DSP Project

By

Youssef Ahmed Ezzat Elsaadany

RTL Code

```
module dsp(
    A,B,C,D,CLK,CARRYIN,OPMODE,BCIN,RSTA,RSTB,
    RSTM,RSTP,RSTC,RSTD,RSTCARRYIN,RSTOPMODE,
    CEA,CEB,CEM,CEC,CED,CECARRYIN,CEOPMODE,CEP,PCIN,
    BCOUT,PCOUT,P,M,CARRYOUT,CARRYOUTF
);
    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";
    parameter ABD_WIDTH = 18;
    parameter CP_WIDTH = 48;
    parameter OPMODE_WIDTH = 8;
    parameter MUL_OUT_WIDTH = 36;
    input CLK;
    input RSTA,RSTB,RSTM,RSTP,RSTC,RSTD,RSTCARRYIN,RSTOPMODE;
    input CEA,CEB,CEM,CEC,CED,CECARRYIN,CEOPMODE,CEP;
    input [ABD_WIDTH-1 : 0] A,B,D,BCIN;
    input [CP_WIDTH-1 : 0] C,PCIN;
    input CARRYIN;
    input [OPMODE_WIDTH-1 : 0]OPMODE;
    output [CP_WIDTH-1 : 0]P,PCOUT;
    output [ABD_WIDTH-1 : 0]BCOUT;
    output [MUL_OUT_WIDTH-1 : 0] M;
    output CARRYOUT,CARRYOUTF;
    wire [7:0]OPMODE_mux;
    wire [ABD_WIDTH-1 : 0] A_mux,B_mux,B_mux_in,D_mux,pre_adder_out,pre_mux_out;
    wire [ABD_WIDTH-1 : 0] mul_in1,mul_in2;
    wire [MUL_OUT_WIDTH-1 : 0] mul_out,M_mux;
    wire [CP_WIDTH-1 : 0] C_mux,P_post_adder_out;
    reg [CP_WIDTH-1 : 0] X_out,Z_out;
    wire CIN,carr_cascade,carry_cascade_mux,cout;
    //Opmodesssssssss
    mux_ff #(.INPUT_WIDTH(8), .OUTPUT_WIDTH(8), .RSTTYPE(RSTTYPE)) op10 (OPMODE[7:0],OPMODE_mux,OPMODEREG,CLK,RSTOPMODE,CEOPMODE);
    //inistentiation of the mux's + ff
```

```
    //inistentiation of the mux's + ff
    mux_ff #(.RSTTYPE(RSTTYPE)) ain (A,A_mux,A0REG,CLK,RSTA,CEA); // A
    mux_ff #(.RSTTYPE(RSTTYPE)) bin (B_mux_in,B_mux,B0REG,CLK,RSTB,CEB); // B
    mux_ff #(.RSTTYPE(RSTTYPE)) din (D,D_mux,DREG,CLK,RSTD,CED); // D
    mux_ff #(.INPUT_WIDTH(CP_WIDTH), .OUTPUT_WIDTH(CP_WIDTH), .RSTTYPE(RSTTYPE)) cin (C,C_mux,CREG,CLK,RSTC,CEC); // C
    mux_ff #(.RSTTYPE(RSTTYPE)) mul1 (pre_mux_out,mul_in1,B1REG,CLK,RSTB,CEB);   // mul in1
    mux_ff #(.RSTTYPE(RSTTYPE)) mul2 (A_mux,mul_in2,A1REG,CLK,RSTA,CEA);        // mul in2
    mux_ff #(.INPUT_WIDTH(MUL_OUT_WIDTH),.OUTPUT_WIDTH(MUL_OUT_WIDTH), .RSTTYPE(RSTTYPE)) mul_muxff(
        mul_out,M_mux,MREG,CLK,RSTM,CEM
    );
    mux_ff #(.INPUT_WIDTH(1), .OUTPUT_WIDTH(1), .RSTTYPE(RSTTYPE)) carryin (carr_cascade,CIN,CARRYINREG,CLK,RSTCARRYIN,CECARRYIN); // carry in
    mux_ff #(.INPUT_WIDTH(1), .OUTPUT_WIDTH(1), .RSTTYPE(RSTTYPE)) CarryOut (cout,CARRYOUT,CARRYOUTREG,CLK,RSTCARRYIN,CECARRYIN); // carry out
    mux_ff #(.INPUT_WIDTH(CP_WIDTH),.OUTPUT_WIDTH(CP_WIDTH), .RSTTYPE(RSTTYPE)) p_mux(
        P_post_adder_out,P,PREG,CLK,RSTP,CEP
    );
    assign B_mux_in = (B_INPUT == "DIRECT")?B :(B_INPUT == "CASCADE")? BCIN: 0;
    assign pre_mux_out = (OPMODE_mux[4])? pre_adder_out:B_mux;
    assign BCOUT = mul_in1;
    assign mul_out = mul_in1 * mul_in2;
    assign M = M_mux;
    assign carr_cascade = (CARRYINSEL == "OPMODE5")? OPMODE_mux[5]: CARRYIN;
    assign pre_adder_out = (OPMODE_mux[6])? (D_mux - B_mux): (D_mux + B_mux);
    assign {cout,P_post_adder_out} = (OPMODE_mux[7])? (Z_out-(X_out+CIN)): (Z_out + X_out + CIN);
    assign CARRYOUTF = CARRYOUT;
    assign PCOUT = P;
    //X MUX
    always @(*) begin
        case (OPMODE_mux[1:0])
            0: X_out = 48'd0;
            1: X_out = {12'd0,M_mux};
            2: X_out = PCOUT;
            3: X_out = {D_mux[11:0],mul_in2,mul_in1};
        endcase
    end
    //Z MUX
    always @(*) begin
        case (OPMODE_mux[3:2])
            0: Z_out =  48'd0;
            1: Z_out = PCIN;
            2: Z_out = P;
            3: Z_out = {12'd0,C_mux};
        endcase
    end
```

Test Bench

```
module tb();
    parameter ABD_WIDTH = 18;
    parameter CP_WIDTH = 48;
    parameter OPMODE_WIDTH = 8;
    parameter MUL_OUT_WIDTH = 36;
    reg CLK;
    reg RSTA,RSTB,RSTM,RSTP,RSTC,RSTD,RSTCARRYIN,RSTOPMODE;
    reg CEA,CEB,CEM,CEC,CED,CECARRYIN,CEOPMODE,CEP;
    reg [ABD_WIDTH-1 : 0] A,B,D,BCIN;
    reg [CP_WIDTH-1 : 0] C,PCIN;
    reg CARRYIN;
    reg [OPMODE_WIDTH-1 : 0]OPMODE;
    wire [CP_WIDTH-1 : 0]P_dut,PCOUT_dut;
    wire [ABD_WIDTH-1 : 0]BCOUT_dut;
    wire [MUL_OUT_WIDTH-1 : 0] M_dut;
    wire CARRYOUT_dut,CARRYOUTF_dut;
    reg [CP_WIDTH-1 : 0]P_ex,PCOUT_ex;
    reg [ABD_WIDTH-1 : 0]BCOUT_ex;
    reg [MUL_OUT_WIDTH-1 : 0] M_ex;
    reg CARRYOUT_ex,CARRYOUTF_ex;
    dsp DUT(A,B,C,D,CLK,CARRYIN,OPMODE,BCIN,RSTA,RSTB,RSTM,RSTP,RSTC,RSTD,RSTCARRYIN,RSTOPMODE,
    CEA,CEB,CEM,CEC,CED,CECARRYIN,CEOPMODE,CEP,PCIN,BCOUT_dut,PCOUT_dut,P_dut,M_dut,CARRYOUT_dut,CARRYOUTF_dut);
    initial begin
        CLK = 0;
        forever begin
            #1 CLK = ~CLK;
        end
    end
    initial begin
        RSTA=1; RSTB=1; RSTM=1; RSTP=1;RSTC=1; RSTD=1; RSTCARRYIN=1; RSTOPMODE=1;
        CEA=$random; CEB=$random; CEM=$random; CEC=$random; CED=$random; CECARRYIN=$random; CEOPMODE=$random; CEP=$random;
        A=$random; B=$random; D=$random; BCIN=$random; OPMODE=$random;
        C=$random; PCIN=$random; CARRYIN=$random;
        @(negedge CLK)
        if (P_dut !== 0 || PCOUT_dut !== 0 || BCOUT_dut !== 0 || M_dut !== 0 || CARRYOUT_dut !== 0 || CARRYOUTF_dut !== 0)begin
            $display("Error in reset condition");
            $stop;
        end
        RSTA=0; RSTB=0; RSTM=0; RSTP=0; RSTC=0; RSTD=0; RSTCARRYIN=0; RSTOPMODE=0;
        CEA=1; CEB=1; CEM=1; CEC=1; CED=1; CECARRYIN=1; CEOPMODE=1; CEP=1;
        //verify path 1
        A = 20; B = 10; C = 350; D = 25;
        OPMODE = 8'b11011101;
        BCIN = $random; PCIN = $random; CARRYIN = $random;
        BCOUT_ex = 18'hf;
```

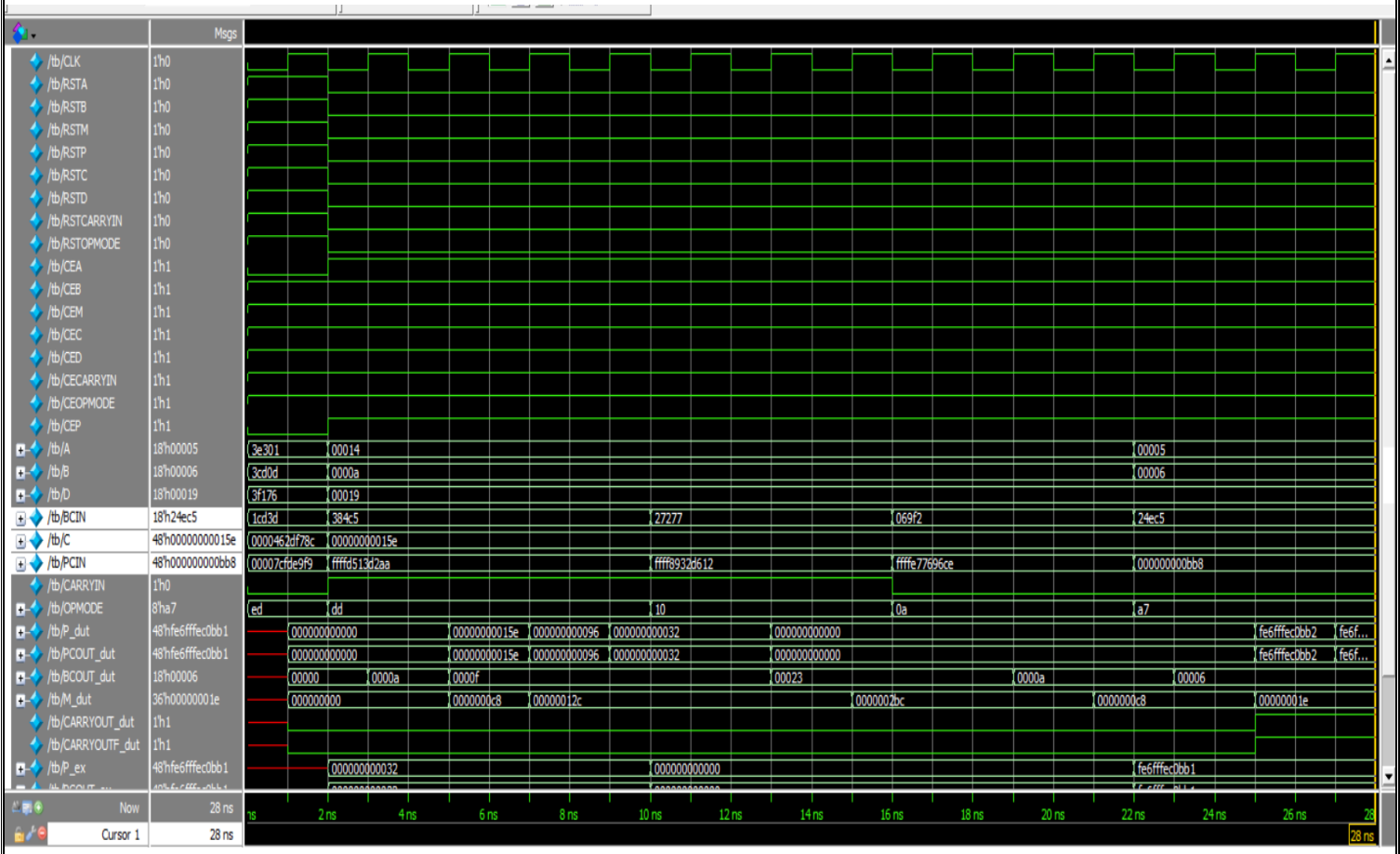
```
        M_ex = 36'h12c;
        P_ex = 48'h32;
        PCOUT_ex = 48'h32;
        CARRYOUT_ex = 0;
        CARRYOUTF_ex = 0;
        repeat(4)begin
          @(negedge CLK);
        end
        if (BCOUT_dut != BCOUT_ex || M_ex != M_dut || P_dut != P_ex || PCOUT_dut != PCOUT_ex || CARRYOUT_dut != CARRYOUT_ex || CARRYOUTF_dut!=CARRYOUTF_ex) begin
            $display("Error in Path 1");
            $stop;
        end
        //verify path 2
        A=20; B=10; C=350; D=25;
        OPMODE = 8'b00010000;
        BCIN = $random; PCIN = $random; CARRYIN = $random;
        BCOUT_ex = 18'h23;
        M_ex = 36'h2bc;
        P_ex = 48'h0;
        PCOUT_ex = 48'h0;
        CARRYOUT_ex = 0;
        CARRYOUTF_ex = 0;
        repeat(3)begin
          @(negedge CLK);
        end
        if (BCOUT_dut != BCOUT_ex || M_ex != M_dut || P_dut != P_ex || PCOUT_dut != PCOUT_ex || CARRYOUT_dut != CARRYOUT_ex || CARRYOUTF_dut!=CARRYOUTF_ex) begin
            $display("Error in Path 2");
            $stop;
        end
        //verify path 3
        A=20; B=10; C=350; D=25;
        OPMODE = 8'b00001010;
        BCIN = $random; PCIN = $random; CARRYIN = $random;
        BCOUT_ex = 18'ha;
        M_ex = 36'hc8;
        P_ex = 48'h0;
        PCOUT_ex = 48'h0;
        CARRYOUT_ex = 0;
        CARRYOUTF_ex = 0;
        repeat(3)begin
          @(negedge CLK);
        end
        if (BCOUT_dut != BCOUT_ex || M_ex != M_dut || P_dut != P_ex || PCOUT_dut != PCOUT_ex || CARRYOUT_dut != CARRYOUT_ex || CARRYOUTF_dut!=CARRYOUTF_ex) begin
            $display("Error in Path 3");
            $stop;
```

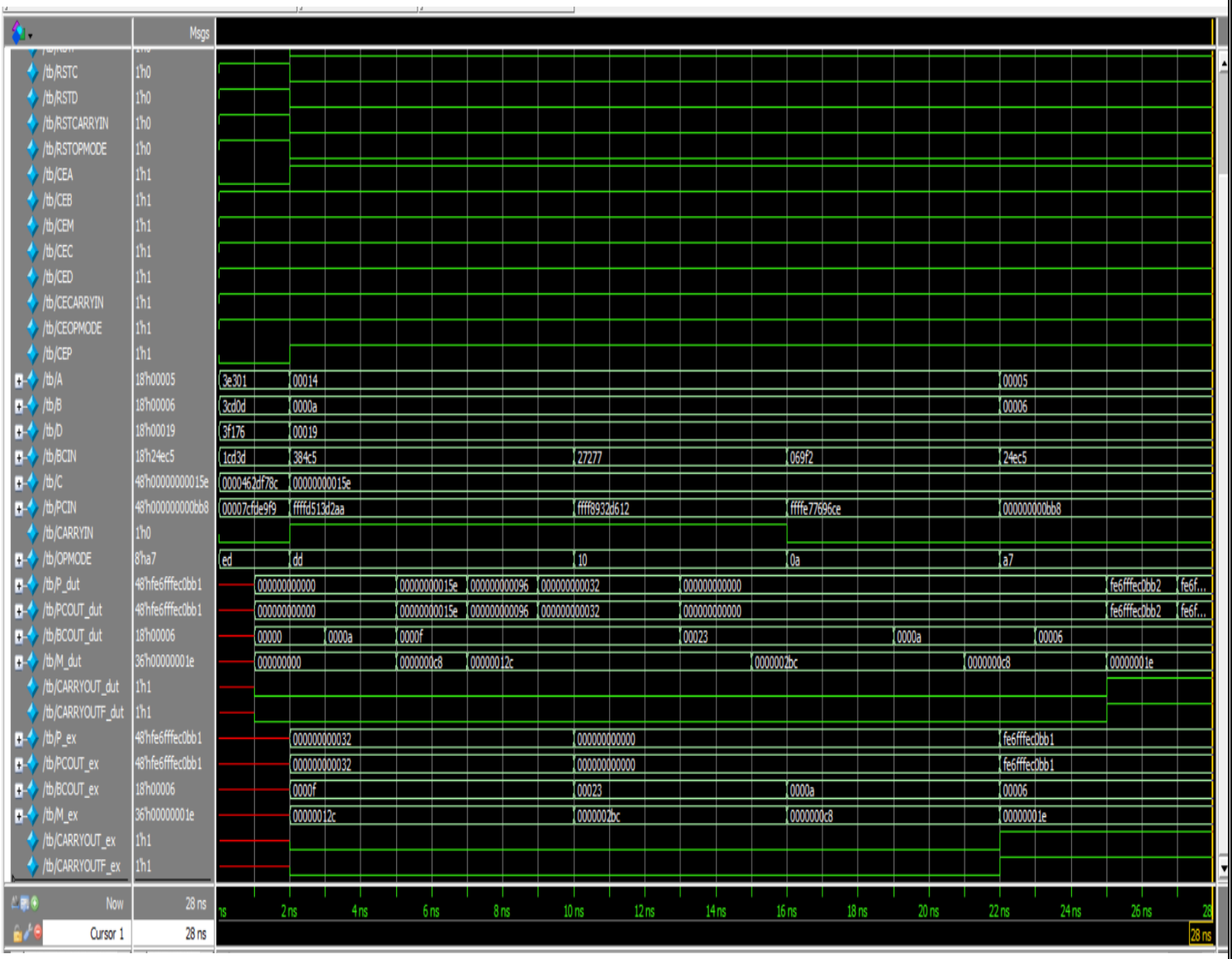
```
90          $stop;
91        end
92        //verify path 4
93        A=5; B=6; C=350; D=25;
94        OPMODE = 8'b10100111;
95        BCIN = $random; PCIN = 3000; CARRYIN = $random;
96        BCOUT_ex = 18'h6;
97        M_ex = 36'h1e;
98        P_ex = 48'hfe6fffec0bb1;
99        PCOUT_ex = 48'hfe6fffec0bb1;
100       CARRYOUT_ex = 1;
101       CARRYOUTF_ex = 1;
102       repeat(3)begin
103         @(negedge CLK);
104       end
105       if (BCOUT_dut != BCOUT_ex || M_ex != M_dut || P_dut != P_ex || PCOUT_dut != PCOUT_ex || CARRYOUT_dut != CARRYOUT_ex || CARRYOUTF_dut!=CARRYOUTF_ex) begin
106          $display("Error in Path 4");
107          $stop;
108       end
109       $stop;
110   end
111 endmodule
```

Do File

```
1   vlib work
2   vlog DSP.v MUX_FF.v DSP_tb.v
3   vsim -voptargs=+acc tb
4   add wave *
5   run -all
6   #quit -sim
7
```

Questa Sim Snippet

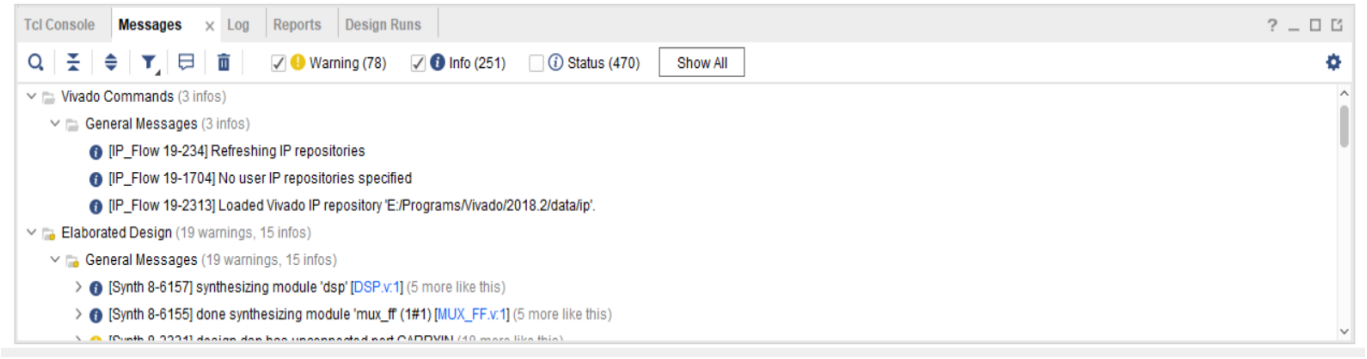


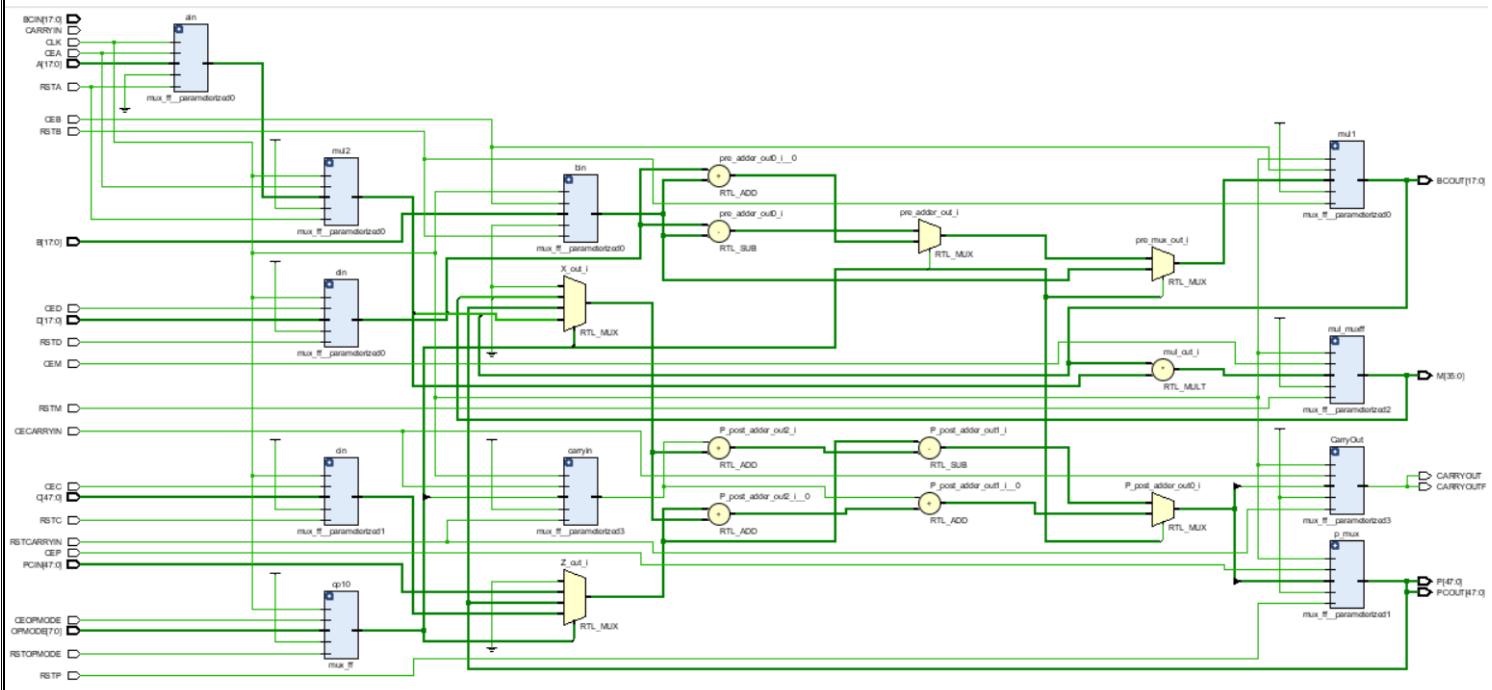

Msgs
/tb/RSTC 1'h0
/tb/RSTD 1'h0
/tb/RSTCARRYIN 1'h0
/tb/RSTOPMODE 1'h0
/tb/CEA 1'h1
/tb/CEB 1'h1
/tb/CEM 1'h1
/tb/CEC 1'h1
/tb/CED 1'h1
/tb/CECARRYIN 1'h1
/tb/CEOPMODE 1'h1
/tb/CEP 1'h1
/tb/A 18'h00005 3e301 00014 00005
/tb/B 18'h00006 3cd0d 0000a 00006
/tb/D 18'h00019 3f176 00019
/tb/BCIN 18'h24ec5 1cd3d 384c5 27277 069f2 24ec5
/tb/C 48'h00000000015e 0000462df78c 0000000015e
/tb/PCIN 48'h000000000bb8 00007cfde9f9 ffffd513d2aa ffff8932d612 ffffe77696ce 000000000bb8
/tb/CARRYIN 1'h0
/tb/OPMODE 8'ha7 ed dd 10 0a a7
/tb/P_dut 48'hfe6fffec0bb1 000000000000 0000000015e 000000000096 000000000032 000000000000 fe6fffec0bb2 fe6f...
/tb/PCOUT_dut 48'hfe6fffec0bb1 000000000000 0000000015e 000000000096 000000000032 000000000000 fe6fffec0bb2 fe6f...
/tb/BCOUT_dut 18'h00006 00000 0000a 0000f 00023 0000a 00006
/tb/M_dut 36'h00000001e 000000000 0000000c8 00000012c 0000002bc 0000000c8 00000001e
/tb/CARRYOUT_dut 1'h1
/tb/CARRYOUTF_dut 1'h1
/tb/P_ex 48'hfe6fffec0bb1 000000000032 000000000000 fe6fffec0bb1
/tb/PCOUT_ex 48'hfe6fffec0bb1 000000000032 000000000000 fe6fffec0bb1
/tb/BCOUT_ex 18'h00006 0000f 00023 0000a 00006
/tb/M_ex 36'h00000001e 0000012c 0000002bc 0000000c8 00000001e
/tb/CARRYOUT_ex 1'h1
/tb/CARRYOUTF_ex 1'h1
Now 28 ns
Cursor 1 28 ns
ns 2 ns 4 ns 6 ns 8 ns 10 ns 12 ns 14 ns 16 ns 18 ns 20 ns 22 ns 24 ns 26 ns
28 ns

Constrain File

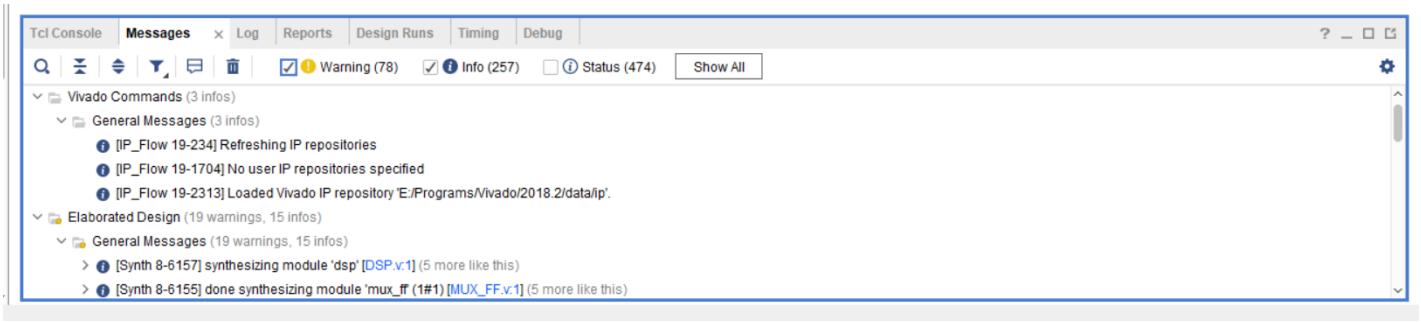
```
## This file is a general .xdc for the Basys3 rev B board
## To use it in a project:
## - uncomment the lines corresponding to used pins
## - rename the used ports (in each line, after get_ports) according to the top level signal names in the proj

## Clock signal
set_property -dict { PACKAGE_PIN W5   IOSTANDARD LVCMOS33 } [get_ports CLK]
create_clock -add -name sys_clk_pin -period 10 -waveform {0 5} [get_ports CLK]


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


## LEDs
#set_property -dict { PACKAGE_PIN U16   IOSTANDARD LVCMOS33 } [get_ports {led[0]}]
#set_property -dict { PACKAGE_PIN E19   IOSTANDARD LVCMOS33 } [get_ports {led[1]}]
#set_property -dict { PACKAGE_PIN U19   IOSTANDARD LVCMOS33 } [get_ports {led[2]}]
#set_property -dict { PACKAGE_PIN V19   IOSTANDARD LVCMOS33 } [get_ports {led[3]}]
#set_property -dict { PACKAGE_PIN W18   IOSTANDARD LVCMOS33 } [get_ports {led[4]}]
#set_property -dict { PACKAGE_PIN U15   IOSTANDARD LVCMOS33 } [get_ports {led[5]}]
#set_property -dict { PACKAGE_PIN U14   IOSTANDARD LVCMOS33 } [get_ports {led[6]}]
#set_property -dict { PACKAGE_PIN V14   IOSTANDARD LVCMOS33 } [get_ports {led[7]}]
#set_property -dict { PACKAGE_PIN V13   IOSTANDARD LVCMOS33 } [get_ports {led[8]}]
#set_property -dict { PACKAGE_PIN V3    IOSTANDARD LVCMOS33 } [get_ports {led[9]}]
#set_property -dict { PACKAGE_PIN W3    IOSTANDARD LVCMOS33 } [get_ports {led[10]}]
#set_property -dict { PACKAGE_PIN U3    IOSTANDARD LVCMOS33 } [get_ports {led[11]}]
#set_property -dict { PACKAGE_PIN P3    IOSTANDARD LVCMOS33 } [get_ports {led[12]}]
#set_property -dict { PACKAGE_PIN N3    IOSTANDARD LVCMOS33 } [get_ports {led[13]}]
#set_property -dict { PACKAGE_PIN P1    IOSTANDARD LVCMOS33 } [get_ports {led[14]}]
```

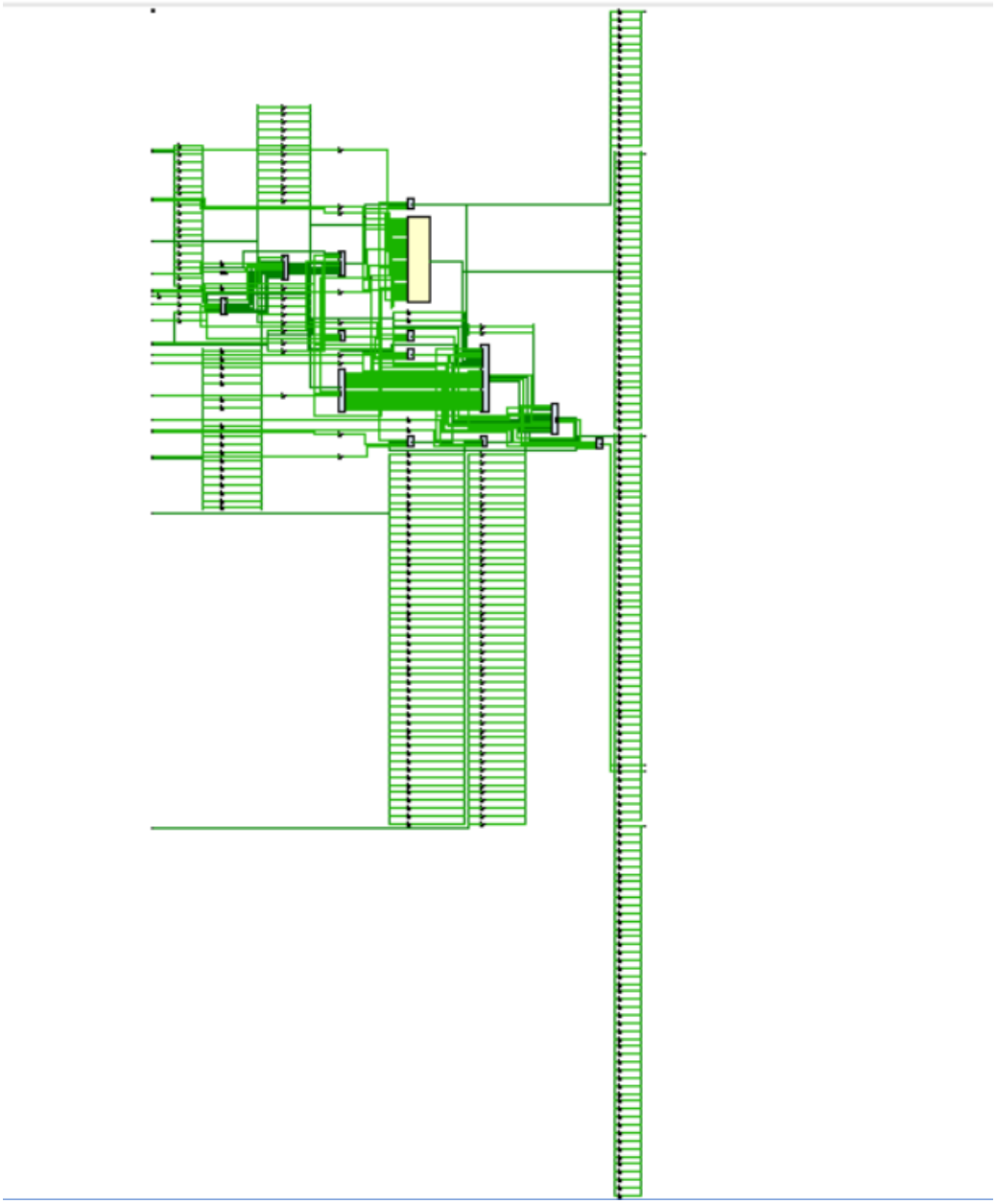
Elaboration

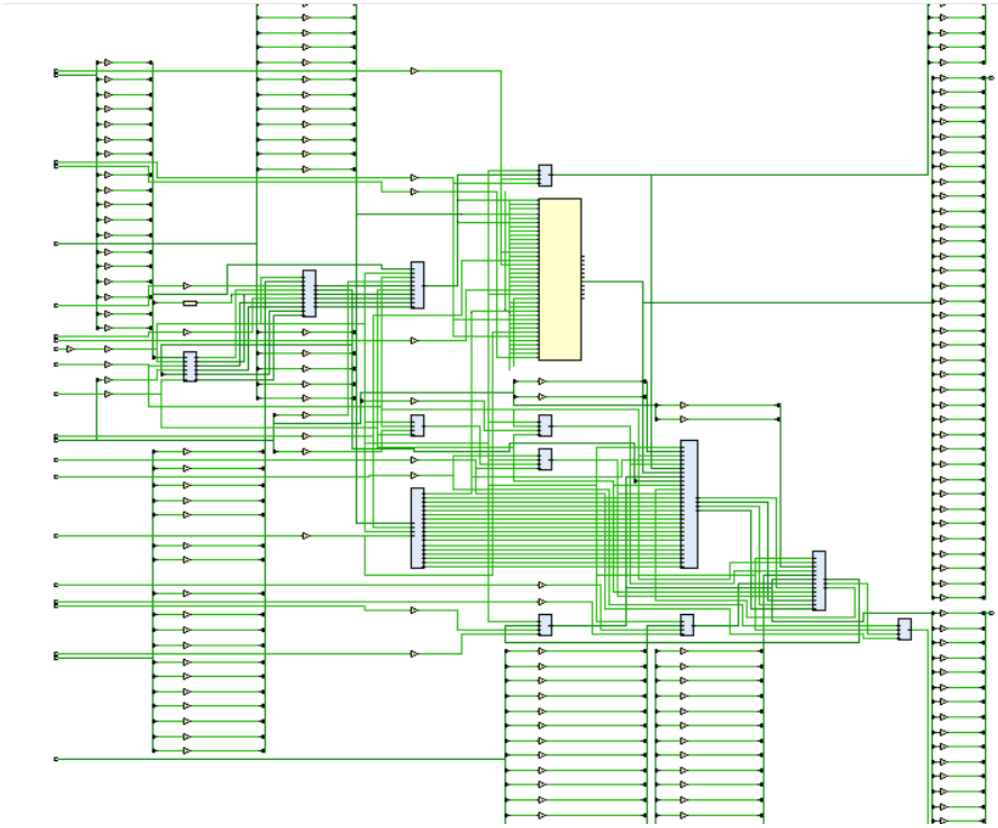




Synthesis


Tcl Console
Messages
Log
Reports
Design Runs
Timing
Debug
Warning (78)
Info (257)
Status (474)
Show All
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 'E:/Programs/Vivado/2018.2/data/ip'.
Elaborated Design (19 warnings, 15 infos)
General Messages (19 warnings, 15 infos)
[Synth 8-6157] synthesizing module 'dsp' [DSP.v:1] (5 more like this)
[Synth 8-6155] done synthesizing module 'mux_ff' (1#1) [MUX_FF.v:1] (5 more like this)





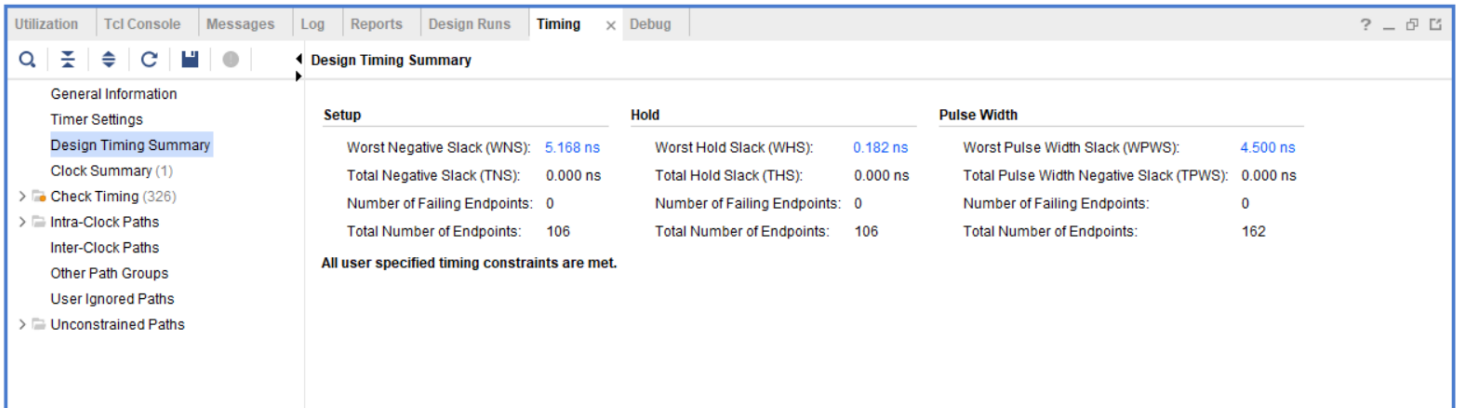
Utilization

Hierarchy

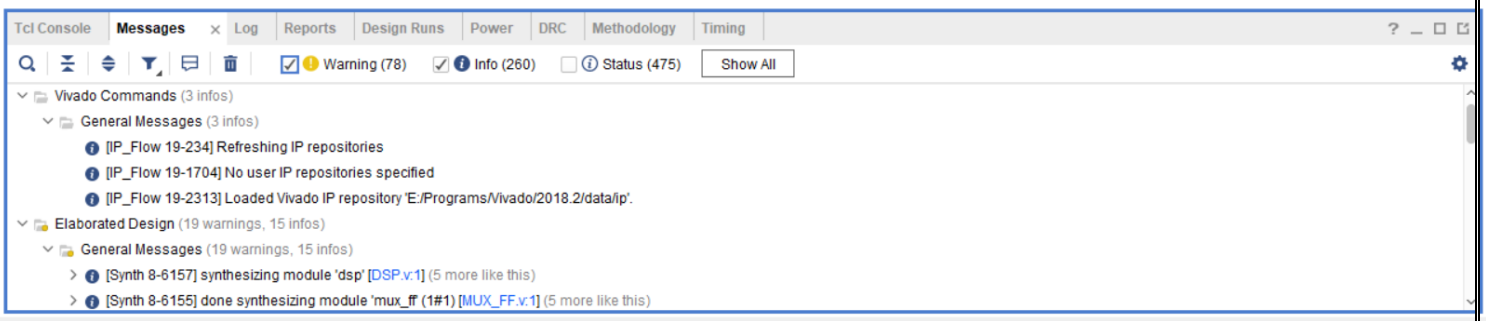
- Hierarchy
- Summary
- Slice Logic
  - Slice LUTs (<1%)
    - LUT as Logic (<1%)
  - Slice Registers (<1%)
    - Register as Flip Flop (<
- Memory
- DSP
  - DSPs (<1%)
    - DSP48E1 only
- IO and GT Specific
  - Bonded IOB (65%)
    - IOB Master Pads
- Clocking
  - BUFGCTRL (3%)
- Specific Feature
- Primitives
- Black Boxes
- Instantiated Netlists

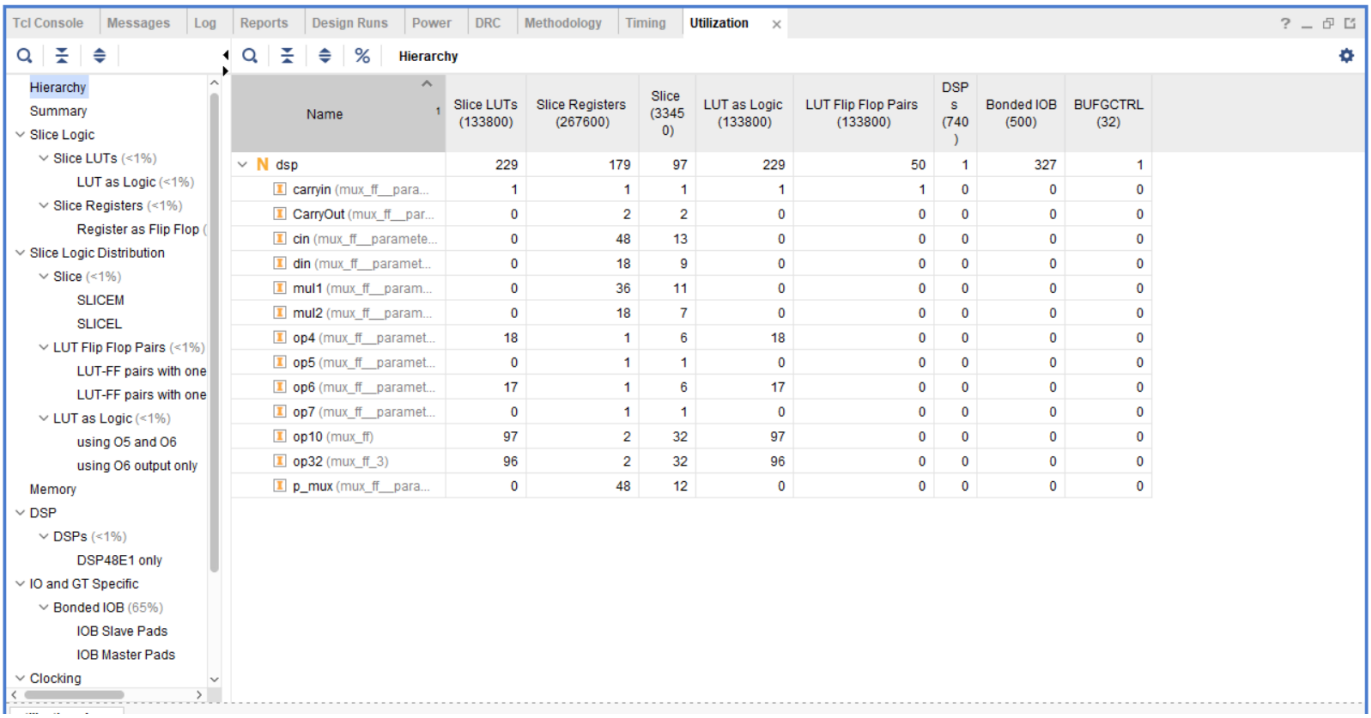
Name	Slice LUTs (134600)	Slice Registers (269200)	DSPs (740)	Bonded IOB (500)	BUFGCTRL (32)
N dsp	230	160	1	327	1
carryin (mux_ff__para...	1	1	0	0	0
CarryOut (mux_ff__par...	0	1	0	0	0
cin (mux_ff__paramete...	0	48	0	0	0
din (mux_ff__paramet...	0	18	0	0	0
mul1 (mux_ff__param...	0	18	0	0	0
mul2 (mux_ff__param...	0	18	0	0	0
op4 (mux_ff__paramet...	18	1	0	0	0
op5 (mux_ff__paramet...	0	1	0	0	0
op6 (mux_ff__paramet...	17	1	0	0	0
op7 (mux_ff__paramet...	0	1	0	0	0
op10 (mux_ff)	97	2	0	0	0
op32 (mux_ff_3)	96	2	0	0	0
p_mux (mux_ff__para...	0	48	0	0	0

utilization_1



Implementation





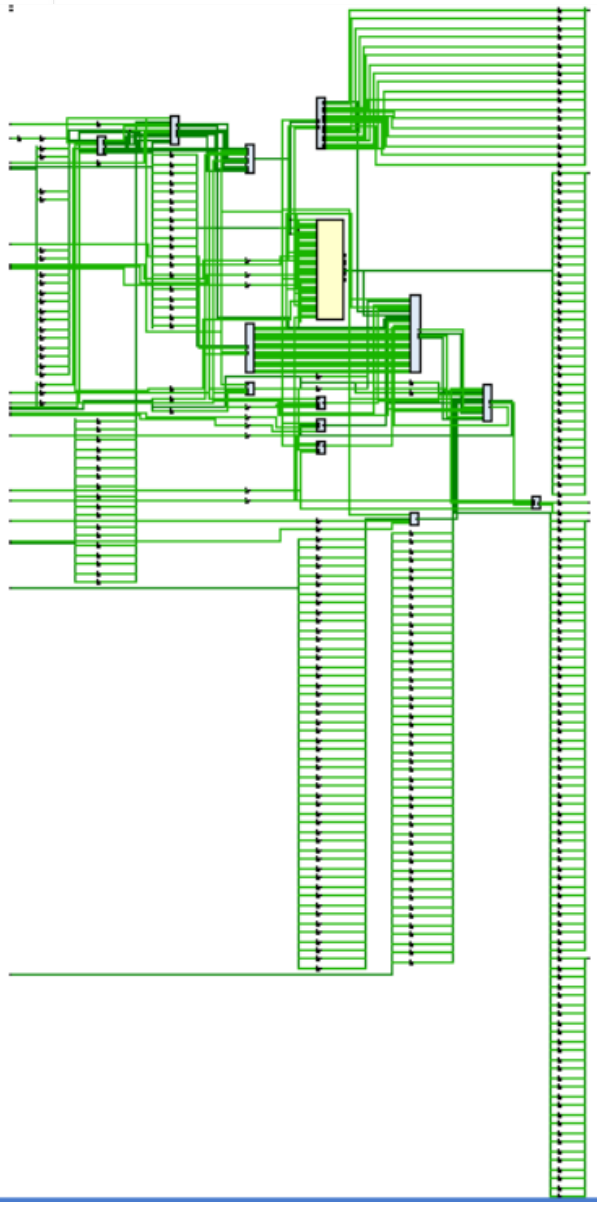
Tcl Console | Messages | Log | Reports | Design Runs | Power | DRC | Methodology | Timing | Utilization

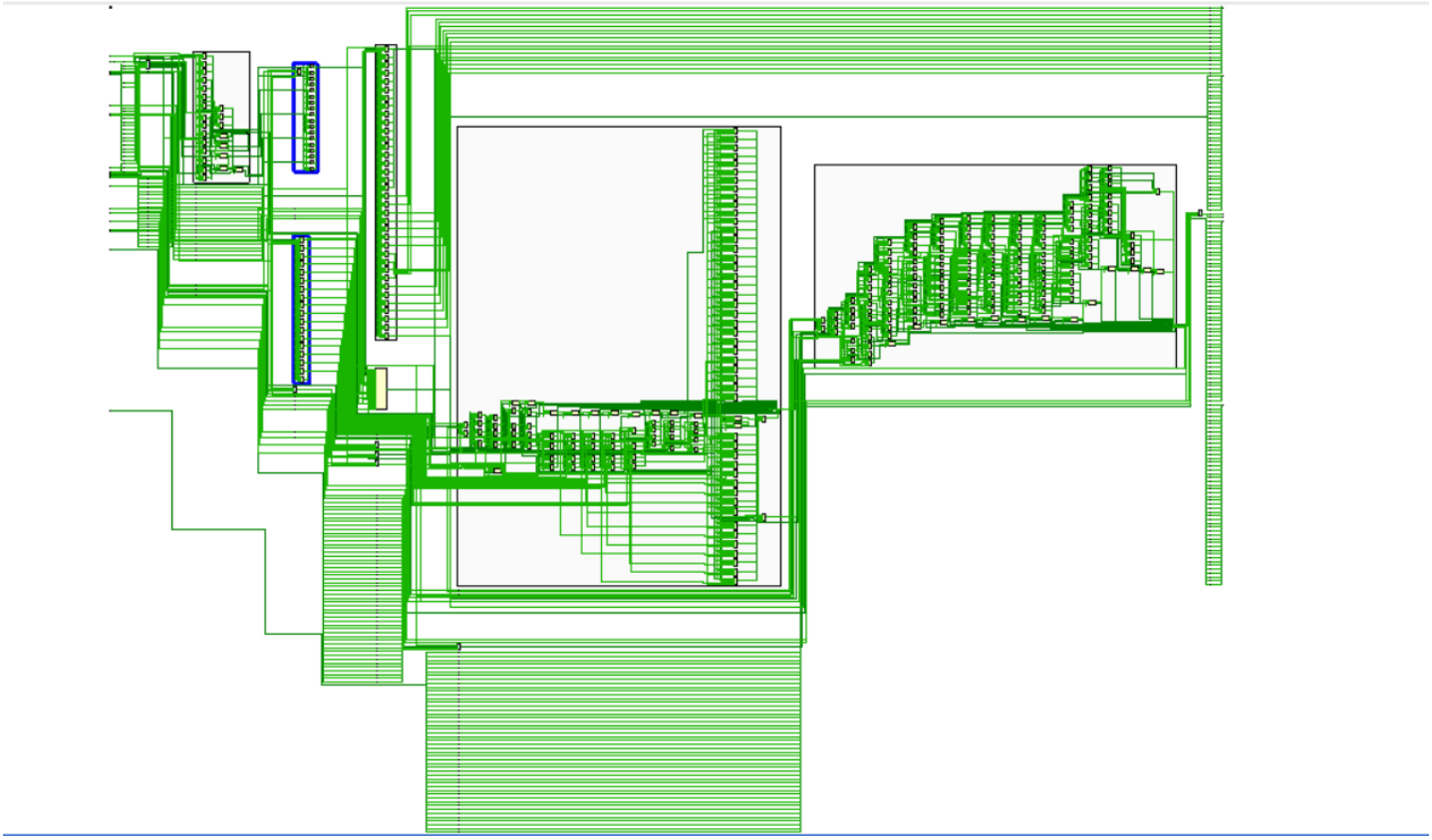
- General Information
- Timer Settings
- Design Timing Summary
- Clock Summary (1)
- Check Timing (326)
- Intra-Clock Paths
- Inter-Clock Paths
- Other Path Groups
- User Ignored Paths
- Unconstrained Paths

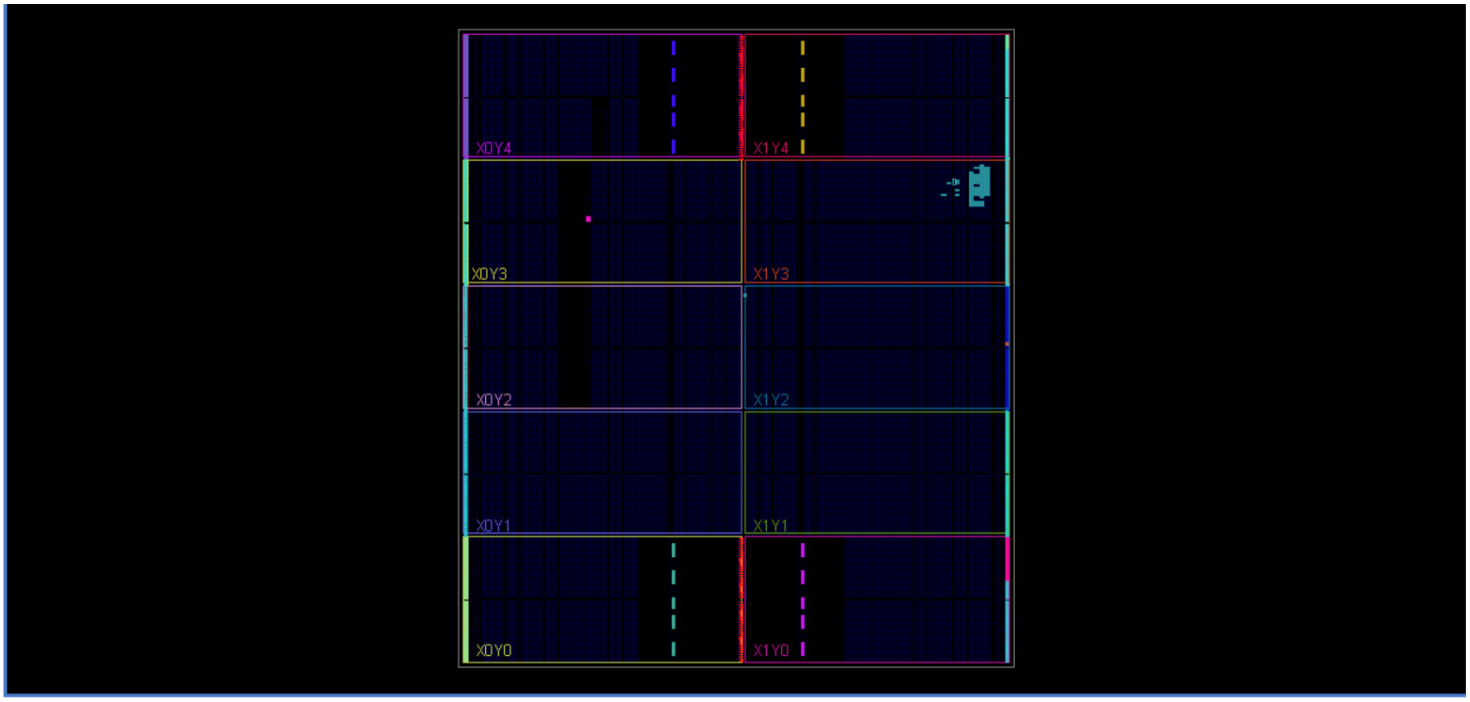
Design Timing Summary

Setup		Hold		Pulse Width	
Worst Negative Slack (WNS):	3.804 ns	Worst Hold Slack (WHS):	0.273 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:	125	Total Number of Endpoints:	125	Total Number of Endpoints:	181

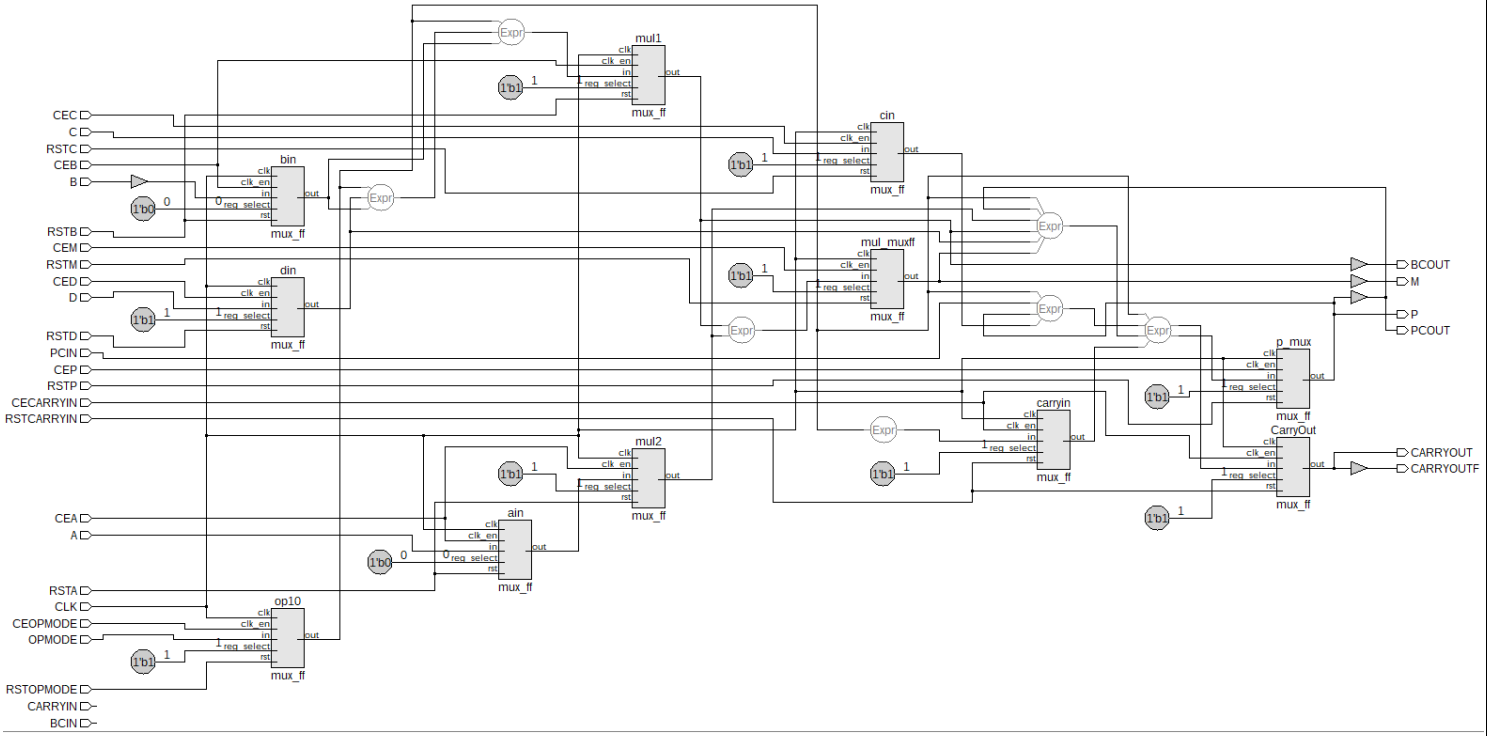
All user specified timing constraints are met.






X0Y4
X1Y4
X0Y3
X1Y3
X0Y2
X1Y2
X0Y1
X1Y1
X0Y0
X1Y0

Lint Tool


CEC
C
RSTC
CEB
B
RSTB
CEM
RSTM
CED
D
RSTD
PCIN
CEP
RSTP
CECARRYIN
RSTCARRYIN
CEA
A
RSTA
CLK
CEOPMODE
OPMODE
RSTOPMODE
CARRYIN
BCIN
BCOUT
M
P
PCOUT
CARRYOUT
CARRYOUTF
bin
din
ain
op10
mul1
mul2
cin
mul_muxff
carryin
p_mux
CarryOut
mux_ff
Expr
1'b0
1'b1