

Name: Mohamed Gamal Mansour

Project1

1-RTL Code:

```
1  module block(clk,enable_clk, reset,in, out);
2  parameter RSTTYPE = "SYNC";
3  parameter mux_sel = 1;
4  parameter NUMBER_BITS=18;
5  input clk, enable_clk, reset;
6  input [NUMBER_BITS-1:0] in;
7  output reg [NUMBER_BITS-1:0] out;
8  reg [NUMBER_BITS-1:0] in_reg_async;
9  reg [NUMBER_BITS-1:0] in_reg_sync;
10 reg [NUMBER_BITS-1:0] in_reg;
11 always @(posedge clk) begin
12     if (enable_clk) begin
13         if (reset) begin
14             in_reg_sync <=0;
15         end
16         else begin
17             in_reg_sync<=in;
18         end
19     end
20 end
21 always @(posedge clk or posedge reset) begin
22     if (reset) begin
23         in_reg_async <=0;
24     end
25     else if (enable_clk) begin
26         in_reg_async<=in;
27     end
28 end
29 always@(*) begin
30     if(RSTTYPE == "SYNC" ) begin
31         in_reg= in_reg_sync;
32     end
33     else begin
34         in_reg=in_reg_async;
35     end
36     if(mux_sel) begin
37         out= in_reg;
38     end
39     else begin
40         out = in;
41     end
42 end
43 endmodule
```

```

1  module DSP48A1(A,B,D,C,CLK,CARRYIN,OPMODE,BCIN,RSTA,RSTB,RSTM,RSTP,RSTC,RSTD,RSTCARRYIN,RSTOPMODE,
2     CEA,CEB,CEM,CEP,CEC,CED,CECARRYIN,CEOPMODE,PCIN,BCOUT,PCOUT,P,M,CARRYOUT,CARRYOUTF);
3  parameter A0REG=0, A1REG=1, B0REG=0, B1REG=1;
4  parameter CREG=1, DREG=1, MREG=1, PREG=1, CARRYINREG=1, CARRYOUTREG=1, OPMODEREG=1;
5  parameter CARRYINSEL= "OPMODE5";
6  parameter B_INPUT = "DIRECT";
7  parameter RSTTYPE="SYNC";
8  input [17:0] A,B,D,BCIN;
9  input [47:0] C,PCIN;
10 input [7:0] OPMODE;
11 input CLK,CARRYIN,RSTA,RSTB,RSTM,RSTP,RSTC,RSTD,RSTCARRYIN,RSTOPMODE,CEA,CEB,CEM,CEP,CEC,CED,CECARRYIN,CEOPMODE;
12 output [17:0] BCOUT;
13 output [47:0] PCOUT, P;
14 output [35:0] M;
15 output CARRYOUT;
16 output CARRYOUTF;
17 reg [17:0] B_OR_BCIN;
18 wire [17:0] D_mux_out,B0_mux_out,A0_mux_out;
19 wire [47:0] C_mux_out;
20 wire [17:0] B1_mux_out,A1_mux_out;
21 wire [7:0] OPMODE_MUX_OUT;
22 reg [35:0] MULTIPLIER_RESULT;
23 reg CARRYCASCADE_MUX, carry_out_postadder;
24 wire CIN;
25 reg [47:0] POST_ADDER;
26 reg [17:0] pre_adder_result, PRE_ADDER_MUX;
27 reg [47:0] x_mux_out, z_mux_out;
28 block #(.RSTTYPE(RSTTYPE), .mux_sel(DREG),.NUMBER_BITS(18)) D_MUX(CLK,CED, RSTD,D, D_mux_out);
29 block #(.RSTTYPE(RSTTYPE), .mux_sel(B0REG),.NUMBER_BITS(18)) B0_MUX(CLK,CEB, RSTB,B_OR_BCIN, B0_mux_out);
30 block #(.RSTTYPE(RSTTYPE), .mux_sel(A0REG),.NUMBER_BITS(18)) A0_MUX(CLK,CEA, RSTA,A, A0_mux_out);
31 block #(.RSTTYPE(RSTTYPE), .mux_sel(CREG),.NUMBER_BITS(48)) C_MUX(CLK,CEC, RSTC,C, C_mux_out);
32 block #(.RSTTYPE(RSTTYPE), .mux_sel(B1REG),.NUMBER_BITS(18)) B1_MUX(CLK,CEB, RSTB,PRE_ADDER_MUX, B1_mux_out);
33 block #(.RSTTYPE(RSTTYPE), .mux_sel(A1REG),.NUMBER_BITS(18)) A1_MUX(CLK,CEA, RSTA,A0_mux_out, A1_mux_out);
34 block #(.RSTTYPE(RSTTYPE), .mux_sel(MREG),.NUMBER_BITS(36)) M_MUX(CLK,CEM, RSTM,MULTIPLIER_RESULT, M);
35
36 block #(.RSTTYPE(RSTTYPE), .mux_sel(CARRYINREG),.NUMBER_BITS(1)) CYI_MUX(CLK,CECARRYIN, RSTCARRYIN,CARRYCASCADE_MUX, CIN);
37 block #(.RSTTYPE(RSTTYPE), .mux_sel(CARRYOUTREG),.NUMBER_BITS(1)) CYO_MUX(CLK,CECARRYIN, RSTCARRYIN,carry_out_postadder, CARRYOUT);
38 block #(.RSTTYPE(RSTTYPE), .mux_sel(PREG),.NUMBER_BITS(48)) P_MUX(CLK,CEP, RSTP,POST_ADDER, P);
39 block #(.RSTTYPE(RSTTYPE), .mux_sel(OPMODEREG),.NUMBER_BITS(8)) OPMODE_MUX(CLK,CEOPMODE, RSTOPMODE,OPMODE, OPMODE_MUX_OUT);
40
41 always @ (*) begin
42     if (B_INPUT == "DIRECT")
43         B_OR_BCIN=B;
44     else if (B_INPUT == "CASCADE")
45         B_OR_BCIN=BCIN;
46     else
47         B_OR_BCIN=0;
48
49     if(OPMODE_MUX_OUT[6] == 0 )
50         pre_adder_result=D_mux_out + B0_mux_out;
51     else
52         pre_adder_result=D_mux_out - B0_mux_out;
53
54     if (OPMODE_MUX_OUT[4] == 0)
55         PRE_ADDER_MUX=B0_mux_out;
56     else

```

Line 42, Column 28

```

41  always @ (*) begin
42      if (B_INPUT == "DIRECT")
43          B_OR_BCIN=B;
44      else if (B_INPUT == "CASCADE")
45          B_OR_BCIN=BCIN;
46      else
47          B_OR_BCIN=0;
48
49      if(OPMODE_MUX_OUT[6] == 0 )
50          pre_adder_result=D_mux_out + B0_mux_out;
51      else
52          pre_adder_result=D_mux_out - B0_mux_out;
53
54      if (OPMODE_MUX_OUT[4] == 0)
55          PRE_ADDER_MUX=B0_mux_out;
56      else
57          PRE_ADDER_MUX=pre_adder_result;
58
59      MULTIPLIER_RESULT= B1_mux_out * A1_mux_out;
60
61      if( CARRYINSEL == "OPMODE5")
62          CARRYCASCADE_MUX= OPMODE_MUX_OUT[5];
63      else if( CARRYINSEL == "CARRYIN")
64          CARRYCASCADE_MUX= CARRYIN;
65      else
66          CARRYCASCADE_MUX= 0;
67
68      case (OPMODE_MUX_OUT[1:0])
69      0: x_mux_out=0;
70      1: x_mux_out={12'd0,M};
71      2: x_mux_out=P;
72      3: x_mux_out={D_mux_out[11:0],A1_mux_out[17:0],B1_mux_out[17:0]};
73      endcase
74
75      case(OPMODE_MUX_OUT[3:2])
76      0: z_mux_out=0;
77      1: z_mux_out=PCIN;
78      2: z_mux_out=P;
79      3: z_mux_out=C_mux_out;
80      endcase
81
82      if(OPMODE_MUX_OUT[7])
83          {carry_out_postadder,POST_ADDER}=z_mux_out - (x_mux_out+CIN);
84      else
85          {carry_out_postadder,POST_ADDER}=z_mux_out+x_mux_out+CIN;
86      end
87
88      assign PCOUT=P;
89      assign CARRYOUTF=CARRYOUT;
90      assign BCOUT=B1_mux_out;
91  endmodule

```

2-Testbench code:

```
1 module DSP48A1_TB();
2 parameter A0REG=0, A1REG=1, B0REG=0, B1REG=1;
3 parameter CREG=1, DREG=1, MREG=1, PREG=1, CARRYINREG=1, CARRYOUTREG=1, OPMODEREG=1;
4 parameter CARRYINSEL= "OPMODE5";
5 parameter B_INPUT = "DIRECT";
6 parameter RSTTYPE="SYNC";
7 reg [17:0] A,B,D,BCIN;
8 reg [47:0] C,PCIN;
9 reg [7:0] OPMODE;
10 reg CLK,CARRYIN,RSTA,RSTB,RSTM,RSTP,RSTC,RSTD,RSTCARRYIN,RSTOPMODE,CEA,CEB,CEM,CEP,CEC,CED,CECARRYIN,CEOPMODE;
11 wire [17:0] BCOUT;
12 wire [47:0] PCOUT, P;
13 wire [35:0] M;
14 wire CARRYOUT;
15 wire CARRYOUTF;
16 DSP48A1 #(A0REG, A1REG, B0REG, B1REG, CREG, DREG, MREG, PREG, CARRYINREG, CARRYOUTREG, OPMODEREG) dut(A,B,D,C,CLK,CARRYIN,OPMODE,BCIN,RSTA,
17 RSTB,RSTM,RSTP,RSTC,RSTD,RSTCARRYIN,RSTOPMODE, CEA,CEB,CEM,CEP,CEC,CED,CECARRYIN,CEOPMODE,PCIN,BCOUT,PCOUT,P,M,CARRYOUT,CARRYOUTF);
18 initial begin
19     CLK=0;
20     forever
21     #1 CLK=~CLK;
22 end
23
24 initial begin
25     CEA=1; CEB=1; CEC=1; CECARRYIN=1; CED=1; CEM=1; CEOPMODE=1; CEP=1;
26     RSTA=1; RSTB=1; RSTC=1; RSTCARRYIN=1; RSTD=1; RSTM=1; RSTOPMODE=1; RSTP=1;
27     D=10; B=10; OPMODE[6]=0; OPMODE[4]=1; A=2; OPMODE[1:0]=1; C=20; OPMODE[7]=0; OPMODE[5]=1; OPMODE[3:2]=3; BCIN=0; PCIN=0; CARRYIN=0;
28     repeat(4)
29     @(negedge CLK);
30     RSTA=0; RSTB=0; RSTC=0; RSTCARRYIN=0; RSTD=0; RSTM=0; RSTOPMODE=0; RSTP=0;
31     repeat(4)
32     @(negedge CLK);
33     if(BCOUT != 20 || P != 61 || CARRYOUT != 0 || M != 40 || PCOUT != 61 || CARRYOUTF !=0) begin
34         $display("There is an error");
35         $stop;
36     end
37
38     D=30; B=10; OPMODE[6]=1; OPMODE[4]=1; A=2; OPMODE[1:0]=1; C=20; OPMODE[7]=0; OPMODE[5]=1; OPMODE[3:2]=3;
39     repeat(4)
40     @(negedge CLK);
41     if(BCOUT != 20 || P != 61 || CARRYOUT != 0 || M != 40 || PCOUT != 61 || CARRYOUTF !=0) begin
42         $display("There is an error");
43         $stop;
44     end
45
46     D=10; B=10; OPMODE[6]=0; OPMODE[4]=1; A=1; OPMODE[1:0]=1; C=40; OPMODE[7]=1; OPMODE[5]=1; OPMODE[3:2]=3;
47     repeat(4)
48     @(negedge CLK);
49     if(BCOUT != 20 || P != 19 || CARRYOUT != 0 || M != 20 || PCOUT != 19 || CARRYOUTF !=0 ) begin
50         $display("There is an error");
51         $stop;
52     end
53
54     D=30; B=10; OPMODE[6]=0; OPMODE[4]=1; A=2; OPMODE[1:0]=0; C=20; OPMODE[7]=0; OPMODE[5]=1; OPMODE[3:2]=0;
55     repeat(4)
56     @(negedge CLK);
```

Line 1, Column 21

```

53
54 D=30; B=10; OPMODE[6]=0; OPMODE[4]=1; A=2; OPMODE[1:0]=0; C=20; OPMODE[7]=0; OPMODE[5]=1; OPMODE[3:2]=0;
55 repeat(4)
56 @(negedge CLK);
57 if(BCOUT != 40 || P != 1 || CARRYOUT != 0 || M != 80 || PCOUT != 1 || CARRYOUTF !=0) begin
58     $display("There is an error");
59     $stop;
60 end
61
62 D=30; B=10; OPMODE[6]=0; OPMODE[4]=1; A=2; OPMODE[1:0]=2; C=20; OPMODE[7]=0; OPMODE[5]=1; OPMODE[3:2]=0;
63 repeat(4)
64 @(negedge CLK);
65 if(BCOUT != 40 || P != 4 || CARRYOUT != 0 || M != 80 || PCOUT != 4 || CARRYOUTF !=0) begin
66     $display("There is an error");
67     $stop;
68 end
69
70 D=0; B=10; OPMODE[6]=0; OPMODE[4]=0; A=4; OPMODE[1:0]=3; C=20; OPMODE[7]=0; OPMODE[5]=0; OPMODE[3:2]=0;
71 repeat(4)
72 @(negedge CLK);
73 if(BCOUT != 10 || P != 48'h00000010000A || CARRYOUT != 0 || M != 40 || PCOUT != 48'h00000010000A || CARRYOUTF !=0) begin
74     $display("There is an error");
75     $stop;
76 end
77 D=0; B=10; OPMODE[6]=0; OPMODE[4]=0; A=0; OPMODE[1:0]=3; C=20; OPMODE[7]=0; OPMODE[5]=0; OPMODE[3:2]=1; PCIN=50;
78 repeat(4)
79 @(negedge CLK);
80 if(BCOUT != 10 || P != 60 || CARRYOUT != 0 || M != 0 || PCOUT != 60 || CARRYOUTF !=0) begin
81     $display("There is an error");
82     $stop;
83 end
84 D=50; B=10; OPMODE[6]=0; OPMODE[4]=0; A=1; OPMODE[1:0]=1; C=20; OPMODE[7]=0; OPMODE[5]=0; OPMODE[3:2]=2; PCIN=50;
85 repeat(3)
86 @(negedge CLK);
87 if(BCOUT != 10 || P != 70 || CARRYOUT != 0 || M != 10 || PCOUT != 70 || CARRYOUTF !=0) begin
88     $display("There is an error");
89     $stop;
90 end
91 D=0; B=10; OPMODE[6]=0; OPMODE[4]=0; A=0; OPMODE[1:0]=0; C=48'hFFFFFFFFF; OPMODE[7]=0; OPMODE[5]=1; OPMODE[3:2]=3; PCIN=50;
92 repeat(3)
93 @(negedge CLK);
94 if(BCOUT != 10 || P != 0 || CARRYOUT != 1 || M != 0 || PCOUT != 0 || CARRYOUTF !=1) begin
95     $display("There is an error");
96     $stop;
97 end
98 $stop;
99 end
100 initial begin
101     $monitor("A=%d,B=%d,DB=%d,CB=%d,CARRYIN=%d,OPMODE=%d,BCIN=%d,RSTA=%d,RSTB=%d,RSTM=%d,RSTP=%d,RSTC=%d,RSTD=%d,RSTCARRYIN=%d,
102     RSTOPMODE=%d,CEA=%d,CEB=%d,CEM=%d,CEP=%d,CEC=%d,CED=%d,CECARRYIN=%d,CEOPMODE=%d,PCIN=%d,BCOUT=%d,PCOUT=%d,P=%d,M=%d,
103     CARRYOUT=%d,CARRYOUTF=%d",A,B,D,C,CARRYIN,OPMODE,BCIN,RSTA,RSTB,RSTM,RSTP,RSTC,RSTD,RSTCARRYIN,RSTOPMODE,
104     CEA,CEB,CEM,CEP,CEC,CED,CECARRYIN,CEOPMODE,PCIN,BCOUT,PCOUT,P,M,CARRYOUT,CARRYOUTF);
105 end
106 endmodule

```

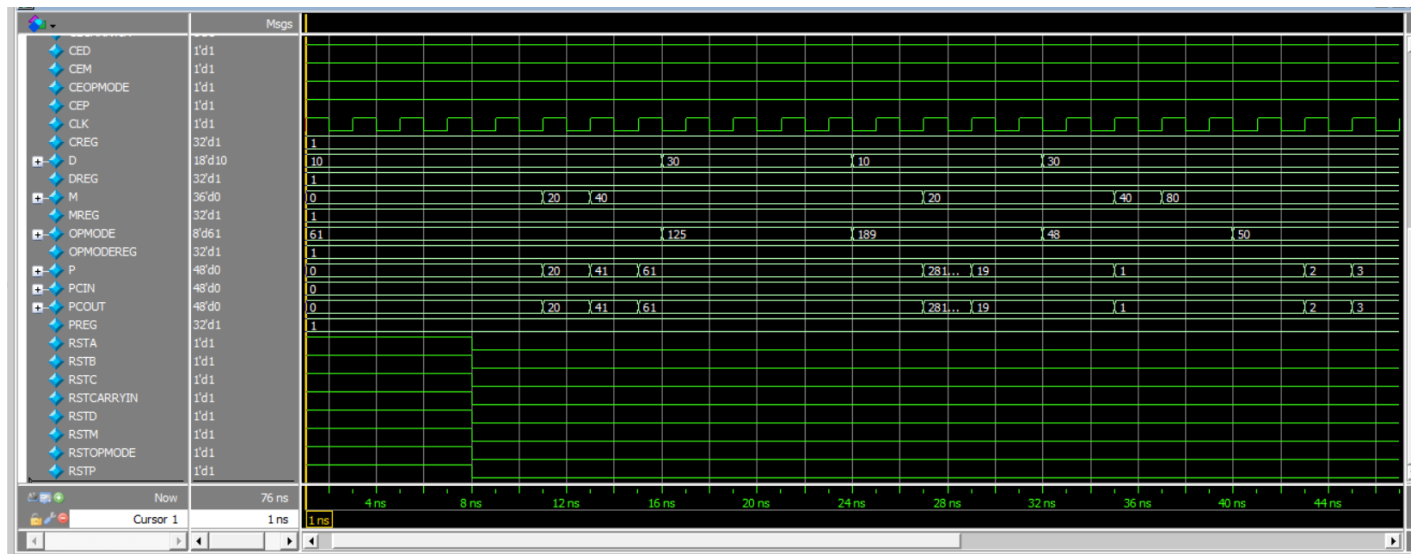
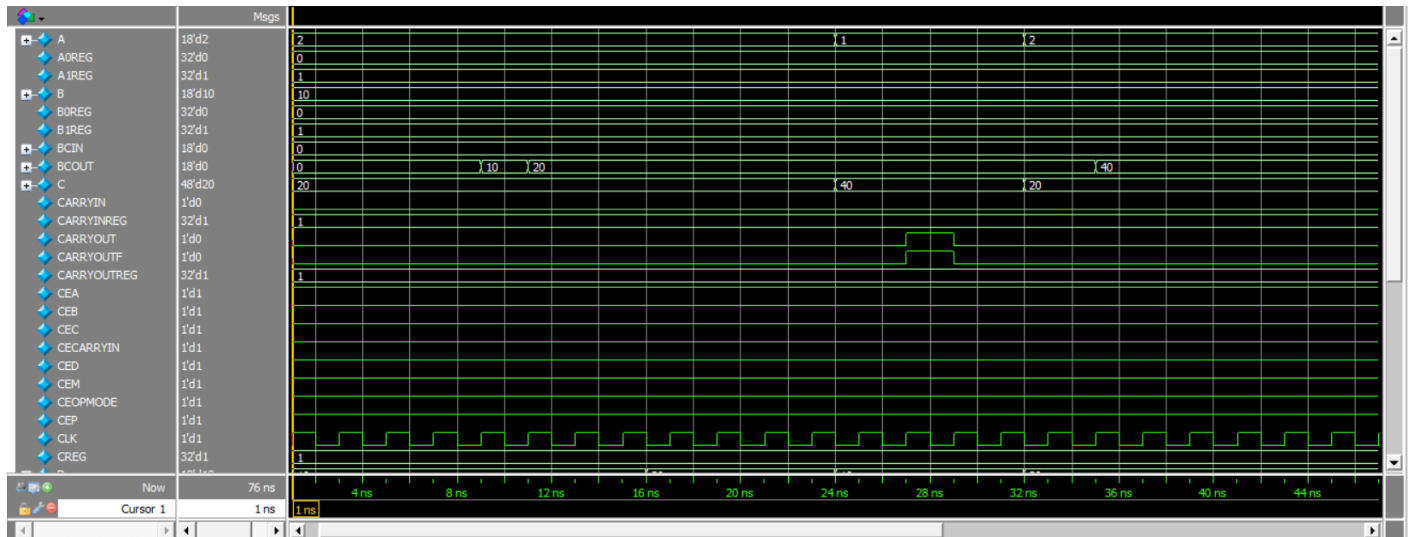
3-Do file:

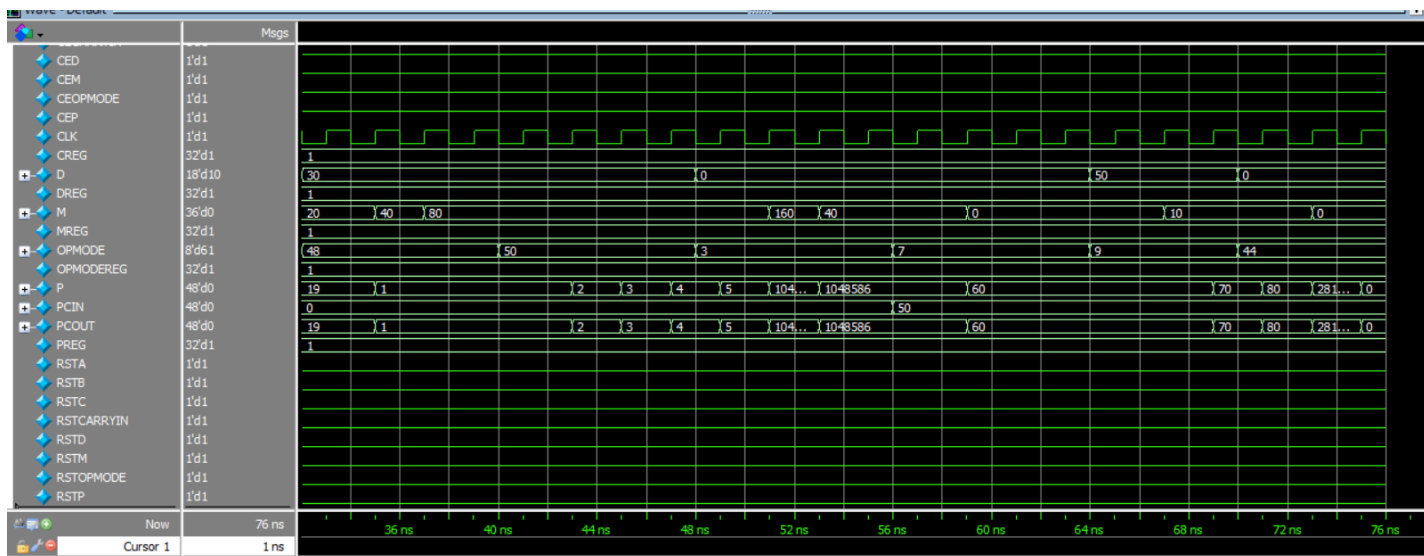
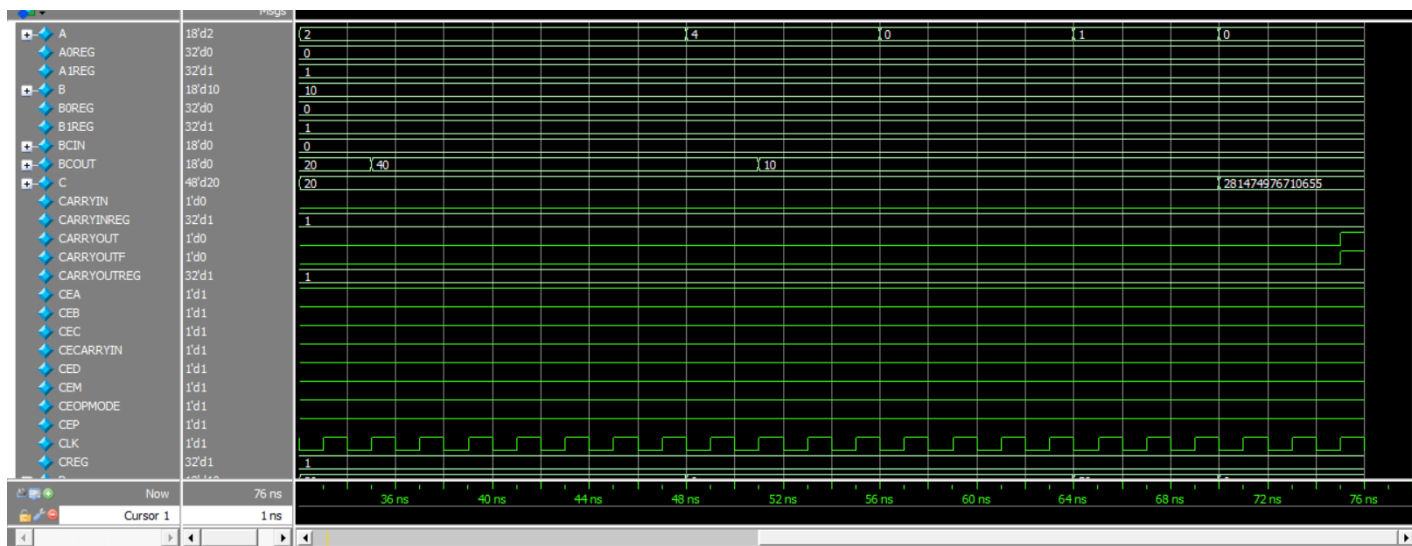
```

1 vlib work
2 vlog DSP48A1.v DSP48A1_tb.v block.v
3 vsim -voptargs=+acc work.DSP48A1_TB
4 add wave *
5 run -all
6 #quit -sim

```

4-QuestaSim Snippets:





```
# CARRYOUT=1,CARRYOUTF=1
# ** Note: $stop : D:/DIGITAL DIPLOMA/project1/DSP48A1_tb.v(99)
# Time: 76 ns Iteration: 1 Instance: /DSP48A1_TB
# Break in Module DSP48A1_TB at D:/DIGITAL DIPLOMA/project1/DSP48A1_tb.v line 99
```

VSIM 17>

1 ns to 47 ns

Project : DSP

Now: 76 ns Delta: 1

CEP

5-Constraint File:

```
## Clock signal
set_property -dict {PACKAGE_PIN W5 IOSTANDARD LVCMOS33} [get_ports CLK]
create_clock -period 10.000 -name sys_clk_pin -waveform {0.000 5.000} -add [get_ports CLK]
```

```
161 create_debug_core u_ila_0 ila
162 set_property ALL_PROBE_SAME MJ true [get_debug_cores u_ila_0]
163 set_property ALL_PROBE_SAME MJ CNT 1 [get_debug_cores u_ila_0]
164 set_property C_ADV_TRIGGER False [get_debug_cores u_ila_0]
165 set_property C_DATA_DEPTH 1024 [get_debug_cores u_ila_0]
166 set_property C_EN_STRG_QUAL false [get_debug_cores u_ila_0]
167 set_property C_INPUT_PIPE_STRIDES 0 [get_debug_cores u_ila_0]
168 set_property C_TRIGIN_EN false [get_debug_cores u_ila_0]
169 set_property C_TRIGOUT_EN false [get_debug_cores u_ila_0]
170 set_property port_width 1 [get_debug_ports u_ila_0/clock]
171 connect_debug_port u_ila_0/clock [get_nets [list CLK IBUF_BUF]]
172 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe0]
173 set_property port_width 18 [get_debug_ports u_ila_0/probe0]
174 connect_debug_port u_ila_0/probe0 [get_nets [list D_IBUF[0] D_IBUF[1] D_IBUF[2] D_IBUF[3] D_IBUF[4] D_IBUF[5] D_IBUF[6] D_IBUF[7] D_IBUF[8] D_IBUF[9] D_IBUF[10] D_IBUF[11] D_IBUF[12] D_IBUF[13] D_IBUF[14] D_IBUF[15] D_IBUF[16] D_IBUF[17]]]
175 create_debug_port u_ila_0 probe
176 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe1]
177 set_property port_width 18 [get_debug_ports u_ila_0/probe1]
178 connect_debug_port u_ila_0/probe1 [get_nets [list BCOU_OBUF[0] BCOU_OBUF[1] BCOU_OBUF[2] BCOU_OBUF[3] BCOU_OBUF[4] BCOU_OBUF[5] BCOU_OBUF[6] BCOU_OBUF[7] BCOU_OBUF[8] BCOU_OBUF[9] BCOU_OBUF[10] BCOU_OBUF[11] BCOU_OBUF[12] BCOU_OBUF[13] BCOU_OBUF[14] BCOU_OBUF[15] BCOU_OBUF[16] BCOU_OBUF[17]]]
179 create_debug_port u_ila_0 probe
180 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe2]
181 set_property port_width 48 [get_debug_ports u_ila_0/probe2]
182 connect_debug_port u_ila_0/probe2 [get_nets [list C_IBUF[0] C_IBUF[1] C_IBUF[2] C_IBUF[3] C_IBUF[4] C_IBUF[5] C_IBUF[6] C_IBUF[7] C_IBUF[8] C_IBUF[9] C_IBUF[10] C_IBUF[11] C_IBUF[12] C_IBUF[13] C_IBUF[14] C_IBUF[15] C_IBUF[16] C_IBUF[17] C_IBUF[18] C_IBUF[19] C_IBUF[20] C_IBUF[21] C_IBUF[22] C_IBUF[23] C_IBUF[24] C_IBUF[25] C_IBUF[26] C_IBUF[27] C_IBUF[28] C_IBUF[29] C_IBUF[30] C_IBUF[31] C_IBUF[32] C_IBUF[33] C_IBUF[34] C_IBUF[35] C_IBUF[36] C_IBUF[37] C_IBUF[38] C_IBUF[39] C_IBUF[40] C_IBUF[41] C_IBUF[42] C_IBUF[43] C_IBUF[44] C_IBUF[45] C_IBUF[46] C_IBUF[47]]]
183 create_debug_port u_ila_0 probe
184 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe3]
185 set_property port_width 48 [get_debug_ports u_ila_0/probe3]
186 connect_debug_port u_ila_0/probe3 [get_nets [list PCIN_IBUF[0] PCIN_IBUF[1] PCIN_IBUF[2] PCIN_IBUF[3] PCIN_IBUF[4] PCIN_IBUF[5] PCIN_IBUF[6] PCIN_IBUF[7] PCIN_IBUF[8] PCIN_IBUF[9] PCIN_IBUF[10] PCIN_IBUF[11] PCIN_IBUF[12] PCIN_IBUF[13] PCIN_IBUF[14] PCIN_IBUF[15] PCIN_IBUF[16] PCIN_IBUF[17] PCIN_IBUF[18] PCIN_IBUF[19] PCIN_IBUF[20] PCIN_IBUF[21] PCIN_IBUF[22] PCIN_IBUF[23] PCIN_IBUF[24] PCIN_IBUF[25] PCIN_IBUF[26] PCIN_IBUF[27] PCIN_IBUF[28] PCIN_IBUF[29] PCIN_IBUF[30] PCIN_IBUF[31] PCIN_IBUF[32] PCIN_IBUF[33] PCIN_IBUF[34] PCIN_IBUF[35] PCIN_IBUF[36] PCIN_IBUF[37] PCIN_IBUF[38] PCIN_IBUF[39] PCIN_IBUF[40] PCIN_IBUF[41] PCIN_IBUF[42] PCIN_IBUF[43] PCIN_IBUF[44] PCIN_IBUF[45] PCIN_IBUF[46] PCIN_IBUF[47]]]
187 create_debug_port u_ila_0 probe
188 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe4]
189 set_property port_width 18 [get_debug_ports u_ila_0/probe4]
190 connect_debug_port u_ila_0/probe4 [get_nets [list E_IBUF[0] E_IBUF[1] E_IBUF[2] E_IBUF[3] E_IBUF[4] E_IBUF[5] E_IBUF[6] E_IBUF[7] E_IBUF[8] E_IBUF[9] E_IBUF[10] E_IBUF[11] E_IBUF[12] E_IBUF[13] E_IBUF[14] E_IBUF[15] E_IBUF[16] E_IBUF[17]]]
191 create_debug_port u_ila_0 probe
192 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe5]
193 set_property port_width 8 [get_debug_ports u_ila_0/probe5]
194 connect_debug_port u_ila_0/probe5 [get_nets [list OPMODE_IBUF[0] OPMODE_IBUF[1] OPMODE_IBUF[2] OPMODE_IBUF[3] OPMODE_IBUF[4] OPMODE_IBUF[5] OPMODE_IBUF[6] OPMODE_IBUF[7]]]
195 create_debug_port u_ila_0 probe
196 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe6]
197 set_property port_width 48 [get_debug_ports u_ila_0/probe6]
198 connect_debug_port u_ila_0/probe6 [get_nets [list P_OBUF[0] P_OBUF[1] P_OBUF[2] P_OBUF[3] P_OBUF[4] P_OBUF[5] P_OBUF[6] P_OBUF[7] P_OBUF[8] P_OBUF[9] P_OBUF[10] P_OBUF[11] P_OBUF[12] P_OBUF[13] P_OBUF[14] P_OBUF[15] P_OBUF[16] P_OBUF[17] P_OBUF[18] P_OBUF[19] P_OBUF[20] P_OBUF[21] P_OBUF[22] P_OBUF[23] P_OBUF[24] P_OBUF[25] P_OBUF[26] P_OBUF[27] P_OBUF[28] P_OBUF[29] P_OBUF[30] P_OBUF[31] P_OBUF[32] P_OBUF[33] P_OBUF[34] P_OBUF[35] P_OBUF[36] P_OBUF[37] P_OBUF[38] P_OBUF[39] P_OBUF[40] P_OBUF[41] P_OBUF[42] P_OBUF[43] P_OBUF[44] P_OBUF[45] P_OBUF[46] P_OBUF[47]]]
199 create_debug_port u_ila_0 probe
200 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe7]
201 set_property port_width 16 [get_debug_ports u_ila_0/probe7]
202 connect_debug_port u_ila_0/probe7 [get_nets [list M_OBUF[0] M_OBUF[1] M_OBUF[2] M_OBUF[3] M_OBUF[4] M_OBUF[5] M_OBUF[6] M_OBUF[7] M_OBUF[8] M_OBUF[9] M_OBUF[10] M_OBUF[11] M_OBUF[12] M_OBUF[13] M_OBUF[14] M_OBUF[15]]]
203 create_debug_port u_ila_0 probe
204 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe8]
205 set_property port_width 18 [get_debug_ports u_ila_0/probe8]
206 connect_debug_port u_ila_0/probe8 [get_nets [list A_IBUF[0] A_IBUF[1] A_IBUF[2] A_IBUF[3] A_IBUF[4] A_IBUF[5] A_IBUF[6] A_IBUF[7] A_IBUF[8] A_IBUF[9] A_IBUF[10] A_IBUF[11] A_IBUF[12] A_IBUF[13] A_IBUF[14] A_IBUF[15] A_IBUF[16] A_IBUF[17]]]
207 create_debug_port u_ila_0 probe
208 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe9]
209 set_property port_width 1 [get_debug_ports u_ila_0/probe9]
210 connect_debug_port u_ila_0/probe9 [get_nets [list CARRYOUTF_OBUF]]
211 create_debug_port u_ila_0 probe
212 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe10]
213 set_property port_width 1 [get_debug_ports u_ila_0/probe10]
214 connect_debug_port u_ila_0/probe10 [get_nets [list CEA_IBUF]]
```



```

215 create_debug_port u_ila_0 probe
216 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe11]
217 set_property port_width 1 [get_debug_ports u_ila_0/probe11]
218 connect_debug_port u_ila_0/probe11 [get_nets [list CEB_IBUF]]
219 create_debug_port u_ila_0 probe
220 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe12]
221 set_property port_width 1 [get_debug_ports u_ila_0/probe12]
222 connect_debug_port u_ila_0/probe12 [get_nets [list CEC_IBUF]]
223 create_debug_port u_ila_0 probe
224 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe13]
225 set_property port_width 1 [get_debug_ports u_ila_0/probe13]
226 connect_debug_port u_ila_0/probe13 [get_nets [list CECARRYIN_IBUF]]
227 create_debug_port u_ila_0 probe
228 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe14]
229 set_property port_width 1 [get_debug_ports u_ila_0/probe14]
230 connect_debug_port u_ila_0/probe14 [get_nets [list CED_IBUF]]
231 create_debug_port u_ila_0 probe
232 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe15]
233 set_property port_width 1 [get_debug_ports u_ila_0/probe15]
234 connect_debug_port u_ila_0/probe15 [get_nets [list CEM_IBUF]]
235 create_debug_port u_ila_0 probe
236 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe16]
237 set_property port_width 1 [get_debug_ports u_ila_0/probe16]
238 connect_debug_port u_ila_0/probe16 [get_nets [list CEOPMODE_IBUF]]
239 create_debug_port u_ila_0 probe
240 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe17]
241 set_property port_width 1 [get_debug_ports u_ila_0/probe17]
242 connect_debug_port u_ila_0/probe17 [get_nets [list CEP_IBUF]]
243 create_debug_port u_ila_0 probe
244 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe18]
245 set_property port_width 1 [get_debug_ports u_ila_0/probe18]
246 connect_debug_port u_ila_0/probe18 [get_nets [list CLK_IBUF]]
247 create_debug_port u_ila_0 probe
248 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe19]
249 set_property port_width 1 [get_debug_ports u_ila_0/probe19]
250 connect_debug_port u_ila_0/probe19 [get_nets [list RSTA_IBUF]]
251 create_debug_port u_ila_0 probe
252 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe20]
253 set_property port_width 1 [get_debug_ports u_ila_0/probe20]
254 connect_debug_port u_ila_0/probe20 [get_nets [list RSTB_IBUF]]
255 create_debug_port u_ila_0 probe
256 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe21]
257 set_property port_width 1 [get_debug_ports u_ila_0/probe21]
258 connect_debug_port u_ila_0/probe21 [get_nets [list RSTC_IBUF]]
259 create_debug_port u_ila_0 probe
260 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe22]
261 set_property port_width 1 [get_debug_ports u_ila_0/probe22]
262 connect_debug_port u_ila_0/probe22 [get_nets [list RSTCARRYIN_IBUF]]
263 create_debug_port u_ila_0 probe
264 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe23]
265 set_property port_width 1 [get_debug_ports u_ila_0/probe23]
266 connect_debug_port u_ila_0/probe23 [get_nets [list RSTD_IBUF]]
267 create_debug_port u_ila_0 probe
268 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe24]
269 set_property port_width 1 [get_debug_ports u_ila_0/probe24]

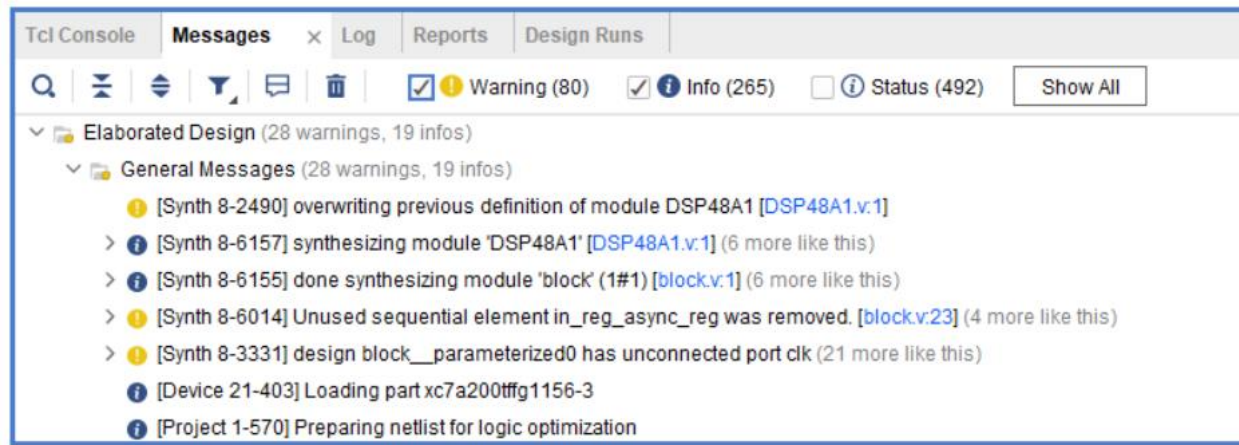
```

```

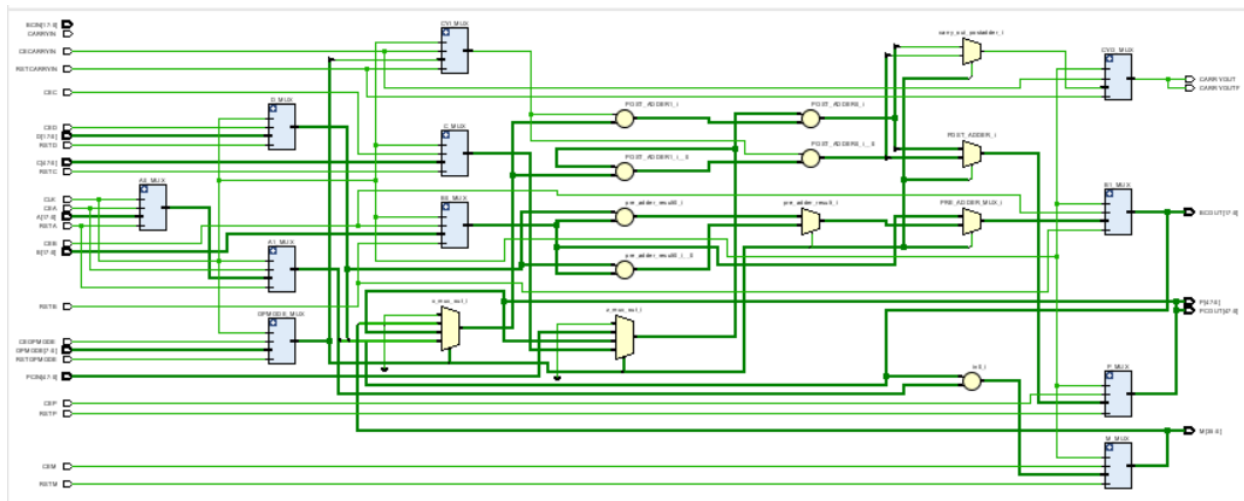
270 create_debug_port u_ila_0 probe
271 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe24]
272 set_property port_width 1 [get_debug_ports u_ila_0/probe24]
273 connect_debug_port u_ila_0/probe24 [get_nets [list RSTM_IBUF]]
274 create_debug_port u_ila_0 probe
275 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe25]
276 set_property port_width 1 [get_debug_ports u_ila_0/probe25]
277 connect_debug_port u_ila_0/probe25 [get_nets [list RSTOPMODE_IBUF]]
278 create_debug_port u_ila_0 probe
279 set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe26]
280 set_property port_width 1 [get_debug_ports u_ila_0/probe26]
281 connect_debug_port u_ila_0/probe26 [get_nets [list RSTP_IBUF]]
282 set_property C_CLK_INPUT_FREQ_HZ 300000000 [get_debug_cores dbg_hub]
283 set_property C_ENABLE_CLK_DIVIDER false [get_debug_cores dbg_hub]
284 set_property C_USER_SCAN_CHAIN 1 [get_debug_cores dbg_hub]
285 connect_debug_port dbg_hub/clk [get_nets CLK_IBUF_BUF6]

```

-Messages:

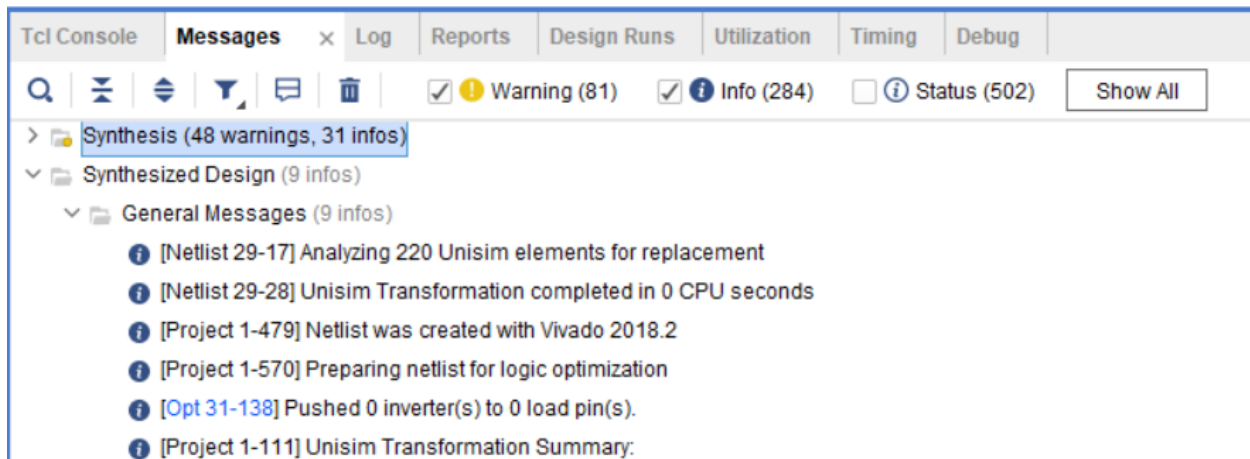


-Schematic:



7-Synthesis:

-Messages:



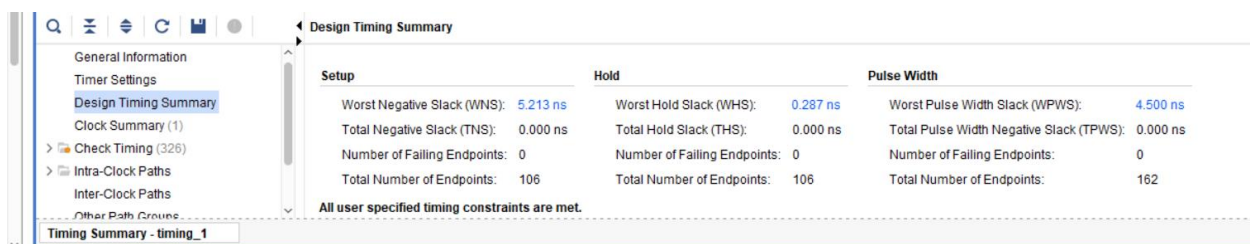
The Messages window displays the following hierarchy and content:

- Synthesis (48 warnings, 31 infos)
 - Synthesized Design (9 infos)
 - General Messages (9 infos)
 - [Netlist 29-17] Analyzing 220 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:

-Utilization report:

Name	Slice LUTs (134600)	Slice Registers (269200)	DSP s (740)	Bonded IOB (500)	BUFGCTRL (32)
DSP48A1	263	160	1	327	1
A1_MUX (block)	1	18	0	0	0
B1_MUX (block_0)	1	18	0	0	0
C_MUX (block_para...	1	48	0	0	0

-Timing Report:

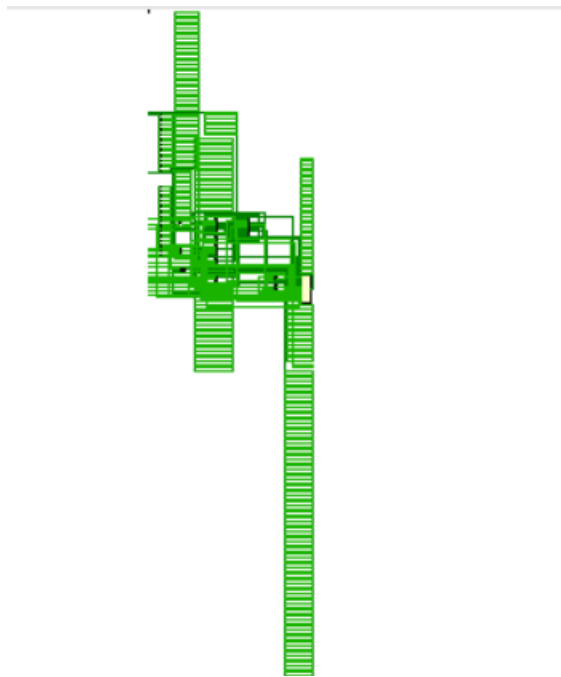


The Design Timing Summary report shows the following data:

Setup	Hold	Pulse Width
Worst Negative Slack (WNS): 5.213 ns	Worst Hold Slack (WHS): 0.287 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: 106	Total Number of Endpoints: 106	Total Number of Endpoints: 162

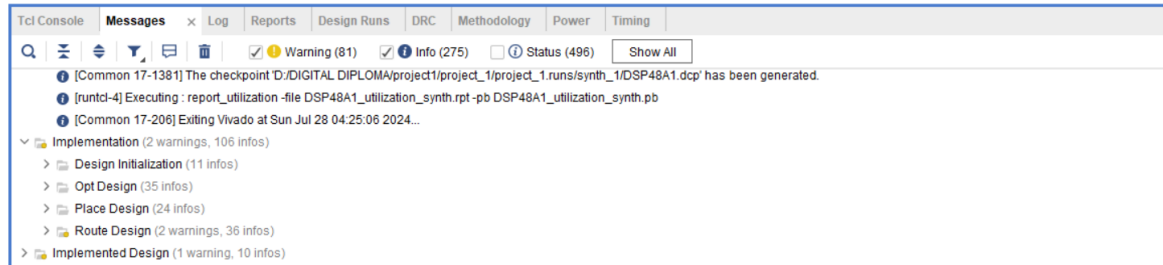
All user specified timing constraints are met.

-Schematic:



8-Implementation:

-Messages:



-Utilization report:

Name	Slice LUTs (133800)	Slice Registers (267600)	F7 Muxes (66900)	F8 Muxes (33450)	Slice (33450)	LUT as Logic (133800)	LUT as Memory (46200)	LUT Flip Flop Pairs (133800)	Block RAM Tile (365)	DSP s (740)
DSP48A1	2737	4225	96	12	1477	2263	474	1553	8	1
A1_MUX (block)	1	18	0	0	7	1	0	0	0	0
B1_MUX (block_0)	1	18	0	0	6	1	0	0	0	0
MUX (block_0)	1	18	0	0	1	1	0	0	0	0

-Timing Report:

Setup	Hold	Pulse Width
Worst Negative Slack (WNS): 2.974 ns	Worst Hold Slack (WHS): 0.057 ns	Worst Pulse Width Slack (WPWS): 3.950 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: 8077	Total Number of Endpoints: 8061	Total Number of Endpoints: 5136

All user specified timing constraints are met.

-Device:

