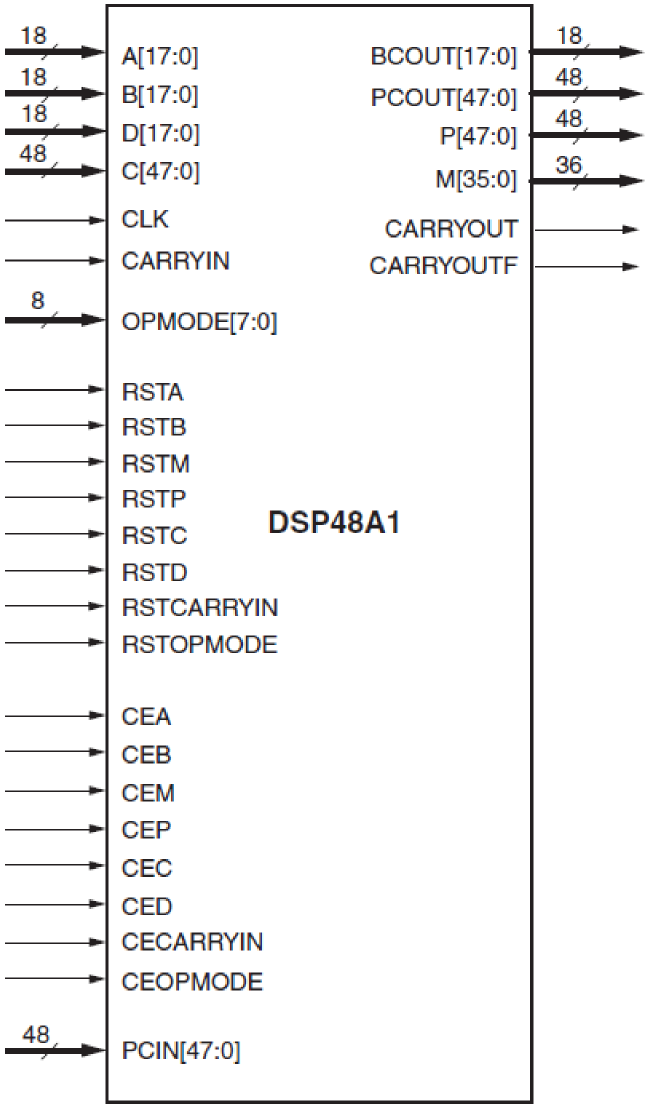
[Date]

[Project 1]

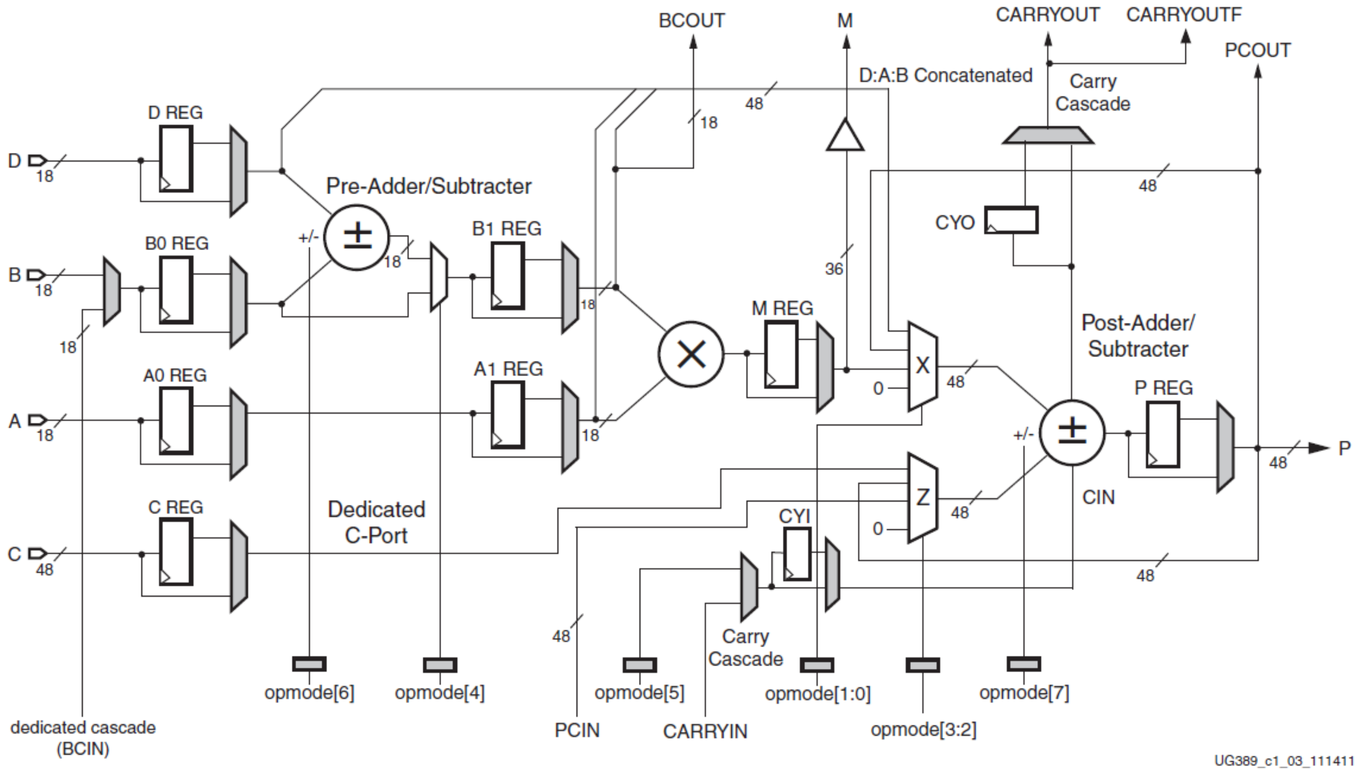
[Spartan6 - DSP48A1]

Mohamed Lotfy

1.DSP48A1 Block:



2. Internal Design:



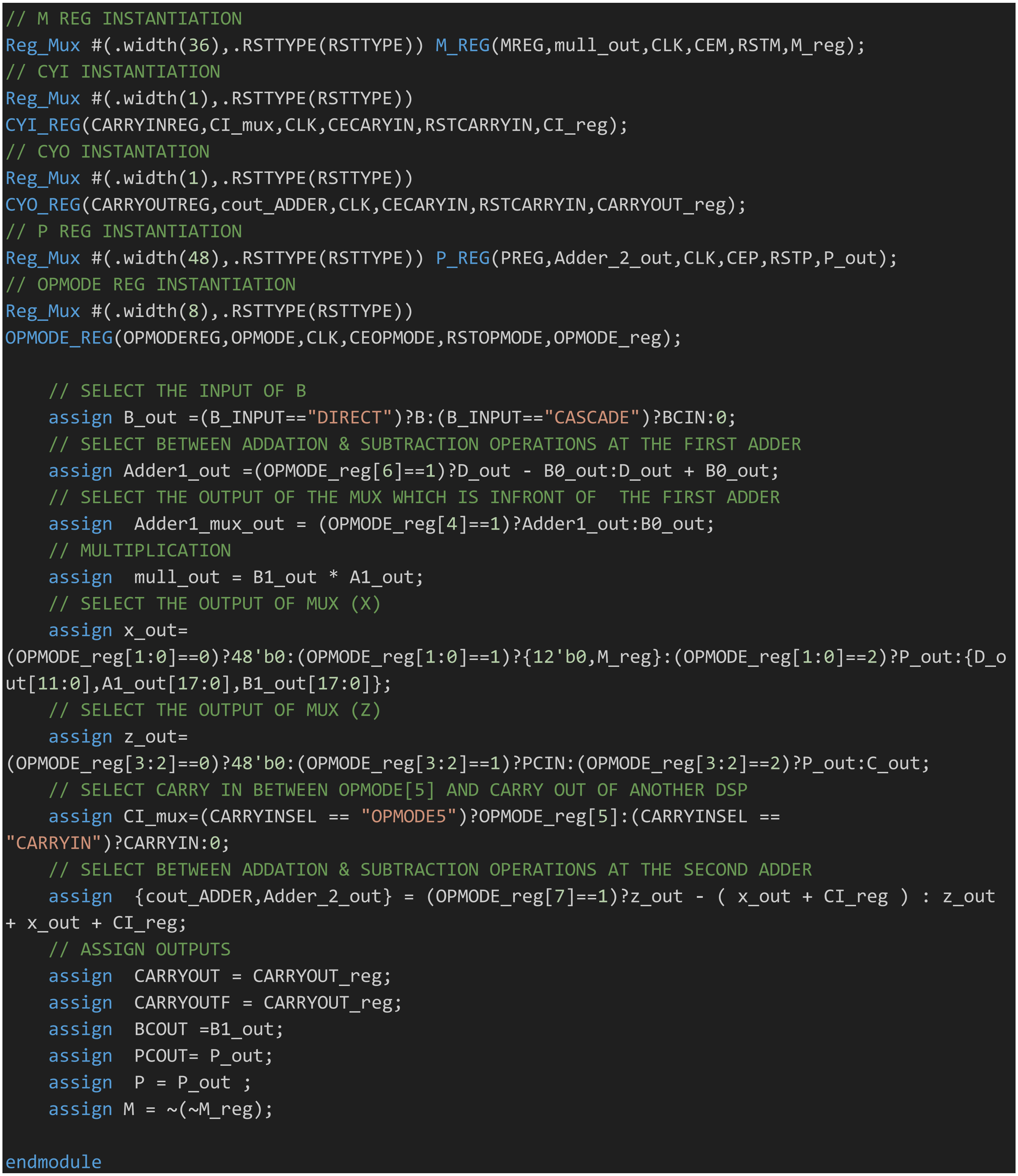
3. Verilog Code:

- Small Module:

```
module Reg_Mux (sel,in,clk,enable,rst,out);
parameter width = 1;
parameter RSTTYPE = "ASYNC";
input [width-1:0] in;
input sel,clk,rst,enable;
output [width-1:0] out;
reg [width-1:0] out_reg;

generate
    if(RSTTYPE == "SYNC" )begin
      always @(posedge clk) begin
          if(rst)begin
          out_reg <=0;
        end
        else if(enable)begin
            out_reg <= in;
        end
      end
    end
    else if(RSTTYPE == "ASYNC")begin
      always @(posedge clk , posedge rst) begin
        if(rst)begin
          out_reg <=0;
        end
        else if(enable) begin
          out_reg <= in;
        end
      end
    end
endgenerate
assign out = (sel==1)?out_reg:in;
endmodule
```

- RTL Code:

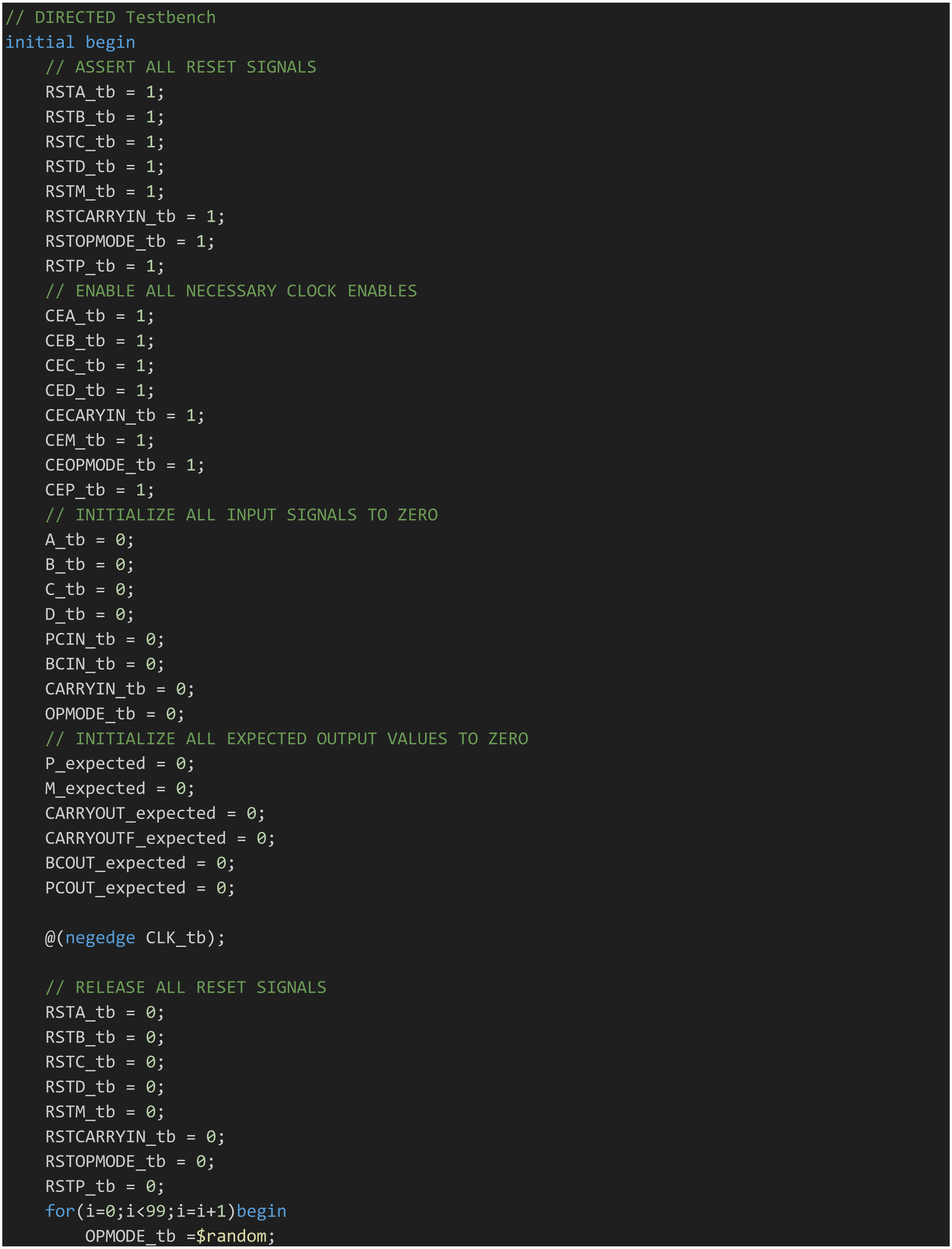
```
module DSP48A1 (A,B,C,D,CARRYIN,M,P,CARRYOUT,CARRYOUTF,CLK,OPMODE,
CEA,CEB,CEC,CED,CECARYIN,CEM,CEOPMODE,CEP,RSTA,RSTB,RSTC,RSTD
,RSTM,RSTCARRYIN,RSTOPMODE,RSTP,PCIN,BCIN,BCOUT,PCOUT);
// DEFINE PARAMETERS
parameter A0REG =0;
parameter A1REG =1;
parameter B0REG =0;
parameter B1REG =1;
parameter CREG =1;
parameter DREG =1;
parameter MREG =1;
parameter PREG =1;
parameter CARRYINREG =1;
parameter CARRYOUTREG =1;
parameter OPMODEREG =1;
parameter CARRYINSEL ="OPMODE5";
parameter B_INPUT ="DIRECT";
parameter RSTTYPE ="SYNC";
// DEFINE INPUTS
input [17:0] A,B,D,BCIN;
input [47:0] C,PCIN;
input [7:0] OPMODE;
input
CLK,CARRYIN,RSTA,RSTB,RSTC,RSTD,RSTM,RSTCARRYIN,RSTOPMODE,RSTP,CEA,CEB,CEC,CED,CECARYIN,CEM,CEOPMODE,CEP;
// DEFINE OUTPUTS
output [17:0] BCOUT;
output [47:0] PCOUT,P;
output [35:0] M;
output CARRYOUT,CARRYOUTF;
// DEFINE INTERNAL WIRES
wire [17:0] A0_out,A1_out,B_out,B0_out,B1_out,D_out,Adder1_out,Adder1_mux_out;
wire [47:0] C_out,x_out,z_out,Adder_2_out,P_out;
wire [7:0] OPMODE_reg;
wire [35:0] mull_out,M_reg;
wire cout_ADDER,CI_reg,CI_mux,CARRYOUT_reg;
// INSTANTIATIONS

// A REG INSTANTIATION
Reg_Mux #(.width(18),.RSTTYPE(RSTTYPE)) A0_REG(A0REG,A,CLK,CEA,RSTA,A0_out);
Reg_Mux #(.width(18),.RSTTYPE(RSTTYPE)) A1_REG(A1REG,A0_out,CLK,CEA,RSTA,A1_out);
// B REG INSTANTIATION
Reg_Mux #(.width(18),.RSTTYPE(RSTTYPE)) B0_REG(B0REG,B_out,CLK,CEB,RSTB,B0_out);
Reg_Mux #(.width(18),.RSTTYPE(RSTTYPE)) B1_REG(B1REG,Adder1_mux_out,CLK,CEB,RSTB,B1_out);
// C REG INSTANTIATION
Reg_Mux #(.width(48),.RSTTYPE(RSTTYPE)) C_REG(CREG,C,CLK,CEC,RSTC,C_out);
// D REG INSTANTIATION
Reg_Mux #(.width(18),.RSTTYPE(RSTTYPE)) D_REG(DREG,D,CLK,CED,RSTD,D_out);
```



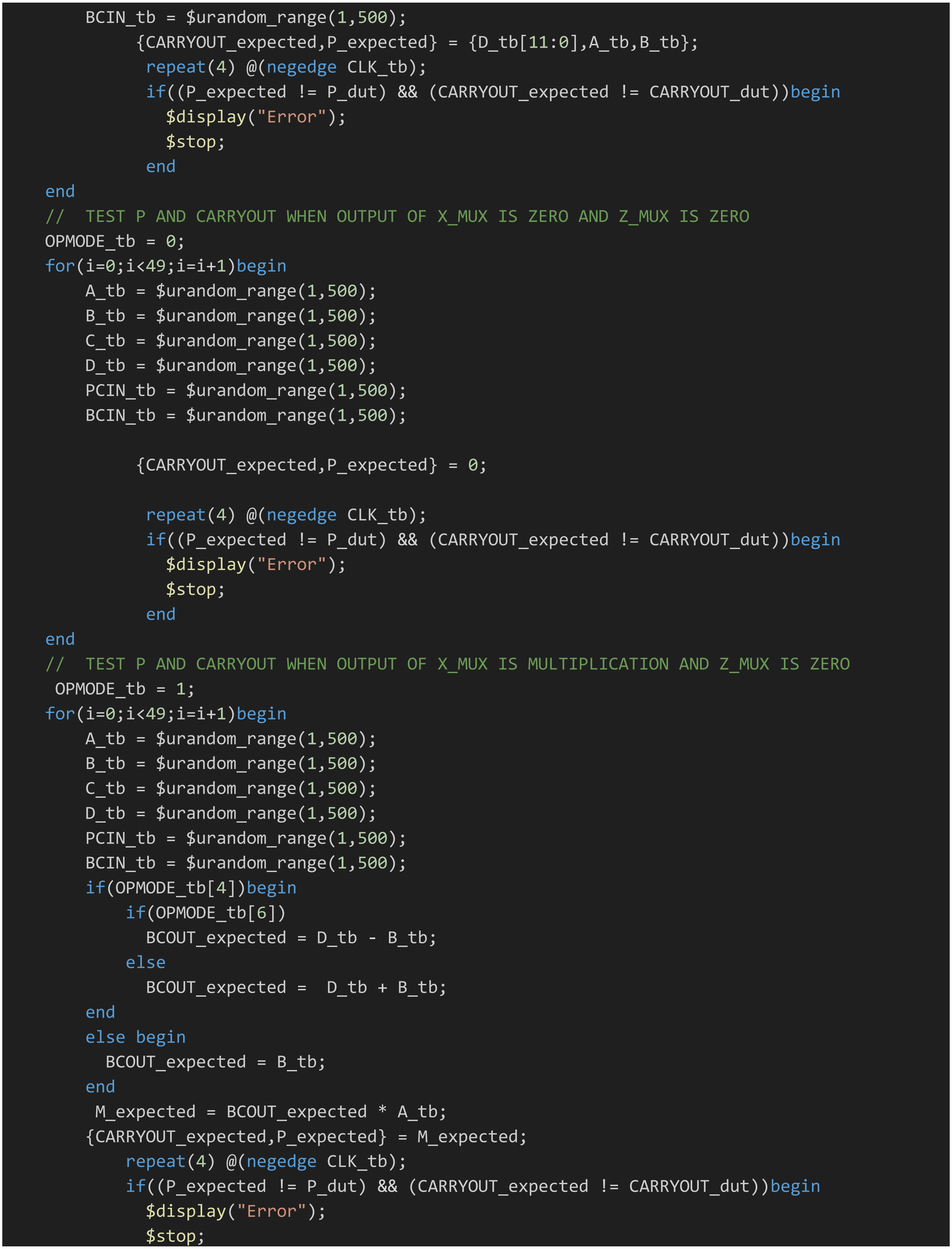
```
// M REG INSTANTIATION
Reg_Mux #(.width(36),.RSTTYPE(RSTTYPE)) M_REG(MREG,mull_out,CLK,CEM,RSTM,M_reg);
// CYI INSTANTIATION
Reg_Mux #(.width(1),.RSTTYPE(RSTTYPE))
CYI_REG(CARRYINREG,CI_mux,CLK,CECARYIN,RSTCARRYIN,CI_reg);
// CYO INSTANTIATION
Reg_Mux #(.width(1),.RSTTYPE(RSTTYPE))
CYO_REG(CARRYOUTREG,cout_ADDER,CLK,CECARYIN,RSTCARRYIN,CARRYOUT_reg);
// P REG INSTANTIATION
Reg_Mux #(.width(48),.RSTTYPE(RSTTYPE)) P_REG(PREG,Adder_2_out,CLK,CEP,RSTP,P_out);
// OPMODE REG INSTANTIATION
Reg_Mux #(.width(8),.RSTTYPE(RSTTYPE))
OPMODE_REG(OPMODEREG,OPMODE,CLK,CEOPMODE,RSTOPMODE,OPMODE_reg);

    // SELECT THE INPUT OF B
    assign B_out =(B_INPUT=="DIRECT")?B:(B_INPUT=="CASCADE")?BCIN:0;
    // SELECT BETWEEN ADDATION & SUBTRACTION OPERATIONS AT THE FIRST ADDER
    assign Adder1_out =(OPMODE_reg[6]==1)?D_out - B0_out:D_out + B0_out;
    // SELECT THE OUTPUT OF THE MUX WHICH IS INFRONT OF  THE FIRST ADDER
    assign  Adder1_mux_out = (OPMODE_reg[4]==1)?Adder1_out:B0_out;
    // MULTIPLICATION
    assign  mull_out = B1_out * A1_out;
    // SELECT THE OUTPUT OF MUX (X)
    assign x_out=
(OPMODE_reg[1:0]==0)?48'b0:(OPMODE_reg[1:0]==1)?{12'b0,M_reg}:(OPMODE_reg[1:0]==2)?P_out:{D_out[11:0],A1_out[17:0],B1_out[17:0]};
    // SELECT THE OUTPUT OF MUX (Z)
    assign z_out=
(OPMODE_reg[3:2]==0)?48'b0:(OPMODE_reg[3:2]==1)?PCIN:(OPMODE_reg[3:2]==2)?P_out:C_out;
    // SELECT CARRY IN BETWEEN OPMODE[5] AND CARRY OUT OF ANOTHER DSP
    assign CI_mux=(CARRYINSEL == "OPMODE5")?OPMODE_reg[5]:(CARRYINSEL ==
"CARRYIN")?CARRYIN:0;
    // SELECT BETWEEN ADDATION & SUBTRACTION OPERATIONS AT THE SECOND ADDER
    assign  {cout_ADDER,Adder_2_out} = (OPMODE_reg[7]==1)?z_out - ( x_out + CI_reg ) : z_out
+ x_out + CI_reg;
    // ASSIGN OUTPUTS
    assign  CARRYOUT = CARRYOUT_reg;
    assign  CARRYOUTF = CARRYOUT_reg;
    assign  BCOUT =B1_out;
    assign  PCOUT= P_out;
    assign  P = P_out ;
    assign M = ~(~M_reg);

endmodule
```

- Testbench Code:

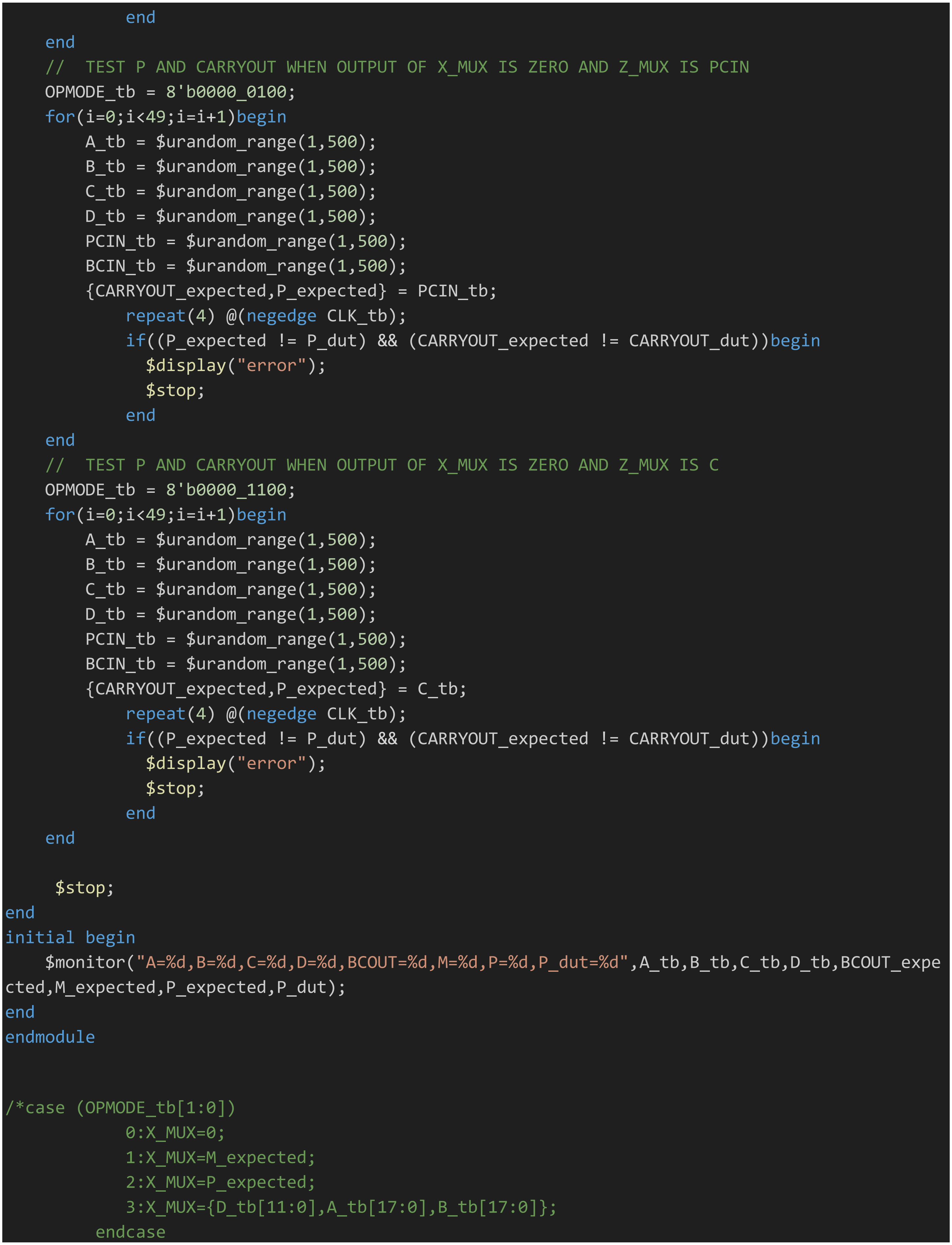
```
module DSP48A1_tb ();
    // DEFINE PARAMETERS
parameter A0REG_tb =0;
parameter A1REG_tb =1;
parameter B0REG_tb =0;
parameter B1REG_tb =1;
parameter CREG_tb =1;
parameter DREG_tb =1;
parameter MREG_tb =1;
parameter PREG_tb =1;
parameter CARRYINREG_tb =1;
parameter CARRYOUTREG_tb =1;
parameter OPMODEREG_tb =1;
parameter CARRYINSEL_tb ="OPMODE5";
parameter B_INPUT_tb ="DIRECT";
parameter RSTTYPE_tb ="SYNC";
// DEFINE INPUTS
reg [17:0] A_tb,B_tb,D_tb,BCIN_tb;
reg [47:0] C_tb,PCIN_tb;
reg [7:0] OPMODE_tb;
reg CLK_tb,CARRYIN_tb,RSTA_tb,RSTB_tb,RSTC_tb,RSTD_tb,RSTM_tb,RSTCARRYIN_tb
,RSTOPMODE_tb,RSTP_tb,CEA_tb,CEB_tb,CEC_tb,CED_tb,CECARYIN_tb,CEM_tb,CEOPMODE_tb,CEP_tb;
// DEFINE INSTANTIATION optputs
wire [17:0] BCOUT_dut;
wire [47:0] PCOUT_dut,P_dut;
wire [35:0] M_dut;
wire CARRYOUT_dut,CARRYOUTF_dut;
// DEFINE EXPECTED OUTPUT
reg [17:0] BCOUT_expected;
reg [47:0] PCOUT_expected,P_expected;
reg [35:0] M_expected;
reg CARRYOUT_expected,CARRYOUTF_expected;
// MODULE INSTANTIATION
DSP48A1
DUT(A_tb,B_tb,C_tb,D_tb,CARRYIN_tb,M_dut,P_dut,CARRYOUT_dut,CARRYOUTF_dut,CLK_tb,OPMODE_tb,
CEA_tb,CEB_tb,CEC_tb,CED_tb,CECARYIN_tb,CEM_tb,CEOPMODE_tb,CEP_tb,RSTA_tb,RSTB_tb,RSTC_tb,RST
D_tb
,RSTM_tb,RSTCARRYIN_tb,RSTOPMODE_tb,RSTP_tb,PCIN_tb,BCIN_tb,BCOUT_dut,PCOUT_dut);

integer i;
// CLOCK GENERATION
initial begin
    CLK_tb =0;
    forever begin
        #1 CLK_tb =~ CLK_tb;
    end
end
```



```
// DIRECTED Testbench
initial begin
    // ASSERT ALL RESET SIGNALS
    RSTA_tb = 1;
    RSTB_tb = 1;
    RSTC_tb = 1;
    RSTD_tb = 1;
    RSTM_tb = 1;
    RSTCARRYIN_tb = 1;
    RSTOPMODE_tb = 1;
    RSTP_tb = 1;
    // ENABLE ALL NECESSARY CLOCK ENABLES
    CEA_tb = 1;
    CEB_tb = 1;
    CEC_tb = 1;
    CED_tb = 1;
    CECARYIN_tb = 1;
    CEM_tb = 1;
    CEOPMODE_tb = 1;
    CEP_tb = 1;
    // INITIALIZE ALL INPUT SIGNALS TO ZERO
    A_tb = 0;
    B_tb = 0;
    C_tb = 0;
    D_tb = 0;
    PCIN_tb = 0;
    BCIN_tb = 0;
    CARRYIN_tb = 0;
    OPMODE_tb = 0;
    // INITIALIZE ALL EXPECTED OUTPUT VALUES TO ZERO
    P_expected = 0;
    M_expected = 0;
    CARRYOUT_expected = 0;
    CARRYOUTF_expected = 0;
    BCOUT_expected = 0;
    PCOUT_expected = 0;

    @(negedge CLK_tb);

    // RELEASE ALL RESET SIGNALS
    RSTA_tb = 0;
    RSTB_tb = 0;
    RSTC_tb = 0;
    RSTD_tb = 0;
    RSTM_tb = 0;
    RSTCARRYIN_tb = 0;
    RSTOPMODE_tb = 0;
    RSTP_tb = 0;
    for(i=0;i<99;i=i+1)begin
        OPMODE_tb =$random;
        A_tb = $urandom_range(1,500);
        B_tb = $urandom_range(1,500);
        C_tb = $urandom_range(1,500);
        D_tb = $urandom_range(1,500);
        PCIN_tb = $urandom_range(1,500);
        BCIN_tb = $urandom_range(1,500);
        CARRYIN_tb = $random;
         repeat(3) @(negedge CLK_tb);
        // TEST BOUT OUTPUT
        if(OPMODE_tb[4])begin
            if(OPMODE_tb[6])
              BCOUT_expected = D_tb - B_tb;
            else
              BCOUT_expected =  D_tb + B_tb;
        end
        else begin
          BCOUT_expected = B_tb;
        end
        // TEST MULTIPLICATION OUTPUT
         M_expected = BCOUT_expected * A_tb;
        #20;
        if((M_expected != M_dut) || (BCOUT_expected != BCOUT_dut))begin
          $display("Error");
          $stop;
        end
    end
    //  TEST P AND CARRYOUT AT OPMODE 0000_0010
    OPMODE_tb = 2;
    for(i=0;i<49;i=i+1)begin
        A_tb = $urandom_range(1,500);
        B_tb = $urandom_range(1,500);
        C_tb = $urandom_range(1,500);
        D_tb = $urandom_range(1,500);
        PCIN_tb = $urandom_range(1,500);
        BCIN_tb = $urandom_range(1,500);
         {CARRYOUT_expected,P_expected} = P_expected;
            repeat(4) @(negedge CLK_tb);
            if((P_expected != P_dut) && (CARRYOUT_expected != CARRYOUT_dut))begin
              $display("Error");
              $stop;
            end
    end
    //  TEST P AND CARRYOUT WHEN OUTPUT OF X_MUX IS {D_tb[11:0],A_tb,B_tb} AND Z_MUX IS ZERO
    OPMODE_tb = 3;
    for(i=0;i<49;i=i+1)begin
        A_tb = $urandom_range(1,500);
        B_tb = $urandom_range(1,500);
        C_tb = $urandom_range(1,500);
        D_tb = $urandom_range(1,500);
        PCIN_tb = $urandom_range(1,500);
```



```
        BCIN_tb = $urandom_range(1,500);
             {CARRYOUT_expected,P_expected} = {D_tb[11:0],A_tb,B_tb};
              repeat(4) @(negedge CLK_tb);
              if((P_expected != P_dut) && (CARRYOUT_expected != CARRYOUT_dut))begin
                $display("Error");
                $stop;
              end
    end
    //  TEST P AND CARRYOUT WHEN OUTPUT OF X_MUX IS ZERO AND Z_MUX IS ZERO
    OPMODE_tb = 0;
    for(i=0;i<49;i=i+1)begin
        A_tb = $urandom_range(1,500);
        B_tb = $urandom_range(1,500);
        C_tb = $urandom_range(1,500);
        D_tb = $urandom_range(1,500);
        PCIN_tb = $urandom_range(1,500);
        BCIN_tb = $urandom_range(1,500);

             {CARRYOUT_expected,P_expected} = 0;

              repeat(4) @(negedge CLK_tb);
              if((P_expected != P_dut) && (CARRYOUT_expected != CARRYOUT_dut))begin
                $display("Error");
                $stop;
              end
    end
    //  TEST P AND CARRYOUT WHEN OUTPUT OF X_MUX IS MULTIPLICATION AND Z_MUX IS ZERO
     OPMODE_tb = 1;
    for(i=0;i<49;i=i+1)begin
        A_tb = $urandom_range(1,500);
        B_tb = $urandom_range(1,500);
        C_tb = $urandom_range(1,500);
        D_tb = $urandom_range(1,500);
        PCIN_tb = $urandom_range(1,500);
        BCIN_tb = $urandom_range(1,500);
        if(OPMODE_tb[4])begin
            if(OPMODE_tb[6])
              BCOUT_expected = D_tb - B_tb;
            else
              BCOUT_expected =  D_tb + B_tb;
        end
        else begin
          BCOUT_expected = B_tb;
        end
         M_expected = BCOUT_expected * A_tb;
        {CARRYOUT_expected,P_expected} = M_expected;
            repeat(4) @(negedge CLK_tb);
            if((P_expected != P_dut) && (CARRYOUT_expected != CARRYOUT_dut))begin
              $display("Error");
              $stop;
```



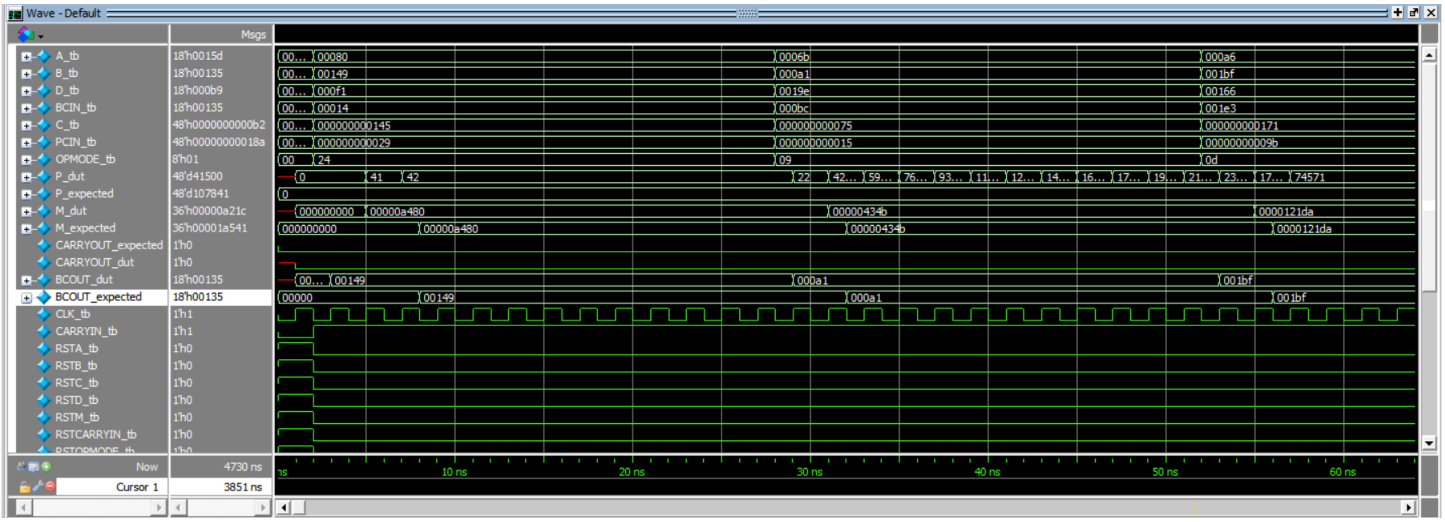
```
            end
    end
    //  TEST P AND CARRYOUT WHEN OUTPUT OF X_MUX IS ZERO AND Z_MUX IS PCIN
    OPMODE_tb = 8'b0000_0100;
    for(i=0;i<49;i=i+1)begin
        A_tb = $urandom_range(1,500);
        B_tb = $urandom_range(1,500);
        C_tb = $urandom_range(1,500);
        D_tb = $urandom_range(1,500);
        PCIN_tb = $urandom_range(1,500);
        BCIN_tb = $urandom_range(1,500);
        {CARRYOUT_expected,P_expected} = PCIN_tb;
            repeat(4) @(negedge CLK_tb);
            if((P_expected != P_dut) && (CARRYOUT_expected != CARRYOUT_dut))begin
              $display("error");
              $stop;
            end
    end
    //  TEST P AND CARRYOUT WHEN OUTPUT OF X_MUX IS ZERO AND Z_MUX IS C
    OPMODE_tb = 8'b0000_1100;
    for(i=0;i<49;i=i+1)begin
        A_tb = $urandom_range(1,500);
        B_tb = $urandom_range(1,500);
        C_tb = $urandom_range(1,500);
        D_tb = $urandom_range(1,500);
        PCIN_tb = $urandom_range(1,500);
        BCIN_tb = $urandom_range(1,500);
        {CARRYOUT_expected,P_expected} = C_tb;
            repeat(4) @(negedge CLK_tb);
            if((P_expected != P_dut) && (CARRYOUT_expected != CARRYOUT_dut))begin
              $display("error");
              $stop;
            end
    end

     $stop;
end
initial begin
    $monitor("A=%d,B=%d,C=%d,D=%d,BCOUT=%d,M=%d,P=%d,P_dut=%d",A_tb,B_tb,C_tb,D_tb,BCOUT_expe
cted,M_expected,P_expected,P_dut);
end
endmodule


/*case (OPMODE_tb[1:0])
            0:X_MUX=0;
            1:X_MUX=M_expected;
            2:X_MUX=P_expected;
            3:X_MUX={D_tb[11:0],A_tb[17:0],B_tb[17:0]};
         endcase
        case (OPMODE_tb[3:2])
           0:Z_MUX=0;
           1:Z_MUX=PCIN_tb;
           2:Z_MUX=P_expected;
           3:Z_MUX=C_tb;
        endcase

        if(OPMODE_tb[7])begin
          if(CARRYINSEL_tb == "OPMODE5")
             {CARRYOUT_expected,P_expected} = Z_MUX - (X_MUX + OPMODE_tb[5]);
           else if(CARRYINSEL_tb == "CARRYIN")
            {CARRYOUT_expected,P_expected} =  Z_MUX - (X_MUX + CARRYIN_tb );
        end
        else begin
          if(CARRYINSEL_tb == "OPMODE5")
             {CARRYOUT_expected,P_expected} = Z_MUX + (X_MUX + OPMODE_tb[5] );
           else if(CARRYINSEL_tb == "CARRYIN")
            {CARRYOUT_expected,P_expected} = Z_MUX + (X_MUX + CARRYIN_tb );
        end*/
```

4. Simulation Tool:

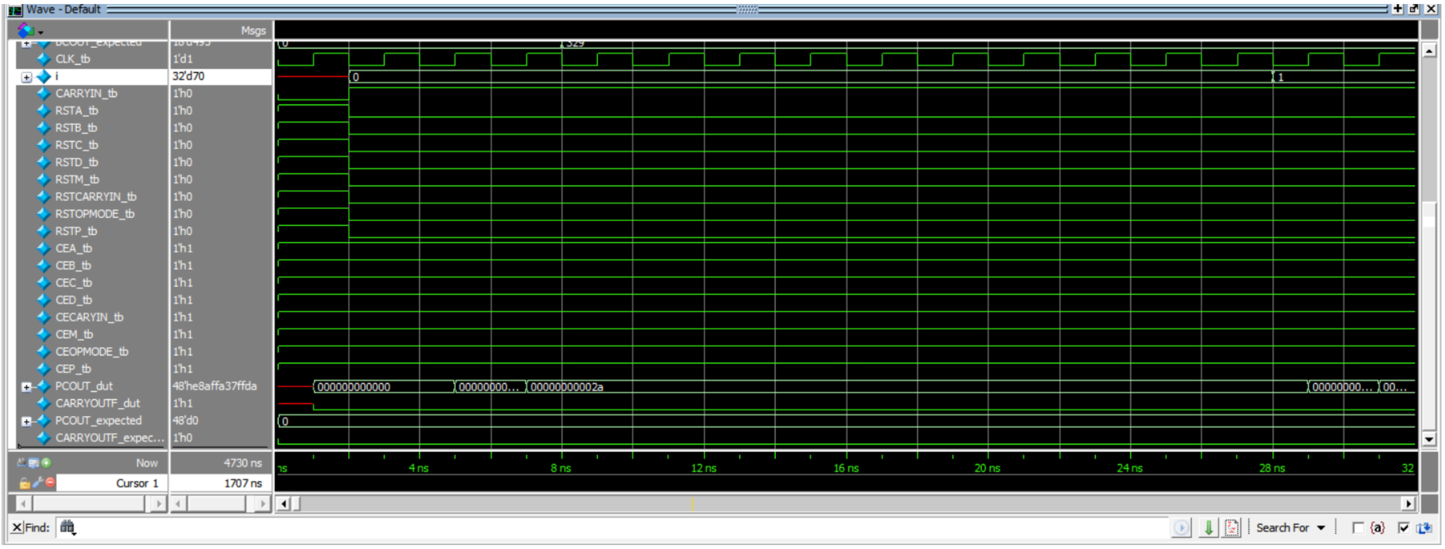
- DO File:

```
vlib work
vlog REG_MUX.v DSP.v DSP_tb.v
vsim -voptargs=+acc work.DSP48A1_tb
add wave *
run -all
//quit -sim
```

- Wave Form:
    - Test M & BOUT outputs

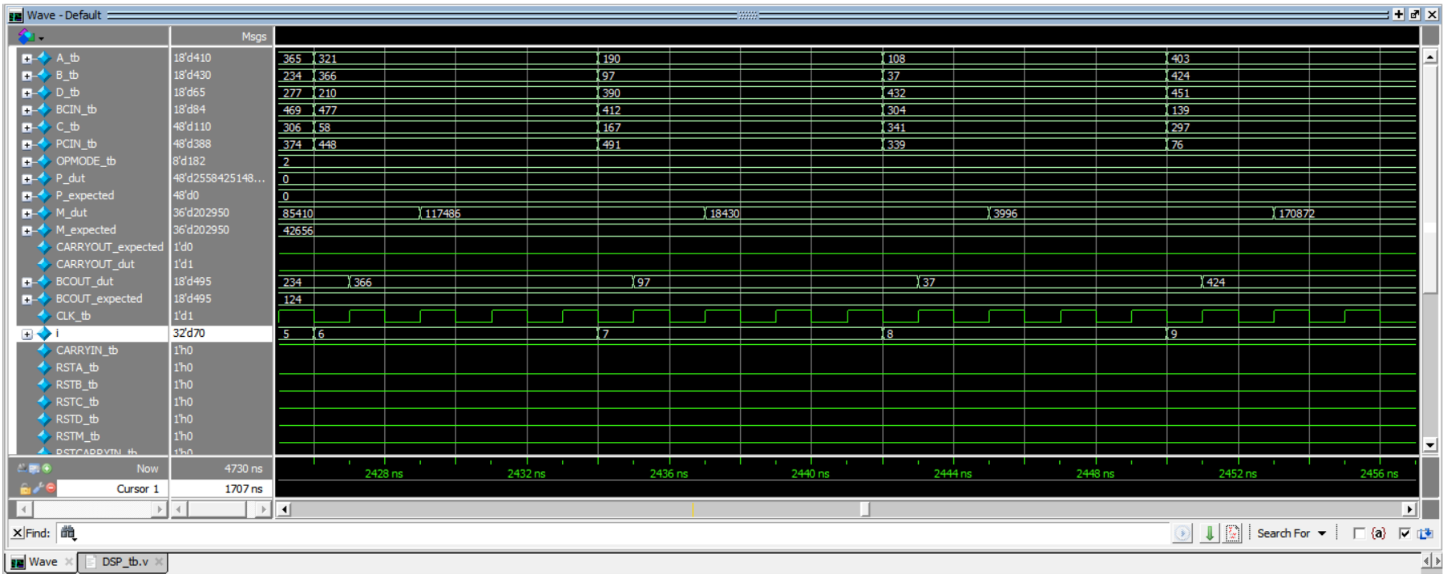


```
Wave - Default
                        Msgs
A_tb                    18'h0015d
B_tb                    18'h00135
D_tb                    18'h000b9
BCIN_tb                 18'h00135
C_tb                    48'h0000000000b2
PCIN_tb                 48'h00000000018a
OPMODE_tb               8'h01
P_dut                   48'd41500
P_expected              48'd107841
M_dut                   36'h00000a21c
M_expected              36'h00001a541
CARRYOUT_expected       1'h0
CARRYOUT_dut            1'h0
BCOUT_dut               18'h00135
BCOUT_expected          18'h00135
CLK_tb                  1'h1
CARRYIN_tb              1'h1
RSTA_tb                 1'h0
RSTB_tb                 1'h0
RSTC_tb                 1'h0
RSTD_tb                 1'h0
RSTM_tb                 1'h0
RSTCARRYIN_tb           1'h0
RSTOPMODE_tb            1'h0
Now                     4730 ns
Cursor 1                3851 ns
```

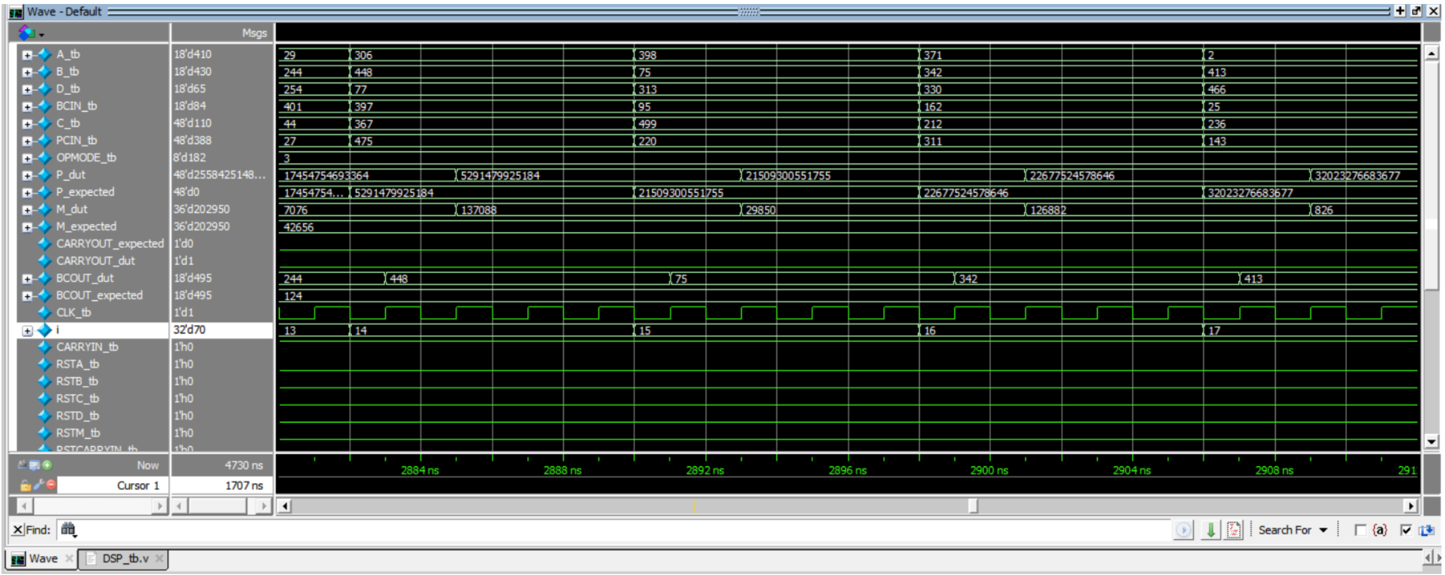


```
Wave - Default
                        Msgs
CLK_tb                  1'd1
i                       32'd70
CARRYIN_tb              1'h0
RSTA_tb                 1'h0
RSTB_tb                 1'h0
RSTC_tb                 1'h0
RSTD_tb                 1'h0
RSTM_tb                 1'h0
RSTCARRYIN_tb           1'h0
RSTOPMODE_tb            1'h0
RSTP_tb                 1'h0
CEA_tb                  1'h1
CEB_tb                  1'h1
CEC_tb                  1'h1
CED_tb                  1'h1
CECARYIN_tb             1'h1
CEM_tb                  1'h1
CEOPMODE_tb             1'h1
CEP_tb                  1'h1
PCOUT_dut               48'he8affa37ffda
CARRYOUTF_dut           1'h1
PCOUT_expected          48'd0
CARRYOUTF_expec...      1'h0
Now                     4730 ns
Cursor 1                1707 ns
```

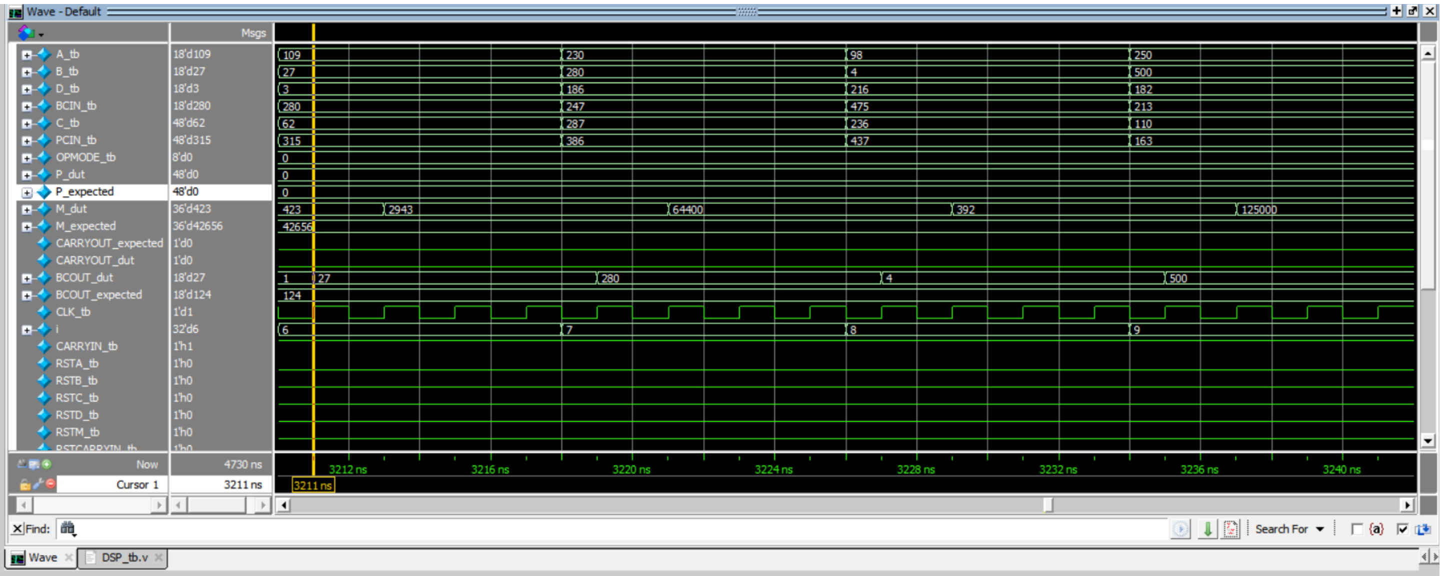
- Test P at opmode = 0



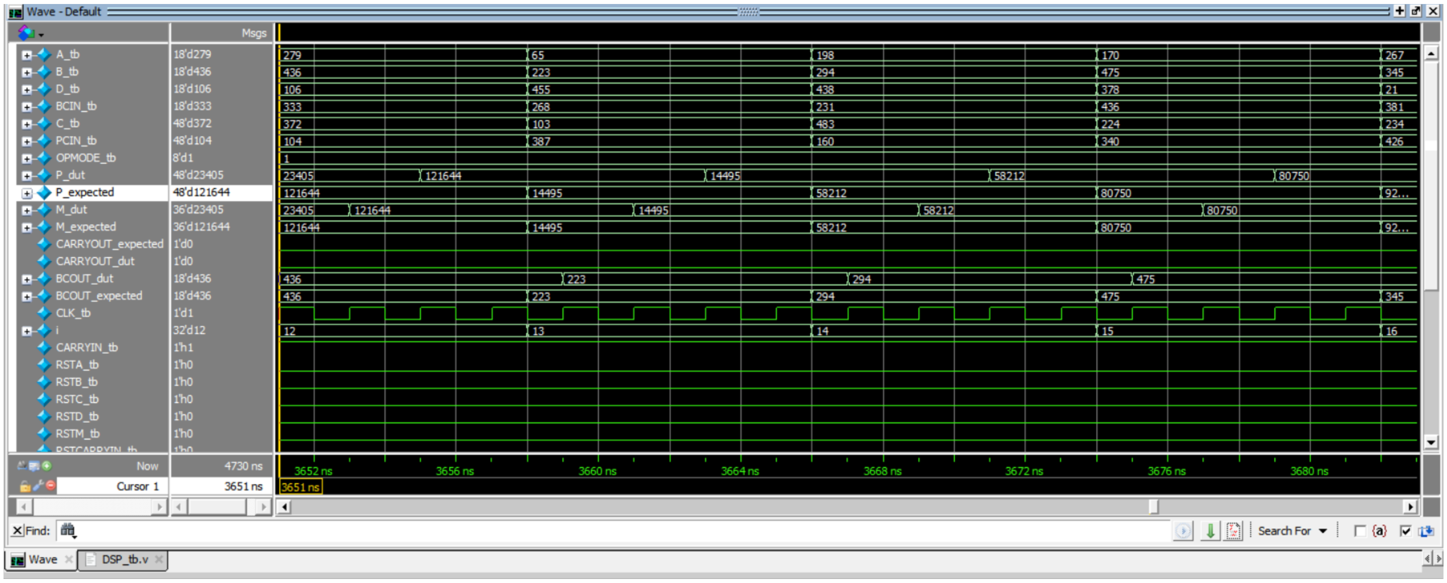
- Test P & CARRYOUT when outputs of X_MUX is {D_tb[11:0],A,B} and Z_MUX is zero



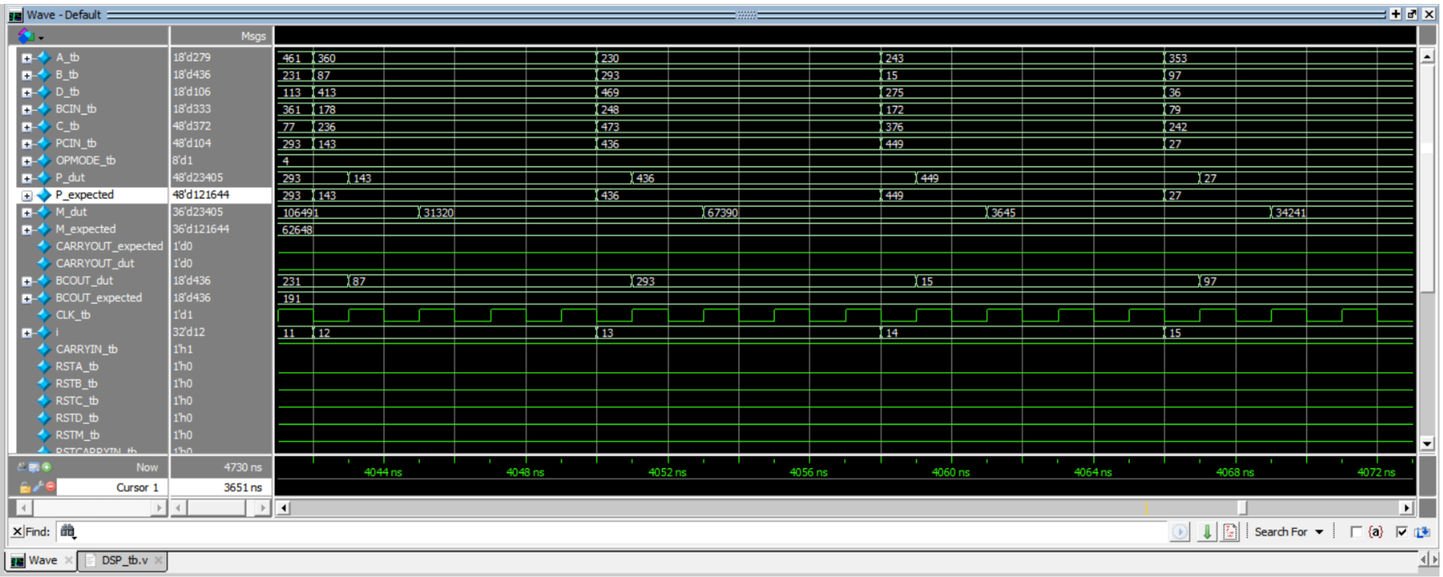
- Test P & CARRYOUT when output of X_MUX is zero and Z_MUX is zero:



- Test P & CARRYOUT when output of X_MUX is M and Z_MUX is zero:



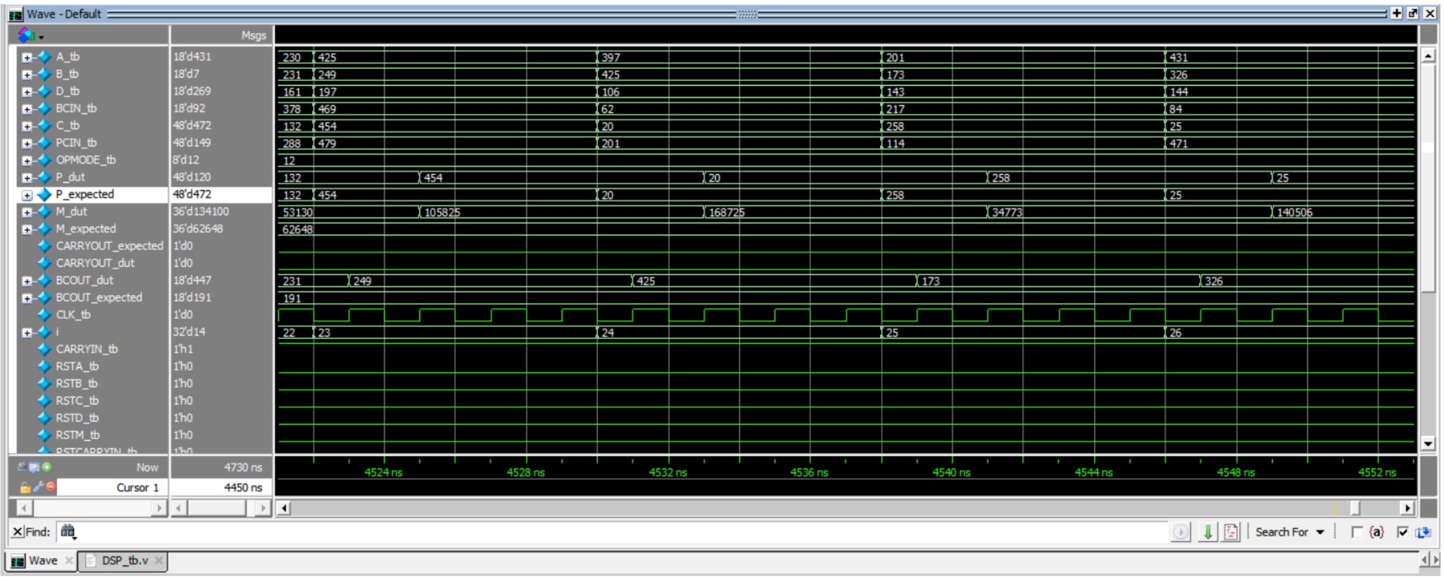
- Test P & CARRYOUT when output of X_MUX is 0 and Z_MUX is PCIN:



Wave - Default

	Msgs
A_tb	18'd279
B_tb	18'd436
D_tb	18'd106
BCIN_tb	18'd333
C_tb	48'd372
PCIN_tb	48'd104
OPMODE_tb	8'd1
P_dut	48'd23405
P_expected	48'd121644
M_dut	36'd23405
M_expected	36'd121644
CARRYOUT_expected	1'd0
CARRYOUT_dut	1'd0
BCOUT_dut	18'd436
BCOUT_expected	18'd436
CLK_tb	1'd1
i	32'd12
CARRYIN_tb	1'h1
RSTA_tb	1'h0
RSTB_tb	1'h0
RSTC_tb	1'h0
RSTD_tb	1'h0
RSTM_tb	1'h0
RSTCARRYIN_tb	1'h0
Now	4730 ns
Cursor 1	3651 ns

- Test P & CARRYOUT when output of X_MUX is 0 and Z_MUX is C:



Wave - Default

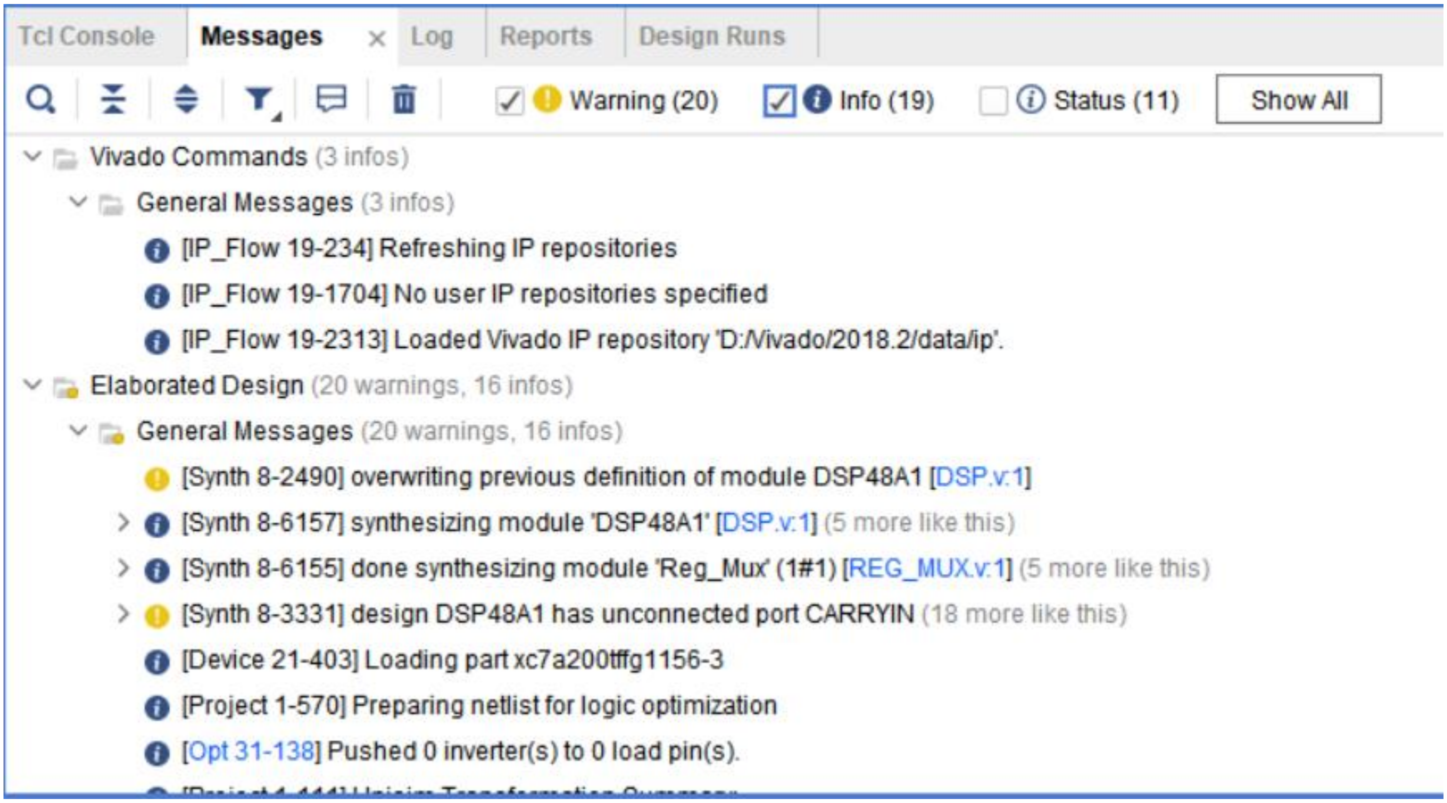
	Msgs
A_tb	18'd431
B_tb	18'd7
D_tb	18'd269
BCIN_tb	18'd92
C_tb	48'd472
PCIN_tb	48'd149
OPMODE_tb	8'd12
P_dut	48'd120
P_expected	48'd472
M_dut	36'd134100
M_expected	36'd62648
CARRYOUT_expected	1'd0
CARRYOUT_dut	1'd0
BCOUT_dut	18'd447
BCOUT_expected	18'd191
CLK_tb	1'd0
i	32'd14
CARRYIN_tb	1'h1
RSTA_tb	1'h0
RSTB_tb	1'h0
RSTC_tb	1'h0
RSTD_tb	1'h0
RSTM_tb	1'h0
RSTCARRYIN_tb	1'h0
Now	4730 ns
Cursor 1	4450 ns

5. Synthesis Tool:

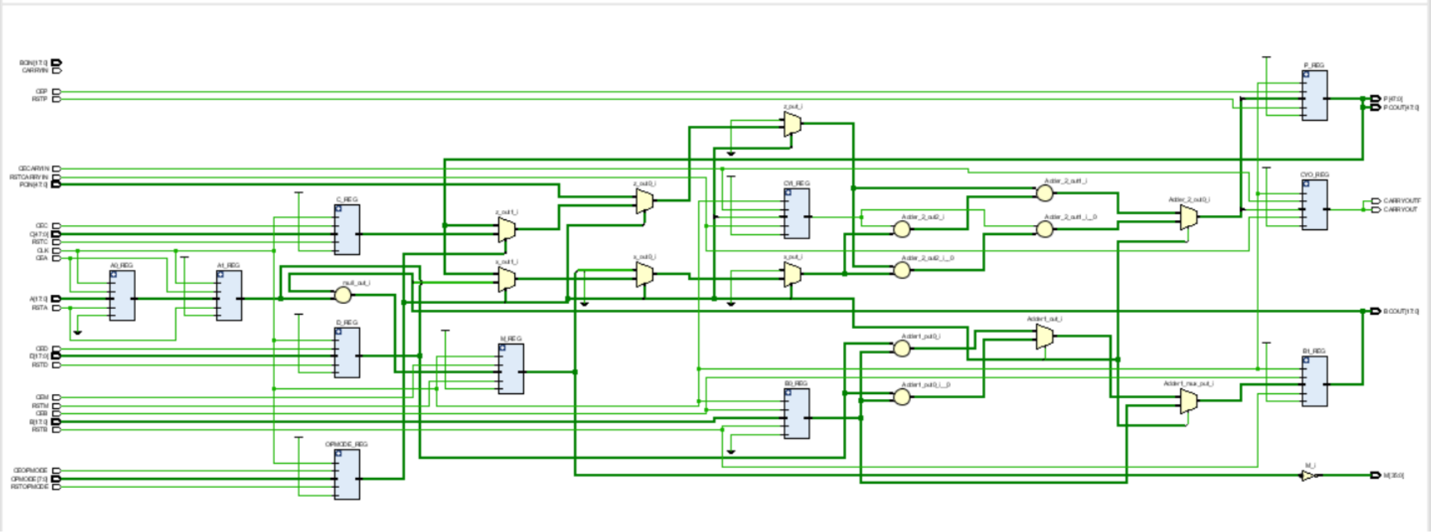
- Constraint File:

```
## 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]

## Configuration options, can be used for all designs
set_property CONFIG_VOLTAGE 3.3 [current_design]
set_property CFGBVS VCCO [current_design]

## SPI configuration mode options for QSPI boot, can be used for all designs
set_property BITSTREAM.GENERAL.COMPRESS TRUE [current_design]
set_property BITSTREAM.CONFIG.CONFIGRATE 33 [current_design]
set_property CONFIG_MODE SPIx4 [current_design]
```

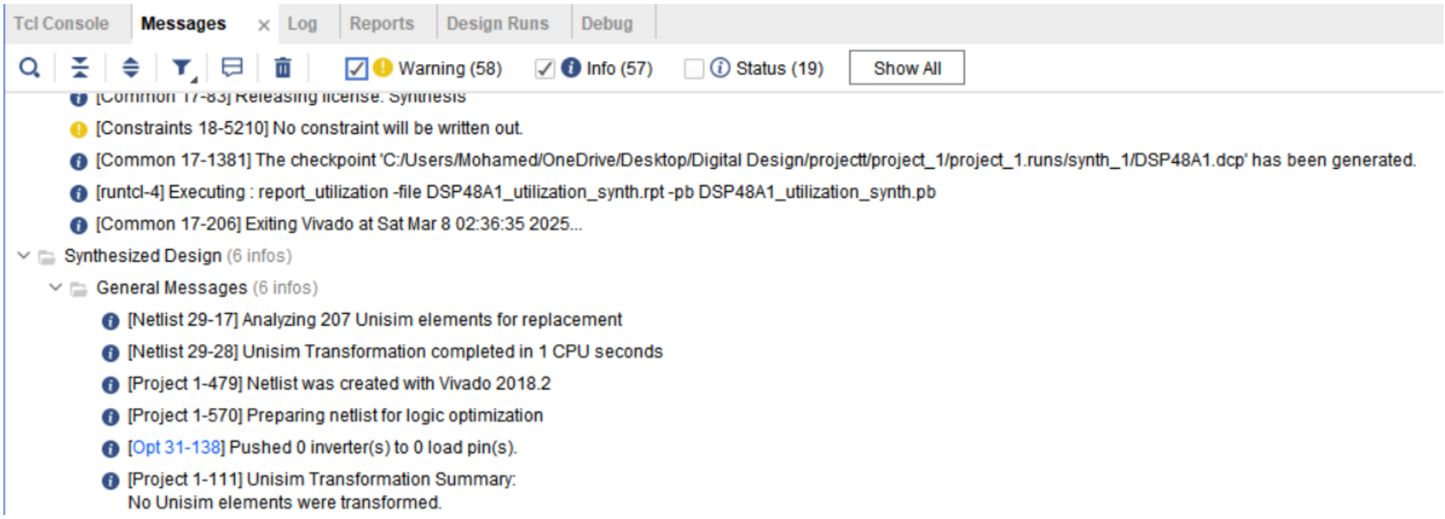
- Elaboration Messages & Schematic:



```
Vivado Commands (3 infos)
  General Messages (3 infos)
    [IP_Flow 19-234] Refreshing IP repositories
    [IP_Flow 19-1704] No user IP repositories specified
    [IP_Flow 19-2313] Loaded Vivado IP repository 'D:/Vivado/2018.2/data/ip'.
Elaborated Design (20 warnings, 16 infos)
  General Messages (20 warnings, 16 infos)
    [Synth 8-2490] overwriting previous definition of module DSP48A1 [DSP.v:1]
    [Synth 8-6157] synthesizing module 'DSP48A1' [DSP.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 DSP48A1 has unconnected port CARRYIN (18 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:
```

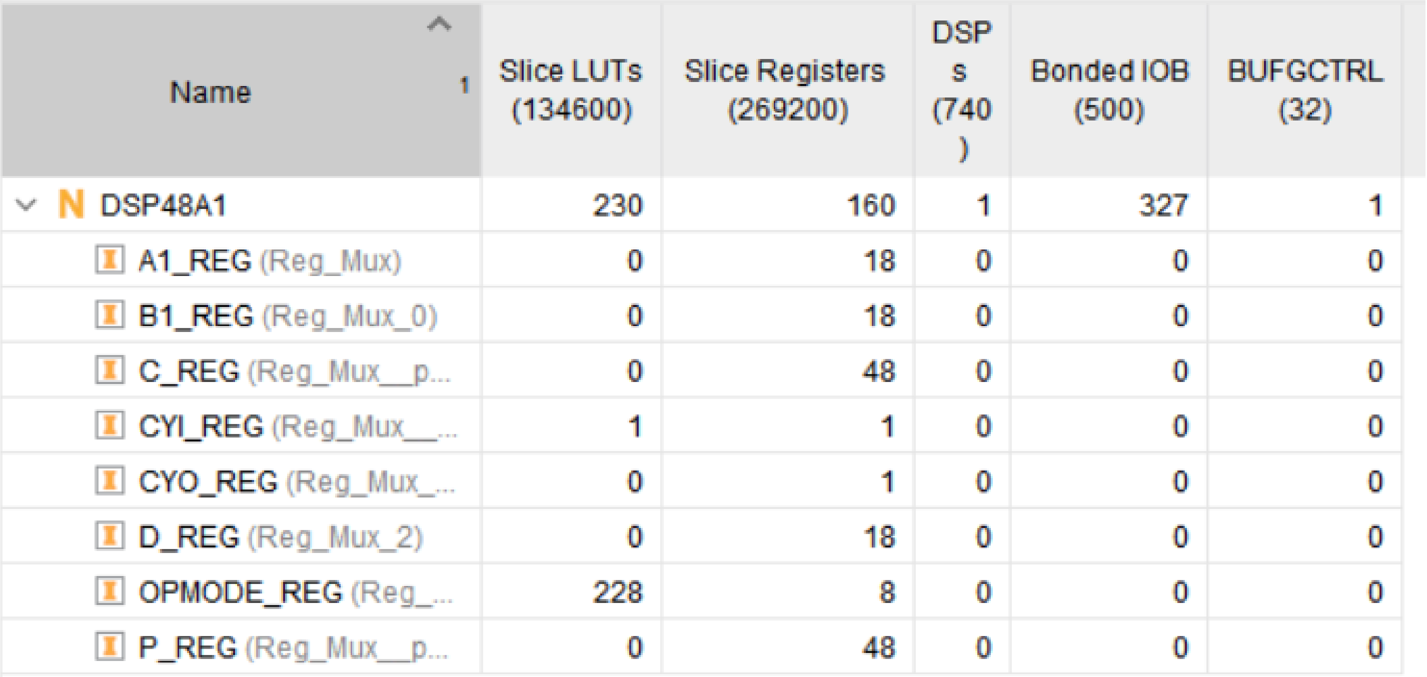

BCIN[17:0]
CARRYIN
CEP
RSTP
CECARRYIN
RSTCARRYIN
PCIN[47:0]
CEC
C[47:0]
RSTC
CLK
CEA
A[17:0]
RSTA
CED
D[17:0]
RSTD
CEM
RSTM
CEB
B[17:0]
RSTB
CEOPMODE
OPMODE[7:0]
RSTOPMODE
A0_REG
A1_REG
C_REG
mult_out_i
D_REG
OPMODE_REG
z_out1_i
x_out1_i
M_REG
z_out0_i
x_out0_i
z_out_i
CYI_REG
x_out_i
B0_REG
Adder_2_out2_i
Adder_2_out2_i__0
Adder1_out0_i
Adder1_out0_i__0
Adder_2_out1_i
Adder_2_out1_i__0
Adder_2_out0_i
Adder1_out_i
Adder1_mux_out_i
P_REG
CYO_REG
B1_REG
M_i
P[47:0]
PCOUT[47:0]
CARRYOUTF
CARRYOUT
BCOUT[17:0]
M[35:0]

- Synthesis Messages, Utilization report, timing report & Schematic snippets:

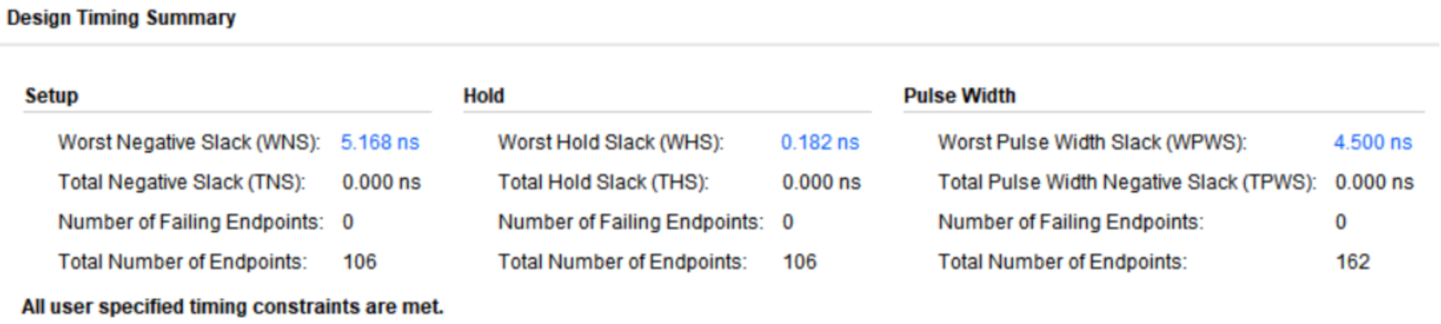


☑ Warning (58) ☑ Info (57) □ Status (19)

```
[Common 17-83] Releasing license: Synthesis
[Constraints 18-5210] No constraint will be written out.
[Common 17-1381] The checkpoint 'C:/Users/Mohamed/OneDrive/Desktop/Digital Design/projectt/project_1/project_1.runs/synth_1/DSP48A1.dcp' has been generated.
[runtcl-4] Executing : report_utilization -file DSP48A1_utilization_synth.rpt -pb DSP48A1_utilization_synth.pb
[Common 17-206] Exiting Vivado at Sat Mar 8 02:36:35 2025...
Synthesized Design (6 infos)
  General Messages (6 infos)
    [Netlist 29-17] Analyzing 207 Unisim elements for replacement
    [Netlist 29-28] Unisim Transformation completed in 1 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.
```



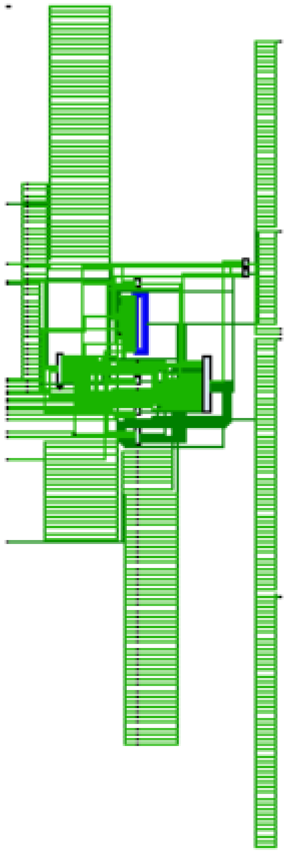
Name	Slice LUTs (134600)	Slice Registers (269200)	DSPs (740)	Bonded IOB (500)	BUFGCTRL (32)
DSP48A1	230	160	1	327	1
A1_REG (Reg_Mux)	0	18	0	0	0
B1_REG (Reg_Mux_0)	0	18	0	0	0
C_REG (Reg_Mux__p...	0	48	0	0	0
CYI_REG (Reg_Mux__...	1	1	0	0	0
CYO_REG (Reg_Mux__...	0	1	0	0	0
D_REG (Reg_Mux_2)	0	18	0	0	0
OPMODE_REG (Reg_...	228	8	0	0	0
P_REG (Reg_Mux__p...	0	48	0	0	0

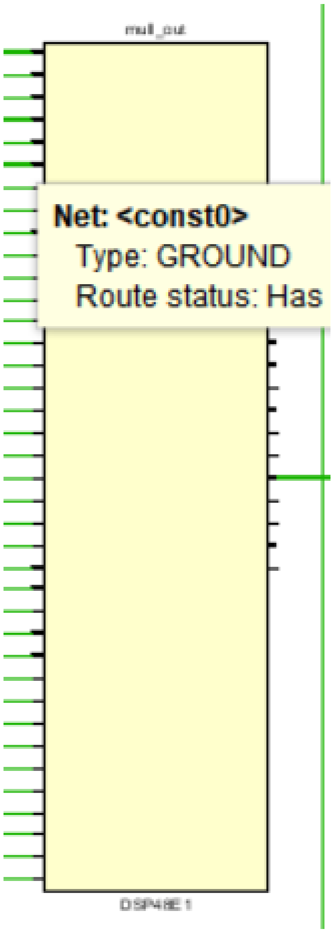


Design Timing Summary

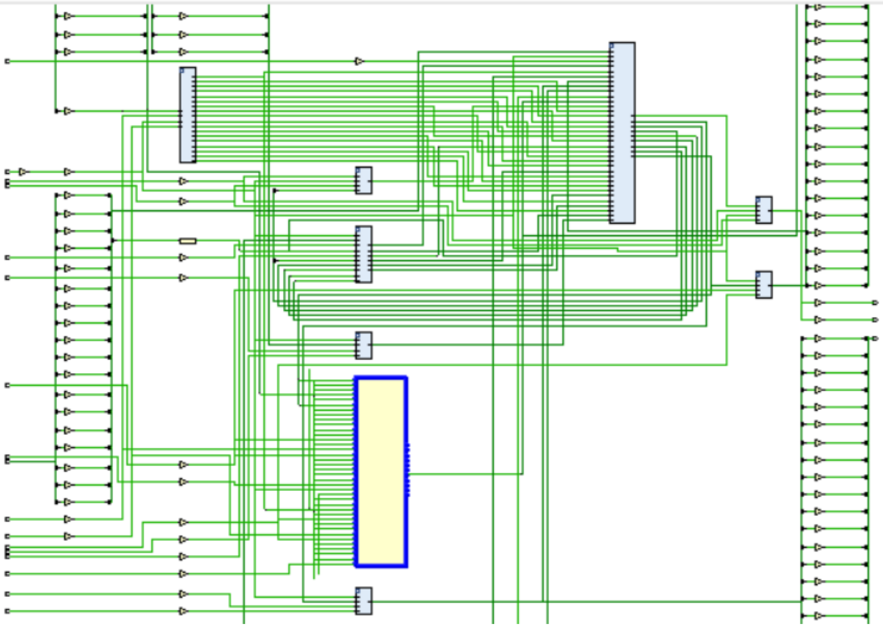
Setup		Hold		Pulse Width	
Worst Negative Slack (WNS):	5.168 ns	Worst Hold Slack (WHS):	0.182 ns	Worst Pulse Width Slack (WPWS):	4.500 ns
Total Negative Slack (TNS):	0.000 ns	Total Hold Slack (THS):	0.000 ns	Total Pulse Width Negative Slack (TPWS):	0.000 ns
Number of Failing Endpoints:	0	Number of Failing 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.

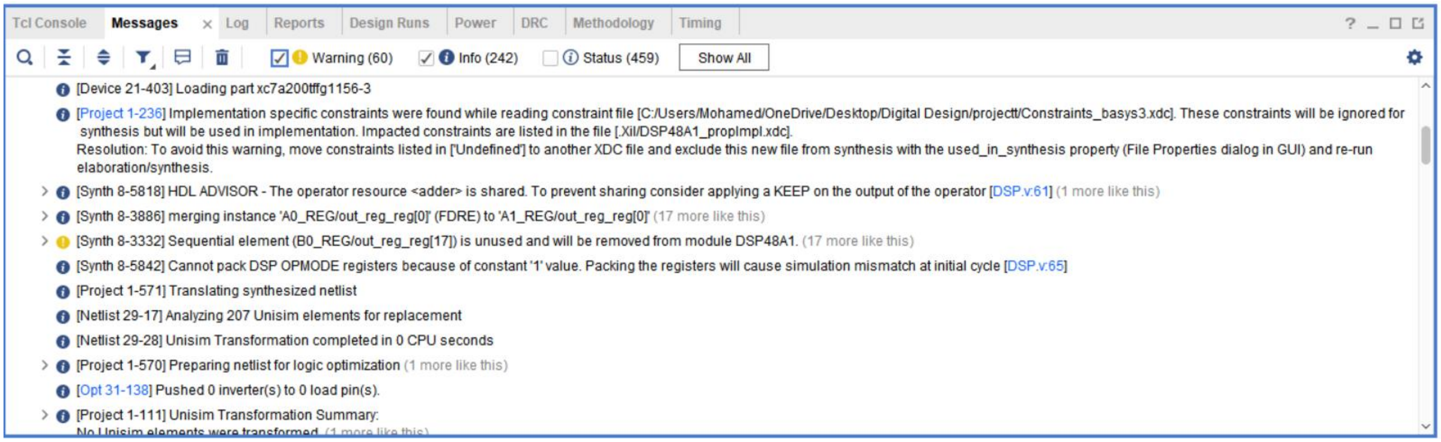




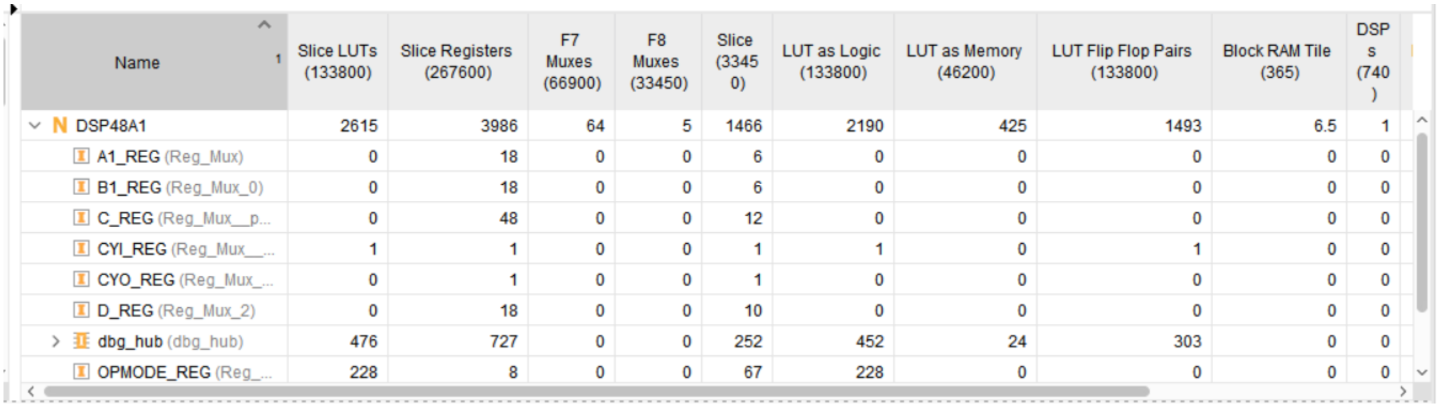
Net: <const0>
Type: GROUND
Route status: Has



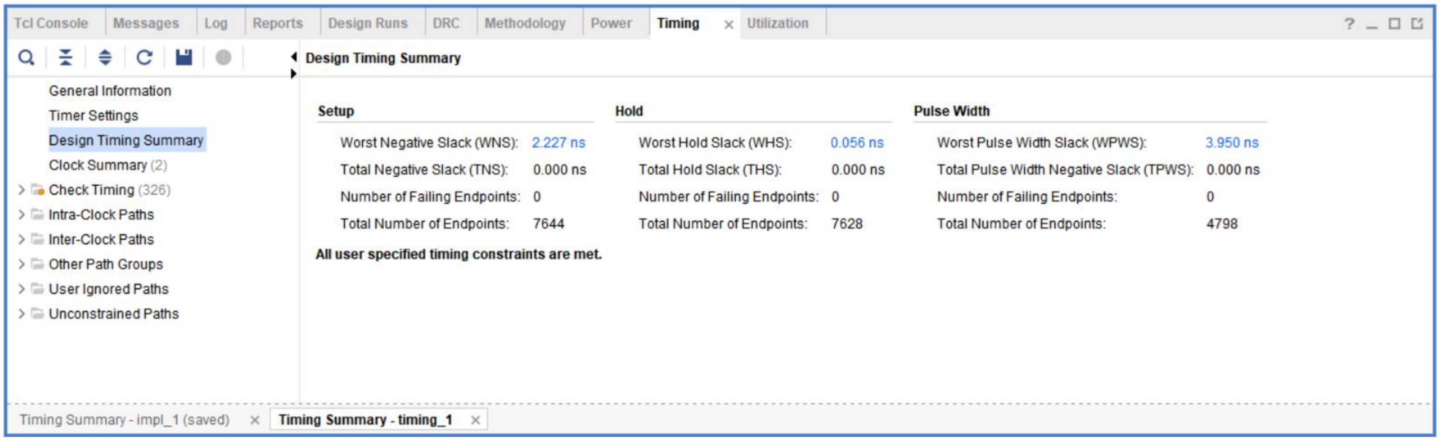
- Implementation Messages, Utilization report, timing report & device snippets



```
[Device 21-403] Loading part xc7a200tffg1156-3
[Project 1-236] Implementation specific constraints were found while reading constraint file [C:/Users/Mohamed/OneDrive/Desktop/Digital Design/projectt/Constraints_basys3.xdc]. These constraints will be ignored for
 synthesis but will be used in implementation. Impacted constraints are listed in the file [.Xil/DSP48A1_propImpl.xdc].
Resolution: To avoid this warning, move constraints listed in ['Undefined'] to another XDC file and exclude this new file from synthesis with the used_in_synthesis property (File Properties dialog in GUI) and re-run
elaboration/synthesis.
> [Synth 8-5818] HDL ADVISOR - The operator resource <adder> is shared. To prevent sharing consider applying a KEEP on the output of the operator [DSP.v:61] (1 more like this)
> [Synth 8-3886] merging instance 'A0_REG/out_reg_reg[0]' (FDRE) to 'A1_REG/out_reg_reg[0]' (17 more like this)
> [Synth 8-3332] Sequential element (B0_REG/out_reg_reg[17]) is unused and will be removed from module DSP48A1. (17 more like this)
[Synth 8-5842] Cannot pack DSP OPMODE registers because of constant '1' value. Packing the registers will cause simulation mismatch at initial cycle [DSP.v:65]
[Project 1-571] Translating synthesized netlist
[Netlist 29-17] Analyzing 207 Unisim elements for replacement
[Netlist 29-28] Unisim Transformation completed in 0 CPU seconds
> [Project 1-570] Preparing netlist for logic optimization (1 more like this)
[Opt 31-138] Pushed 0 inverter(s) to 0 load pin(s).
> [Project 1-111] Unisim Transformation Summary:
No Unisim elements were transformed. (1 more like this)
```



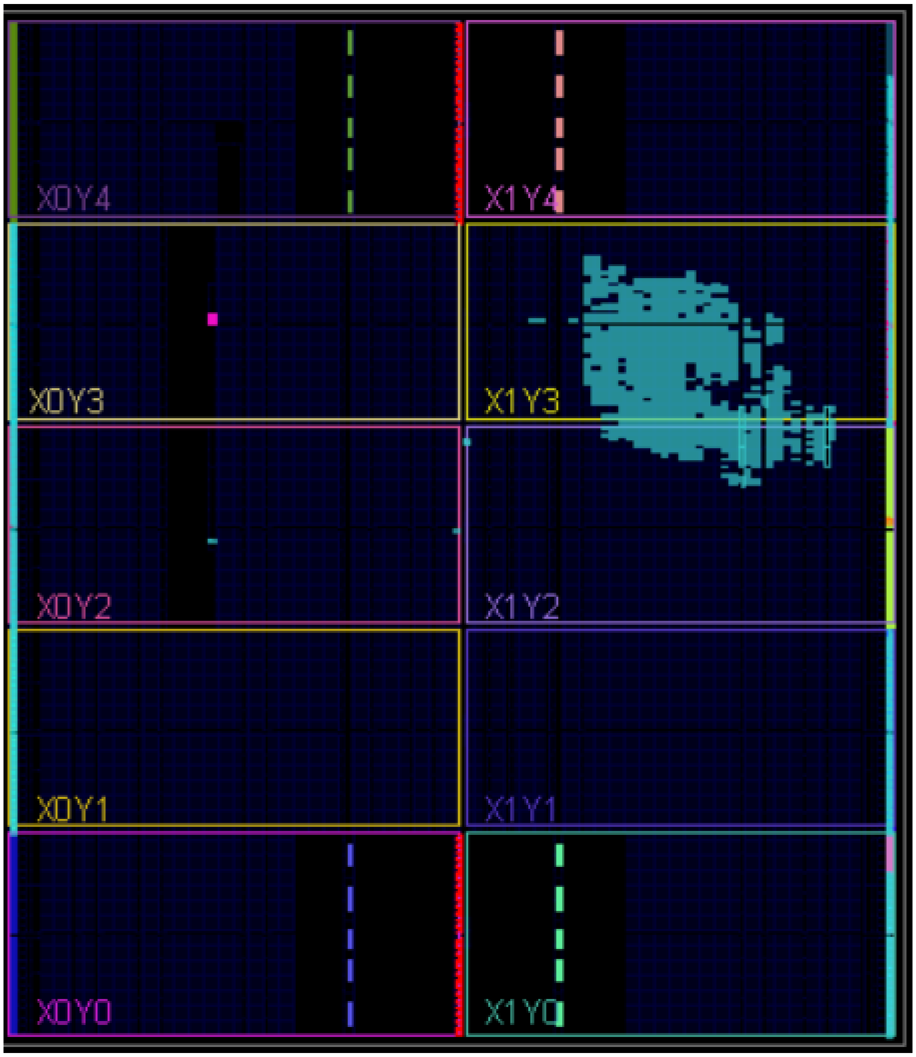
Name	Slice LUTs (133800)	Slice Registers (267600)	F7 Muxes (66900)	F8 Muxes (33450)	Slice (33450)	LUT as Logic (133800)	LUT as Memory (46200)	LUT Flip Flop Pairs (133800)	Block RAM Tile (365)	DSPs (740)
DSP48A1	2615	3986	64	5	1466	2190	425	1493	6.5	1
A1_REG (Reg_Mux)	0	18	0	0	6	0	0	0	0	0
B1_REG (Reg_Mux_0)	0	18	0	0	6	0	0	0	0	0
C_REG (Reg_Mux__p...	0	48	0	0	12	0	0	0	0	0
CYI_REG (Reg_Mux__...	1	1	0	0	1	1	0	1	0	0
CYO_REG (Reg_Mux_...	0	1	0	0	1	0	0	0	0	0
D_REG (Reg_Mux_2)	0	18	0	0	10	0	0	0	0	0
dbg_hub (dbg_hub)	476	727	0	0	252	452	24	303	0	0
OPMODE_REG (Reg_...	228	8	0	0	67	228	0	0	0	0



Design Timing Summary

Setup		Hold		Pulse Width	
Worst Negative Slack (WNS):	2.227 ns	Worst Hold Slack (WHS):	0.056 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:	7644	Total Number of Endpoints:	7628	Total Number of Endpoints:	4798

All user specified timing constraints are met.


X0Y4
X1Y4
X0Y3
X1Y3
X0Y2
X1Y2
X0Y1
X1Y1
X0Y0
X1Y0