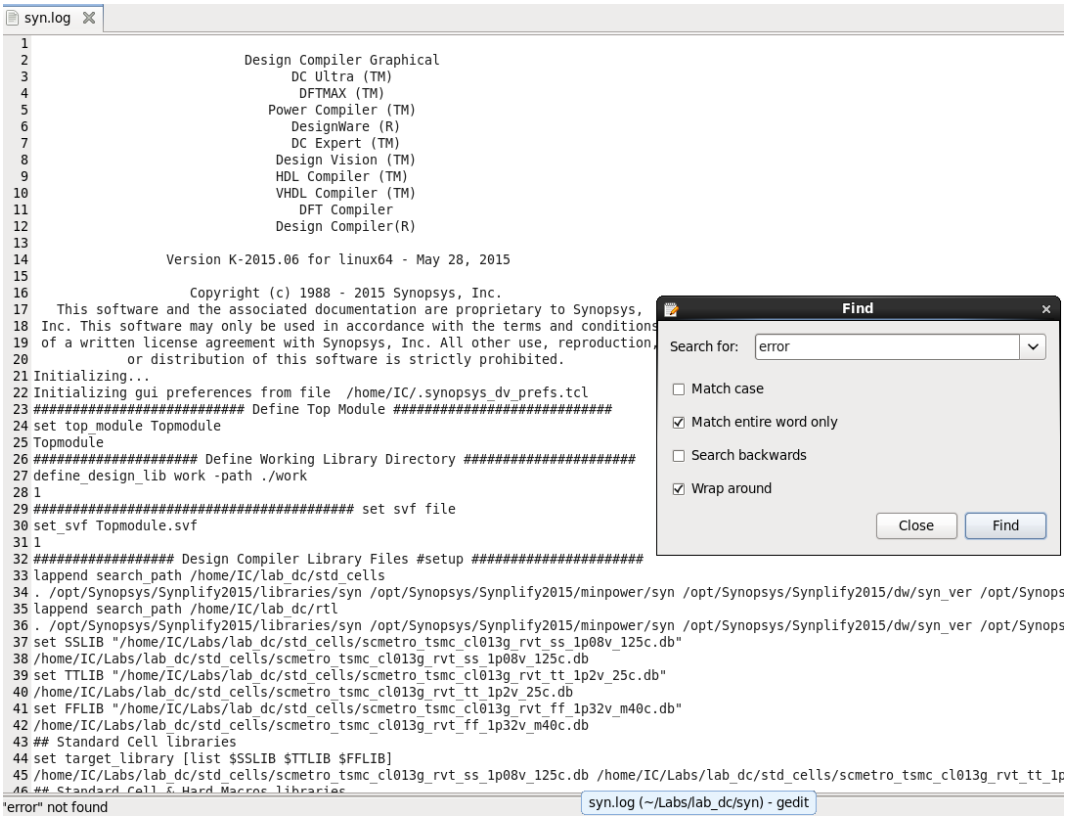


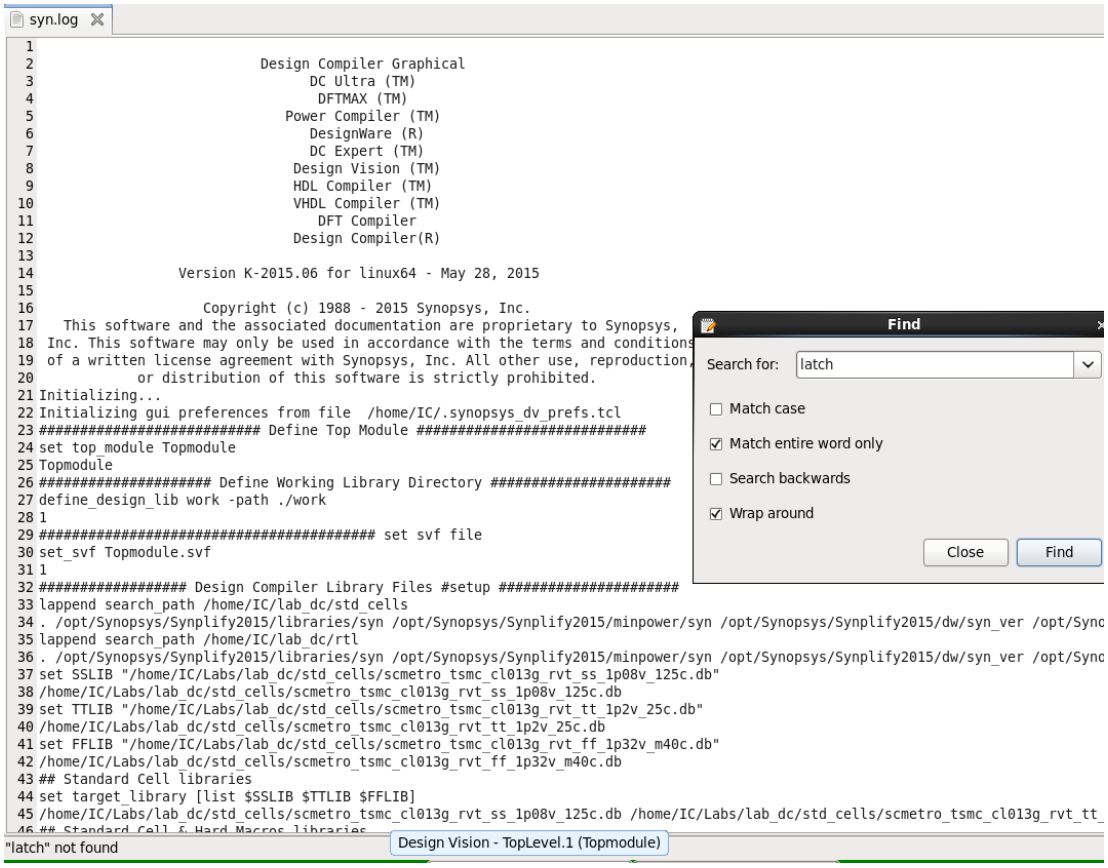
FSM



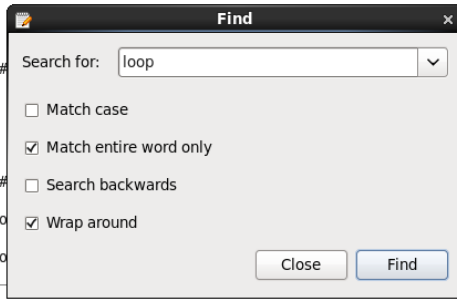
Counter

❖ no error found in log:





```
                  Copyright (c) 1988 - 2015 Synopsys, Inc.
   This software and the associated documentation are proprietary to Synopsys,
 Inc. This software may only be used in accordance with the terms and conditions
 of a written license agreement with Synopsys, Inc. All other use, reproduction,
            or distribution of this software is strictly prohibited.
Initializing...
Initializing gui preferences from file  /home/IC/.synopsys_dv_prefs.tcl
######################### Define Top Module ###########################
set top_module Topmodule
Topmodule
#################### Define Working Library Directory ####################
define_design_lib work -path ./work
1
######################################## set svf file
set_svf Topmodule.svf
1
################# Design Compiler Library Files #setup ##################
lappend search_path /home/IC/lab_dc/std_cells
. /opt/Synopsys/Synplify2015/libraries/syn /opt/Synopsys/Synplify2015/minpo
lappend search_path /home/IC/lab_dc/rtl
. /opt/Synopsys/Synplify2015/libraries/syn /opt/Synopsys/Synplify2015/minpo
set SSLIB "/home/IC/Labs/lab_dc/std_cells/scmetro_tsmc_cl013g_rvt_ss_1p08v_
/home/IC/Labs/lab_dc/std_cells/scmetro_tsmc_cl013g_rvt_ss_1p08v_125c.db
set TTLIB "/home/IC/Labs/lab_dc/std_cells/scmetro_tsmc_cl013g_rvt_tt_1p2v_25c.db"
/home/IC/Labs/lab_dc/std_cells/scmetro_tsmc_cl013g_rvt_tt_1p2v_25c.db
set FFLIB "/home/IC/Labs/lab_dc/std_cells/scmetro_tsmc_cl013g_rvt_ff_1p32v_m40c.db"
/home/IC/Labs/lab_dc/std_cells/scmetro_tsmc_cl013g_rvt_ff_1p32v_m40c.db
## Standard Cell libraries
set target_library [list $SSLIB $TTLIB $FFLIB]
/home/IC/Labs/lab_dc/std_cells/scmetro_tsmc_cl013g_rvt_ss_1p08v_125c.db /home/IC/Labs/lab_dc/std_cells/scmetro_tsmc_cl013g_rvt
## Standard Cell & Hard Macros libraries
"loop" not found
```



Design Vision - TopLevel.1 (Topmodule)

❖ constraint Report:

```
1
2 ****************************************
3 Report : constraint
4          -all_violators
5 Design : Topmodule
6 Version: K-2015.06
7 Date   : Sun Dec 15 22:58:57 2024
8 ****************************************
9
10 This design has no violated constraints.
11
12 1
```

❖ scripts files:

- **syn_script.tcl:**

```
2  ########################## Define Top Module ############################
3
4  set top_module Topmodule
5
6  ###################### Define Working Library Directory ######################
7
8  define_design_lib work -path ./work
9
10 ######################################### set svf file
11 set_svf Topmodule.svf
12 ################## Design Compiler Library Files #setup ####################
13
14 lappend search_path /home/IC/lab_dc/std_cells
15 lappend search_path /home/IC/lab_dc/rtl
16
17 set SSLIB "/home/IC/Labs/lab_dc/std_cells/scmetro_tsmc_cl013g_rvt_ss_1p08v_125c.db"
18 set TTLIB "/home/IC/Labs/lab_dc/std_cells/scmetro_tsmc_cl013g_rvt_tt_1p2v_25c.db"
19 set FFLIB "/home/IC/Labs/lab_dc/std_cells/Sscmetro_tsmc_cl013g_rvt_ff_1p32v_m40c.db"
20
21 ## Standard Cell libraries
22 set target_library [list $SSLIB $TTLIB $FFLIB]
23
24 ## Standard Cell & Hard Macros libraries
25 set link_library [list * $SSLIB $TTLIB $FFLIB]
26
27 #echo "################################################"
28 #echo "############# Reading RTL Files  ###############"
29 #echo "################################################"
30
31 #alu File
32 set file_format verilog
33 read_file -format $file_format Topmodule.v
34
35 ###################### Defining toplevel ###################################
36
37 current_design $top_module
38
39 ################### Liniking All The Design Parts ##########################
40 puts "################################################"
41 puts "######## Liniking All The Design Parts ########"
42 puts "################################################"
43
44 link
```

```
46 ################### Liniking All The Design Parts ##########################
47 puts "################################################"
48 puts "######## checking design consistency ##########"
49 puts "################################################"
50
51 check_design
52 ################################## Path groups ###################################
53 puts "################################################"
54 puts "################ Path groups ###################"
55 puts "################################################"
56
57 group_path -name INREG -from [all_inputs]
58 group_path -name REGOUT -to [all_outputs]
59 group_path -name INOUT -from [all_inputs] -to [all_outputs]
60
61 ###################### Define Design Constraints ##########################
62 puts "################################################"
63 puts "############ Design Constraints #### ##########"
64 puts "################################################"
65
66 source -echo ./cons.tcl
67 ###################### Mapping and optimization ##########################
68 puts "################################################"
69 puts "########## Mapping & Optimization #############"
70 puts "################################################"
71
72 compile -map_effort high
73 set_svf -off
74 ##############################################################################
75 # Write out Design after initial compile
76 ##############################################################################
77 write_file -format verilog -hierarchy -output netlists/Topmodule_netlist.v
78 write_file -format ddc     -hierarchy -output netlists/Topmodule_netlist.ddc
79 write_sdc -nosplit SDC/Topmodule.sdc
80 write_sdf SDF/Topmodule.sdf
81 ################## reporting ###########################
82 report_area -hierarchy > reports/area.rpt
83 report_power -hierarchy > reports/power.rpt
84 report_timing -max_paths 100 -delay_type max > reports/setup.rpt
85 report_timing -max_paths 100 -delay_type min > reports/hold.rpt
86 report_clock -attributes > reports/clocks.rpt
87 report_constraint -all_violators > reports/constraints.rpt
88 ################## starting graphical user interface ######################
89 gui_start
```

- **cons.tcl:**

```
25 ##################################################################################
26 # 1. Master Clock Definitions
27 # 2. Generated Clock Definitions
28 # 3. Clock Latencies
29 # 4. Clock Uncertainties
30 # 4. Clock Transitions
31 ##################################################################################
32 set CLK_NAME TRAFFIC_CONTROLLER_CLK
33 set CLK_PER 100
34 set CLK_SETUP_SKEW 0.25
35 set CLK_HOLD_SKEW 0.05
36 set CLK_LAT 0
37 set CLK_RISE 0.1
38 set CLK_FALL 0.1
39
40 create_clock -name $CLK_NAME -period $CLK_PER -waveform "0 [expr $CLK_PER/2]" [get_ports clk]
41 set_clock_uncertainty -setup $CLK_SETUP_SKEW [get_clocks $CLK_NAME]
42 set_clock_uncertainty -hold  $CLK_HOLD_SKEW  [get_clocks $CLK_NAME]
43 set_clock_transition  -rise  $CLK_RISE       [get_clocks $CLK_NAME]
44 set_clock_transition  -fall  $CLK_FALL       [get_clocks $CLK_NAME]
45 set_clock_latency $CLK_LAT [get_clocks $CLK_NAME]
46 set_dont_touch_network {clk rst_n}
47
48             ############################################################
49                    #### Section 2 : Clocks Relationships ####
50             ############################################################
51             ############################################################
52               #### Section 3 : set input/output delay on ports ####
53             ############################################################
54 set in_delay  [expr 0.3*$CLK_PER]
55 set out_delay [expr 0.3*$CLK_PER]
56 #Constrain Input Paths
57 set_input_delay $in_delay -clock $CLK_NAME [get_port Sa]
58 set_input_delay $in_delay -clock $CLK_NAME [get_port Sb]
59 set_input_delay $in_delay -clock $CLK_NAME [get_port Sc]
60 set_input_delay $in_delay -clock $CLK_NAME [get_port Sd]
61 set_input_delay $in_delay -clock $CLK_NAME [get_port rst_n]
62 #Constrain Output Paths
63 set_output_delay $out_delay -clock $CLK_NAME [get_port Ta]
64 set_output_delay $out_delay -clock $CLK_NAME [get_port Tb]
65 set_output_delay $out_delay -clock $CLK_NAME [get_port Tc]
66 set_output_delay $out_delay -clock $CLK_NAME [get_port Td]
67             ############################################################
68                    #### Section 4 : Driving cells ####
69             ############################################################
```

```
67           ########################################################
68                  #### Section 4 : Driving cells ####
69           ########################################################
70 set_driving_cell -library scmetro_tsmc_cl013g_rvt_ss_1p08v_125c -lib_cell BUFX2M -pin Y [get_port Sa]
71 set_driving_cell -library scmetro_tsmc_cl013g_rvt_ss_1p08v_125c -lib_cell BUFX2M -pin Y [get_port Sb]
72 set_driving_cell -library scmetro_tsmc_cl013g_rvt_ss_1p08v_125c -lib_cell BUFX2M -pin Y [get_port Sc]
73 set_driving_cell -library scmetro_tsmc_cl013g_rvt_ss_1p08v_125c -lib_cell BUFX2M -pin Y [get_port Sd]
74 set_driving_cell -library scmetro_tsmc_cl013g_rvt_ss_1p08v_125c -lib_cell BUFX2M -pin Y [get_port rst_n]
75           ########################################################
76                  #### Section 5 : Output load ####
77           ########################################################
78 set_load 0.5 [get_port Ta]
79 set_load 0.5 [get_port Tb]
80 set_load 0.5 [get_port Tc]
81 set_load 0.5 [get_port Td]
82           ########################################################
83                ### Section 6 : Operating Condition ####
84           ########################################################
85
86 # Define the Worst Library for Max(#setup) analysis
87 # Define the Best Library for Min(hold) analysis
88 set_operating_condition -max_library "scmetro_tsmc_cl013g_rvt_ss_1p08v_125c" -max "scmetro_tsmc_cl013g_rvt_ss_1p08v_125c" \
89                         -min_library "scmetro_tsmc_cl013g_rvt_ff_1p32v_m40c" -min "scmetro_tsmc_cl013g_rvt_ff_1p32v_m40c"
90
91 ##################################################################################
92           ########################################################
93                  #### Section 7 : wireload Model ####
94           ########################################################
95 ##################################################################################
96 set_wire_load_model -name tsmc13_wl10 -library scmetro_tsmc_cl013g_rvt_ss_1p08v_125c
```

- **Topmodule.sdc:**

```
#####################################################################

# Created by write_sdc on Mon Dec 16 03:03:19 2024

#####################################################################
set sdc_version 2.0

set_units -time ns -resistance kOhm -capacitance pF -voltage V -current mA
set_operating_conditions -max scmetro_tsmc_cl013g_rvt_ss_1p08v_125c -max_library scmetro_tsmc_cl013g_rvt_ss_1p08v_125c
 -min scmetro_tsmc_cl013g_rvt_ff_1p32v_m40c -min_library scmetro_tsmc_cl013g_rvt_ff_1p32v_m40c      You, 14 hours ago
set_wire_load_model -name tsmc13_wl10 -library scmetro_tsmc_cl013g_rvt_ss_1p08v_125c
set_driving_cell -lib_cell BUFX2M -library scmetro_tsmc_cl013g_rvt_ss_1p08v_125c -pin Y [get_ports rst_n]
set_driving_cell -lib_cell BUFX2M -library scmetro_tsmc_cl013g_rvt_ss_1p08v_125c -pin Y [get_ports {Sa[1]}]
set_driving_cell -lib_cell BUFX2M -library scmetro_tsmc_cl013g_rvt_ss_1p08v_125c -pin Y [get_ports {Sa[0]}]
set_driving_cell -lib_cell BUFX2M -library scmetro_tsmc_cl013g_rvt_ss_1p08v_125c -pin Y [get_ports {Sb[1]}]
set_driving_cell -lib_cell BUFX2M -library scmetro_tsmc_cl013g_rvt_ss_1p08v_125c -pin Y [get_ports {Sb[0]}]
set_driving_cell -lib_cell BUFX2M -library scmetro_tsmc_cl013g_rvt_ss_1p08v_125c -pin Y [get_ports {Sc[1]}]
set_driving_cell -lib_cell BUFX2M -library scmetro_tsmc_cl013g_rvt_ss_1p08v_125c -pin Y [get_ports {Sc[0]}]
set_driving_cell -lib_cell BUFX2M -library scmetro_tsmc_cl013g_rvt_ss_1p08v_125c -pin Y [get_ports {Sd[1]}]
set_driving_cell -lib_cell BUFX2M -library scmetro_tsmc_cl013g_rvt_ss_1p08v_125c -pin Y [get_ports {Sd[0]}]
set_load -pin_load 0.5 [get_ports {Ta[2]}]
set_load -pin_load 0.5 [get_ports {Ta[1]}]
set_load -pin_load 0.5 [get_ports {Ta[0]}]
set_load -pin_load 0.5 [get_ports {Tb[2]}]
set_load -pin_load 0.5 [get_ports {Tb[1]}]
set_load -pin_load 0.5 [get_ports {Tb[0]}]
set_load -pin_load 0.5 [get_ports {Tc[2]}]
set_load -pin_load 0.5 [get_ports {Tc[1]}]
set_load -pin_load 0.5 [get_ports {Tc[0]}]
set_load -pin_load 0.5 [get_ports {Td[2]}]
set_load -pin_load 0.5 [get_ports {Td[1]}]
set_load -pin_load 0.5 [get_ports {Td[0]}]
create_clock [get_ports clk]  -name TRAFFIC_CONTROLLER_CLK  -period 100  -waveform {0 50}
set_clock_latency 0  [get_clocks TRAFFIC_CONTROLLER_CLK]
set_clock_uncertainty -setup 0.25  [get_clocks TRAFFIC_CONTROLLER_CLK]
set_clock_uncertainty -hold 0.05  [get_clocks TRAFFIC_CONTROLLER_CLK]
```

```
set_clock_transition -max -rise 0.1 [get_clocks TRAFFIC_CONTROLLER_CLK]
set_clock_transition -min -rise 0.1 [get_clocks TRAFFIC_CONTROLLER_CLK]
set_clock_transition -max -fall 0.1 [get_clocks TRAFFIC_CONTROLLER_CLK]
set_clock_transition -min -fall 0.1 [get_clocks TRAFFIC_CONTROLLER_CLK]
group_path -name INOUT  -from [list [get_ports clk] [get_ports rst_n] [get_ports {Sa[1]}] [get_ports {Sa[0]}] [get_ports {Sb[1]}] [get_ports {Sb[0]}] [g
group_path -name INREG  -from [list [get_ports clk] [get_ports rst_n] [get_ports {Sa[1]}] [get_ports {Sa[0]}] [get_ports {Sb[1]}] [get_ports {Sb[0]}] [g
group_path -name REGOUT  -to [list [get_ports {Ta[2]}] [get_ports {Ta[1]}] [get_ports {Ta[0]}] [get_ports {Tb[2]}] [get_ports {Tb[1]}] [get_ports {Tb[0]
set_input_delay -clock TRAFFIC_CONTROLLER_CLK  30  [get_ports rst_n]
set_input_delay -clock TRAFFIC_CONTROLLER_CLK  30  [get_ports {Sa[1]}]
set_input_delay -clock TRAFFIC_CONTROLLER_CLK  30  [get_ports {Sa[0]}]
set_input_delay -clock TRAFFIC_CONTROLLER_CLK  30  [get_ports {Sb[1]}]
set_input_delay -clock TRAFFIC_CONTROLLER_CLK  30  [get_ports {Sb[0]}]
set_input_delay -clock TRAFFIC_CONTROLLER_CLK  30  [get_ports {Sc[1]}]
set_input_delay -clock TRAFFIC_CONTROLLER_CLK  30  [get_ports {Sc[0]}]
set_input_delay -clock TRAFFIC_CONTROLLER_CLK  30  [get_ports {Sd[1]}]
set_input_delay -clock TRAFFIC_CONTROLLER_CLK  30  [get_ports {Sd[0]}]
set_output_delay -clock TRAFFIC_CONTROLLER_CLK  30  [get_ports {Ta[2]}]
set_output_delay -clock TRAFFIC_CONTROLLER_CLK  30  [get_ports {Ta[1]}]
set_output_delay -clock TRAFFIC_CONTROLLER_CLK  30  [get_ports {Ta[0]}]
set_output_delay -clock TRAFFIC_CONTROLLER_CLK  30  [get_ports {Tb[2]}]
set_output_delay -clock TRAFFIC_CONTROLLER_CLK  30  [get_ports {Tb[1]}]
set_output_delay -clock TRAFFIC_CONTROLLER_CLK  30  [get_ports {Tb[0]}]
set_output_delay -clock TRAFFIC_CONTROLLER_CLK  30  [get_ports {Tc[2]}]
set_output_delay -clock TRAFFIC_CONTROLLER_CLK  30  [get_ports {Tc[1]}]
set_output_delay -clock TRAFFIC_CONTROLLER_CLK  30  [get_ports {Tc[0]}]
set_output_delay -clock TRAFFIC_CONTROLLER_CLK  30  [get_ports {Td[2]}]
set_output_delay -clock TRAFFIC_CONTROLLER_CLK  30  [get_ports {Td[1]}]
set_output_delay -clock TRAFFIC_CONTROLLER_CLK  30  [get_ports {Td[0]}]

```

❖ Netlist:

```
module Topmodule ( clk, rst_n, Sa, Sb, Sc, Sd, Ta, Tb, Tc, Td );
  input [1:0] Sa;
  input [1:0] Sb;
  input [1:0] Sc;
  input [1:0] Sd;
  output [2:0] Ta;
  output [2:0] Tb;
  output [2:0] Tc;
  output [2:0] Td;
  input clk, rst_n;
  wire   load_en;
  wire   [1:0] Sa_reading;
  wire   [1:0] Sb_reading;
  wire   [1:0] Sc_reading;
  wire   [1:0] Sd_reading;
  wire   [4:0] counter_output;
  wire   [4:0] load_value;
  wire   SYNOPSYS_UNCONNECTED__0;

  Traffic_Controller FSM ( .Sa(Sa_reading), .Sb(Sb_reading), .Sc(Sc_reading),
        .Sd(Sd_reading), .clk(clk), .rst_n(rst_n), .counter_value(
        counter_output), .Ta(Ta), .Tb(Tb), .Tc(Tc), .Td(Td), .load_counter(
        load_en), .load_value({load_value[4:2], SYNOPSYS_UNCONNECTED__0,
        load_value[0]}) );
  counter internal_counter ( .clk(clk), .rst_n(rst_n), .load(load_en), .data({
        load_value[4:2], 1'b1, load_value[0]}), .count(counter_output) );
  CLKXOR2X2M U9 ( .A(Sb[1]), .B(Sb[0]), .Y(Sb_reading[0]) );
  CLKXOR2X2M U10 ( .A(Sc[1]), .B(Sc[0]), .Y(Sc_reading[0]) );
  CLKXOR2X2M U11 ( .A(Sa[1]), .B(Sa[0]), .Y(Sa_reading[0]) );
  CLKXOR2X2M U12 ( .A(Sd[1]), .B(Sd[0]), .Y(Sd_reading[0]) );
  AND2X2M U13 ( .A(Sd[0]), .B(Sd[1]), .Y(Sd_reading[1]) );
  AND2X2M U14 ( .A(Sc[0]), .B(Sc[1]), .Y(Sc_reading[1]) );
  AND2X2M U15 ( .A(Sb[0]), .B(Sb[1]), .Y(Sb_reading[1]) );
  AND2X2M U16 ( .A(Sa[0]), .B(Sa[1]), .Y(Sa_reading[1]) );
endmodule
```

```
module counter ( clk, rst_n, load, data, count );
  input [4:0] data;
  output [4:0] count;
  input clk, rst_n, load;
  wire   N5, N6, N7, N8, N9, N10, N11, N12, N13, N14, n1, n2, n3, n4, n5;

  DFFSQX2M \count_reg[1]  ( .D(N11), .CK(clk), .SN(rst_n), .Q(count[1]) );
  DFFSQX2M \count_reg[4]  ( .D(N14), .CK(clk), .SN(rst_n), .Q(count[4]) );
  DFFSQX2M \count_reg[2]  ( .D(N12), .CK(clk), .SN(rst_n), .Q(count[2]) );
  DFFSQX2M \count_reg[3]  ( .D(N13), .CK(clk), .SN(rst_n), .Q(count[3]) );
  DFFSQX2M \count_reg[0]  ( .D(N10), .CK(clk), .SN(rst_n), .Q(count[0]) );
  INVX2M U3 ( .A(load), .Y(n5) );
  AO22X1M U4 ( .A0(data[2]), .A1(load), .B0(N7), .B1(n5), .Y(N12) );
  AO22X1M U5 ( .A0(data[0]), .A1(load), .B0(N5), .B1(n5), .Y(N10) );
  INVX2M U6 ( .A(count[0]), .Y(N5) );
  AO22X1M U7 ( .A0(load), .A1(data[4]), .B0(N9), .B1(n5), .Y(N14) );
  AO22X1M U8 ( .A0(data[3]), .A1(load), .B0(N8), .B1(n5), .Y(N13) );
  AO22X1M U9 ( .A0(data[1]), .A1(load), .B0(N6), .B1(n5), .Y(N11) );
  INVX2M U10 ( .A(count[2]), .Y(n4) );
  NOR2X1M U11 ( .A(count[1]), .B(count[0]), .Y(n1) );
  AO21XLM U12 ( .A0(count[0]), .A1(count[1]), .B0(n1), .Y(N6) );
  CLKNAND2X2M U13 ( .A(n1), .B(n4), .Y(n2) );
  OAI21X1M U14 ( .A0(n1), .A1(n4), .B0(n2), .Y(N7) );
  XNOR2X1M U15 ( .A(count[3]), .B(n2), .Y(N8) );
  NOR2X1M U16 ( .A(count[3]), .B(n2), .Y(n3) );
  CLKXOR2X2M U17 ( .A(count[4]), .B(n3), .Y(N9) );
endmodule
```

```
module Traffic_Controller ( Sa, Sb, Sc, Sd, clk, rst_n, counter_value, Ta, Tb,
        Tc, Td, load_counter, load_value );
  input [1:0] Sa;
  input [1:0] Sb;
  input [1:0] Sc;
  input [1:0] Sd;
  input [4:0] counter_value;
  output [2:0] Ta;
  output [2:0] Tb;
  output [2:0] Tc;
  output [2:0] Td;
  output [4:0] load_value;
  input clk, rst_n;
  output load_counter;
  wire   n33, n34, n35, n36, n37, n38, n39, n40, N158, n94, n95, n96, n97, n98,
         n99, n100, n101, n102, n103, n104, n105, n106, n107, n108, n109, n110,
         n111, n112, n113, n114, n115, n116, n117, n118, n119, n120, n121,
         n122, n123, n124, n125, n126, n127, n128, n129, n130, n131, n132,
         n133, n134, n135, n136, n137, n138, n139, n140, n141, n142, n143,
         n144, n1, \load_value[4] , n12, n13, n14, n15, n16, n17, n18, n19,
         n20, n21, n22, n23, n24, n25, n26, n27, n28, n29, n30, n31, n32;
  wire   [2:0] current_state;
  wire   [2:0] next_state;
  assign load_value[1] = 1'b1;
  assign load_value[0] = N158;
  assign load_value[2] = \load_value[4] ;
  assign load_value[3] = \load_value[4] ;
  assign load_value[4] = \load_value[4] ;

  NOR3BX4M U18 ( .AN(n117), .B(n118), .C(n119), .Y(n99) );
  DFFRX1M \current_state_reg[1]  ( .D(next_state[1]), .CK(clk), .RN(rst_n),
      .Q(current_state[1]), .QN(n21) );
  DFFRX1M \current_state_reg[0]  ( .D(next_state[0]), .CK(clk), .RN(rst_n),
      .Q(current_state[0]), .QN(n15) );
  DFFRX1M \current_state_reg[2]  ( .D(N158), .CK(clk), .RN(rst_n), .Q(
      current_state[2]), .QN(n20) );
  OR3X2M U3 ( .A(current_state[1]), .B(current_state[2]), .C(n15), .Y(n1) );
  NAND3X12M U4 ( .A(n1), .B(n13), .C(n125), .Y(Ta[2]) );
  NAND3X12M U5 ( .A(n19), .B(n17), .C(n125), .Y(Tb[2]) );
  NOR3BX4M U6 ( .AN(n100), .B(Td[0]), .C(n38), .Y(n125) );
  NAND3X12M U7 ( .A(n16), .B(n18), .C(n123), .Y(Td[2]) );
  CLKINVX8M U8 ( .A(n16), .Y(Tc[0]) );
  NOR3X2M U9 ( .A(current_state[0]), .B(current_state[2]), .C(n21), .Y(n38) );
  INVX2M U10 ( .A(n38), .Y(n16) );
  CLKINVX8M U11 ( .A(n13), .Y(Tb[1]) );
  NOR3X2M U12 ( .A(n15), .B(current_state[1]), .C(n20), .Y(n35) );
  INVX2M U13 ( .A(n35), .Y(n13) );
  CLKBUFX8M U14 ( .A(n36), .Y(Tc[2]) );
  AND3X1M U15 ( .A(n124), .B(n1), .C(n17), .Y(n123) );
```

```
56        AND3X1M U15 ( .A(n124), .B(n1), .C(n17), .Y(n123) );
57        NAND3BX2M U16 ( .AN(Td[0]), .B(n14), .C(n123), .Y(n36) );
58        CLKINVX8M U17 ( .A(n1), .Y(Tb[0]) );
59        CLKINVX8M U19 ( .A(n18), .Y(Tc[1]) );
60        NOR3X2M U20 ( .A(n20), .B(current_state[0]), .C(n21), .Y(n37) );
61        INVX2M U21 ( .A(n37), .Y(n18) );
62        CLKINVX8M U22 ( .A(n19), .Y(Ta[1]) );
63        INVX2M U23 ( .A(n33), .Y(n19) );
64        NOR3X2M U24 ( .A(current_state[0]), .B(current_state[1]), .C(n20), .Y(n33)
65              );
66        OR3X2M U25 ( .A(n15), .B(current_state[2]), .C(n21), .Y(n40) );
67        INVX8M U26 ( .A(n40), .Y(Td[0]) );
68        CLKINVX8M U27 ( .A(n17), .Y(Ta[0]) );
69        NOR3X2M U28 ( .A(current_state[1]), .B(current_state[2]), .C(
70              current_state[0]), .Y(n34) );
71        INVX2M U29 ( .A(n34), .Y(n17) );
72        CLKINVX8M U30 ( .A(n14), .Y(Td[1]) );
73        NOR3X2M U31 ( .A(n20), .B(n15), .C(n21), .Y(n39) );
74        INVX2M U32 ( .A(n39), .Y(n14) );
75        NOR2XLM U33 ( .A(n33), .B(n35), .Y(n124) );
76        NOR2X1M U34 ( .A(n37), .B(n39), .Y(n100) );
77        OAI221X1M U35 ( .A0(n16), .A1(n126), .B0(n1), .B1(n127), .C0(n128), .Y(N158)
78              );
79        INVX2M U36 ( .A(n108), .Y(n23) );
80        OAI32X1M U37 ( .A0(n14), .A1(n99), .A2(n23), .B0(n12), .B1(n100), .Y(n98) );
81        OAI221X1M U38 ( .A0(n23), .A1(n18), .B0(n19), .B1(n22), .C0(n94), .Y(n106)
82              );
83        INVX2M U39 ( .A(n99), .Y(n22) );
84        INVX2M U40 ( .A(N158), .Y(\load_value[4] ) );
85        NAND3BX2M U41 ( .AN(n102), .B(n110), .C(n12), .Y(n95) );
86        OAI22X1M U42 ( .A0(n99), .A1(n19), .B0(n13), .B1(n107), .Y(n110) );
87        NOR3BX2M U43 ( .AN(n114), .B(n115), .C(n116), .Y(n102) );
88        NAND3X2M U44 ( .A(n111), .B(n112), .C(n113), .Y(n107) );
89        NAND3BX2M U45 ( .AN(n120), .B(n121), .C(n122), .Y(n108) );
90        AOI21X2M U46 ( .A0(n100), .A1(n124), .B0(n12), .Y(n131) );
91        OAI32X2M U47 ( .A0(n24), .A1(Sb[0]), .A2(n133), .B0(Sb[1]), .B1(n26), .Y(
92              n119) );
93        OAI32X2M U48 ( .A0(n24), .A1(Sc[0]), .A2(n134), .B0(Sc[1]), .B1(n26), .Y(
94              n115) );
95        OAI31X1M U49 ( .A0(n112), .A1(n121), .A2(n117), .B0(n12), .Y(n126) );
96        OAI31X1M U50 ( .A0(n111), .A1(n122), .A2(n114), .B0(n12), .Y(n127) );
97        AOI221XLM U51 ( .A0(n129), .A1(Td[0]), .B0(n130), .B1(n34), .C0(n131), .Y(
98              n128) );
99        AOI32X1M U52 ( .A0(Sa[0]), .A1(n31), .A2(n132), .B0(n32), .B1(Sa[1]), .Y(
100             n113) );
101       AOI32X1M U53 ( .A0(Sc[0]), .A1(n31), .A2(n136), .B0(n32), .B1(Sc[1]), .Y(
102             n112) );
```

```
103       AOI32X1M U54 ( .A0(Sb[0]), .A1(n31), .A2(n139), .B0(n32), .B1(Sb[1]), .Y(
104             n111) );
105       AOI32X1M U55 ( .A0(Sb[0]), .A1(n29), .A2(n140), .B0(n30), .B1(Sb[1]), .Y(
106             n114) );
107       AOI2BB1X2M U56 ( .A0N(n141), .A1N(n27), .B0(n133), .Y(n122) );
108       OAI21X2M U57 ( .A0(Sb[1]), .A1(n26), .B0(n24), .Y(n141) );
109       AOI2BB1X2M U58 ( .A0N(n144), .A1N(n29), .B0(n134), .Y(n121) );
110       OAI21X2M U59 ( .A0(Sc[1]), .A1(n26), .B0(n24), .Y(n144) );
111       AOI31X2M U60 ( .A0(n115), .A1(n119), .A2(n25), .B0(n109), .Y(n130) );
112       INVX2M U61 ( .A(n113), .Y(n25) );
113       AOI31X2M U62 ( .A0(n118), .A1(n120), .A2(n116), .B0(n109), .Y(n129) );
114       INVX2M U63 ( .A(Sa[1]), .Y(n26) );
115       OAI21X2M U64 ( .A0(n31), .A1(n135), .B0(n136), .Y(n116) );
116       OAI21X2M U65 ( .A0(Sd[1]), .A1(n30), .B0(n29), .Y(n135) );
117       OAI21X2M U66 ( .A0(n31), .A1(n137), .B0(n132), .Y(n120) );
118       OAI21X2M U67 ( .A0(Sd[1]), .A1(n26), .B0(n24), .Y(n137) );
119       INVX2M U68 ( .A(Sa[0]), .Y(n24) );
120       INVX2M U69 ( .A(Sb[1]), .Y(n28) );
121       NOR2X2M U70 ( .A(n28), .B(Sa[1]), .Y(n133) );
122       NOR2X2M U71 ( .A(n30), .B(Sa[1]), .Y(n134) );
123       OAI21X2M U72 ( .A0(n31), .A1(n138), .B0(n139), .Y(n118) );
124       OAI21X2M U73 ( .A0(Sd[1]), .A1(n28), .B0(n27), .Y(n138) );
125       NAND2X2M U74 ( .A(Sd[1]), .B(n26), .Y(n132) );
126       NAND2X2M U75 ( .A(Sd[1]), .B(n28), .Y(n139) );
127       NAND2X2M U76 ( .A(Sd[1]), .B(n30), .Y(n136) );
128       INVX2M U77 ( .A(Sb[0]), .Y(n27) );
129       INVX2M U78 ( .A(Sc[0]), .Y(n29) );
130       INVX2M U79 ( .A(Sc[1]), .Y(n30) );
131       NAND2X2M U80 ( .A(Sc[1]), .B(n28), .Y(n140) );
132       INVX2M U81 ( .A(Sd[1]), .Y(n32) );
133       OA21X2M U82 ( .A0(n29), .A1(n143), .B0(n140), .Y(n117) );
134       OAI21X2M U83 ( .A0(Sc[1]), .A1(n28), .B0(n27), .Y(n143) );
135       NAND2BXLM U84 ( .AN(n107), .B(Tc[1]), .Y(n94) );
136       INVX2M U85 ( .A(n109), .Y(n12) );
137       XNOR2X4M U86 ( .A(\load_value[4] ), .B(current_state[2]), .Y(load_counter) );
138       INVX2M U87 ( .A(Sd[0]), .Y(n31) );
139       NAND4X2M U88 ( .A(n103), .B(n95), .C(n104), .D(n105), .Y(next_state[0]) );
140       OAI21XLM U89 ( .A0(Td[1]), .A1(Tb[1]), .B0(n109), .Y(n104) );
141       NAND3XLM U90 ( .A(Td[1]), .B(n108), .C(n99), .Y(n103) );
142       AOI22X1M U91 ( .A0(n12), .A1(n106), .B0(current_state[0]), .B1(n20), .Y(n105) );
143       NAND4X2M U92 ( .A(n94), .B(n95), .C(n96), .D(n97), .Y(next_state[1]) );
144       NAND3X2M U93 ( .A(n12), .B(n101), .C(n102), .Y(n96) );
145       AOI21X2M U94 ( .A0(current_state[1]), .A1(n20), .B0(n98), .Y(n97) );
146       OAI21XLM U95 ( .A0(n99), .A1(n19), .B0(n13), .Y(n101) );
147       NAND3BX2M U96 ( .AN(counter_value[1]), .B(counter_value[0]), .C(n142), .Y(n109) );
148       NOR3X2M U97 ( .A(counter_value[2]), .B(counter_value[4]), .C(counter_value[3]), .Y(n142) )
149     endmodule
```

❖ Formal Verification:

- **Script:**

syn_fm_script.tcl

```
############################  Search PATH ################################
set PROJECT_PATH /home/IC/Labs

lappend search_path $PROJECT_PATH/lab_dc/std_cells
lappend search_path $PROJECT_PATH/lab_dc/rtl
lappend search_path $PROJECT_PATH/lab_dc/syn
lappend search_path $PROJECT_PATH/lab_dc/syn/netlists

########################## Define Top Module ############################

set top_module Topmodule

######################### Formality Setup File ###########################

set synopsys_auto_setup true

set_svf "../syn/$top_module.svf"

######################## Read Reference tech libs ########################


set SSLIB "/home/IC/Labs/lab_dc/std_cells/scmetro_tsmc_cl013g_rvt_ss_1p08v_125c.db"
set TTLIB "/home/IC/Labs/lab_dc/std_cells/scmetro_tsmc_cl013g_rvt_tt_1p2v_25c.db"
set FFLIB "/home/IC/Labs/lab_dc/std_cells/scmetro_tsmc_cl013g_rvt_ff_1p32v_m40c.db"

read_db -container Ref [list $SSLIB $TTLIB $FFLIB]

###################  Read Reference Design Files ########################
read_verilog -container Ref "Topmodule.v"

######################### set the top Reference Design ###################

set_reference_design Topmodule
set_top Topmodule

######################## Read Implementation tech libs ###################

read_db -container Imp [list $SSLIB $TTLIB $FFLIB]

#################### Read Implementation Design Files ####################

read_verilog -container Imp -netlist "Topmodule_netlist.v"

```

```

###################  set the top Implementation Design ###################

set_implementation_design Topmodule
set_top Topmodule


## matching Compare points
if {[match]} {
        echo "Matching Succeed"
}

## verify
set successful [verify]
if {!$successful} {
diagnose
analyze_points -failing
}

report_passing_points > "reports/passing_points.rpt"
report_failing_points > "reports/failing_points.rpt"
report_aborted_points > "reports/aborted_points.rpt"
report_unverified_points > "reports/unverified_points.rpt"


start_gui
```

- **Reports:**

failing_points.rpt | *passing_points.rpt | unverified_points.rpt

```
1 ************************************************
2 Report         : unverified_points
3
4 Reference      : Ref:/WORK/Topmodule
5 Implementation : Imp:/WORK/Topmodule
6 Version        : L-2016.03-SP1
7 Date           : Mon Dec 16 02:11:40 2024
8 ************************************************
9
10 No unverified compare points.
11
12 1
```

Unverified points

failing_points.rpt

```
1 ************************************************
2 Report         : failing_points
3
4 Reference      : Ref:/WORK/Topmodule
5 Implementation : Imp:/WORK/Topmodule
6 Version        : L-2016.03-SP1
7 Date           : Mon Dec 16 02:11:40 2024
8 ************************************************
9
10 No failing compare points.
11
12 1
```

Failing points

aborted_points.rpt

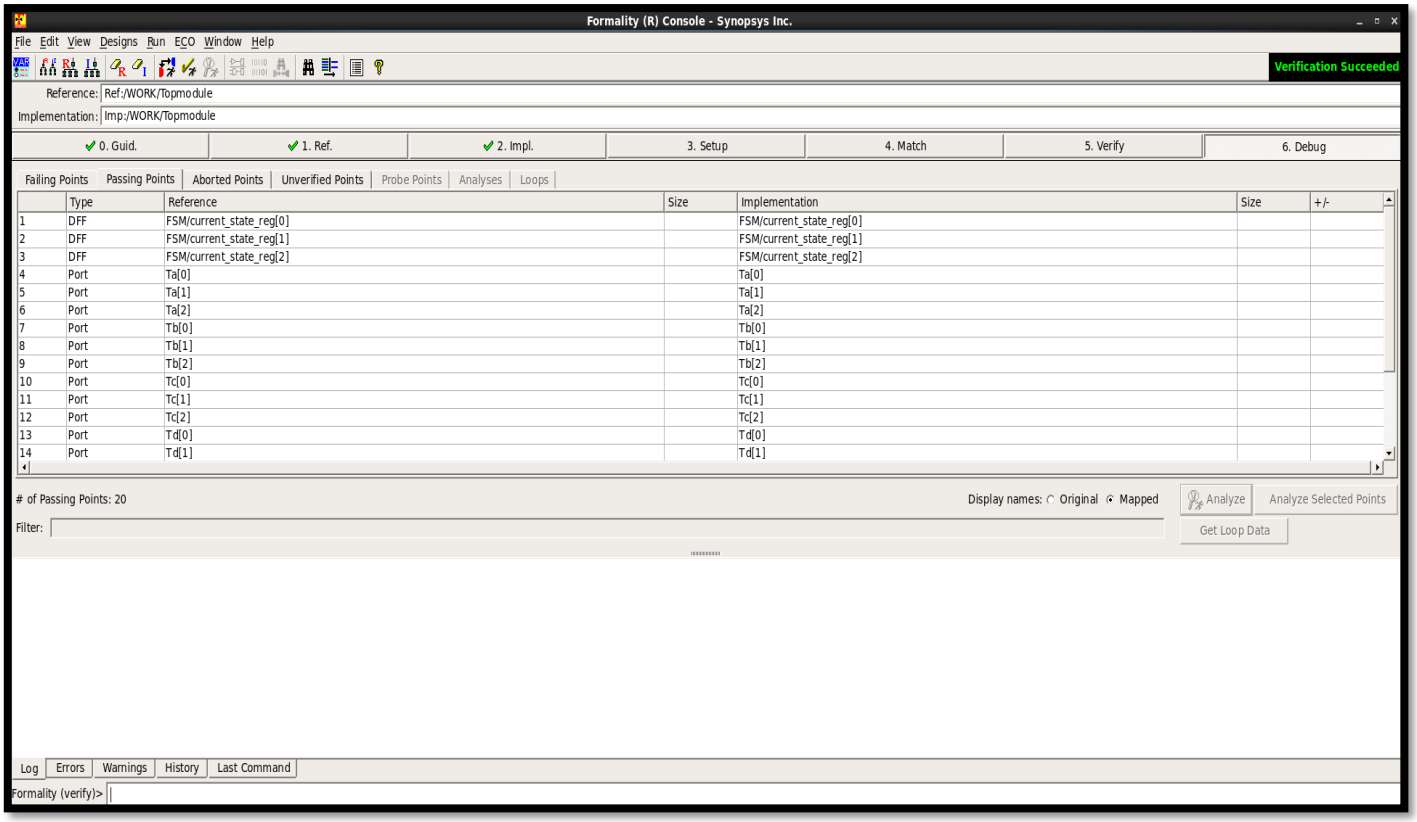
```
1 ************************************************
2 Report         : aborted_points
3
4 Reference      : Ref:/WORK/Topmodule
5 Implementation : Imp:/WORK/Topmodule
6 Version        : L-2016.03-SP1
7 Date           : Mon Dec 16 02:11:40 2024
8 ************************************************
9
10 No aborted compare points.
11
12 1
```

Abroted points

failing_points.rpt | *passing_points.rpt

```
1 S************************************************
2 Report         : passing_points
3
4 Reference      : Ref:/WORK/Topmodule
5 Implementation : Imp:/WORK/Topmodule
6 Version        : L-2016.03-SP1
7 Date           : Mon Dec 16 02:11:40 2024
8 ************************************************
9
10 20 Passing compare points:
11
12  Ref  DFF        Ref:/WORK/Topmodule/FSM/current_state_reg[0]
13  Impl DFF        Imp:/WORK/Topmodule/FSM/current_state_reg[0]
14
15  Ref  DFF        Ref:/WORK/Topmodule/FSM/current_state_reg[1]
16  Impl DFF        Imp:/WORK/Topmodule/FSM/current_state_reg[1]
17
18  Ref  DFF        Ref:/WORK/Topmodule/FSM/current_state_reg[2]
19  Impl DFF        Imp:/WORK/Topmodule/FSM/current_state_reg[2]
20
21  Ref  DFF        Ref:/WORK/Topmodule/internal_counter/count_reg[0]
22  Impl DFF        Imp:/WORK/Topmodule/internal_counter/count_reg[0]
23
24  Ref  DFF        Ref:/WORK/Topmodule/internal_counter/count_reg[1]
25  Impl DFF        Imp:/WORK/Topmodule/internal_counter/count_reg[1]
26
27  Ref  DFF        Ref:/WORK/Topmodule/internal_counter/count_reg[2]
28  Impl DFF        Imp:/WORK/Topmodule/internal_counter/count_reg[2]
29
30  Ref  DFF        Ref:/WORK/Topmodule/internal_counter/count_reg[3]
31  Impl DFF        Imp:/WORK/Topmodule/internal_counter/count_reg[3]
32
33  Ref  DFF        Ref:/WORK/Topmodule/internal_counter/count_reg[4]
34  Impl DFF        Imp:/WORK/Topmodule/internal_counter/count_reg[4]
35
36  Ref  Port       Ref:/WORK/Topmodule/Ta[0]
37  Impl Port       Imp:/WORK/Topmodule/Ta[0]
38
39  Ref  Port       Ref:/WORK/Topmodule/Ta[1]
40  Impl Port       Imp:/WORK/Topmodule/Ta[1]
41
42  Ref  Port       Ref:/WORK/Topmodule/Ta[2]
```

Passing points


Formality (R) Console - Synopsys Inc.
File Edit View Designs Run ECO Window Help
Verification Succeeded
Reference: Ref:/WORK/Topmodule
Implementation: Imp:/WORK/Topmodule
0. Guid.
1. Ref.
2. Impl.
3. Setup
4. Match
5. Verify
6. Debug
Failing Points
Passing Points
Aborted Points
Unverified Points
Probe Points
Analyses
Loops
Type
Reference
Size
Implementation
Size
+/-
1 DFF FSM/current_state_reg[0] FSM/current_state_reg[0]
2 DFF FSM/current_state_reg[1] FSM/current_state_reg[1]
3 DFF FSM/current_state_reg[2] FSM/current_state_reg[2]
4 Port Ta[0] Ta[0]
5 Port Ta[1] Ta[1]
6 Port Ta[2] Ta[2]
7 Port Tb[0] Tb[0]
8 Port Tb[1] Tb[1]
9 Port Tb[2] Tb[2]
10 Port Tc[0] Tc[0]
11 Port Tc[1] Tc[1]
12 Port Tc[2] Tc[2]
13 Port Td[0] Td[0]
14 Port Td[1] Td[1]
# of Passing Points: 20
Display names: Original Mapped
Analyze
Analyze Selected Points
Filter:
Get Loop Data
Log
Errors
Warnings
History
Last Command
Formality (verify)>