به نام خدا

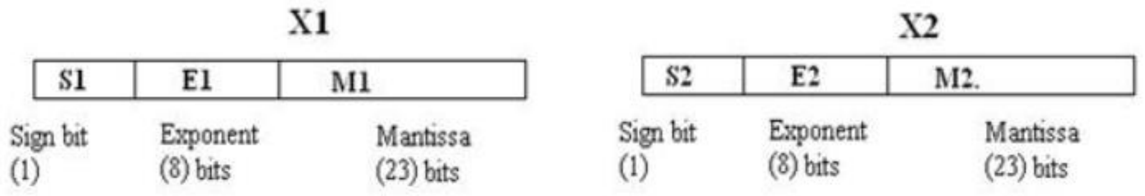
گزارشکار پروژه 6 سیستم های دیجیتال 1

دکتر نوابی

دانشجو : فاطمه نائینیان

شماره دانشجویی :810198479

تیر 1400



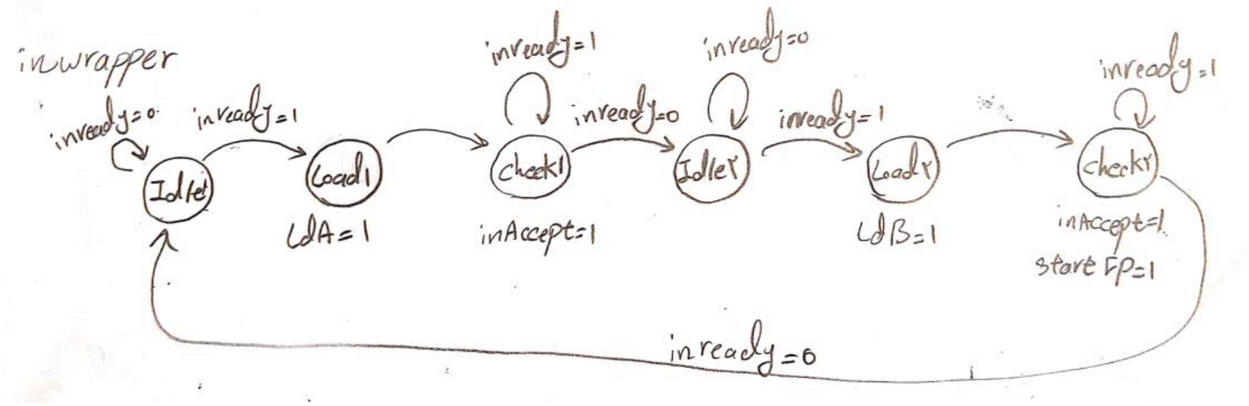
Result X3 = X1 * X2 = $(-1)^{s1}$ $(M1 \times 2^{E1}) * (-1)^{s2} (M2 \times 2^{E2})$

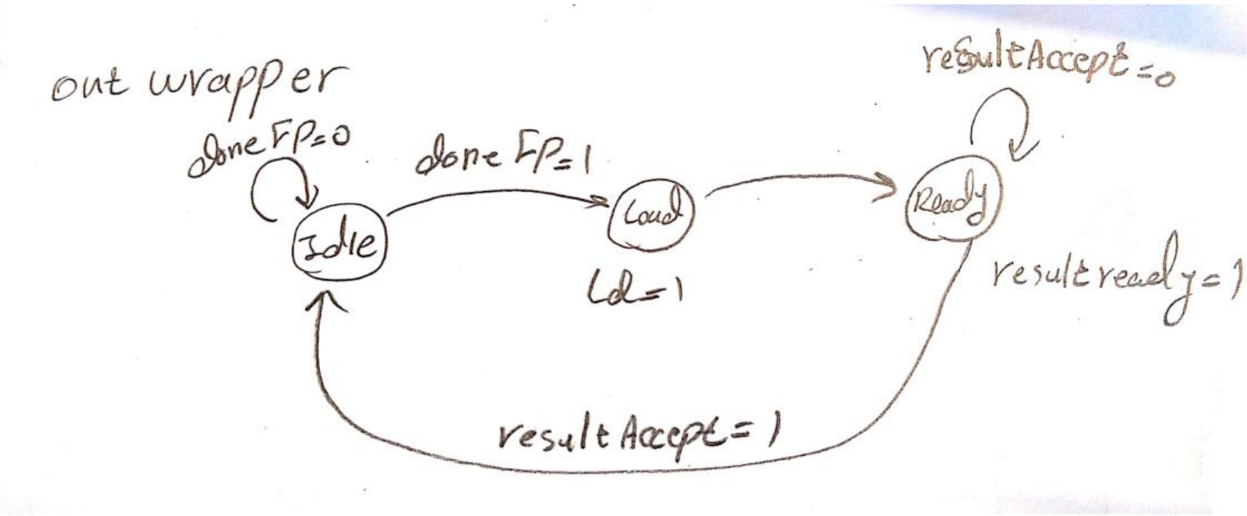
S3=S1^S2

M3=M1 * M2

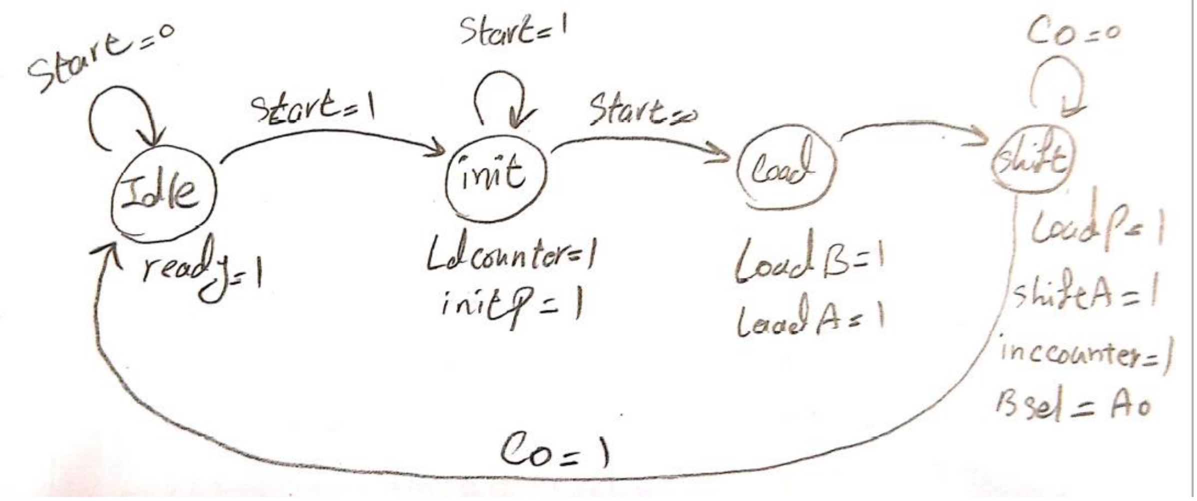
E3=E1+E2-bias

a)Block digram

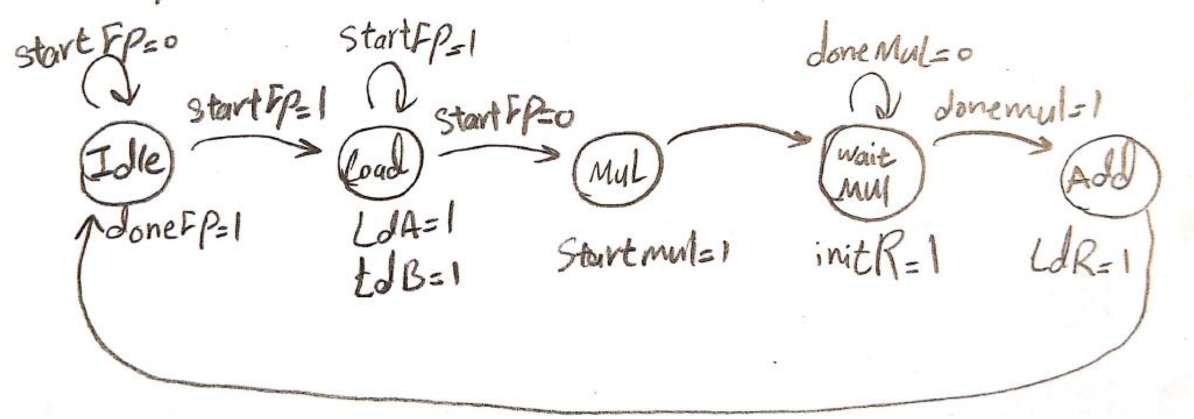




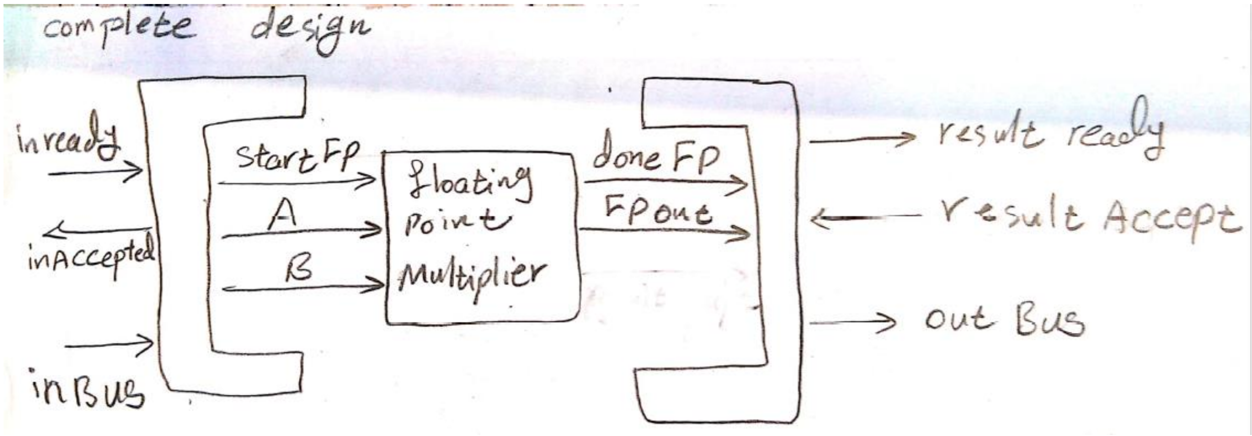
Sequential



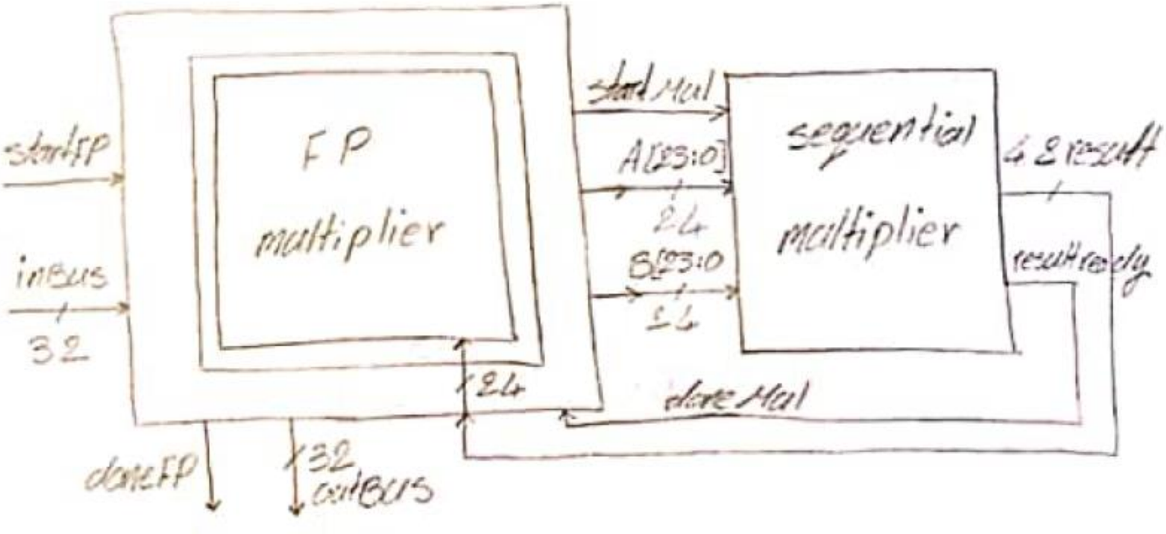
FP multiplier



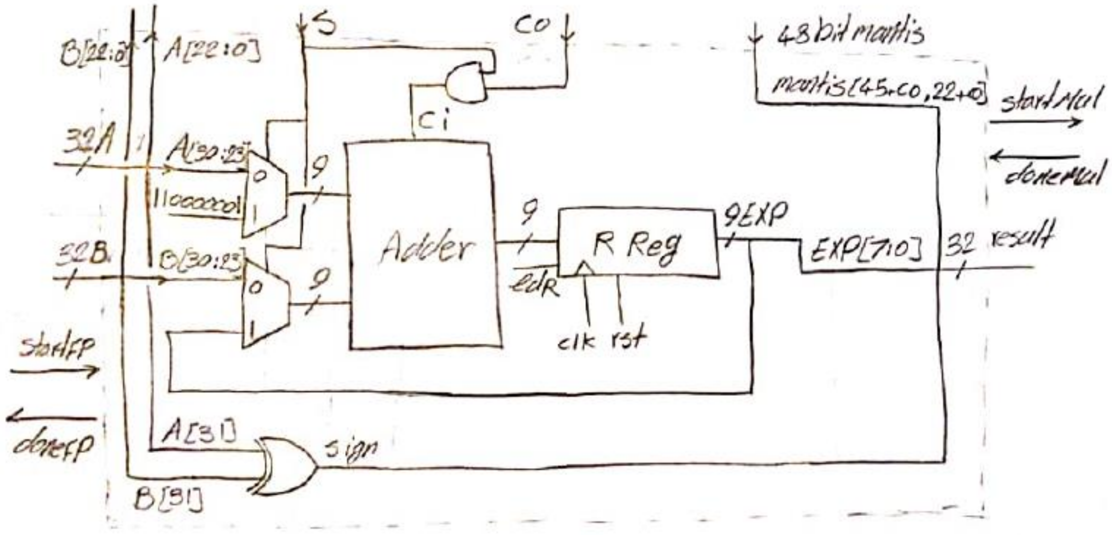
B)

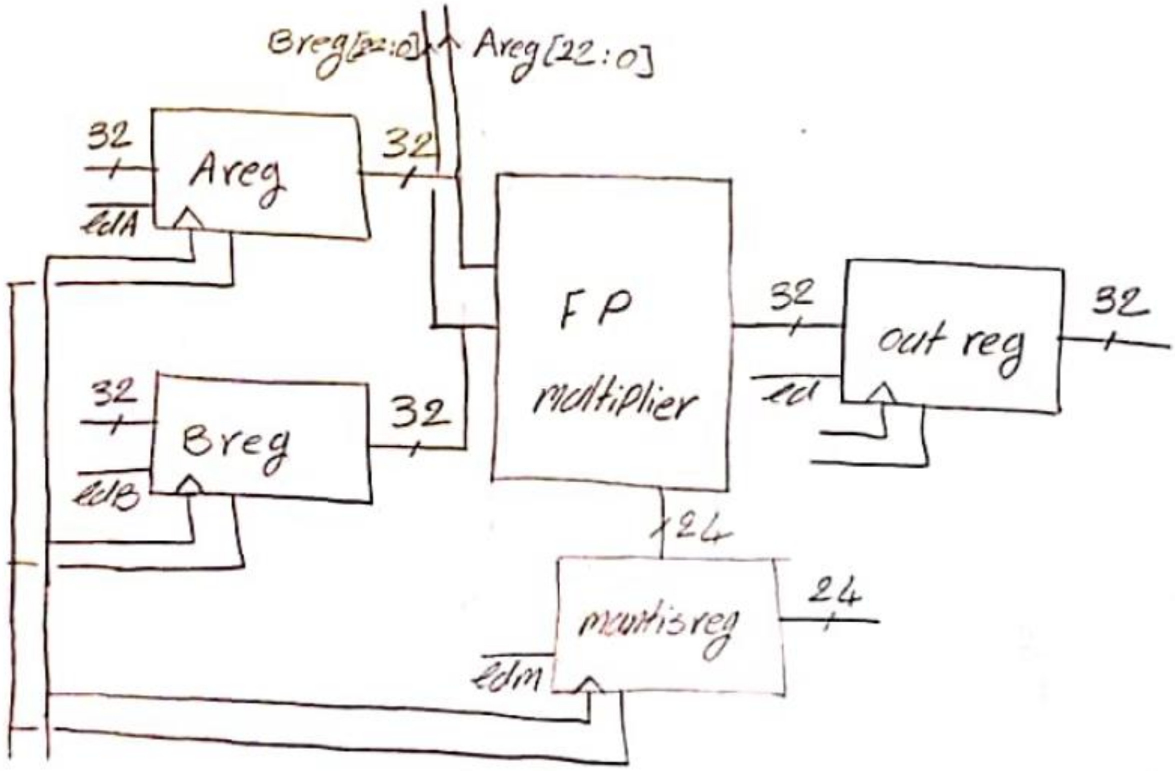


Fp multiplier

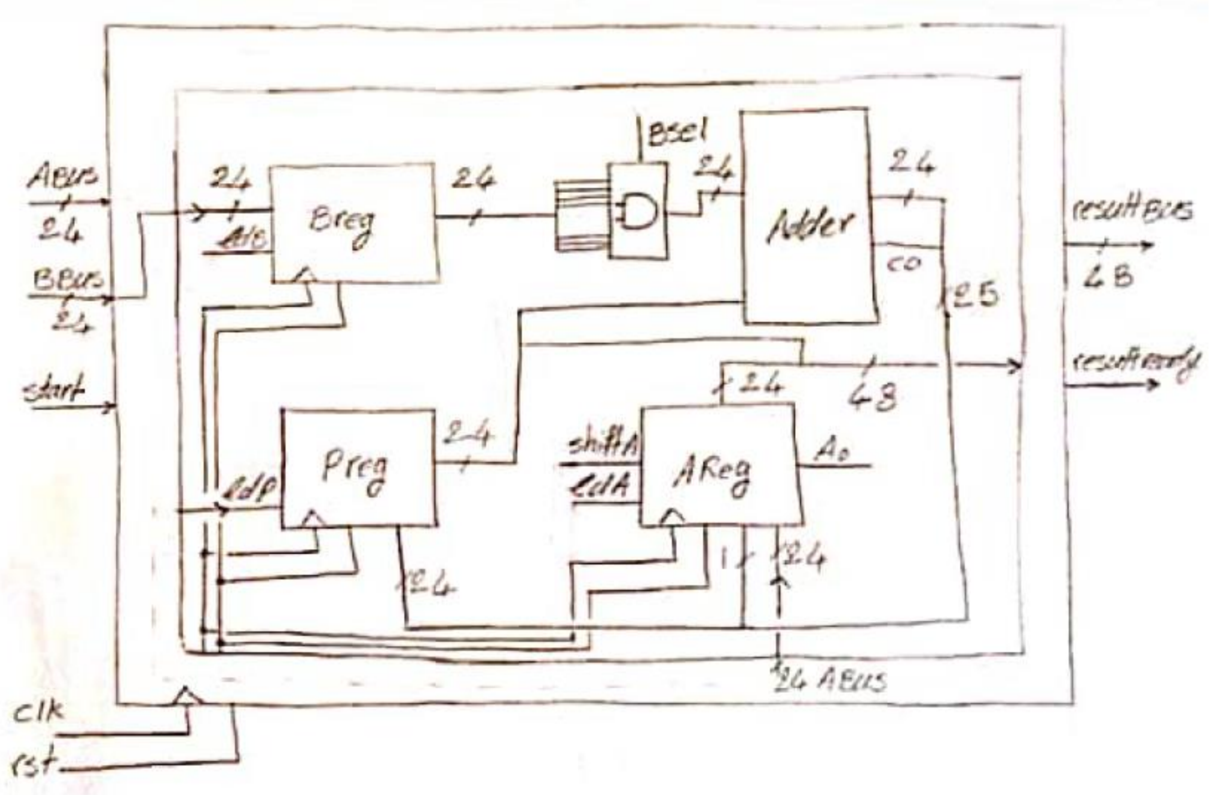


c)wrapper


B[22:0]
A[22:0]
S
co
48 bit mantis
mantis[45+co,22+co]
startMul
doneMul
32 A
A[30:23]
0
1
1100000001
9
ci
32 B
B[30:23]
0
1
9
Adder
9
ldR
R Reg
clk
rst
9 EXP
EXP[7:0]
32
result
startFP
doneFP
A[31]
B[31]
sign

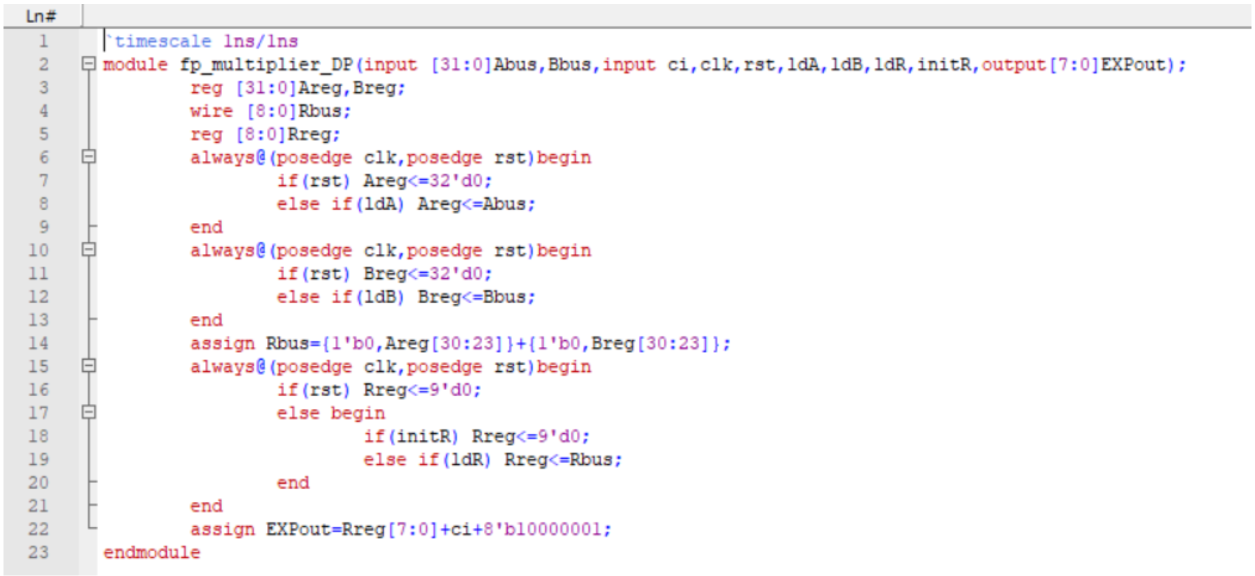

Breg[22:0]
Areg[22:0]
32
Areg
32
ldA
32
Breg
32
ldB
FP
multiplier
32
out reg
32
ld
24
mantisreg
24
ldm

D)Sequential multiplier



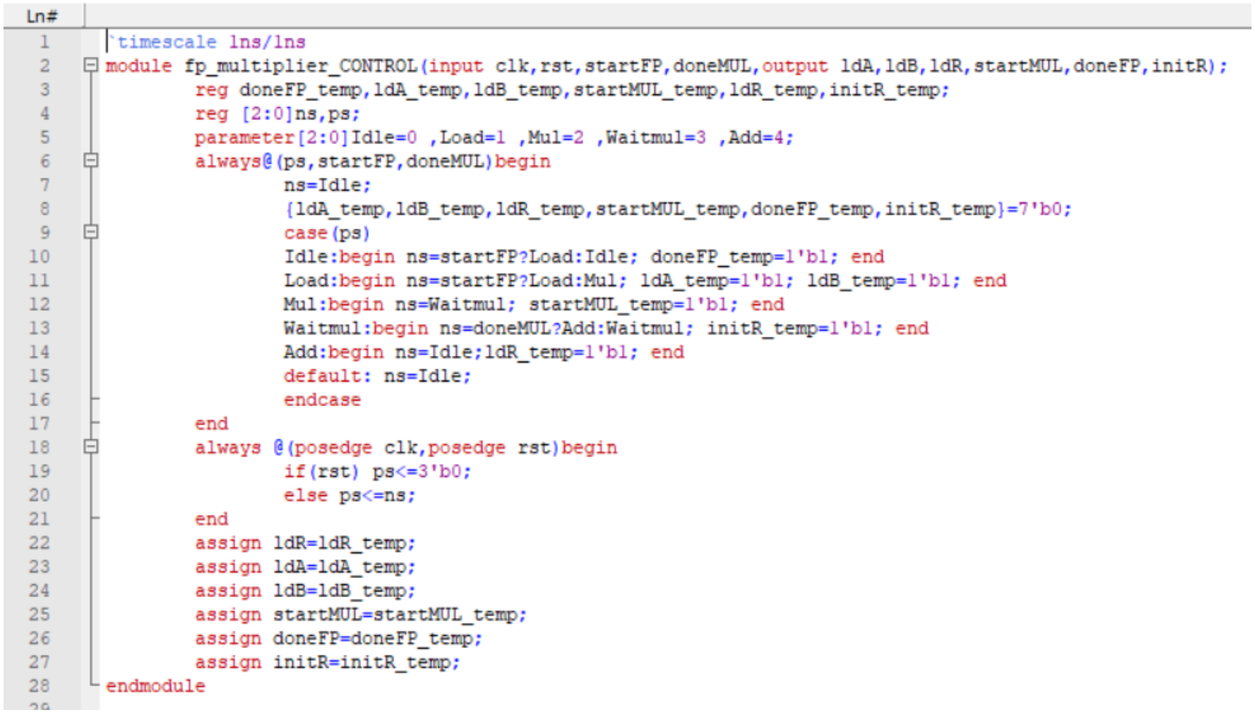
E) FP multiplier

Datapass



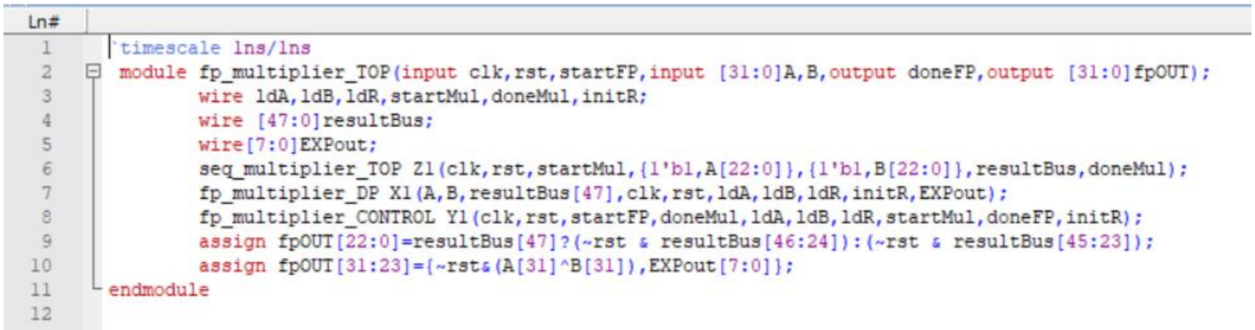
```
`timescale 1ns/1ns
module fp_multiplier_DP(input [31:0]Abus,Bbus,input ci,clk,rst,ldA,ldB,ldR,initR,output[7:0]EXPout);
        reg [31:0]Areg,Breg;
        wire [8:0]Rbus;
        reg [8:0]Rreg;
        always@(posedge clk,posedge rst)begin
                if(rst) Areg<=32'd0;
                else if(ldA) Areg<=Abus;
        end
        always@(posedge clk,posedge rst)begin
                if(rst) Breg<=32'd0;
                else if(ldB) Breg<=Bbus;
        end
        assign Rbus={1'b0,Areg[30:23]}+{1'b0,Breg[30:23]};
        always@(posedge clk,posedge rst)begin
                if(rst) Rreg<=9'd0;
                else begin
                        if(initR) Rreg<=9'd0;
                        else if(ldR) Rreg<=Rbus;
                end
        end
        assign EXPout=Rreg[7:0]+ci+8'b10000001;
endmodule
```

Control



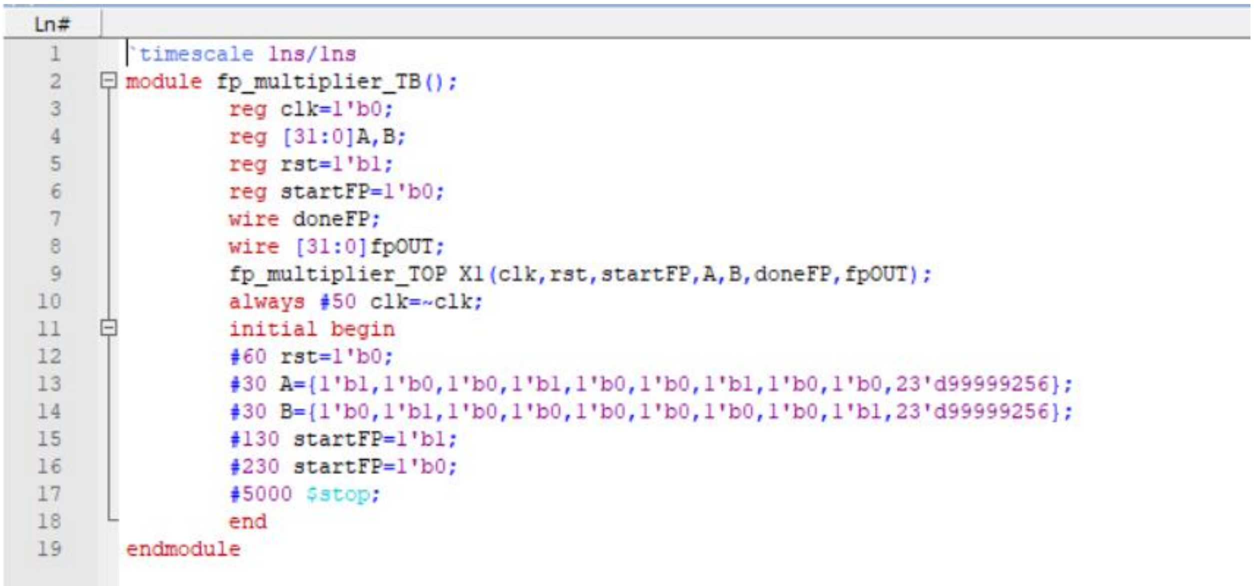
```
`timescale 1ns/1ns
module fp_multiplier_CONTROL(input clk,rst,startFP,doneMUL,output ldA,ldB,ldR,startMUL,doneFP,initR);
        reg doneFP_temp,ldA_temp,ldB_temp,startMUL_temp,ldR_temp,initR_temp;
        reg [2:0]ns,ps;
        parameter[2:0]Idle=0 ,Load=1 ,Mul=2 ,Waitmul=3 ,Add=4;
        always@(ps,startFP,doneMUL)begin
                ns=Idle;
                {ldA_temp,ldB_temp,ldR_temp,startMUL_temp,doneFP_temp,initR_temp}=7'b0;
                case(ps)
                Idle:begin ns=startFP?Load:Idle; doneFP_temp=1'b1; end
                Load:begin ns=startFP?Load:Mul; ldA_temp=1'b1; ldB_temp=1'b1; end
                Mul:begin ns=Waitmul; startMUL_temp=1'b1; end
                Waitmul:begin ns=doneMUL?Add:Waitmul; initR_temp=1'b1; end
                Add:begin ns=Idle;ldR_temp=1'b1; end
                default: ns=Idle;
                endcase
        end
        always @(posedge clk,posedge rst)begin
                if(rst) ps<=3'b0;
                else ps<=ns;
        end
        assign ldR=ldR_temp;
        assign ldA=ldA_temp;
        assign ldB=ldB_temp;
        assign startMUL=startMUL_temp;
        assign doneFP=doneFP_temp;
        assign initR=initR_temp;
endmodule
```

Top level



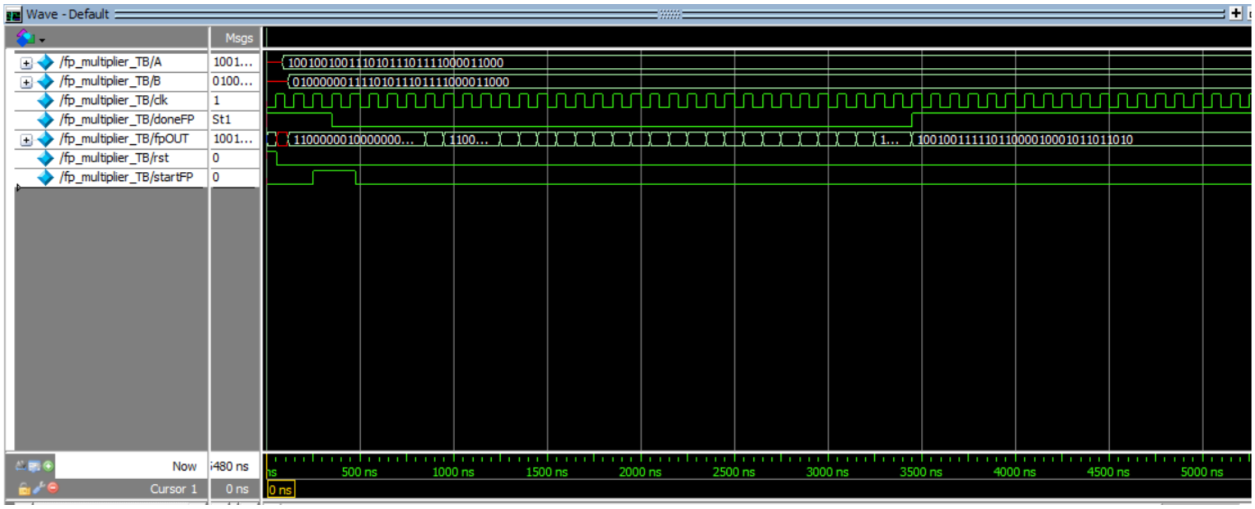
```
`timescale 1ns/1ns
 module fp_multiplier_TOP(input clk,rst,startFP,input [31:0]A,B,output doneFP,output [31:0]fpOUT);
        wire ldA,ldB,ldR,startMul,doneMul,initR;
        wire [47:0]resultBus;
        wire[7:0]EXPout;
        seq_multiplier_TOP Z1(clk,rst,startMul,{1'b1,A[22:0]},{1'b1,B[22:0]},resultBus,doneMul);
        fp_multiplier_DP X1(A,B,resultBus[47],clk,rst,ldA,ldB,ldR,initR,EXPout);
        fp_multiplier_CONTROL Y1(clk,rst,startFP,doneMul,ldA,ldB,ldR,startMul,doneFP,initR);
        assign fpOUT[22:0]=resultBus[47]?(~rst & resultBus[46:24]):(~rst & resultBus[45:23]);
        assign fpOUT[31:23]={~rst&(A[31]^B[31]),EXPout[7:0]};
endmodule
```

Test bench



```
`timescale 1ns/1ns
module fp_multiplier_TB();
        reg clk=1'b0;
        reg [31:0]A,B;
        reg rst=1'b1;
        reg startFP=1'b0;
        wire doneFP;
        wire [31:0]fpOUT;
        fp_multiplier_TOP X1(clk,rst,startFP,A,B,doneFP,fpOUT);
        always #50 clk=~clk;
        initial begin
        #60 rst=1'b0;
        #30 A={1'b1,1'b0,1'b0,1'b1,1'b0,1'b0,1'b1,1'b0,1'b0,23'd99999256};
        #30 B={1'b0,1'b1,1'b0,1'b0,1'b0,1'b0,1'b0,1'b0,1'b1,23'd99999256};
        #130 startFP=1'b1;
        #230 startFP=1'b0;
        #5000 $stop;
        end
endmodule
```

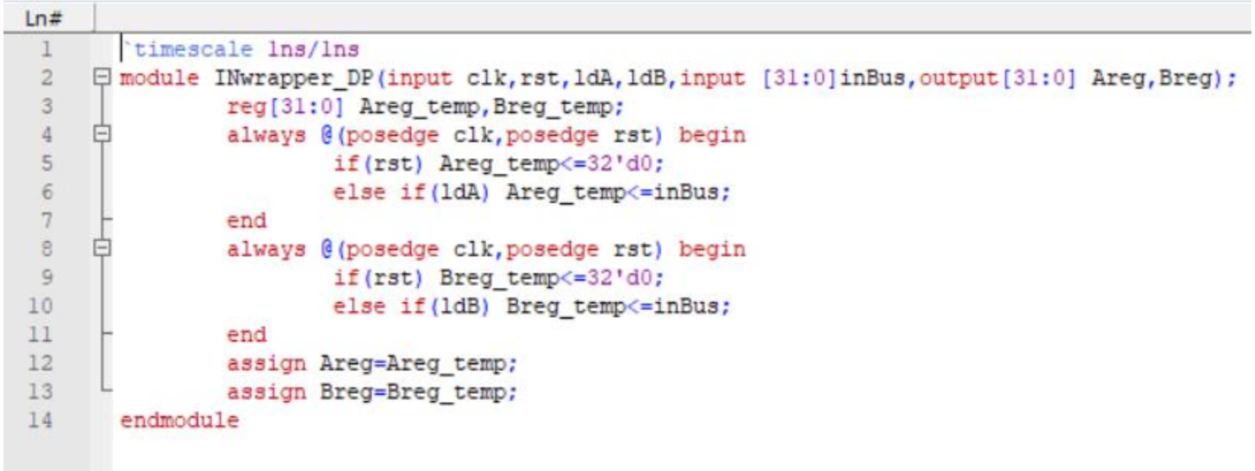
Waveform



F)wrapper

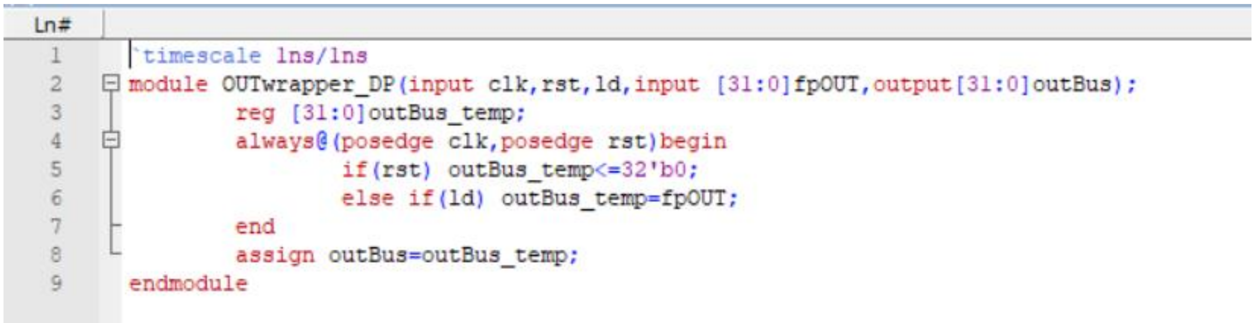
Datapass

inwrapper



```
`timescale 1ns/1ns
module INwrapper_DP(input clk,rst,ldA,ldB,input [31:0]inBus,output[31:0] Areg,Breg);
        reg[31:0] Areg_temp,Breg_temp;
        always @(posedge clk,posedge rst) begin
                if(rst) Areg_temp<=32'd0;
                else if(ldA) Areg_temp<=inBus;
        end
        always @(posedge clk,posedge rst) begin
                if(rst) Breg_temp<=32'd0;
                else if(ldB) Breg_temp<=inBus;
        end
        assign Areg=Areg_temp;
        assign Breg=Breg_temp;
endmodule
```

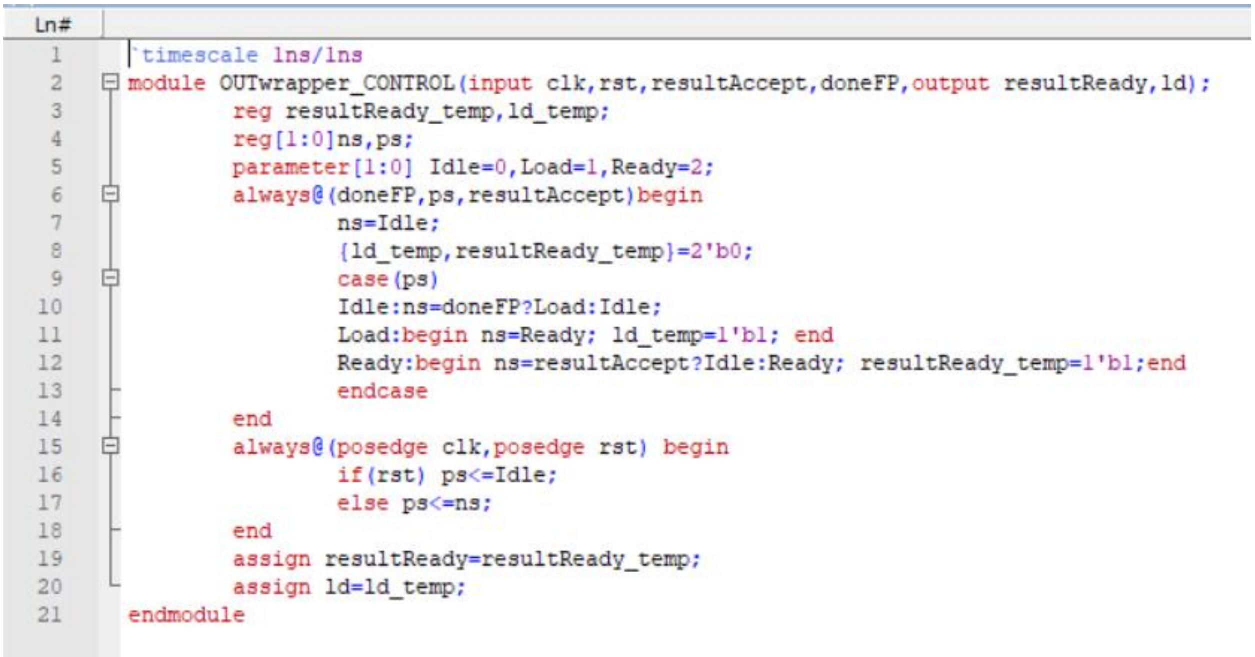
outwrapper



```
`timescale 1ns/1ns
module OUTwrapper_DP(input clk,rst,ld,input [31:0]fpOUT,output[31:0]outBus);
        reg [31:0]outBus_temp;
        always@(posedge clk,posedge rst)begin
                if(rst) outBus_temp<=32'b0;
                else if(ld) outBus_temp=fpOUT;
        end
        assign outBus=outBus_temp;
endmodule
```

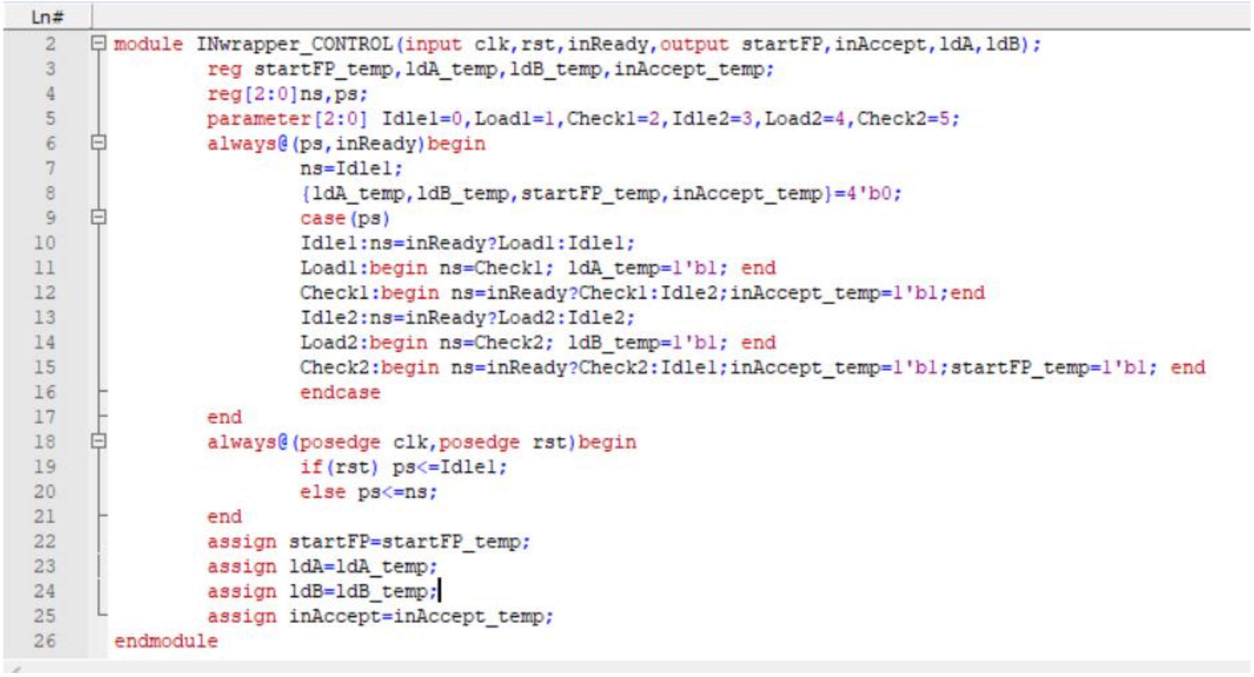
Control

outwrapper



```
`timescale 1ns/1ns
module OUTwrapper_CONTROL(input clk,rst,resultAccept,doneFP,output resultReady,ld);
        reg resultReady_temp,ld_temp;
        reg[1:0]ns,ps;
        parameter[1:0] Idle=0,Load=1,Ready=2;
        always@(doneFP,ps,resultAccept)begin
                ns=Idle;
                {ld_temp,resultReady_temp}=2'b0;
                case(ps)
                Idle:ns=doneFP?Load:Idle;
                Load:begin ns=Ready; ld_temp=1'b1; end
                Ready:begin ns=resultAccept?Idle:Ready; resultReady_temp=1'b1;end
                endcase
        end
        always@(posedge clk,posedge rst) begin
                if(rst) ps<=Idle;
                else ps<=ns;
        end
        assign resultReady=resultReady_temp;
        assign ld=ld_temp;
endmodule
```

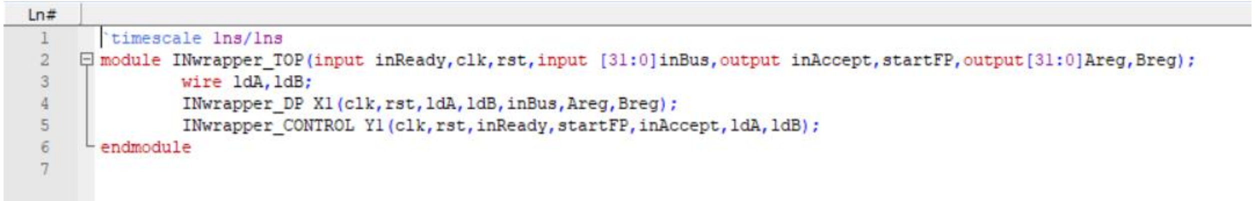
inwrapper



```
module INwrapper_CONTROL(input clk,rst,inReady,output startFP,inAccept,ldA,ldB);
        reg startFP_temp,ldA_temp,ldB_temp,inAccept_temp;
        reg[2:0]ns,ps;
        parameter[2:0] Idle1=0,Load1=1,Check1=2,Idle2=3,Load2=4,Check2=5;
        always@(ps,inReady)begin
                ns=Idle1;
                {ldA_temp,ldB_temp,startFP_temp,inAccept_temp}=4'b0;
                case(ps)
                Idle1:ns=inReady?Load1:Idle1;
                Load1:begin ns=Check1; ldA_temp=1'b1; end
                Check1:begin ns=inReady?Check1:Idle2;inAccept_temp=1'b1;end
                Idle2:ns=inReady?Load2:Idle2;
                Load2:begin ns=Check2; ldB_temp=1'b1; end
                Check2:begin ns=inReady?Check2:Idle1;inAccept_temp=1'b1;startFP_temp=1'b1; end
                endcase
        end
        always@(posedge clk,posedge rst)begin
                if(rst) ps<=Idle1;
                else ps<=ns;
        end
        assign startFP=startFP_temp;
        assign ldA=ldA_temp;
        assign ldB=ldB_temp;
        assign inAccept=inAccept_temp;
endmodule
```

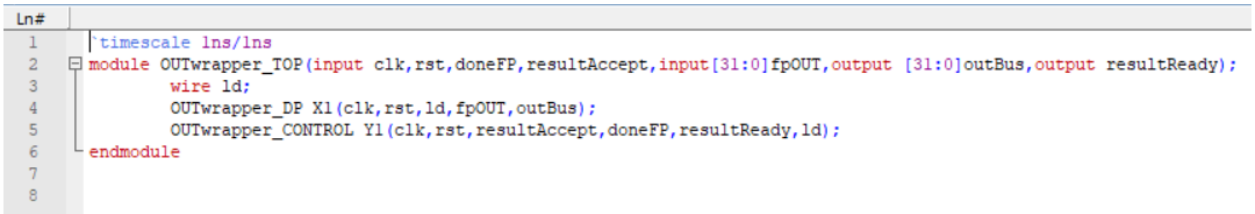
Toplevel

inwrapper



```
`timescale 1ns/1ns
module INwrapper_TOP(input inReady,clk,rst,input [31:0]inBus,output inAccept,startFP,output[31:0]Areg,Breg);
        wire ldA,ldB;
        INwrapper_DP X1(clk,rst,ldA,ldB,inBus,Areg,Breg);
        INwrapper_CONTROL Y1(clk,rst,inReady,startFP,inAccept,ldA,ldB);
endmodule
```

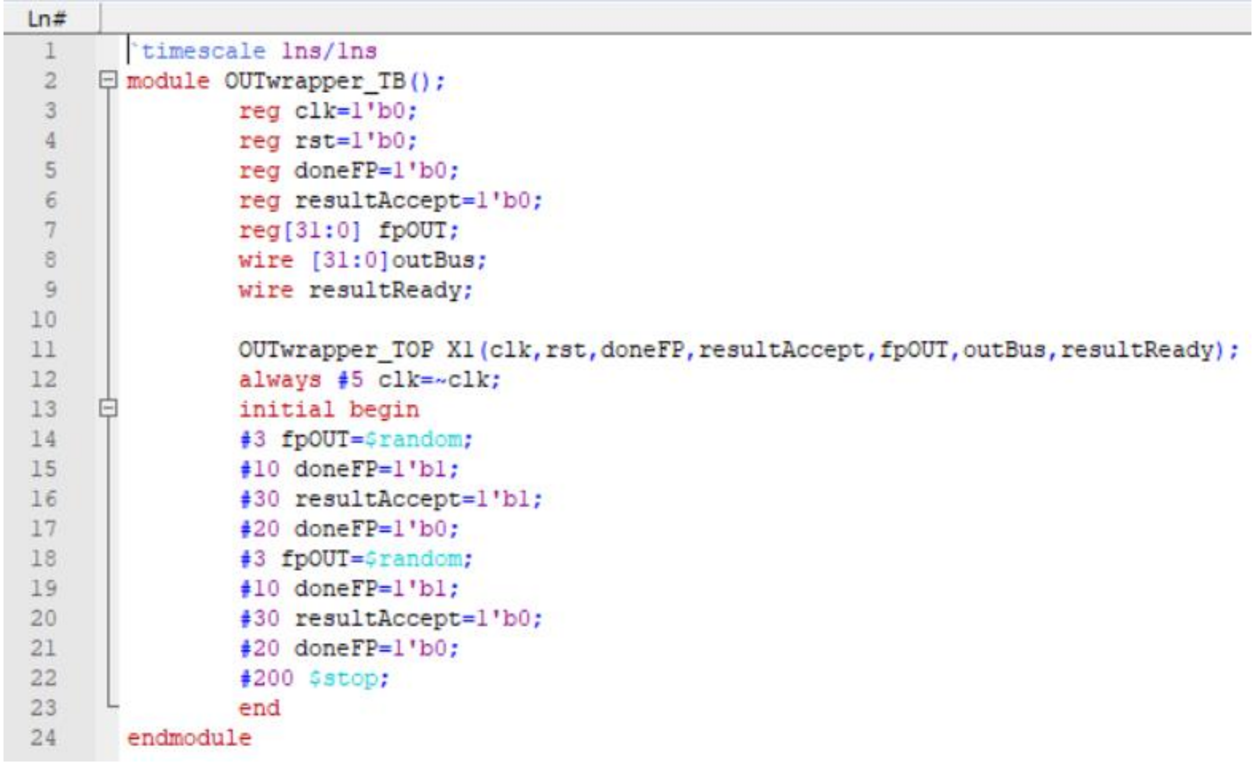
outwrapper



```
`timescale 1ns/1ns
module OUTwrapper_TOP(input clk,rst,doneFP,resultAccept,input[31:0]fpOUT,output [31:0]outBus,output resultReady);
        wire ld;
        OUTwrapper_DP X1(clk,rst,ld,fpOUT,outBus);
        OUTwrapper_CONTROL Y1(clk,rst,resultAccept,doneFP,resultReady,ld);
endmodule
```

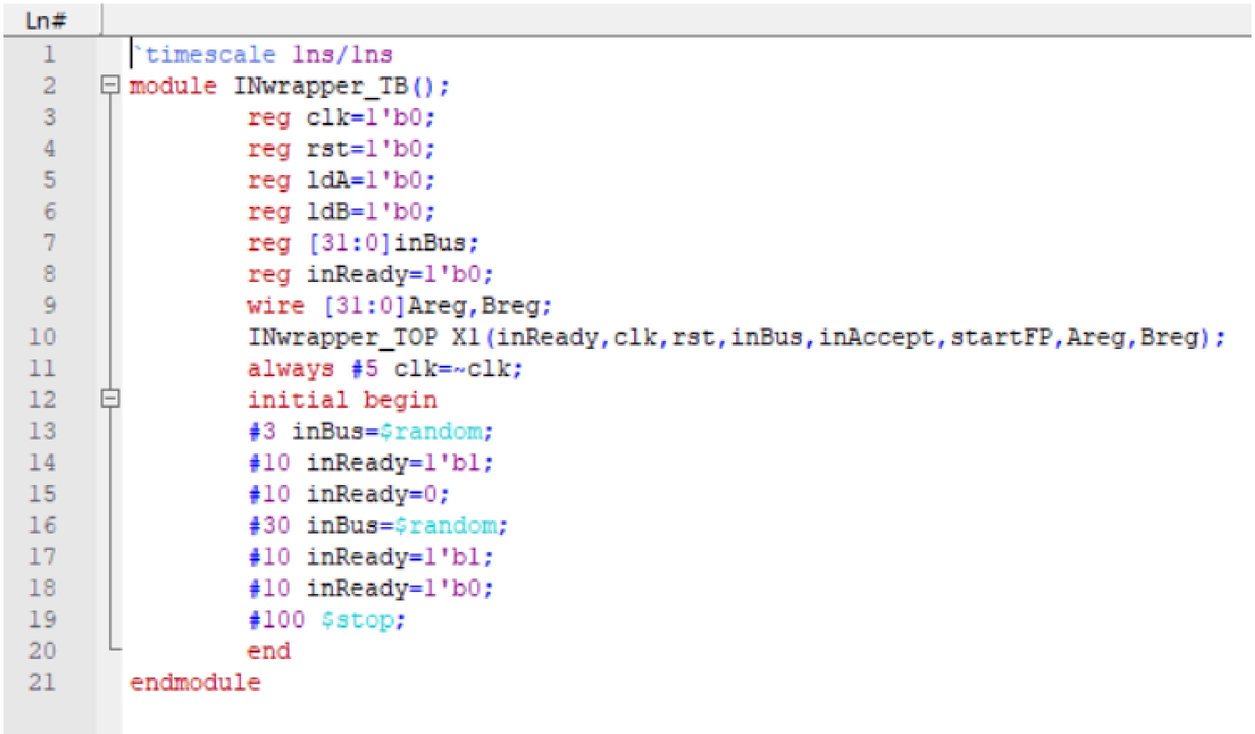
Testbench

outwrapper



```
`timescale 1ns/1ns
module OUTwrapper_TB();
        reg clk=1'b0;
        reg rst=1'b0;
        reg doneFP=1'b0;
        reg resultAccept=1'b0;
        reg[31:0] fpOUT;
        wire [31:0]outBus;
        wire resultReady;

        OUTwrapper_TOP X1(clk,rst,doneFP,resultAccept,fpOUT,outBus,resultReady);
        always #5 clk=~clk;
        initial begin
        #3 fpOUT=$random;
        #10 doneFP=1'b1;
        #30 resultAccept=1'b1;
        #20 doneFP=1'b0;
        #3 fpOUT=$random;
        #10 doneFP=1'b1;
        #30 resultAccept=1'b0;
        #20 doneFP=1'b0;
        #200 $stop;
        end
endmodule
```

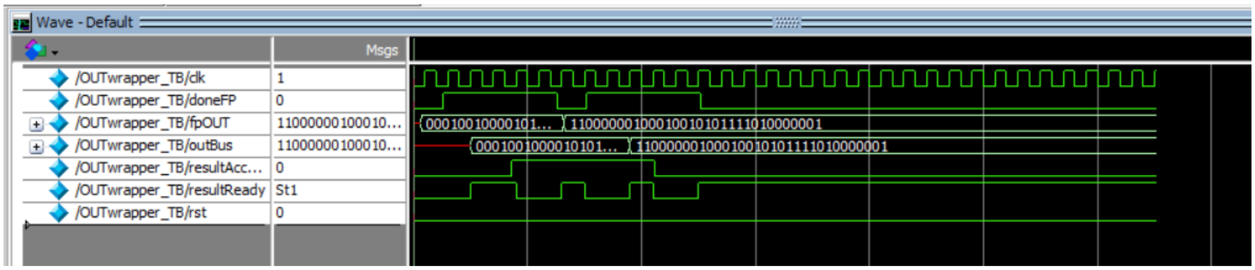
inwrapper



```
`timescale 1ns/1ns
module INwrapper_TB();
        reg clk=1'b0;
        reg rst=1'b0;
        reg ldA=1'b0;
        reg ldB=1'b0;
        reg [31:0]inBus;
        reg inReady=1'b0;
        wire [31:0]Areg,Breg;
        INwrapper_TOP X1(inReady,clk,rst,inBus,inAccept,startFP,Areg,Breg);
        always #5 clk=~clk;
        initial begin
        #3 inBus=$random;
        #10 inReady=1'b1;
        #10 inReady=0;
        #30 inBus=$random;
        #10 inReady=1'b1;
        #10 inReady=1'b0;
        #100 $stop;
        end
endmodule
```

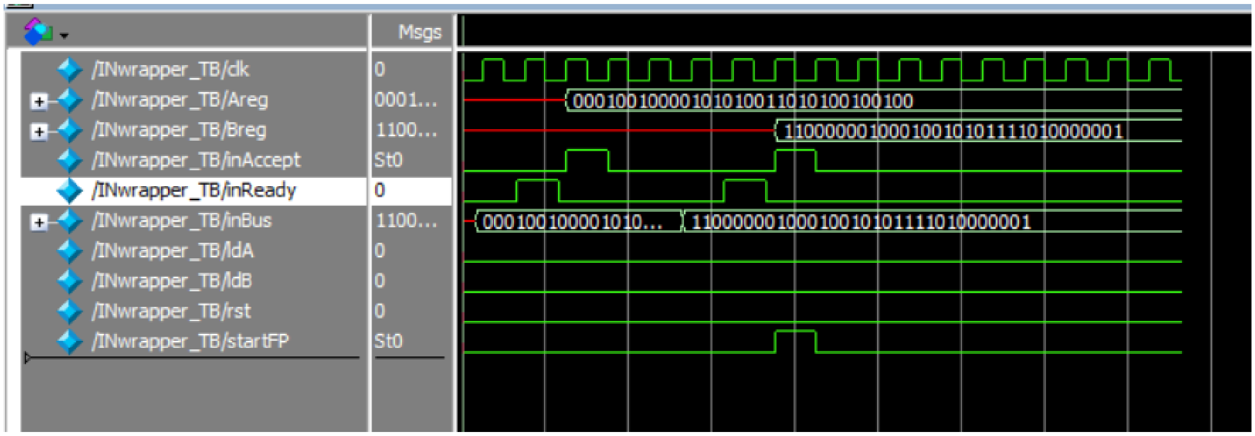
Waveform

outwrapper



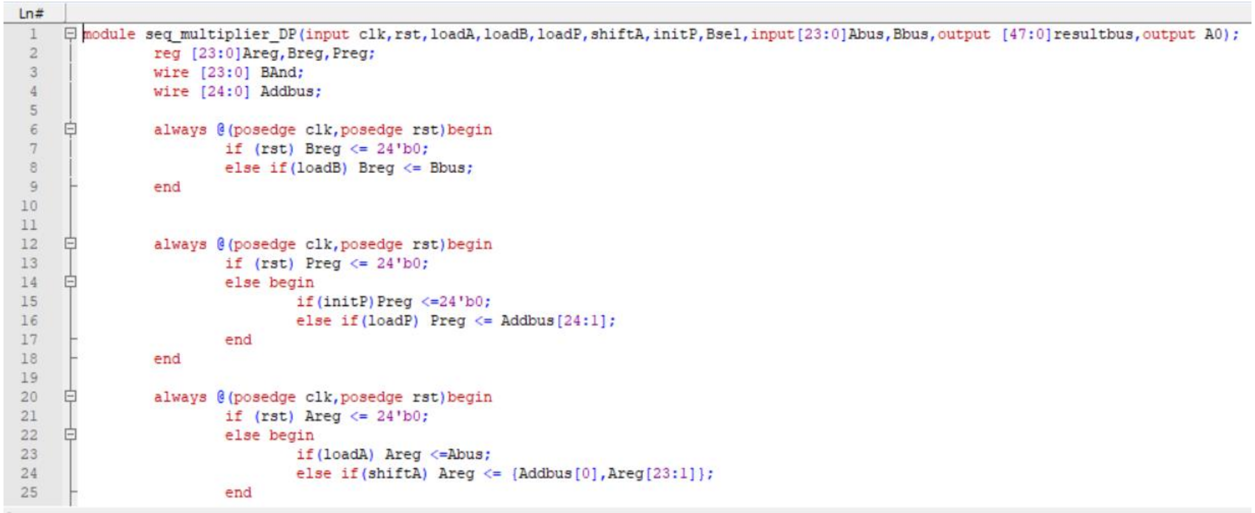
```
/OUTwrapper_TB/clk           1
/OUTwrapper_TB/doneFP        0
/OUTwrapper_TB/fpOUT         11000000100010...
/OUTwrapper_TB/outBus        11000000100010...
/OUTwrapper_TB/resultAcc...  0
/OUTwrapper_TB/resultReady   St1
/OUTwrapper_TB/rst           0
```

inwrapper

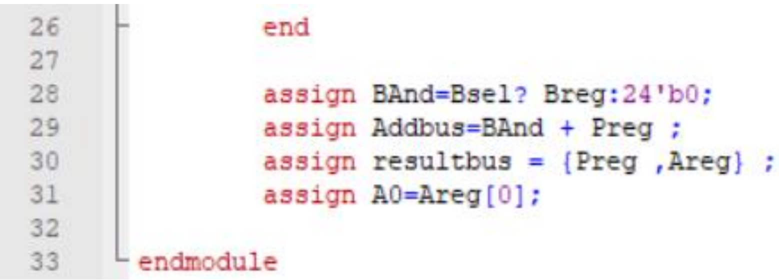


G) sequential

Datapass

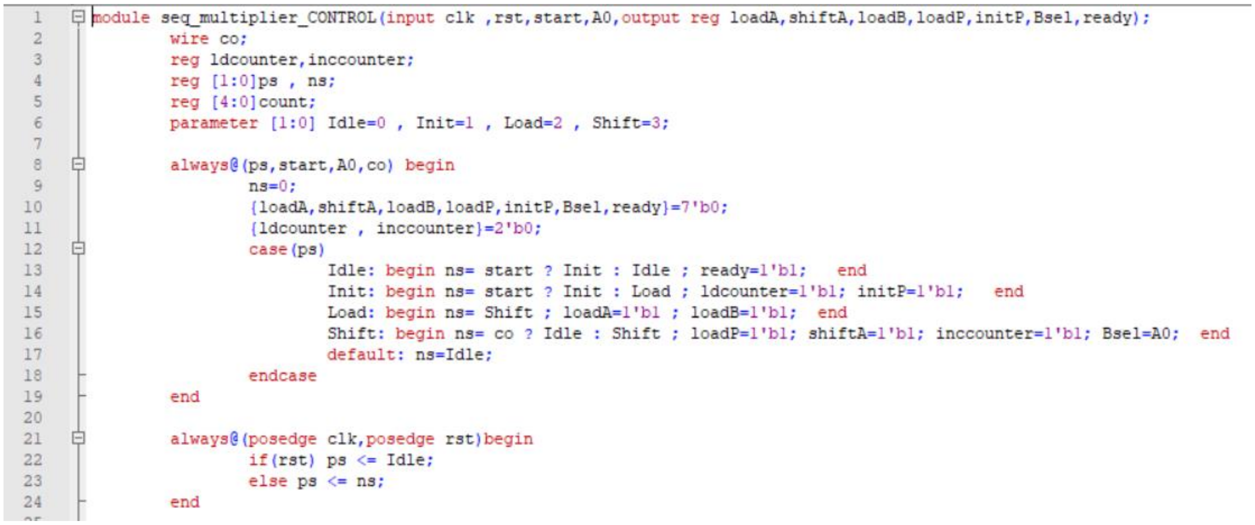


```
module seq_multiplier_DP(input clk,rst,loadA,loadB,loadP,shiftA,initP,Bsel,input[23:0]Abus,Bbus,output [47:0]resultbus,output A0);
        reg [23:0]Areg,Breg,Preg;
        wire [23:0] BAnd;
        wire [24:0] Addbus;

        always @(posedge clk,posedge rst)begin
                if (rst) Breg <= 24'b0;
                else if(loadB) Breg <= Bbus;
        end


        always @(posedge clk,posedge rst)begin
                if (rst) Preg <= 24'b0;
                else begin
                        if(initP)Preg <=24'b0;
                        else if(loadP) Preg <= Addbus[24:1];
                end
        end

        always @(posedge clk,posedge rst)begin
                if (rst) Areg <= 24'b0;
                else begin
                        if(loadA) Areg <=Abus;
                        else if(shiftA) Areg <= {Addbus[0],Areg[23:1]};
                end
```

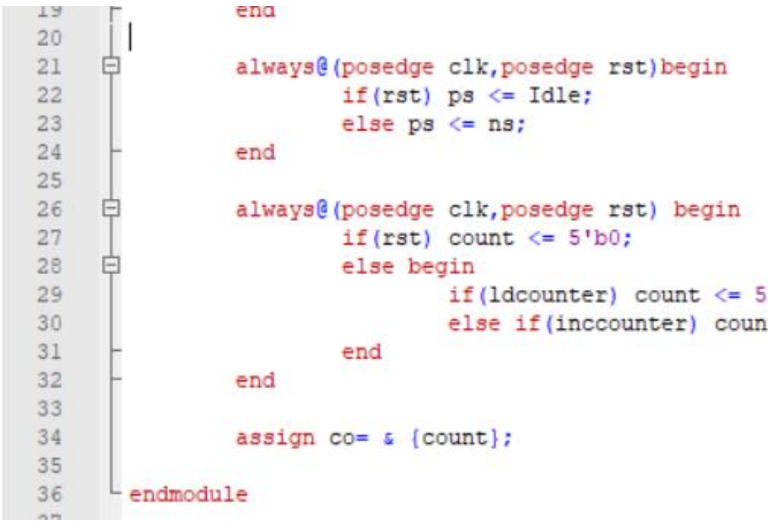


```
                end

                assign BAnd=Bsel? Breg:24'b0;
                assign Addbus=BAnd + Preg ;
                assign resultbus = {Preg ,Areg} ;
                assign A0=Areg[0];

endmodule
```

Control

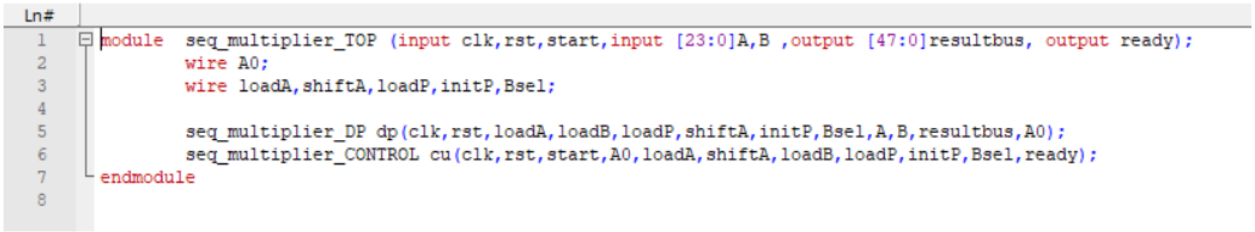


```
module seq_multiplier_CONTROL(input clk ,rst,start,A0,output reg loadA,shiftA,loadB,loadP,initP,Bsel,ready);
        wire co;
        reg ldcounter,inccounter;
        reg [1:0]ps , ns;
        reg [4:0]count;
        parameter [1:0] Idle=0 , Init=1 , Load=2 , Shift=3;

        always@(ps,start,A0,co) begin
                ns=0;
                {loadA,shiftA,loadB,loadP,initP,Bsel,ready}=7'b0;
                {ldcounter , inccounter}=2'b0;
                case(ps)
                        Idle: begin ns= start ? Init : Idle ; ready=1'b1;   end
                        Init: begin ns= start ? Init : Load ; ldcounter=1'b1; initP=1'b1;   end
                        Load: begin ns= Shift ; loadA=1'b1 ; loadB=1'b1;  end
                        Shift: begin ns= co ? Idle : Shift ; loadP=1'b1; shiftA=1'b1; inccounter=1'b1; Bsel=A0;  end
                        default: ns=Idle;
                endcase
        end

        always@(posedge clk,posedge rst)begin
                if(rst) ps <= Idle;
                else ps <= ns;
        end
```



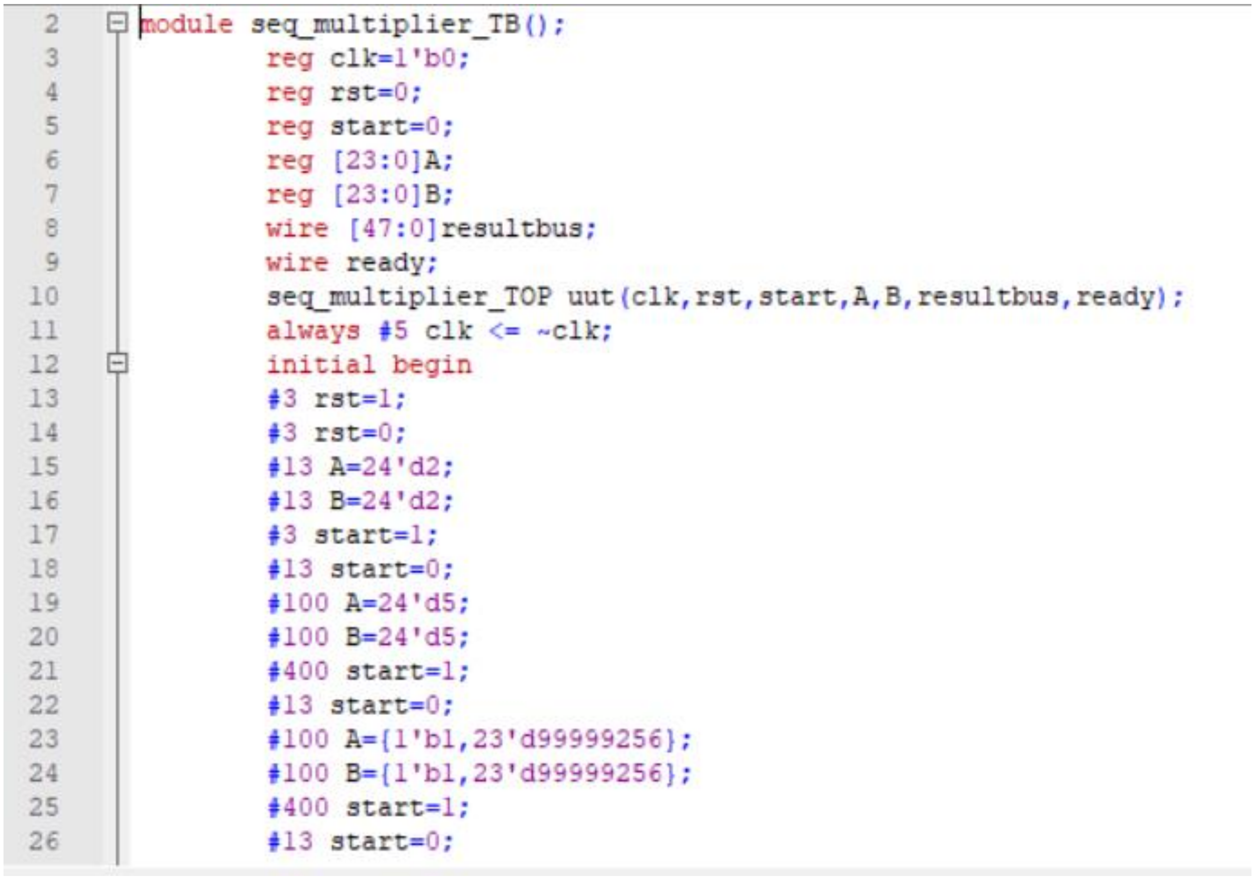
```
        end

        always@(posedge clk,posedge rst)begin
                if(rst) ps <= Idle;
                else ps <= ns;
        end

        always@(posedge clk,posedge rst) begin
                if(rst) count <= 5'b0;
                else begin
                        if(ldcounter) count <= 5'd8;
                        else if(inccounter) count <=count+1;
                end
        end

        assign co= & {count};

endmodule
```

Toplevel



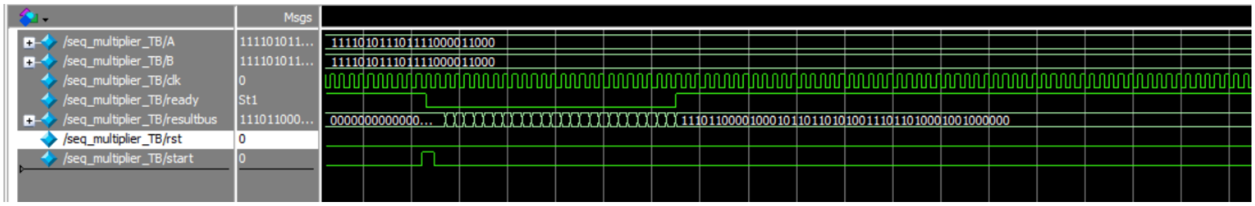
```
module  seq_multiplier_TOP (input clk,rst,start,input [23:0]A,B ,output [47:0]resultbus, output ready);
        wire A0;
        wire loadA,shiftA,loadP,initP,Bsel;

        seq_multiplier_DP dp(clk,rst,loadA,loadB,loadP,shiftA,initP,Bsel,A,B,resultbus,A0);
        seq_multiplier_CONTROL cu(clk,rst,start,A0,loadA,shiftA,loadB,loadP,initP,Bsel,ready);
endmodule
```

Testbench



```
module seq_multiplier_TB();
        reg clk=1'b0;
        reg rst=0;
        reg start=0;
        reg [23:0]A;
        reg [23:0]B;
        wire [47:0]resultbus;
        wire ready;
        seq_multiplier_TOP uut(clk,rst,start,A,B,resultbus,ready);
        always #5 clk <= ~clk;
        initial begin
        #3 rst=1;
        #3 rst=0;
        #13 A=24'd2;
        #13 B=24'd2;
        #3 start=1;
        #13 start=0;
        #100 A=24'd5;
        #100 B=24'd5;
        #400 start=1;
        #13 start=0;
        #100 A={1'b1,23'd99999256};
        #100 B={1'b1,23'd99999256};
        #400 start=1;
        #13 start=0;
```

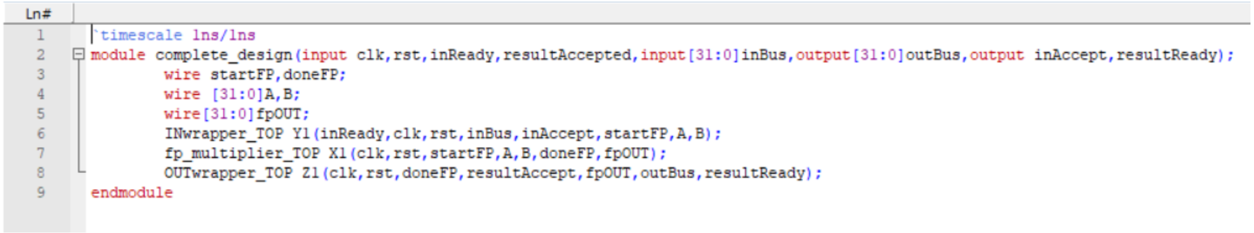
Waveform



```
                            Msgs
/seq_multiplier_TB/A          111101011...   111101011101111000011000
/seq_multiplier_TB/B          111101011...   111101011101111000011000
/seq_multiplier_TB/clk        0
/seq_multiplier_TB/ready      St1
/seq_multiplier_TB/resultbus  111011000...   0000000000000...   111011000010001011011010100111011010001001000000
/seq_multiplier_TB/rst        0
/seq_multiplier_TB/start      0
```

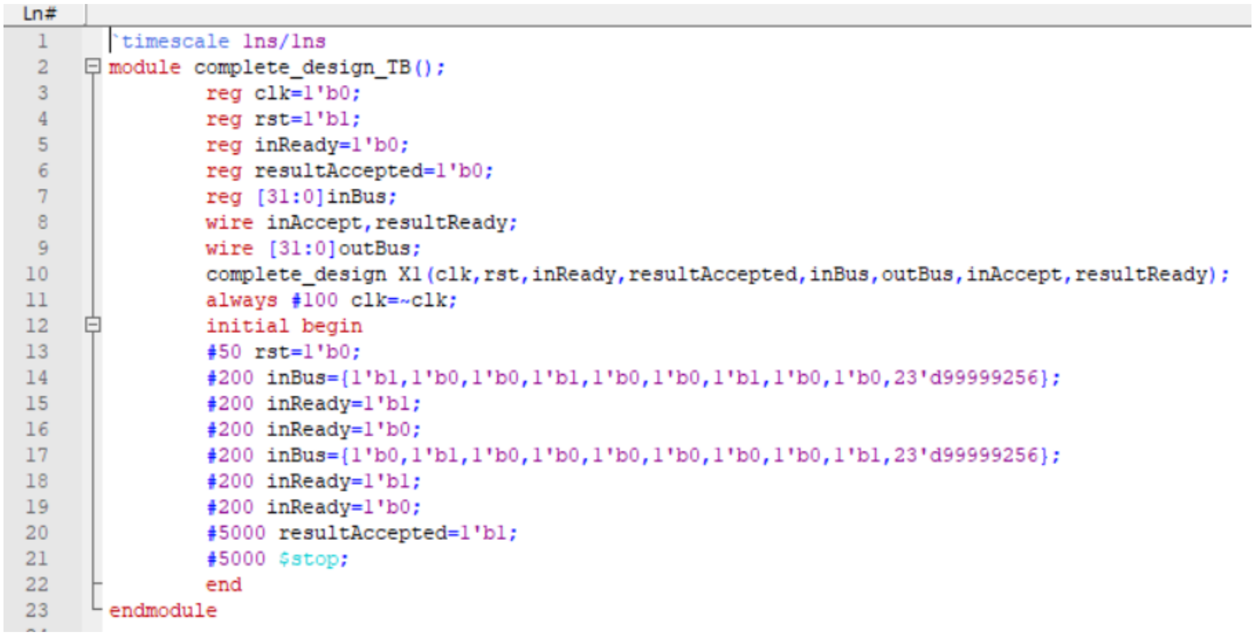
H) complete design

Toplevel



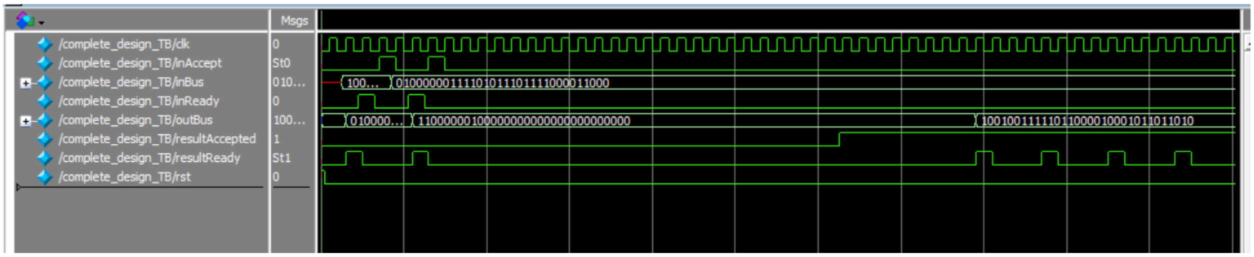
```
`timescale 1ns/1ns
module complete_design(input clk,rst,inReady,resultAccepted,input[31:0]inBus,output[31:0]outBus,output inAccept,resultReady);
        wire startFP,doneFP;
        wire [31:0]A,B;
        wire[31:0]fpOUT;
        INwrapper_TOP Y1(inReady,clk,rst,inBus,inAccept,startFP,A,B);
        fp_multiplier_TOP X1(clk,rst,startFP,A,B,doneFP,fpOUT);
        OUTwrapper_TOP Z1(clk,rst,doneFP,resultAccept,fpOUT,outBus,resultReady);
endmodule
```

Testbench



```
`timescale 1ns/1ns
module complete_design_TB();
        reg clk=1'b0;
        reg rst=1'b1;
        reg inReady=1'b0;
        reg resultAccepted=1'b0;
        reg [31:0]inBus;
        wire inAccept,resultReady;
        wire [31:0]outBus;
        complete_design X1(clk,rst,inReady,resultAccepted,inBus,outBus,inAccept,resultReady);
        always #100 clk=~clk;
        initial begin
        #50 rst=1'b0;
        #200 inBus={1'b1,1'b0,1'b0,1'b1,1'b0,1'b0,1'b1,1'b0,1'b0,23'd99999256};
        #200 inReady=1'b1;
        #200 inReady=1'b0;
        #200 inBus={1'b0,1'b1,1'b0,1'b0,1'b0,1'b0,1'b0,1'b0,1'b1,23'd99999256};
        #200 inReady=1'b1;
        #200 inReady=1'b0;
        #5000 resultAccepted=1'b1;
        #5000 $stop;
        end
endmodule
```

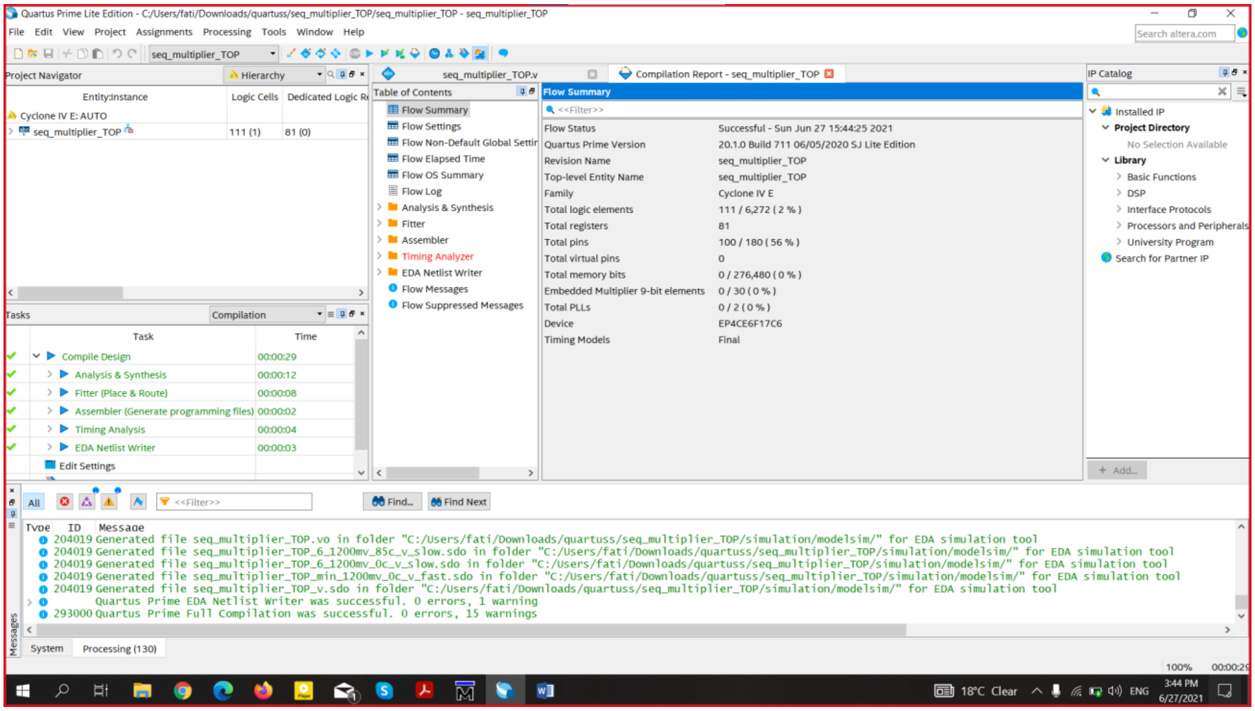
Waveform



Msgs
/complete_design_TB/clk 0
/complete_design_TB/inAccept St0
/complete_design_TB/inBus 010...
/complete_design_TB/inReady 0
/complete_design_TB/outBus 100...
/complete_design_TB/resultAccepted 1
/complete_design_TB/resultReady St1
/complete_design_TB/rst 0
100...
01000000111101011101111000011000
010000...
11000000100000000000000000000000
10010011111011000010001011011010

i)

Sequential



Quartus Prime Lite Edition - C:/Users/fati/Downloads/quartuss/seq_multiplier_TOP/seq_multiplier_TOP - seq_multiplier_TOP

Entity:Instance	Logic Cells	Dedicated Logic Re
Cyclone IV E: AUTO		
seq_multiplier_TOP	111 (1)	81 (0)

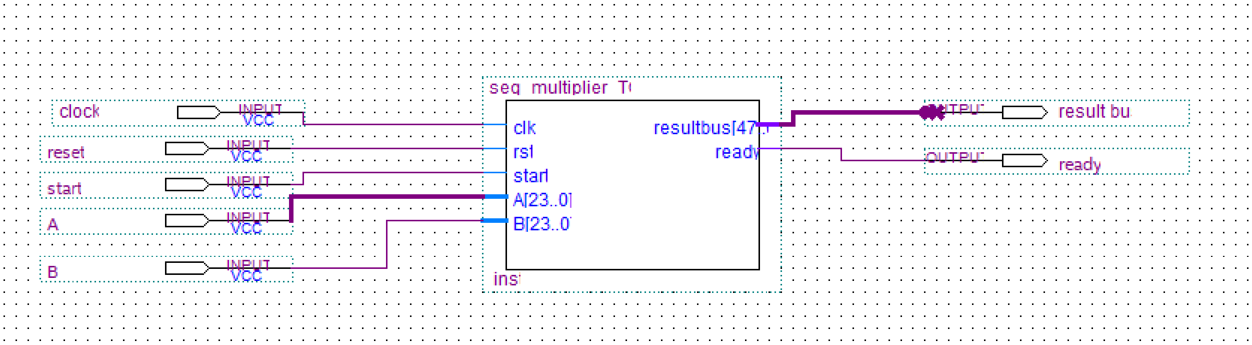
Task	Time
Compile Design	00:00:29
Analysis & Synthesis	00:00:12
Fitter (Place & Route)	00:00:08
Assembler (Generate programming files)	00:00:02
Timing Analysis	00:00:04
EDA Netlist Writer	00:00:03
Edit Settings	

Compilation Report - seq_multiplier_TOP

Flow Summary

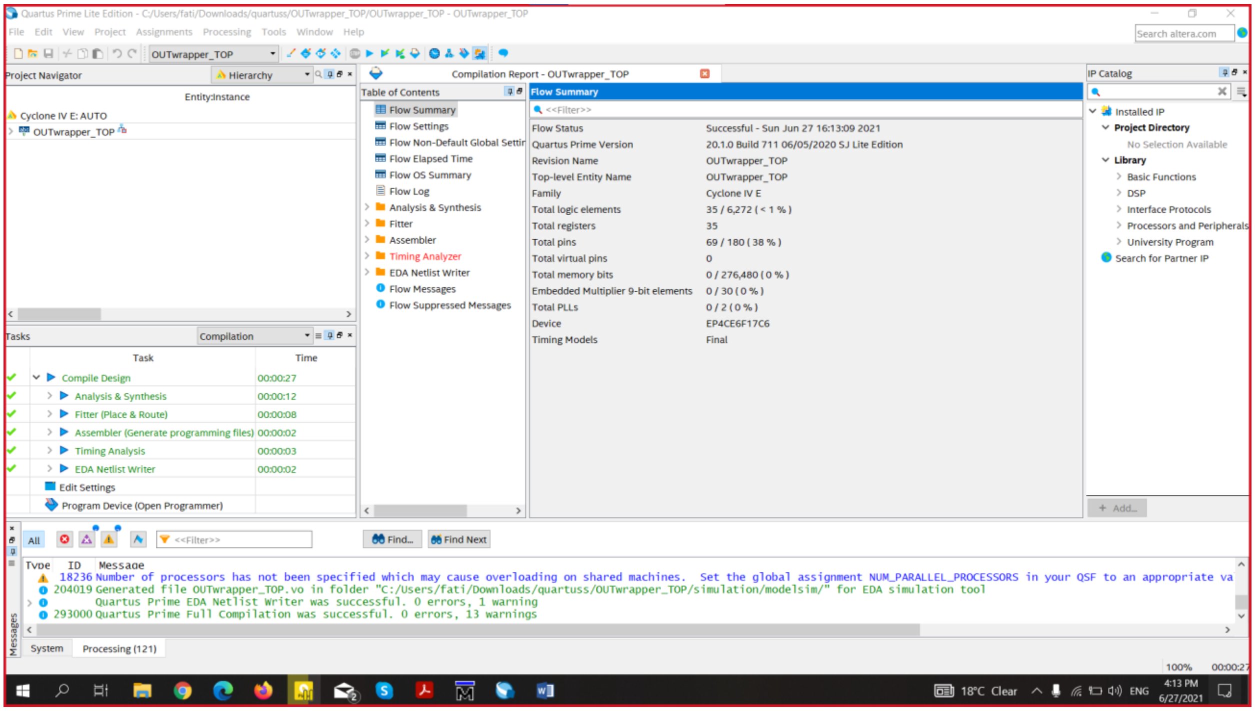
Flow Status	Successful - Sun Jun 27 15:44:25 2021
Quartus Prime Version	20.1.0 Build 711 06/05/2020 SJ Lite Edition
Revision Name	seq_multiplier_TOP
Top-level Entity Name	seq_multiplier_TOP
Family	Cyclone IV E
Total logic elements	111 / 6,272 (2 %)
Total registers	81
Total pins	100 / 180 (56 %)
Total virtual pins	0
Total memory bits	0 / 276,480 (0 %)
Embedded Multiplier 9-bit elements	0 / 30 (0 %)
Total PLLs	0 / 2 (0 %)
Device	EP4CE6F17C6
Timing Models	Final

```
Type  ID     Message
      204019 Generated file seq_multiplier_TOP.vo in folder "C:/Users/fati/Downloads/quartuss/seq_multiplier_TOP/simulation/modelsim/" for EDA simulation tool
      204019 Generated file seq_multiplier_TOP_6_1200mv_85c_v_slow.sdo in folder "C:/Users/fati/Downloads/quartuss/seq_multiplier_TOP/simulation/modelsim/" for EDA simulation tool
      204019 Generated file seq_multiplier_TOP_6_1200mv_0c_v_slow.sdo in folder "C:/Users/fati/Downloads/quartuss/seq_multiplier_TOP/simulation/modelsim/" for EDA simulation tool
      204019 Generated file seq_multiplier_TOP_min_1200mv_0c_v_fast.sdo in folder "C:/Users/fati/Downloads/quartuss/seq_multiplier_TOP/simulation/modelsim/" for EDA simulation tool
      204019 Generated file seq_multiplier_TOP_v.sdo in folder "C:/Users/fati/Downloads/quartuss/seq_multiplier_TOP/simulation/modelsim/" for EDA simulation tool
             Quartus Prime EDA Netlist Writer was successful. 0 errors, 1 warning
      293000 Quartus Prime Full Compilation was successful. 0 errors, 15 warnings
```



j)

Outwrapper



Quartus Prime Lite Edition - C:/Users/fati/Downloads/quartuss/OUTwrapper_TOP/OUTwrapper_TOP - OUTwrapper_TOP

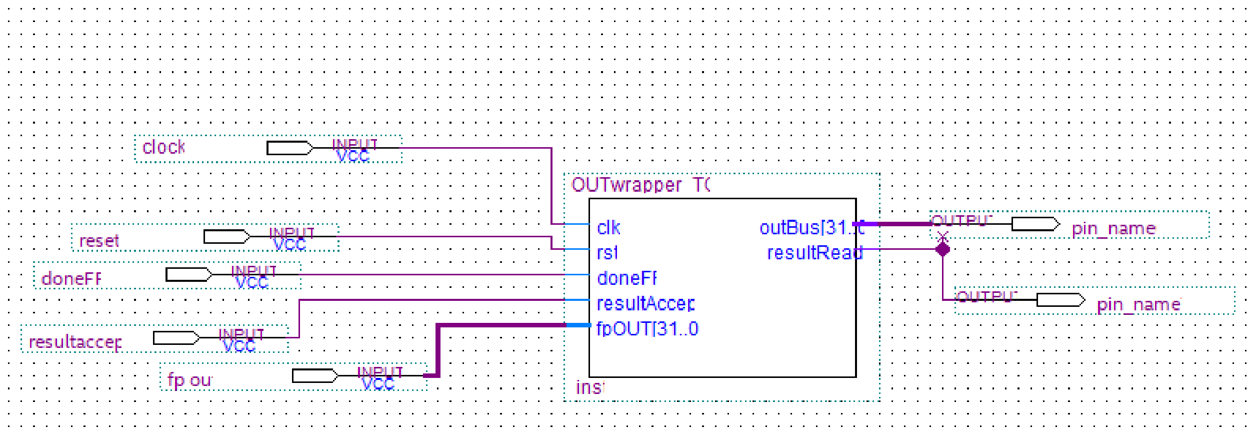
Flow Summary

Flow Status	Successful - Sun Jun 27 16:13:09 2021
Quartus Prime Version	20.1.0 Build 711 06/05/2020 SJ Lite Edition
Revision Name	OUTwrapper_TOP
Top-level Entity Name	OUTwrapper_TOP
Family	Cyclone IV E
Total logic elements	35 / 6,272 (< 1 %)
Total registers	35
Total pins	69 / 180 (38 %)
Total virtual pins	0
Total memory bits	0 / 276,480 (0 %)
Embedded Multiplier 9-bit elements	0 / 30 (0 %)
Total PLLs	0 / 2 (0 %)
Device	EP4CE6F17C6
Timing Models	Final

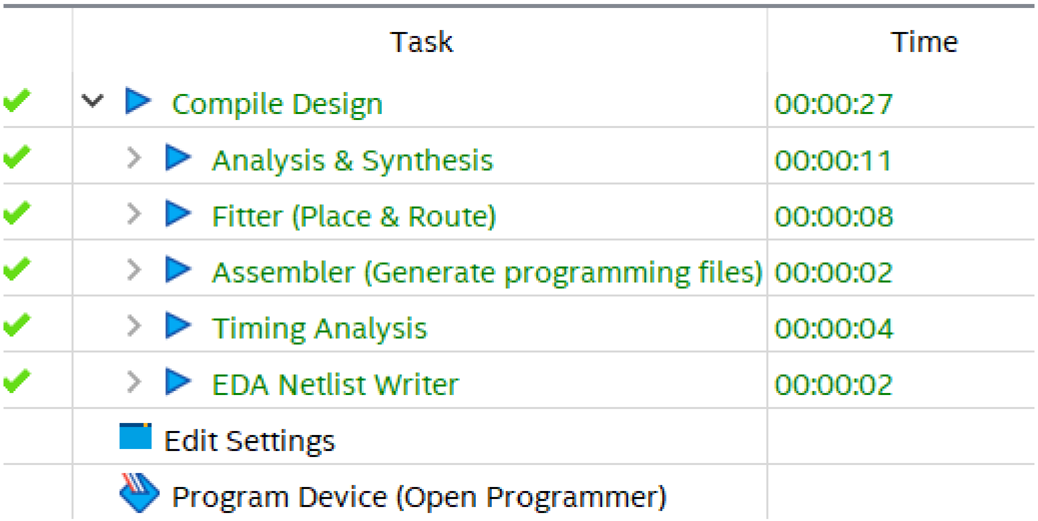
Tasks — Compilation

	Task	Time
✓	Compile Design	00:00:27
✓	Analysis & Synthesis	00:00:12
✓	Fitter (Place & Route)	00:00:08
✓	Assembler (Generate programming files)	00:00:02
✓	Timing Analysis	00:00:03
✓	EDA Netlist Writer	00:00:02
	Edit Settings	
	Program Device (Open Programmer)	

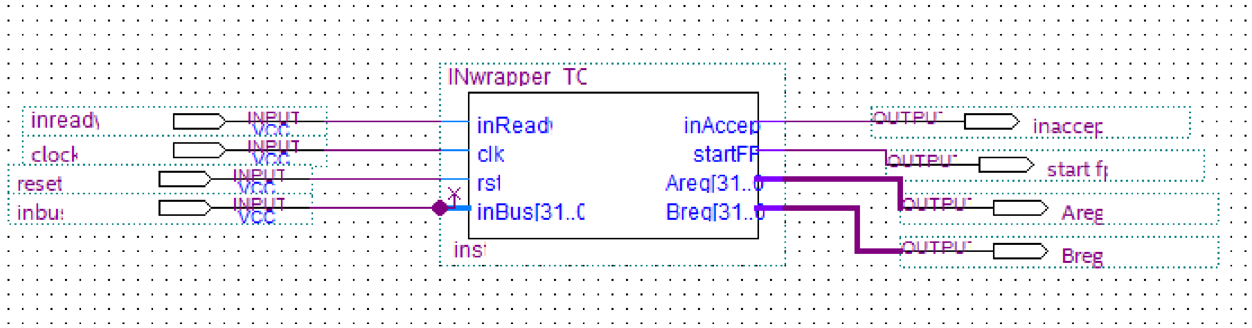
```
Type  ID    Message
  ⚠  18236 Number of processors has not been specified which may cause overloading on shared machines.  Set the global assignment NUM_PARALLEL_PROCESSORS in your QSF to an appropriate va
  ●  204019 Generated file OUTwrapper_TOP.vo in folder "C:/Users/fati/Downloads/quartuss/OUTwrapper_TOP/simulation/modelsim/" for EDA simulation tool
> ●         Quartus Prime EDA Netlist Writer was successful. 0 errors, 1 warning
  ●  293000 Quartus Prime Full Compilation was successful. 0 errors, 13 warnings
```



Inwrapper

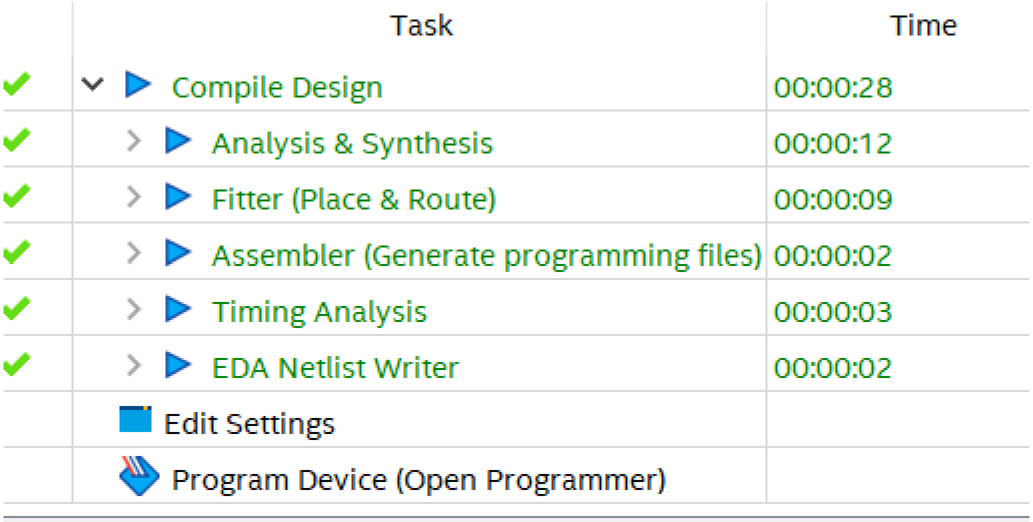


	Task	Time
✓	Compile Design	00:00:27
✓	Analysis & Synthesis	00:00:11
✓	Fitter (Place & Route)	00:00:08
✓	Assembler (Generate programming files)	00:00:02
✓	Timing Analysis	00:00:04
✓	EDA Netlist Writer	00:00:02
	Edit Settings	
	Program Device (Open Programmer)	

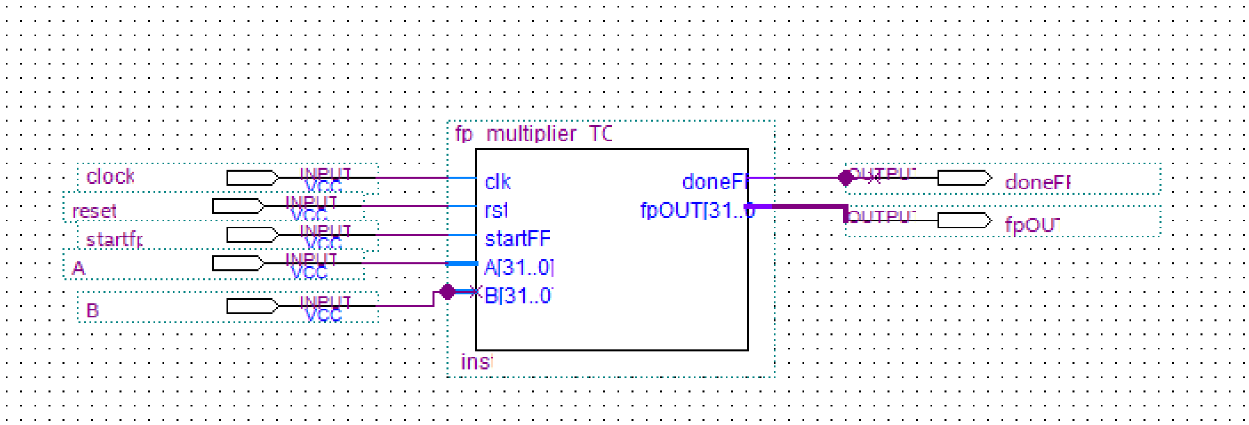


k)

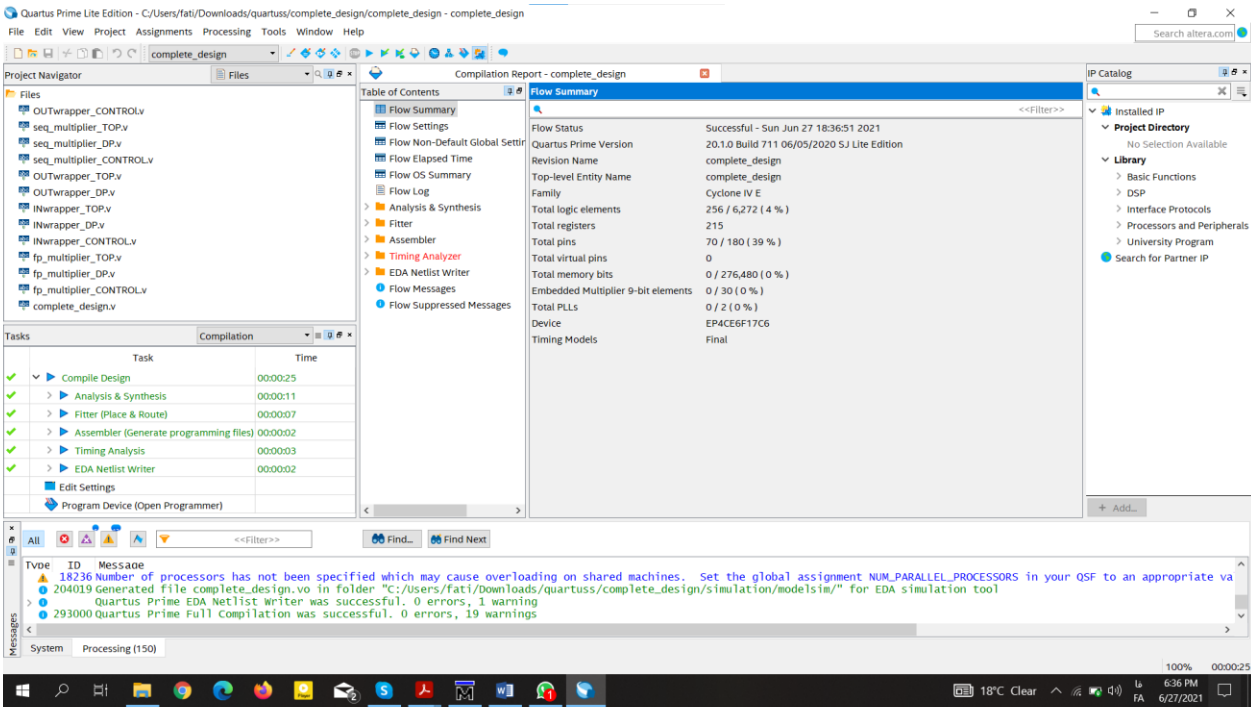
Fp



	Task	Time
✓	Compile Design	00:00:28
✓	Analysis & Synthesis	00:00:12
✓	Fitter (Place & Route)	00:00:09
✓	Assembler (Generate programming files)	00:00:02
✓	Timing Analysis	00:00:03
✓	EDA Netlist Writer	00:00:02
	Edit Settings	
	Program Device (Open Programmer)	



Complete design



Quartus Prime Lite Edition - C:/Users/fati/Downloads/quartuss/complete_design/complete_design - complete_design

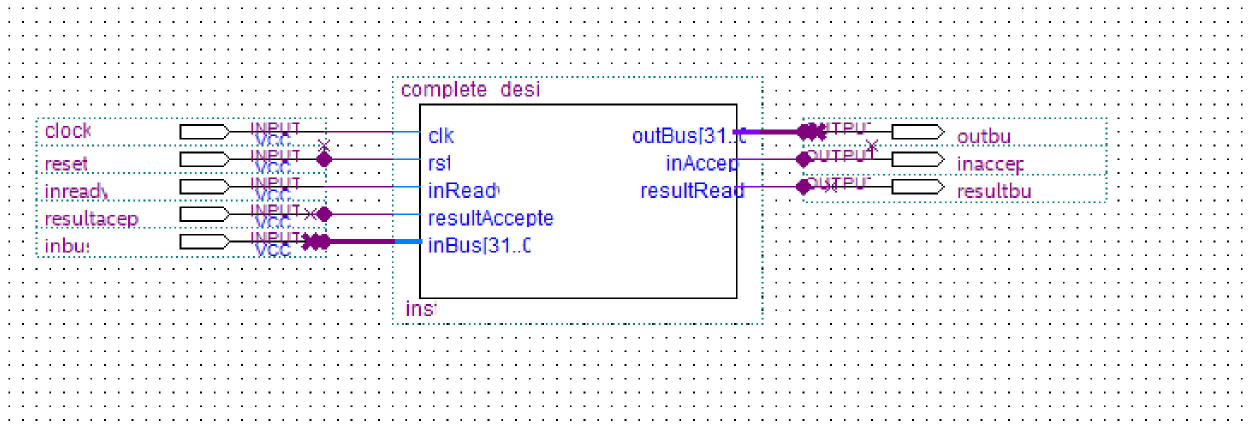
Tasks — Compilation

Task	Time
✓ Compile Design	00:00:25
✓ Analysis & Synthesis	00:00:11
✓ Fitter (Place & Route)	00:00:07
✓ Assembler (Generate programming files)	00:00:02
✓ Timing Analysis	00:00:03
✓ EDA Netlist Writer	00:00:02
Edit Settings	
Program Device (Open Programmer)	

Flow Summary

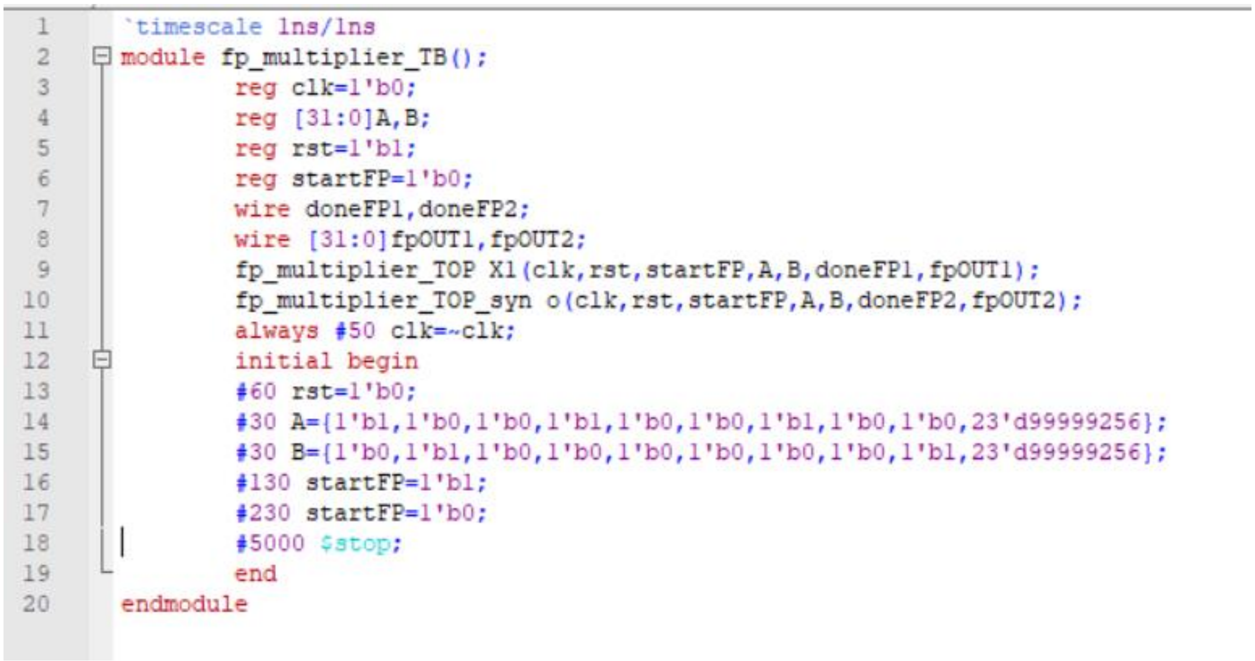
Flow Status	Successful - Sun Jun 27 18:36:51 2021
Quartus Prime Version	20.1.0 Build 711 06/05/2020 SJ Lite Edition
Revision Name	complete_design
Top-level Entity Name	complete_design
Family	Cyclone IV E
Total logic elements	256 / 6,272 (4 %)
Total registers	215
Total pins	70 / 180 (39 %)
Total virtual pins	0
Total memory bits	0 / 276,480 (0 %)
Embedded Multiplier 9-bit elements	0 / 30 (0 %)
Total PLLs	0 / 2 (0 %)
Device	EP4CE6F17C6
Timing Models	Final

```
Type   ID   Message
  18236 Number of processors has not been specified which may cause overloading on shared machines.  Set the global assignment NUM_PARALLEL_PROCESSORS in your QSF to an appropriate va
 204019 Generated file complete_design.vo in folder "C:/Users/fati/Downloads/quartuss/complete_design/simulation/modelsim/" for EDA simulation tool
        Quartus Prime EDA Netlist Writer was successful. 0 errors, 1 warning
 293000 Quartus Prime Full Compilation was successful. 0 errors, 19 warnings
```

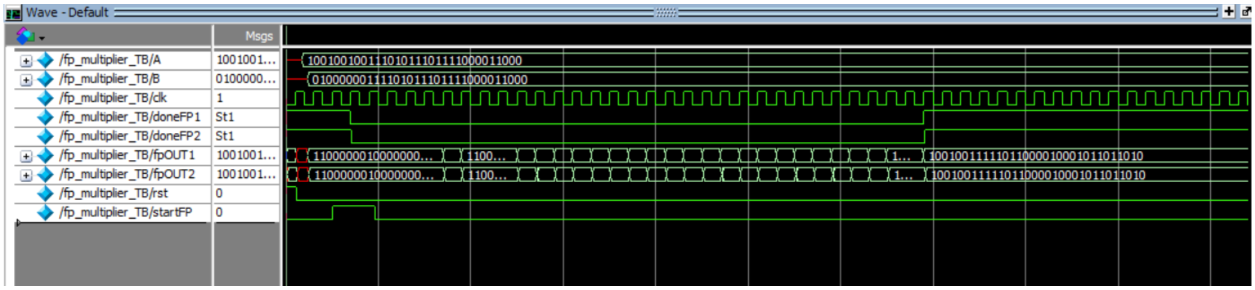


L)

Fp multiplier

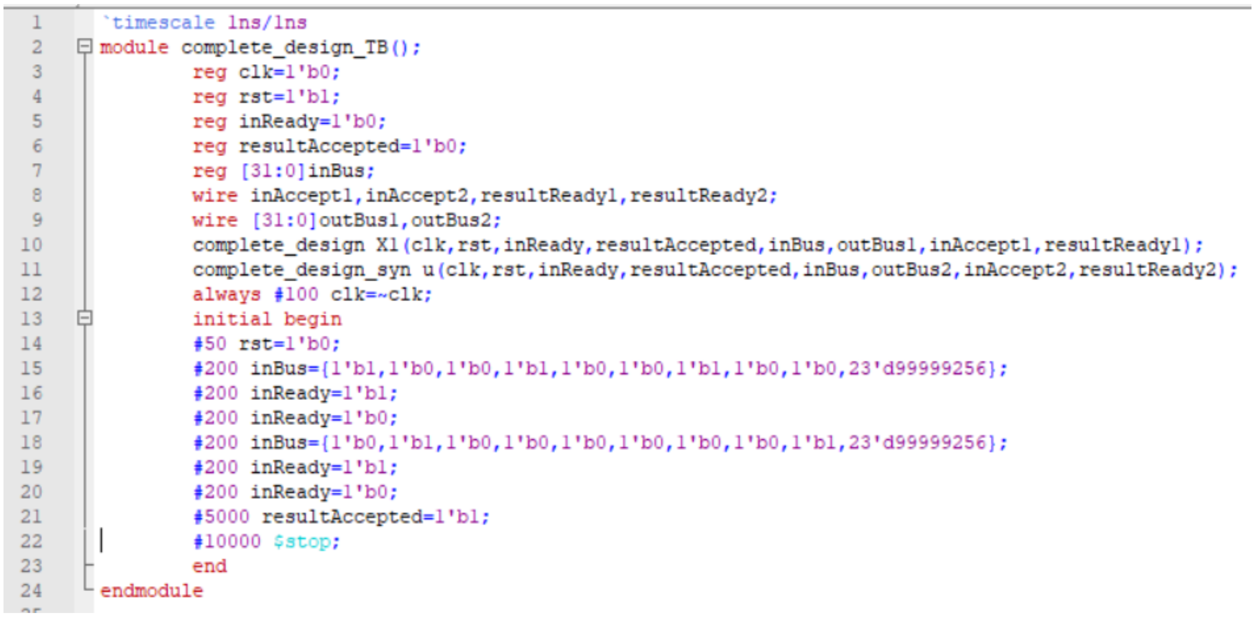


```verilog
`timescale 1ns/1ns
module fp_multiplier_TB();
        reg clk=1'b0;
        reg [31:0]A,B;
        reg rst=1'b1;
        reg startFP=1'b0;
        wire doneFP1,doneFP2;
        wire [31:0]fpOUT1,fpOUT2;
        fp_multiplier_TOP X1(clk,rst,startFP,A,B,doneFP1,fpOUT1);
        fp_multiplier_TOP_syn o(clk,rst,startFP,A,B,doneFP2,fpOUT2);
        always #50 clk=~clk;
        initial begin
        #60 rst=1'b0;
        #30 A={1'b1,1'b0,1'b0,1'b1,1'b0,1'b0,1'b1,1'b0,1'b0,23'd99999256};
        #30 B={1'b0,1'b1,1'b0,1'b0,1'b0,1'b0,1'b0,1'b0,1'b1,23'd99999256};
        #130 startFP=1'b1;
        #230 startFP=1'b0;
        #5000 $stop;
        end
endmodule
```

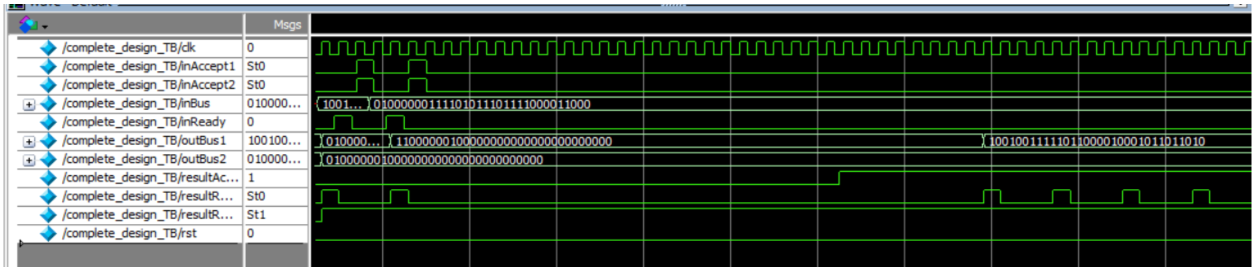


```
/fp_multiplier_TB/A         1001001...   10010010011101011101111000011000
/fp_multiplier_TB/B         0100000...   01000000111101011101111000011000
/fp_multiplier_TB/clk       1
/fp_multiplier_TB/doneFP1   St1
/fp_multiplier_TB/doneFP2   St1
/fp_multiplier_TB/fpOUT1    1001001...   1100000010000000...  1100...  1...  10010011111011000010001011011010
/fp_multiplier_TB/fpOUT2    1001001...   1100000010000000...  1100...  1...  10010011111011000010001011011010
/fp_multiplier_TB/rst       0
/fp_multiplier_TB/startFP   0
```

Complete design

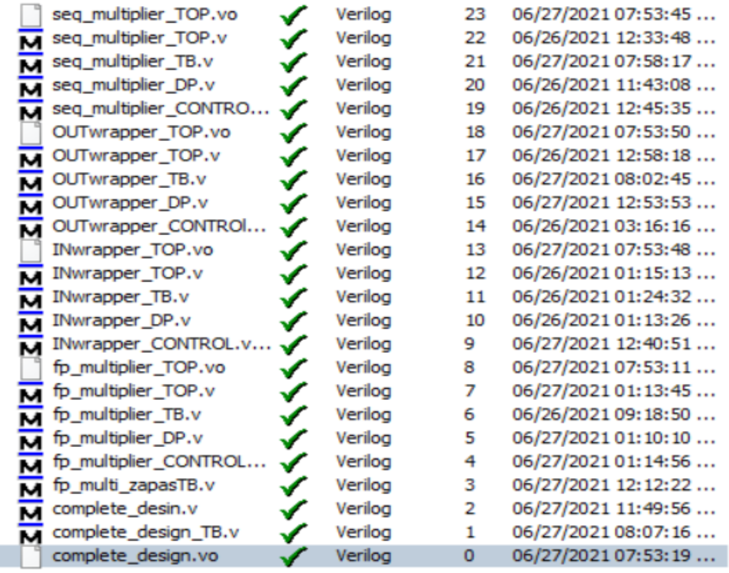


```verilog
`timescale 1ns/1ns
module complete_design_TB();
        reg clk=1'b0;
        reg rst=1'b1;
        reg inReady=1'b0;
        reg resultAccepted=1'b0;
        reg [31:0]inBus;
        wire inAccept1,inAccept2,resultReady1,resultReady2;
        wire [31:0]outBus1,outBus2;
        complete_design X1(clk,rst,inReady,resultAccepted,inBus,outBus1,inAccept1,resultReady1);
        complete_design_syn u(clk,rst,inReady,resultAccepted,inBus,outBus2,inAccept2,resultReady2);
        always #100 clk=~clk;
        initial begin
        #50 rst=1'b0;
        #200 inBus={1'b1,1'b0,1'b0,1'b1,1'b0,1'b0,1'b1,1'b0,1'b0,23'd99999256};
        #200 inReady=1'b1;
        #200 inReady=1'b0;
        #200 inBus={1'b0,1'b1,1'b0,1'b0,1'b0,1'b0,1'b0,1'b0,1'b1,23'd99999256};
        #200 inReady=1'b1;
        #200 inReady=1'b0;
        #5000 resultAccepted=1'b1;
        #10000 $stop;
        end
endmodule
```



	Msgs
/complete_design_TB/clk	0
/complete_design_TB/inAccept1	St0
/complete_design_TB/inAccept2	St0
/complete_design_TB/inBus	010000...
/complete_design_TB/inReady	0
/complete_design_TB/outBus1	100100...
/complete_design_TB/outBus2	010000...
/complete_design_TB/resultAc...	1
/complete_design_TB/resultR...	St0
/complete_design_TB/resultR...	St1
/complete_design_TB/rst	0

m)



seq_multiplier_TOP.vo	✓	Verilog	23	06/27/2021 07:53:45 ...
seq_multiplier_TOP.v	✓	Verilog	22	06/26/2021 12:33:48 ...
seq_multiplier_TB.v	✓	Verilog	21	06/27/2021 07:58:17 ...
seq_multiplier_DP.v	✓	Verilog	20	06/26/2021 11:43:08 ...
seq_multiplier_CONTRO...	✓	Verilog	19	06/26/2021 12:45:35 ...
OUTwrapper_TOP.vo	✓	Verilog	18	06/27/2021 07:53:50 ...
OUTwrapper_TOP.v	✓	Verilog	17	06/26/2021 12:58:18 ...
OUTwrapper_TB.v	✓	Verilog	16	06/27/2021 08:02:45 ...
OUTwrapper_DP.v	✓	Verilog	15	06/27/2021 12:53:53 ...
OUTwrapper_CONTROl...	✓	Verilog	14	06/26/2021 03:16:16 ...
INwrapper_TOP.vo	✓	Verilog	13	06/27/2021 07:53:48 ...
INwrapper_TOP.v	✓	Verilog	12	06/26/2021 01:15:13 ...
INwrapper_TB.v	✓	Verilog	11	06/26/2021 01:24:32 ...
INwrapper_DP.v	✓	Verilog	10	06/26/2021 01:13:26 ...
INwrapper_CONTROL.v...	✓	Verilog	9	06/27/2021 12:40:51 ...
fp_multiplier_TOP.vo	✓	Verilog	8	06/27/2021 07:53:11 ...
fp_multiplier_TOP.v	✓	Verilog	7	06/27/2021 01:13:45 ...
fp_multiplier_TB.v	✓	Verilog	6	06/26/2021 09:18:50 ...
fp_multiplier_DP.v	✓	Verilog	5	06/27/2021 01:10:10 ...
fp_multiplier_CONTROL...	✓	Verilog	4	06/27/2021 01:14:56 ...
fp_multi_zapasTB.v	✓	Verilog	3	06/27/2021 12:12:22 ...
complete_desin.v	✓	Verilog	2	06/27/2021 11:49:56 ...
complete_design_TB.v	✓	Verilog	1	06/27/2021 08:07:16 ...
complete_design.vo	✓	Verilog	0	06/27/2021 07:53:19 ...