Project 1

DSP48A1

Submitted by:

1- Abdelrahman Maher Hassan

2- Zyad Mohammed Abd El-Halim

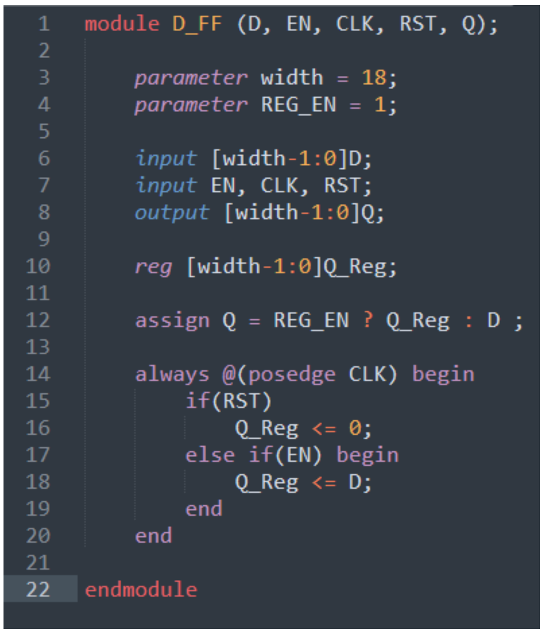
3- Peter Magdy

Table of Contents

List of Figures

1. RTL Code

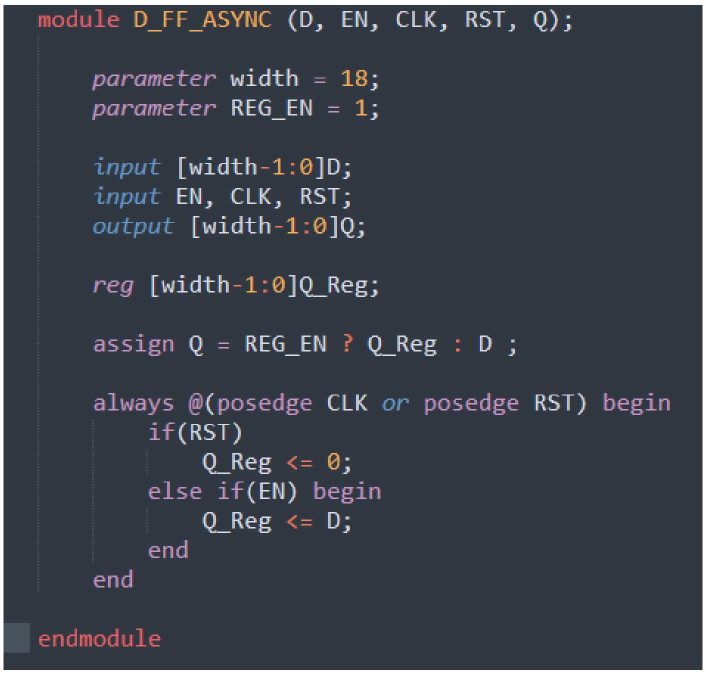
1.1. D Flip flop Synchronous



```verilog
module D_FF (D, EN, CLK, RST, Q);

    parameter width = 18;
    parameter REG_EN = 1;

    input [width-1:0]D;
    input EN, CLK, RST;
    output [width-1:0]Q;

    reg [width-1:0]Q_Reg;

    assign Q = REG_EN ? Q_Reg : D ;

    always @(posedge CLK) begin
        if(RST)
            Q_Reg <= 0;
        else if(EN) begin
            Q_Reg <= D;
        end
    end

endmodule
```

Figure 1: D Flipflop Sync with a mux.

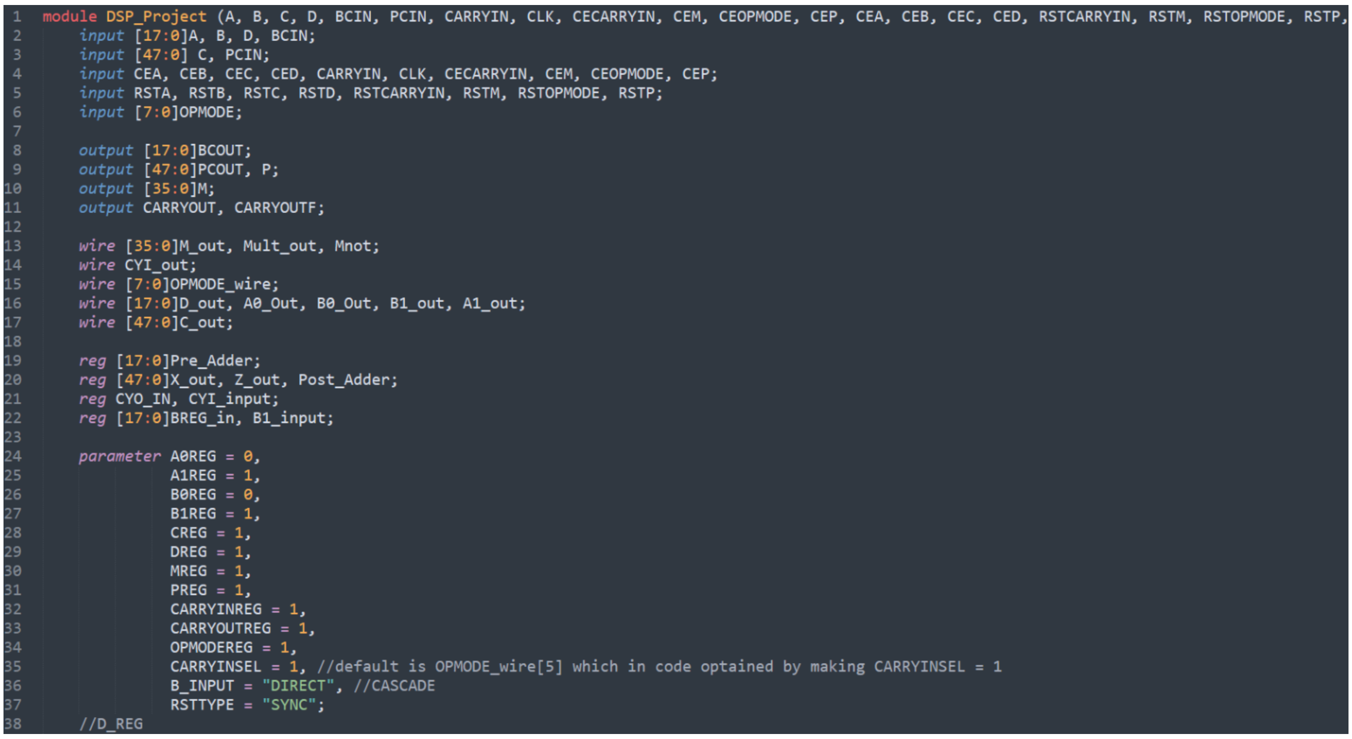
1.2. D Flip flop Asynchronous



```verilog
module D_FF_ASYNC (D, EN, CLK, RST, Q);

    parameter width = 18;
    parameter REG_EN = 1;

    input [width-1:0]D;
    input EN, CLK, RST;
    output [width-1:0]Q;

    reg [width-1:0]Q_Reg;

    assign Q = REG_EN ? Q_Reg : D ;

    always @(posedge CLK or posedge RST) begin
        if(RST)
            Q_Reg <= 0;
        else if(EN) begin
            Q_Reg <= D;
        end
    end

endmodule
```

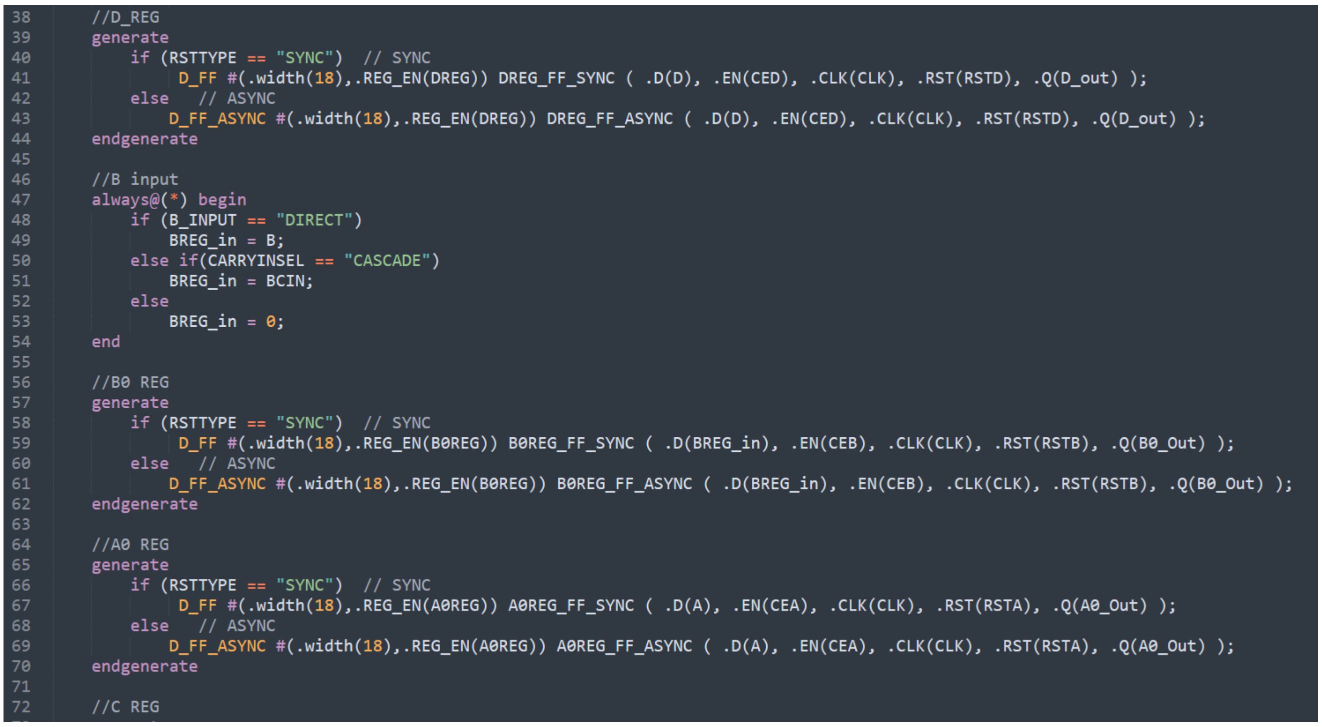
Figure 2: D Flipflop Async with a mux.

1.3. DSP Code



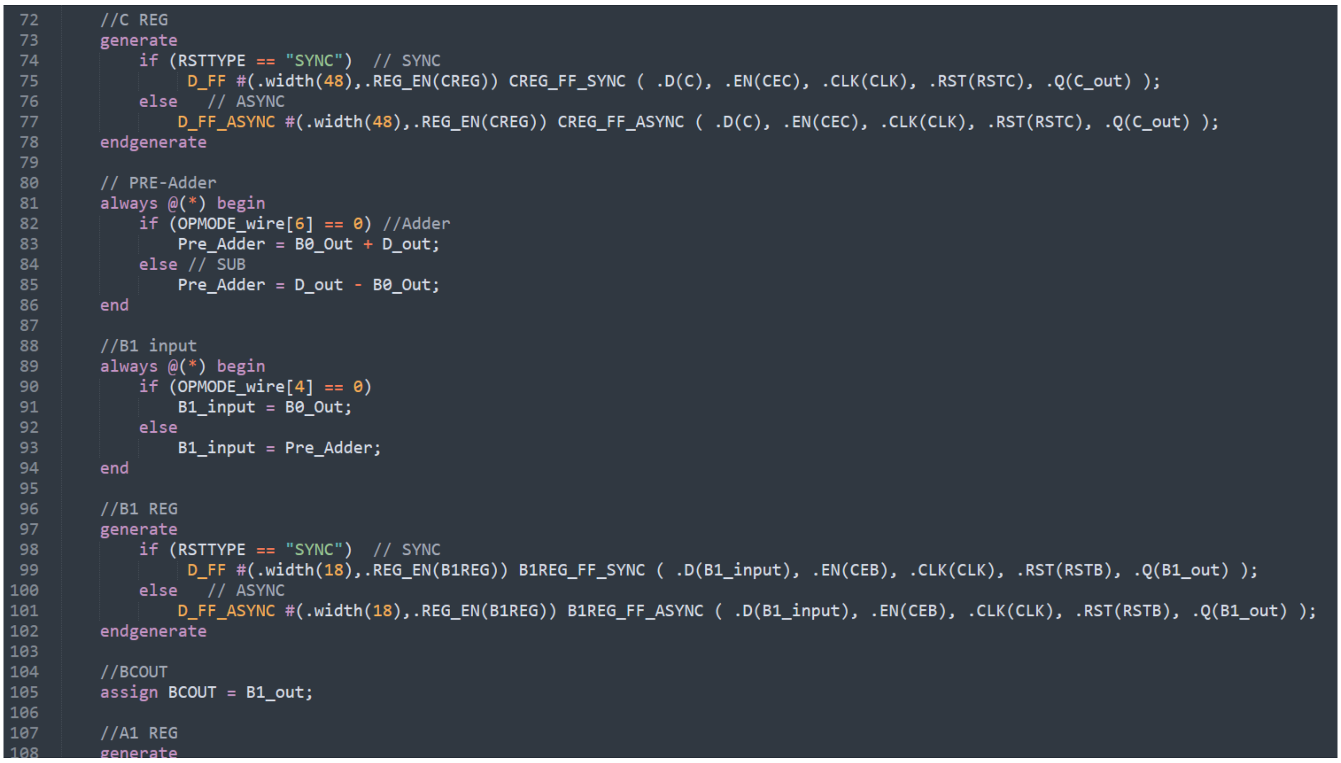
```
module DSP_Project (A, B, C, D, BCIN, PCIN, CARRYIN, CLK, CECARRYIN, CEM, CEOPMODE, CEP, CEA, CEB, CEC, CED, RSTCARRYIN, RSTM, RSTOPMODE, RSTP,
    input [17:0]A, B, D, BCIN;
    input [47:0] C, PCIN;
    input CEA, CEB, CEC, CED, CARRYIN, CLK, CECARRYIN, CEM, CEOPMODE, CEP;
    input RSTA, RSTB, RSTC, RSTD, RSTCARRYIN, RSTM, RSTOPMODE, RSTP;
    input [7:0]OPMODE;

    output [17:0]BCOUT;
    output [47:0]PCOUT, P;
    output [35:0]M;
    output CARRYOUT, CARRYOUTF;

    wire [35:0]M_out, Mult_out, Mnot;
    wire CYI_out;
    wire [7:0]OPMODE_wire;
    wire [17:0]D_out, A0_Out, B0_Out, B1_out, A1_out;
    wire [47:0]C_out;

    reg [17:0]Pre_Adder;
    reg [47:0]X_out, Z_out, Post_Adder;
    reg CYO_IN, CYI_input;
    reg [17:0]BREG_in, B1_input;

    parameter A0REG = 0,
              A1REG = 1,
              B0REG = 0,
              B1REG = 1,
              CREG = 1,
              DREG = 1,
              MREG = 1,
              PREG = 1,
              CARRYINREG = 1,
              CARRYOUTREG = 1,
              OPMODEREG = 1,
              CARRYINSEL = 1, //default is OPMODE_wire[5] which in code optained by making CARRYINSEL = 1
              B_INPUT = "DIRECT", //CASCADE
              RSTTYPE = "SYNC";
    //D_REG
```

Figure 3: DSP Code part 1.



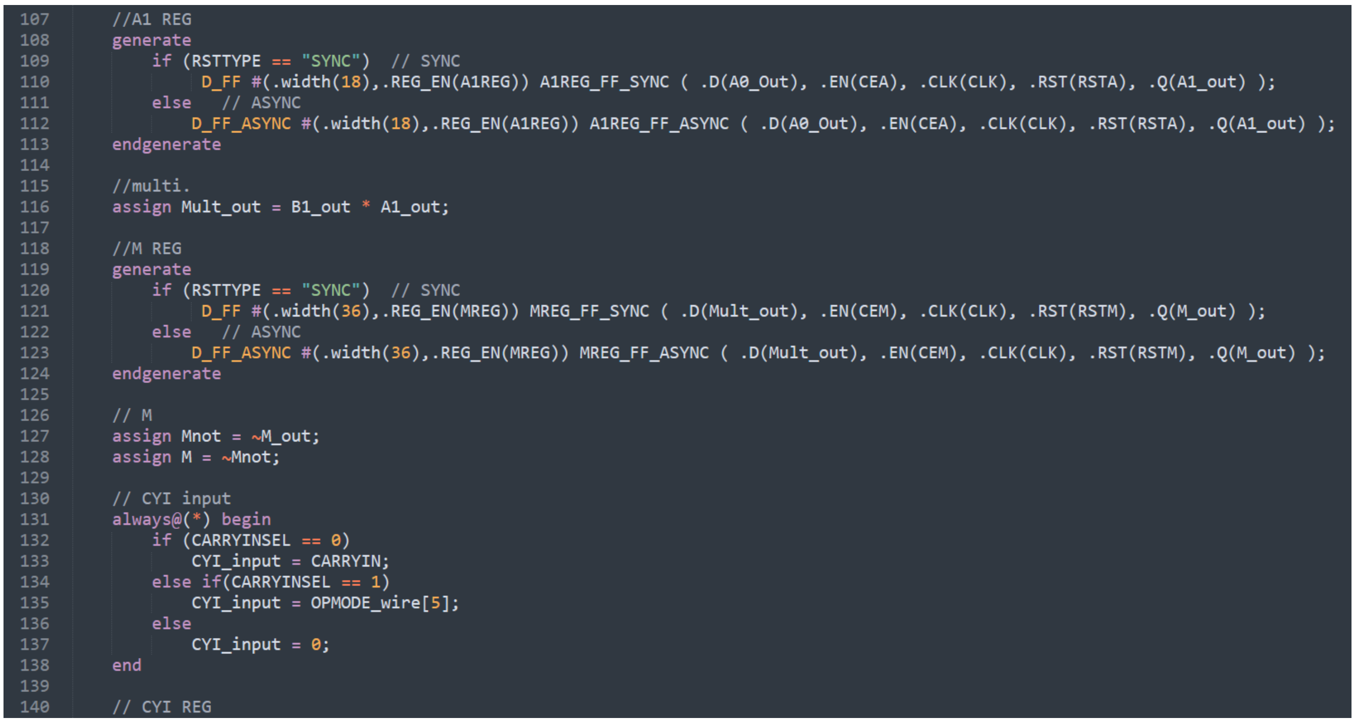
```
    //D_REG
    generate
        if (RSTTYPE == "SYNC")  // SYNC
             D_FF #(.width(18),.REG_EN(DREG)) DREG_FF_SYNC ( .D(D), .EN(CED), .CLK(CLK), .RST(RSTD), .Q(D_out) );
        else   // ASYNC
            D_FF_ASYNC #(.width(18),.REG_EN(DREG)) DREG_FF_ASYNC ( .D(D), .EN(CED), .CLK(CLK), .RST(RSTD), .Q(D_out) );
    endgenerate

    //B input
    always@(*) begin
        if (B_INPUT == "DIRECT")
            BREG_in = B;
        else if(CARRYINSEL == "CASCADE")
            BREG_in = BCIN;
        else
            BREG_in = 0;
    end

    //B0 REG
    generate
        if (RSTTYPE == "SYNC")  // SYNC
             D_FF #(.width(18),.REG_EN(B0REG)) B0REG_FF_SYNC ( .D(BREG_in), .EN(CEB), .CLK(CLK), .RST(RSTB), .Q(B0_Out) );
        else   // ASYNC
            D_FF_ASYNC #(.width(18),.REG_EN(B0REG)) B0REG_FF_ASYNC ( .D(BREG_in), .EN(CEB), .CLK(CLK), .RST(RSTB), .Q(B0_Out) );
    endgenerate

    //A0 REG
    generate
        if (RSTTYPE == "SYNC")  // SYNC
             D_FF #(.width(18),.REG_EN(A0REG)) A0REG_FF_SYNC ( .D(A), .EN(CEA), .CLK(CLK), .RST(RSTA), .Q(A0_Out) );
        else   // ASYNC
            D_FF_ASYNC #(.width(18),.REG_EN(A0REG)) A0REG_FF_ASYNC ( .D(A), .EN(CEA), .CLK(CLK), .RST(RSTA), .Q(A0_Out) );
    endgenerate

    //C REG
```

Figure 4: DSP Code part 2.



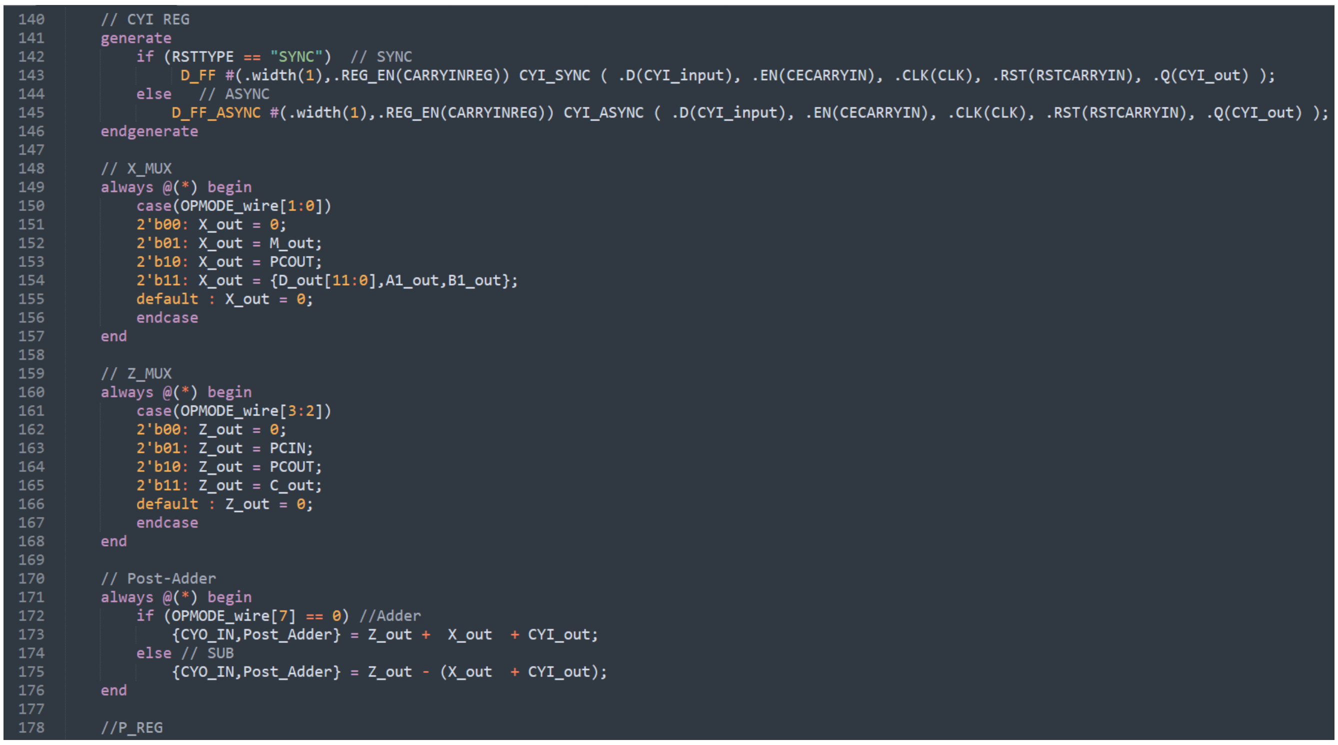
```verilog
    //C REG
    generate
        if (RSTTYPE == "SYNC")  // SYNC
             D_FF #(.width(48),.REG_EN(CREG)) CREG_FF_SYNC ( .D(C), .EN(CEC), .CLK(CLK), .RST(RSTC), .Q(C_out) );
        else   // ASYNC
            D_FF_ASYNC #(.width(48),.REG_EN(CREG)) CREG_FF_ASYNC ( .D(C), .EN(CEC), .CLK(CLK), .RST(RSTC), .Q(C_out) );
    endgenerate

    // PRE-Adder
    always @(*) begin
        if (OPMODE_wire[6] == 0) //Adder
            Pre_Adder = B0_Out + D_out;
        else // SUB
            Pre_Adder = D_out - B0_Out;
    end

    //B1 input
    always @(*) begin
        if (OPMODE_wire[4] == 0)
            B1_input = B0_Out;
        else
            B1_input = Pre_Adder;
    end

    //B1 REG
    generate
        if (RSTTYPE == "SYNC")  // SYNC
             D_FF #(.width(18),.REG_EN(B1REG)) B1REG_FF_SYNC ( .D(B1_input), .EN(CEB), .CLK(CLK), .RST(RSTB), .Q(B1_out) );
        else   // ASYNC
            D_FF_ASYNC #(.width(18),.REG_EN(B1REG)) B1REG_FF_ASYNC ( .D(B1_input), .EN(CEB), .CLK(CLK), .RST(RSTB), .Q(B1_out) );
    endgenerate

    //BCOUT
    assign BCOUT = B1_out;

    //A1 REG
    generate
```

Figure 5: DSP Code part 3.



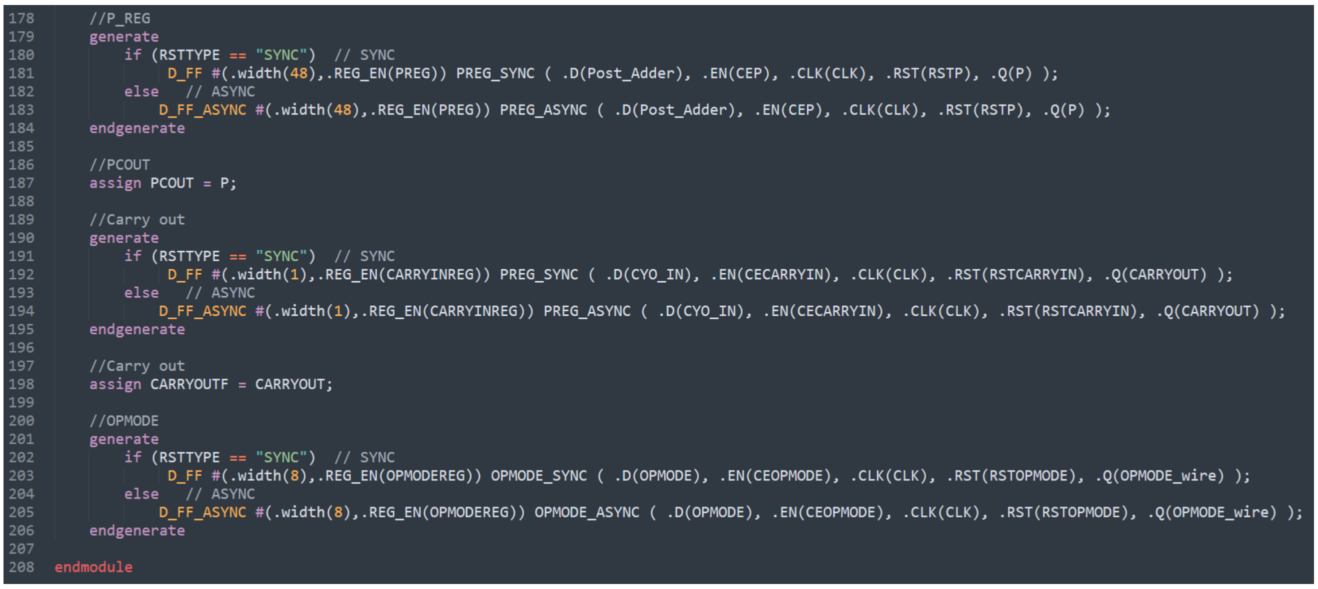
```verilog
    //A1 REG
    generate
        if (RSTTYPE == "SYNC")  // SYNC
             D_FF #(.width(18),.REG_EN(A1REG)) A1REG_FF_SYNC ( .D(A0_Out), .EN(CEA), .CLK(CLK), .RST(RSTA), .Q(A1_out) );
        else   // ASYNC
            D_FF_ASYNC #(.width(18),.REG_EN(A1REG)) A1REG_FF_ASYNC ( .D(A0_Out), .EN(CEA), .CLK(CLK), .RST(RSTA), .Q(A1_out) );
    endgenerate

    //multi.
    assign Mult_out = B1_out * A1_out;

    //M REG
    generate
        if (RSTTYPE == "SYNC")  // SYNC
             D_FF #(.width(36),.REG_EN(MREG)) MREG_FF_SYNC ( .D(Mult_out), .EN(CEM), .CLK(CLK), .RST(RSTM), .Q(M_out) );
        else   // ASYNC
            D_FF_ASYNC #(.width(36),.REG_EN(MREG)) MREG_FF_ASYNC ( .D(Mult_out), .EN(CEM), .CLK(CLK), .RST(RSTM), .Q(M_out) );
    endgenerate

    // M
    assign Mnot = ~M_out;
    assign M = ~Mnot;

    // CYI input
    always@(*) begin
        if (CARRYINSEL == 0)
            CYI_input = CARRYIN;
        else if(CARRYINSEL == 1)
            CYI_input = OPMODE_wire[5];
        else
            CYI_input = 0;
    end

    // CYI REG
```

Figure 6: DSP Code part 4.



```verilog
    // CYI REG
    generate
        if (RSTTYPE == "SYNC")  // SYNC
             D_FF #(.width(1),.REG_EN(CARRYINREG)) CYI_SYNC ( .D(CYI_input), .EN(CECARRYIN), .CLK(CLK), .RST(RSTCARRYIN), .Q(CYI_out) );
        else   // ASYNC
            D_FF_ASYNC #(.width(1),.REG_EN(CARRYINREG)) CYI_ASYNC ( .D(CYI_input), .EN(CECARRYIN), .CLK(CLK), .RST(RSTCARRYIN), .Q(CYI_out) );
    endgenerate

    // X_MUX
    always @(*) begin
        case(OPMODE_wire[1:0])
        2'b00: X_out = 0;
        2'b01: X_out = M_out;
        2'b10: X_out = PCOUT;
        2'b11: X_out = {D_out[11:0],A1_out,B1_out};
        default : X_out = 0;
        endcase
    end

    // Z_MUX
    always @(*) begin
        case(OPMODE_wire[3:2])
        2'b00: Z_out = 0;
        2'b01: Z_out = PCIN;
        2'b10: Z_out = PCOUT;
        2'b11: Z_out = C_out;
        default : Z_out = 0;
        endcase
    end

    // Post-Adder
    always @(*) begin
        if (OPMODE_wire[7] == 0) //Adder
            {CYO_IN,Post_Adder} = Z_out +  X_out  + CYI_out;
        else // SUB
            {CYO_IN,Post_Adder} = Z_out - (X_out  + CYI_out);
    end

    //P_REG
```

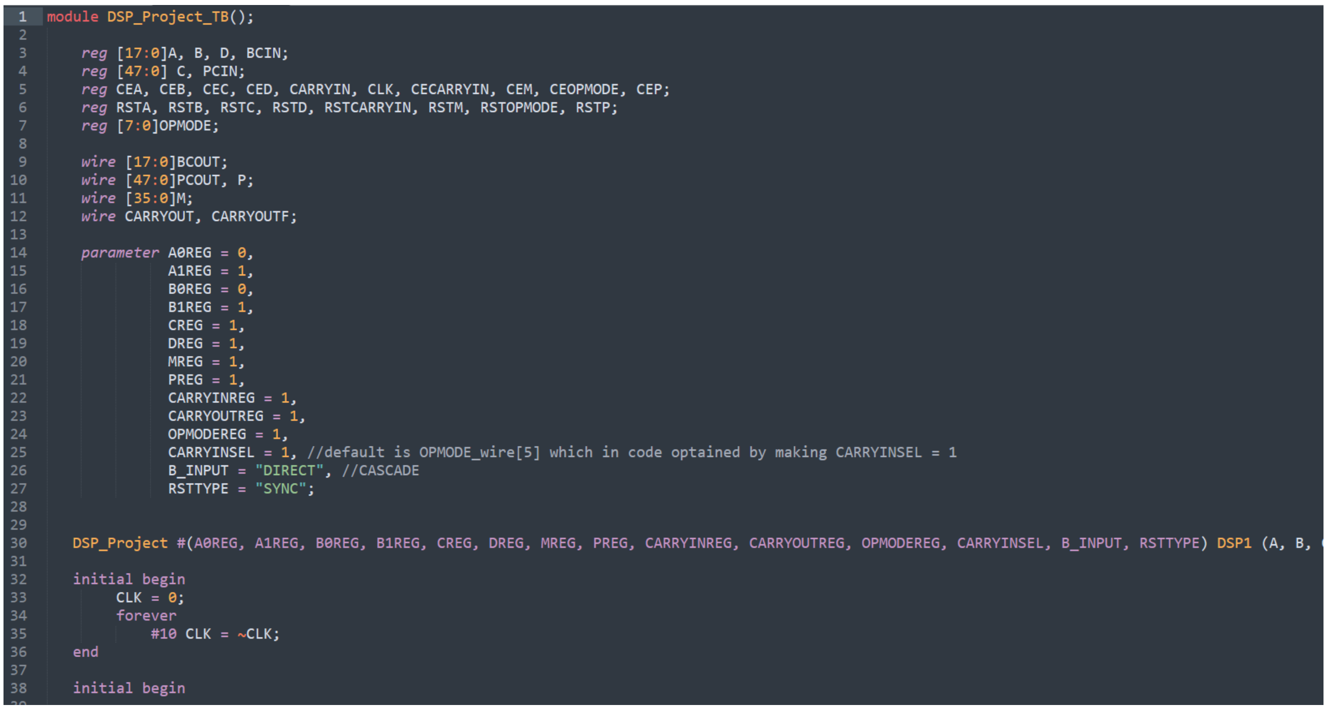
Figure 7: DSP Code part 5.



```verilog
    //P_REG
    generate
        if (RSTTYPE == "SYNC")  // SYNC
             D_FF #(.width(48),.REG_EN(PREG)) PREG_SYNC ( .D(Post_Adder), .EN(CEP), .CLK(CLK), .RST(RSTP), .Q(P) );
        else   // ASYNC
            D_FF_ASYNC #(.width(48),.REG_EN(PREG)) PREG_ASYNC ( .D(Post_Adder), .EN(CEP), .CLK(CLK), .RST(RSTP), .Q(P) );
    endgenerate

    //PCOUT
    assign PCOUT = P;

    //Carry out
    generate
        if (RSTTYPE == "SYNC")  // SYNC
             D_FF #(.width(1),.REG_EN(CARRYINREG)) PREG_SYNC ( .D(CYO_IN), .EN(CECARRYIN), .CLK(CLK), .RST(RSTCARRYIN), .Q(CARRYOUT) );
        else   // ASYNC
            D_FF_ASYNC #(.width(1),.REG_EN(CARRYINREG)) PREG_ASYNC ( .D(CYO_IN), .EN(CECARRYIN), .CLK(CLK), .RST(RSTCARRYIN), .Q(CARRYOUT) );
    endgenerate

    //Carry out
    assign CARRYOUTF = CARRYOUT;

    //OPMODE
    generate
        if (RSTTYPE == "SYNC")  // SYNC
             D_FF #(.width(8),.REG_EN(OPMODEREG)) OPMODE_SYNC ( .D(OPMODE), .EN(CEOPMODE), .CLK(CLK), .RST(RSTOPMODE), .Q(OPMODE_wire) );
        else   // ASYNC
            D_FF_ASYNC #(.width(8),.REG_EN(OPMODEREG)) OPMODE_ASYNC ( .D(OPMODE), .EN(CEOPMODE), .CLK(CLK), .RST(RSTOPMODE), .Q(OPMODE_wire) );
    endgenerate

endmodule
```

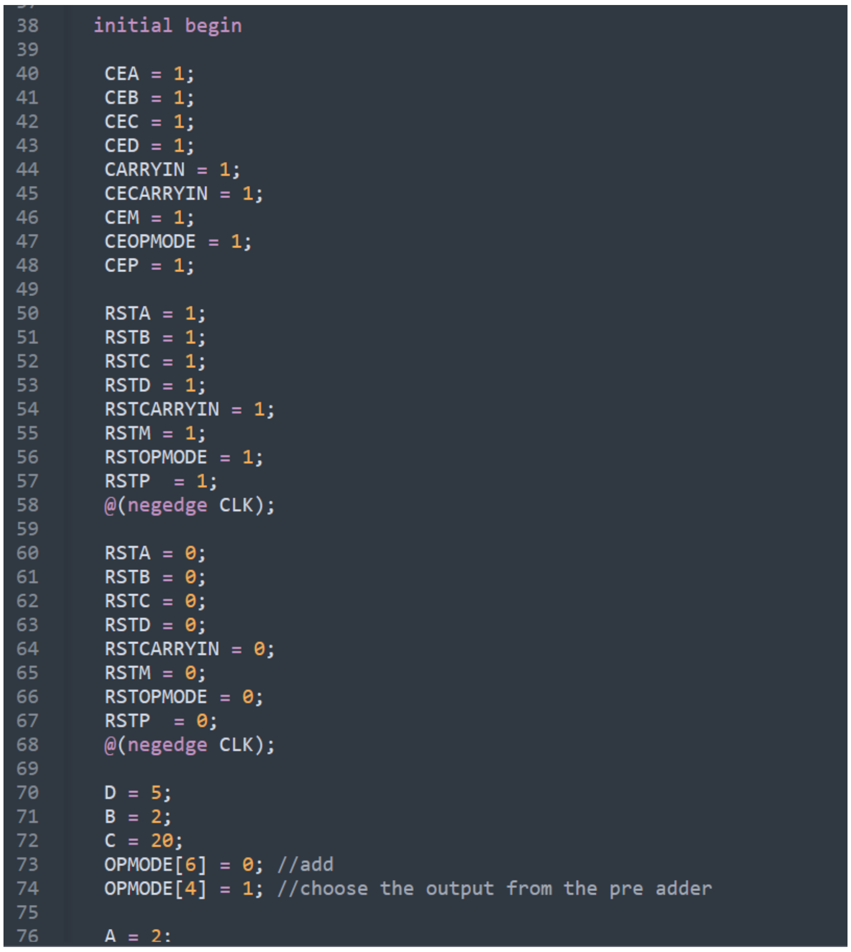
Figure 8: DSP Code part 6.

2. Testbench Code



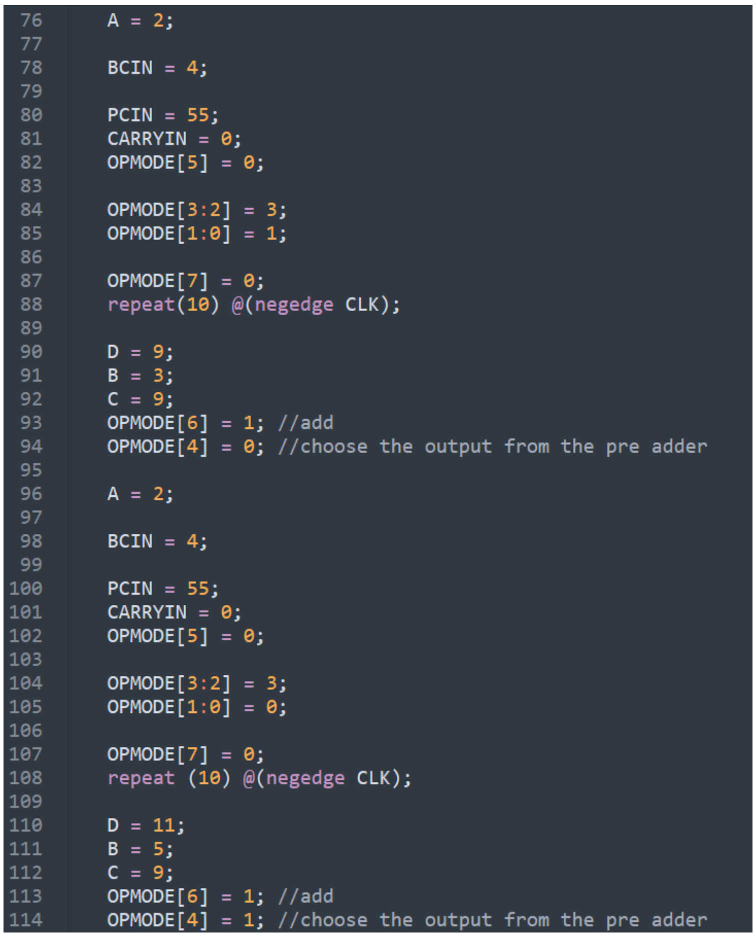
```verilog
module DSP_Project_TB();

    reg [17:0]A, B, D, BCIN;
    reg [47:0] C, PCIN;
    reg CEA, CEB, CEC, CED, CARRYIN, CLK, CECARRYIN, CEM, CEOPMODE, CEP;
    reg RSTA, RSTB, RSTC, RSTD, RSTCARRYIN, RSTM, RSTOPMODE, RSTP;
    reg [7:0]OPMODE;

    wire [17:0]BCOUT;
    wire [47:0]PCOUT, P;
    wire [35:0]M;
    wire CARRYOUT, CARRYOUTF;

    parameter A0REG = 0,
              A1REG = 1,
              B0REG = 0,
              B1REG = 1,
              CREG = 1,
              DREG = 1,
              MREG = 1,
              PREG = 1,
              CARRYINREG = 1,
              CARRYOUTREG = 1,
              OPMODEREG = 1,
              CARRYINSEL = 1, //default is OPMODE_wire[5] which in code optained by making CARRYINSEL = 1
              B_INPUT = "DIRECT", //CASCADE
              RSTTYPE = "SYNC";


   DSP_Project #(A0REG, A1REG, B0REG, B1REG, CREG, DREG, MREG, PREG, CARRYINREG, CARRYOUTREG, OPMODEREG, CARRYINSEL, B_INPUT, RSTTYPE) DSP1 (A, B,

   initial begin
        CLK = 0;
        forever
            #10 CLK = ~CLK;
   end

   initial begin
```

Figure 9: Test Bench code part 1.



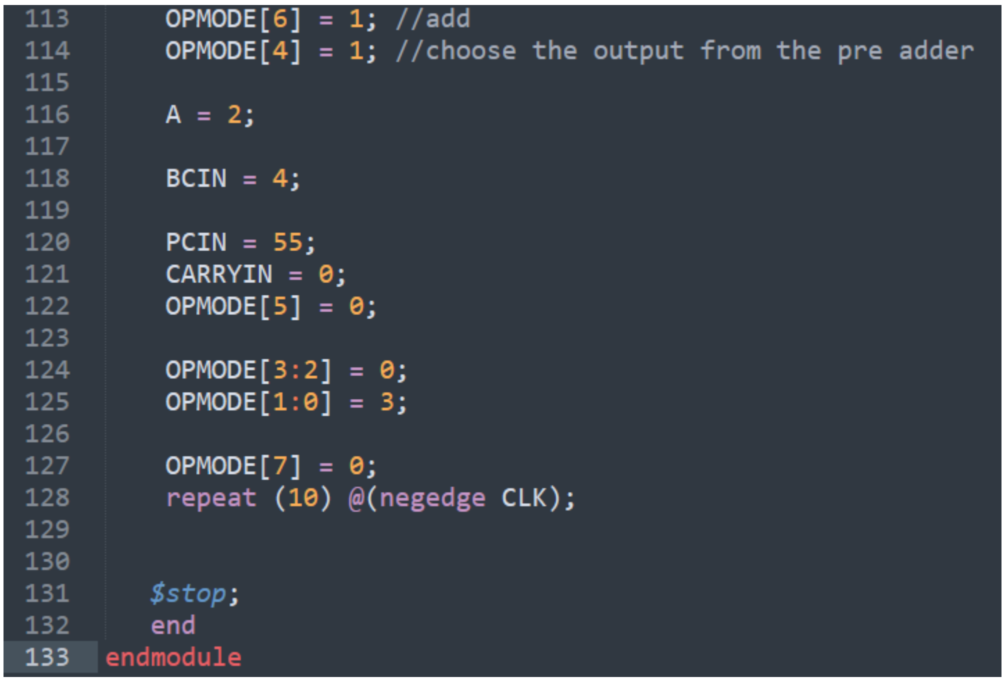
```verilog
   initial begin

    CEA = 1;
    CEB = 1;
    CEC = 1;
    CED = 1;
    CARRYIN = 1;
    CECARRYIN = 1;
    CEM = 1;
    CEOPMODE = 1;
    CEP = 1;

    RSTA = 1;
    RSTB = 1;
    RSTC = 1;
    RSTD = 1;
    RSTCARRYIN = 1;
    RSTM = 1;
    RSTOPMODE = 1;
    RSTP  = 1;
    @(negedge CLK);

    RSTA = 0;
    RSTB = 0;
    RSTC = 0;
    RSTD = 0;
    RSTCARRYIN = 0;
    RSTM = 0;
    RSTOPMODE = 0;
    RSTP  = 0;
    @(negedge CLK);

    D = 5;
    B = 2;
    C = 20;
    OPMODE[6] = 0; //add
    OPMODE[4] = 1; //choose the output from the pre adder

    A = 2;
```

Figure 10: Test Bench code part 2.



```
    A = 2;

    BCIN = 4;

    PCIN = 55;
    CARRYIN = 0;
    OPMODE[5] = 0;

    OPMODE[3:2] = 3;
    OPMODE[1:0] = 1;

    OPMODE[7] = 0;
    repeat(10) @(negedge CLK);

    D = 9;
    B = 3;
    C = 9;
    OPMODE[6] = 1; //add
    OPMODE[4] = 0; //choose the output from the pre adder

    A = 2;

    BCIN = 4;

    PCIN = 55;
    CARRYIN = 0;
    OPMODE[5] = 0;

    OPMODE[3:2] = 3;
    OPMODE[1:0] = 0;

    OPMODE[7] = 0;
    repeat (10) @(negedge CLK);

    D = 11;
    B = 5;
    C = 9;
    OPMODE[6] = 1; //add
    OPMODE[4] = 1; //choose the output from the pre adder
```

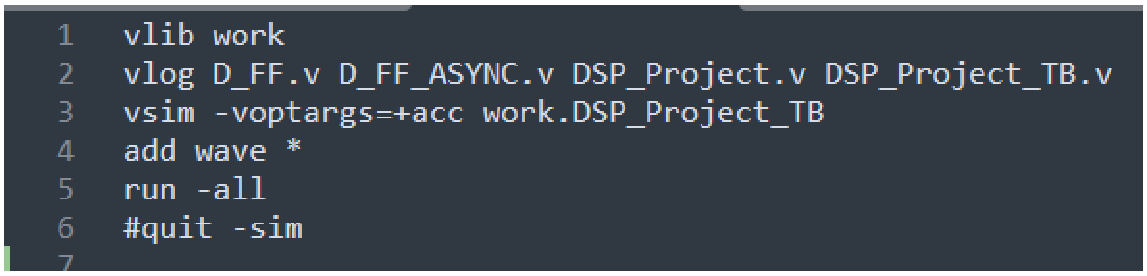
Figure 11: Test Bench code part 3.



```
    OPMODE[6] = 1; //add
    OPMODE[4] = 1; //choose the output from the pre adder

    A = 2;

    BCIN = 4;

    PCIN = 55;
    CARRYIN = 0;
    OPMODE[5] = 0;

    OPMODE[3:2] = 0;
    OPMODE[1:0] = 3;

    OPMODE[7] = 0;
    repeat (10) @(negedge CLK);


  $stop;
  end
endmodule
```

Figure 12: Test Bench code part 4.

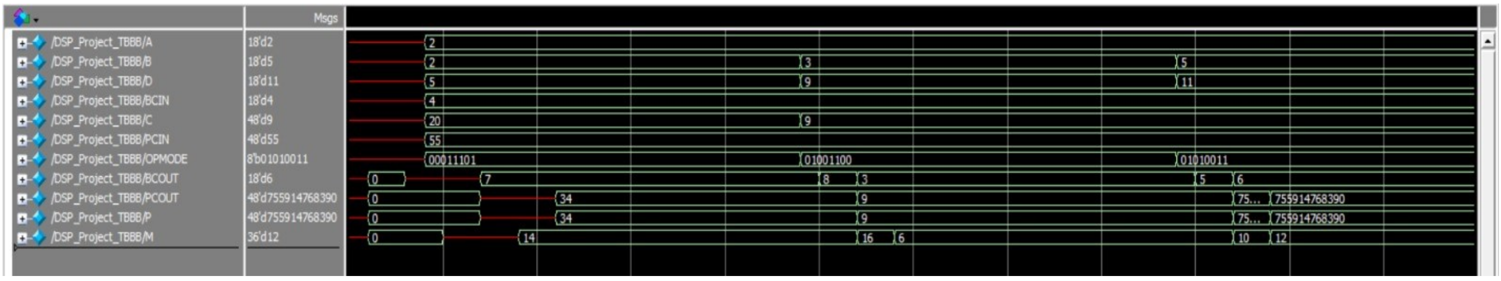
3. Do File



```
vlib work
vlog D_FF.v D_FF_ASYNC.v DSP_Project.v DSP_Project_TB.v
vsim -voptargs=+acc work.DSP_Project_TB
add wave *
run -all
#quit -sim
```

Figure 13: Do File.

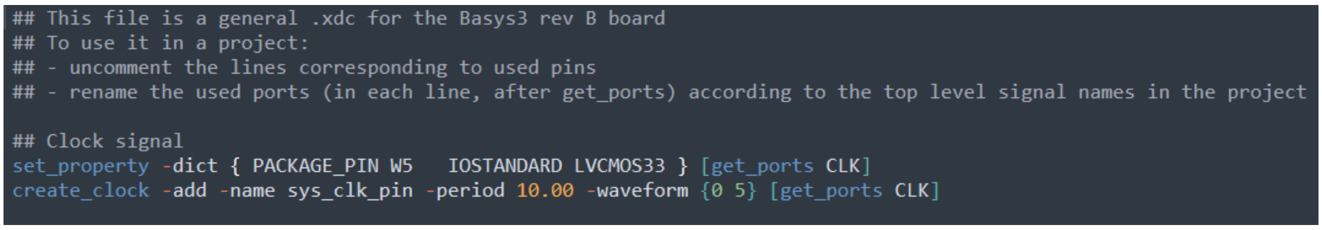
4. Simulation “QuestaSim Snippets”



	Msgs
/DSP_Project_TBBB/A	18'd2
/DSP_Project_TBBB/B	18'd5
/DSP_Project_TBBB/D	18'd11
/DSP_Project_TBBB/BCIN	18'd4
/DSP_Project_TBBB/C	48'd9
/DSP_Project_TBBB/PCIN	48'd55
/DSP_Project_TBBB/OPMODE	8'b01010011
/DSP_Project_TBBB/BCOUT	18'd6
/DSP_Project_TBBB/PCOUT	48'd755914768390
/DSP_Project_TBBB/P	48'd755914768390
/DSP_Project_TBBB/M	36'd12

Figure 14: Test bench output.

5. Constraint File

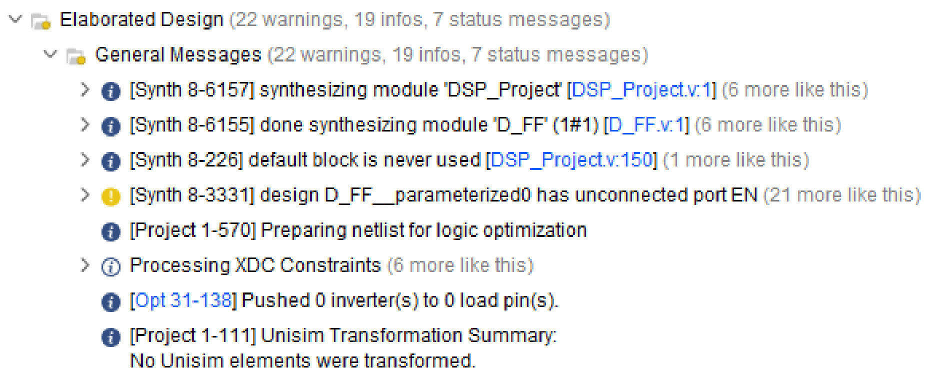


```
## This file is a general .xdc for the Basys3 rev B board
## To use it in a project:
## - uncomment the lines corresponding to used pins
## - rename the used ports (in each line, after get_ports) according to the top level signal names in the project

## Clock signal
set_property -dict { PACKAGE_PIN W5   IOSTANDARD LVCMOS33 } [get_ports CLK]
create_clock -add -name sys_clk_pin -period 10.00 -waveform {0 5} [get_ports CLK]
```

Figure 15: Constraint file.

6. Elaboration

6.1. Messages Tab



Elaborated Design (22 warnings, 19 infos, 7 status messages)

General Messages (22 warnings, 19 infos, 7 status messages)

```
[Synth 8-6157] synthesizing module 'DSP_Project' [DSP_Project.v:1] (6 more like this)
[Synth 8-6155] done synthesizing module 'D_FF' (1#1) [D_FF.v:1] (6 more like this)
[Synth 8-226] default block is never used [DSP_Project.v:150] (1 more like this)
[Synth 8-3331] design D_FF__parameterized0 has unconnected port EN (21 more like this)
[Project 1-570] Preparing netlist for logic optimization
Processing XDC Constraints (6 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.
```

Figure 16: Message tab from elaboration section.

6.2. Schematic Snippets

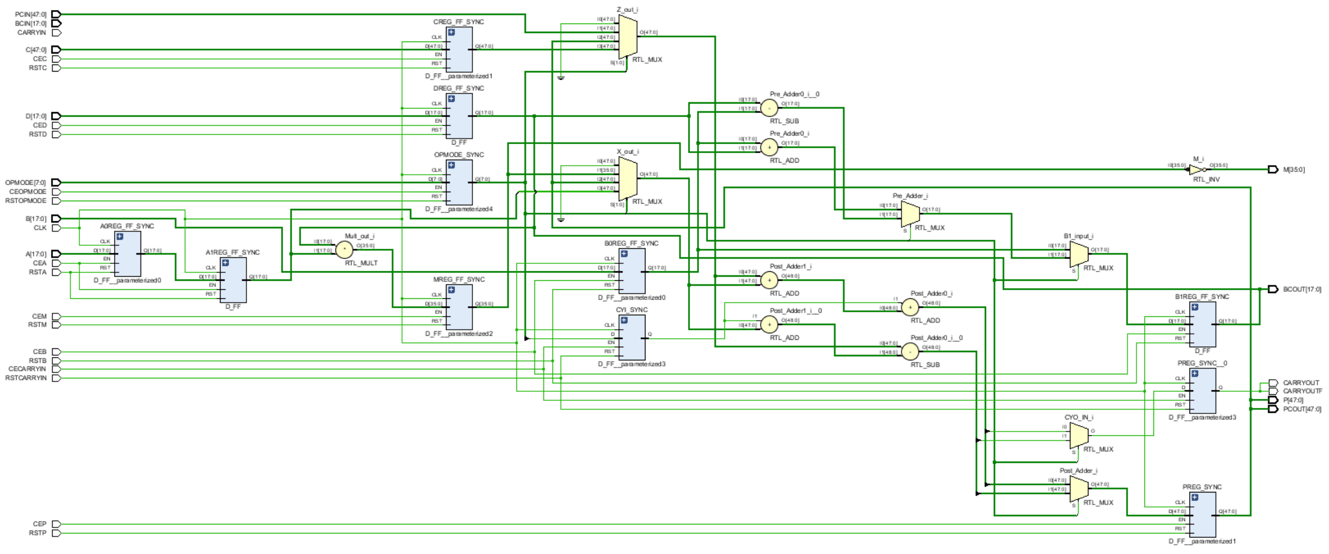
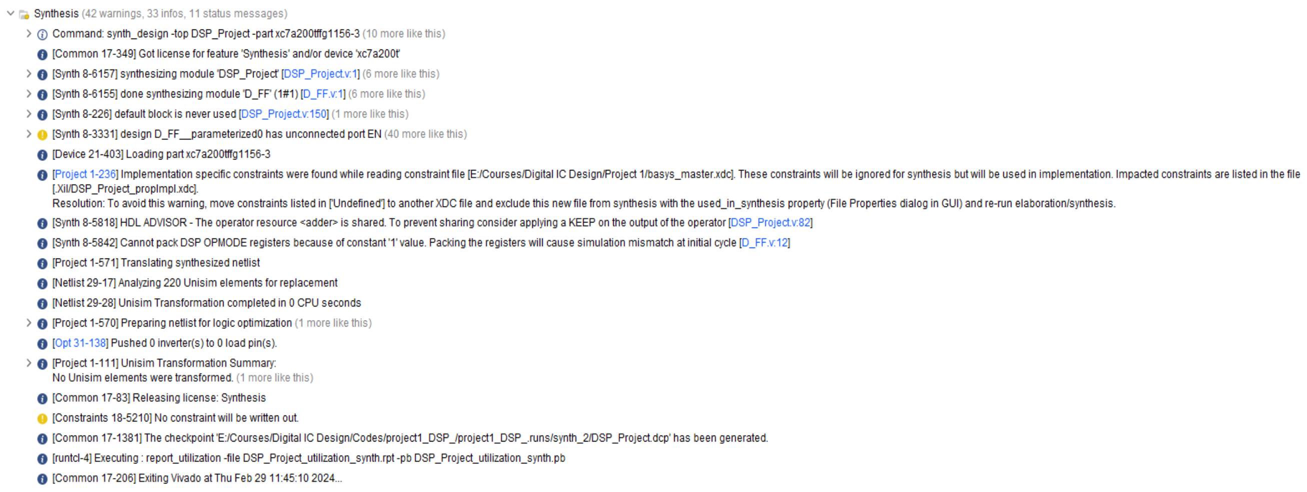


Figure 17: Schematic from elaboration section.

7. Synthesis

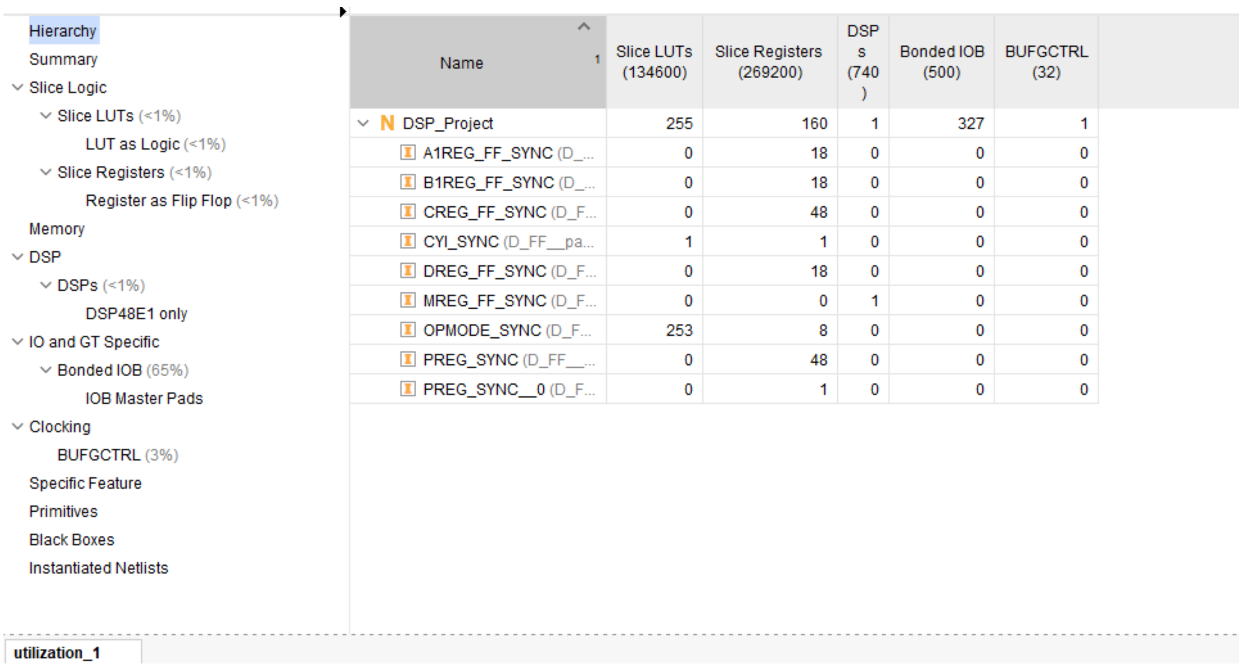
7.1. Messages tab



```
Synthesis (42 warnings, 33 infos, 11 status messages)
  Command: synth_design -top DSP_Project -part xc7a200tffg1156-3 (10 more like this)
  [Common 17-349] Got license for feature 'Synthesis' and/or device 'xc7a200t'
  [Synth 8-6157] synthesizing module 'DSP_Project' [DSP_Project.v:1] (6 more like this)
  [Synth 8-6155] done synthesizing module 'D_FF' (1#1) [D_FF.v:1] (6 more like this)
  [Synth 8-226] default block is never used [DSP_Project.v:150] (1 more like this)
  [Synth 8-3331] design D_FF__parameterized0 has unconnected port EN (40 more like this)
  [Device 21-403] Loading part xc7a200tffg1156-3
  [Project 1-236] Implementation specific constraints were found while reading constraint file [E:/Courses/Digital IC Design/Project 1/basys_master.xdc]. These constraints will be ignored for synthesis but will be used in implementation. Impacted constraints are listed in the file
  [.Xil/DSP_Project_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_Project.v:82]
  [Synth 8-5842] Cannot pack DSP OPMODE registers because of constant '1' value. Packing the registers will cause simulation mismatch at initial cycle [D_FF.v:12]
  [Project 1-571] Translating synthesized netlist
  [Netlist 29-17] Analyzing 220 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
  [Constraints 18-5210] No constraint will be written out.
  [Common 17-1381] The checkpoint 'E:/Courses/Digital IC Design/Codes/project1_DSP_/project1_DSP_.runs/synth_2/DSP_Project.dcp' has been generated.
  [runtcl-4] Executing : report_utilization -file DSP_Project_utilization_synth.rpt -pb DSP_Project_utilization_synth.pb
  [Common 17-206] Exiting Vivado at Thu Feb 29 11:45:10 2024...
```

Figure 18: Message tab from Synthesis section.

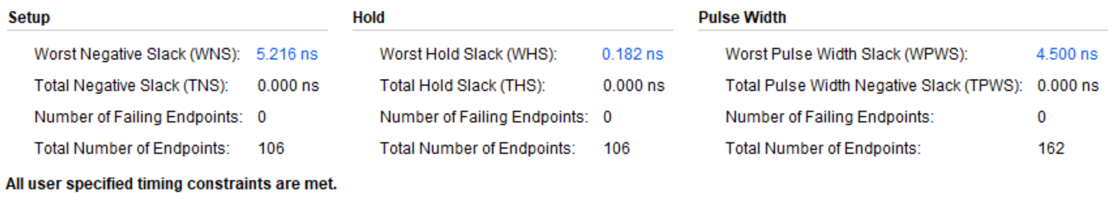
7.2. Utilization Report



Name	Slice LUTs (134600)	Slice Registers (269200)	DSPs (740)	Bonded IOB (500)	BUFGCTRL (32)
DSP_Project	255	160	1	327	1
A1REG_FF_SYNC (D_...	0	18	0	0	0
B1REG_FF_SYNC (D_...	0	18	0	0	0
CREG_FF_SYNC (D_F...	0	48	0	0	0
CYI_SYNC (D_FF__pa...	1	1	0	0	0
DREG_FF_SYNC (D_F...	0	18	0	0	0
MREG_FF_SYNC (D_F...	0	0	1	0	0
OPMODE_SYNC (D_F...	253	8	0	0	0
PREG_SYNC (D_FF__...	0	48	0	0	0
PREG_SYNC__0 (D_F...	0	1	0	0	0

Figure 19: Utilization from Synthesis section.

7.3. Timing Report



Setup		Hold		Pulse Width	
Worst Negative Slack (WNS):	5.216 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.

Figure 20: Timing report from Synthesis section.

7.4. Schematic Snippets

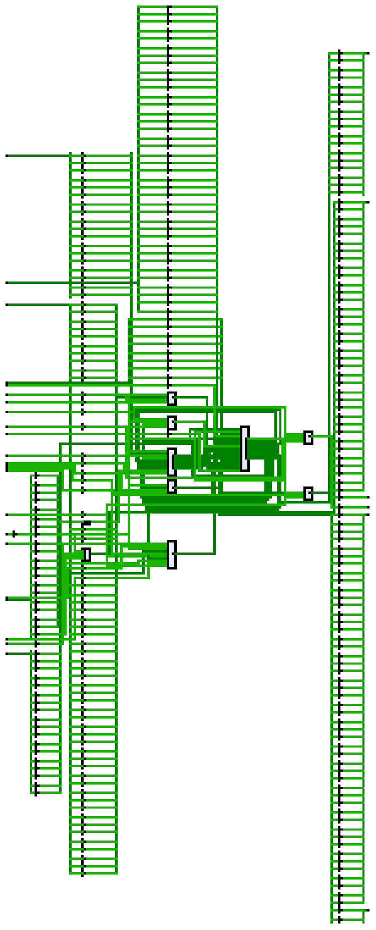


Figure 21: Schematic from Synthesis *section.*

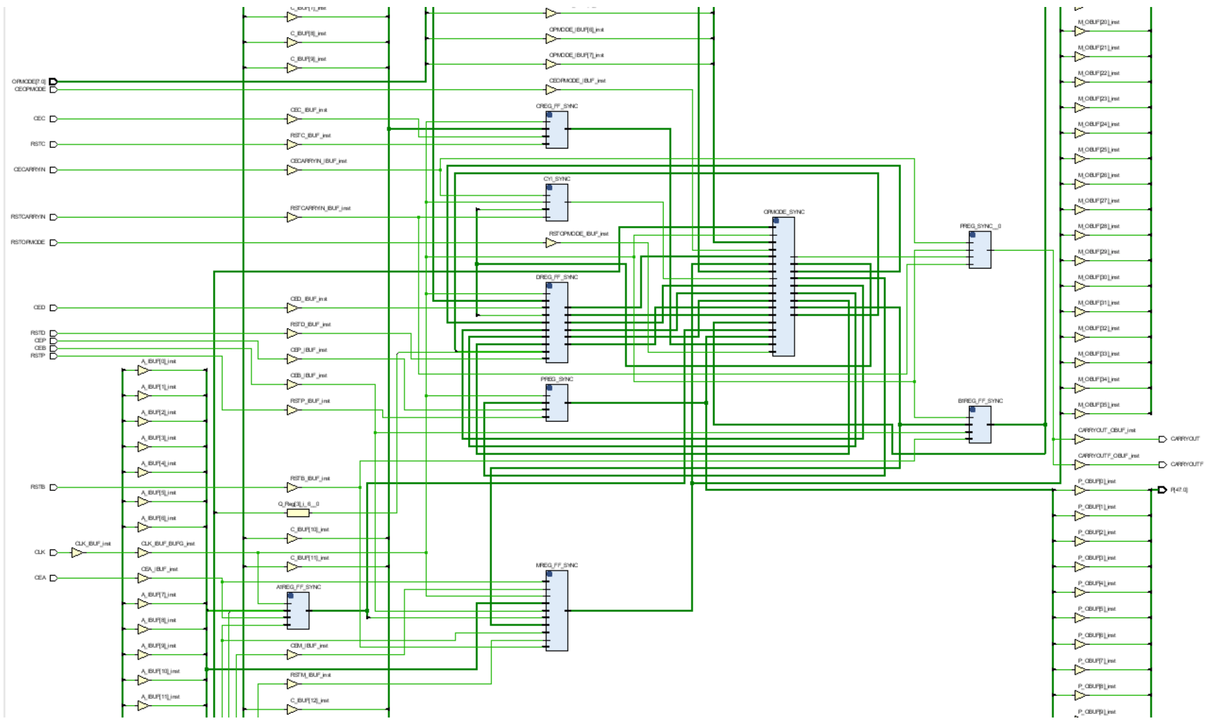
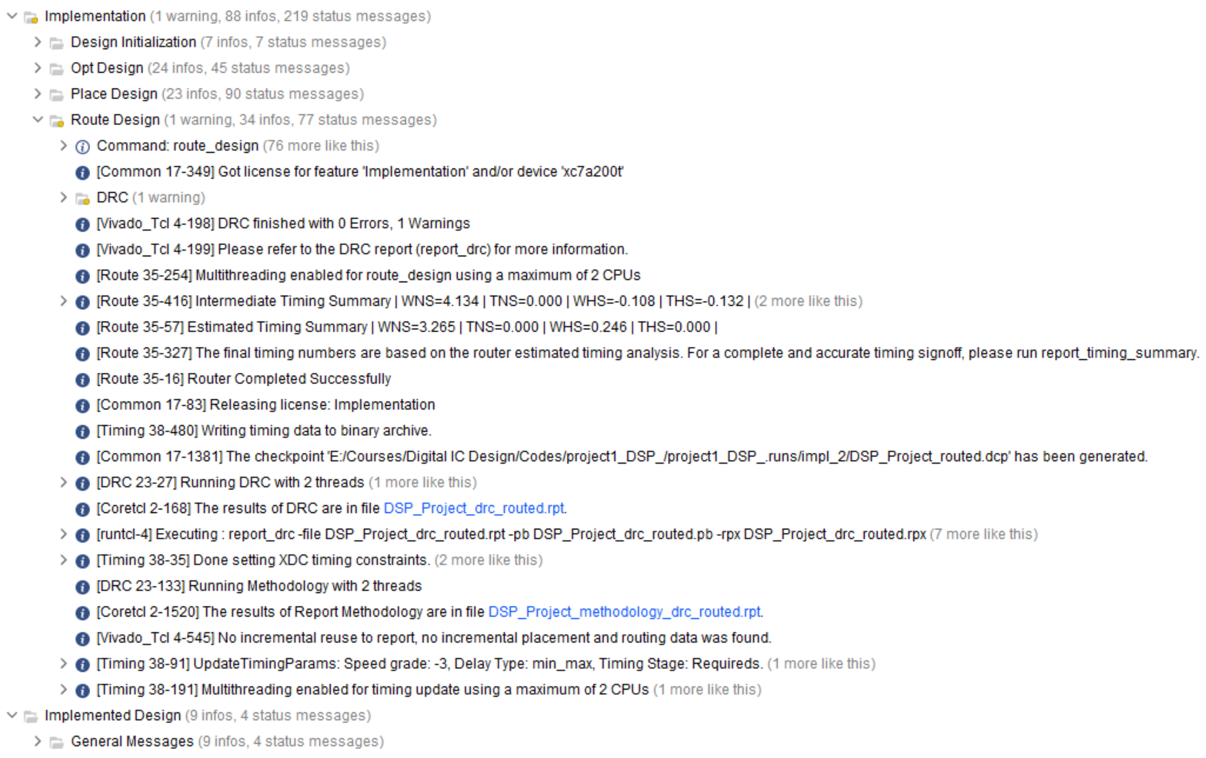


Figure 22: Schematic from Synthesis section zoomed in.

8. Implementation

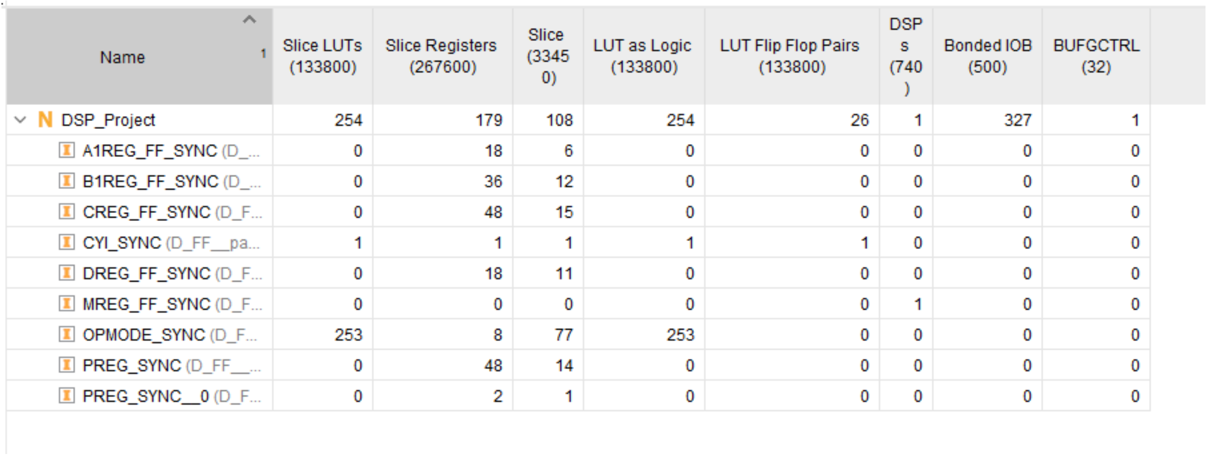
8.1. Messages tab



```
v Implementation (1 warning, 88 infos, 219 status messages)
  > Design Initialization (7 infos, 7 status messages)
  > Opt Design (24 infos, 45 status messages)
  > Place Design (23 infos, 90 status messages)
  v Route Design (1 warning, 34 infos, 77 status messages)
    > Command: route_design (76 more like this)
      [Common 17-349] Got license for feature 'Implementation' and/or device 'xc7a200t'
    > DRC (1 warning)
      [Vivado_Tcl 4-198] DRC finished with 0 Errors, 1 Warnings
      [Vivado_Tcl 4-199] Please refer to the DRC report (report_drc) for more information.
      [Route 35-254] Multithreading enabled for route_design using a maximum of 2 CPUs
    > [Route 35-416] Intermediate Timing Summary | WNS=4.134 | TNS=0.000 | WHS=-0.108 | THS=-0.132 | (2 more like this)
      [Route 35-57] Estimated Timing Summary | WNS=3.265 | TNS=0.000 | WHS=0.246 | THS=0.000 |
      [Route 35-327] The final timing numbers are based on the router estimated timing analysis. For a complete and accurate timing signoff, please run report_timing_summary.
      [Route 35-16] Router Completed Successfully
      [Common 17-83] Releasing license: Implementation
      [Timing 38-480] Writing timing data to binary archive.
      [Common 17-1381] The checkpoint 'E:/Courses/Digital IC Design/Codes/project1_DSP_/project1_DSP_.runs/impl_2/DSP_Project_routed.dcp' has been generated.
    > [DRC 23-27] Running DRC with 2 threads (1 more like this)
      [Coretcl 2-168] The results of DRC are in file DSP_Project_drc_routed.rpt.
    > [runtcl-4] Executing : report_drc -file DSP_Project_drc_routed.rpt -pb DSP_Project_drc_routed.pb -rpx DSP_Project_drc_routed.rpx (7 more like this)
    > [Timing 38-35] Done setting XDC timing constraints. (2 more like this)
      [DRC 23-133] Running Methodology with 2 threads
      [Coretcl 2-1520] The results of Report Methodology are in file DSP_Project_methodology_drc_routed.rpt.
      [Vivado_Tcl 4-545] No incremental reuse to report, no incremental placement and routing data was found.
    > [Timing 38-91] UpdateTimingParams: Speed grade: -3, Delay Type: min_max, Timing Stage: Requireds. (1 more like this)
    > [Timing 38-191] Multithreading enabled for timing update using a maximum of 2 CPUs (1 more like this)
v Implemented Design (9 infos, 4 status messages)
  > General Messages (9 infos, 4 status messages)
```

Figure 23: *Message tab from Implementation section.*

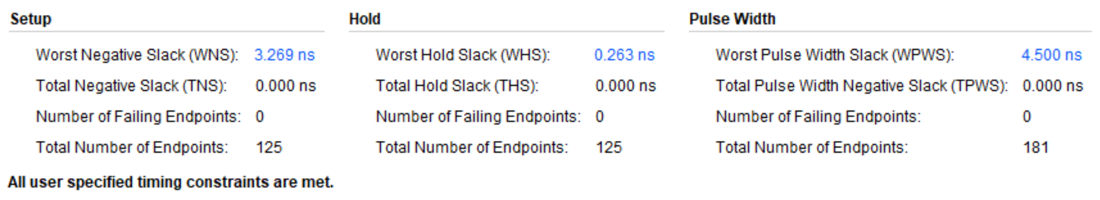
8.2. Utilization Report



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)
DSP_Project	254	179	108	254	26	1	327	1
A1REG_FF_SYNC (D_...	0	18	6	0	0	0	0	0
B1REG_FF_SYNC (D_...	0	36	12	0	0	0	0	0
CREG_FF_SYNC (D_F...	0	48	15	0	0	0	0	0
CYI_SYNC (D_FF__pa...	1	1	1	1	1	0	0	0
DREG_FF_SYNC (D_F...	0	18	11	0	0	0	0	0
MREG_FF_SYNC (D_F...	0	0	0	0	0	1	0	0
OPMODE_SYNC (D_F...	253	8	77	253	0	0	0	0
PREG_SYNC (D_FF__...	0	48	14	0	0	0	0	0
PREG_SYNC__0 (D_F...	0	2	1	0	0	0	0	0

Figure 24: *Utilization from Implementation section.*

8.3. Timing Report



Setup

Worst Negative Slack (WNS):	3.269 ns
Total Negative Slack (TNS):	0.000 ns
Number of Failing Endpoints:	0
Total Number of Endpoints:	125

Hold

Worst Hold Slack (WHS):	0.263 ns
Total Hold Slack (THS):	0.000 ns
Number of Failing Endpoints:	0
Total Number of Endpoints:	125

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.

Figure 25: Timing report from Implementation section.

8.4. Device Snippets

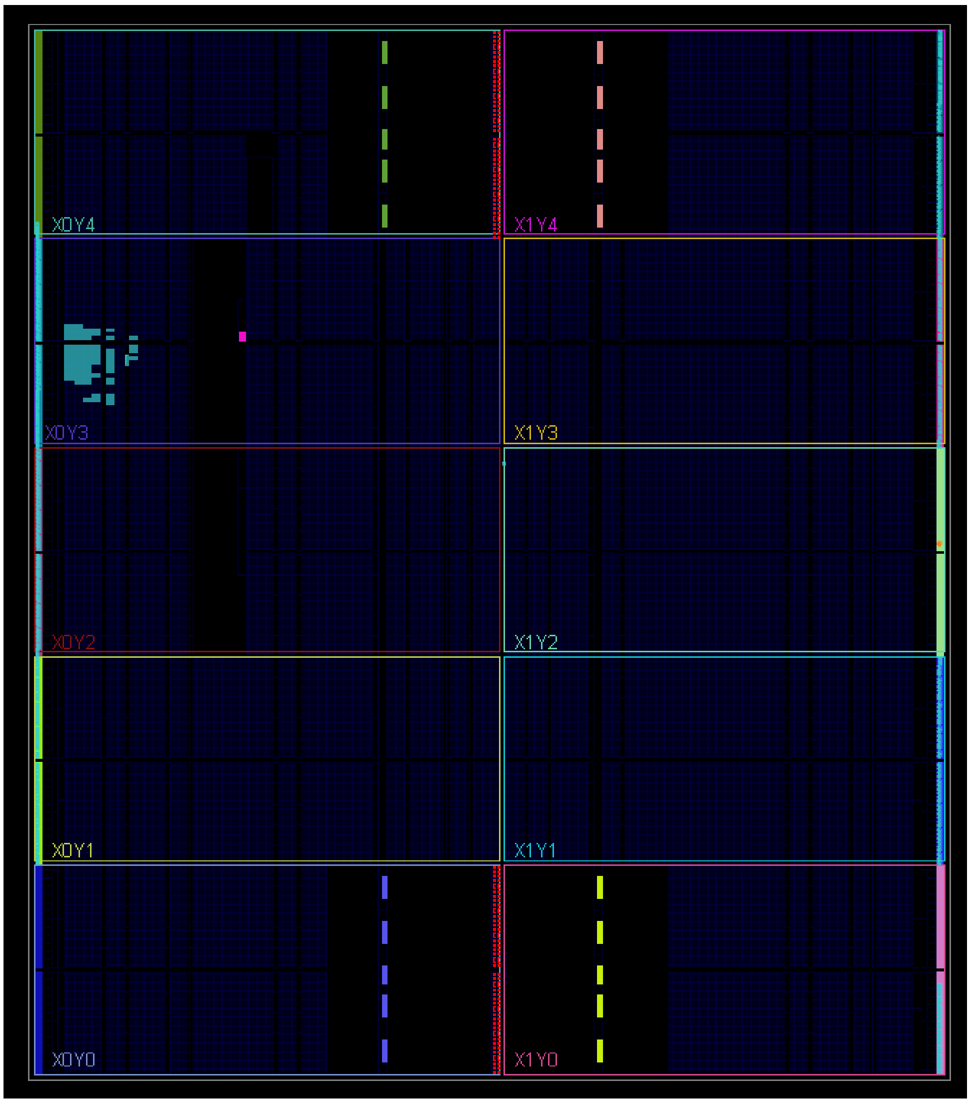


Figure 26: Device snippets from implementation section.