

Project 1

DSP48A1

Youssef Sameh Fawzy

1 – RTL Code:

I first observed that there is a repeated block of a register and a mux.

Reg&Mux module:

```
1  module Reg_Mux#(
2      parameter DATAWIDTH = 18,
3      parameter RSTTYPE = "SYNC",
4      parameter PIPLINE = 0
5  )(
6      input [DATAWIDTH-1:0] in1,
7      input CLK, rst, clkenable,
8      output reg [DATAWIDTH-1:0] out
9  );
10
11 generate
12     if (PIPLINE == 1) begin
13         if (RSTTYPE == "SYNC") begin : sync_reset
14             always @(posedge CLK) begin
15                 if (rst)
16                     out <= 0;
17                 else if(clkenable)
18                     out <= in1;
19             end
20         end else if (RSTTYPE == "ASYNC") begin : async_reset
21             always @(posedge CLK or posedge rst) begin
22                 if (rst)
23                     out <= 0;
24                 else if (clkenable)
25                     out <= in1;
26             end
27         end
28     end begin : no_pipeline
29         always @(*) begin
30             out = in1;
31         end
32     end
33 endgenerate
34 endmodule
35
```

The top DSP module:

In the first stage we pass the inputs A,B,C,D,OPMODE to the Reg_Mux and the output of B and D is added or subbed depending on the opcode[6]. Then it's passed to a mux that gives its output depending on opmode[4].

```
50   wire [17:0] Bin;
51   generate
52     if (B_INPUT == "DIRECT") begin
53       assign Bin = B;
54     end else if (B_INPUT == "CASCADE") begin
55       assign Bin = BCIN;
56     end else begin
57       assign Bin = 0;
58     end
59   endgenerate
60   wire [17:0] D_reg , A_reg , B_reg;
61   wire [47:0] C_reg;
62   wire [7:0] OP_reg;
63   Reg_Mux #(DATAWIDTH(18) ,RSTTYPE(RSTTYPE) , .PIPLINE(DREG)) regmux1 (.CLK(CLK), .rst(RSTD), .clkenable(CED), .in1(D), .out(D_reg));
64   Reg_Mux #(DATAWIDTH(18) ,RSTTYPE(RSTTYPE) , .PIPLINE(BREG)) regmux2 (.CLK(CLK), .rst(RSTB), .in1(Bin), .clkenable(CEB), .out(B_reg));
65   Reg_Mux #(DATAWIDTH(18) ,RSTTYPE(RSTTYPE) , .PIPLINE(AOREG)) regmux3 (.CLK(CLK), .rst(RSTA), .in1(A), .clkenable(CEA), .out(A_reg));
66   Reg_Mux #(DATAWIDTH(48) ,RSTTYPE(RSTTYPE) , .PIPLINE(CREG)) regmux4 (.CLK(CLK), .rst(RSTC), .in1(C), .clkenable(CEC), .out(C_reg));
67   Reg_Mux #(DATAWIDTH(8) ,RSTTYPE(RSTTYPE) , .PIPLINE(OPMODEREG)) regmux5 (.CLK(CLK), .rst(RSTOPMODE), .in1(OPMODE), .clkenable(CEOPMODE), .out(OP_reg));
68   wire [17:0] pre_addersub_out;
69   wire [17:0] pre_addersub_out_mux;
70   assign pre_addersub_out = (OPMODE[6]) ? D_reg - B_reg : D_reg + B_reg;
71   assign pre_addersub_out_mux = (OPMODE[4]) ? pre_addersub_out : B_reg;
72
73
74
```

Then in the 2nd stage we pass the mux out and the first A reg output the another reg_mux and we multiply them then take the output to another reg_mux , the output of the reg_mux is taken on the output M.

```
74
75   wire [17:0] B1_reg , A1_reg;
76   Reg_Mux #(DATAWIDTH(18) ,RSTTYPE(RSTTYPE) , .PIPLINE(B1REG)) regmux6 (.CLK(CLK), .rst(RSTB), .in1(pre_addersub_out_mux), .clkenable(CEB), .out(B1_reg));
77   Reg_Mux #(DATAWIDTH(18) ,RSTTYPE(RSTTYPE) , .PIPLINE(A1REG)) regmux7 (.CLK(CLK), .rst(RSTA), .in1(A_reg), .clkenable(CEA), .out(A1_reg));
78   wire [35:0] multiplier_out;
79   assign BCOUT = B1_reg;
80   assign multiplier_out = B1_reg * A1_reg;
81   wire [35:0] multiplier_out_reg;
82   Reg_Mux #(DATAWIDTH(36) ,RSTTYPE(RSTTYPE) , .PIPLINE(MREG)) regmux8 (.CLK(CLK), .rst(RSTM), .in1(multiplier_out), .clkenable(CEM), .out(multiplier_out_reg));
83   assign M = multiplier_out_reg;
84
```

Then we have 2 muxs that gives the out based on opmode[1:0] and [3:2].

```
84   wire [47:0] D_A_B_concatenated;
85   assign D_A_B_concatenated = {D_reg[11:0],A1_reg,B1_reg};
86   wire [47:0] P_reg;
87   wire [47:0] mux_x_out, mux_z_out;
88   assign mux_x_out = (OPMODE[1:0] == 0) ? 0 : (OPMODE[1:0] == 1) ? {12{1'b0}},multiplier_out_reg} : (OPMODE[1:0] == 2) ? P_reg : D_A_B_concatenated;
89   assign mux_z_out = (OPMODE[3:2] == 0) ? 0 : (OPMODE[3:2] == 1) ? PCIN : (OPMODE[3:2] == 2) ? P_reg : C_reg;
90
91
92
93
```

At last we pass the carryin based on the parameter carryinsel,

We pass the output of the muxs z and x to an addersub.

And it's output is based on another reg_mux and that's our final output P.

```
96   wire carryinmux_out;
97   generate
98     if(CARRYINSEL == "OPMODE5") begin
99       assign carryinmux_out = OPMODE[5];
100    end else if(CARRYINSEL == "CARRYIN") begin
101      assign carryinmux_out = CARRYIN;
102    end
103    else begin
104      assign carryinmux_out = 0;
105    end
106  endgenerate
107
108  wire cin_reg;
109  Reg_Mux #(,DATAWIDTH(1),.RSTTYPE(RSTTYPE), .PIPELINE(CARRYINREG)) regmux9 (.CLK(CLK), .rst(RSTCARRYIN), .in1(carryinmux_out),.clkenable(CECARRYIN), .out(cin_reg));
110  wire [47:0] post_addersub_out;
111  wire cout;
112  assign {cout,post_addersub_out} = (OPMODE[7]) ? (mux_z_out - (mux_x_out+ cin_reg)) : mux_z_out + mux_x_out + cin_reg;
113  Reg_Mux #(,DATAWIDTH(48),.RSTTYPE(RSTTYPE), .PIPELINE(PREG)) regmux10 (.CLK(CLK), .rst(RSTP), .in1(post_addersub_out),.clkenable(CEP), .out(P_reg));
114  assign P = P_reg;
115  assign PCOUT = P_reg;
116  wire cout_reg;
117  Reg_Mux #(,DATAWIDTH(1),.RSTTYPE(RSTTYPE), .PIPELINE(CARRYOUTREG)) regmux11 (.CLK(CLK), .rst(RSTCARRYIN), .in1(cout),.clkenable(CECARRYIN), .out(cout_reg));
118  assign CARRYOUT = cout_reg;
119  assign CARRYOUTF = cout_reg;
120
121 endmodule
122
```

2 – Testbench:

Test 2.1:

```
77 initial begin
78     $display("Starting Test 2.1");
79     RSTA = 1;
80     RSTB = 1;
81     RSTM = 1;
82     RSTC = 1;
83     RSTD = 1;
84     RSTCARRYIN = 1;
85     RSTOPMODE = 1;
86     RSTP = 1;
87     A = $random;
88     B = $random;
89     D = $random;
90     C = $random;
91     CARRYIN = $random;
92     OPMODE = $random;
93     BCIN = $random;
94     CEA = $random;
95     CEB = $random;
96     CEM = $random;
97     CEP = $random;
98     CEC = $random;
99     CED = $random;
100    CECARRYIN = $random;
101    CEOPMODE = $random;
102    PCIN = $random;
103    @(negedge CLK);
104    if (P !== 48'd0 || M !== 36'd0 || CARRYOUT !== 1'b0 || CARRYOUTF !== 1'b0 || BCOUT !== 18'd0 || PCOUT !== 48'd0) begin
105        $display("2.1 Self-check FAILED");
106        $display("P = %h, M = %h, CARRYOUT = %b, CARRYOUTF = %b, BCOUT = %h, PCOUT = %h", P, M, CARRYOUT, CARRYOUTF, BCOUT, PCOUT);
107    end else begin
108        $display("2.1 Self-check PASSED");
109    end
110
111
```

```
112     RSTA = 0;
113     RSTB = 0;
114     RSTM = 0;
115     RSTC = 0;
116     RSTD = 0;
117     RSTCARRYIN = 0;
118     RSTOPMODE = 0;
119     RSTP = 0;
120     CEA = 1;
121     CEB = 1;
122     CEM = 1;
123     CEP = 1;
124     CEC = 1;
125     CED = 1;
126     CECARRYIN = 1;
127     CEOPMODE = 1;
```

Test 2.2:

```
150     $display("Starting Test 2.2");
151     OPMODE = 8'b11011101;
152     A = 20;
153     B = 10;
154     C = 350;
155     D = 25;
156     BCIN = $random;
157     PCIN = $random;
158     CARRYIN = $random;
159     repeat(4) @(negedge CLK);
160     if (BCOUT !== 18'h0000F || M !== 36'h00000000012C ||
161         P !== 48'h00000000032 || PCOUT !== 48'h00000000032 ||
162         CARRYOUT !== 1'b0 || CARRYOUTF !== 1'b0) begin
163         $display("2.2 Self-check FAILED");
164         $display("Expected -> BCOUT = %h, M = %h, P = %h, PCOUT = %h, CARRYOUT = %b, CARRYOUTF = %b",
165                 18'h0000F, 36'h12C, 48'h32, 48'h32, 1'b0, 1'b0);
166         $display("Actual    -> BCOUT = %h, M = %h, P = %h, PCOUT = %h, CARRYOUT = %b, CARRYOUTF = %b",
167                 BCOUT, M, P, PCOUT, CARRYOUT, CARRYOUTF);
168     end else begin
169         $display("2.2 Self-check PASSED");
170     end
```

Test 2.3:

```
152     $display("Starting Test 2.3");
153     OPMODE = 8'b00010000;
154     A = 20;
155     B = 10;
156     C = 350;
157     D = 25;
158     BCIN = $random;
159     PCIN = $random;
160     CARRYIN = $random;
161     repeat(3) @(negedge CLK);
162     if (BCOUT !== 18'h00023 || M !== 36'h000000002BC ||
163         P !== 48'h000000000000 || PCOUT !== 48'h000000000000 ||
164         CARRYOUT !== 1'b0 || CARRYOUTF !== 1'b0) begin
165         $display("Self-check FAILED at time %0t", $time);
166         $display("Expected -> BCOUT = %h, M = %h, P = %h, PCOUT = %h, CARRYOUT = %b, CARRYOUTF = %b",
167                 18'h00023, 36'h2BC, 48'h0, 48'h0, 1'b0, 1'b0);
168         $display("Actual    -> BCOUT = %h, M = %h, P = %h, PCOUT = %h, CARRYOUT = %b, CARRYOUTF = %b",
169                 BCOUT, M, P, PCOUT, CARRYOUT, CARRYOUTF);
170     end else begin
171         $display("2.3 Self-check PASSED");
172     end
```

Test 2.4:

```
174     $display("Starting Test 2.4");
175     OPMODE = 8'b00001010;
176     A = 20;
177     B = 10;
178     C = 350;
179     D = 25;
180     BCIN = $random;
181     PCIN = $random;
182     CARRYIN = $random;
183     prev_P = P;
184     prev_CARRYOUT = CARRYOUT;
185
186     repeat(3) @(negedge CLK);
187     if (BCOUT !== 18'h0000A || M !== 36'h00000000C8 ||
188         P !== PCOUT || P !== prev_P ||
189         CARRYOUT !== CARRYOUTF || CARRYOUT !== prev_CARRYOUT) begin
190
191         $display("Test 2.4 Self-check FAILED");
192         $display("Expected -> BCOUT = %h, M = %h, P = PCOUT = previous P = %h, CARRYOUT = CARRYOUTF = previous CARRYOUT = %b",
193                 | 18'hA, 36'h8, prev_P, prev_CARRYOUT);
194         $display("Actual    -> BCOUT = %h, M = %h, P = %h, PCOUT = %h, CARRYOUT = %b, CARRYOUTF = %b",
195                 | BCOUT, M, P, PCOUT, CARRYOUT, CARRYOUTF);
196
197     end else begin
198         $display("Test 2.4 Self-check PASSED");
199     end
```

Test 2.5:

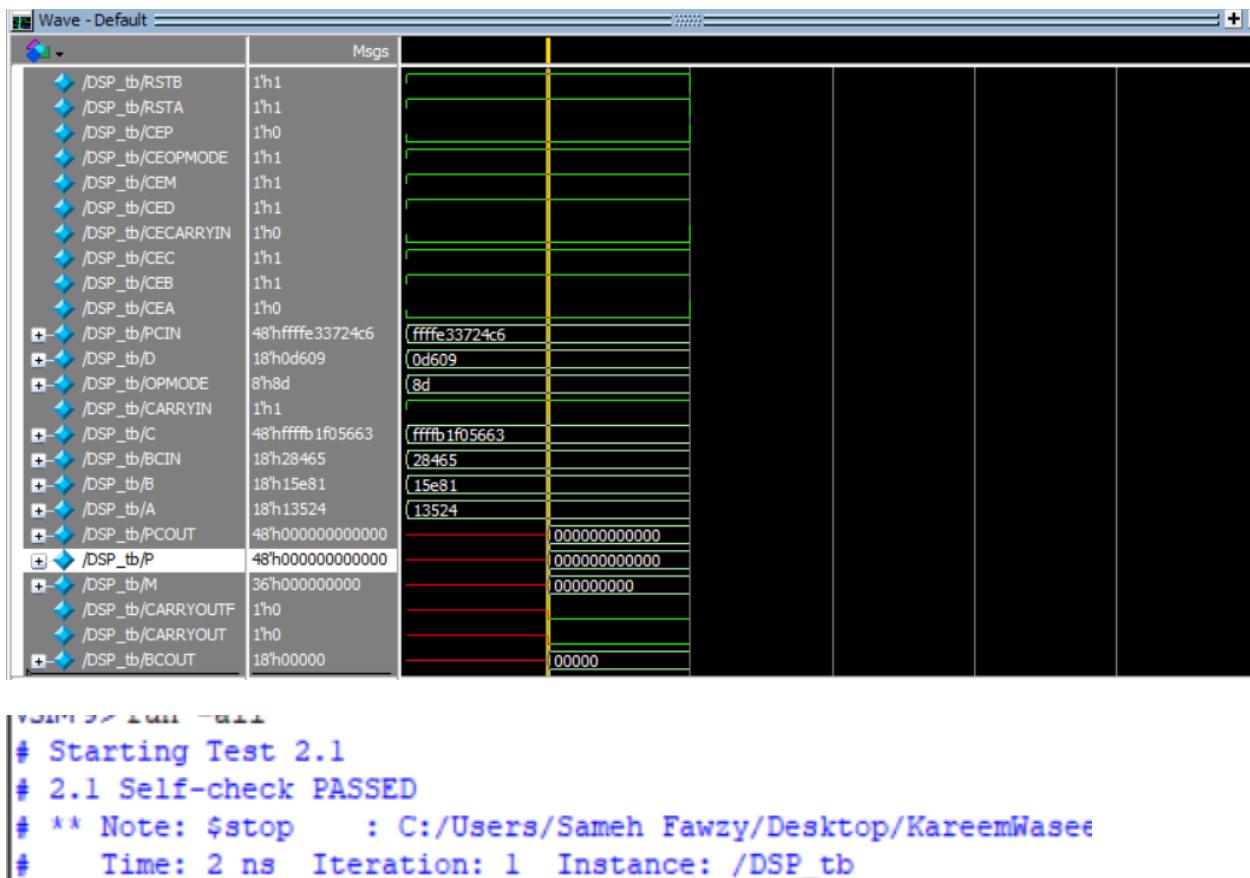
```
201     $display("Starting Test 2.5");
202     OPMODE = 8'b10100111;
203     A = 5;
204     B = 6;
205     C = 350;
206     D = 25;
207     BCIN = $random;
208     PCIN = 3000;
209     CARRYIN = $random;
210     repeat(3) @(negedge CLK);
211     if (BCOUT !== 18'h00006 || M !== 36'h0000000001E ||
212         P !== 48'hFE6FFFEC0BB1 || PCOUT !== 48'hFE6FFFEC0BB1 ||
213         CARRYOUT !== 1'b1 || CARRYOUTF !== 1'b1) begin
214         $display("Test 2.5 Self-check FAILED");
215         $display("Expected -> BCOUT = %h, M = %h, P = PCOUT = %h, CARRYOUT = CARRYOUTF = %b",
216                 | 18'h6, 36'h1E, 48'hFE6FFFEC0BB1, 1'b1);
217         $display("Actual    -> BCOUT = %h, M = %h, P = %h, PCOUT = %h, CARRYOUT = %b, CARRYOUTF = %b",
218                 | BCOUT, M, P, PCOUT, CARRYOUT, CARRYOUTF);
219     end else begin
220         $display("Test 2.5 Self-check PASSED");
221     end
```

3 – Do File:

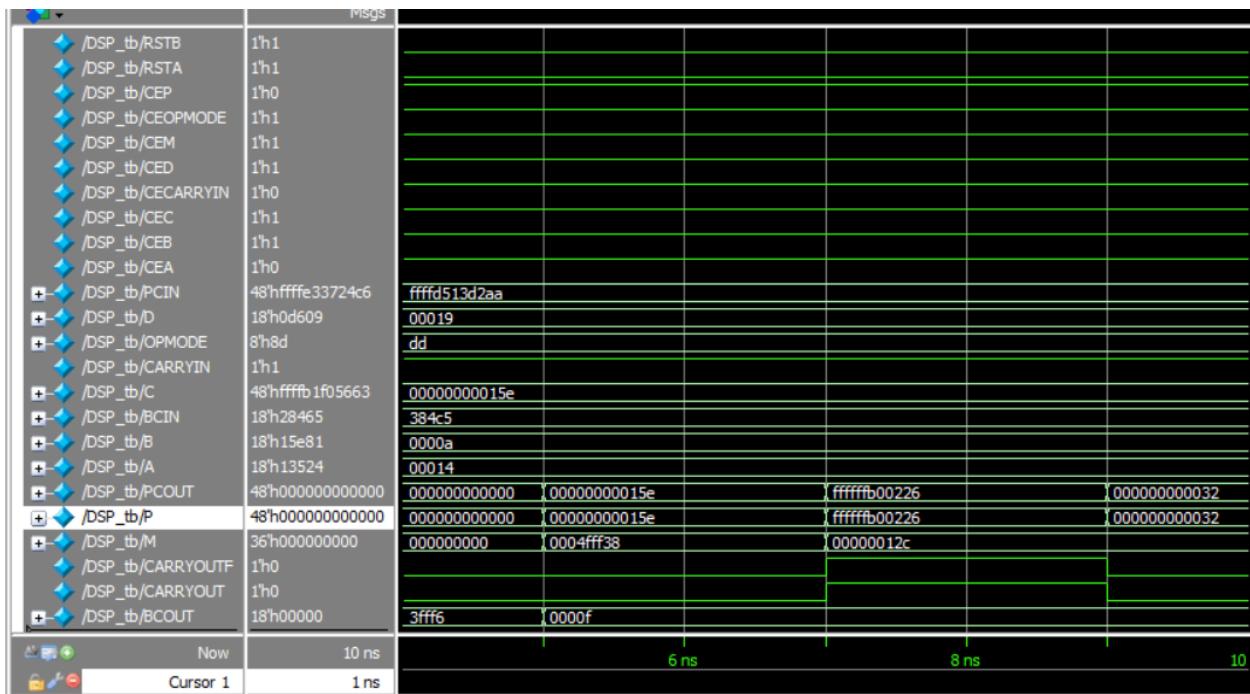
```
1 vlib work
2 vlog DSP_tb.v DSP.v Reg&Mux.v
3 vsim -voptargs=+acc work.DSP_tb.v
4 add wave *
5 run -all
6 #quit -sim
```

4 – QuestaSim Snippets:

For Test 2.1:

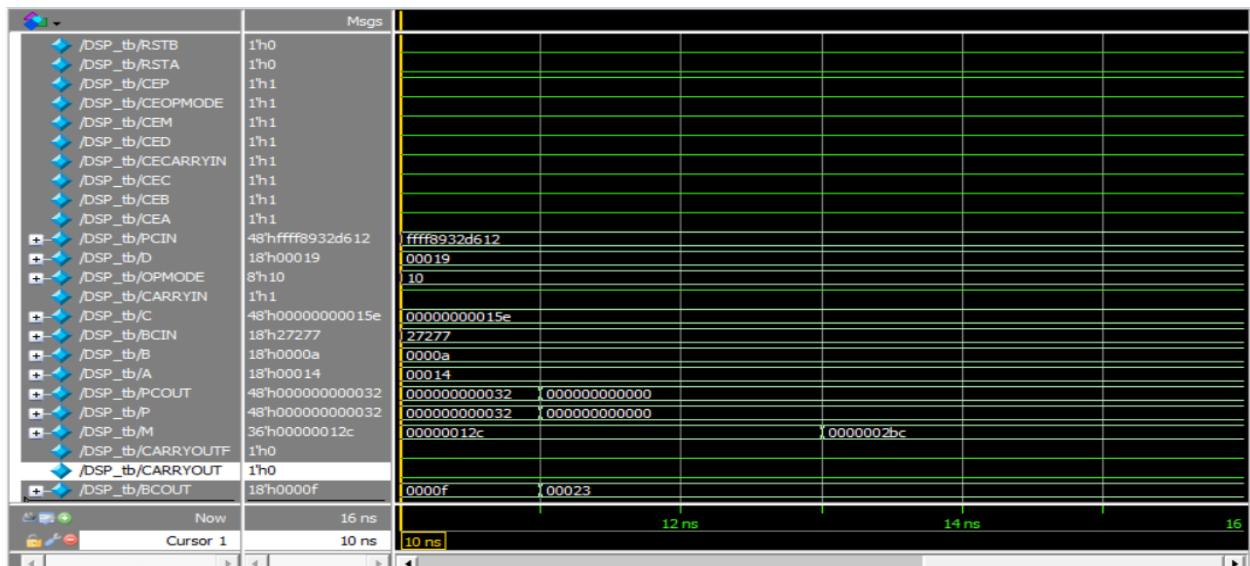


Test 2.2:



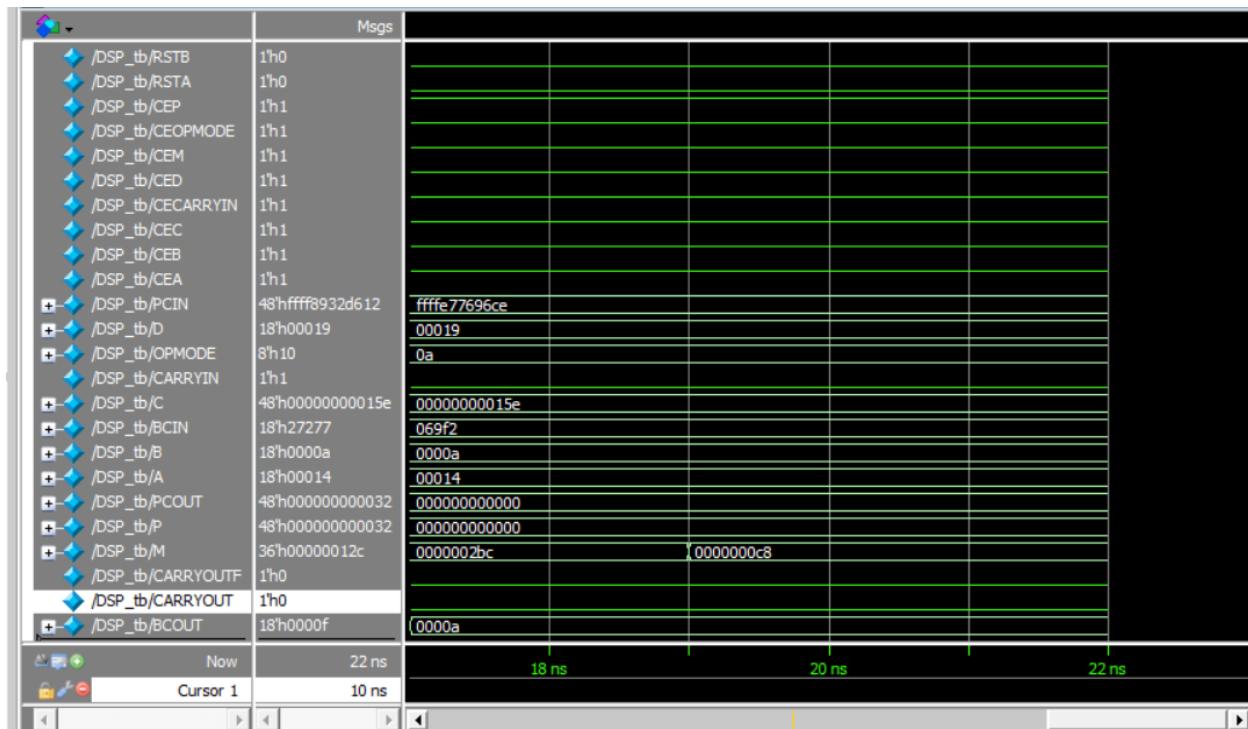
```
# Starting Test 2.2
# 2.2 Self-check PASSED
# ** Note: $stop      : C:/Users/Sameh Fawzy/Desktop/Verilog/Project/verilog/rtl/DSP_tb.sv
#           Time: 10 ns  Iteration: 1  Instance: /DSP_tb
```

Test 2.3:



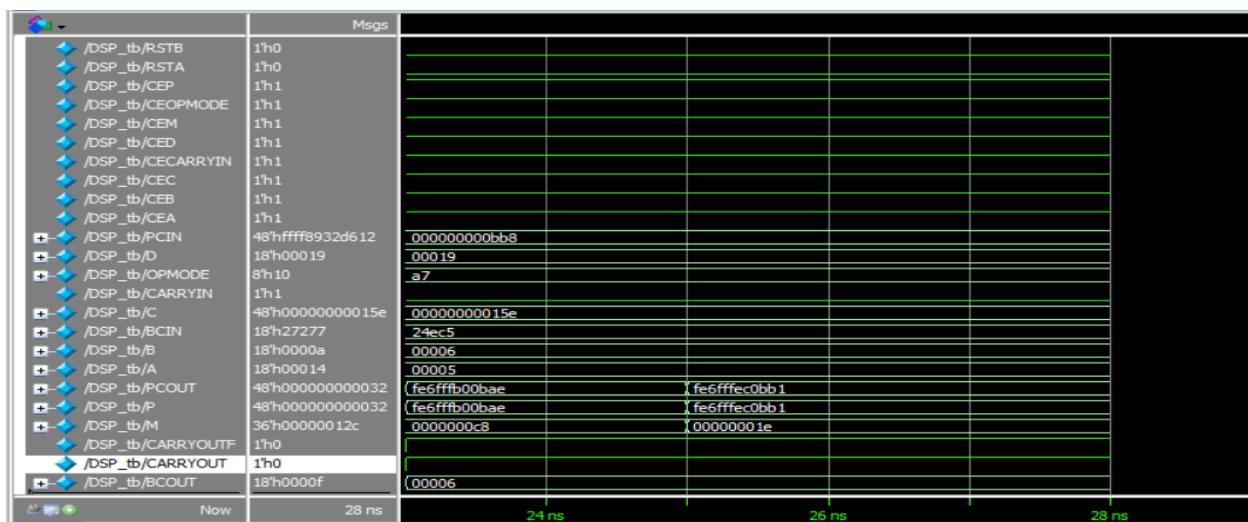
```
# Starting Test 2.3
# 2.3 Self-check PASSED
# ** Note: $stop    : C:/Users/Sameh Fawzy/Desktop/KareemWaseem/Project 1 DSP/DSP_tb.v(175)
#      Time: 16 ns Iteration: 1 Instance: /DSP tb
```

Test 2.4:



```
# Starting Test 2.4
# Test 2.4 Self-check PASSED
# ** Note: $stop      : C:/Users/Sameh Fawzy/Desktop/KareemWaseem/Project 1 DSP/DSP_tb.v(202)
#   Time: 22 ns Iteration: 1 Instance: /DSP tb
```

Test 2.5:



```

# Starting Test 2.5
# Test 2.5 Self-check PASSED
# ** Note: $stop    : C:/Users/Sameh Fawzy/Desktop/KareemWaseem/Project 1 DSP/DSP_tb.v(225)
#      Time: 28 ns  Iteration: 1  Instance: /DSP_tb

```

5 – Constraint File:

We will only define the clock frequency.

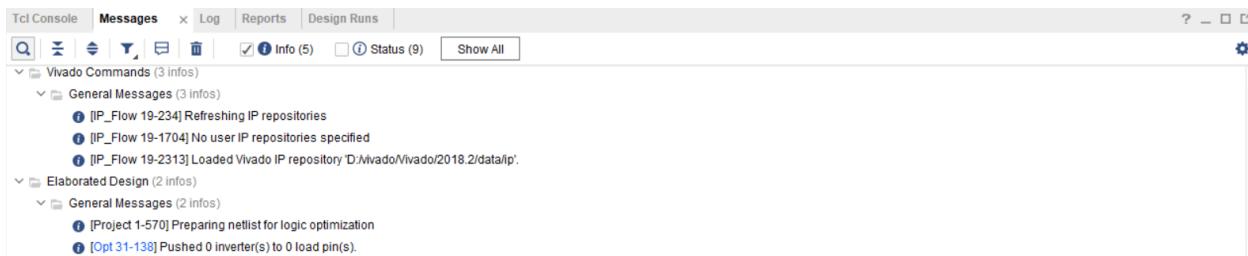
```

@ DSP.xdc
1  ## This file is a general .xdc for the Basys3 rev B board
2  ## To use it in a project:
3  ## - uncomment the lines corresponding to used pins
4  ## - rename the used ports (in each line, after get_ports) according to the top level signal names in the project
5
6  # Clock signal
7  set_property -dict { PACKAGE_PIN W5  IOSTANDARD LVCMOS33 } [get_ports CLK]
8  create_clock -add -name sys_clk_pin -period 10.00 -waveform {0 5} [get_ports CLK]
9
10
11 ## Switches
12 #set_property -dict { PACKAGE_PIN V17  IOSTANDARD LVCMOS33 } [get_ports {sw[0]}]
13 #set_property -dict { PACKAGE_PIN V16  IOSTANDARD LVCMOS33 } [get_ports {sw[1]}]
14 #set_property -dict { PACKAGE_PIN W16  IOSTANDARD LVCMOS33 } [get_ports {sw[2]}]
15 #set_property -dict { PACKAGE_PIN W17  IOSTANDARD LVCMOS33 } [get_ports {sw[3]}]
16 #set_property -dict { PACKAGE_PIN W15  IOSTANDARD LVCMOS33 } [get_ports {sw[4]}]
17 #set_property -dict { PACKAGE_PIN V15  IOSTANDARD LVCMOS33 } [get_ports {sw[5]}]
18 #set_property -dict { PACKAGE_PIN W14  IOSTANDARD LVCMOS33 } [get_ports {sw[6]}]
19 #set_property -dict { PACKAGE_PIN W13  IOSTANDARD LVCMOS33 } [get_ports {sw[7]}]
20 #set_property -dict { PACKAGE_PIN V2  IOSTANDARD LVCMOS33 } [get_ports {sw[8]}]
21 #set_property -dict { PACKAGE_PIN T3  IOSTANDARD LVCMOS33 } [get_ports {sw[9]}]
22 #set_property -dict { PACKAGE_PIN T2  IOSTANDARD LVCMOS33 } [get_ports {sw[10]}]
23 #set_property -dict { PACKAGE_PIN R3  IOSTANDARD LVCMOS33 } [get_ports {sw[11]}]
24 #set_property -dict { PACKAGE_PIN W2  IOSTANDARD LVCMOS33 } [get_ports {sw[12]}]
25 #set_property -dict { PACKAGE_PIN U1  IOSTANDARD LVCMOS33 } [get_ports {sw[13]}]
26 #set_property -dict { PACKAGE_PIN T1  IOSTANDARD LVCMOS33 } [get_ports {sw[14]}]
27 #set_property -dict { PACKAGE_PIN R2  IOSTANDARD LVCMOS33 } [get_ports {sw[15]}]
28
29
30 ## LEDs
31 #set_property -dict { PACKAGE_PIN U16  IOSTANDARD LVCMOS33 } [get_ports {led[0]}]
32 #set_property -dict { PACKAGE_PIN E19  IOSTANDARD LVCMOS33 } [get_ports {led[1]}]
33 #set_property -dict { PACKAGE_PIN U19  IOSTANDARD LVCMOS33 } [get_ports {led[2]}]
34 #set_property -dict { PACKAGE_PIN V19  IOSTANDARD LVCMOS33 } [get_ports {led[3]}]
35 #set_property -dict { PACKAGE_PIN W18  IOSTANDARD LVCMOS33 } [get_ports {led[4]}]
36 #set_property -dict { PACKAGE_PIN U15  IOSTANDARD LVCMOS33 } [get_ports {led[5]}]
37 #set_property -dict { PACKAGE_PIN U14  IOSTANDARD LVCMOS33 } [get_ports {led[6]}]
38 #set_property -dict { PACKAGE_PIN V14  IOSTANDARD LVCMOS33 } [get_ports {led[7]}]
39 #set_property -dict { PACKAGE_PIN V13  IOSTANDARD LVCMOS33 } [get_ports {led[8]}]
40 #set_property -dict { PACKAGE_PIN V3  IOSTANDARD LVCMOS33 } [get_ports {led[9]}]
41 #set_property -dict { PACKAGE_PIN W3  IOSTANDARD LVCMOS33 } [get_ports {led[10]}]
42 #set_property -dict { PACKAGE_PIN U3  IOSTANDARD LVCMOS33 } [get_ports {led[11]}]
43 #set_property -dict { PACKAGE_PIN P3  IOSTANDARD LVCMOS33 } [get_ports {led[12]}]
44 #set_property -dict { PACKAGE_PIN N3  IOSTANDARD LVCMOS33 } [get_ports {led[13]}]
45 #set_property -dict { PACKAGE_PIN P1  IOSTANDARD LVCMOS33 } [get_ports {led[14]}]

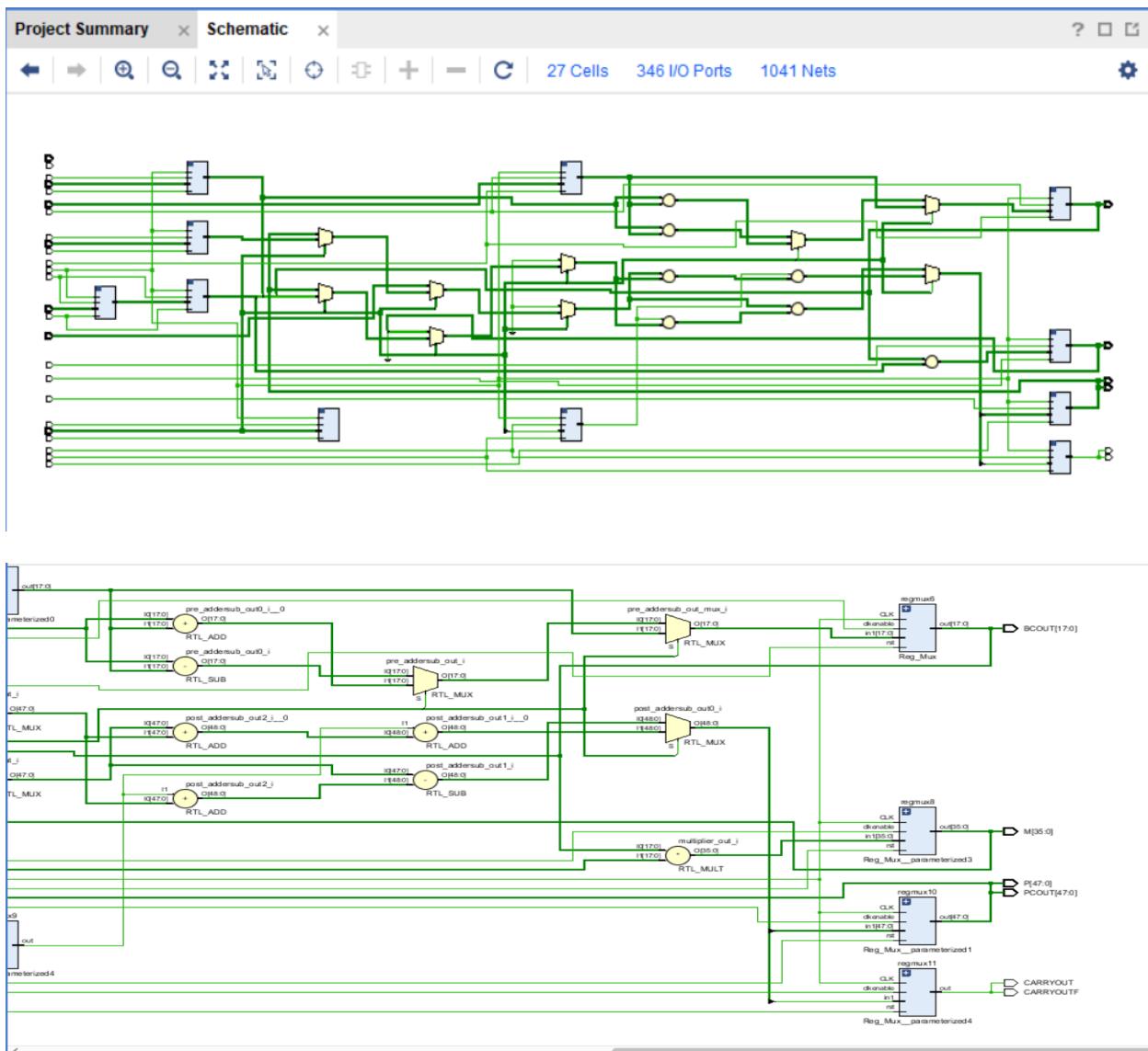
```

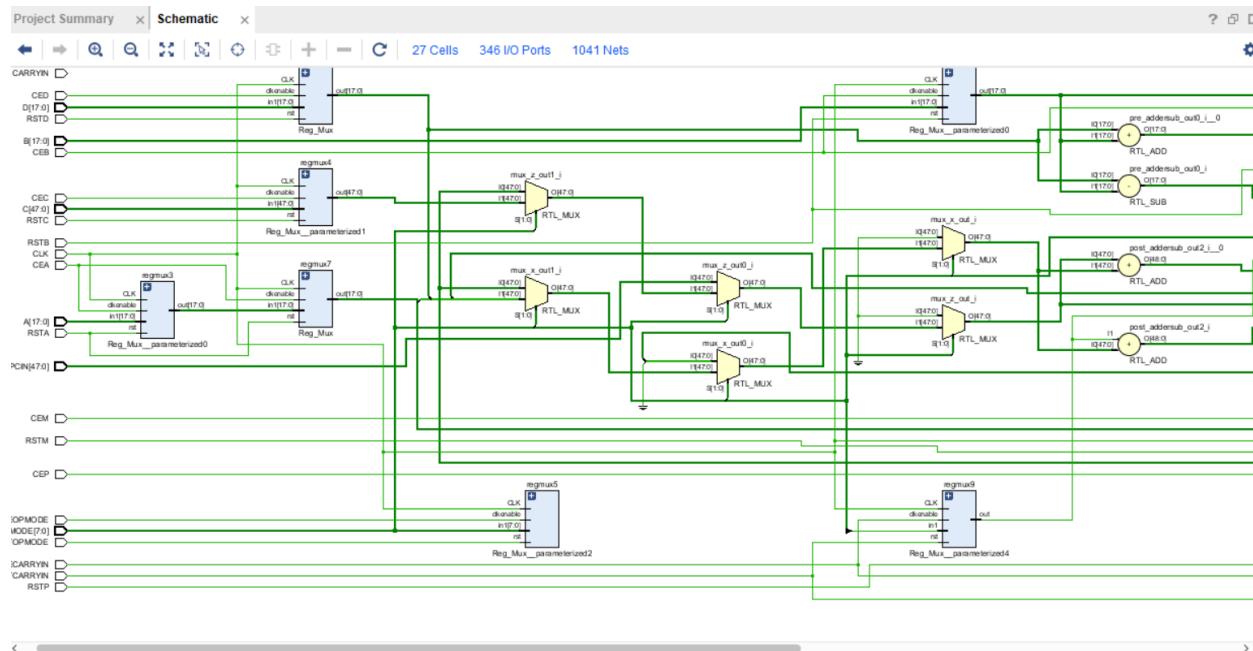
6 – Elaboration :

Messages tab:



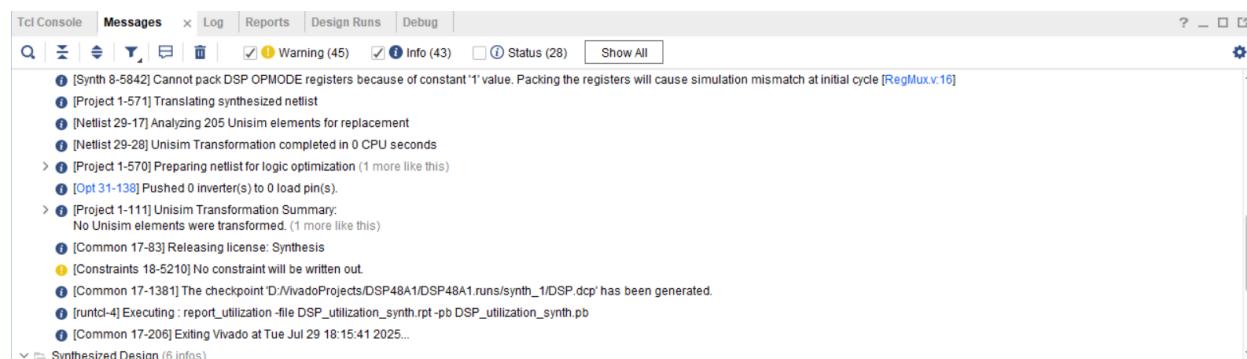
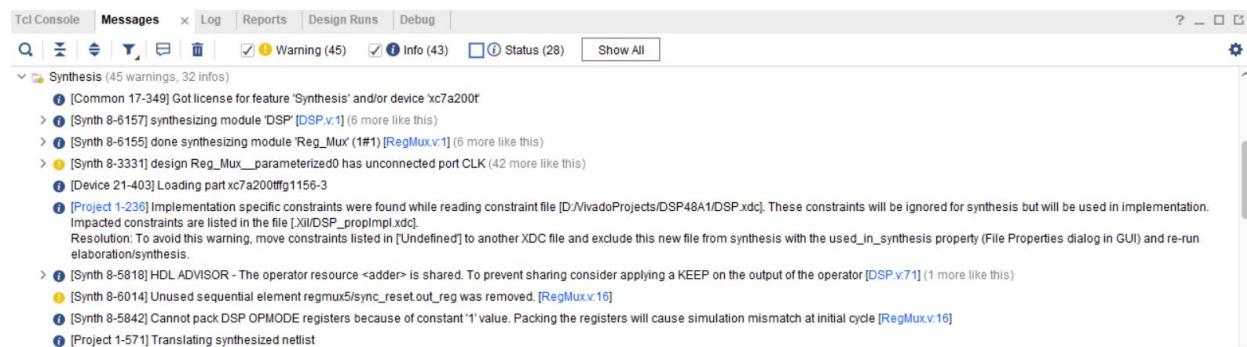
Schematic snippets:





7 – Synthesis:

Messages:



Tcl Console Messages Log Reports Design Runs Debug

Warning (45) Info (43) Status (28) Show All

(Common 17-83) Releasing license: Synthesis

- [Constraints 18-5210] No constraint will be written out.
- [Common 17-1381] The checkpoint 'D:/VivadoProjects/DSP48A1/DSP48A1.runs/synth_1/DSP.dcp' has been generated.
- [runrtl-4] Executing : report_utilization -file DSP_utilization_synth.rpt -pb DSP_utilization_synth.pb
- [Common 17-206] Exiting Vivado at Tue Jul 29 18:15:41 2025...

Synthesized Design (6 infos)

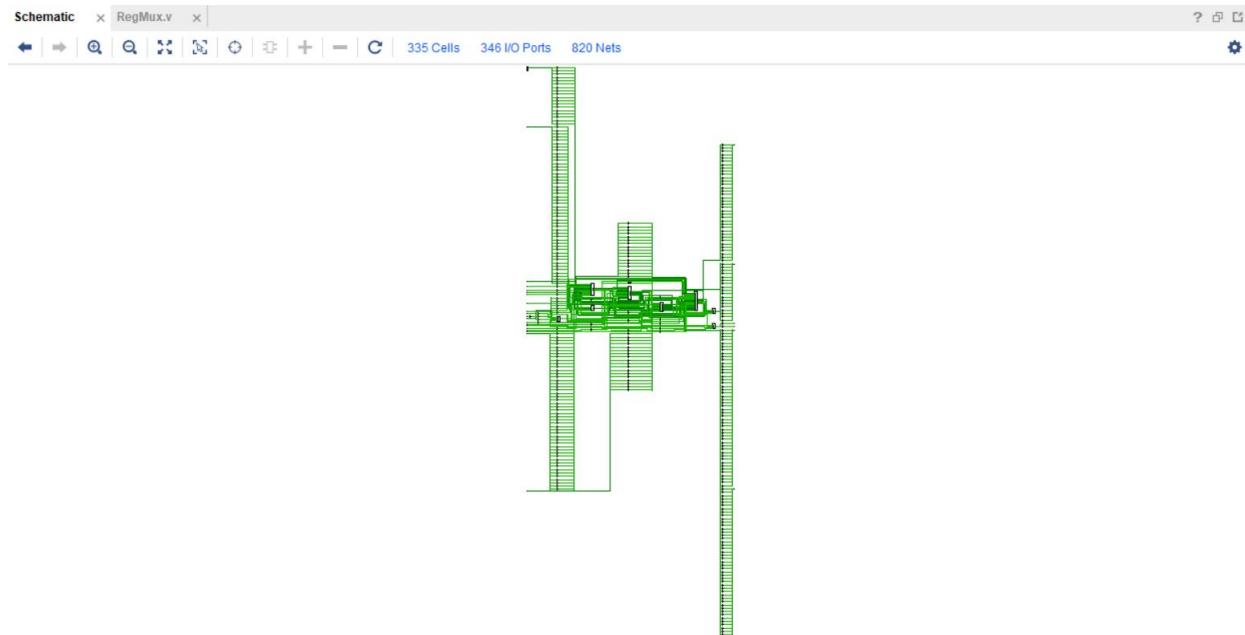
General Messages (6 infos)

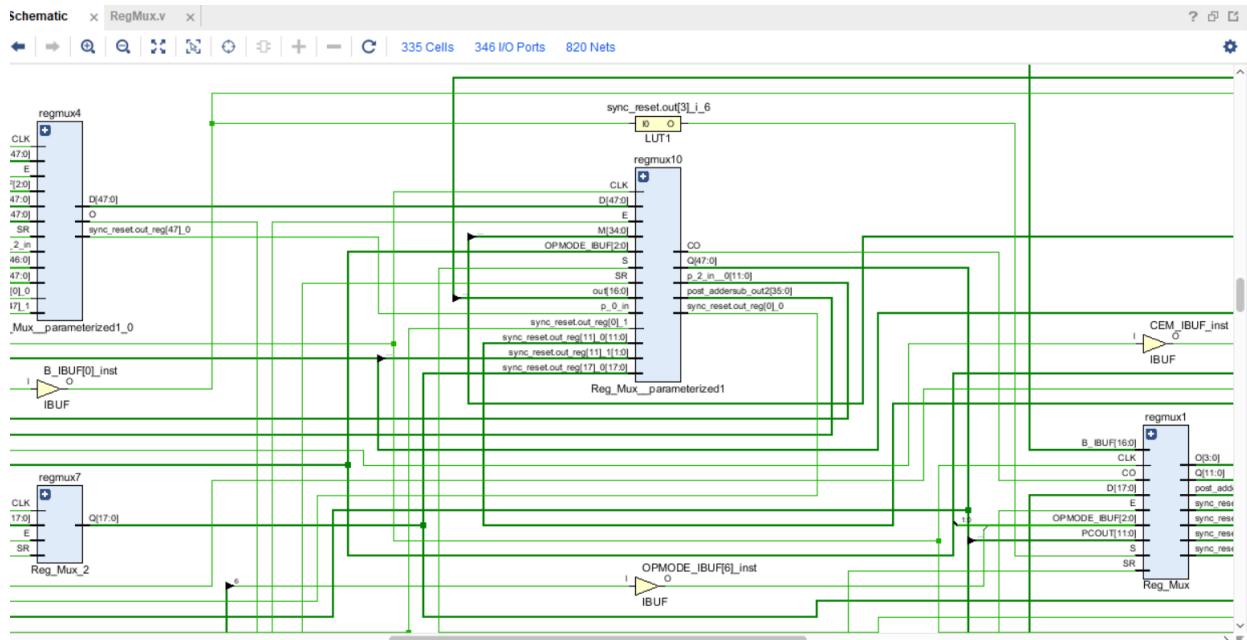
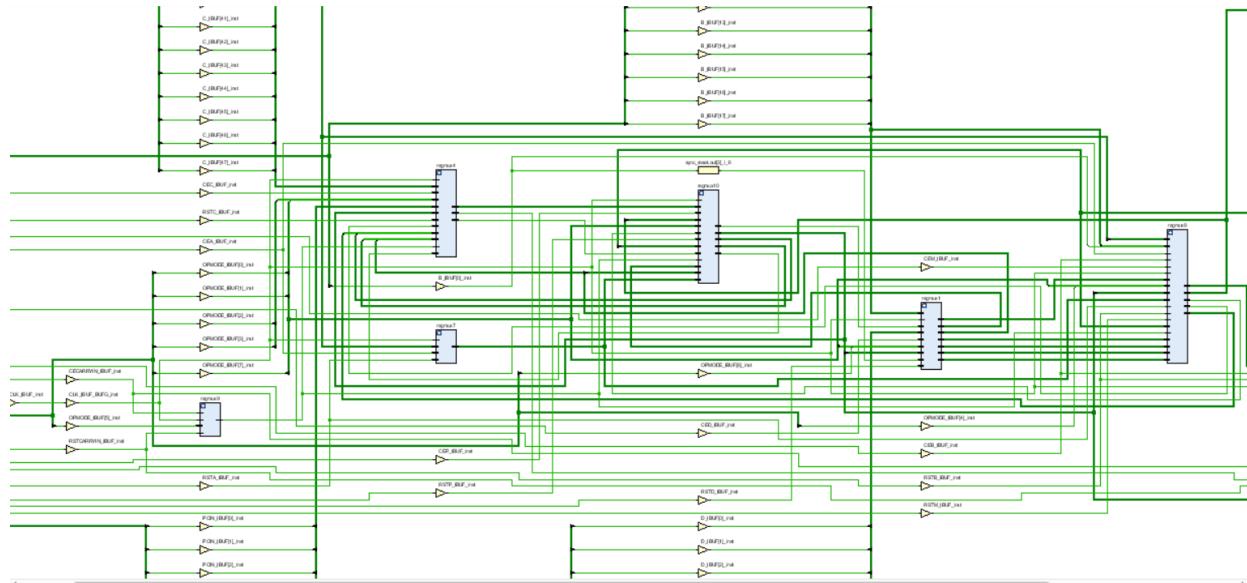
- [Netlist 29-17] Analyzing 205 Unisim elements for replacement
- [Netlist 29-28] Unisim Transformation completed in 0 CPU seconds
- [Project 1-479] Netlist was created with Vivado 2018.2
- [Project 1-570] Preparing netlist for logic optimization
- [Opt 31-138] Pushed 0 inverter(s) to 0 load pin(s).
- [Project 1-111] Unisim Transformation Summary:
No Unisim elements were transformed.

Note: This warning is because if parameter B_input is not cascade we won't use BCIN in anything.

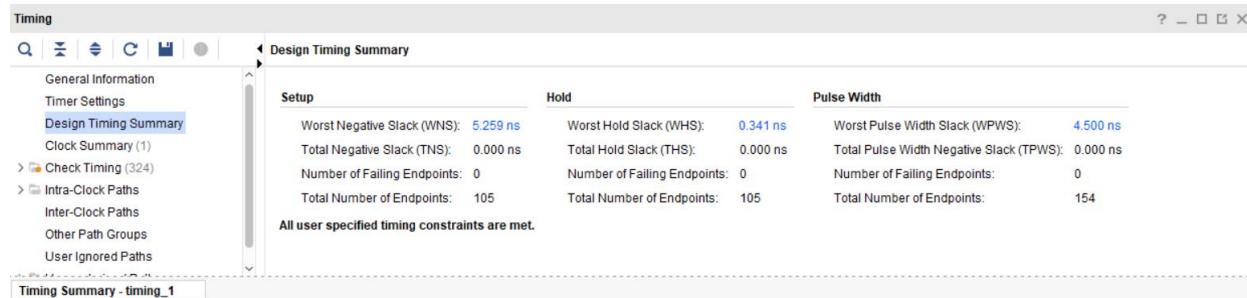
- [Synth 8-3331] design DSP has unconnected port BCIN[12]
- [Synth 8-3331] design DSP has unconnected port BCIN[11]
- [Synth 8-3331] design DSP has unconnected port BCIN[10]
- [Synth 8-3331] design DSP has unconnected port BCIN[9]
- [Synth 8-3331] design DSP has unconnected port BCIN[8]
- [Synth 8-3331] design DSP has unconnected port BCIN[7]
- [Synth 8-3331] design DSP has unconnected port BCIN[6]
- [Synth 8-3331] design DSP has unconnected port BCIN[5]
- [Synth 8-3331] design DSP has unconnected port BCIN[4]
- [Synth 8-3331] design DSP has unconnected port BCIN[3]
- [Synth 8-3331] design DSP has unconnected port BCIN[2]
- [Synth 8-3331] design DSP has unconnected port BCIN[1]
- [Synth 8-3331] design DSP has unconnected port BCIN[0]

Schematic:

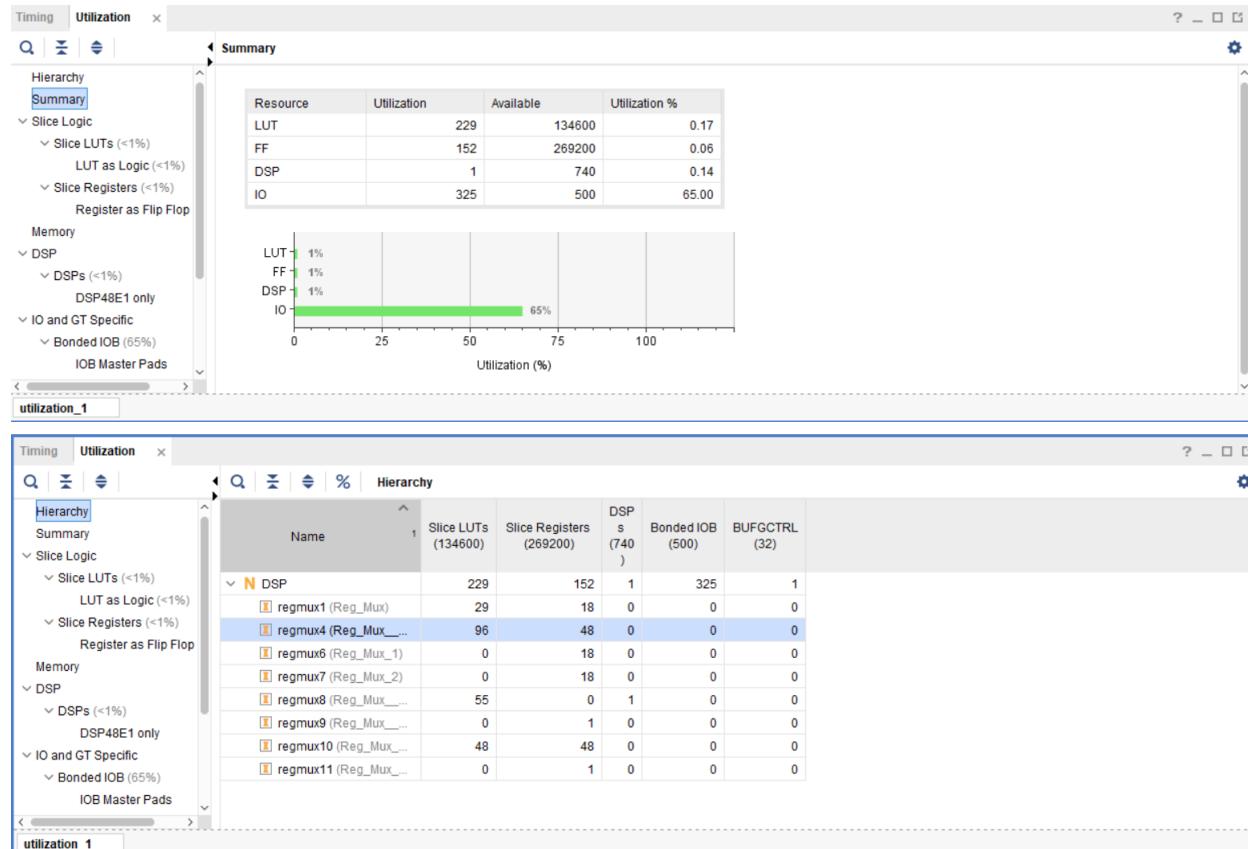




Timing report:

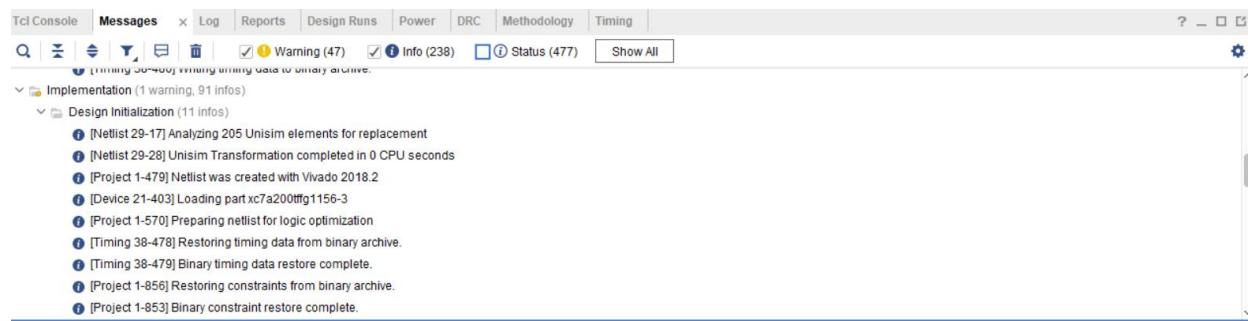


Utilization report:



8 – Implementation:

Messages:



Tcl Console Messages Log Reports Design Runs Power DRC Methodology Timing

Q | X | D | T | M | B | Warning (47) Info (238) Status (477) Show All

- [Common 17-349] Got license for feature 'Implementation' and/or device 'xc7a200t'
- [Project 1-481] DRC finished with 0 Errors
- [Project 1-462] Please refer to the DRC report (report_drc) for more information.
- [Opt 31-49] Retargeted 0 cell(s).
- > [Opt 31-138] Pushed 0 inverter(s) to 0 load pin(s). (1 more like this)
- > [Opt 31-389] Phase Retarget created 0 cells and removed 0 cells (4 more like this)
- [Opt 31-662] Phase BUFG optimization created 0 cells of which 0 are BUFGs and removed 0 cells.
- [Pwropt 34-132] Skipping clock gating for clocks with a period < 2.00 ns.
- [Common 17-83] Releasing license: Implementation
- [Timing 38-480] Writing timing data to binary archive.
- [Common 17-1381] The checkpoint D:\VivadoProjects\DSP48A1\DSP48A1.runs\impl_1\DSP_opt.dcp has been generated.
- [InfoCl-4] Executing report_drc file DSP_drc_opted.rpt_nb DSP_drc_opted.nb_mx DSP_drc_opted.mv

Tcl Console Messages Log Reports Design Runs Power DRC Methodology Timing

Q | X | D | T | M | B | Warning (47) Info (238) Status (477) Show All

- > [Vivado_Tcl 4-199] Please refer to the DRC report (report_drc) for more information. (1 more like this)
- [Place 30-611] Multithreading enabled for place_design using a maximum of 2 CPUs
- [Opt 31-138] Pushed 0 inverter(s) to 0 load pin(s).
- > [Timing 38-35] Done setting XDC timing constraints. (2 more like this)
- [Physopt 32-65] No nets found for high-fanout optimization.
- [Physopt 32-232] Optimized 0 net. Created 0 new instance.
- [Physopt 32-775] End 1 Pass. Optimized 0 net or cell. Created 0 new cell, deleted 0 existing cell and moved 0 existing cell
- [Place 46-31] BUFG insertion identified 0 candidate nets, 0 success, 0 skipped for placement/routing, 0 skipped for timing, 0 skipped for netlist change reason.
- [Place 30-746] Post Placement Timing Summary WNS=4.328. For the most accurate timing information please run report_timing.
- [Common 17-83] Releasing license: Implementation
- [Timing 38-480] Writing timing data to binary archive.

Tcl Console Messages Log Reports Design Runs Power DRC Methodology Timing

Q | X | D | T | M | B | Warning (47) Info (238) Status (477) Show All

- Implemented Design (9 infos)
 - General Messages (9 infos)
 - [Netlist 29-17] Analyzing 205 Unisim elements for replacement
 - [Netlist 29-28] Unisim Transformation completed in 0 CPU seconds
 - [Project 1-479] Netlist was created with Vivado 2018.2
 - [Project 1-570] Preparing netlist for logic optimization
 - [Timing 38-478] Restoring timing data from binary archive.
 - [Timing 38-479] Binary timing data restore complete.
 - [Project 1-856] Restoring constraints from binary archive.
 - [Project 1-853] Binary constraint restore complete.
 - [Project 1-111] Unisim Transformation Summary:
No Unisim elements were transformed.

There is no critical warnings or errors.

Timing report:

Tcl Console Messages Log Reports Design Runs Power DRC Methodology Timing

Design Timing Summary

Setup	Hold	Pulse Width
Worst Negative Slack (WNS): 4.184 ns	Worst Hold Slack (WHS): 0.391 ns	Worst Pulse Width Slack (WPWS): 4.500 ns
Total Negative Slack (TNS): 0.000 ns	Total Hold Slack (THS): 0.000 ns	Total Pulse Width Negative Slack (TPWNS): 0.000 ns
Number of Failing Endpoints: 0	Number of Failing Endpoints: 0	Number of Failing Endpoints: 0
Total Number of Endpoints: 124	Total Number of Endpoints: 124	Total Number of Endpoints: 173

All user specified timing constraints are met.

Timing Summary - impl_1 (saved) | Timing Summary - timing_1 |

Utilization:

Tcl Console | Messages | Log | Reports | Design Runs | Power | DRC | Methodology | Timing | Utilization | x

Summary

Hierarchy

- Summary
- Slice Logic
 - Slice LUTs (<1%)
 - LUT as Logic (<1%)
 - Slice Registers (<1%)
 - Register as Flip Flop (
- Slice Logic Distribution
 - Slice (<1%)
 - SLICEM
 - SLICEL
 - LUT Flip Flop Pairs (<1%)
 - LUT-FF pairs with one
 - LUT-FF pairs with one
 - LUT as Logic (<1%)
 - using O5 and O6
 - using O6 output only

Resource	Utilization	Available	Utilization %
LUT	228	133800	0.17
FF	171	267600	0.06
DSP	1	740	0.14
IO	325	500	65.00

Utilization (%)

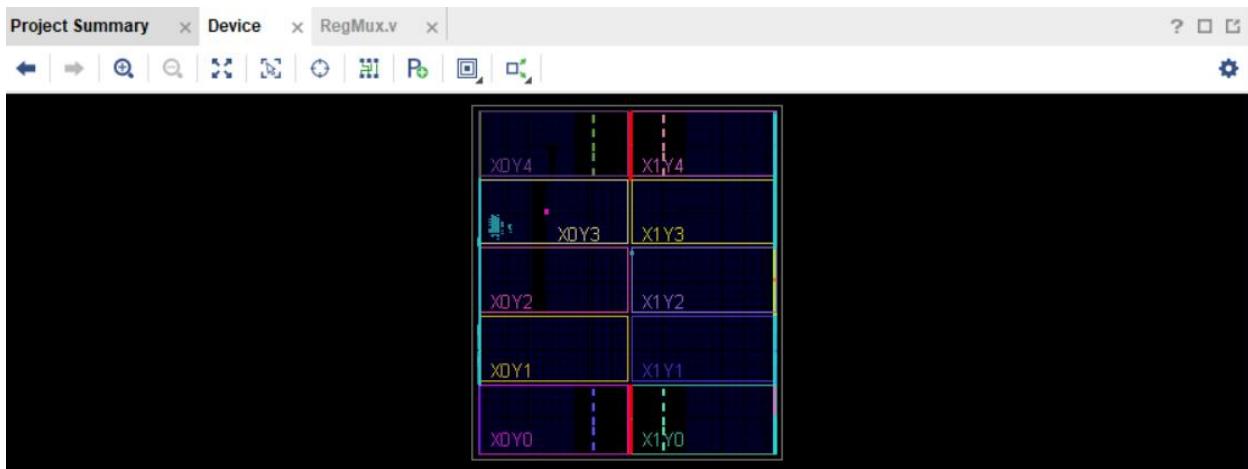
Tcl Console | Messages | Log | Reports | Design Runs | Power | DRC | Methodology | Timing | Utilization | ? | - | x | ⚙️

Hierarchy

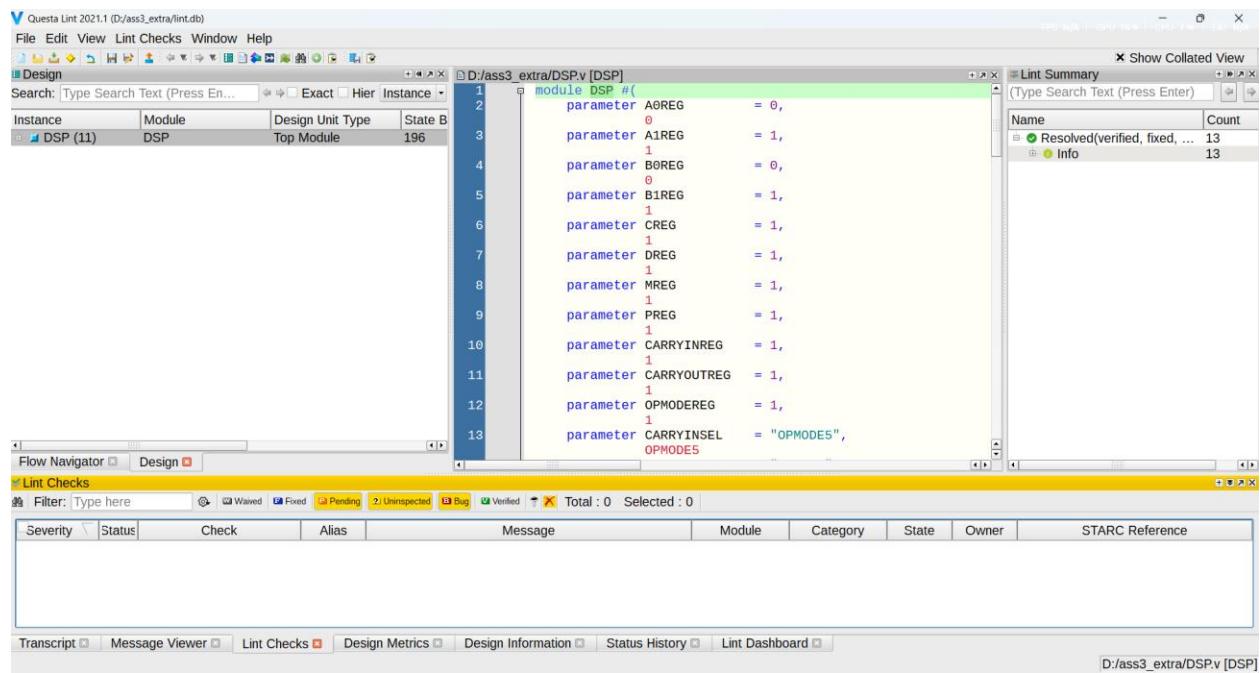
Name	Slice LUTs (133800)	Slice Registers (267600)	Slice (33450)	LUT as Logic (133800)	LUT Flip Flop Pairs (133800)	DSPs (740)	Bonded IOB (500)	BUFGCTRL (32)
regmux1 (Reg_Mux)	29	18	15	29	0	0	0	0
regmux4 (Reg_Mux_...)	96	48	45	96	0	0	0	0
regmux6 (Reg_Mux_1)	0	36	16	0	0	0	0	0
regmux7 (Reg_Mux_2)	0	18	7	0	0	0	0	0
regmux8 (Reg_Mux_...)	55	0	19	55	0	1	0	0
regmux9 (Reg_Mux_...)	0	1	1	0	0	0	0	0
regmux10 (Reg_Mux_...)	48	48	25	48	0	0	0	0
regmux11 (Reg_Mux_...)	0	2	2	0	0	0	0	0

utilization_1

Device:



9 – Linting Tool:



No errors.

