Name: Mohamed Gamal Mansour

Project1

1-RTL Code:

```
module block(clk,enable_clk, reset,in, out);
parameter RSTTYPE = "SYNC";
parameter mux_sel = 1;
parameter NUMBER_BITS=18;
input clk, enable_clk, reset;
input [NUMBER_BITS-1:0] in;
output reg [NUMBER_BITS-1:0] out;
 reg [NUMBER_BITS-1:0] in_reg_async;
 reg [NUMBER_BITS-1:0] in_reg_sync;
 reg [NUMBER_BITS-1:0] in_reg;
 always @(posedge clk) begin
   if (enable_clk) begin
      if (reset) begin
        in_reg_sync <=0;
        end
        else begin
        in_reg_sync<=in;
        end
   end
 end
 always @(posedge clk or posedge reset) begin
  if (reset) begin
    in_reg_async <=0;
    end
   else if (enable_clk) begin
   in_reg_async<=in;
   end
 end
always@(*) begin
  if(RSTTYPE == "SYNC" ) begin
    in_reg= in_reg_sync;
    end
    else begin
    in_reg=in_reg_async;
       end
    if(mux_sel) begin
         out= in_reg;
         end
         else begin
         out = in;
         end
end
endmodule
```

```
module DSP48A1(A,B,D,C,CLK,CARRYIN,OPMODE,BCIN,RSTA,RSTB,RSTM,RSTP,RSTC,RSTD,RSTCARRYIN,RSTOPMODE,
    CEA,CEB,CEM,CEP,CEC,CED,CECARRYIN,CEOPMODE,PCIN,BCOUT,PCOUT,P,M,CARRYOUT,CARRYOUTF);
parameter A0REG=0, A1REG=1, B0REG=0, B1REG=1;
parameter CREG=1, DREG=1, MREG=1, PREG=1, CARRYINREG=1, CARRYOUTREG=1, OPMODEREG=1;
parameter CARRYINSEL= "OPMODE5";
parameter B_INPUT = "DIRECT";
parameter RSTTYPE="SYNC";
input [17:0] A,B,D,BCIN;
input [47:0] C,PCIN;
input [7:0] OPMODE;
input CLK,CARRYIN,RSTA,RSTB,RSTM,RSTP,RSTC,RSTD,RSTCARRYIN,RSTOPMODE,CEA,CEB,CEM,CEP,CEC,CED,CECARRYIN,CEOPMODE;
output [17:0] BCOUT;
output [47:0] PCOUT, P;
output [35:0] M;
output   CARRYOUT;
output CARRYOUTF;
reg [17:0] B_OR_BCIN;
wire [17:0] D_mux_out,B0_mux_out,A0_mux_out;
wire [47:0] C_mux_out;
wire [17:0] B1_mux_out,A1_mux_out;
wire [7:0] OPMODE_MUX_OUT;
reg  [35:0] MULTPLIER_RESULT;
reg CARRYCASCADE_MUX, carry_out_postadder;
wire CIN;
reg [47:0] POST_ADDER;
reg [17:0] pre_adder_result, PRE_ADDER_MUX;
reg [47:0] x_mux_out, z_mux_out;
block #(.RSTTYPE(RSTTYPE), .mux_sel(DREG),.NUMBER_BITS(18)) D_MUX(CLK,CED, RSTD,D, D_mux_out);
block #(.RSTTYPE(RSTTYPE), .mux_sel(B0REG),.NUMBER_BITS(18)) B0_MUX(CLK,CEB, RSTB,B_OR_BCIN, B0_mux_out);
block #(.RSTTYPE(RSTTYPE), .mux_sel(A0REG),.NUMBER_BITS(18)) A0_MUX(CLK,CEA, RSTA,A, A0_mux_out);
block #(.RSTTYPE(RSTTYPE), .mux_sel(CREG),.NUMBER_BITS(48)) C_MUX(CLK,CEC, RSTC,C, C_mux_out);
block #(.RSTTYPE(RSTTYPE), .mux_sel(B1REG),.NUMBER_BITS(18)) B1_MUX(CLK,CEB, RSTB,PRE_ADDER_MUX, B1_mux_out);
block #(.RSTTYPE(RSTTYPE), .mux_sel(A1REG),.NUMBER_BITS(18)) A1_MUX(CLK,CEA, RSTA,A0_mux_out, A1_mux_out);
block #(.RSTTYPE(RSTTYPE), .mux_sel(MREG),.NUMBER_BITS(36)) M_MUX(CLK,CEM, RSTM,MULTPLIER_RESULT, M);

block #(.RSTTYPE(RSTTYPE), .mux_sel(CARRYINREG),.NUMBER_BITS(1)) CYI_MUX(CLK,CECARRYIN, RSTCARRYIN,CARRYCASCADE_MUX, CIN);
block #(.RSTTYPE(RSTTYPE), .mux_sel(CARRYOUTREG),.NUMBER_BITS(1)) CYO_MUX(CLK,CECARRYIN, RSTCARRYIN,carry_out_postadder, CARRYOUT);
block #(.RSTTYPE(RSTTYPE), .mux_sel(PREG),.NUMBER_BITS(48)) P_MUX(CLK,CEP, RSTP,POST_ADDER, P);
block #(.RSTTYPE(RSTTYPE), .mux_sel(OPMODEREG),.NUMBER_BITS(8)) OPMODE_MUX(CLK,CEOPMODE, RSTOPMODE,OPMODE, OPMODE_MUX_OUT);

always @ (*) begin
   if (B_INPUT == "DIRECT")
       B_OR_BCIN=B;
       else if (B_INPUT == "CASCADE")
        B_OR_BCIN=BCIN;
        else
        B_OR_BCIN=0;

     if(OPMODE_MUX_OUT[6] == 0 )
           pre_adder_result=D_mux_out + B0_mux_out;
       else
        pre_adder_result=D_mux_out - B0_mux_out;

        if (OPMODE_MUX_OUT[4] == 0)
             PRE_ADDER_MUX=B0_mux_out;
         else
```

Line 42, Column 28

```
always @ (*) begin
   if (B_INPUT == "DIRECT")
       B_OR_BCIN=B;
       else if (B_INPUT == "CASCADE")
        B_OR_BCIN=BCIN;
        else
        B_OR_BCIN=0;

    if(OPMODE_MUX_OUT[6] == 0 )
           pre_adder_result=D_mux_out + B0_mux_out;
       else
        pre_adder_result=D_mux_out - B0_mux_out;

        if (OPMODE_MUX_OUT[4] == 0)
              PRE_ADDER_MUX=B0_mux_out;
         else
            PRE_ADDER_MUX=pre_adder_result;

         MULTPLIER_RESULT= B1_mux_out * A1_mux_out;

         if( CARRYINSEL == "OPMODE5")
              CARRYCASCADE_MUX= OPMODE_MUX_OUT[5];
           else if( CARRYINSEL == "CARRYIN")
             CARRYCASCADE_MUX= CARRYIN;
            else
             CARRYCASCADE_MUX= 0;

             case (OPMODE_MUX_OUT[1:0])
             0: x_mux_out=0;
             1: x_mux_out={12'd0,M};
             2: x_mux_out=P;
             3: x_mux_out={D_mux_out[11:0],A1_mux_out[17:0],B1_mux_out[17:0]};
             endcase

             case(OPMODE_MUX_OUT[3:2])
             0: z_mux_out=0;
             1: z_mux_out=PCIN;
             2: z_mux_out=P;
             3: z_mux_out=C_mux_out;
             endcase

             if(OPMODE_MUX_OUT[7])
                {carry_out_postadder,POST_ADDER}=z_mux_out - (x_mux_out+CIN);
              else
                {carry_out_postadder,POST_ADDER}=z_mux_out+x_mux_out+CIN;
end

assign PCOUT=P;
assign CARRYOUTF=CARRYOUT;
assign BCOUT=B1_mux_out;
endmodule
```

Line 42, Column 28

2-Testbench code:

```
module DSP48A1_TB();
parameter A0REG=0, A1REG=1, B0REG=0, B1REG=1;
parameter CREG=1, DREG=1, MREG=1, PREG=1, CARRYINREG=1, CARRYOUTREG=1, OPMODEREG=1;
parameter CARRYINSEL= "OPMODE5";
parameter B_INPUT = "DIRECT";
parameter RSTTYPE="SYNC";
reg [17:0] A,B,D,BCIN;
reg [47:0] C,PCIN;
reg [7:0] OPMODE;
reg CLK,CARRYIN,RSTA,RSTB,RSTM,RSTP,RSTC,RSTD,RSTCARRYIN,RSTOPMODE,CEA,CEB,CEM,CEP,CEC,CED,CECARRYIN,CEOPMODE;
wire [17:0] BCOUT;
wire [47:0] PCOUT, P;
wire [35:0] M;
wire CARRYOUT;
wire CARRYOUTF;
DSP48A1 #(A0REG, A1REG, B0REG, B1REG, CREG, DREG, MREG, PREG, CARRYINREG, CARRYOUTREG, OPMODEREG) dut(A,B,D,C,CLK,CARRYIN,OPMODE,BCIN,RSTA,
    RSTB,RSTM,RSTP,RSTC,RSTD,RSTCARRYIN,RSTOPMODE, CEA,CEB,CEM,CEP,CEC,CED,CECARRYIN,CEOPMODE,PCIN,BCOUT,PCOUT,P,M,CARRYOUT,CARRYOUTF);
initial begin
    CLK=0;
    forever
    #1 CLK=~CLK;
end

initial begin
    CEA=1; CEB=1; CEC=1; CECARRYIN=1; CED=1; CEM=1; CEOPMODE=1; CEP=1;
    RSTA=1; RSTB=1; RSTC=1; RSTCARRYIN=1; RSTD=1; RSTM=1; RSTOPMODE=1; RSTP=1;
    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;
    repeat(4)
    @(negedge CLK);
    RSTA=0; RSTB=0; RSTC=0; RSTCARRYIN=0; RSTD=0; RSTM=0; RSTOPMODE=0; RSTP=0;
    repeat(4)
    @(negedge CLK);
    if(BCOUT != 20 || P != 61 || CARRYOUT != 0 || M != 40 || PCOUT != 61 || CARRYOUTF !=0) begin
      $display("There is an error");
      $stop;
    end

    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;
    repeat(4)
    @(negedge CLK);
    if(BCOUT != 20 || P != 61 || CARRYOUT != 0 || M != 40 || PCOUT != 61 || CARRYOUTF !=0) begin
      $display("There is an error");
      $stop;
    end

    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;
    repeat(4)
    @(negedge CLK);
    if(BCOUT != 20 || P != 19 || CARRYOUT != 0 || M != 20 || PCOUT != 19 || CARRYOUTF !=0 ) begin
      $display("There is an error");
      $stop;
    end

     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;
     repeat(4)
    @(negedge CLK);
```

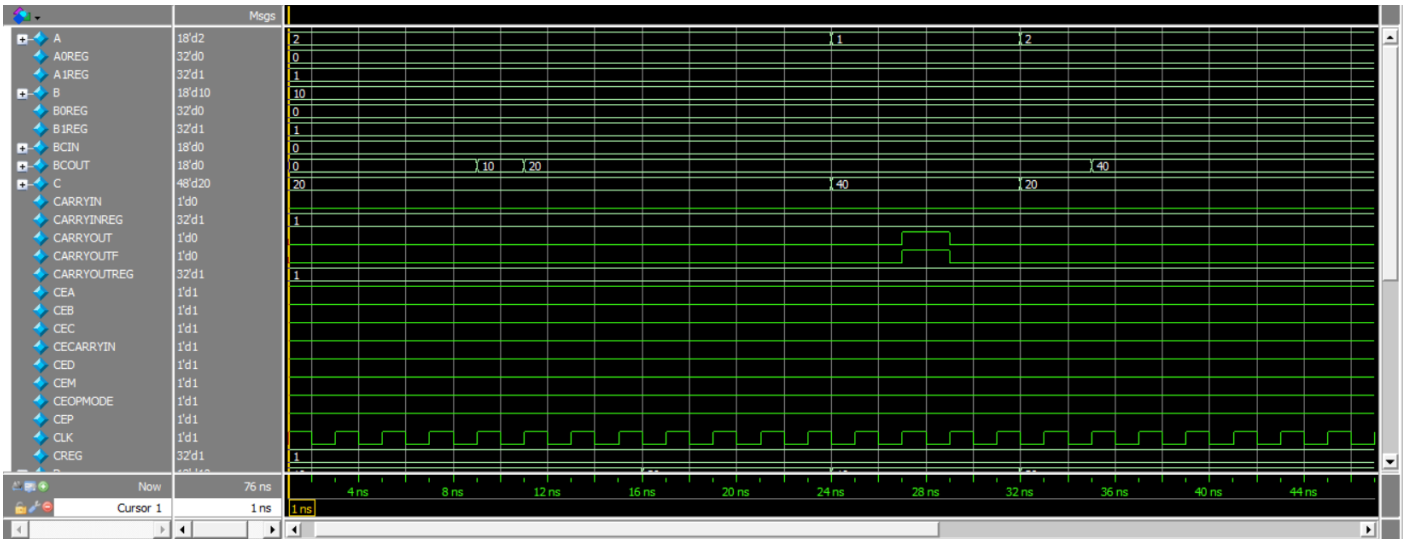
Line 1, Column 21

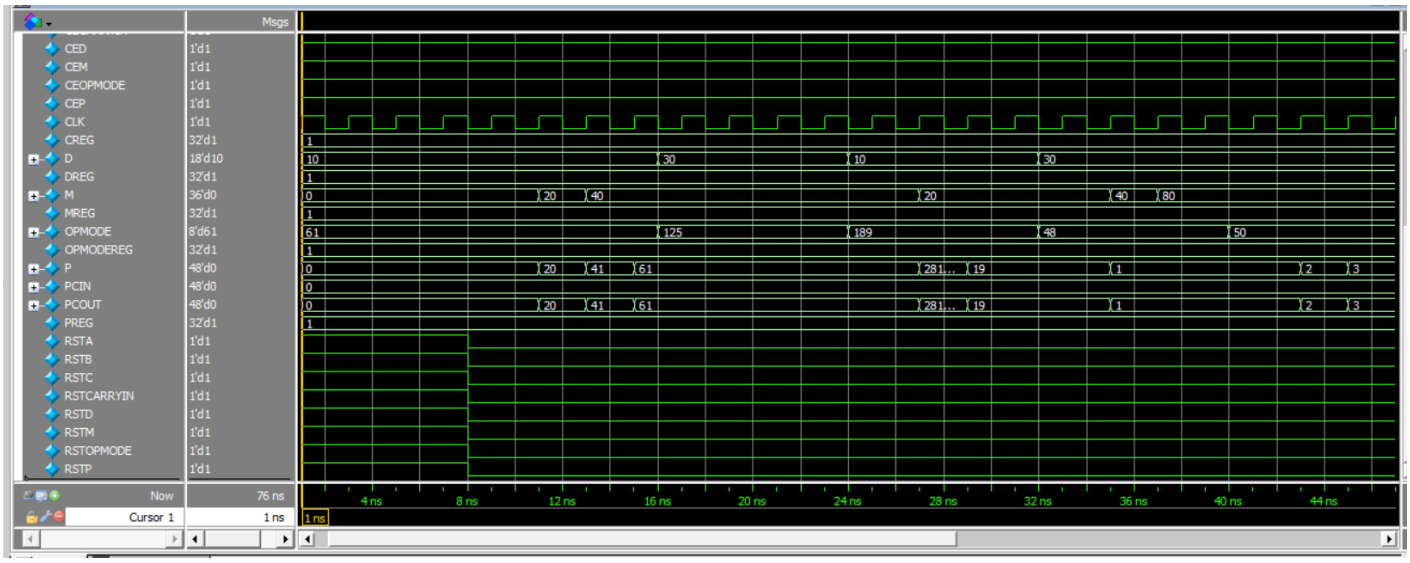
```
53
54          D=30; B=10; OPMODE[6]=0; OPMODE[4]=1; A=2; OPMODE[1:0]=0; C=20; OPMODE[7]=0; OPMODE[5]=1; OPMODE[3:2]=0;
55          repeat(4)
56         @(negedge CLK);
57         if(BCOUT != 40 || P != 1 || CARRYOUT != 0 || M != 80 || PCOUT != 1 || CARRYOUTF !=0) begin
58           $display("There 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'h000000010000A || CARRYOUT != 0 || M != 40 || PCOUT != 48'h000000010000A || 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'hFFFFFFFFFFFF; 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
```

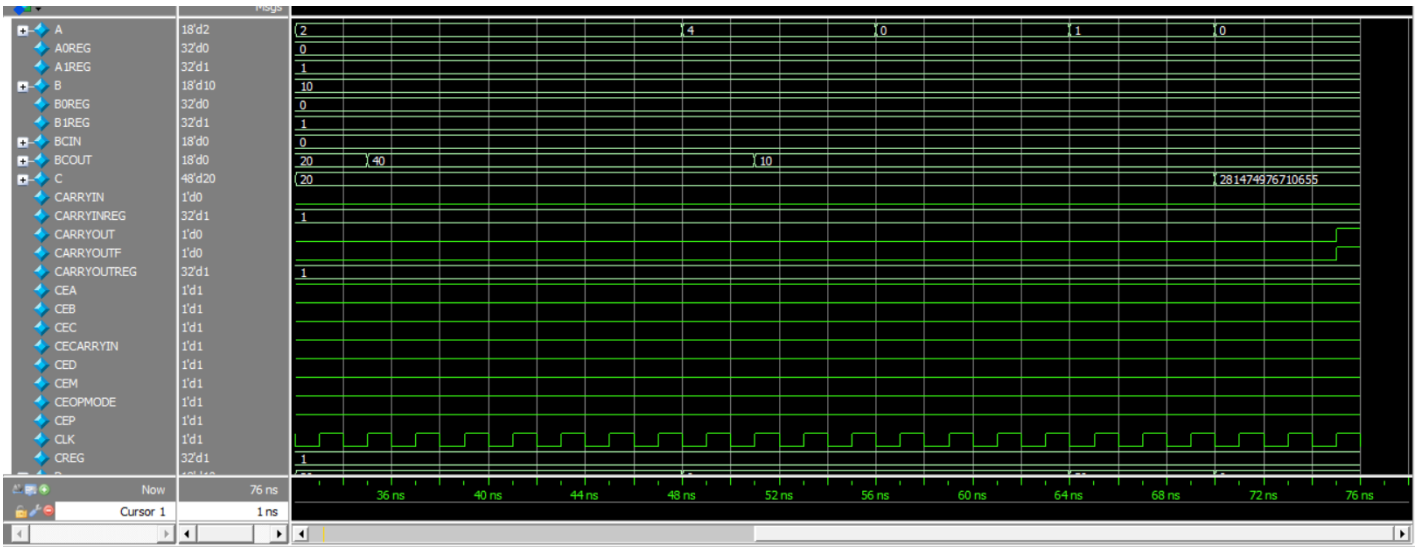
3-Do file:

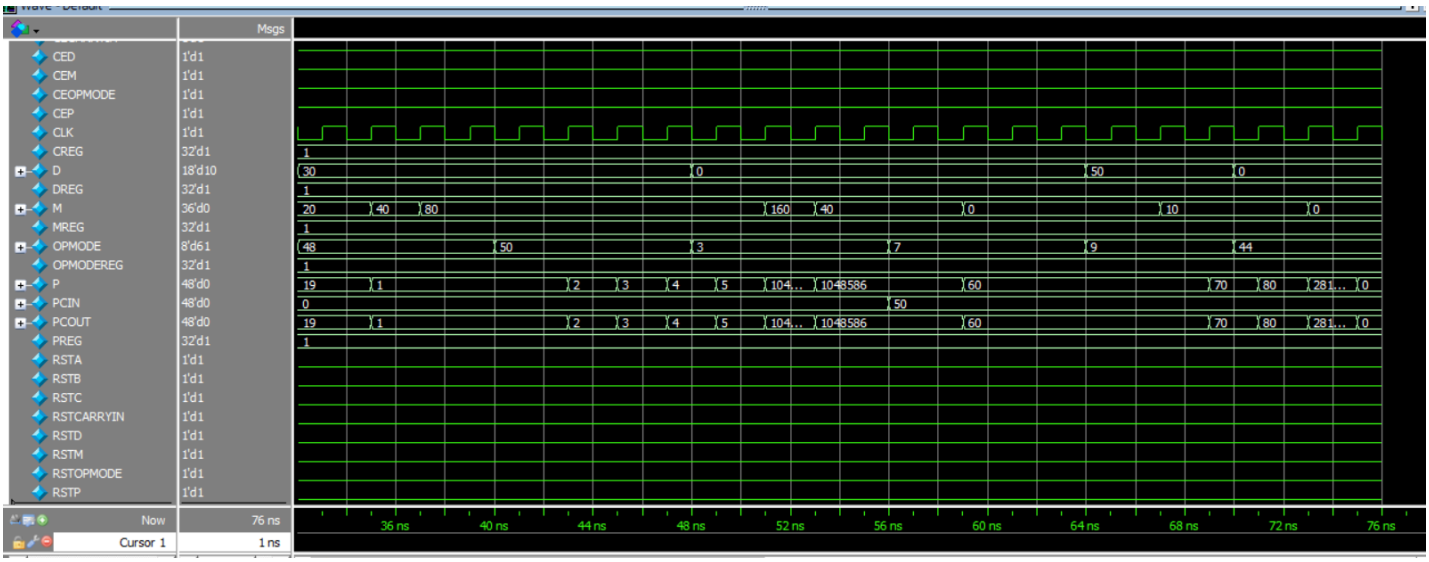
```
1   vlib work
2   vlog DSP48A1.v DSP48A1_tb.v block.v
3   vsim -voptargs=+acc work.DSP48A1_TB
4   add wave *
5   run -all
6   #quit -sim
```

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]
```

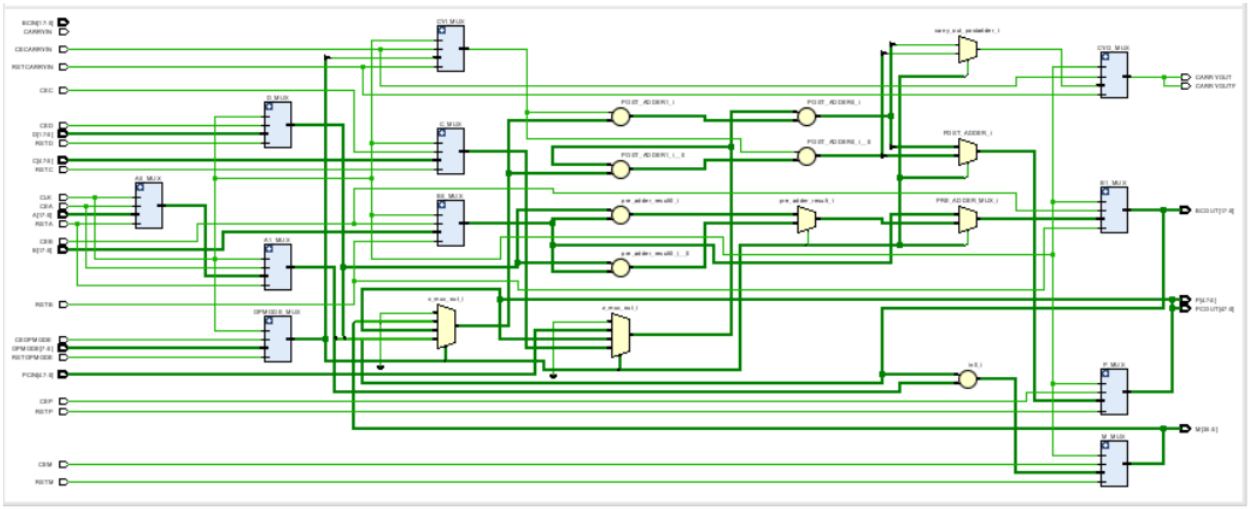
```
create_debug_core u_ila_0 ila
set_property ALL_PROBE_SAME_MU true [get_debug_cores u_ila_0]
set_property ALL_PROBE_SAME_MU_CNT 1 [get_debug_cores u_ila_0]
set_property C_ADV_TRIGGER false [get_debug_cores u_ila_0]
set_property C_DATA_DEPTH 1024 [get_debug_cores u_ila_0]
set_property C_EN_STRG_QUAL false [get_debug_cores u_ila_0]
set_property C_INPUT_PIPE_STAGES 0 [get_debug_cores u_ila_0]
set_property C_TRIGIN_EN false [get_debug_cores u_ila_0]
set_property C_TRIGOUT_EN false [get_debug_cores u_ila_0]
set_property port_width 1 [get_debug_ports u_ila_0/clk]
connect_debug_port u_ila_0/clk [get_nets [list CLK_IBUF_BUFG]]
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe0]
set_property port_width 18 [get_debug_ports u_ila_0/probe0]
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
create_debug_port u_ila_0 probe
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe1]
set_property port_width 18 [get_debug_ports u_ila_0/probe1]
connect_debug_port u_ila_0/probe1 [get_nets [list {BCOUT_OBUF[0]} {BCOUT_OBUF[1]} {BCOUT_OBUF[2]} {BCOUT_OBUF[3]} {BCOUT_OBUF[4]} {BCOUT_OBUF[5]} {BCOUT_OBUF[6]} {BCOUT_OBUF[7]} {BCOUT_OBUF[8]} {BCOUT_OBUF[9]} {BCOUT
create_debug_port u_ila_0 probe
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe2]
set_property port_width 48 [get_debug_ports u_ila_0/probe2]
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
create_debug_port u_ila_0 probe
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe3]
set_property port_width 48 [get_debug_ports u_ila_0/probe3]
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]} {
create_debug_port u_ila_0 probe
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe4]
set_property port_width 18 [get_debug_ports u_ila_0/probe4]
connect_debug_port u_ila_0/probe4 [get_nets [list {B_IBUF[0]} {B_IBUF[1]} {B_IBUF[2]} {B_IBUF[3]} {B_IBUF[4]} {B_IBUF[5]} {B_IBUF[6]} {B_IBUF[7]} {B_IBUF[8]} {B_IBUF[9]} {B_IBUF[10]} {B_IBUF[11]} {B_IBUF[12]} {B_IBUF
create_debug_port u_ila_0 probe
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe5]
set_property port_width 8 [get_debug_ports u_ila_0/probe5]
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]}]]
create_debug_port u_ila_0 probe
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe6]
set_property port_width 48 [get_debug_ports u_ila_0/probe6]
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
create_debug_port u_ila_0 probe
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe7]
set_property port_width 36 [get_debug_ports u_ila_0/probe7]
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
create_debug_port u_ila_0 probe
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe8]
set_property port_width 18 [get_debug_ports u_ila_0/probe8]
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
create_debug_port u_ila_0 probe
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe9]
set_property port_width 1 [get_debug_ports u_ila_0/probe9]
connect_debug_port u_ila_0/probe9 [get_nets [list CARRYOUTF_OBUF]]
create_debug_port u_ila_0 probe
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe10]
set_property port_width 1 [get_debug_ports u_ila_0/probe10]
connect_debug_port u_ila_0/probe10 [get_nets [list CEA_IBUF]]
```

```
create_debug_port u_ila_0 probe
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe11]
set_property port_width 1 [get_debug_ports u_ila_0/probe11]
connect_debug_port u_ila_0/probe11 [get_nets [list CEB_IBUF]]
create_debug_port u_ila_0 probe
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe12]
set_property port_width 1 [get_debug_ports u_ila_0/probe12]
connect_debug_port u_ila_0/probe12 [get_nets [list CEC_IBUF]]
create_debug_port u_ila_0 probe
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe13]
set_property port_width 1 [get_debug_ports u_ila_0/probe13]
connect_debug_port u_ila_0/probe13 [get_nets [list CECARRYIN_IBUF]]
create_debug_port u_ila_0 probe
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe14]
set_property port_width 1 [get_debug_ports u_ila_0/probe14]
connect_debug_port u_ila_0/probe14 [get_nets [list CED_IBUF]]
create_debug_port u_ila_0 probe
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe15]
set_property port_width 1 [get_debug_ports u_ila_0/probe15]
connect_debug_port u_ila_0/probe15 [get_nets [list CEM_IBUF]]
create_debug_port u_ila_0 probe
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe16]
set_property port_width 1 [get_debug_ports u_ila_0/probe16]
connect_debug_port u_ila_0/probe16 [get_nets [list CEOPMODE_IBUF]]
create_debug_port u_ila_0 probe
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe17]
set_property port_width 1 [get_debug_ports u_ila_0/probe17]
connect_debug_port u_ila_0/probe17 [get_nets [list CEP_IBUF]]
create_debug_port u_ila_0 probe
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe18]
set_property port_width 1 [get_debug_ports u_ila_0/probe18]
connect_debug_port u_ila_0/probe18 [get_nets [list CLK_IBUF]]
create_debug_port u_ila_0 probe
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe19]
set_property port_width 1 [get_debug_ports u_ila_0/probe19]
connect_debug_port u_ila_0/probe19 [get_nets [list RSTA_IBUF]]
create_debug_port u_ila_0 probe
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe20]
set_property port_width 1 [get_debug_ports u_ila_0/probe20]
connect_debug_port u_ila_0/probe20 [get_nets [list RSTB_IBUF]]
create_debug_port u_ila_0 probe
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe21]
set_property port_width 1 [get_debug_ports u_ila_0/probe21]
connect_debug_port u_ila_0/probe21 [get_nets [list RSTC_IBUF]]
create_debug_port u_ila_0 probe
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe22]
set_property port_width 1 [get_debug_ports u_ila_0/probe22]
connect_debug_port u_ila_0/probe22 [get_nets [list RSTCARRYIN_IBUF]]
create_debug_port u_ila_0 probe
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe23]
set_property port_width 1 [get_debug_ports u_ila_0/probe23]
connect_debug_port u_ila_0/probe23 [get_nets [list RSTD_IBUF]]
create_debug_port u_ila_0 probe
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe24]
set_property port_width 1 [get_debug_ports u_ila_0/probe24]
connect_debug_port u_ila_0/probe24 [get_nets [list RSTM_IBUF]]
create_debug_port u_ila_0 probe
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe25]
set_property port_width 1 [get_debug_ports u_ila_0/probe25]
connect_debug_port u_ila_0/probe25 [get_nets [list RSTOPMODE_IBUF]]
create_debug_port u_ila_0 probe
set_property PROBE_TYPE DATA_AND_TRIGGER [get_debug_ports u_ila_0/probe26]
set_property port_width 1 [get_debug_ports u_ila_0/probe26]
connect_debug_port u_ila_0/probe26 [get_nets [list RSTP_IBUF]]
set_property C_CLK_INPUT_FREQ_HZ 300000000 [get_debug_cores dbg_hub]
set_property C_ENABLE_CLK_DIVIDER false [get_debug_cores dbg_hub]
set_property C_USER_SCAN_CHAIN 1 [get_debug_cores dbg_hub]
connect_debug_port dbg_hub/clk [get_nets CLK_IBUF_BUFG]
```

6-Elaboration:

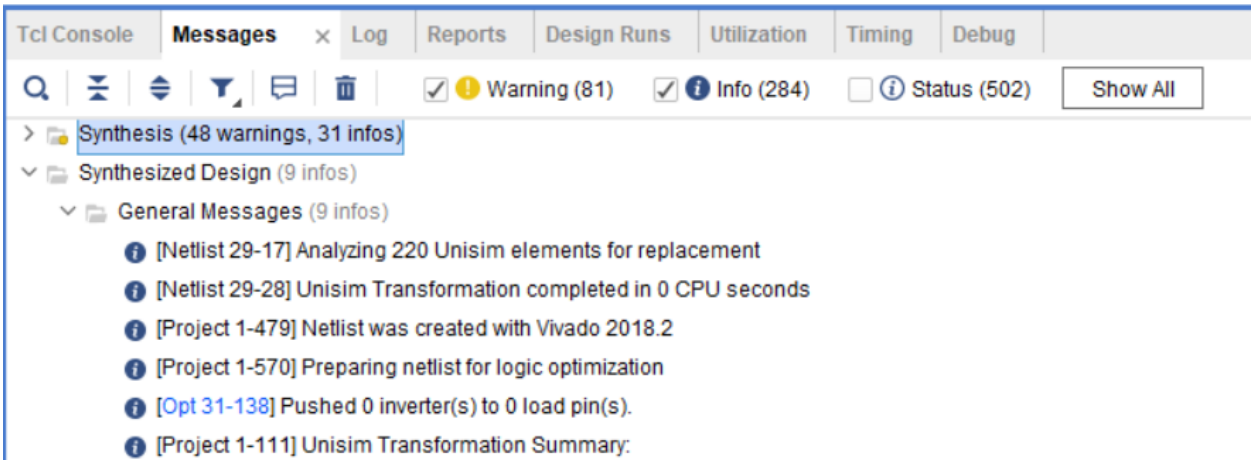
-Messages:

-Schematic:



7-Synthesis:

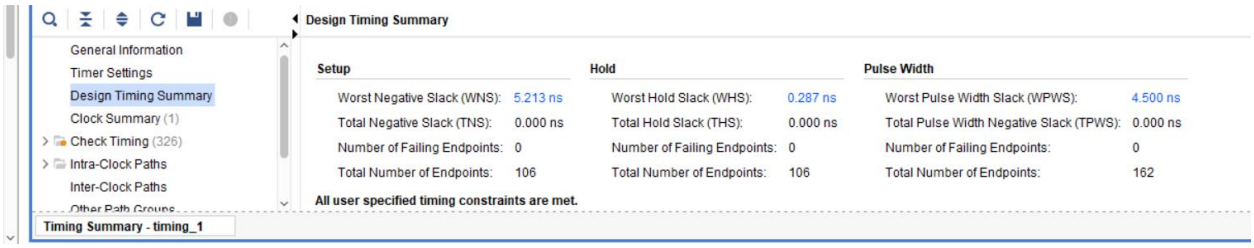
-Messages:



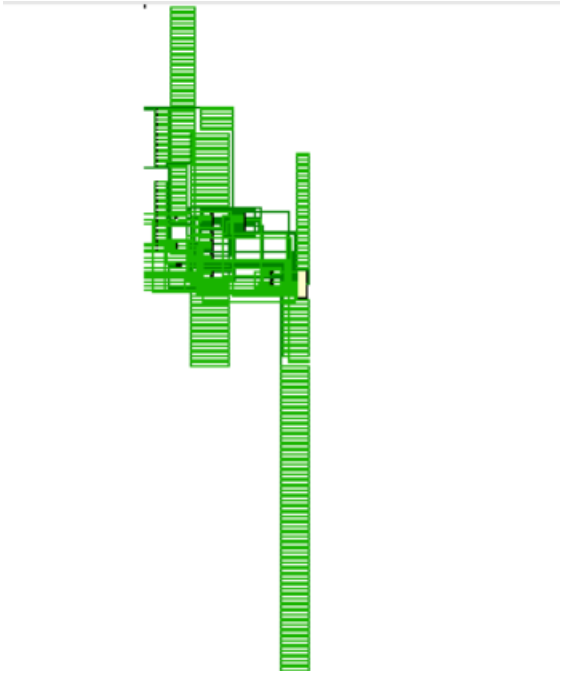
-Utilization report:

Name	Slice LUTs (134600)	Slice Registers (269200)	DSPs (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:

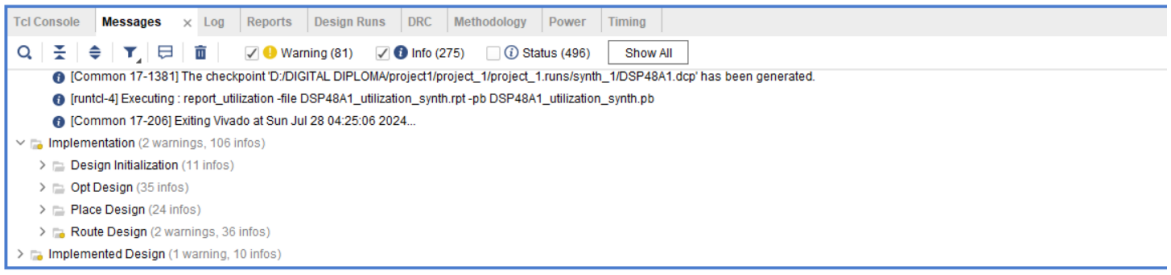


-Schematic:

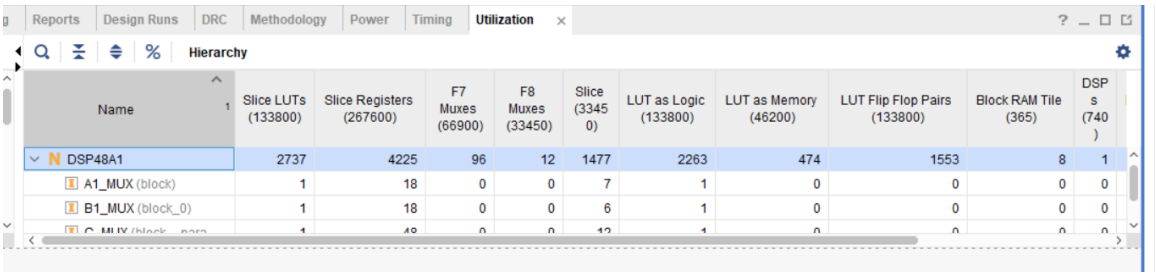


8-Implementation:

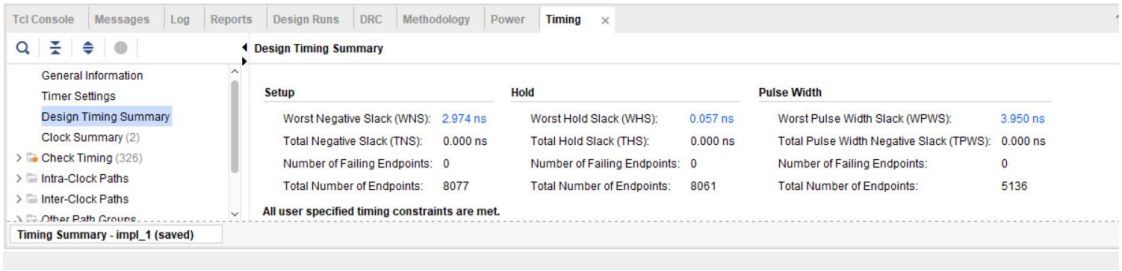
-Messages:



-Utilization report:



-Timing Report:



-Device: