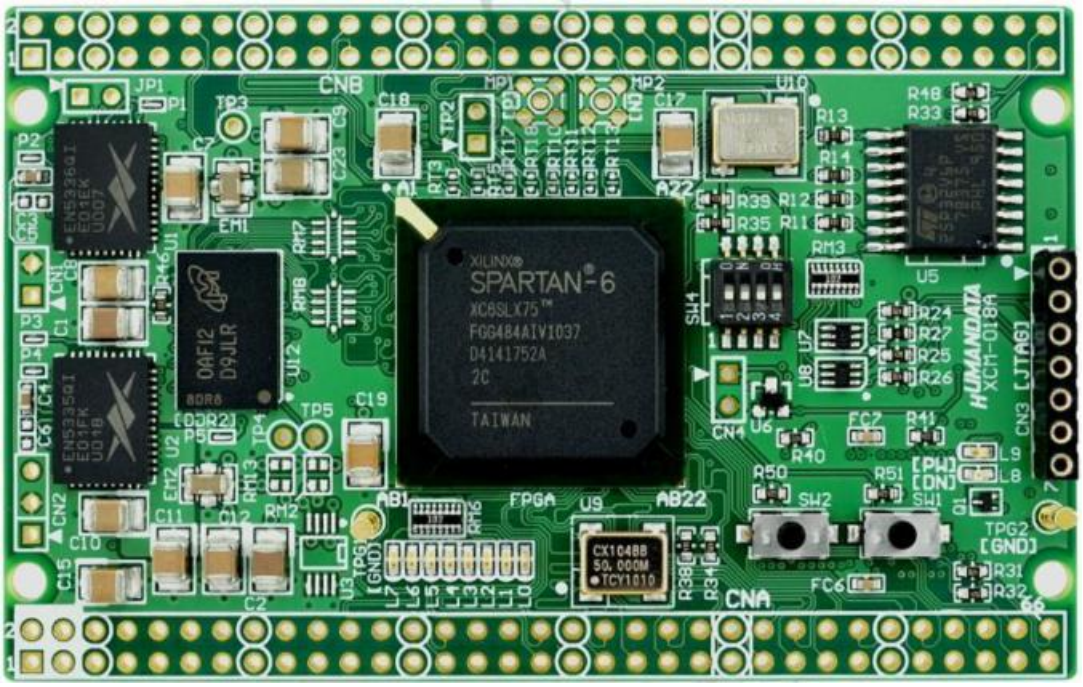
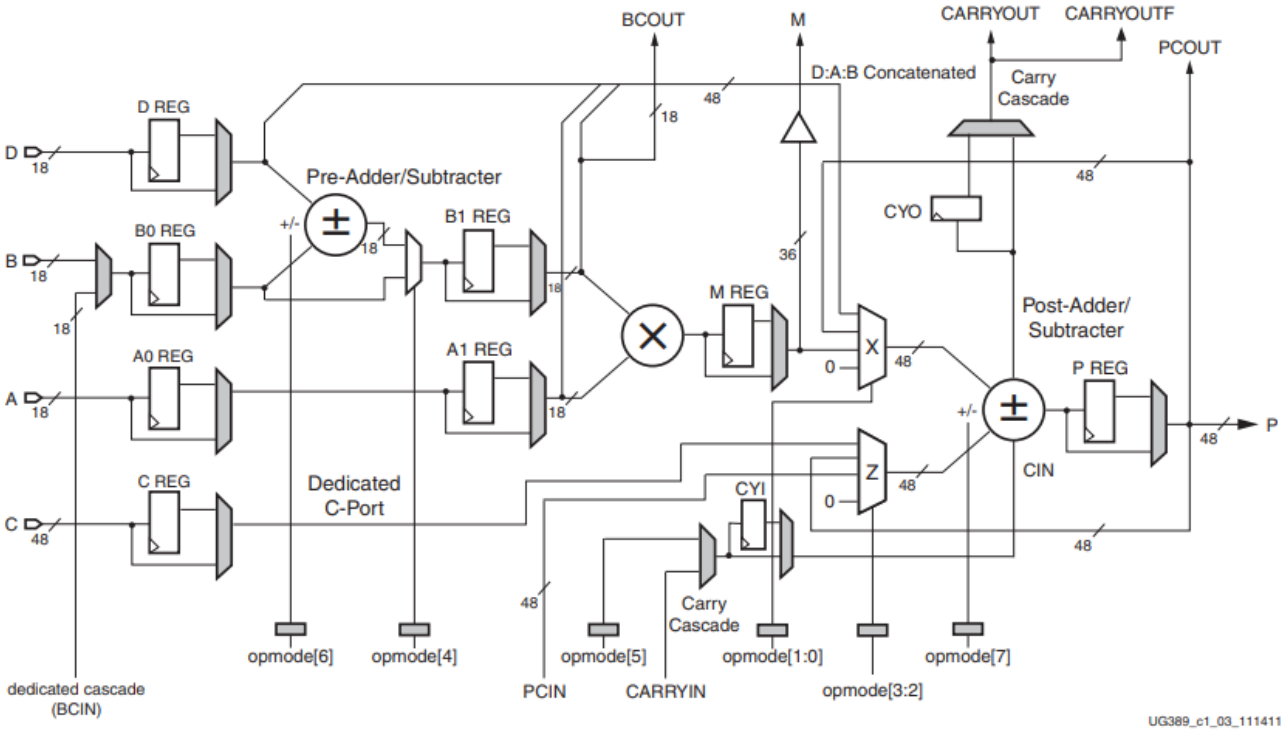
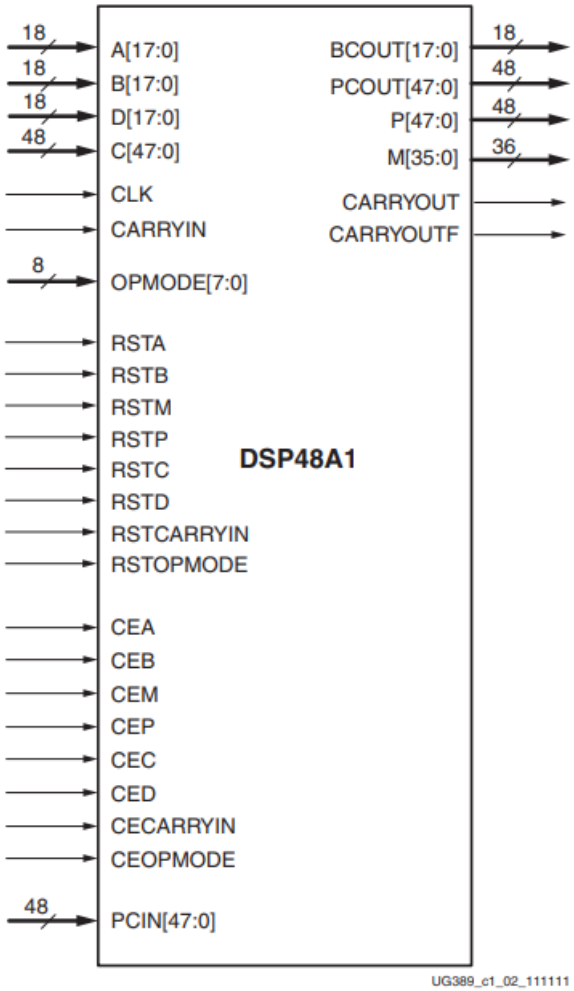
Spartan6 – DSP48A1



Prepared By : Mohamed Mostafa Mohamed Ali

Submitted to : Eng/Kareem Waseem


BCOUT
M
CARRYOUT
CARRYOUTF
PCOUT
D:A:B Concatenated
Carry Cascade
D REG
D
18
Pre-Adder/Subtracter
+/-
B0 REG
B
18
B1 REG
CYO
M REG
Post-Adder/ Subtracter
A0 REG
A1 REG
X
P REG
P
A
Z
+/-
CIN
C REG
Dedicated C-Port
C
48
CYI
0
Carry Cascade
opmode[6]
opmode[4]
opmode[5]
opmode[1:0]
opmode[7]
dedicated cascade (BCIN)
PCIN
CARRYIN
opmode[3:2]
UG389_c1_03_111411


A[17:0]
B[17:0]
D[17:0]
C[47:0]
CLK
CARRYIN
OPMODE[7:0]
RSTA
RSTB
RSTM
RSTP
RSTC
RSTD
RSTCARRYIN
RSTOPMODE
CEA
CEB
CEM
CEP
CEC
CED
CECARRYIN
CEOPMODE
PCIN[47:0]
DSP48A1
BCOUT[17:0]
PCOUT[47:0]
P[47:0]
M[35:0]
CARRYOUT
CARRYOUTF
UG389_c1_02_111111

1) RTL Code :

```
module DSP48A1 (A , B , C , D , CARRYIN, BCIN , M , P , CARRYOUT , CARRYOUTF , CLK , OPMODE , CEA,
                CEB, CEC , CED , CECARRYIN , CEM , CEOPMODE , CEP , RSTA , RSTB , RSTC ,
                RSTCARRYIN, RSTD, RSTM , RSTOPMODE , RSTP , BCOUT , PCIN , PCOUT);
parameter A0REG = 0;
parameter A1REG = 1;
parameter B0REG = 0;
parameter B1REG = 1;
parameter CREG  = 1;
parameter DREG  = 1;
parameter MREG  = 1;
parameter PREG  = 1;
parameter CARRYINREG = 1;
parameter CARRYOUTREG = 1;
parameter OPMODEREG = 1;
parameter CARRYINSEL = "OPMODES"; // takes values ("OPMODES" or "CARRYIN")
parameter B_INPUT = "DIRECT"; // takes values ("DIRECT" or "CASCADE")
parameter RSTTYPE = "SYNC"; // takes values ("SYNC" or "ASYNC")

input [17:0] A , B , D;
input [47:0] C ;
input CLK , CARRYIN;
input [7:0] OPMODE;
input [17:0] BCIN;
input RSTA , RSTB , RSTC , RSTCARRYIN , RSTD, RSTM , RSTOPMODE , RSTP;
input CEA , CEB , CEC , CECARRYIN, CED , CEM , CEOPMODE , CEP;
input [47:0] PCIN;

output [17:0] BCOUT;
output [47:0] PCOUT;
output [47:0] P;
output [35:0] M;
output CARRYOUT , CARRYOUTF;

wire [17:0] A0_IN;
wire [17:0] A_IN;
wire [17:0] B0_IN;
wire [17:0] B_IN;
wire [47:0] C_IN;
```

```
  wire [17:0] A0_IN;
  wire [17:0] A_IN;
  wire [17:0] B0_IN;
  wire [17:0] B_IN;
  wire [47:0] C_IN;
  wire [17:0] D_IN;
  wire [17:0] pre_a_s; // the pre-adder/subtractor result
  wire [17:0] MUX_4_OUT; //the output of the mux between add/sub and muul
  wire [35:0] MUL;    // the result of the multiplier
  wire [35:0] M_OUT;  // output of the M_REG
  wire [47:0] POUT_wire;  // output of the P_REG
  wire [47:0] MUX_X;
  wire [47:0] MUX_Z;
  wire CIN;
  wire Out_Carry;
  wire [47:0] post_a_s; // the post-adder/subtractor result
  wire COUT; // Carry out from the adder/subtractor
  // INSTANTIATION FOR A0_REG_MUX
    reg_mux #(.WIDTH(18) , .REG(A0REG))A0_REG_MUX
            (.in(A), .clk(CLK), .clk_en(CEA), .rst(RSTA), .out_mux(A0_IN));

  // INSTANTIATION FOR A1_REG_MUX
    reg_mux #(.WIDTH(18) , .REG(A1REG))A1_REG_MUX
            (.in(A0_IN), .clk(CLK), .clk_en(CEA), .rst(RSTA), .out_mux(A_IN));

  // INSTANTIATION FOR D_REG_MUX
    reg_mux #(.WIDTH(18) , .REG(DREG))D_REG_MUX
            (.in(D), .clk(CLK), .clk_en(CED), .rst(RSTD), .out_mux(D_IN));
```

```
63  // INSTANTIATION FOR B0_REG_MUX
64  generate
65    if(B_INPUT == "DIRECT")
66      begin
67        reg_mux #(.WIDTH(18) , .REG(B0REG))B0_REG_MUX_DIRECT
68                  (.in(B), .clk(CLK), .clk_en(CEB), .rst(RSTB), .out_mux(B0_IN));
69      end
70    else if(B_INPUT == "CASCADE")
71      begin
72        reg_mux #(.WIDTH(18) , .REG(B0REG))B0_REG_MUX_CASCADE
73                  (.in(BCIN), .clk(CLK), .clk_en(CEB), .rst(RSTB), .out_mux(B0_IN));
74      end
75    else
76        assign B0_IN = 18'b0; // Default case if B_INPUT is neither DIRECT nor CASCADE
77  endgenerate
78
79  //PRE ADDER/SUBTRACTOR
80  assign pre_a_s = OPMODE[6] ? (D_IN - B0_IN) : (D_IN + B0_IN) ; // pre_a_s is the result of the adder/subtractor for
81
82  assign MUX_4_OUT = OPMODE[4] ? pre_a_s : B0_IN;
83
84  // INSTANTIATION FOR B1_REG_MUX
85  reg_mux #(.WIDTH(18) , .REG(B1REG))B1_REG_MUX
86            (.in(MUX_4_OUT), .clk(CLK), .clk_en(CEB), .rst(RSTB), .out_mux(B_IN));
87
88  assign BCOUT = B_IN; // BCOUT (our first output)
89  assign MUL = B_IN * A_IN; // the result of the multiplier
90
91  // INSTANTIATION FOR C_REG_MUX
92  reg_mux #(.WIDTH(48) , .REG(CREG))C_REG_MUX
93            (.in(C), .clk(CLK), .clk_en(CEC), .rst(RSTC), .out_mux(C_IN));
94
95  // INSTANTIATION FOR M_REG_MUX
96  reg_mux #(.WIDTH(36) , .REG(MREG))M_REG_MUX
97            (.in(MUL), .clk(CLK), .clk_en(CEM), .rst(RSTM), .out_mux(M_OUT));
98
99  assign M = M_OUT; // M is the output of the multiplier
```

```
101   // Creating MUX X & Z
102 assign MUX_X = (OPMODE[1:0] == 2'b00) ? 48'b0 :
103                 (OPMODE[1:0] == 2'b01) ? {12'b0 , M_OUT} :
104                 (OPMODE[1:0] == 2'b10) ? POUT_wire :
105                 {D_IN[11:0], A_IN[17:0], B_IN[17:0]};
106 assign MUX_Z = (OPMODE[3:2] == 2'b00) ? 48'b0 :
107                 (OPMODE[3:2] == 2'b01) ? PCIN :
108                 (OPMODE[3:2] == 2'b10) ? POUT_wire :
109                 C_IN;
110   // INSTANTIATION FOR CIN_REG
111 generate
112   if(CARRYINSEL == "OPMODES")
113     begin
114       reg_mux #(.WIDTH(1) , .REG(CARRYINREG))cin_opmode5
115                 (.in(OPMODE[5]), .clk(CLK), .clk_en(CECARRYIN), .rst(RSTCARRYIN), .out_mux(CIN));
116     end
117   else if(B_INPUT == "CARRYIN")
118     begin
119      reg_mux #(.WIDTH(1) , .REG(CARRYINREG))cin_carry_cascade
120                 (.in(CARRYIN), .clk(CLK), .clk_en(CECARRYIN), .rst(RSTCARRYIN), .out_mux(CIN));
121     end
122   else
123       assign CIN = 18'b0; // Default case if CARRYINSEL is neither OPMODES nor CARRYIN
124   endgenerate
125   //POST ADDER/SUBTRACTOR
126   assign {Out_Carry , post_a_s} = OPMODE[7] ? (MUX_Z - (MUX_X + CIN)) : (MUX_X + MUX_Z + CIN);
127 // INSTANTIATION FOR P_REG_MUX
128   reg_mux #(.WIDTH(48) , .REG(PREG))P_REG_MUX
129             (.in(post_a_s), .clk(CLK), .clk_en(CEP), .rst(RSTP), .out_mux(POUT_wire));
130   assign P = POUT_wire; // P is the output of the DSP48A1
131   assign PCOUT = POUT_wire;
132 // CARRYOUT and CARRYOUTF
133   reg_mux #(.WIDTH(1) , .REG(CARRYOUTREG))CARRYOUT_REG_MUX
134             (.in(Out_Carry), .clk(CLK), .clk_en(CECARRYIN), .rst(RSTCARRYIN), .out_mux(COUT));
135   assign CARRYOUT = COUT;
136   assign CARRYOUTF = COUT;
137   endmodule
```

reg_Mux Module :

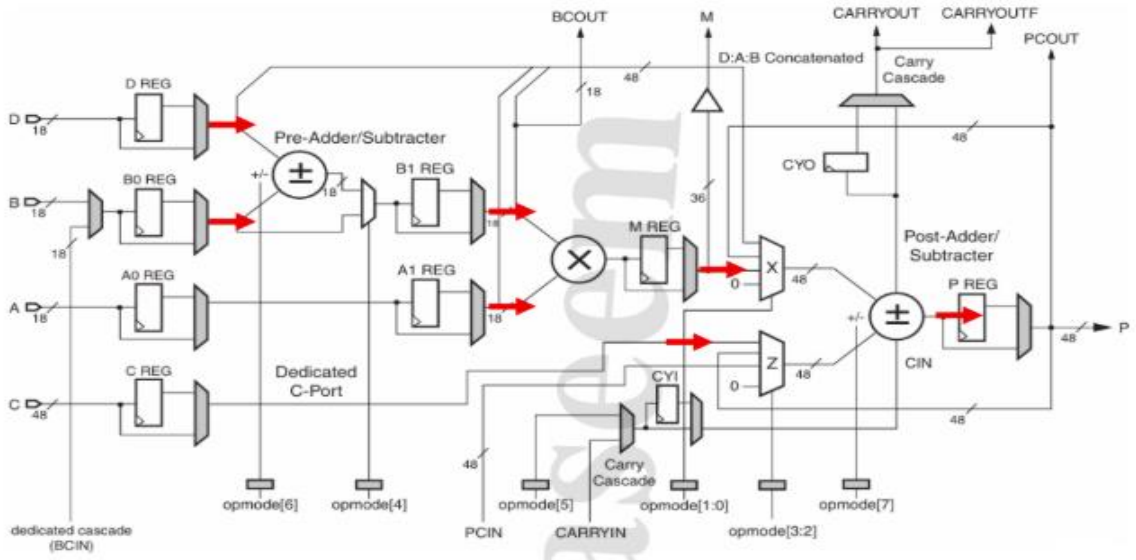
```
module reg_mux(in , clk , clk_en , rst , out_mux);
parameter WIDTH = 18; parameter REG = 1 ;
parameter RSTTYPE = "SYNC"; // takes values ("SYNC" or "ASYNC")
input [WIDTH-1:0] in;  input clk, rst , clk_en;
output [WIDTH-1:0] out_mux;
reg [WIDTH-1:0] out_reg;
generate
  if (RSTTYPE == "ASYNC")
    begin
      if(REG)
        begin
          always @(posedge clk or posedge rst)
            begin
                if (rst)
                    out_reg <= {WIDTH{1'b0}};
                else if (clk_en)
                    out_reg <= in;
            end
          assign out_mux = out_reg;
        end
      else
        assign out_mux = in; ;
    end
  else // RSTTYPE == "SYNC"
    begin
      if(REG)
        begin
          always @(posedge clk)
            begin
                if (rst)
                    out_reg <= {WIDTH{1'b0}};
                else if (clk_en)
                    out_reg <= in;
            end
          assign out_mux = out_reg;
        end
      else
        assign out_mux = in; ;
```

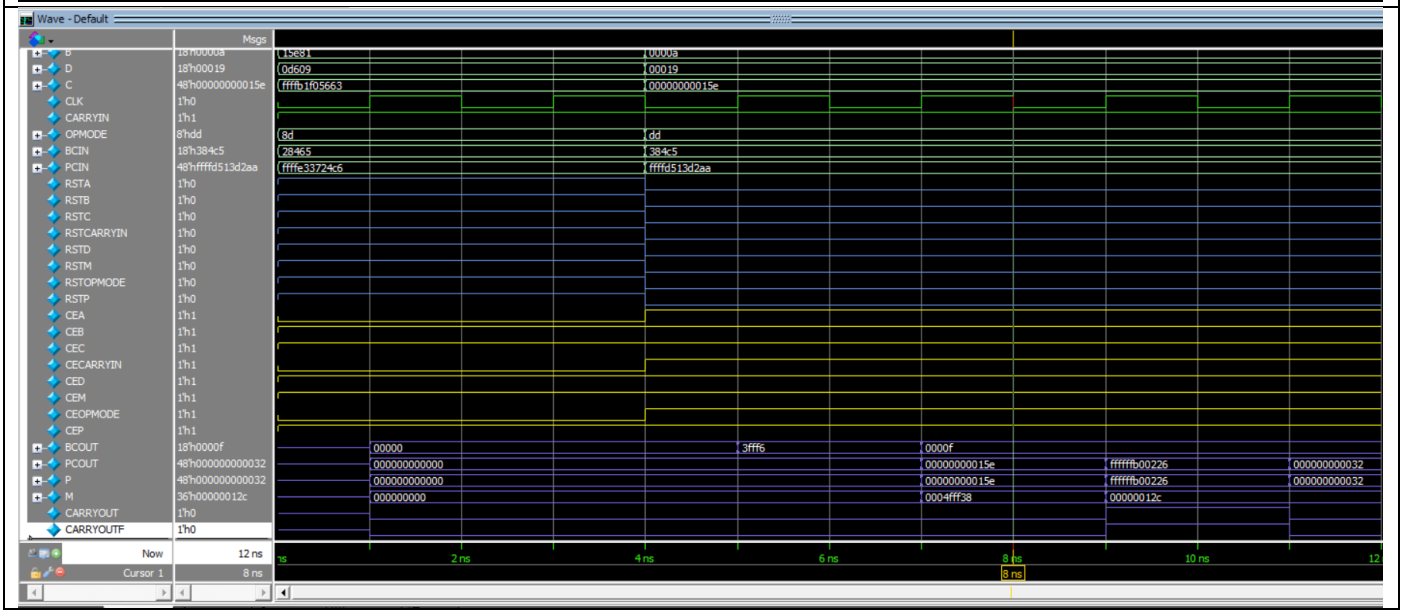
2) Testbench code :

```
module DSP_tb();
reg [17:0] A, B, D; reg [47:0] C ; reg CLK , CARRYIN; reg [7:0] OPMODE; reg [17:0] BCIN;
reg RSTA , RSTB , RSTC , RSTCARRYIN , RSTD, RSTM , RSTOPMODE , RSTP;
reg CEA , CEB , CEC , CECARRYIN, CED , CEM , CEOPMODE , CEP;
reg [47:0] PCIN;

wire [17:0] BCOUT; wire [47:0] PCOUT; wire [47:0] P; wire [35:0] M;
wire CARRYOUT , CARRYOUTF;
//with default parameters
DSP48A1 DUT (.A(A), .B(B), .D(D), .C(C), .CLK(CLK), .CARRYIN(CARRYIN), .OPMODE(OPMODE),
             .BCIN(BCIN), .RSTA(RSTA), .RSTB(RSTB), .RSTC(RSTC), .RSTCARRYIN(RSTCARRYIN),
             .RSTD(RSTD), .RSTM(RSTM), .RSTOPMODE(RSTOPMODE), .RSTP(RSTP),
             .CEA(CEA), .CEB(CEB), .CEC(CEC), .CECARRYIN(CECARRYIN), .CED(CED),
             .CEM(CEM), .CEOPMODE(CEOPMODE), .CEP(CEP), .PCIN(PCIN),.BCOUT(BCOUT), .PCOUT(PCOUT),
             .P(P), .M(M), .CARRYOUT(CARRYOUT), .CARRYOUTF(CARRYOUTF));
initial
  begin
    CLK = 0 ;
    forever #1 CLK = ~CLK;
  end
initial
  begin
    // Verify Reset Operation
    {RSTA, RSTB, RSTC, RSTCARRYIN, RSTD, RSTM, RSTOPMODE, RSTP} = 8'b11111111;
    A = $random; B = $random; D = $random; C = $random;
    CARRYIN = $random; OPMODE = $random; BCIN = $random; CEA = $random;
    CEB = $random; CEC = $random; CECARRYIN = $random; CED = $random; CEM = $random;
    CEOPMODE = $random; CEP = $random; PCIN = $random;
    @(negedge CLK);
    if (P != 48'b0 || M != 36'b0 || CARRYOUT != 1'b0 || CARRYOUTF != 1'b0)
      begin
        $display("Reset failed, P: %b, M: %b, CARRYOUT: %b, CARRYOUTF: %b", P, M, CARRYOUT, CARRYOUTF);
        $stop;
      end
    @(negedge CLK);
    {RSTA, RSTB, RSTC, RSTCARRYIN, RSTD, RSTM, RSTOPMODE, RSTP} = 8'b00000000;
    {CEA, CEB, CEC, CECARRYIN, CED, CEM, CEOPMODE, CEP} = 8'b11111111;
```

Path 1 :

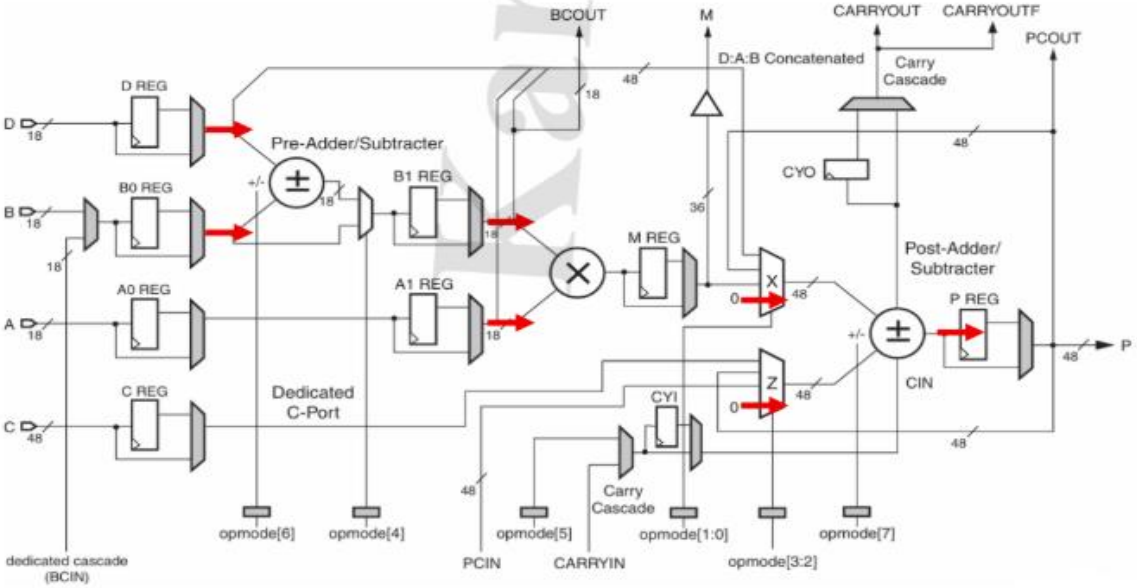
```
38      //verify DSP path 1
39      A = 20;
40      B = 10;
41      C = 350;
42      D = 25;
43      BCIN = $random;
44      PCIN = $random;
45      CARRYIN = $random;
46      OPMODE = 8'b11011101;
47      repeat(4) @(negedge CLK);
48      if(BCOUT != 'hf || M != 'h12c || P != 'h32 || PCOUT != 'h32
49         || CARRYOUTF != 0 || CARRYOUT != 0)
50         begin
51             $display("DSP Path 1 failed, BCOUT: %h, M: %h, P: %h, PCOUT: %h, CARRYOUTF: %b, CARRYOUT: %b",
52                     BCOUT, M, P, PCOUT, CARRYOUTF, CARRYOUT);
53             $stop;
54         end
55      $stop;
```

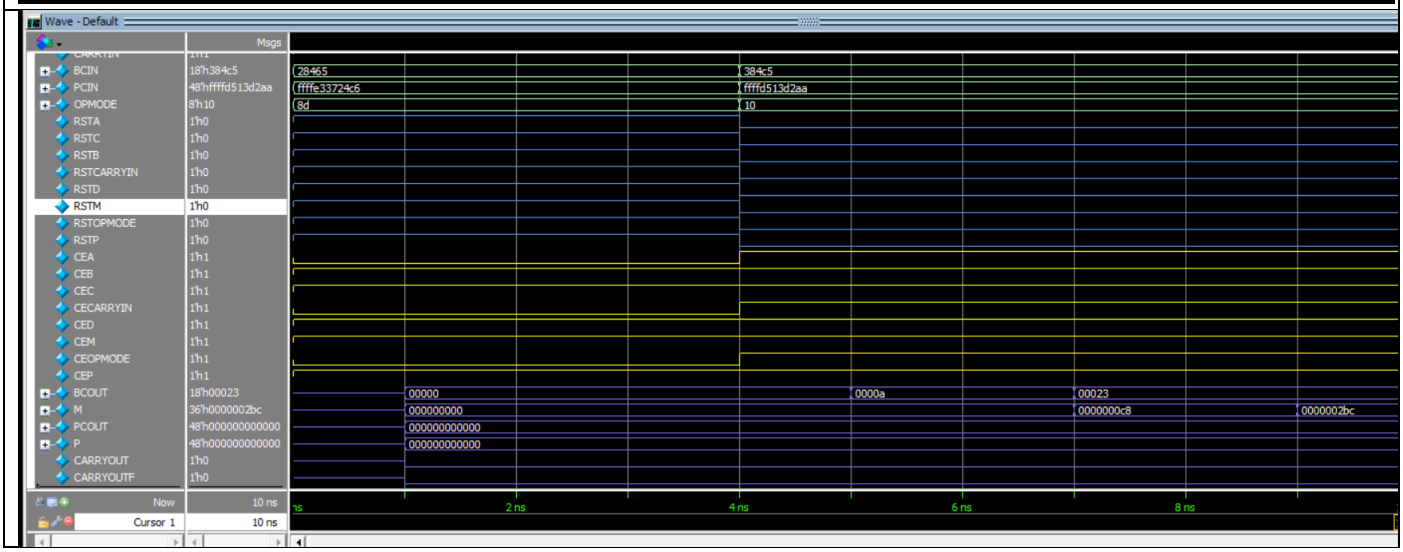




Path 2 :

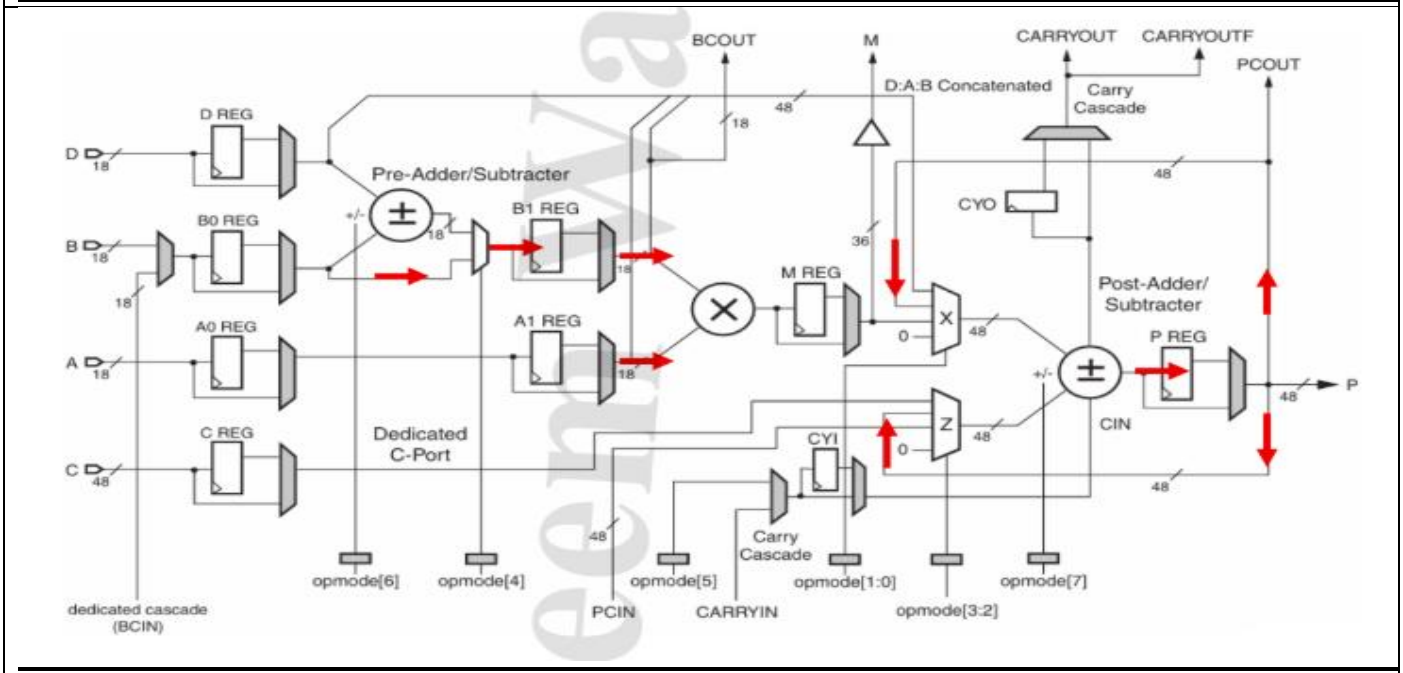
```
56      //verify DSP path 2
57      A = 20;
58      B = 10;
59      C = 350;
60      D = 25;
61      BCIN = $random;
62      PCIN = $random;
63      CARRYIN = $random;
64      OPMODE = 8'b00010000;
65      repeat(3) @(negedge CLK);
66      if(BCOUT != 'h23 || M != 'h2bc || P != 0 || PCOUT != 0
67          || CARRYOUTF != 0 || CARRYOUT != 0)
68          begin
69              $display("DSP Path 2 failed, BCOUT: %h, M: %h, P: %h, PCOUT: %h, CARRYOUTF: %b, CARRYOUT: %b",
70                      BCOUT, M, P, PCOUT, CARRYOUTF, CARRYOUT);
71              $stop;
72          end
73      $stop;
```

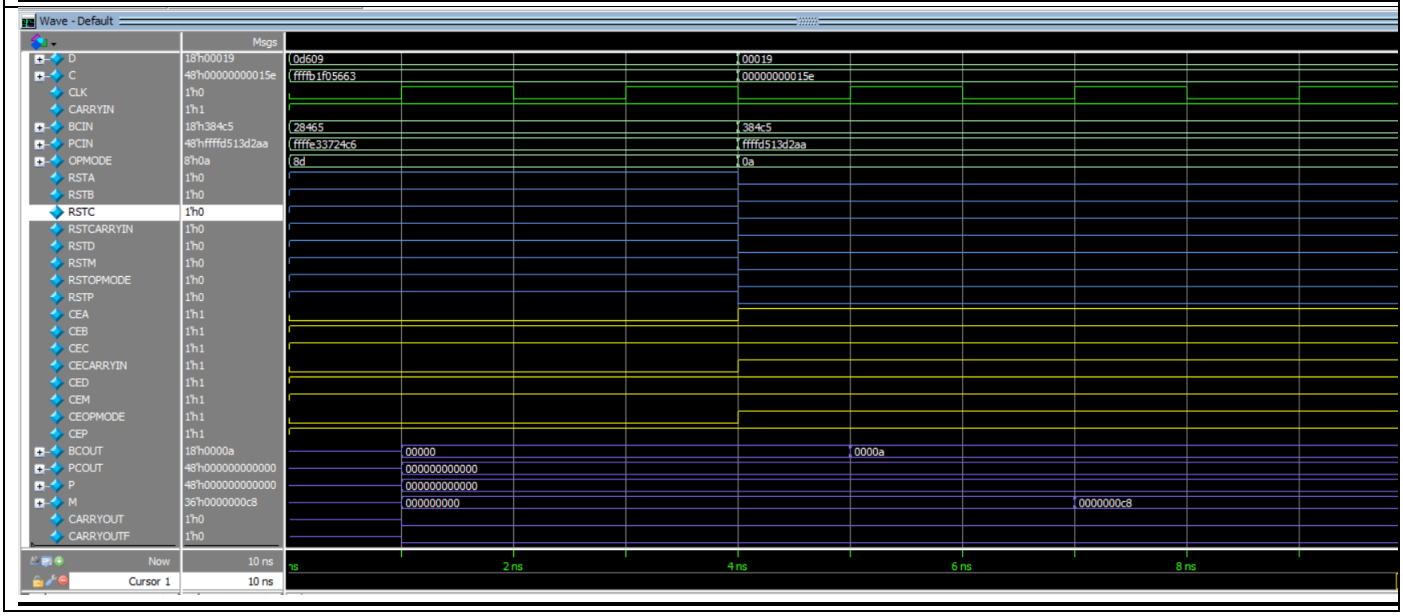




Path 3 :

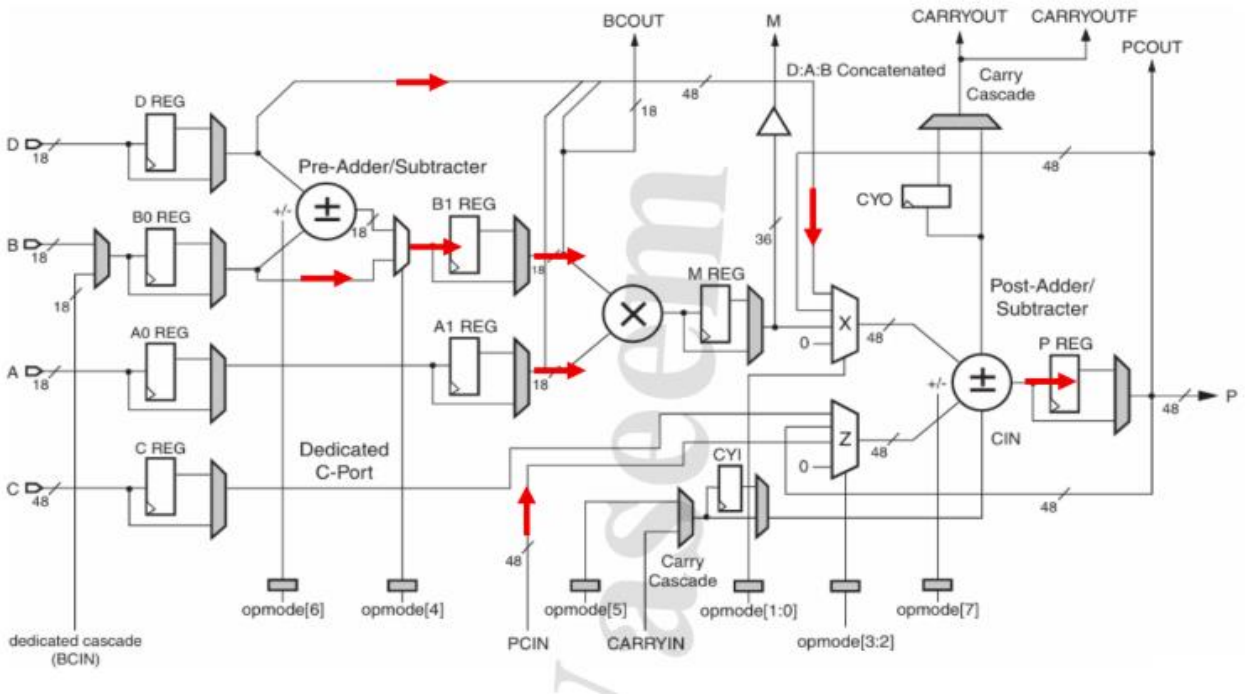
```
74      //verify DSP path 3
75      A = 20;
76      B = 10;
77      C = 350;
78      D = 25;
79      BCIN = $random;
80      PCIN = $random;
81      CARRYIN = $random;
82      OPMODE = 8'b00001010;
83      repeat(3) @(negedge CLK);
84      if(BCOUT != 'ha || M != 'hc8 || P != PCOUT || CARRYOUTF != CARRYOUT)
85          begin
86              $display("DSP Path 3 failed, BCOUT: %h, M: %h, P: %h, PCOUT: %h, CARRYOUTF: %b, CARRYOUT: %b",
87                      BCOUT, M, P, PCOUT, CARRYOUTF, CARRYOUT);
88              $stop;
89          end
90      $stop;
```

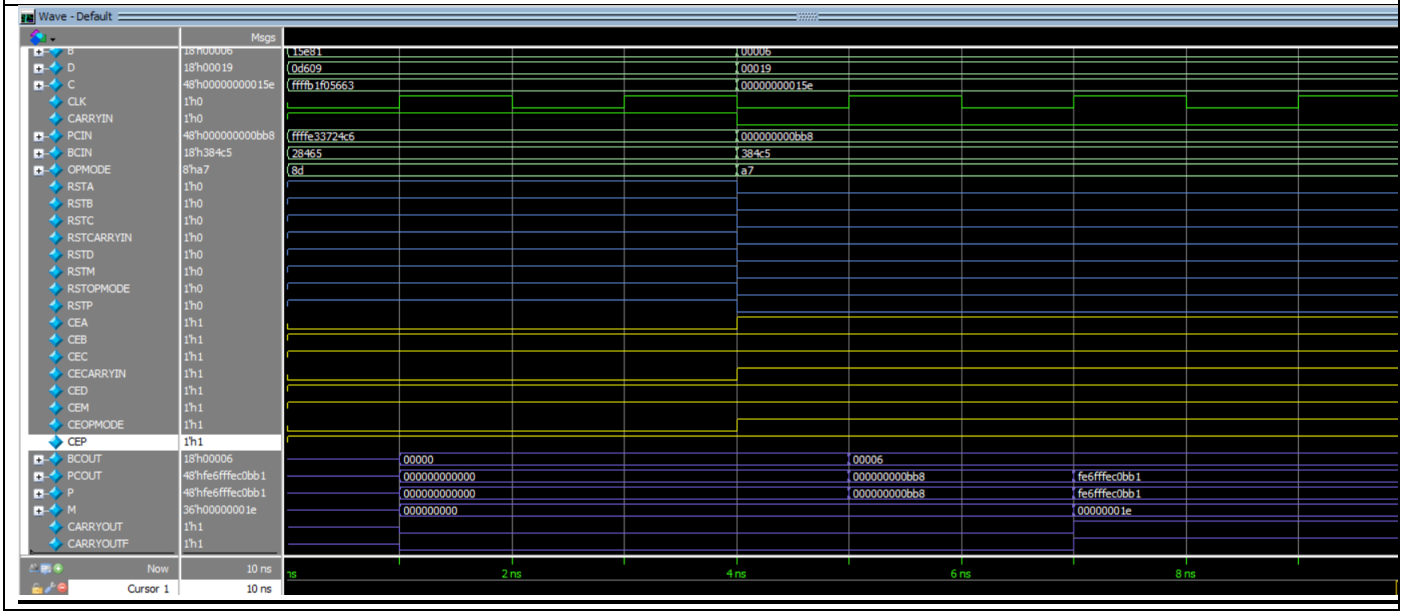




Path 4 :

```
91      //verify DSP path 4
92      A = 5;
93      B = 6;
94      C = 350;
95      D = 25;
96      PCIN = 3000;
97      BCIN = $random;
98      CARRYIN = $random;
99      OPMODE = 8'b10100111;
100     repeat(3) @(negedge CLK);
101     if(BCOUT != 'h6 || M != 'h1e || P != 'hfe6fffec0bb1 || PCOUT != 'hfe6fffec0bb1
102        || CARRYOUTF != 1 || CARRYOUT != 1)
103        begin
104            $display("DSP Path 4 failed, BCOUT: %h, M: %h, P: %h, PCOUT: %h, CARRYOUTF: %b, CARRYOUT: %b",
105                    BCOUT, M, P, PCOUT, CARRYOUTF, CARRYOUT);
106            $stop;
107        end
108     $stop;
109   end
110 endmodule
```





3) Do File :

run.do

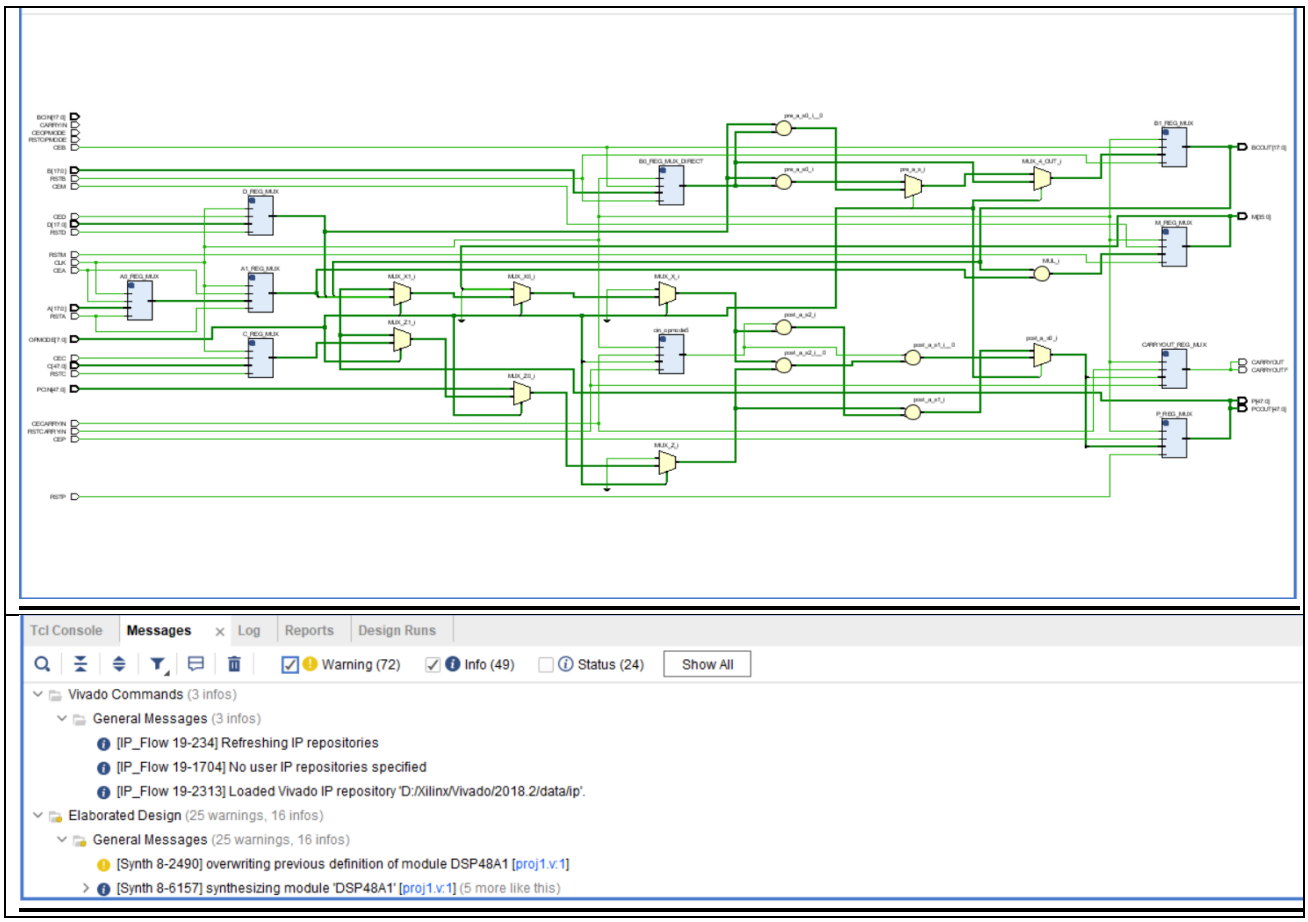
```
vlib work
vlog reg_mux.v proj1.v DSP_tb.v
vsim -voptargs=+acc work.DSP_tb
add wave *
run -all
#quit -sim
```

4) Constraint File :

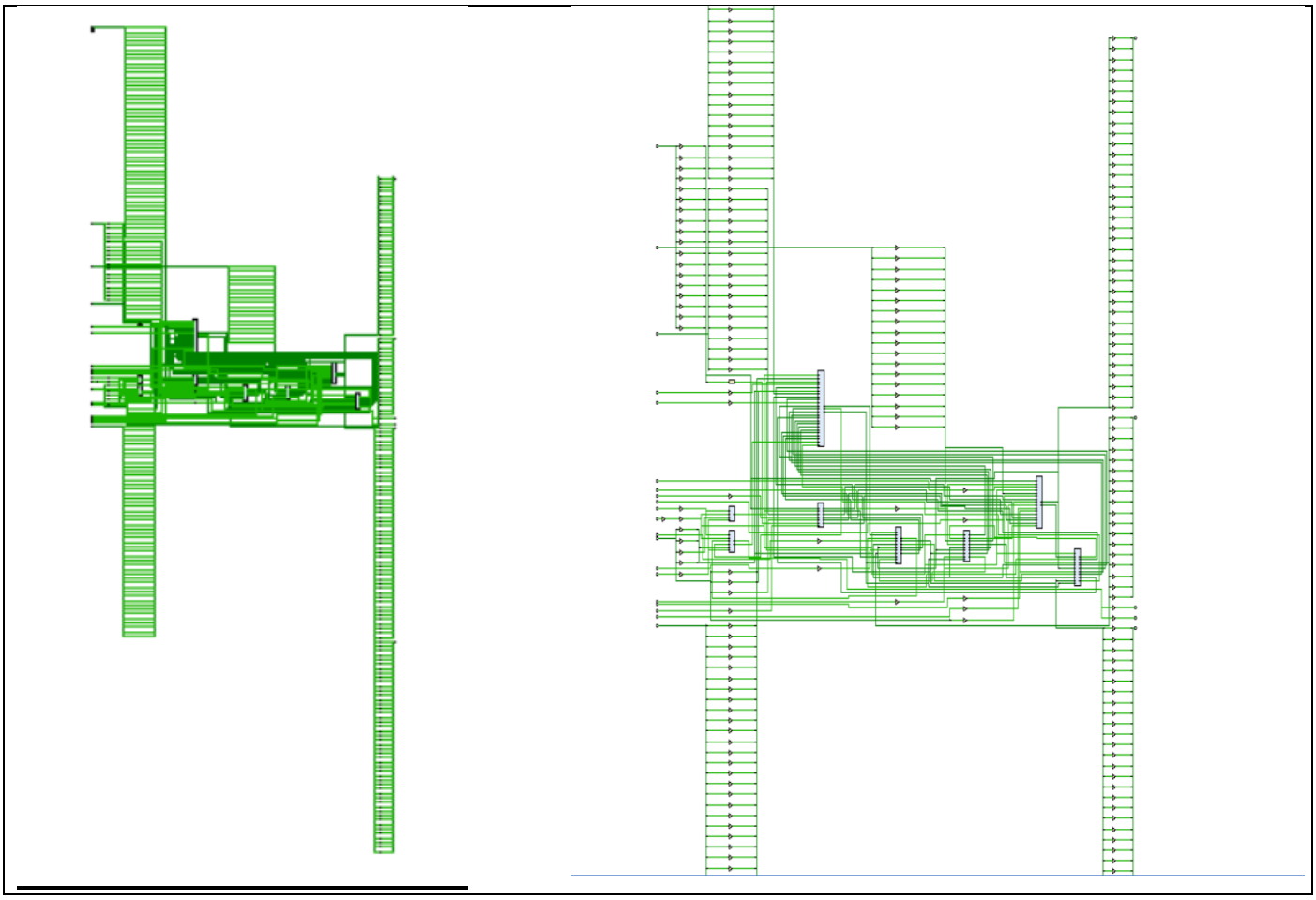
Constraints_basys3.xdc

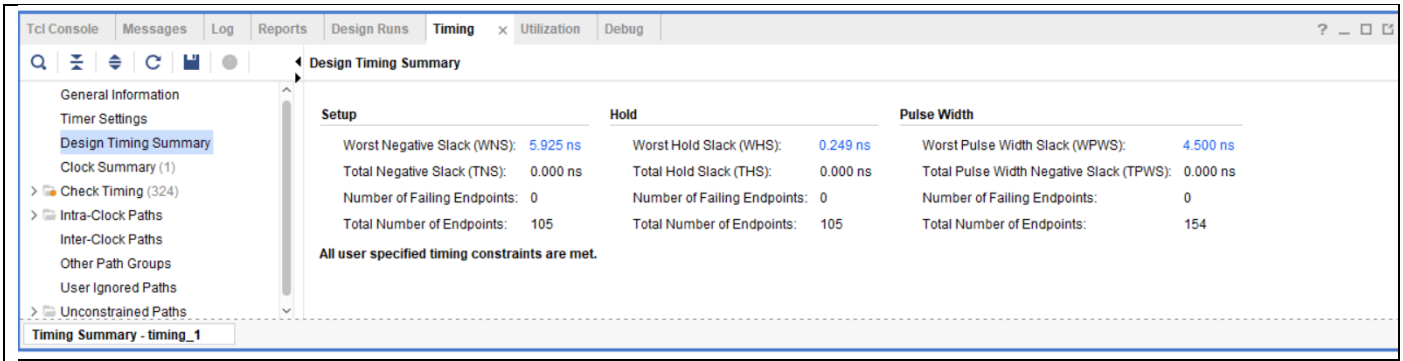
```
## This file is a general .xdc for the Basys3 rev B board
## To use it in a project:
## - uncomment the lines corresponding to used pins
## - rename the used ports (in each line, after get_ports) according to the top level signal names in the project

## Clock signal
set_property -dict { PACKAGE_PIN W5   IOSTANDARD LVCMOS33 } [get_ports CLK]
create_clock -add -name sys_clk_pin -period 10.00 -waveform {0 5} [get_ports CLK]


## Switches
#set_property -dict { PACKAGE_PIN V17   IOSTANDARD LVCMOS33 } [get_ports {sw[0]}]
#set_property -dict { PACKAGE_PIN V16   IOSTANDARD LVCMOS33 } [get_ports {sw[1]}]
#set_property -dict { PACKAGE_PIN W16   IOSTANDARD LVCMOS33 } [get_ports {sw[2]}]
#set_property -dict { PACKAGE_PIN W17   IOSTANDARD LVCMOS33 } [get_ports {sw[3]}]
#set_property -dict { PACKAGE_PIN W15   IOSTANDARD LVCMOS33 } [get_ports {sw[4]}]
#set_property -dict { PACKAGE_PIN V15   IOSTANDARD LVCMOS33 } [get_ports {sw[5]}]
#set_property -dict { PACKAGE_PIN W14   IOSTANDARD LVCMOS33 } [get_ports {sw[6]}]
#set_property -dict { PACKAGE_PIN W13   IOSTANDARD LVCMOS33 } [get_ports {sw[7]}]
#set_property -dict { PACKAGE_PIN V2    IOSTANDARD LVCMOS33 } [get_ports {sw[8]}]
#set_property -dict { PACKAGE_PIN T3    IOSTANDARD LVCMOS33 } [get_ports {sw[9]}]
#set_property -dict { PACKAGE_PIN T2    IOSTANDARD LVCMOS33 } [get_ports {sw[10]}]
#set_property -dict { PACKAGE_PIN R3    IOSTANDARD LVCMOS33 } [get_ports {sw[11]}]
#set_property -dict { PACKAGE_PIN W2    IOSTANDARD LVCMOS33 } [get_ports {sw[12]}]
#set_property -dict { PACKAGE_PIN U1    IOSTANDARD LVCMOS33 } [get_ports {sw[13]}]
#set_property -dict { PACKAGE_PIN T1    IOSTANDARD LVCMOS33 } [get_ports {sw[14]}]
#set_property -dict { PACKAGE_PIN R2    IOSTANDARD LVCMOS33 } [get_ports {sw[15]}]


## LEDs
#set_property -dict { PACKAGE_PIN U16   IOSTANDARD LVCMOS33 } [get_ports {led[0]}]
#set_property -dict { PACKAGE_PIN E19   IOSTANDARD LVCMOS33 } [get_ports {led[1]}]
#set_property -dict { PACKAGE_PIN U19   IOSTANDARD LVCMOS33 } [get_ports {led[2]}]
#set_property -dict { PACKAGE_PIN V19   IOSTANDARD LVCMOS33 } [get_ports {led[3]}]
#set_property -dict { PACKAGE_PIN W18   IOSTANDARD LVCMOS33 } [get_ports {led[4]}]
#set_property -dict { PACKAGE_PIN U15   IOSTANDARD LVCMOS33 } [get_ports {led[5]}]
#set_property -dict { PACKAGE_PIN U14   IOSTANDARD LVCMOS33 } [get_ports {led[6]}]
#set_property -dict { PACKAGE_PIN V14   IOSTANDARD LVCMOS33 } [get_ports {led[7]}]
```

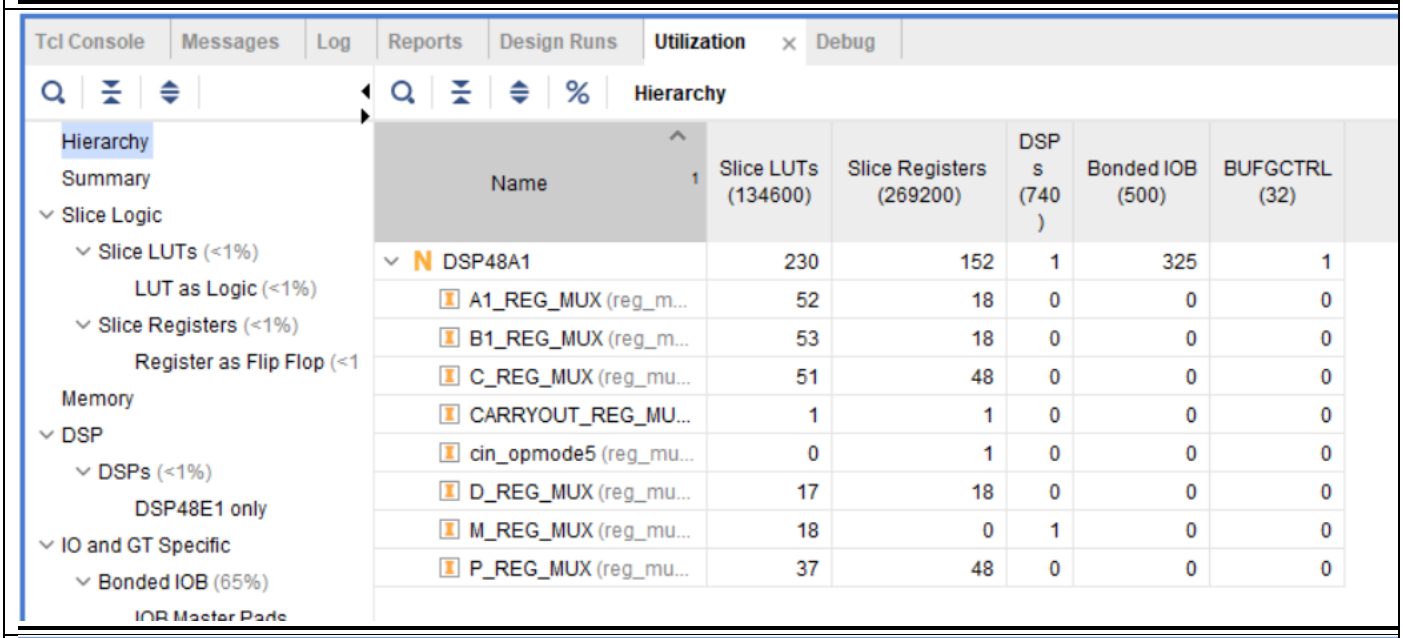
Elaboration :

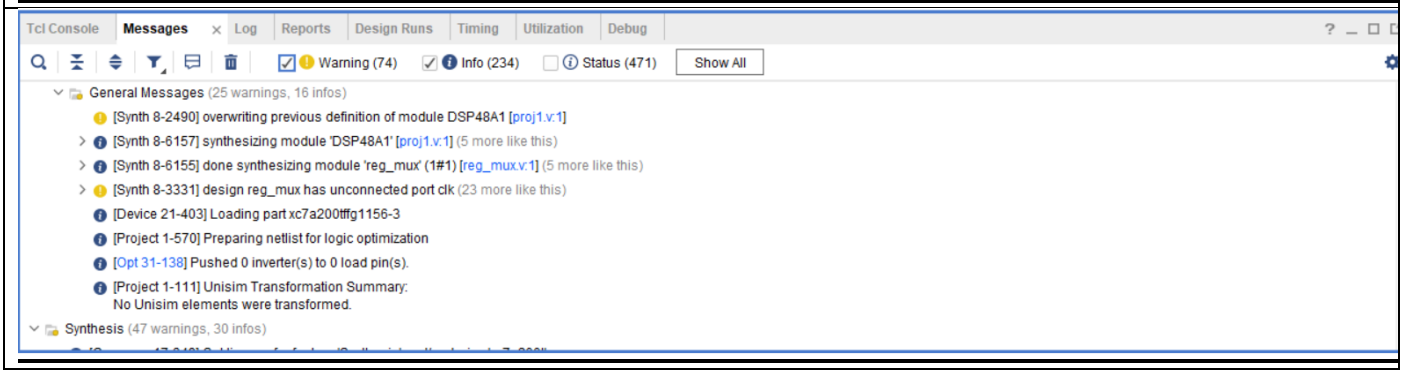


5) Synthesis :

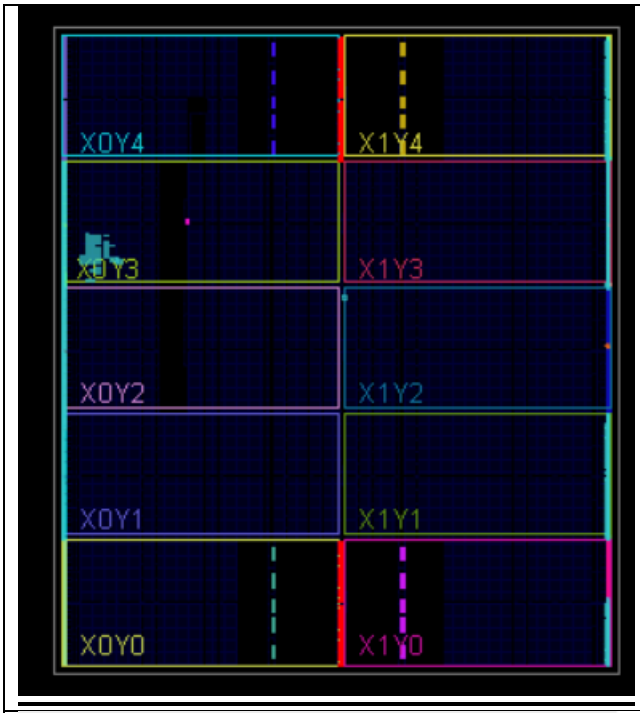



Tcl Console
Messages
Log
Reports
Design Runs
Timing
Utilization
Debug
Design Timing Summary
General Information
Timer Settings
Design Timing Summary
Clock Summary (1)
Check Timing (324)
Intra-Clock Paths
Inter-Clock Paths
Other Path Groups
User Ignored Paths
Unconstrained Paths
Setup
Worst Negative Slack (WNS): 5.925 ns
Total Negative Slack (TNS): 0.000 ns
Number of Failing Endpoints: 0
Total Number of Endpoints: 105
Hold
Worst Hold Slack (WHS): 0.249 ns
Total Hold Slack (THS): 0.000 ns
Number of Failing Endpoints: 0
Total Number of Endpoints: 105
Pulse Width
Worst Pulse Width Slack (WPWS): 4.500 ns
Total Pulse Width Negative Slack (TPWS): 0.000 ns
Number of Failing Endpoints: 0
Total Number of Endpoints: 154
All user specified timing constraints are met.
Timing Summary - timing_1


Tcl Console
Messages
Log
Reports
Design Runs
Utilization
Debug
Hierarchy
Hierarchy
Summary
Slice Logic
Slice LUTs (<1%)
LUT as Logic (<1%)
Slice Registers (<1%)
Register as Flip Flop (<1
Memory
DSP
DSPs (<1%)
DSP48E1 only
IO and GT Specific
Bonded IOB (65%)
IOB Master Pads
Name
Slice LUTs (134600)
Slice Registers (269200)
DSPs (740)
Bonded IOB (500)
BUFGCTRL (32)
DSP48A1 230 152 1 325 1
A1_REG_MUX (reg_m... 52 18 0 0 0
B1_REG_MUX (reg_m... 53 18 0 0 0
C_REG_MUX (reg_mu... 51 48 0 0 0
CARRYOUT_REG_MU... 1 1 0 0 0
cin_opmode5 (reg_mu... 0 1 0 0 0
D_REG_MUX (reg_mu... 17 18 0 0 0
M_REG_MUX (reg_mu... 18 0 1 0 0
P_REG_MUX (reg_mu... 37 48 0 0 0


Tcl Console
Messages
Log
Reports
Design Runs
Timing
Utilization
Debug
Warning (74)
Info (234)
Status (471)
Show All
General Messages (25 warnings, 16 infos)
[Synth 8-2490] overwriting previous definition of module DSP48A1 [proj1.v:1]
[Synth 8-6157] synthesizing module 'DSP48A1' [proj1.v:1] (5 more like this)
[Synth 8-6155] done synthesizing module 'reg_mux' (1#1) [reg_mux.v:1] (5 more like this)
[Synth 8-3331] design reg_mux has unconnected port clk (23 more like this)
[Device 21-403] Loading part xc7a200tffg1156-3
[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.
Synthesis (47 warnings, 30 infos)

6) Implementation :



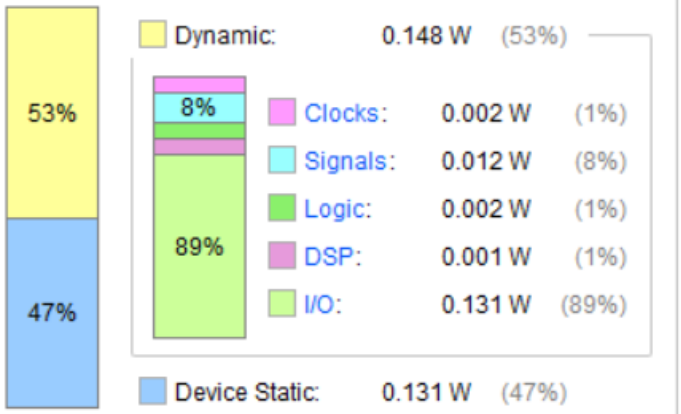
Summary

Power analysis from Implemented netlist. Activity derived from constraints files, simulation files or vectorless analysis.

Total On-Chip Power:	**0.279 W**
Design Power Budget:	**Not Specified**
Power Budget Margin:	**N/A**
Junction Temperature:	**25.4°C**
Thermal Margin:	74.6°C (50.8 W)
Effective ϑJA:	1.5°C/W
Power supplied to off-chip devices:	0 W
Confidence level:	Low

Launch Power Constraint Advisor to find and fix invalid switching activity

On-Chip Power



Reports | Design Runs | DRC | Methodology | Power | Timing | **Utilization** ×

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)
DSP48A1	229	171	93	229	50	1	325	1
A1_REG_MUX (reg_m...	52	18	21	52	0	0	0	0
B1_REG_MUX (reg_m...	53	36	28	53	1	0	0	0
C_REG_MUX (reg_mu...	51	48	45	51	0	0	0	0
CARRYOUT_REG_MU...	1	2	2	1	1	0	0	0
cin_opmode5 (reg_mu...	0	1	1	0	0	0	0	0
D_REG_MUX (reg_mu...	17	18	10	17	0	0	0	0
M_REG_MUX (reg_mu...	18	0	8	18	0	1	0	0
P_REG_MUX (reg_mu...	37	48	20	37	12	0	0	0

ts | Design Runs | DRC | Methodology | Power | **Timing** × | Utilization

Design Timing Summary

Setup		Hold		Pulse Width	
Worst Negative Slack (WNS):	4.864 ns	Worst Hold Slack (WHS):	0.384 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 (TPWS):	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.

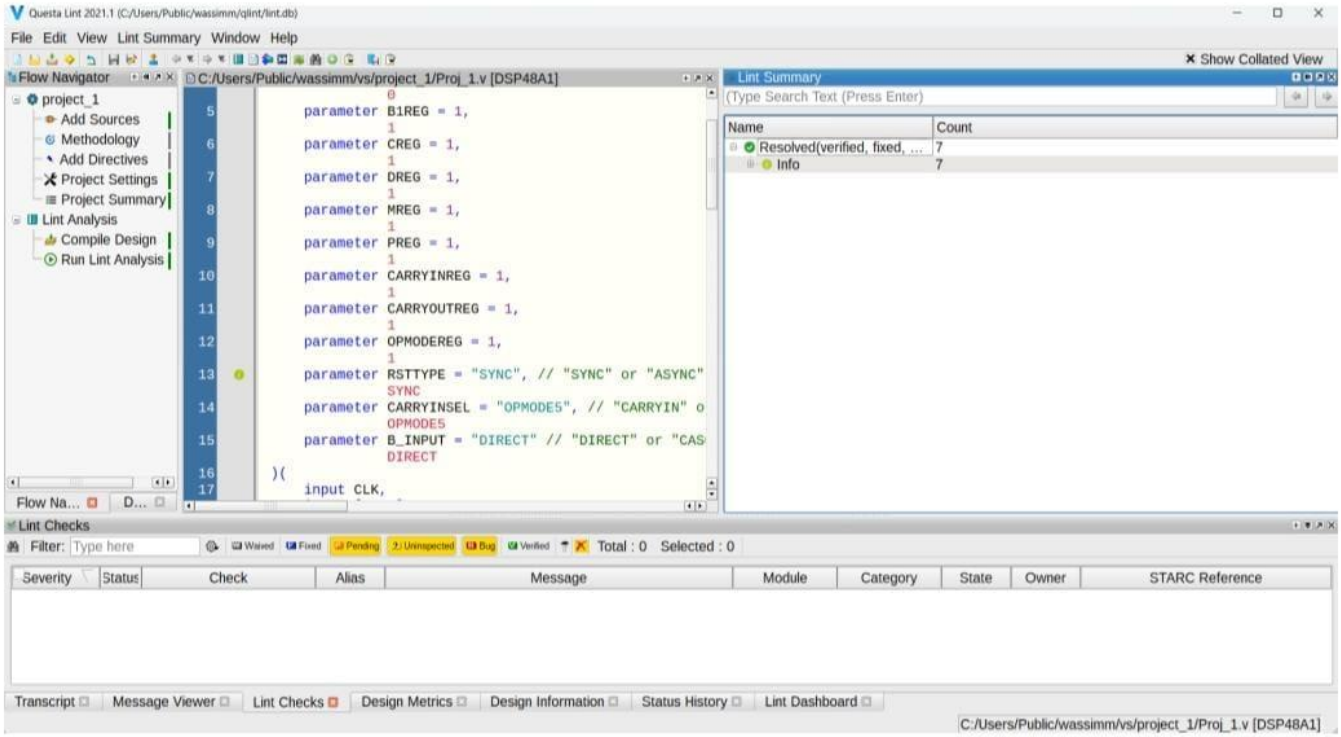
Tcl Console | **Messages** × | Log | Reports | Design Runs | DRC | Methodology | Power | Timing | Utilization

Warning (74) | Info (245) | Status (476) | Show All

General Messages (25 warnings, 16 infos)

- [Synth 8-2490] overwriting previous definition of module DSP48A1 [proj1.v:1]
- [Synth 8-6157] synthesizing module 'DSP48A1' [proj1.v:1] (5 more like this)
- [Synth 8-6155] done synthesizing module 'reg_mux' (1#1) [reg_mux.v:1] (5 more like this)
- [Synth 8-3331] design reg_mux has unconnected port clk (23 more like this)
- [Device 21-403] Loading part xc7a200tffg1156-3
- [Project 1-570] Preparing netlist for logic optimization
- [Opt 31-138] Pushed 0 inverter(s) to 0 load pin(s).

7) Linting :


Questa Lint 2021.1 (C:/Users/Public/wassimm/qlint/lint.db)
File Edit View Lint Summary Window Help
Show Collated View
Flow Navigator
project_1
Add Sources
Methodology
Add Directives
Project Settings
Project Summary
Lint Analysis
Compile Design
Run Lint Analysis
C:/Users/Public/wassimm/vs/project_1/Proj_1.v [DSP48A1]
parameter B1REG = 1,
parameter CREG = 1,
parameter DREG = 1,
parameter MREG = 1,
parameter PREG = 1,
parameter CARRYINREG = 1,
parameter CARRYOUTREG = 1,
parameter OPMODEREG = 1,
parameter RSTTYPE = "SYNC", // "SYNC" or "ASYNC"
parameter CARRYINSEL = "OPMODE5", // "CARRYIN" o
parameter B_INPUT = "DIRECT" // "DIRECT" or "CAS
)(
input CLK,
Lint Summary
(Type Search Text (Press Enter))
Name
Count
Resolved(verified, fixed, ...
7
Info
7
Lint Checks
Filter: Type here
Waived
Fixed
Pending
Uninspected
Bug
Verified
Total : 0
Selected : 0
Severity
Status
Check
Alias
Message
Module
Category
State
Owner
STARC Reference
Transcript
Message Viewer
Lint Checks
Design Metrics
Design Information
Status History
Lint Dashboard
C:/Users/Public/wassimm/vs/project_1/Proj_1.v [DSP48A1]