Engineering Sector
Digital Electronics Design Diploma
Eng. Kareem Waseem

# Project 1

# Spartan-6 FPGA DSP48A1 Slice

## BY: Youssef Ekramy

Under the supervision of:

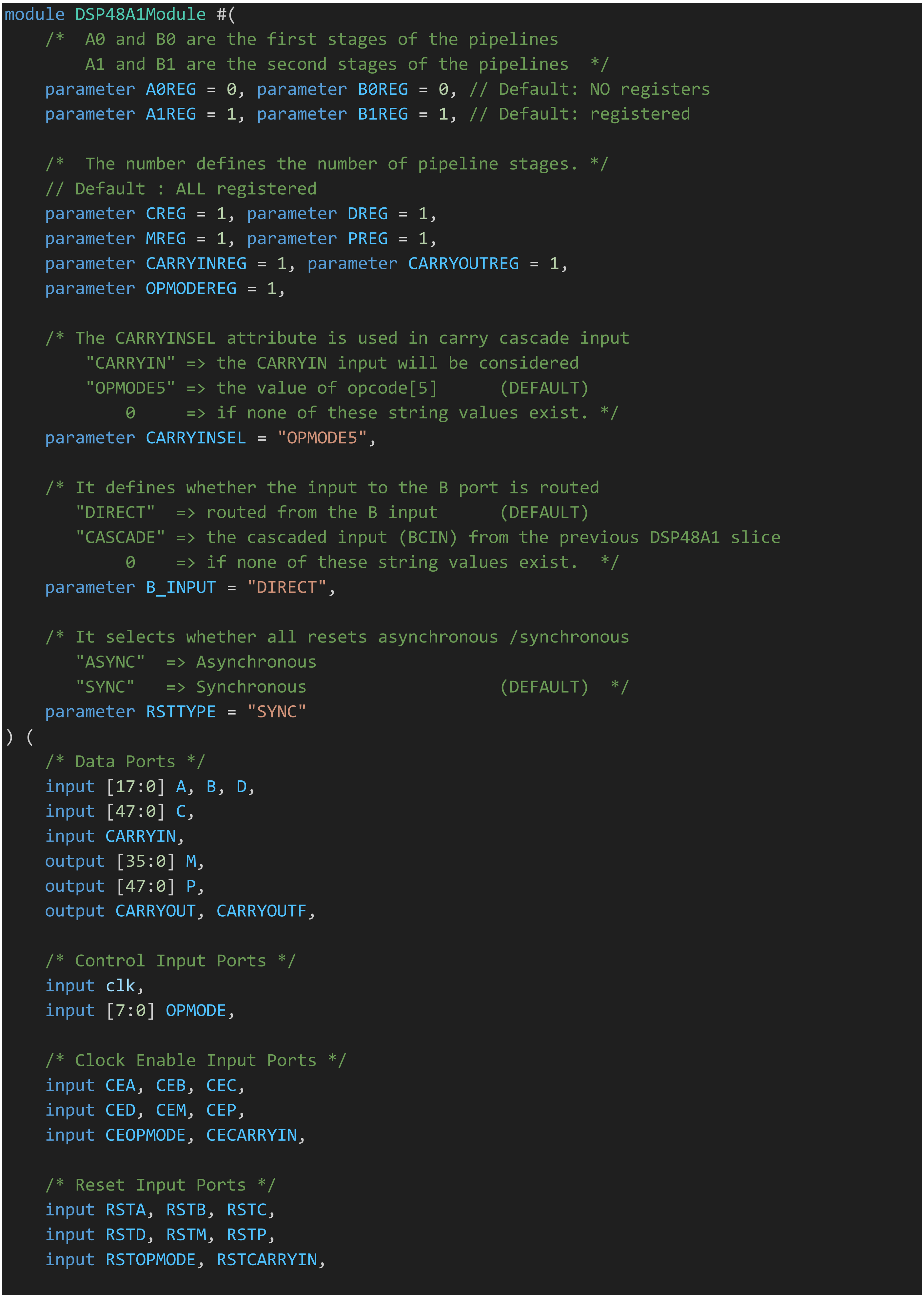
**Eng. Kareem Waseem**

# Table of Contents

# 1 DSP 48A1 Module

## 1.1 Code

```
module reg_mux_pipeline #(
    /* This attribute determines if there is register or not
        1 => Registered                 (DEFAULT)
        0 => NO Register                                 */
    parameter REG = 1,

    /* This attribute determines the width of the inputs */
    parameter WIDTH = 18,

    /* It selects whether all resets asynchronous /synchronous
       "ASYNC"  => Asynchronous
       "SYNC"   => Synchronous                   (DEFAULT)  */
    parameter RSTTYPE = "SYNC"
) (
    input clk, enable, rst,
    input [WIDTH-1:0] in,
    output [WIDTH-1:0] out
);

    generate
        /* Create register and mux */
        if (REG == 1) begin
            reg [WIDTH-1:0] in_r;
            if(RSTTYPE == "ASYNC")
                always @(posedge clk or posedge rst) begin
                    if(rst)
                        in_r <= 0;
                    else if(enable)
                        in_r <= in;
                end
            else if(RSTTYPE == "SYNC")
                always @(posedge clk) begin
                    if(rst)
                        in_r <= 0;
                    else if(enable)
                        in_r <= in;
                end
            assign out = in_r;
        end

        else
            assign out = in;

    endgenerate


endmodule
```



```
module DSP48A1Module #(
    /*  A0 and B0 are the first stages of the pipelines
        A1 and B1 are the second stages of the pipelines  */
    parameter A0REG = 0, parameter B0REG = 0, // Default: NO registers
    parameter A1REG = 1, parameter B1REG = 1, // Default: registered

    /*  The number defines the number of pipeline stages. */
    // Default : ALL registered
    parameter CREG = 1, parameter DREG = 1,
    parameter MREG = 1, parameter PREG = 1,
    parameter CARRYINREG = 1, parameter CARRYOUTREG = 1,
    parameter OPMODEREG = 1,

    /* The CARRYINSEL attribute is used in carry cascade input
        "CARRYIN" => the CARRYIN input will be considered
        "OPMODE5" => the value of opcode[5]      (DEFAULT)
            0     => if none of these string values exist. */
    parameter CARRYINSEL = "OPMODE5",

    /* It defines whether the input to the B port is routed
       "DIRECT"  => routed from the B input      (DEFAULT)
       "CASCADE" => the cascaded input (BCIN) from the previous DSP48A1 slice
            0    => if none of these string values exist.  */
    parameter B_INPUT = "DIRECT",

    /* It selects whether all resets asynchronous /synchronous
       "ASYNC"  => Asynchronous
       "SYNC"   => Synchronous                   (DEFAULT)  */
    parameter RSTTYPE = "SYNC"
) (
    /* Data Ports */
    input [17:0] A, B, D,
    input [47:0] C,
    input CARRYIN,
    output [35:0] M,
    output [47:0] P,
    output CARRYOUT, CARRYOUTF,

    /* Control Input Ports */
    input clk,
    input [7:0] OPMODE,

    /* Clock Enable Input Ports */
    input CEA, CEB, CEC,
    input CED, CEM, CEP,
    input CEOPMODE, CECARRYIN,

    /* Reset Input Ports */
    input RSTA, RSTB, RSTC,
    input RSTD, RSTM, RSTP,
    input RSTOPMODE, RSTCARRYIN,
    /* Cascade Ports */
    input [17:0] BCIN,
    output [17:0] BCOUT,
    input [47:0] PCIN,
    output [47:0] PCOUT
);
    /*  First pipeline stage of inputs
        instance name in_(name of Input)
        in case we have two stages we add 0 for stage 1 */

    reg [17:0] B0_in;    // Handling the input of B0

    always @(*) begin
        case(B_INPUT)
        "DIRECT": B0_in = B;
        "CASCADE": B0_in = BCIN;
        default: B0_in = 0;
        endcase
    end


    wire [17:0] A0_out, B0_out, D_out; // Outputs of first pipeline stage
    wire [47:0] C_out; wire [7:0] OPMODE_out;

    reg_mux_pipeline #(.REG(A0REG),.RSTTYPE(RSTTYPE)) in_A0 (.clk(clk),
                                                            .enable(CEA),
                                                            .rst(RSTA),
                                                            .in(A),
                                                            .out(A0_out));

    reg_mux_pipeline #(.REG(B0REG),.RSTTYPE(RSTTYPE)) in_B0 (.clk(clk),
                                                            .enable(CEB),
                                                            .rst(RSTB),
                                                            .in(B0_in),
                                                            .out(B0_out));

    reg_mux_pipeline #(.REG(CREG),.WIDTH(48),.RSTTYPE(RSTTYPE)) in_C (.clk(clk),
                                                                    .enable(CEC),
                                                                    .rst(RSTC),
                                                                    .in(C),
                                                                    .out(C_out));

    reg_mux_pipeline #(.REG(DREG),.RSTTYPE(RSTTYPE)) in_D  (.clk(clk),
                                                            .enable(CED),
                                                            .rst(RSTD),
                                                            .in(D),
                                                            .out(D_out));

    reg_mux_pipeline #(.REG(OPMODEREG),.WIDTH(8),.RSTTYPE(RSTTYPE)) in_OPMODE  (.clk(clk),
                                                                                .enable(CE
OPMODE),
                                                                                .rst(RSTOP
MODE),
                                                                                .in(OPMODE
),
                                                                                .out(OPMOD
E_out));
    /*  PRE-ADDER/SUBTRACTOR level*/
    wire [17:0] PRE_ADD_SUB_out;
    assign PRE_ADD_SUB_out = (OPMODE_out[6] == 0)?
                             (D_out + B0_out):
                             (D_out - B0_out);


    /*  Second pipeline stage of inputs
        instance name in_(name of Input)
        in case we have two stages we add 1 for stage 2 */

    wire [17:0] B1_in;    // Handling the input of B1
    assign B1_in = (OPMODE_out[4] == 0)? B0_out : PRE_ADD_SUB_out;
    wire [17:0] A1_out, B1_out;

    reg_mux_pipeline #(.REG(A1REG),.RSTTYPE(RSTTYPE)) in_A1 (.clk(clk),
                                                            .enable(CEA),
                                                            .rst(RSTA),
                                                            .in(A0_out),
                                                            .out(A1_out));

    reg_mux_pipeline #(.REG(B1REG),.RSTTYPE(RSTTYPE)) in_B1 (.clk(clk),
                                                            .enable(CEB),
                                                            .rst(RSTB),
                                                            .in(B1_in),
                                                            .out(B1_out));

    assign BCOUT = B1_out;

    /* MULTIPLIER level */
    wire [35:0] Multiplier_out;
    assign Multiplier_out = A1_out * B1_out;

    wire [35:0] M_out;

    reg_mux_pipeline #(.REG(MREG),.WIDTH(36),.RSTTYPE(RSTTYPE)) in_M (.clk(clk),
                                                                    .enable(CEM),
                                                                    .rst(RSTM),
                                                                    .in(Multiplier_out),
                                                                    .out(M_out));

    assign M = M_out; //Buffered M Output

    /* MUX X level */
    reg [47:0] X_out;
    wire [47:0] P_out;
    always @(*) begin
        case (OPMODE_out[1:0])
            0: X_out = 0;  // disable the post-adder/subtracter and propagate the MUX Z 
result to P
            1: X_out = {{12{1'b0}},M_out}; // Use the multiplier product
            2: X_out = P_out;    // Use the P output signal (accumulator)
            3: X_out = {D_out[11:0],A1_out[17:0],B1_out[17:0]}; // Use the concatenated 
D:A:B input signals
        endcase
    end

    /* MUX Z level */
    reg [47:0] Z_out;
    always @(*) begin
        case (OPMODE_out[3:2])
            0: Z_out = 0; //disable the post-adder/subtracter and propagate the multiplier 
product or other X result to P)
            1: Z_out = PCIN;
            2: Z_out = P_out;
            3: Z_out = C_out;
        endcase
    end

    /* CARRYIN POST ADDER/SUBTRACTOR level */
    reg CYI_in;
    wire CYI_out;

    always @(*) begin
        case (CARRYINSEL)
            "CARRYIN": CYI_in = CARRYIN;
            "OPMODE5": CYI_in = OPMODE_out[5];
            default: CYI_in = 0;
        endcase
    end

    reg_mux_pipeline #(.REG(CARRYINREG),.WIDTH(1),.RSTTYPE(RSTTYPE)) in_CYI(.clk(clk),
                                                                             .enable(CECARR
YIN),
                                                                             .rst(RSTCARRYI
N),
                                                                             .in(CYI_in),
                                                                             .out(CYI_out))
;

    /*  POST-ADDER/SUBTRACTOR level*/
    wire [47:0] POST_ADD_SUB_out;
    wire CYO_in,CYO_out;
    assign {CYO_in,POST_ADD_SUB_out} =  (OPMODE_out[7] == 0)?
                                        (Z_out + X_out + CYI_out):
                                        (Z_out - (X_out + CYI_out));

    /* P pipeline register */
    reg_mux_pipeline #(.REG(PREG),.WIDTH(48),.RSTTYPE(RSTTYPE)) in_P(.clk(clk),
                                                                    .enable(CEP),
                                                                    .rst(RSTP),
                                                                    .in(POST_ADD_SUB_out),
                                                                    .out(P_out));

    assign P = P_out; assign PCOUT = P_out;

    /* CARRYOUT POST ADDER/SUBTRACTOR level */
    reg_mux_pipeline #(.REG(CARRYOUTREG),.WIDTH(1),.RSTTYPE(RSTTYPE)) in_CYO(.clk(clk),
                                                                            .enable(CECARR
YIN),
                                                                            .rst(RSTCARRYI
N),
                                                                            .in(CYO_in),
                                                                            .out(CYO_out))
;

    assign CARRYOUT = CYO_out; assign CARRYOUTF =
CYO_out;
endmodule
```

## 1.2 Testbench Code

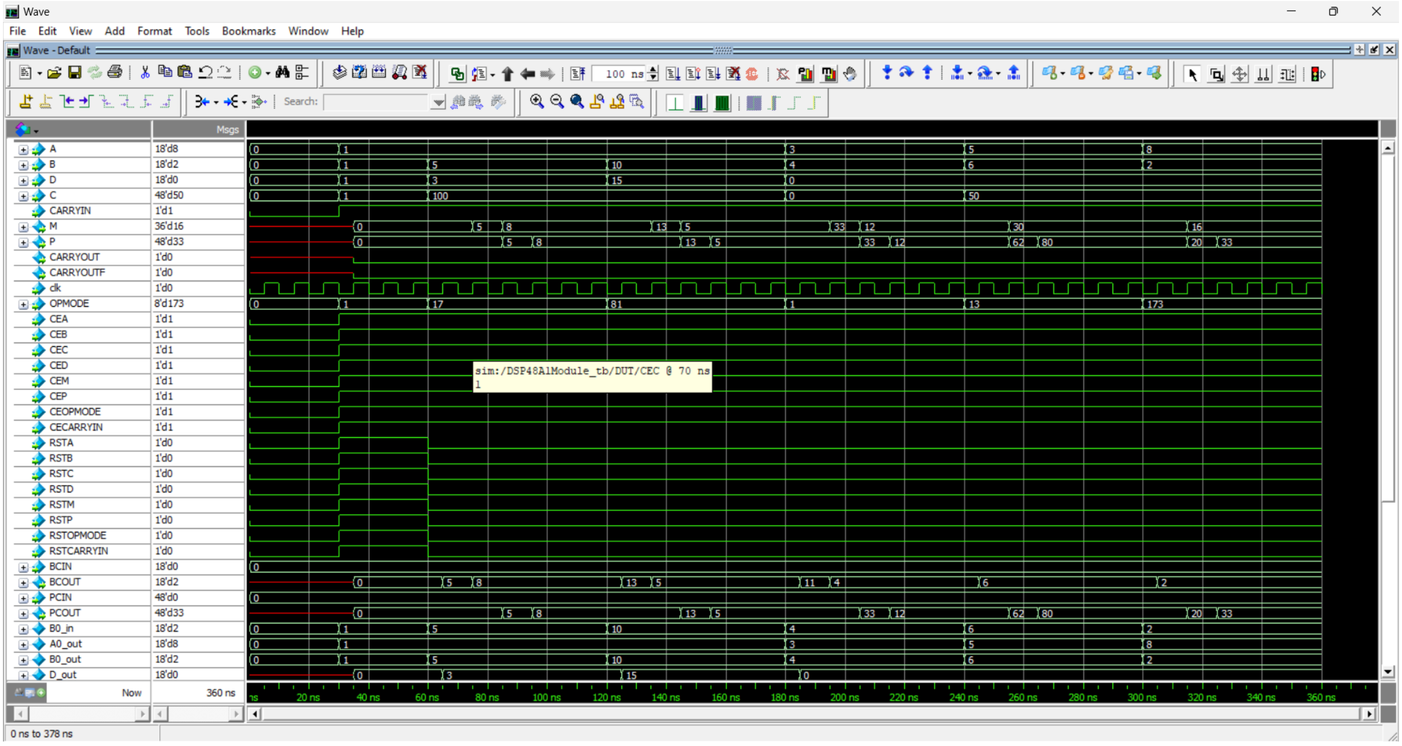
```
module DSP48A1Module_tb();
    /* Add All the parameters */
    parameter A0REG = 0; parameter B0REG = 0;
    parameter A1REG = 1; parameter B1REG = 1;
    parameter CREG  = 1; parameter DREG  = 1;
    parameter MREG  = 1; parameter PREG  = 1;

    parameter CARRYINREG = 1; parameter CARRYOUTREG = 1;
    parameter OPMODEREG = 1;

    parameter CARRYINSEL = "OPMODE5"; parameter B_INPUT = "DIRECT";
    parameter RSTTYPE = "SYNC";

    /* Add all the inputs */

    /* Data Ports */
    reg [17:0] A, B, D;
    reg [47:0] C;
    reg CARRYIN;
    wire [35:0] M;
    wire [47:0] P;
    wire CARRYOUT, CARRYOUTF;

    /* Control Input Ports */
    reg clk;
    reg [7:0] OPMODE;

    /* Clock Enable Input Ports */
    reg CEA, CEB, CEC;
    reg CED, CEM, CEP;
    reg CEOPMODE, CECARRYIN;

    /* Reset Input Ports */
    reg RSTA, RSTB, RSTC;
    reg RSTD, RSTM, RSTP;
    reg RSTOPMODE, RSTCARRYIN;

    /* Cascade Ports */
    reg [17:0] BCIN;
    wire [17:0] BCOUT;
    reg [47:0] PCIN;
    wire [47:0] PCOUT;

    /* Self Checker Port*/
    reg [47:0] P_expected;

    /* Instantiate the DUT */
    DSP48A1Module #(.A0REG(A0REG),.A1REG(A1REG),
                    .B0REG(B0REG),.B1REG(B1REG),
                    .CREG(CREG),.DREG(DREG),
                    .MREG(MREG),.PREG(PREG),
                    .CARRYINREG(CARRYINREG),.CARRYOUTREG(CARRYOUTREG),
                    .OPMODEREG(OPMODEREG),
                    .CARRYINSEL(CARRYINSEL),.B_INPUT(B_INPUT),
                    .RSTTYPE(RSTTYPE))
                    DUT (.*);

    /* Clock Generation */
    initial begin
         clk = 0;
         forever begin
         #5; clk = ~clk;
         end
    end

    /* DIRECTED and RANDOMIZED TESTBENCH */
    initial begin
        $display ("START THE SIMULATION");
        $display ("-------------------------");
        /* Initialize Inputs */
        A = 0; B = 0; D = 0; C = 0;
        CARRYIN = 0; OPMODE = 0;
        CEA = 0; CEB = 0; CEC = 0;
        CED = 0; CEM = 0; CEP = 0;
        CEOPMODE = 0; CECARRYIN = 0;
        RSTA = 0; RSTB = 0; RSTC = 0;
        RSTD = 0; RSTM = 0; RSTP = 0;
        RSTOPMODE = 0; RSTCARRYIN = 0;
        BCIN = 0; PCIN = 0;
        repeat(3) @(negedge clk);

        /* Check the reset functionality */
        // Initialize all the inputs by 1
        A = 1; B = 1; D = 1; C = 1;
        CARRYIN = 1; OPMODE = 1;

        // Initialize clock enables by 1
        CEA = 1; CEB = 1; CEC = 1;
        CED = 1; CEM = 1; CEP = 1;
        CEOPMODE = 1; CECARRYIN = 1;

        // Initialize reset by 1
        RSTA = 1; RSTB = 1; RSTC = 1;
        RSTD = 1; RSTM = 1; RSTP = 1;
        RSTOPMODE = 1; RSTCARRYIN = 1;
        repeat(3) @(negedge clk);

        // Check the reset functionality
        $display ("Reset_functionality TEST");
        self_checker(P,0);
        $display ("-------------------------");
```

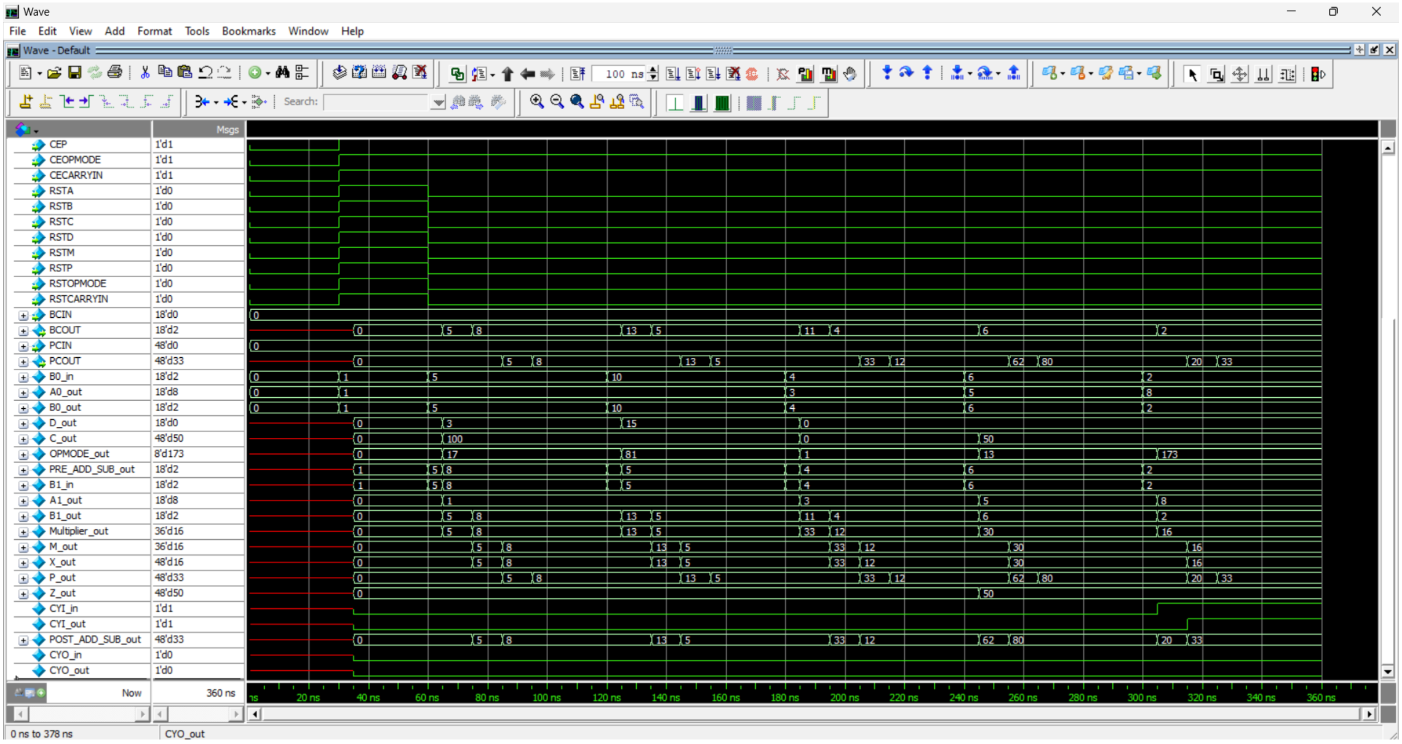
```
        // Initialize reset by 0
        RSTA = 0; RSTB = 0; RSTC = 0;
        RSTD = 0; RSTM = 0; RSTP = 0;
        RSTOPMODE = 0; RSTCARRYIN = 0;


        /* PRE-ADDER output (Addition) */
        A = 18'd1;
        B = 18'd5;
        D = 18'd3;
        C = 48'd100;
        OPMODE = 8'b0001_0001;
        repeat(4) @(negedge clk);
        $display("PRE_ADDER_Addition TEST");
        self_checker(P,(B + D) * A);
        repeat(2) @(negedge clk);
        $display ("-------------------------");

        /* PRE-ADDER output (Subtraction) */
        A = 18'd1;
        B = 18'd10;
        D = 18'd15;
        OPMODE = 8'b0101_0001;
        repeat(4) @(negedge clk);
        $display("PRE_ADDER_Subtraction TEST");
        self_checker(P,(D - B));
        repeat(2) @(negedge clk);
        $display ("-------------------------");

        /* MULTIPLIER output */
        A = 18'd3;
        B = 18'd4;
        C = 48'd0;
        D = 18'd0;
        OPMODE = 8'b0000_0001;
        repeat(4) @(negedge clk);
        $display("MULTIPLIER TEST");
        self_checker(P,(A * B));
        repeat(2) @(negedge clk);
        $display ("-------------------------");

        /* POST-ADDER output (Addition) */
        A = 18'd5;
        B = 18'd6;
        C = 48'd50;
        OPMODE = 8'b0000_1101;
        repeat(4) @(negedge clk);
        $display("POST_ADDER_Addition TEST");
        self_checker(P,((A * B) + C));
        repeat(2) @(negedge clk);
        $display ("-------------------------");
        /* POST-ADDER output (Subtraction) */
        A = 18'd8;
        B = 18'd2;
        D = 18'd0;
        C = 48'd50;
        OPMODE = 8'b1010_1101;
        repeat(4) @(negedge clk);
        $display("POST_ADDER_Subtraction TEST");
        self_checker(P,(C - ((A * B) + OPMODE[5])));
        repeat(2) @(negedge clk);
        $display ("-------------------------");

        $display ("END OF SIMULATION");
        $stop;
    end

    /* Self Checking Task */
    task self_checker(input [47:0] out,expected);
        begin
            if(expected != out) begin
                $display("Error: Expected %d, got %d", expected, out);
                $stop;
            end
            else
                $display("Self Checker: Passed");
        end
    endtask

    /* Monitor the inputs and outputs */
    /* initial begin
         $monitor("OPMODE = %b,A = %d,B = %d, C = %d, D = %d, P = %d",
                OPMODE, A, B, C, D, P);
    end */
endmodule
```

## 1.3 Do file

```
vlib work

vlog DSP48A1Module.v DSP48A1Module_tb.v

vsim -voptargs=+acc work.DSP48A1Module_tb

# add Internal signals
add wave DUT/*

run -all
```

## 1.4 Waveform

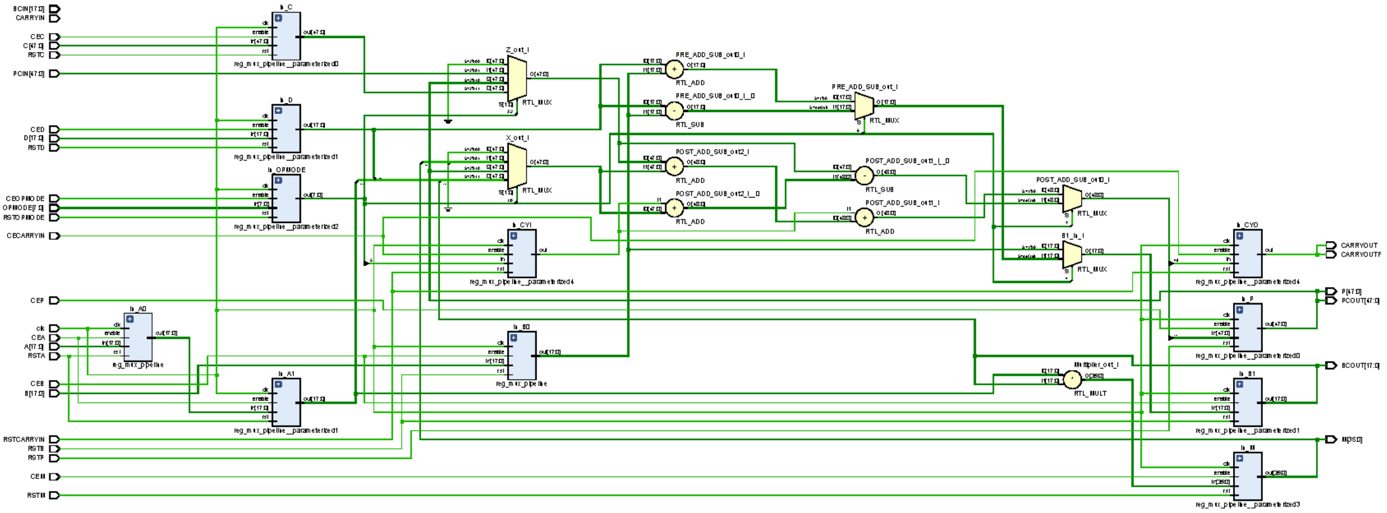

Wave
File Edit View Add Format Tools Bookmarks Window Help
Wave - Default
Msgs
A 18'd8
B 18'd2
D 18'd0
C 48'd50
CARRYIN 1'd1
M 36'd16
P 48'd33
CARRYOUT 1'd0
CARRYOUTF 1'd0
clk 1'd0
OPMODE 8'd173
CEA 1'd1
CEB 1'd1
CEC 1'd1
CED 1'd1
CEM 1'd1
CEP 1'd1
CEOPMODE 1'd1
CECARRYIN 1'd1
RSTA 1'd0
RSTB 1'd0
RSTC 1'd0
RSTD 1'd0
RSTM 1'd0
RSTP 1'd0
RSTOPMODE 1'd0
RSTCARRYIN 1'd0
BCIN 18'd0
BCOUT 18'd2
PCIN 48'd0
PCOUT 48'd33
B0_in 18'd2
A0_out 18'd8
B0_out 18'd2
D_out 18'd0
sim:/DSP48A1Module_tb/DUT/CEC @ 70 ns
1
Now 360 ns
0 ns to 378 ns


Wave
File Edit View Add Format Tools Bookmarks Window Help
Wave - Default
Msgs
CEP 1'd1
CEOPMODE 1'd1
CECARRYIN 1'd1
RSTA 1'd0
RSTB 1'd0
RSTC 1'd0
RSTD 1'd0
RSTM 1'd0
RSTP 1'd0
RSTOPMODE 1'd0
RSTCARRYIN 1'd0
BCIN 18'd0
BCOUT 18'd2
PCIN 48'd0
PCOUT 48'd33
B0_in 18'd2
A0_out 18'd8
B0_out 18'd2
D_out 18'd0
C_out 48'd50
OPMODE_out 8'd173
PRE_ADD_SUB_out 18'd2
B1_in 18'd2
A1_out 18'd8
B1_out 18'd2
Multiplier_out 36'd16
M_out 36'd16
X_out 48'd16
P_out 48'd33
Z_out 48'd50
CYI_in 1'd1
CYI_out 1'd1
POST_ADD_SUB_out 48'd33
CYO_in 1'd0
CYO_out 1'd0
Now 360 ns
0 ns to 378 ns
CYO_out

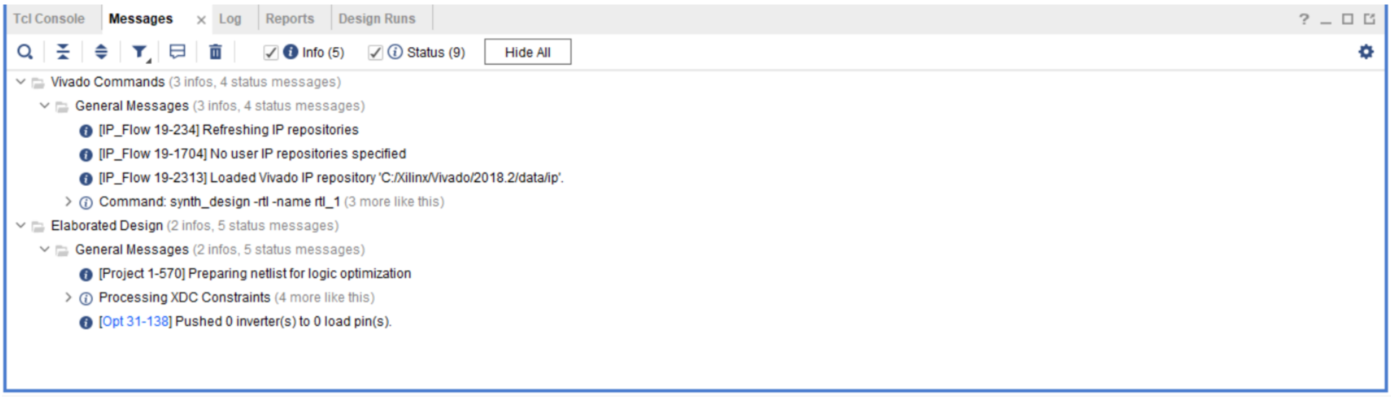
## 1.5 Constraint File

```
## Clock signal

# w5 PIN CONNECTED TO CLOCK  33 IS THE DEFNITION OF 3.3v PASSED TO PINS
set_property -dict {PACKAGE_PIN W5 IOSTANDARD LVCMOS33} [get_ports clk]
#add The name of clock in design after -name
create_clock -period 10.00 -name clk -waveform {0.000 5.000} -add [get_ports clk]
```

## 1.6 RTL snippets

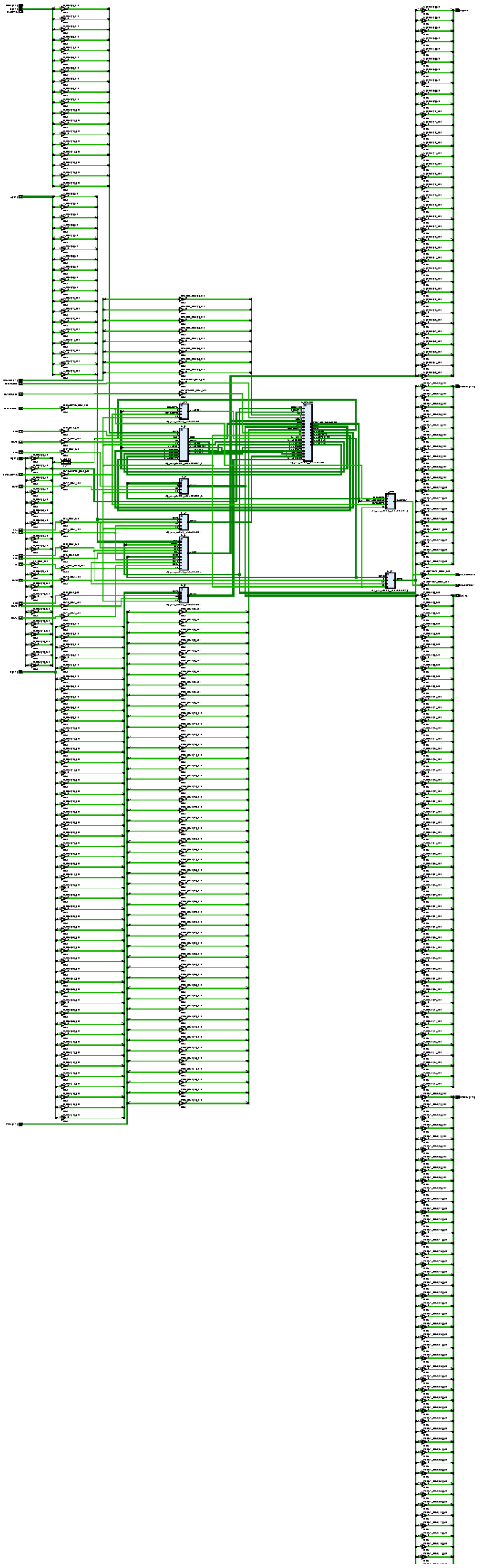


## 1.7 Elaboration messages



Vivado Commands (3 infos, 4 status messages)
General Messages (3 infos, 4 status messages)
[IP_Flow 19-234] Refreshing IP repositories
[IP_Flow 19-1704] No user IP repositories specified
[IP_Flow 19-2313] Loaded Vivado IP repository 'C:/Xilinx/Vivado/2018.2/data/ip'.
Command: synth_design -rtl -name rtl_1 (3 more like this)
Elaborated Design (2 infos, 5 status messages)
General Messages (2 infos, 5 status messages)
[Project 1-570] Preparing netlist for logic optimization
Processing XDC Constraints (4 more like this)
[Opt 31-138] Pushed 0 inverter(s) to 0 load pin(s).

## 1.8 Synthesis Schematic



## 1.9 Synthesis Messages

- Enable and rst of reg_mux_pipeline design for A0,B0 as they are connected to the input directly without creating a register due to parameters

```
    parameter A0REG = 0, parameter B0REG = 0,   // Default: NO registers
```

- CARRYIN are unconnected by design due to the parameter of { parameter CARRYINSEL = "OPMODE5"} once it changes to "CARRYIN", They will be connected to them.

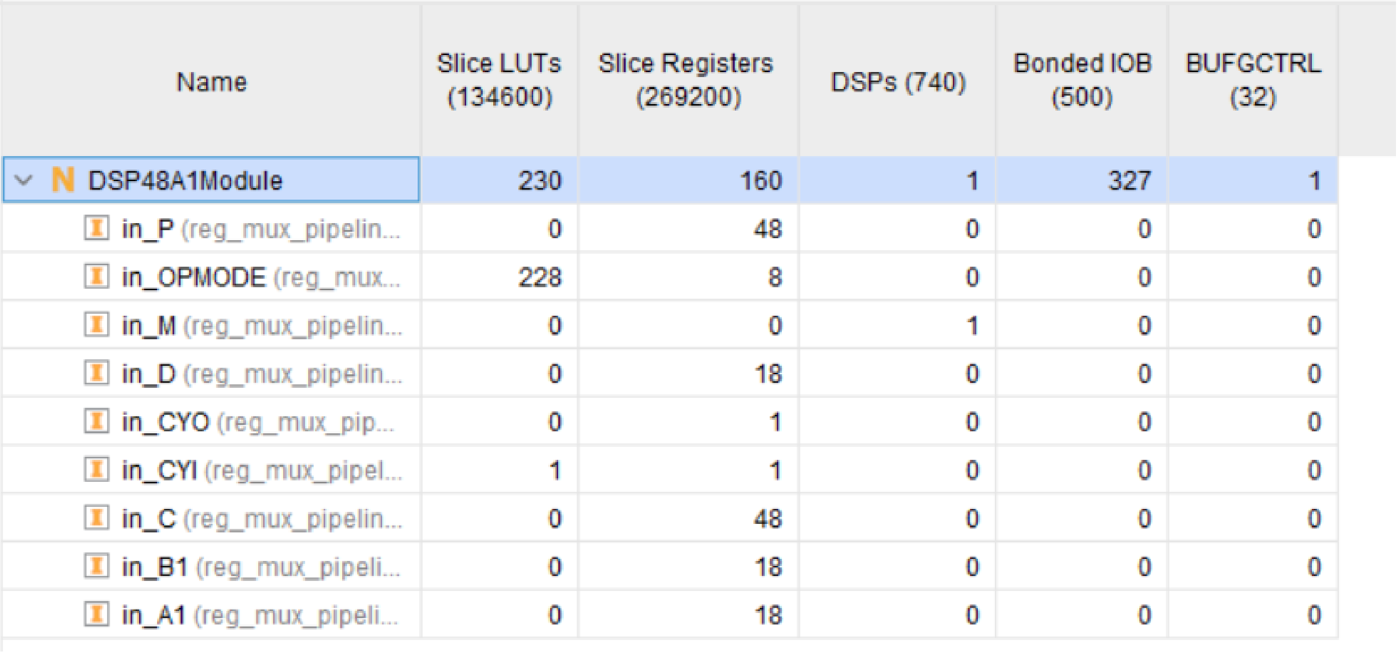
- BCIN are unconnected by design due to the parameter of { parameter B_INPUT = "DIRECT" } once it changes to "CASCADE", They will be connected to them.

| **Name** | **Severity** | **Details** |
|---|---|---|
| **Vivado_Tcl 4-321** | Status | Starting synth_design |
| **Common 17-347** | Status | Attempting to get a license for feature 'Synthesis' and/or device 'xc7a200t' |
| **Project 1-262** | Status | Processing XDC Constraints |
| **Project 1-569** | Status | Initializing timing engine |
| **Designutils 20-179** | Status | Parsing XDC File [F:/Electronics/Digital Electronics - KW/Mini_project_DSP48A1/DSP_CONSTRAINT_FILE.xdc] |
| **Designutils 20-178** | Status | Finished Parsing XDC File [F:/Electronics/Digital Electronics - KW/Mini_project_DSP48A1/DSP_CONSTRAINT_FILE.xdc] |
| **Project 1-263** | Status | Completed Processing XDC Constraints |
| **Vivado_Tcl 4-41** | Status | 29 Infos, 41 Warnings, 0 Critical Warnings and 0 Errors encountered. |
| **Vivado_Tcl 4-42** | Status | synth_design completed successfully |
| **#UNDEF** | Status | report_utilization: Time (s): cpu = 00:00:00 ; elapsed = 00:00:00.082 . Memory (MB): peak = 906.184 ; gain = 0.000 |
| **Common 17-349** | Information | Got license for feature 'Synthesis' and/or device 'xc7a200t' |
| **Synth 8-6157** | Information | synthesizing module 'reg_mux_pipeline' |
| **Synth 8-6157** | Information | synthesizing module 'reg_mux_pipeline__parameterized0' |
| **Synth 8-6157** | Information | synthesizing module 'reg_mux_pipeline__parameterized1' |
| **Synth 8-6157** | Information | synthesizing module 'reg_mux_pipeline__parameterized2' |
| **Synth 8-6157** | Information | synthesizing module 'reg_mux_pipeline__parameterized3' |
| **Synth 8-6157** | Information | synthesizing module 'reg_mux_pipeline__parameterized4' |
| **Synth 8-6155** | Information | done synthesizing module 'reg_mux_pipeline__parameterized0' (1#1) |
| **Synth 8-6155** | Information | done synthesizing module 'reg_mux_pipeline__parameterized1' (1#1) |
| **Synth 8-6155** | Information | done synthesizing module 'reg_mux_pipeline__parameterized2' (1#1) |
| **Synth 8-6155** | Information | done synthesizing module 'reg_mux_pipeline__parameterized3' (1#1) |
| **Synth 8-6155** | Information | done synthesizing module 'reg_mux_pipeline__parameterized4' (1#1) |
| **Synth 8-6155** | Information | done synthesizing module 'DSP48A1Module' (2#1) |
| **Synth 8-3331** | Warning | design reg_mux_pipeline has unconnected port enable |
| **Synth 8-3331** | Warning | design reg_mux_pipeline has unconnected port rst |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port CARRYIN |

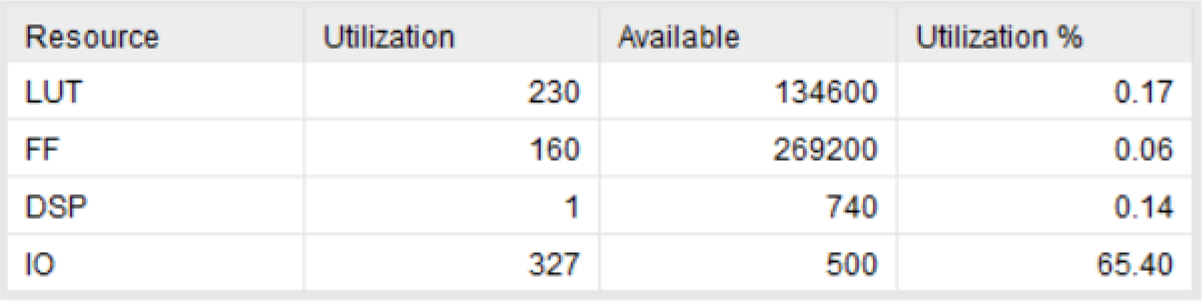
| | | |
|---|---|---|
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[17] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[16] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[15] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[14] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[13] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[12] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[11] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[10] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[9] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[8] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[7] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[6] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[5] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[4] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[3] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[2] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[1] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[0] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port CARRYIN |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[17] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[16] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[15] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[14] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[13] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[12] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[11] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[10] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[9] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[8] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[7] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[6] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[5] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[4] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[3] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[2] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[1] |
| **Synth 8-3331** | Warning | design DSP48A1Module has unconnected port BCIN[0] |
| **Device 21-403** | Information | Loading part xc7a200tffg1156-3 |
| **Project 1-236** | Information | Implementation specific constraints were found while reading constraint file [F:/Electronics/Digital Electronics -KW/Mini_project_DSP48A1/DSP_CONSTRAINT_FILE.xdc]. These constraints will be ignored for synthesis but will be used in implementation. Impacted constraints are listed in the file [.Xil/DSP48A1Module_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** | Information | HDL ADVISOR - The operator resource <adder> is shared. To prevent sharing consider applying a KEEP on the output of the operator |
| **Synth 8-5842** | Information | Cannot pack DSP OPMODE registers because of constant '1' value. Packing the registers will cause simulation mismatch at initial cycle |
| **Project 1-571** | Information | Translating synthesized netlist |
| **Netlist 29-17** | Information | Analyzing 207 Unisim elements for replacement |
| **Netlist 29-28** | Information | Unisim Transformation completed in 0 CPU seconds |
| **Project 1-570** | Information | Preparing netlist for logic optimization |
| **Opt 31-138** | Information | Pushed 0 inverter(s) to 0 load pin(s). |
| **Project 1-111** | Information | Unisim Transformation Summary:<br>No Unisim elements were transformed. |
| **Common 17-83** | Information | Releasing license: Synthesis |
| **Constraints 18-5210** | Warning | No constraint will be written out. |
| **Common 17-1381** | Information | The checkpoint 'F:/Electronics/Digital Electronics -KW/Mini_project_DSP48A1/DSP48A1/DSP48A1.runs/synth_1/DSP48A1Module.dcp' has been generated. |
| **runtcl-4** | Information | Executing : report_utilization -file DSP48A1Module_utilization_synth.rpt -pb DSP48A1Module_utilization_synth.pb |
| **Common 17-206** | Information | Exiting Vivado at Sun Jul 28 02:59:45 2024... |

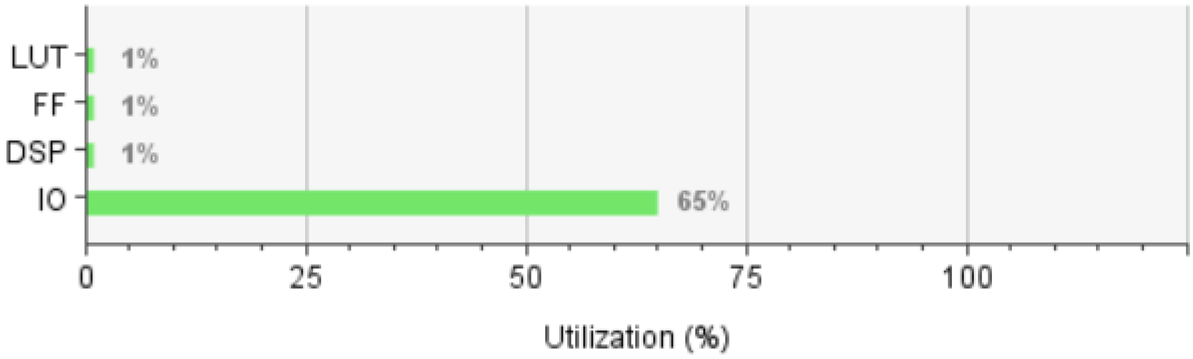
## 1.10 Synthesis Utilization report



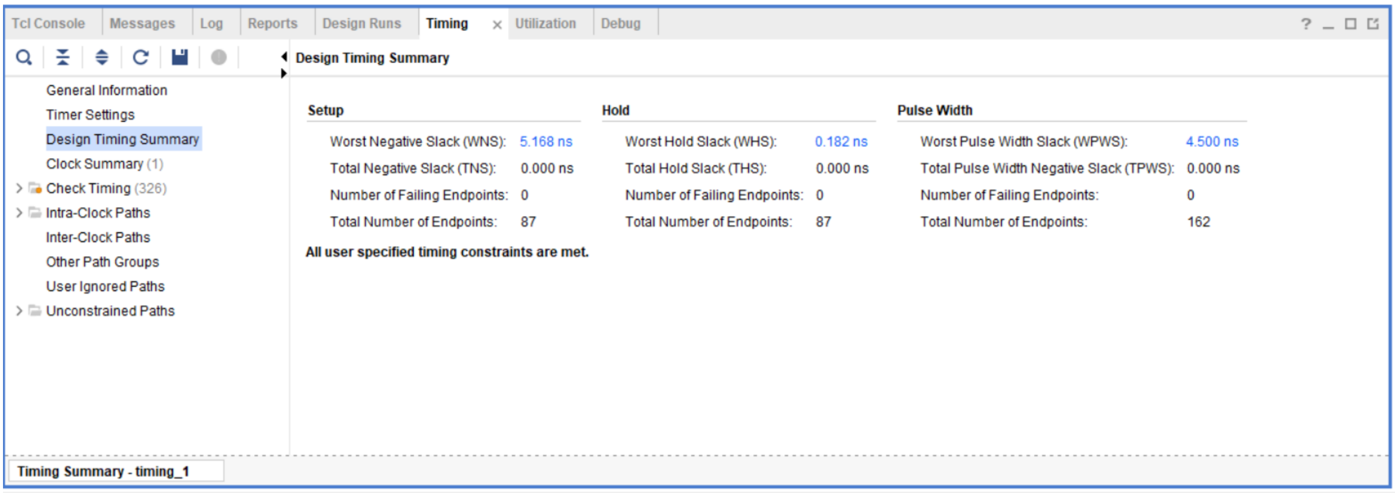
| Name | Slice LUTs (134600) | Slice Registers (269200) | DSPs (740) | Bonded IOB (500) | BUFGCTRL (32) |
|---|---|---|---|---|---|
| DSP48A1Module | 230 | 160 | 1 | 327 | 1 |
| in_P (reg_mux_pipelin... | 0 | 48 | 0 | 0 | 0 |
| in_OPMODE (reg_mux... | 228 | 8 | 0 | 0 | 0 |
| in_M (reg_mux_pipelin... | 0 | 0 | 1 | 0 | 0 |
| in_D (reg_mux_pipelin... | 0 | 18 | 0 | 0 | 0 |
| in_CYO (reg_mux_pip... | 0 | 1 | 0 | 0 | 0 |
| in_CYI (reg_mux_pipel... | 1 | 1 | 0 | 0 | 0 |
| in_C (reg_mux_pipelin... | 0 | 48 | 0 | 0 | 0 |
| in_B1 (reg_mux_pipeli... | 0 | 18 | 0 | 0 | 0 |
| in_A1 (reg_mux_pipeli... | 0 | 18 | 0 | 0 | 0 |



| Resource | Utilization | Available | Utilization % |
|---|---|---|---|
| LUT | 230 | 134600 | 0.17 |
| FF | 160 | 269200 | 0.06 |
| DSP | 1 | 740 | 0.14 |
| IO | 327 | 500 | 65.40 |



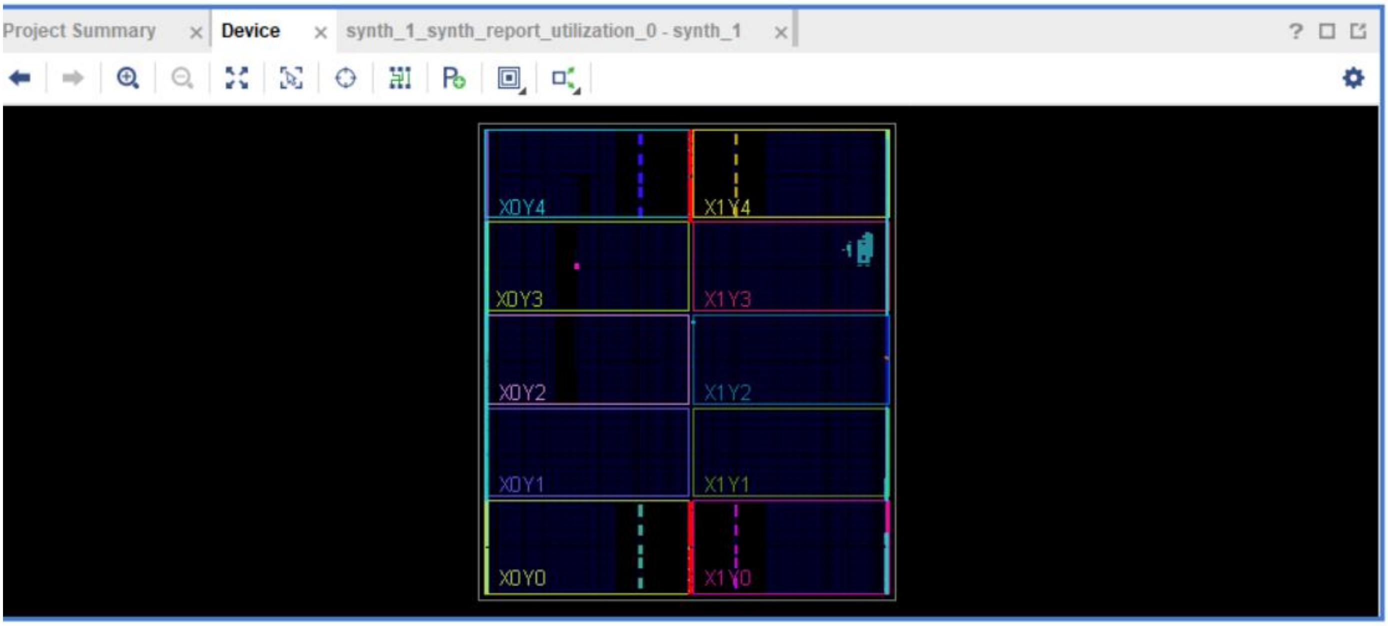
## 1.11 Synthesis Timing Report



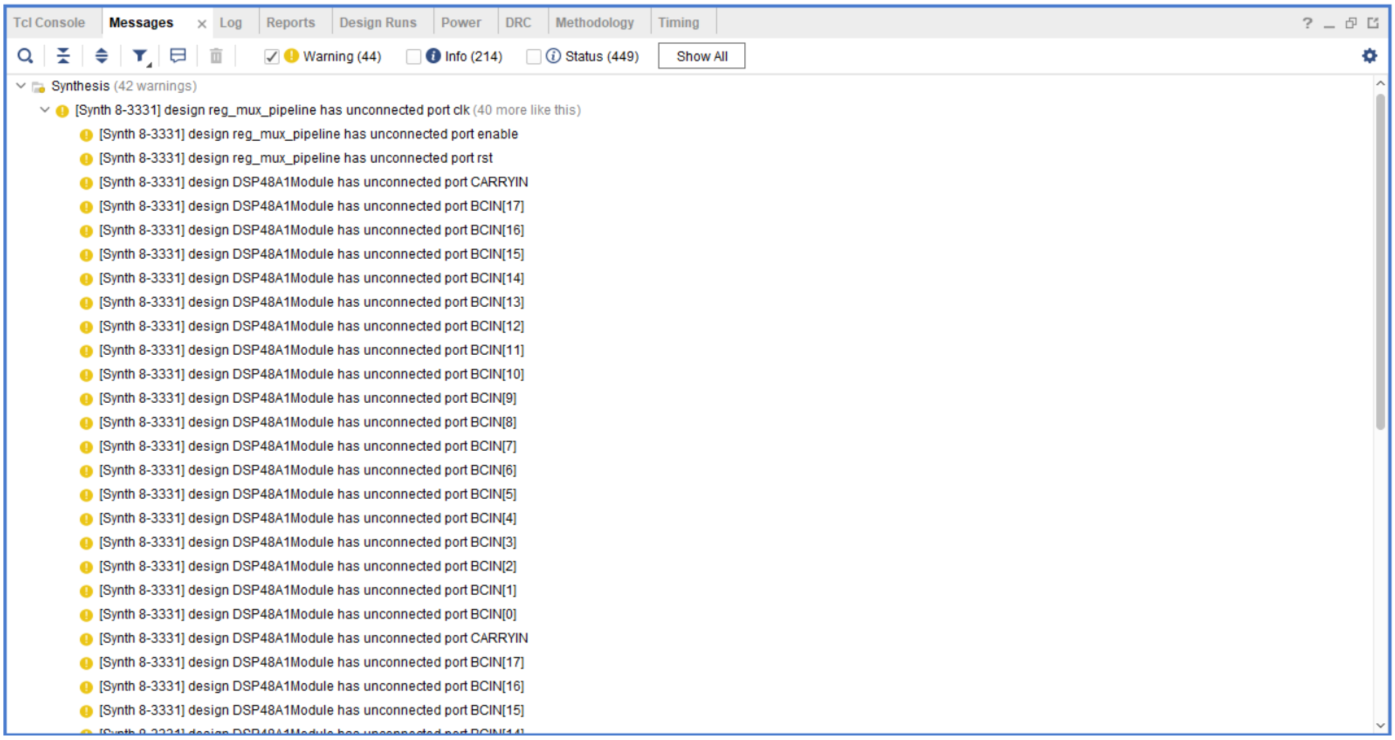
| Setup | | Hold | | Pulse Width | |
|---|---|---|---|---|---|
| Worst Negative Slack (WNS): | 5.168 ns | Worst Hold Slack (WHS): | 0.182 ns | Worst Pulse Width Slack (WPWS): | 4.500 ns |
| Total Negative Slack (TNS): | 0.000 ns | Total Hold Slack (THS): | 0.000 ns | Total Pulse Width Negative Slack (TPWS): | 0.000 ns |
| Number of Failing Endpoints: | 0 | Number of Failing Endpoints: | 0 | Number of Failing Endpoints: | 0 |
| Total Number of Endpoints: | 87 | Total Number of Endpoints: | 87 | Total Number of Endpoints: | 162 |

All user specified timing constraints are met.

## 1.12 Implementation Schematic

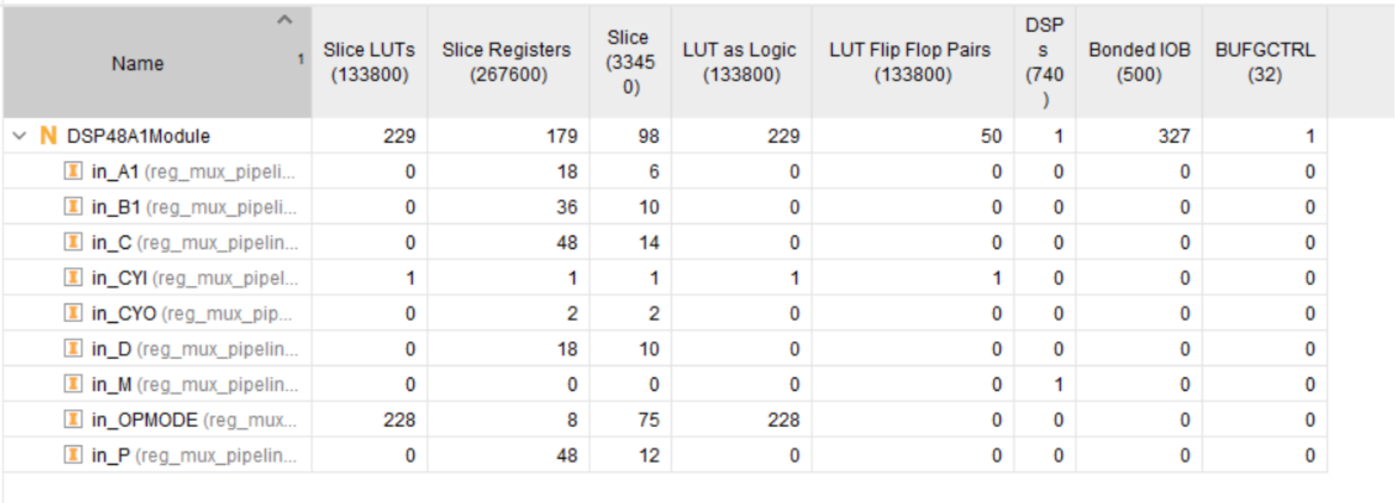


## 1.13 Implementation Messages

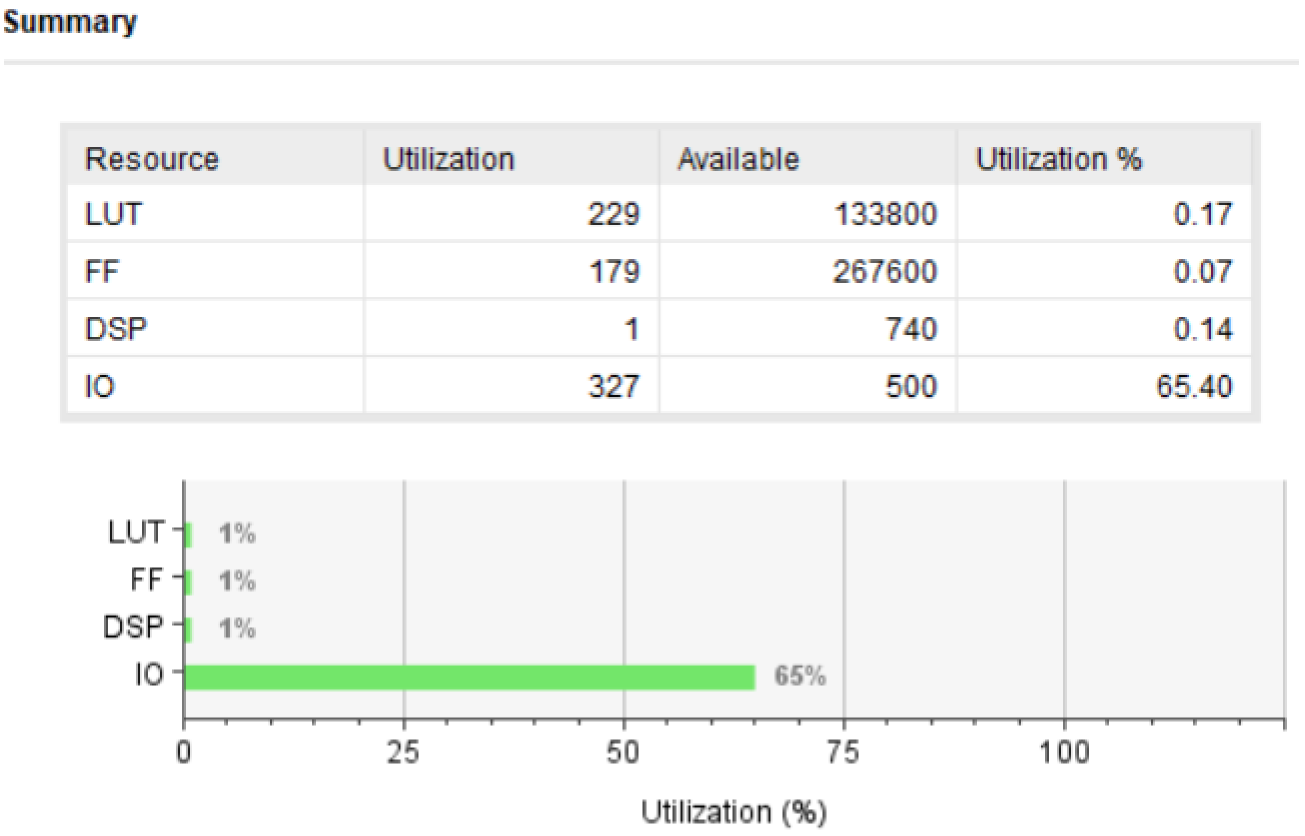


```
Synthesis (42 warnings)
  [Synth 8-3331] design reg_mux_pipeline has unconnected port clk (40 more like this)
    [Synth 8-3331] design reg_mux_pipeline has unconnected port enable
    [Synth 8-3331] design reg_mux_pipeline has unconnected port rst
    [Synth 8-3331] design DSP48A1Module has unconnected port CARRYIN
    [Synth 8-3331] design DSP48A1Module has unconnected port BCIN[17]
    [Synth 8-3331] design DSP48A1Module has unconnected port BCIN[16]
    [Synth 8-3331] design DSP48A1Module has unconnected port BCIN[15]
    [Synth 8-3331] design DSP48A1Module has unconnected port BCIN[14]
    [Synth 8-3331] design DSP48A1Module has unconnected port BCIN[13]
    [Synth 8-3331] design DSP48A1Module has unconnected port BCIN[12]
    [Synth 8-3331] design DSP48A1Module has unconnected port BCIN[11]
    [Synth 8-3331] design DSP48A1Module has unconnected port BCIN[10]
    [Synth 8-3331] design DSP48A1Module has unconnected port BCIN[9]
    [Synth 8-3331] design DSP48A1Module has unconnected port BCIN[8]
    [Synth 8-3331] design DSP48A1Module has unconnected port BCIN[7]
    [Synth 8-3331] design DSP48A1Module has unconnected port BCIN[6]
    [Synth 8-3331] design DSP48A1Module has unconnected port BCIN[5]
    [Synth 8-3331] design DSP48A1Module has unconnected port BCIN[4]
    [Synth 8-3331] design DSP48A1Module has unconnected port BCIN[3]
    [Synth 8-3331] design DSP48A1Module has unconnected port BCIN[2]
    [Synth 8-3331] design DSP48A1Module has unconnected port BCIN[1]
    [Synth 8-3331] design DSP48A1Module has unconnected port BCIN[0]
    [Synth 8-3331] design DSP48A1Module has unconnected port CARRYIN
    [Synth 8-3331] design DSP48A1Module has unconnected port BCIN[17]
    [Synth 8-3331] design DSP48A1Module has unconnected port BCIN[16]
    [Synth 8-3331] design DSP48A1Module has unconnected port BCIN[15]
    [Synth 8-3331] design DSP48A1Module has unconnected port BCIN[14]
```

## 1.14 Implementation Utilization Report



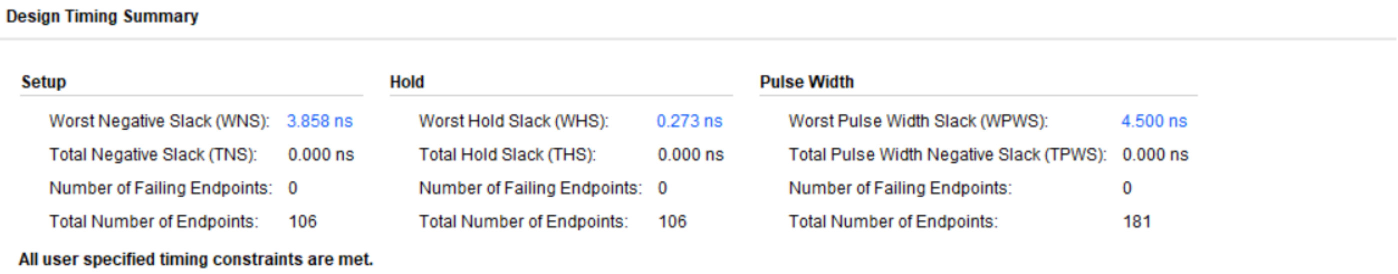
| Name | Slice LUTs (133800) | Slice Registers (267600) | Slice (33450) | LUT as Logic (133800) | LUT Flip Flop Pairs (133800) | DSPs (740) | Bonded IOB (500) | BUFGCTRL (32) |
|---|---|---|---|---|---|---|---|---|
| N DSP48A1Module | 229 | 179 | 98 | 229 | 50 | 1 | 327 | 1 |
| in_A1 (reg_mux_pipeli... | 0 | 18 | 6 | 0 | 0 | 0 | 0 | 0 |
| in_B1 (reg_mux_pipeli... | 0 | 36 | 10 | 0 | 0 | 0 | 0 | 0 |
| in_C (reg_mux_pipelin... | 0 | 48 | 14 | 0 | 0 | 0 | 0 | 0 |
| in_CYI (reg_mux_pipel... | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| in_CYO (reg_mux_pip... | 0 | 2 | 2 | 0 | 0 | 0 | 0 | 0 |
| in_D (reg_mux_pipelin... | 0 | 18 | 10 | 0 | 0 | 0 | 0 | 0 |
| in_M (reg_mux_pipelin... | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| in_OPMODE (reg_mux... | 228 | 8 | 75 | 228 | 0 | 0 | 0 | 0 |
| in_P (reg_mux_pipelin... | 0 | 48 | 12 | 0 | 0 | 0 | 0 | 0 |



### Summary

| Resource | Utilization | Available | Utilization % |
|---|---|---|---|
| LUT | 229 | 133800 | 0.17 |
| FF | 179 | 267600 | 0.07 |
| DSP | 1 | 740 | 0.14 |
| IO | 327 | 500 | 65.40 |

## 1.15 Implementation Timing Report



**Design Timing Summary**

**Setup**

| | |
|---|---|
| Worst Negative Slack (WNS): | 3.858 ns |
| Total Negative Slack (TNS): | 0.000 ns |
| Number of Failing Endpoints: | 0 |
| Total Number of Endpoints: | 106 |

**Hold**

| | |
|---|---|
| Worst Hold Slack (WHS): | 0.273 ns |
| Total Hold Slack (THS): | 0.000 ns |
| Number of Failing Endpoints: | 0 |
| Total Number of Endpoints: | 106 |

**Pulse Width**

| | |
|---|---|
| Worst Pulse Width Slack (WPWS): | 4.500 ns |
| Total Pulse Width Negative Slack (TPWS): | 0.000 ns |
| Number of Failing Endpoints: | 0 |
| Total Number of Endpoints: | 181 |

**All user specified timing constraints are met.**