

**Project1: Spartan6 - DSP48A1**

**Name: Mina Adel**

# RTL Code

## Register multiplexer block

```
module reg_mul(clk,rst,ce,in,out);
    parameter REG=1;//select
    parameter RSTtype="SYNC";//rst is syncronus
    parameter width=18;
    input clk,rst,ce;
    input [width-1:0] in;
    output reg [width-1:0] out;
    generate
        if(REG==1)begin
            if (RSTtype == "SYNC") begin
                always @(posedge clk) begin
                    if(rst)
                        out<=0;
                    else if(ce)begin
                        out<=in;
                    end
                end
            end
            else if (RSTtype == "ASYNC") begin
                always @(posedge clk or posedge rst) begin
                    if(rst)
                        out<=0;
                    else if(ce)begin
                        out<=in;
                    end
                end
            end
        end
        else begin
            always @(*) begin
                out=in;
            end
        end
    endgenerate
endmodule
```

# DSP

```
module DSP(a,b,c,d,cin,opmode,Pcin,Bcin,clk,CEa,CEb,CEc,CEcin,CEd,CEm,  
CEopmode,CEp,rsta,rstb,rstc,rstcin,rstd,rstM,rstopmode,rstP,M,P,cout,coutf,bcout,  
Pcout);  
  
parameter A0REG=0 , A1REG=1 ,B0REG=0 ,B1REG=1;  
parameter CREG=1, DREG=1, MREG=1, PREG=1, CARRYINREG=1, CARRYOUTREG=1,  
OPREG=1;  
parameter CARRYINSEL="OPMODE5";  
parameter B_INPUT="DIRECT";  
parameter RSTTYPE ="SYNC";  
parameter w1=1, w8=8, w18=18, w36=36, w48=48;  
  
input [w18-1:0] a, b,d;  
input [w1-1:0] cin;  
input [w48-1:0] c;  
input [w8-1:0] opmode;  
input [47:0] Pcin;  
input [17:0] Bcin;  
input clk,CEa,CEb,CEc,CEcin,CEd,CEm,CEopmode,CEp;  
input rsta,rstb,rstc,rstcin,rstd,rstM,rstopmode,rstP;  
  
output [35:0] M;  
output [47:0] P;  
output cout,coutf;  
output [17:0] bcout;  
output [47:0] Pcout;  
  
wire [w18-1:0] d_reg;  
wire [w18-1:0] b0_reg;  
wire [w18-1:0] bb;  
wire [w18-1:0] a0_reg;  
wire [w48-1:0] c_reg;  
wire [w8-1:0] opmode_reg;  
wire [w18-1:0] k0;  
wire [w18-1:0] add1_out;  
wire [w18-1:0] k0_reg;  
wire [w18-1:0] a1_reg;  
wire [w36-1:0] m;  
wire [w36-1:0] m_reg;  
wire k1;
```

```

wire Cin;
wire [47:0] conc;
wire [47:0] x_out,z_out;
wire [47:0] add2_out;
wire cout0;

assign bb=(B_INPUT=="DIRECT")?b:(B_INPUT=="CASCADE")?Bcin:18'b0;

reg_mul #(.REG(DREG),.RSTtype(RSTTYPE),.width(w18))
D_REG(clk,rstd,CEd,d,d_reg);
reg_mul #(.REG(B0REG),.RSTtype(RSTTYPE),.width(w18))
B0_REG(clk,rstb,CEb,bb,b0_reg);
reg_mul #(.REG(A0REG),.RSTtype(RSTTYPE),.width(w18))
A0_REG(clk,rsta,CEa,a,a0_reg);
reg_mul #(.REG(CREG),.RSTtype(RSTTYPE),.width(w48))
C_REG(clk,rstc,CEc,c,c_reg);
reg_mul #(.REG(OPREG),.RSTtype(RSTTYPE),.width(w8))
opmode_REG(clk,rstopmode,CEopmode,opmode,opmode_reg);

assign add1_out=(opmode_reg[6]==0)?d_reg+b0_reg:d_reg-b0_reg;
assign k0=(opmode_reg[4]==0)?b0_reg:add1_out;

reg_mul #(.REG(B1REG),.RSTtype(RSTTYPE),.width(w18))
B1_REG(clk,rstb,CEb,k0,k0_reg);
reg_mul #(.REG(A1REG),.RSTtype(RSTTYPE),.width(w18))
A1_REG(clk,rsta,CEa,a0_reg,a1_reg);

assign bcout=k0_reg;
assign m=k0_reg*a1_reg;

reg_mul #(.REG(MREG),.RSTtype(RSTTYPE),.width(w36))
M_REG(clk,rstM,CEm,m,m_reg);

assign M=m_reg;
assign k1=(CARRYINSEL=="OPMODE5")?opmode_reg[5]:cin;

reg_mul #(.REG(CARRYINREG),.RSTtype(RSTTYPE),.width(w1))
CYI(clk,rstcin,CEcin,k1,Cin);

assign conc={d[11:0],a1_reg[17:0],k0_reg[17:0]};

```

```

    assign x_out=(opmode_reg[1:0]==2'b00)?48'b0:(opmode_reg[1:0]==2'b01)?{{(w48 -
w36){1'b0}},m_reg}:
        (opmode_reg[1:0]==2'b10)?Pcout:conc;

    assign
z_out=(opmode_reg[3:2]==2'b00)?48'b0:(opmode_reg[3:2]==2'b01)?Pcin:(opmode_reg[3:
2]==2'b10)?Pcout:c_reg;
    assign {cout0,add2_out}=(opmode_reg[7]==0)?z_out+x_out+Cin : z_out-
(x_out+Cin);

    reg_mul #( .REG(CARRYOUTREG),.RSTtype(RSTTYPE),.width(w1))
CY0(clk,rstcin,CEcin,cout0,cout);

    assign coutf=cout;

    reg_mul #( .REG(PREG),.RSTtype(RSTTYPE),.width(w48))
P_REG(clk,rstP,CEp,add2_out,P);

    assign Pcout = P;

endmodule

```

## DSP Testbench

```
module DSP_tb();

parameter A0REG_tb=0 , A1REG_tb=1 ,B0REG_tb=0 ,B1REG_tb=1;
parameter CREG_tb=1, DREG_tb=1, MREG_tb=1, PREG_tb=1, CARRYINREG_tb=1,
CARRYOUTREG_tb=1, OPREG_tb=1;
parameter CARRYINSEL_tb="OPMODE5";
parameter B_INPUT_tb="DIRECT";
parameter RSTTYPE_tb ="SYNC";
parameter w1_tb=1, w8_tb=8, w18_tb=18, w36_tb=36, w48_tb=48;

reg [w18_tb-1:0] a_tb, b_tb,d_tb;
reg [w1_tb-1:0] cin_tb;
reg [w48_tb-1:0] c_tb;
reg [w8_tb-1:0] opmode_tb;
reg [47:0] Pcin_tb;
reg [17:0] Bcin_tb;
reg clk_tb,CEa_tb,CEb_tb,CEc_tb,CEcin_tb,CEd_tb,CEm_tb,CEopmode_tb,CEp_tb;
reg rsta_tb,rstb_tb,rstc_tb,rstcin_tb,rstd_tb,rstM_tb,rstopmode_tb,rstP_tb;

wire [35:0] M_dut;
wire [47:0] P_dut;
wire cout_dut,coutf_dut;
wire [17:0] bcout_dut;
wire [47:0] Pcout_dut;

DSP #( .A0REG(A0REG_tb), .A1REG(A1REG_tb), .B0REG(B0REG_tb), .B1REG(B1REG_tb),
.CREG(CREG_tb), .DREG(DREG_tb), .MREG(MREG_tb), .PREG(PREG_tb),
.CARRYINREG(CARRYINREG_tb), .CARRYOUTREG(CARRYOUTREG_tb), .OPREG(OPREG_
tb),
.CARRYINSEL(CARRYINSEL_tb), .B_INPUT(B_INPUT_tb), .RSTTYPE(RSTTYPE_tb),
.w1(w1_tb), .w8(w8_tb), .w18(w18_tb), .w36(w36_tb), .w48(w48_tb))

D1(a_tb, b_tb, c_tb, d_tb, cin_tb, opmode_tb, Pcin_tb, Bcin_tb, clk_tb,
CEa_tb, CEb_tb, CEc_tb, CEcin_tb, CEd_tb, CEm_tb, CEopmode_tb, CEp_tb,
rsta_tb, rstb_tb, rstc_tb, rstcin_tb, rstd_tb, rstM_tb, rstopmode_tb,
rstP_tb, M_dut, P_dut, cout_dut, coutf_dut, bcout_dut, Pcout_dut);
```

```

initial begin
    clk_tb=0;
    forever begin
        #1 clk_tb=~clk_tb;
    end
end

integer i;
initial begin
    rsta_tb=1; rstb_tb=1; rstc_tb=1; rstdin_tb=1; rstd_tb=1; rstm_tb=1;
rststopmode_tb=1; rstP_tb=1;
    a_tb=$random; b_tb=$random; c_tb=$random; d_tb=$random;
    cin_tb=$random;
    opmode_tb=$random;
    Pcin_tb=$random;
    Bcin_tb=$random;
    CEa_tb=$random; CEb_tb=$random; CEc_tb=$random; CEcin_tb=$random;
CED_tb=$random;
    CEm_tb=$random; CEopmode_tb=$random; CEp_tb=$random;

    @(negedge clk_tb);
    if (M_dut!=0 || P_dut!=0 || cout_dut!=0 || coutf_dut!=0 || bcout_dut!=0
|| Pcout_dut!=0 ) begin
        $display("error");
        $stop;
    end
    else begin
        $display("BCOUT=%h, M=%h, P=%h, PCOUT=%h, Carryout=%h,
Carryoutf=%h",bcout_dut, M_dut, P_dut, Pcout_dut, cout_dut, coutf_dut);
        $display("reset function is good");
    end

    rsta_tb=0; rstb_tb=0; rstc_tb=0; rstdin_tb=0; rstd_tb=0; rstm_tb=0;
rststopmode_tb=0; rstP_tb=0;
    CEa_tb=1; CEb_tb=1; CEc_tb=1; CEcin_tb=1; CEd_tb=1; CEm_tb=1;
CEopmode_tb=1; CEp_tb=1;

    opmode_tb= 8'b11011101;
    a_tb = 20; b_tb = 10; c_tb= 350; d_tb = 25;
    cin_tb=$random; Pcin_tb=$random; Bcin_tb=$random;

repeat(4) @(negedge clk_tb);

```

```

        if (bcout_dut !='hf || M_dut != 'h12c || P_dut !='h32 || Pcout_dut != 'h32 || cout_dut != 0 || coutf_dut!= 0) begin
            $display("error");
            $stop;
        end
        else begin
            $display("BCOUT=%h, M=%h, P=%h, PCOUT=%h, Carryout=%h,
Carryoutf=%h", bcout_dut,M_dut, P_dut, Pcout_dut, cout_dut, coutf_dut);
            $display("no errors in path 1");
        end

        opmode_tb= 8'b00010000;
        a_tb = 20; b_tb = 10; c_tb= 350; d_tb = 25;
        cin_tb=$random; Pcin_tb=$random; Bcin_tb=$random;
        repeat(3) @(posedge clk_tb);
        if (bcout_dut !='h23 || M_dut != 'h2bc || P_dut !='h0 || Pcout_dut != 'h0 || cout_dut != 0 || coutf_dut!= 0) begin
            $display("error");
            $stop;
        end
        else begin
            $display("BCOUT=%h, M=%h, P=%h, PCOUT=%h, Carryout=%h,
Carryoutf=%h", bcout_dut, M_dut, P_dut, Pcout_dut, cout_dut, coutf_dut);
            $display("no errors in path 2");
        end

        opmode_tb= 8'b00001010;
        a_tb = 20; b_tb = 10; c_tb= 350; d_tb = 25;
        cin_tb=$random; Pcin_tb=$random; Bcin_tb=$random;
        repeat(3) @(posedge clk_tb);
        if (bcout_dut !='ha || M_dut != 'hc8 || P_dut !='h0 || Pcout_dut != 'h0 || cout_dut != 0 || coutf_dut!= 0) begin
            $display("error");
            $stop;
        end
        else begin
            $display("BCOUT=%h, M=%h, P=%h, PCOUT=%h, Carryout=%h,
Carryoutf=%h", bcout_dut, M_dut, P_dut, Pcout_dut, cout_dut, coutf_dut);
            $display("no errors in path 3");
        end

        opmode_tb= 8'b10100111;
        a_tb = 5; b_tb = 6; c_tb= 350; d_tb = 25; Pcin_tb=3000;

```

```

    cin_tb=$random; Bcin_tb=$random;
    repeat(3) @(posedge clk_tb);
        if (bcout_dut !='h6 || M_dut != 'h1e || P_dut !='hfe6ffffec0bb1 ||
Pcout_dut !='hfe6ffffec0bb1 || cout_dut != 1 || coutf_dut!= 1) begin
            $display("BCOUT=%h, M=%h, P=%h, PCOUT=%h, Carryout=%h,
Carryoutf=%h",bcout_dut, M_dut, P_dut, Pcout_dut, cout_dut, coutf_dut);
            $display("error");
            $stop;
        end
        else begin
            $display("BCOUT=%h, M=%h, P=%h, PCOUT=%h, Carryout=%h,
Carryoutf=%h",bcout_dut, M_dut, P_dut, Pcout_dut, cout_dut, coutf_dut);
            $display("no errors in path 4");
        end

        $display("all is good isa");
        $stop;
    end
endmodule

```

## Do file

(automating simulation)

```
vlib work
```

```
vlog reg_mul_block.v DSP.v DSP_tb.v
```

```
vsim -voptargs=+acc work.DSP_tb
```

```
add wave *
```

```
run -all
```

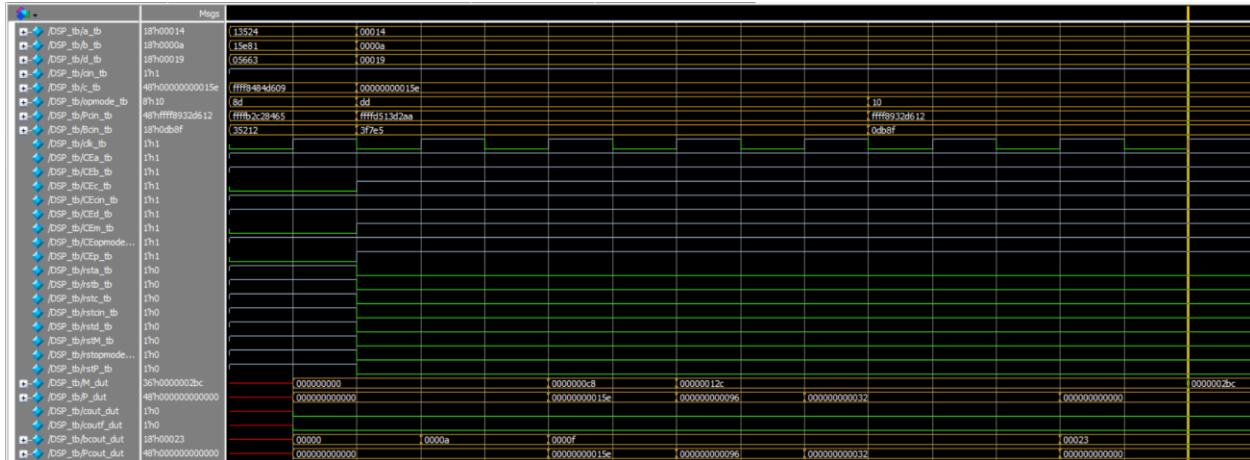
```
#quit -sim
```

# QuestaSim Snippets

## Transcript

```
# BCOUT=00000, M=000000000, P=0000000000000, PCOUT=0000000000000, Carryout=0, Carryoutf=0
# reset function is good
# BCOUT=0000f, M=00000012c, P=000000000032, PCOUT=000000000032, Carryout=0, Carryoutf=0
# no errors in path 1
# BCOUT=00023, M=0000002bc, P=0000000000000, PCOUT=0000000000000, Carryout=0, Carryoutf=0
# no errors in path 2
# BCOUT=0000a, M=0000000c8, P=0000000000000, PCOUT=0000000000000, Carryout=0, Carryoutf=0
# no errors in path 3
# BCOUT=00006, M=00000001e, P=fe6ffffec0bb1, PCOUT=fe6ffffec0bb1, Carryout=1, Carryoutf=1
# no errors in path 4
# all is good isa
```

## Waveform

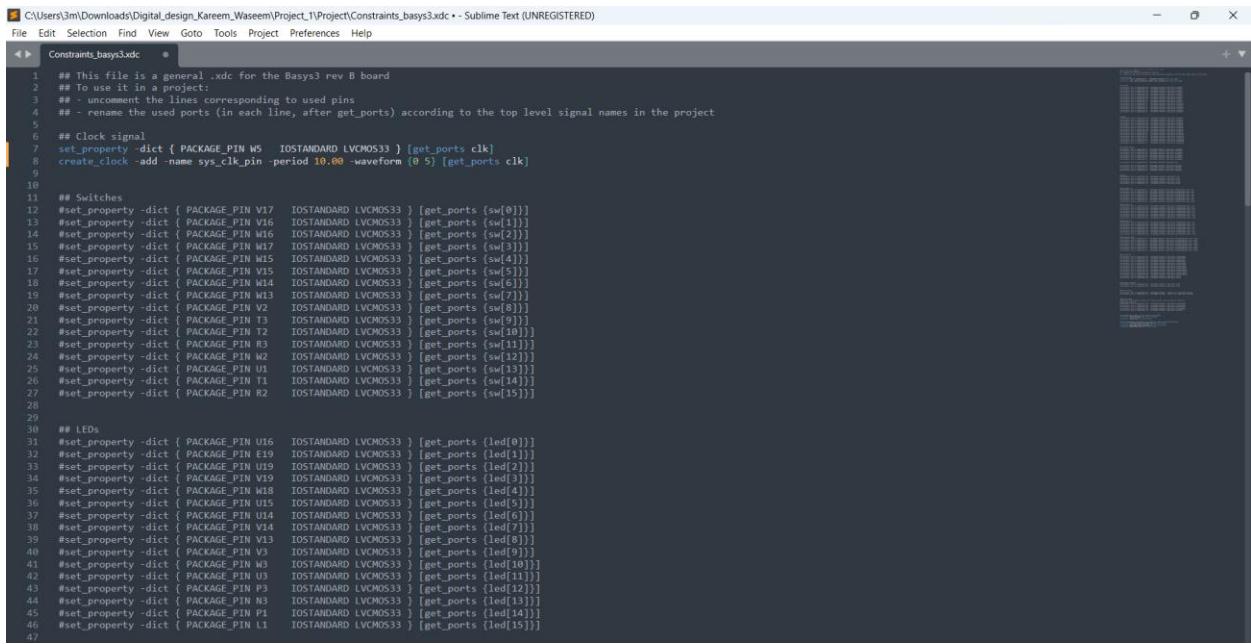


# Linting

The screenshot shows the Questa Lint 2021.1 interface. The main window displays the Verilog code for a module named DSP. The code includes parameters for A0REG, A1REG, B0REG, B1REG, CREG, DREG, MREG, PREG, and CARRYINSEL, along with various input port declarations. To the right of the code editor is the 'Lint Summary' pane, which lists four findings: two 'Open(uninspected, pending)' items and two 'Info' items. Below the code editor is the 'Lint Checks' table, which details the four specific violations found in the code.

Severity	Status	Check	Alias	Message	Module	Category	State	Owner	STARC Reference
Info	Pending	condition_const		Condition expression is a constant. Module DSP. File ... DSP	Rtl Design Style open	unassigned			
Info	Pending	condition_const		Condition expression is a constant. Module DSP. File ... DSP	Rtl Design Style open	unassigned			
Info	Pending	multi_ports_in_single_line		Multiple ports are declared in one line. Module DSP. F... DSP	Rtl Design Style open	unassigned		3.5.6.3	
Info	Pending	multi_ports_in_single_line		Multiple ports are declared in one line. Module reg_m... reg_mul	Rtl Design Style open	unassigned		3.5.6.3	

# Constraint File



The screenshot shows a Sublime Text window displaying a constraint file named "Constraints\_basys3.xdc". The file contains XDC (Xilinx Design Constraints) code used to map pins to specific logic components. The code includes comments explaining the purpose of the file and how to use it. It defines clock signals, switches, LEDs, and various I/O pins with their corresponding properties like IO STANDARD and LVC MOS33.

```
1  ## This file is a general .xdc for the Basys3 rev B board
2  ## To use it in a project:
3  ## - uncomment the lines corresponding to used pins
4  ## - rename the used ports (in each line, after get_ports) according to the top level signal names in the project
5
6  ## Clock signal
7  set_property dict { PACKAGE_PIN W5 :IOSTANDARD LVC MOS33 } [get_ports clk]
8  create_clock -add -name sys_clk_pin -period 10.00 -waveform {0 5} [get_ports clk]
9
10
11 ## Switches
12 #set_property -dict { PACKAGE_PIN V17 :IOSTANDARD LVC MOS33 } [get_ports {sw[0]}]
13 #set_property -dict { PACKAGE_PIN V16 :IOSTANDARD LVC MOS33 } [get_ports {sw[1]}]
14 #set_property -dict { PACKAGE_PIN W16 :IOSTANDARD LVC MOS33 } [get_ports {sw[2]}]
15 #set_property -dict { PACKAGE_PIN W17 :IOSTANDARD LVC MOS33 } [get_ports {sw[3]}]
16 #set_property -dict { PACKAGE_PIN W15 :IOSTANDARD LVC MOS33 } [get_ports {sw[4]}]
17 #set_property -dict { PACKAGE_PIN V15 :IOSTANDARD LVC MOS33 } [get_ports {sw[5]}]
18 #set_property -dict { PACKAGE_PIN W14 :IOSTANDARD LVC MOS33 } [get_ports {sw[6]}]
19 #set_property -dict { PACKAGE_PIN W13 :IOSTANDARD LVC MOS33 } [get_ports {sw[7]}]
20 #set_property -dict { PACKAGE_PIN V12 :IOSTANDARD LVC MOS33 } [get_ports {sw[8]}]
21 #set_property -dict { PACKAGE_PIN V10 :IOSTANDARD LVC MOS33 } [get_ports {sw[9]}]
22 #set_property -dict { PACKAGE_PIN T2 :IOSTANDARD LVC MOS33 } [get_ports {sw[10]}]
23 #set_property -dict { PACKAGE_PIN R3 :IOSTANDARD LVC MOS33 } [get_ports {sw[11]}]
24 #set_property -dict { PACKAGE_PIN N2 :IOSTANDARD LVC MOS33 } [get_ports {sw[12]}]
25 #set_property -dict { PACKAGE_PIN U3 :IOSTANDARD LVC MOS33 } [get_ports {sw[13]}]
26 #set_property -dict { PACKAGE_PIN T1 :IOSTANDARD LVC MOS33 } [get_ports {sw[14]}]
27 #set_property -dict { PACKAGE_PIN R2 :IOSTANDARD LVC MOS33 } [get_ports {sw[15]}]
28
29
30 ## LEDs
31 #set_property -dict { PACKAGE_PIN U16 :IOSTANDARD LVC MOS33 } [get_ports {led[0]}]
32 #set_property -dict { PACKAGE_PIN E19 :IOSTANDARD LVC MOS33 } [get_ports {led[1]}]
33 #set_property -dict { PACKAGE_PIN U19 :IOSTANDARD LVC MOS33 } [get_ports {led[2]}]
34 #set_property -dict { PACKAGE_PIN V19 :IOSTANDARD LVC MOS33 } [get_ports {led[3]}]
35 #set_property -dict { PACKAGE_PIN W19 :IOSTANDARD LVC MOS33 } [get_ports {led[4]}]
36 #set_property -dict { PACKAGE_PIN V15 :IOSTANDARD LVC MOS33 } [get_ports {led[5]}]
37 #set_property -dict { PACKAGE_PIN U14 :IOSTANDARD LVC MOS33 } [get_ports {led[6]}]
38 #set_property -dict { PACKAGE_PIN V14 :IOSTANDARD LVC MOS33 } [get_ports {led[7]}]
39 #set_property -dict { PACKAGE_PIN V13 :IOSTANDARD LVC MOS33 } [get_ports {led[8]}]
40 #set_property -dict { PACKAGE_PIN V12 :IOSTANDARD LVC MOS33 } [get_ports {led[9]}]
41 #set_property -dict { PACKAGE_PIN W3 :IOSTANDARD LVC MOS33 } [get_ports {led[10]}]
42 #set_property -dict { PACKAGE_PIN T3 :IOSTANDARD LVC MOS33 } [get_ports {led[11]}]
43 #set_property -dict { PACKAGE_PIN R3 :IOSTANDARD LVC MOS33 } [get_ports {led[12]}]
44 #set_property -dict { PACKAGE_PIN N3 :IOSTANDARD LVC MOS33 } [get_ports {led[13]}]
45 #set_property -dict { PACKAGE_PIN P3 :IOSTANDARD LVC MOS33 } [get_ports {led[14]}]
46 #set_property -dict { PACKAGE_PIN L3 :IOSTANDARD LVC MOS33 } [get_ports {led[15]}]
```

# Vivado

## Elaboration

### Messages

to check that there is no errors or critical

The screenshot shows the Vivado interface with the 'Messages' tab selected. The title bar reads 'ELABORATED DESIGN - xc7a200tffg1156-3'. The 'Messages' tab has a checked warning icon. Below the tabs are search and filter icons. The main pane displays a tree view of messages under 'Elaborated Design' and 'General Messages'. The 'General Messages' node is expanded, showing numerous warnings related to port connections in module 'DSP'.

- Elaborated Design (22 warnings, 17 infos)
  - General Messages (22 warnings, 17 infos)
    - [Synth 8-7079] Multithreading enabled for synth\_design using a maximum of 2 processes.
    - [Synth 8-6157] synthesizing module 'DSP' [DSP.v:1] (6 more like this)
    - [Synth 8-6155] done synthesizing module 'reg\_mul' (0#1) [reg\_mul\_block.v:1] (6 more like this)
    - [Synth 8-7129] Port clk in module reg\_mul\_parameterized0 is either unconnected or has no load (21 more like this)
      - [Synth 8-7129] Port rst in module reg\_mul\_parameterized0 is either unconnected or has no load
      - [Synth 8-7129] Port ce in module reg\_mul\_parameterized0 is either unconnected or has no load
      - [Synth 8-7129] Port cin[0] in module DSP is either unconnected or has no load
      - [Synth 8-7129] Port Bcin[17] in module DSP is either unconnected or has no load
      - [Synth 8-7129] Port Bcin[16] in module DSP is either unconnected or has no load
      - [Synth 8-7129] Port Bcin[15] in module DSP is either unconnected or has no load
      - [Synth 8-7129] Port Bcin[14] in module DSP is either unconnected or has no load
      - [Synth 8-7129] Port Bcin[13] in module DSP is either unconnected or has no load
      - [Synth 8-7129] Port Bcin[12] in module DSP is either unconnected or has no load
      - [Synth 8-7129] Port Bcin[11] in module DSP is either unconnected or has no load
      - [Synth 8-7129] Port Bcin[10] in module DSP is either unconnected or has no load
      - [Synth 8-7129] Port Bcin[9] in module DSP is either unconnected or has no load
      - [Synth 8-7129] Port Bcin[8] in module DSP is either unconnected or has no load
      - [Synth 8-7129] Port Bcin[7] in module DSP is either unconnected or has no load
      - [Synth 8-7129] Port Bcin[6] in module DSP is either unconnected or has no load
      - [Synth 8-7129] Port Bcin[5] in module DSP is either unconnected or has no load
      - [Synth 8-7129] Port Bcin[4] in module DSP is either unconnected or has no load
      - [Synth 8-7129] Port Bcin[3] in module DSP is either unconnected or has no load
      - [Synth 8-7129] Port Bcin[2] in module DSP is either unconnected or has no load
      - [Synth 8-7129] Port Bcin[1] in module DSP is either unconnected or has no load
      - [Synth 8-7129] Port Bcin[0] in module DSP is either unconnected or has no load

The 22 errors because of the selector parameters when we select an input port another input port will be unused so there will be a warning for it

## Analyzing warnings

Some warnings is reg\_mul.v file as when REG ==0 so clk and rst and ce will not be used and others in DSP.v file as when carryinsel == opmode5 so cin will not be used and when B\_INPUT == DIRECT so b\_cin will not be used

```
// Dummy usage to avoid unconnected port warnings
// generate
//     if (CARRYINSEL == "OPMODE5") begin
//         (* DONT_TOUCH = "true" *) wire cin_unused = |cin;
//     end

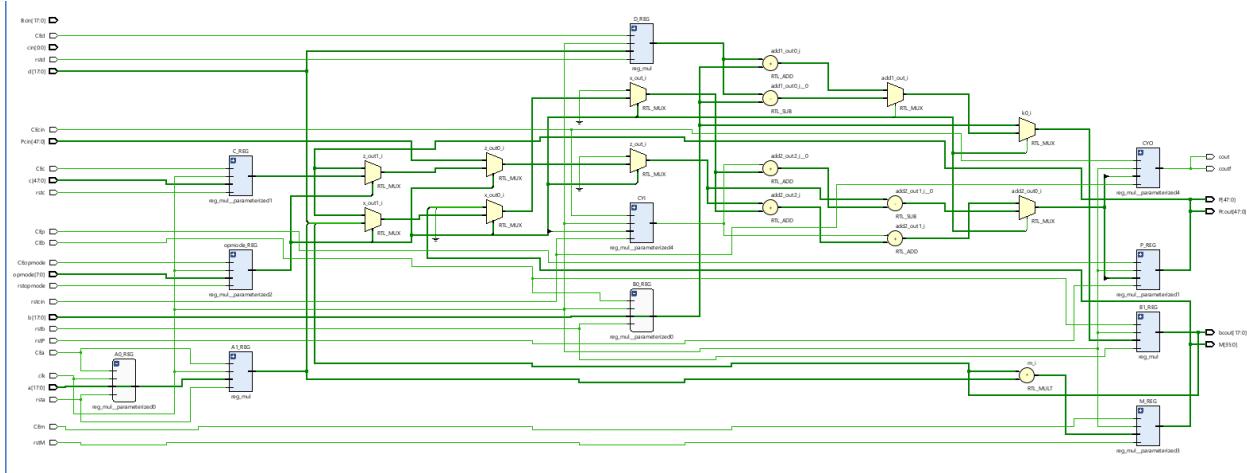
//     if (B_INPUT != "CASCADE") begin
//         (* DONT_TOUCH = "true" *) wire bcin_unused = |Bcin;
//     end
// endgenerate

else begin
    always @(*) begin
        out=in;
    end
    // Dummy usage to avoid unconnected port warnings

    // (* DONT_TOUCH = "true" *) reg clk_unused, rst_unused, ce_unused;
    // always @(posedge clk) begin
    //     clk_unused <= clk_unused;
    //     rst_unused <= rst;
    //     ce_unused<=ce;
    // end
end
```

these is a solutions but it will increase the hardware so it  
doesn't make sense to avoide some not important warnings  
increase the hardware

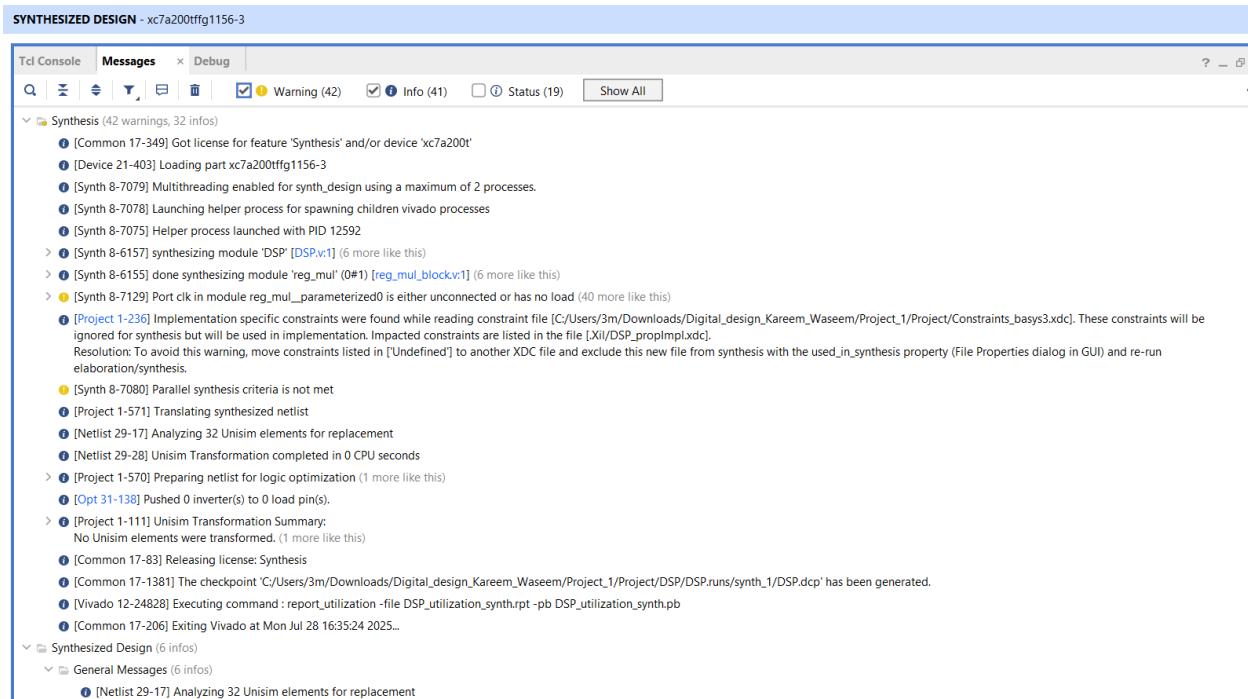
# Schematic



# Synthesis

## Messages

to check that there is no errors or critical



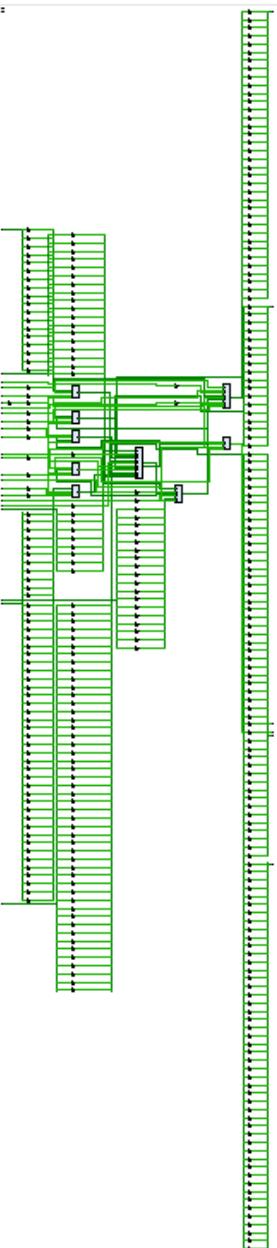
The screenshot shows the 'Messages' tab in the Vivado IDE interface. The title bar reads 'SYNTHESIZED DESIGN - xc7a200tffg1156-3'. The 'Messages' tab is selected, showing a list of messages. The message list is organized into sections: 'Synthesis' (42 warnings, 32 infos), 'Synthesized Design' (6 infos), and 'General Messages' (6 infos). The 'Synthesis' section contains numerous warnings related to synthesis, including license issues, device loading, multithreading, launching helper processes, synthesizing modules, and parallel synthesis criteria. One warning specifically mentions unconnected ports in a module named 'reg\_mul'. The 'Synthesized Design' and 'General Messages' sections contain fewer, mostly informational messages.

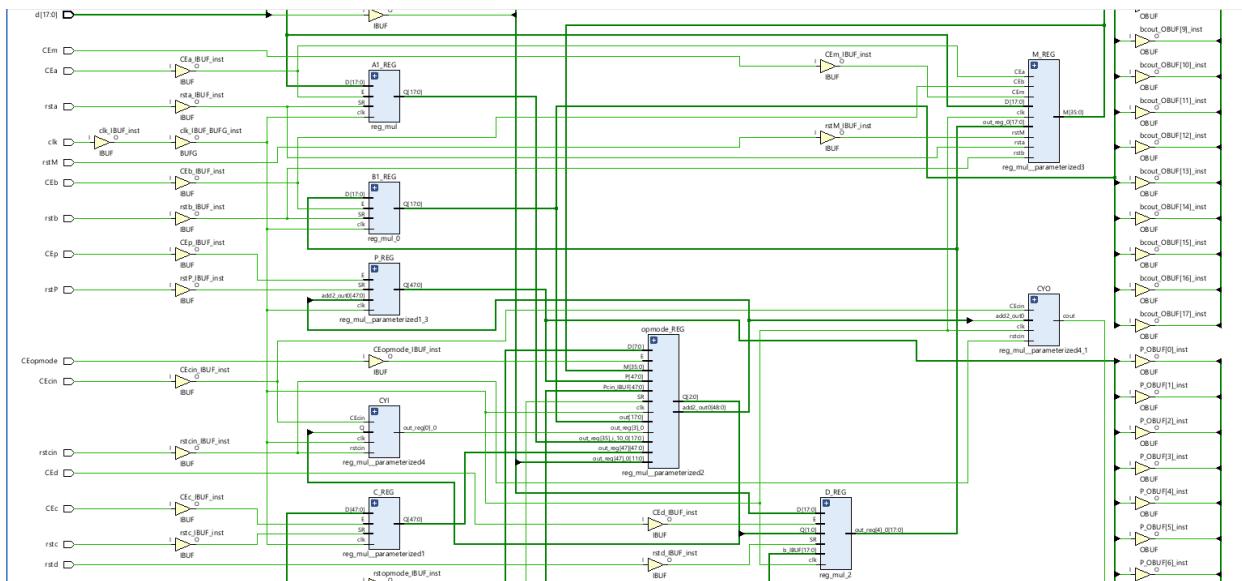
```
SYNTHESIZED DESIGN - xc7a200tffg1156-3
Tcl Console Messages Debug
? - Show All
Q W D E M B S I Status (19) Warning (42) Info (41)
Synthesis (42 warnings, 32 infos)
  [Common 17-349] Got license for feature 'Synthesis' and/or device 'xc7a200tffg1156-3'
  [Device 21-403] Loading part xc7a200tffg1156-3
  [Synth 8-7079] Multithreading enabled for synth_design using a maximum of 2 processes.
  [Synth 8-7078] Launching helper process for spawning children vivado processes
  [Synth 8-7075] Helper process launched with PID 12592
  > [Synth 8-6157] synthesizing module 'DSP' [DSP.v1] (6 more like this)
  > [Synth 8-6155] done synthesizing module 'reg_mul' (#1) [reg_mul_block.v1] (6 more like this)
  > [Synth 8-7129] Port clk in module reg_mul__parameterized0 is either unconnected or has no load (40 more like this)
  [Project 1-236] Implementation specific constraints were found while reading constraint file [C:/Users/3m/Downloads/Digital_design_Kareem_Waseem/Project_1/Project/Constraints_basys3.xdc]. These constraints will be ignored for synthesis but will be used in implementation. Impacted constraints are listed in the file [Xil/DSP_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 8-7080] Parallel synthesis criteria is not met
  [Project 1-571] Translating synthesized netlist
  [Netlist 29-17] Analyzing 32 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)
  [Common 17-83] Releasing license: Synthesis
  [Common 17-1381] The checkpoint 'C:/Users/3m/Downloads/Digital_design_Kareem_Waseem/Project_1/Project/DSP/DSP.runs/synth_1/DSP.dcp' has been generated.
  [Vivado 12-24828] Executing command : report_utilization -file DSP_utilization_synth.rpt -pb DSP_utilization_synth.pb
  [Common 17-206] Exiting Vivado at Mon Jul 28 16:35:24 2025...
Synthesized Design (6 infos)
  General Messages (6 infos)
    [Netlist 29-17] Analyzing 32 Unisim elements for replacement
```

Warning because the same thing about unconnected ports in some events so we can ignore them

## Schematic snippets

---





# Utilization report

The Utilization report interface shows a hierarchical tree on the left and a detailed utilization table on the right.

**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

**Utilization Table:**

Name	Slice LUTs (134600)	Slice Registers (269200)	DSPs (740)	Bonded IOB (500)	BUFGCTRL (32)
N DSP	218	160	1	327	1
A1_REG (reg_mul)	0	18	0	0	0
B1_REG (reg_mul_0)	0	18	0	0	0
C_REG (reg_mul_parameterized1)	0	48	0	0	0
CYI (reg_mul_parameterized4)	1	1	0	0	0
CYO (reg_mul_parameterized4_1)	0	1	0	0	0
D_REG (reg_mul_2)	36	18	0	0	0
M_REG (reg_mul_parameterized3)	0	0	1	0	0
opmode_REG (reg_mul_parameterized2)	181	8	0	0	0
P_REG (reg_mul_parameterized1_3)	0	48	0	0	0

# timing report

The timing report interface shows a hierarchical tree on the left and a detailed design timing summary table on the right.

**General Information:**

- Timer Settings
- Design Timing Summary
- Clock Summary (1)
- Methodology Summary
- Check Timing (326)
  - Intra-Clock Paths
  - Inter-Clock Paths
  - Other Path Groups
  - User Ignored Paths
  - Unconstrained Paths

**Design Timing Summary:**

Setup	Hold	Pulse Width
Worst Negative Slack (WNS): 5.512 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: 104	Total Number of Endpoints: 104	Total Number of Endpoints: 162

All user specified timing constraints are met.

# Implementation

## Messages

to check that there is no errors or critical

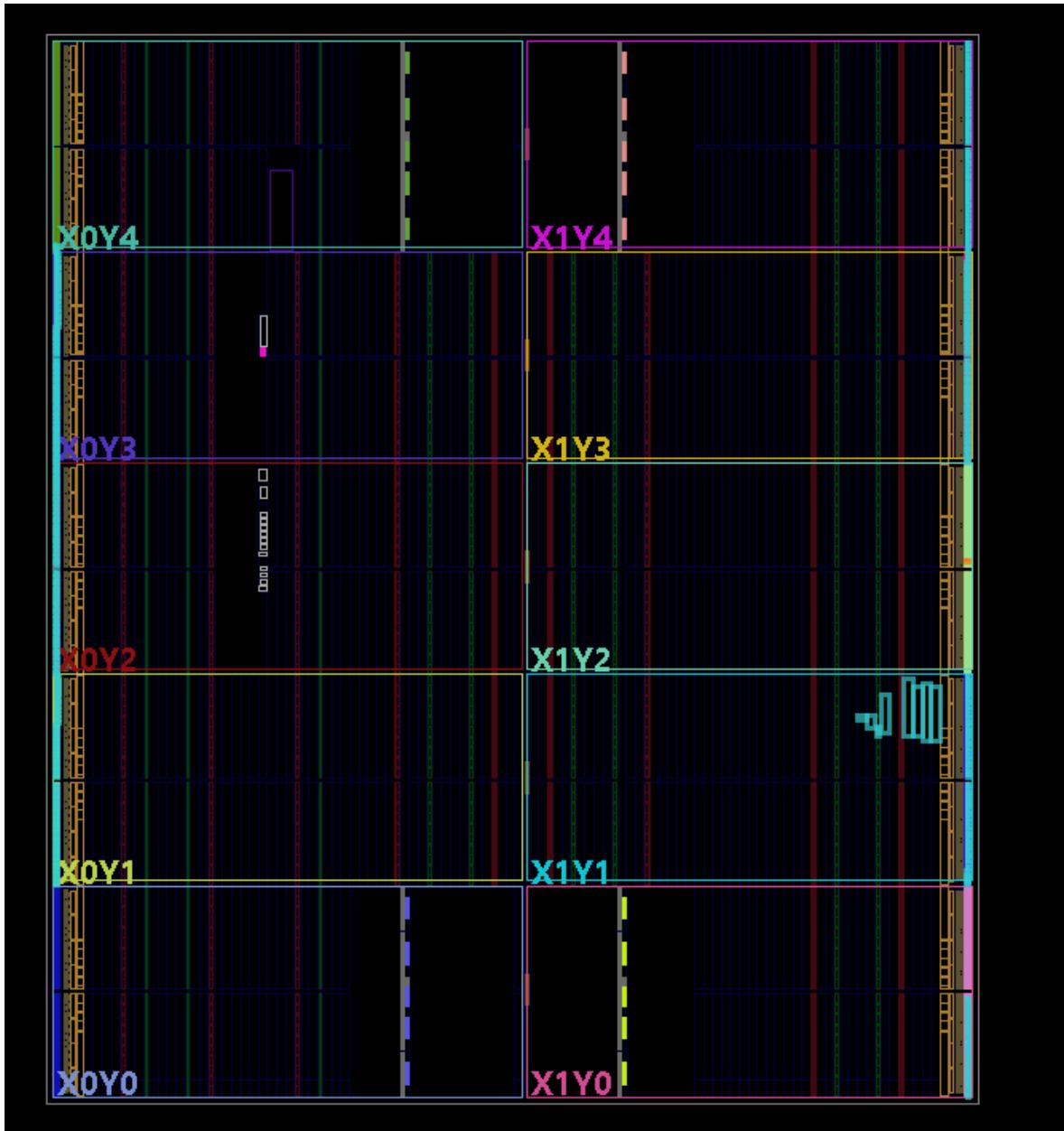
IMPLEMENTED DESIGN - xc7a200tffg1156-3

The screenshot shows the 'Messages' tab in the Vivado IDE. The title bar says 'IMPLEMENTED DESIGN - xc7a200tffg1156-3'. The tabs include Tcl Console, Messages (selected), Log, Reports, Design Runs, DRC, Methodology, Power, and Timing. Below the tabs are search and filter controls: a search bar, a date range selector, a message type dropdown, and checkboxes for Warning (42), Info (158), and Status (319). A 'Show All' button is also present. The main area displays a hierarchical log of messages. The 'Implementation' section contains 113 infos, including 'Design Initialization' (12 infos) and 'Opt Design' (27 infos). Other sections like 'Timing' and 'Power' have 0 messages.

- Implementation (113 infos)
  - Design Initialization (12 infos)
    - [Device 21-403] Loading part xc7a200tffg1156-3
    - [Netlist 29-17] Analyzing 32 Unisim elements for replacement
    - [Netlist 29-28] Unisim Transformation completed in 0 CPU seconds
    - [Project 1-479] Netlist was created with Vivado 2024.2.2
    - [Project 1-570] Preparing netlist for logic optimization
    - [Timing 38-478] Restoring timing data from binary archive.
    - [Timing 38-479] Binary timing data restore complete.
    - [Project 1-856] Restoring constraints from binary archive.
    - [Project 1-853] Binary constraint restore complete.
    - [Designutils 20-5722] Start Reading Physical Databases.
    - [Project 1-111] Unisim Transformation Summary:  
No Unisim elements were transformed.
    - [Project 1-604] Checkpoint was created with Vivado v2024.2.2 (64-bit) build 6060944
  - Opt Design (27 infos)
    - [Common 17-349] Got license for feature 'Implementation' and/or device 'xc7a200t'
    - [Project 1-461] DRC finished with 0 Errors
    - [Project 1-462] Please refer to the DRC report (report\_drc) for more information.
    - [Opt 31-1851] Number of loadless carry chains removed were: 0
    - [Opt 31-1834] Total Chains To Be Transformed Were: 0 AND Number of Transformed insts Created are: 0
    - [Opt 31-49] Retargeted 0 cell(s).
      - [Opt 31-138] Pushed 0 inverter(s) to 0 load pin(s). (1 more like this)
      - [Opt 31-389] Phase Retarget created 0 cells and removed 0 cells (4 more like this)
    - [Constraints 18-11670] Building netlist checker database with flags, 0x8
    - [Opt 31-662] Phase BUFG optimization created 0 cells of which 0 are BUFGs and removed 0 cells.
    - [Opt 31-1064] SRL Remap converted 0 SRLs to 0 registers and converted 0 registers of register chains to 0 SRLs

no warnings no errors

# Device



# Utilization report

Hierarchy								
Name	Slice LUTs (134600)	Slice Registers (269200)	Slice (33650)	LUT as Logic (134600)	DSPs (740)	Bonded IOB (500)	BUFGCTRL (32)	
N DSP	209	179	98	209	1	327	1	
A1_REG (reg_mul)	0	18	5	0	0	0	0	
B1_REG (reg_mul_0)	0	36	14	0	0	0	0	
C_REG (reg_mul_parameterized1)	0	48	20	0	0	0	0	
CY1 (reg_mul_parameterized4)	1	1	1	1	0	0	0	
CYO (reg_mul_parameterized4_1)	0	2	1	0	0	0	0	

# timing report

Design Timing Summary								
Setup			Hold			Pulse Width		
Worst Negative Slack (WNS):	4.688 ns		Worst Hold Slack (WHS):	0.234 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:	123		Total Number of Endpoints:	123		Total Number of Endpoints:	181	