



**数字逻辑与系统设计**

**实验报告**

**学 院 计算机学院**

**专 业 计算机科学与技术**

**年级班别 2020级（3）班**

**学 号 3120005043**

**学生姓名 张俊鸿**

**指导教师 张海笑**

**2022 年 1 月**

**基于Libero SoC的数字逻辑实验**

|  |  |  |
| --- | --- | --- |
| 序号 | 实验内容 | 完成情况 |
| 1 | 基本门电路 |  |
| 2 | 门电路综合实验 |  |
| 3 | 组合逻辑电路实验 |  |
| 4 | 实验考核 |  |
| 5 | 时序逻辑电路实验 |  |
| 6 | 有限状态机实验 |  |
| 7 | 综合实验 |  |
| 8 | 实验考核 |  |

**实验报告**

实验题目 基于Libero Soc的基本门电路实验

**注：所有基于Libero Soc的实验，都在一个工程项目中完成。**

**工程文件名（Project Name）：D学号+下划线+姓名拼音首字母**

**例：学号3115000001姓名张小童，工程文件名为：D3115000001\_zxt**

一、实验目的

1、了解基于Verilog的基本门电路的设计及其验证。

2、熟悉利用EDA工具进行设计及仿真的流程。

3、学习针对实际门电路芯片74HC00、74HC02、74HC04、74HC08、74HC32、74HC86进行VerilogHDL设计的方法。

4、掌握Libero软件的使用方法。

二、实验环境

Libero仿真软件。

三、实验内容

1、新建一个设计代码文件（Verilog Source File）

文件命名规则举例：**3115000001\_BasGate.v**（注：蓝色部分为学号）

新建一个测试平台文件（HDL Stimulus File），文件命名：**test\_BasGate.v**

2、在设计代码文件中，建立一个功能模块，要求如下：

模块名举例：**zxt\_BasGate**（注：蓝色部分为姓名首字母）

输入信号：A,B

输出信号：Y1，Y2，Y3，Y4，Y5，Y6

逻辑功能：Y1~Y5分别实现A、B的与、与非、或、或非、异或逻辑，Y6实现A的非逻辑。

3、设计测试平台，对上述功能模块进行功能仿真（综合前仿真）

测试平台模块名举例：**test\_zxt\_BasGate**（注：蓝色部分为姓名首字母）

4、对上述功能模块进行综合，并进行综合后仿真。

5、对上述功能模块进行布局布线，并进行布局布线后的仿真。

6、烧录及接电测试

7、记录实验过程。

8、保存工程文档。

四、实验结果和数据处理

1、**门电路**模块清单及测试平台代码清单

（1）硬件功能模块的代码清单（关键代码应有注释）

module zjh\_BasGate(A,B,Y1,Y2,Y3,Y4,Y5,Y6);

input A,B;

output Y1,Y2,Y3,Y4,Y5,Y6;

//连续赋值语句一般用于描述组合逻辑

assign Y1=A&B; //与

assign Y2=~(A&B); //与非

assign Y3=A|B; //或

assign Y4=~(A|B); //或非

assign Y5=A^B; //异或

assign Y6=~A; //非

endmodule

（2）测试平台模块的代码清单（关键语句应有注释）

`timescale 10ns/1ns

module test\_zjh\_BasGate;

reg a,b;

wire y1,y2,y3,y4,y5,y6;

zjh\_BasGate u0(a,b,y1,y2,y3,y4,y5,y6);

//调用前面的gates模块，按端口连接

initial a,b的值将按照00-01-11-10的顺序产生

begin

a=0;b=0;

#10 b=1;

#10 a=1;

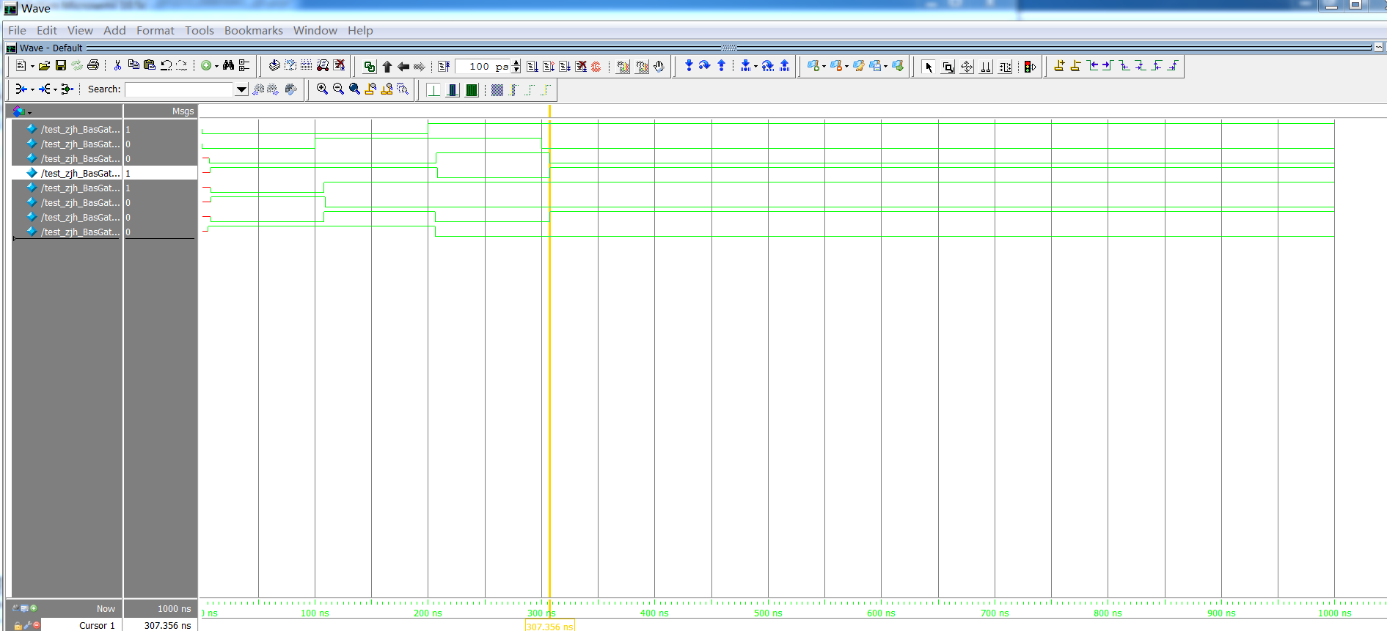
#10 b=0;

end

endmodule

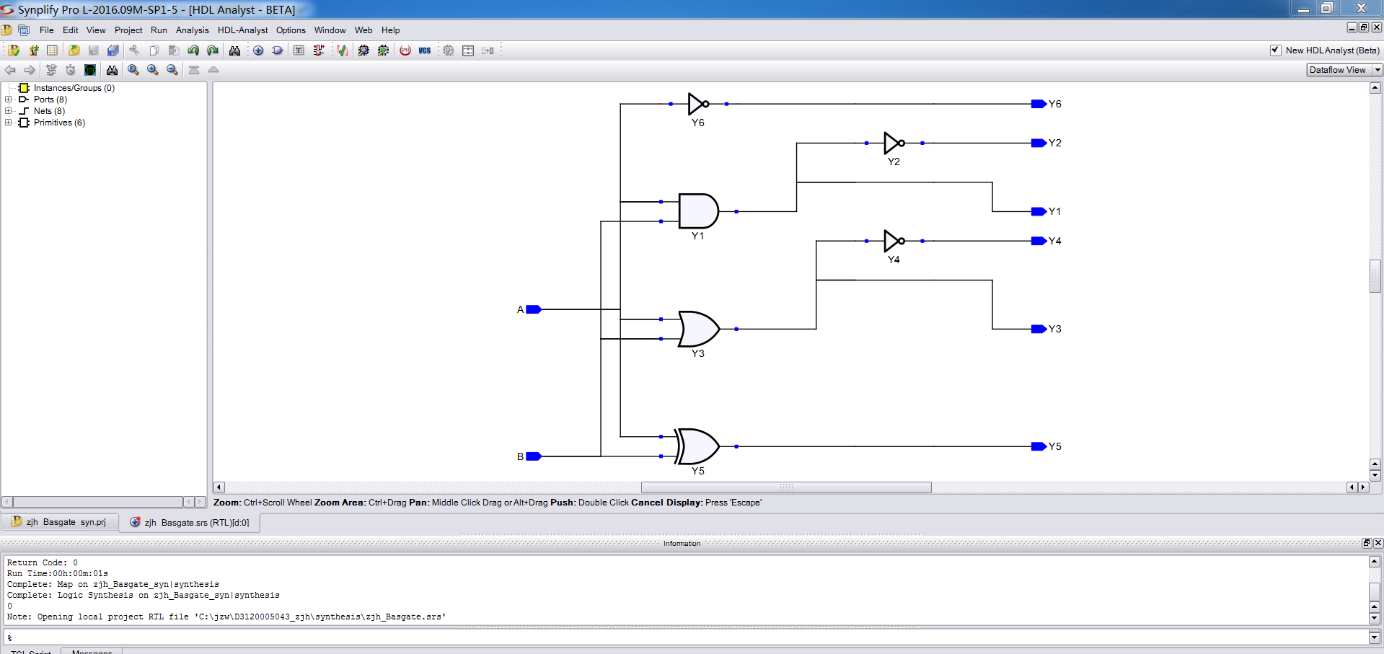
2、综合前仿真结果（**截图**）。

先将波形窗口背景设为**白色**，调整窗口至合适大小，使波形能完整显示，再对窗口**截图**。后面实验中的仿真使用相同方法处理。



3、综合结果（**截图**）。

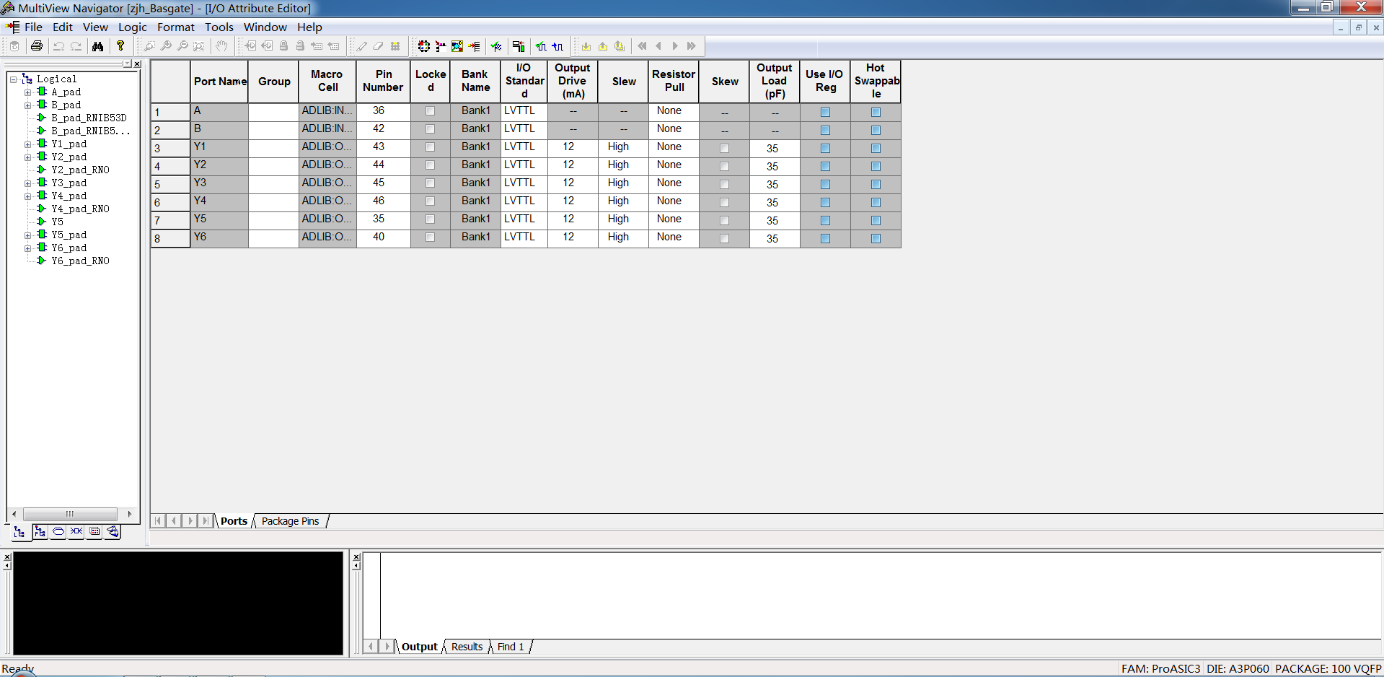
先将相关窗口调至合适大小，使RTL图能完整显示，对窗口截图，后面实验中的综合使用相同方法处理）



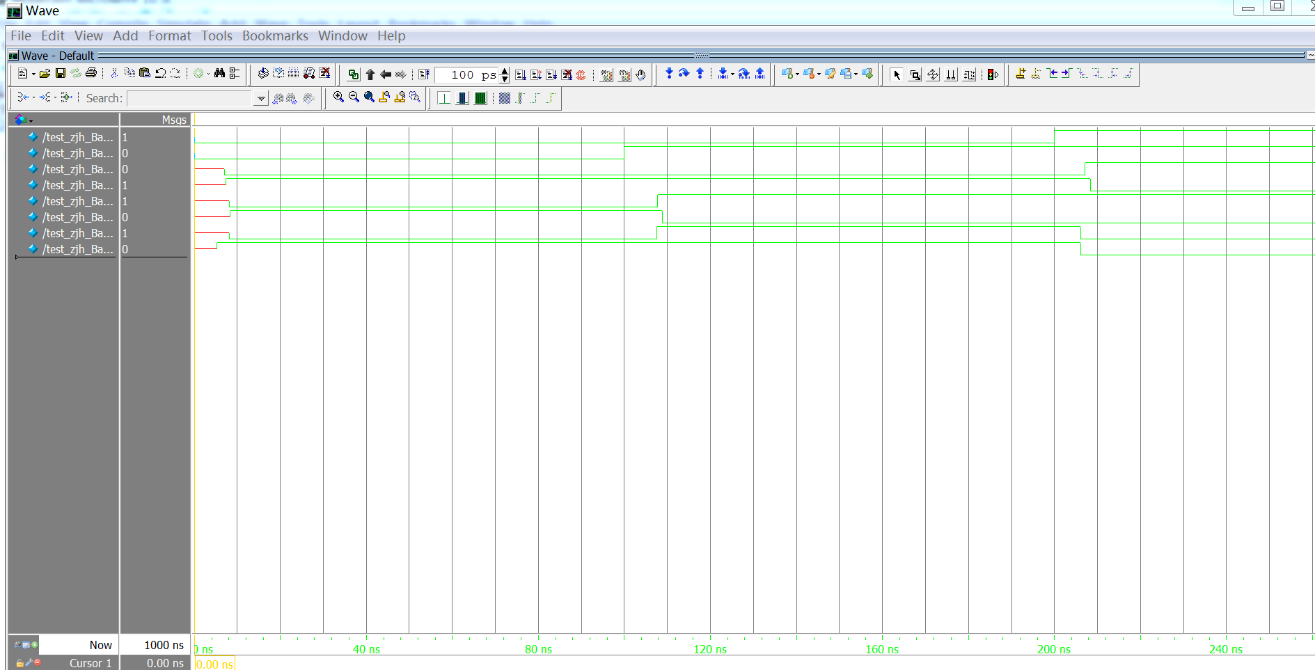
4、综合后仿真结果（**截图**）。



5、布局布线的引脚分配（**截图**）。



6、第三次仿真结果（布局布线后）（**截图**）。



7、回答问题：3次仿真，输出信号是否有延迟，延迟时间约为多少？

答：第一次仿真没有延迟，第二次和三次仿真延迟约0.3ns。

8、更改布局布线分配的引脚，对延迟时间是否有影响？

答：有影响

**实验报告**

实验题目 基于Libero Soc的基本门电路综合实验

一、实验目的

1、了解基于Verilog的组合逻辑电路的设计及其验证。

2、熟悉利用EDA工具进行设计及仿真的流程。

3、学习针对实际要求进行VerilogHDL设计的方法。

4、掌握Libero软件的使用方法。

二、实验环境

Libero仿真软件。

三、实验内容

继续在上一实验所建的“学号+下划线+BasGate（例：3115000001\_BasGate.v）”文件中添加两段模块设计代码，分别完成以下第1、2项实验内容。

两个设计所对应的测试平台模块代码继续放在test\_BasGate.v文件中。

1、裁判表决电路

设计一个3输入、1输入的举重裁判表决电路，使用VerilogHDL描述该电路的功能，设计相关测试平台，完成综合、布局布线、仿真。

模块名举例：**zxt\_cp**（注：蓝色部分为姓名首字母）

测试平台名举例：**test\_zxt\_cp**（注：蓝色部分为姓名首字母）

2、交通灯故障检测电路

设计一个3输入、1输入的交通灯故障检测电路，使用VerilogHDL描述该电路的功能，设计相关测试平台，完成综合、布局布线、仿真。

模块名举例：**zxt\_jtd**（注：蓝色部分为姓名首字母）

测试平台名举例：**test\_zxt\_jtd**（注：蓝色部分为姓名首字母）

3、以上两个电路任选一个完成烧录和接电测试，并给老师检查。

四、实验结果和数据处理

1、所有模块及测试平台代码清单

//举重裁判表决电路

module zjh\_cp(A,B,C,Y);

input A,B,C;

output Y;

assign Y=(A&B)|(A&C)|(B&C);

//只有两个以上同时为1，结果才会为1

endmodule

//举重裁判表决电路测试平台

module test\_zjh\_cp;

reg a,b,c;

wire y;

zjh\_cp u1(a,b,c,y);

initial

begin

a=0;b=0;c=0; //000

#10 c=1; //001

#10 b=1; //011

#10 c=0; //010

#10 a=1; //100

#10 b=0; //100

#10 c=1; //101

#10 b=1; //111

#10 c=0; //110

end

endmodule

//交通灯故障检测电路

module zjh\_jtd(A, B, C, Y);

input A, B, C;

output Y;

assign Y = (A&B)|(A&C)|(B&C)|~(A|B|C);

//当三个灯都灭或者存在两个以上同时亮时Y输出1

endmodule

//交通灯故障检测电路测试平台

module test\_zjh\_jtd;

reg a,b,c;

wire y;

zjh\_jtd u2(a,b,c,y);

initial

begin

a=0;b=0;c=0; //000

#10 c=1; //001

#10 b=1; //011

#10 c=0; //010

#10 a=1; //100

#10 b=0; //100

#10 c=1; //101

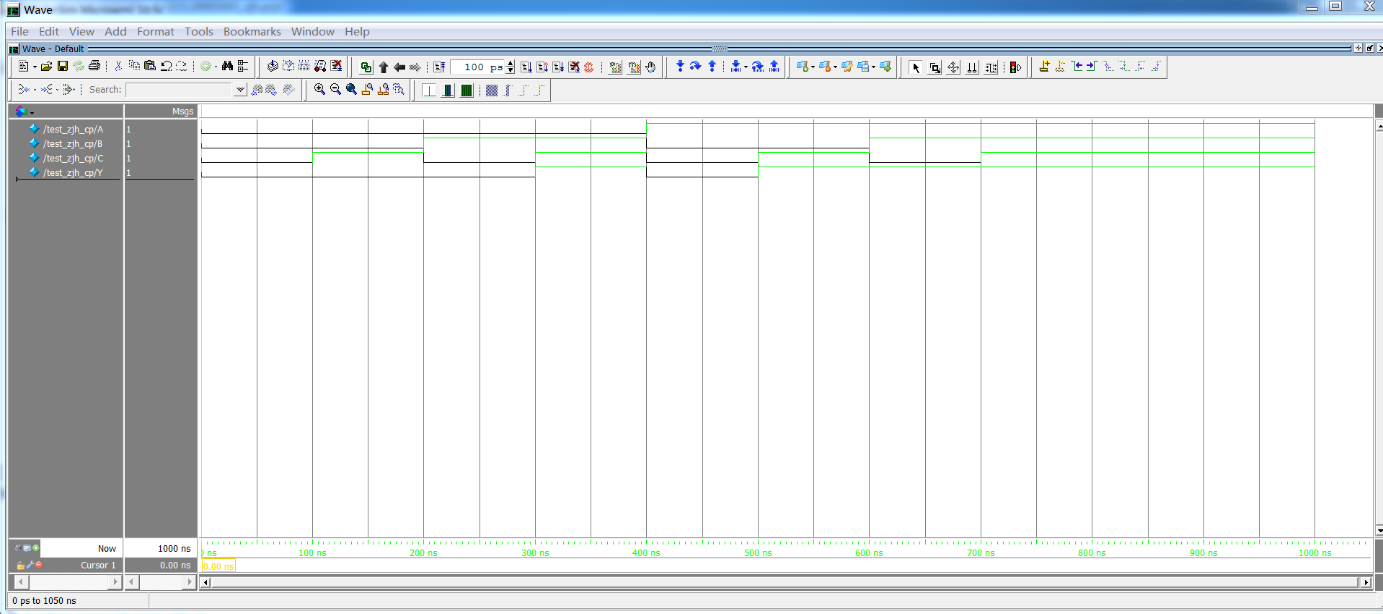
#10 b=1; //111

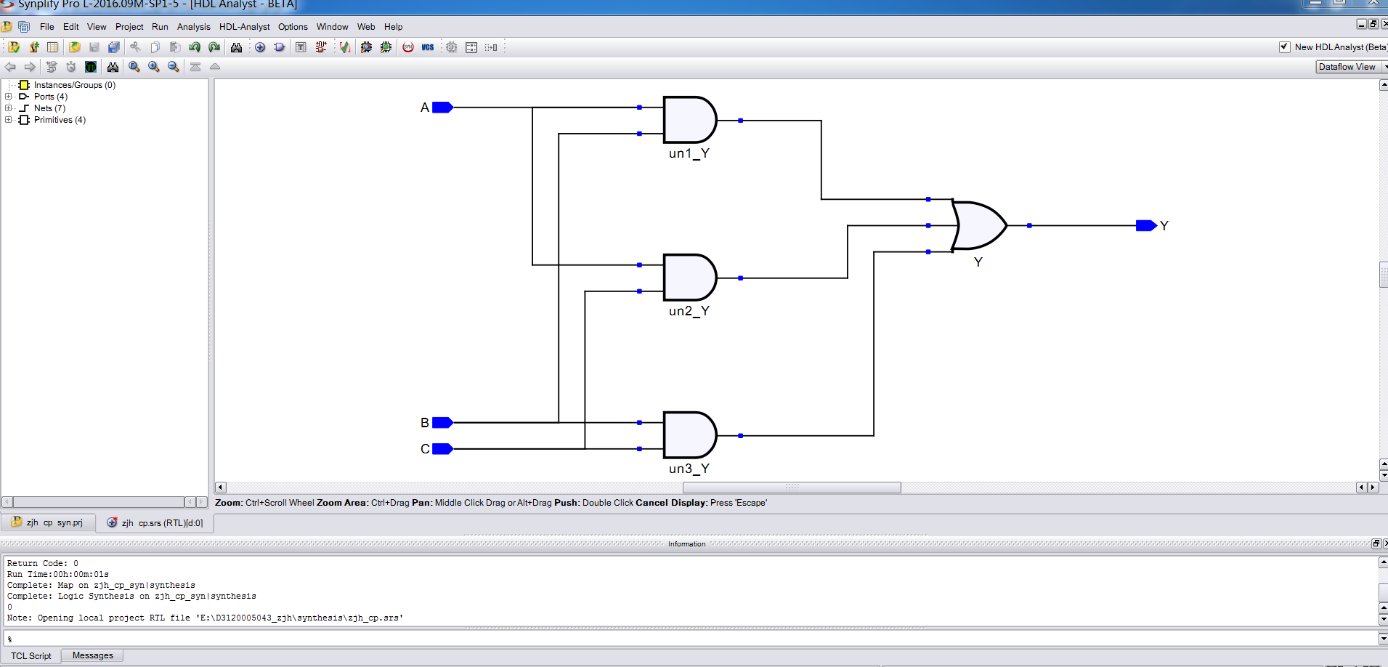
#10 c=0; //110

end

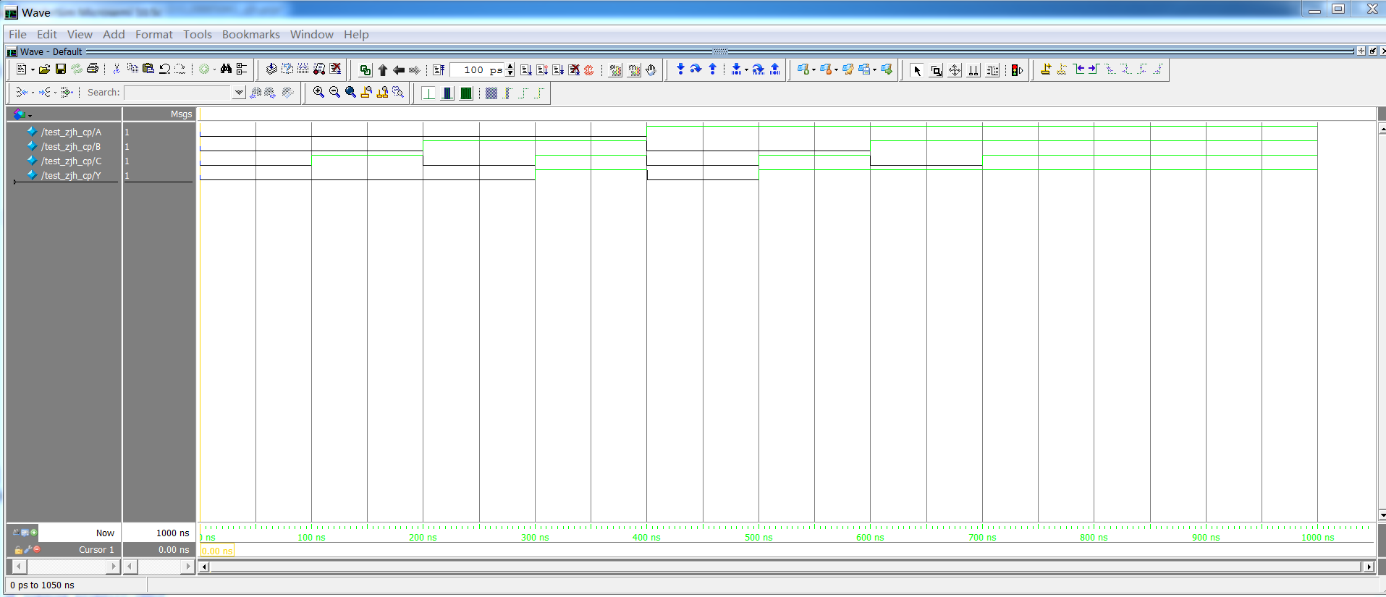
endmodule

2、综合前仿真结果截图（任选一个模块，请注明）

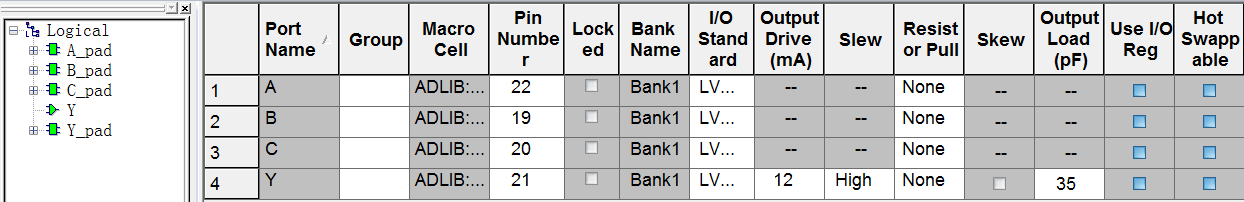
3、综合结果RTL视图截图（任选一个模块，请注明）



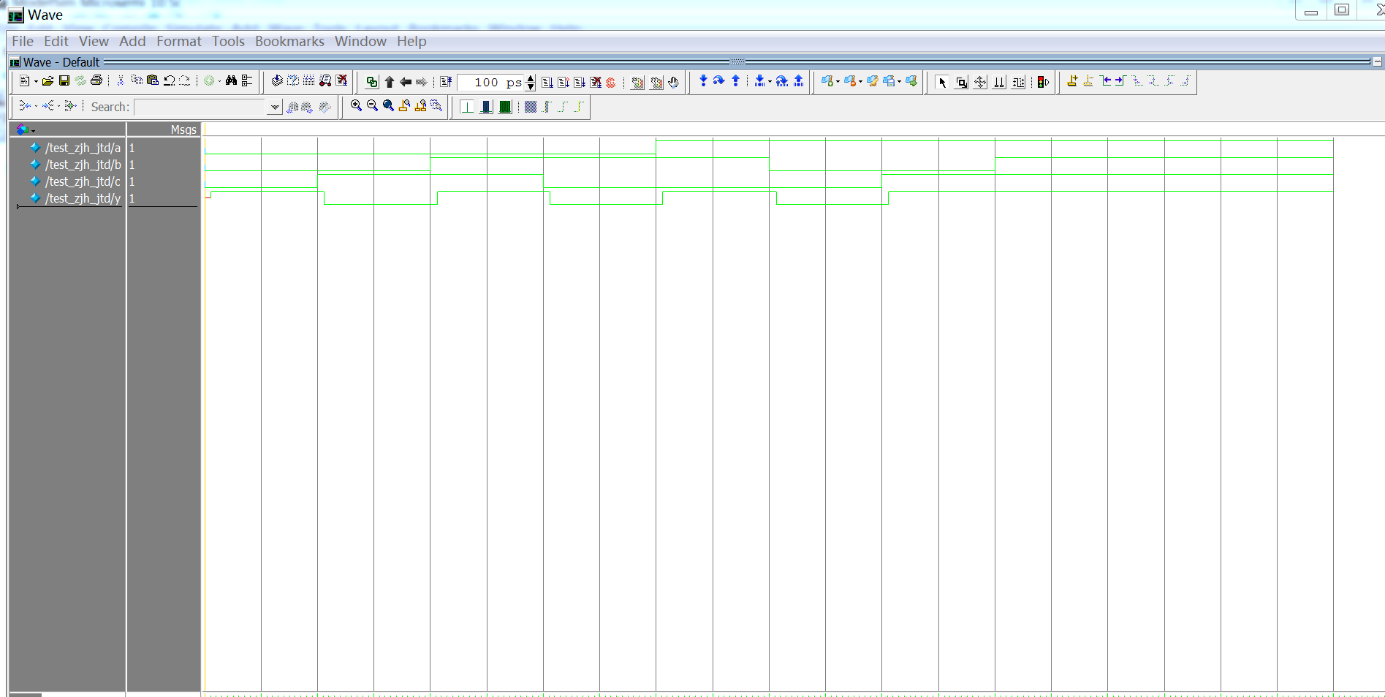
4、综合后仿真截图（任选一个模块，请注明）。



5、布局布线引脚分配窗口截图



6、布局布线后仿真结果截图。



**实验报告**

实验题目 基于Libero Soc的组合逻辑电路设计实验

一、实验目的

1、了解基于Verilog的组合逻辑电路的设计及其验证。

2、熟悉利用EDA工具进行设计及仿真的流程。

3、熟悉实验箱的使用和程序下载（烧录）及测试的方法。

4、学习针对实际组合逻辑电路芯片74HC148、74HC138、74HC153、74HC85、74HC283、74HC4511进行VerilogHDL设计的方法。

5、学习组合逻辑电路的设计方法。

二、实验环境及仪器

1、Libero仿真软件。

2、数字逻辑与系统设计实验箱。

3、Actel A3P060 FPGA芯片及Flash Pro5烧录器。

三、实验内容

1、打开自己的工程文件，以下是本实验新建设计文件及测试平台文件以及各模块的命名要求：

（1）新建一个设计文件（Create HDL）

文件命名规则举例：**3115000001\_comb.v**（注：蓝色部分为学号）

包含以下模块，模块名命名规则如下：（注：蓝色部分为姓名首字母）

**zxt\_74HC148**

**zxt\_74HC138**

**zxt\_74HC153**

**zxt\_74HC85**

**zxt\_74HC283**

**zxt\_74HC4511**

**zxt\_TF\_CC //**原码-补码转换器

（2）新建一个测试平台文件（Create HDL testbench）

文件命名规则举例：**test\_comb.v**

包含以下模块，模块名命名规则如下：（注：蓝色部分为姓名首字母）

**test\_zxt\_74HC148**

**test\_zxt\_74HC138**

**test\_zxt\_74HC153**

**test\_zxt\_74HC85**

**test\_zxt\_74HC283**

**test\_zxt\_74HC4511**

**test\_zxt\_TF\_CC //**原码-补码转换器测试平台

2、按课本第2章对74HC148、74HC138、74HC153、74HC85、74HC283、74HC4511芯片以及原码-补码转换器的功能描述，编写每个模块的设计代码及测试平台代码。

其中74HC4511需设计成扩展型，即能显示数字0~9、字母a~f。

3、上述内容均要求完成综合、布局布线、三次仿真及烧录。

4、老师检查烧录后的74HC4511及其余任意一个组合电路。

5、提交针对74HC148、74HC138、74HC153、74HC85、74HC283、74HC4511、原码-补码转换器的的综合结果，以及相应的仿真结果。

四、实验结果和数据处理

1、所有模块及测试平台代码清单（关键语句要有注释）

//74HC148代码

module zjh\_74HC148(DataIn,EO,Dataout,EI,GS);

input[7:0] DataIn;

input EI;

output EO,GS;

output[2:0]Dataout;

reg [2:0] Dataout;

reg EO, GS;

integer I;

always@ (DataIn or EI) //DataIn,EI为敏感事件，一旦发生变化，即触发后面代码的执行

begin

if(EI)

begin

Dataout=7;EO=1;GS=1;

end

else if (DataIn==8'b11111111)

begin

Dataout=7;EO=0;GS=1;

end

else

for (I=0;I<8;I=I+1)

begin

if(~DataIn[I])

begin

Dataout= ~I;EO=1;GS=0;

end

end

end

endmodule

//74HC148测试平台代码

`timescale 1ns / 1ns

module test\_zjh\_74HC148;

reg ei;

reg [7:0]turn;

wire[7:0]in= ~turn;

wire [2:0]out;

wire eo, gs;

zjh\_74HC148 u2(in,eo,out,ei,gs);

initial

begin

ei=1;turn=8'b1;

repeat (8)

#10turn=turn<<1;ei=0;turn=8'b1;

repeat (8)

#10turn=turn<<1;

end

endmodule

//74HC138代码

module zjh\_74HC138(E,A,Y);

input [2:0]E,A;

output [7:0]Y;

reg [7:0]Y;

integer I;

always@(E or A)

begin

if(E[2]==0||E[1]==1||E[0]==1)

begin

Y='b11111111;

end

else if(E==3'b100)

begin

for(I=0;I<=7;I=I+1)

begin

if(A==I)

Y[I]=0;

else

Y[I]=1;

end

end

end

endmodule

//74HC138测试平台代码

module test\_zjh\_74HC138;

reg[2:0]e,a;

wire[7:0]y;

integer I;

zjh\_74HC138 u3(e,a,y);

initial

begin

e=3'bxx1;

#10 e=3'bx1x;

#10 e=3'b0xx;

#10 e=3'b100;

for(I=0;I<=7;I=I+1)

#10 a=I;

end

endmodule

//74HC153代码

module zjh\_74HC153(DateOut, DateIn, Sel, Enable);

input [3:0]DateIn;

input [1:0]Sel;

input Enable;

output reg DateOut;

always @(Enable or Sel or DateIn)

if(Enable) DateOut = 0;

else DateOut = DateIn[Sel];

endmodule

//74HC153测试平台代码

module test\_zjh\_74HC153;

wire out;

reg [3:0]in;

reg [1:0]sel;

reg ei;

zjh\_74HC153 u(out, in, sel, ei);

initial

begin

ei = 0; sel = 0; in = 4'b1010;

repeat(4)

#10 sel = sel +1;

ei = 1; sel = 0; in = 4'b1010;

repeat(4)

#10 sel = sel +1;

end

endmodule

//74HC85代码

module zjh\_74HC85(A,B,IAGB,IASB,IAEB,QAGB,QASB,QAEB);

input [3:0]A,B;

input IAGB,IASB,IAEB;

output QAGB,QASB,QAEB;

reg QAGB,QASB,QAEB;

always@(A or B or IAGB or IAEB or IASB)

begin

if(A>B)

begin

QAGB=1;QASB=0;QAEB=0;

end

else if(A<B)

begin

QAGB=0;QASB=1;QAEB=0;

end

else if(IAGB&&!IAEB&&!IASB)

begin

QAGB=1;QASB=0;QAEB=0;

end

else if(!IAGB&&IASB&&!IAEB)

begin

QAGB=0;QASB=1;QAEB=0;

end

else if(IAEB==1)

begin

QAGB=0;QASB=0;QAEB=1;

end

else if(IAGB&&!IAEB&&IASB)

begin

QAGB=0;QASB=0;QAEB=0;

end

else if(!IAGB&&!IAEB&&!IASB)

begin

QAGB=1;QASB=1;QAEB=0;

end

end

endmodule

//74HC85测试平台代码

module test\_zjh\_74HC85;

reg [3:0]a,b;

reg iagb,iasb,iaeb;

wire qagb,qasb,qaeb;

zjh\_74HC85 u4(a,b,iagb,iasb,iaeb,qagb,qasb,qaeb);

initial

begin

a=0;

repeat(20)

#10 a=$random;

end

initial

begin

b=0;

repeat(20)

#10 b=$random;

end

initial

begin

#210 a=0;b=0;iaeb=1;

#10 iagb=0;iaeb=0;iasb=0;

#10 iagb=0;iaeb=0;iasb=1;

#10 iagb=1;iaeb=0;iasb=0;

#10 iagb=1;iaeb=0;iasb=1;

end

endmodule

//74HC283代码

module zjh\_74HC283(A,B,Cin,S,Cout);

input [3:0]A,B;

input Cin;

output[3:0]S;

reg[3:0]S;

output Cout;

reg Cout;

always@(A or B or Cin)

{Cout,S}=A+B+Cin;

endmodule

//74HC283测试平台代码

module test\_zjh\_74HC283;

reg [3:0]a,b;

reg cin;

wire [3:0]s;

wire cout;

zjh\_74HC283 u5(a,b,cin,s,cout);

initial

begin

a=0;

repeat(20)

#10 a=$random;

end

initial

begin

b=0;

repeat(20)

#10 b=$random;

end

initial

begin

cin=0;

repeat(20)

#10 cin=$random;

end

endmodule

//74HC4511代码

module zjh\_74HC4511(D,LE,BI,LT,L);

input LE,BI,LT;

input [3:0]D;

output reg [6:0]L;

always @(\*)begin

if(LE==0&&BI==1&&LT==1)begin

case(D[3:0])

4'b0000:L[6:0]=7'b1111110;

4'b0001:L[6:0]=7'b0110000;

4'b0010:L[6:0]=7'b1101101;

4'b0011:L[6:0]=7'b1111001;

4'b0100:L[6:0]=7'b0110011;

4'b0101:L[6:0]=7'b1011011;

4'b0110:L[6:0]=7'b1011111;

4'b0111:L[6:0]=7'b1110000;

4'b1000:L[6:0]=7'b1111111;

4'b1001:L[6:0]=7'b1111011;

4'b1010:L[6:0]=7'b1110111;

4'b1011:L[6:0]=7'b0011111;

4'b1100:L[6:0]=7'b1001110;

4'b1101:L[6:0]=7'b0111101;

4'b1110:L[6:0]=7'b1001111;

4'b1111:L[6:0]=7'b1000111;

endcase

end

else if(LT==0)L[6:0]=7'b1111111;

else if(BI==0&&LT==1)L[6:0]=7'b0000000;

else L<=L;

end

endmodule

//74HC4511测试平台代码

module test\_zjh\_74HC4511;

reg LE,BI,LT;

reg [3:0]D;

wire [6:0]L;

zjh\_74HC4511 u7(D,LE,BI,LT,L);

initial begin

LE=0;BI=0;LT=0;D=4'b0000;

#5

LT=1;

#5

BI=1;

#5

D=4'b0001;

#5

D=4'b0010;

#5

D=4'b0011;

#5

D=4'b0100;

#5

D=4'b0101;

#5

D=4'b0110;

#5

D=4'b0111;

#5

D=4'b1000;

#5

D=4'b1001;

#5

D=4'b1010;

#5

D=4'b1011;

#5

D=4'b1100;

#5

D=4'b1101;

#5

D=4'b1110;

#5

D=4'b1111;

end

endmodule

//8位原码-补码转换器

module zjh\_TF\_CC(A,B);

input [3:0]A;

output [3:0]B;

reg [3:0]B;

always @(A)

begin

if(~A[3]) B=A;

else

begin

B={1'b1,~A[2:0]+1};

end

end

endmodule

//8位原码-补码转换器测试平台代码

module test\_zjh\_TF\_CC;

reg[3:0]a;

wire [3:0]b;

integer I;

zjh\_TF\_CC u6(a,b);

initial

begin

for(I=0;I<16;I=I+1)

begin

a=I; #5;

end

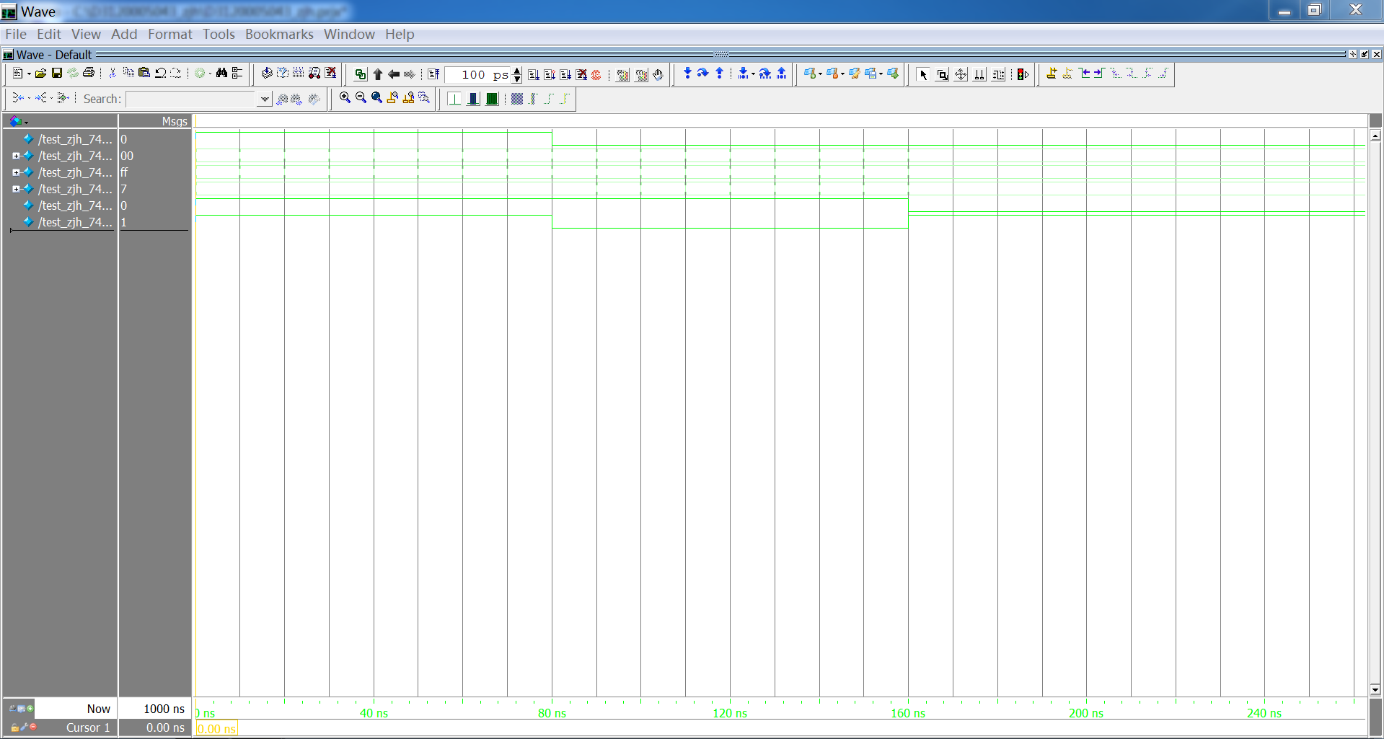
end

initial

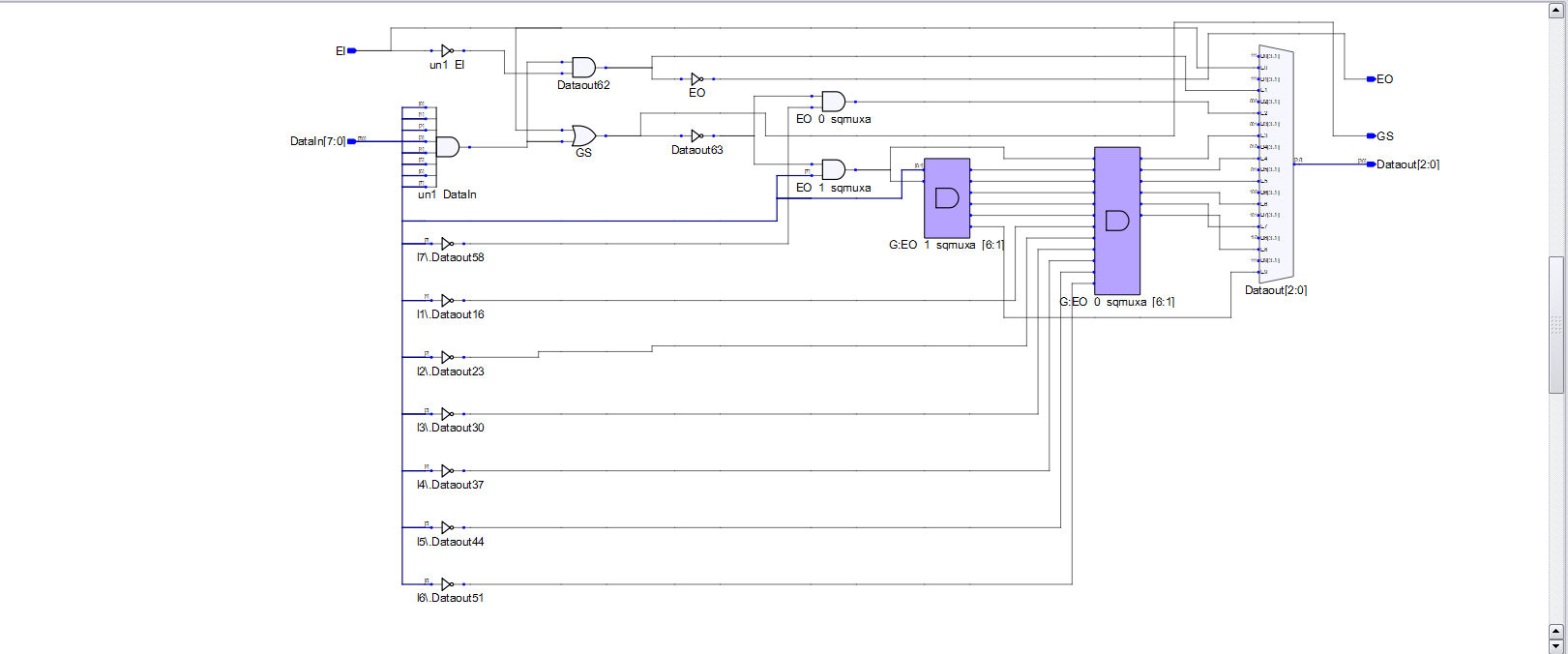
$monitor("time=%t,a=%b,b=%b",$time,a,b);

endmodule

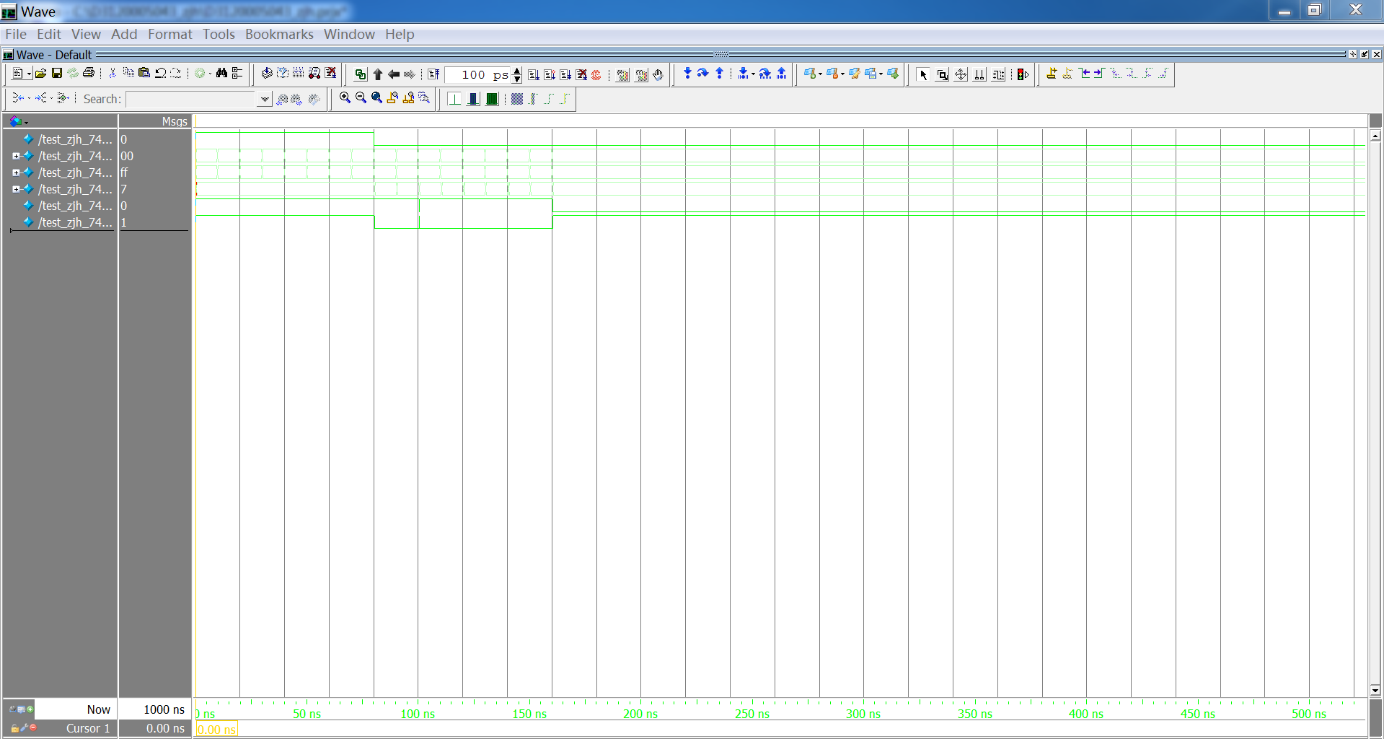
2、第一次仿真结果截图（电路名称：74HC148）



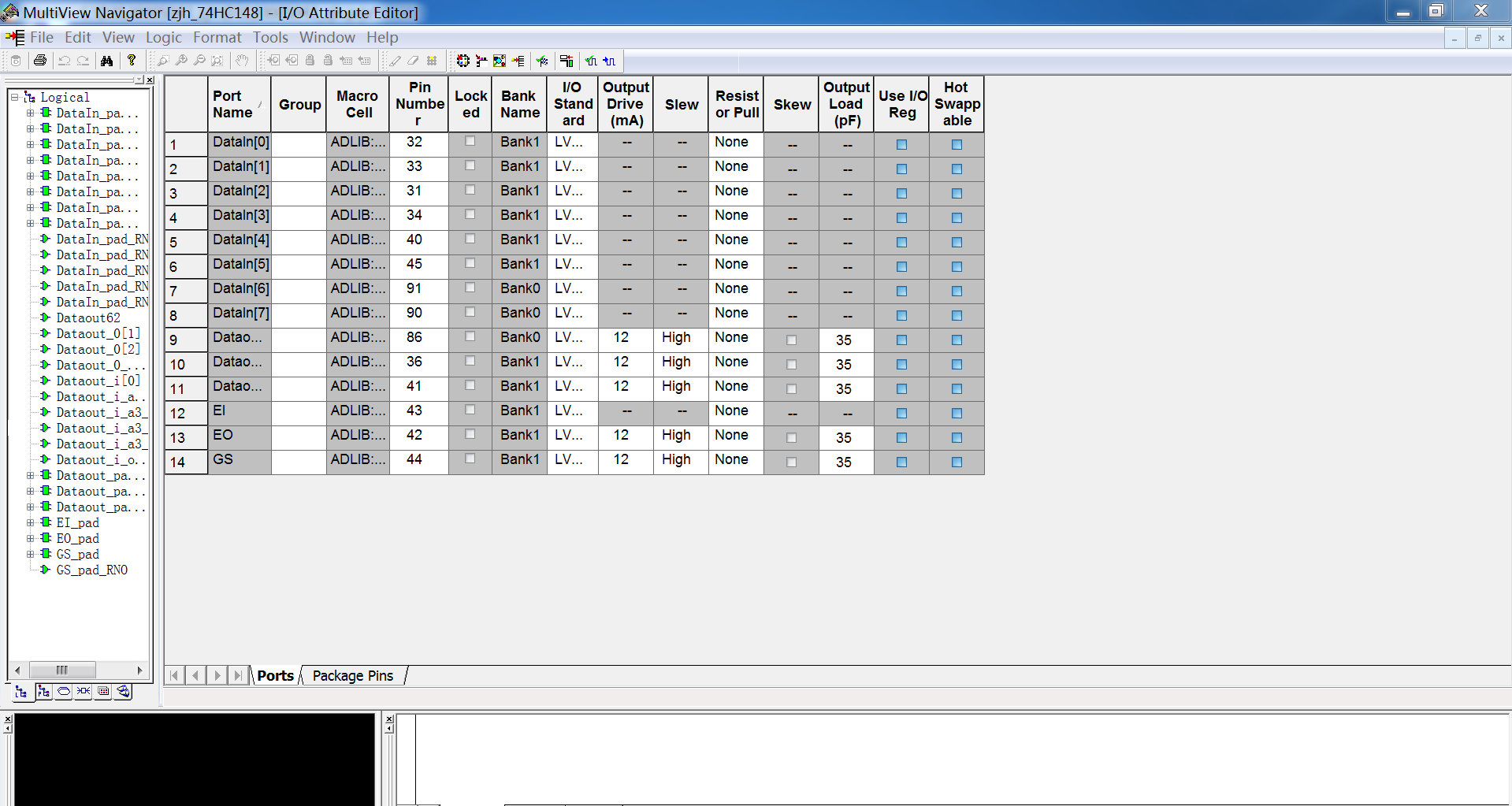
3、综合结果（截图）



4、第二次仿真结果（综合后仿真截图）。

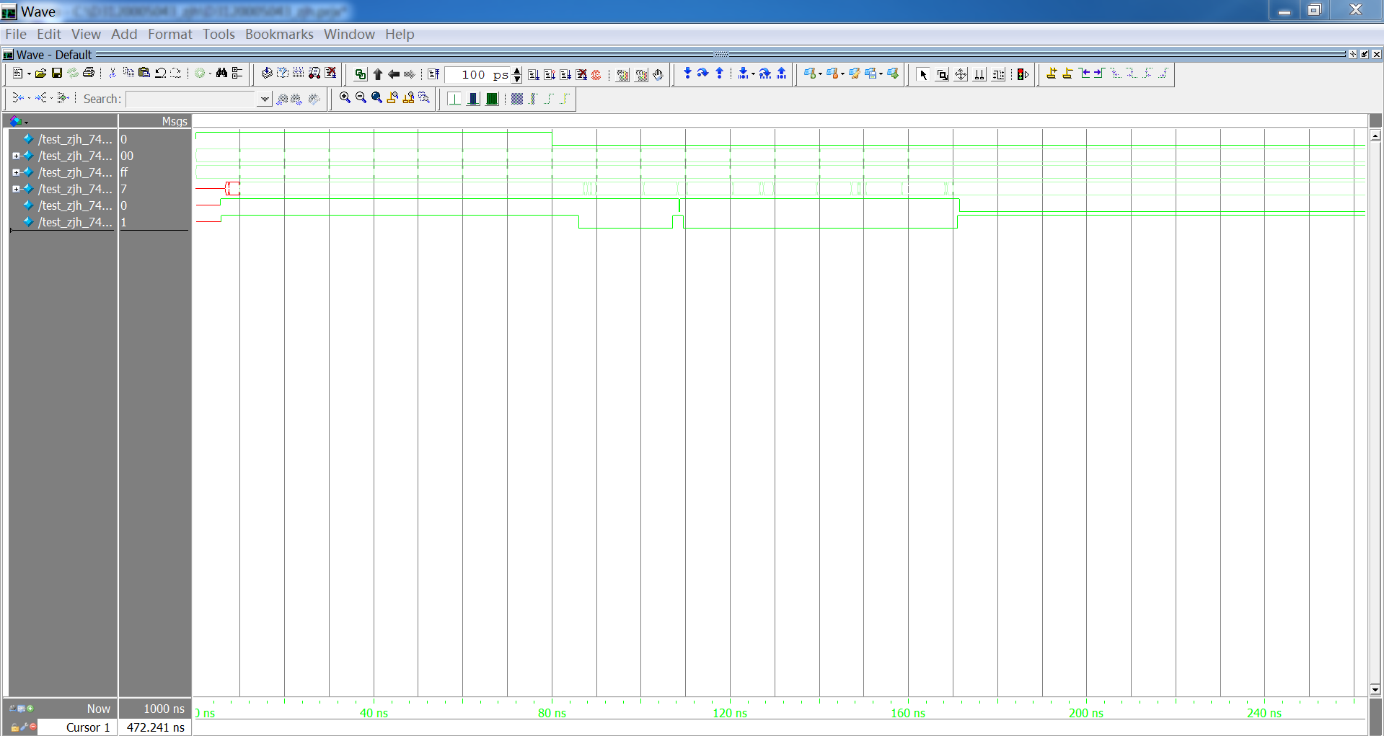


5、布局布线（引脚分配截图）



6、第三次仿真结果（布局布线后）。回答输出信号是否有延迟，最长延迟时间约为多少？分析是否有出现竞争冒险。

答：输出信号有延迟，最初延迟时间约为5.1ns，出现竞争冒险。

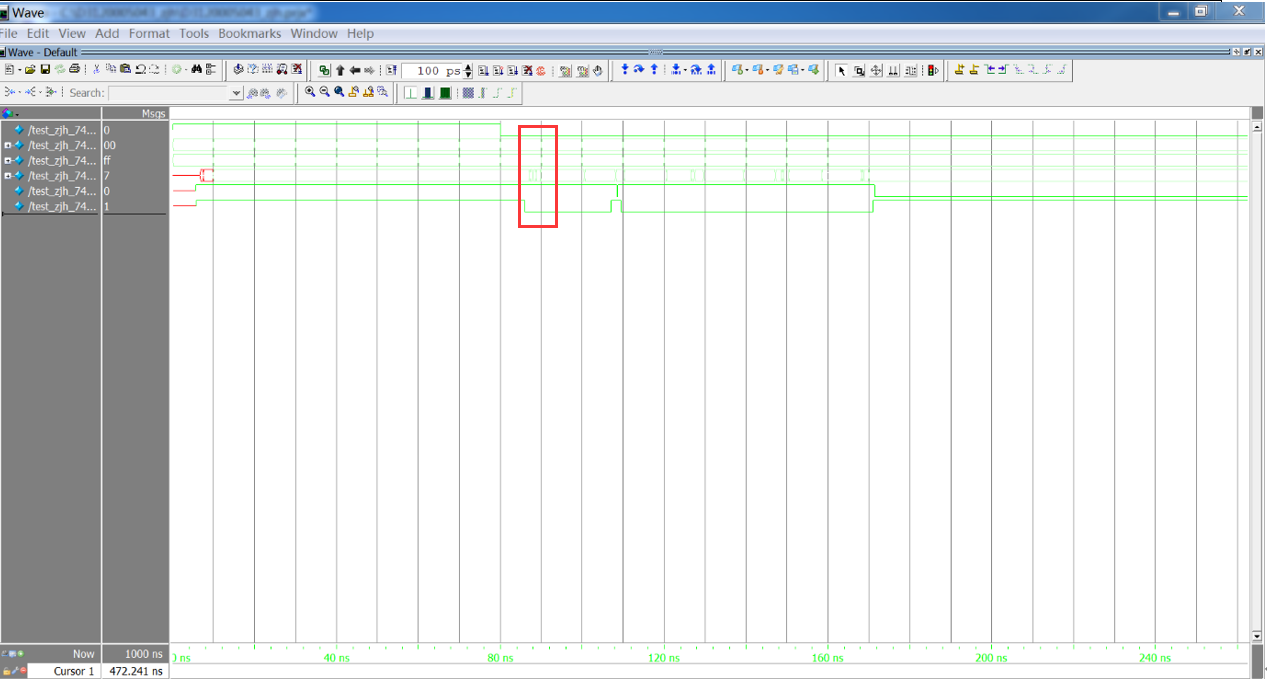


7、回答问题：3次仿真，输出信号是否有延迟，延迟时间约为多少？

答：第一次仿真无延迟，第二次仿真延迟约为0.5ns，第三次仿真延迟约为5.1ns。

8、分析第3次仿真是否有竞争冒险，请截图标注说明。

答：有竞争冒险



如图，截图部分为竞争冒险举例

**实验报告**

实验题目 基于Libero Soc的时序逻辑电路设计实验

一、实验目的

1、了解基于Verilog的时序逻辑电路的设计及其验证。

2、熟悉利用EDA工具进行设计及仿真的流程。

3、熟悉实验箱的使用和程序下载（烧录）及测试的方法。

4、学习针对实际时序逻辑电路芯片74HC74、74HC112、74HC194、74HC161进行VerilogHDL设计的方法。

二、实验环境及仪器

1、Libero仿真软件。

2、数字逻辑与系统设计实验箱。

3、Actel A3P060 FPGA芯片及Flash Pro5烧录器。

三、实验内容

1、打开自己的工程文件，以下是本实验的新建设计文件及测试平台文件以及各模块的命名要求：

（1）新建一个设计文件（Create HDL）

文件命名规则举例：**3115000001\_seq.v**（注：蓝色部分为学号）

包含以下模块，模块名命名规则如下：（注：蓝色部分为姓名首字母）

**zxt\_74HC74**

**zxt\_74HC112**

**zxt\_74HC194**

**zxt\_74HC161**

（2）新建一个测试平台文件（Create HDL testbench）

文件命名规则举例：**test\_seq.v**

包含以下模块，模块名命名规则如下：（注：蓝色部分为姓名首字母）

**test\_zxt\_74HC74**

**test\_zxt\_74HC112**

**test\_zxt\_74HC194**

**test\_zxt\_74HC161**

2、按课本第3章对74HC74、74HC112、74HC194、74HC161的功能描述，编写每个模块的设计代码、测试平台代码。

3、上述内容均要求完成综合、布局布线及三次仿真，74HC74/74HC112二选一、74HC194/74HC161二选一，完成烧录及接电测试后，**给老师检查**。

四、实验结果和数据处理

1、**所有**模块及测试平台代码清单（关键语句需要有注释）

//74HC74代码

module zjh\_74HC74(Sd,Rd,Clk,D,Q);

input Sd,Rd,Clk,D;

output Q;

reg Q;

assign Q\_n=~Q;

always @(negedge Sd,negedge Rd,posedge Clk)

begin

if(!Sd) Q<=1;

else if(!Rd) Q<=0;

else Q<=D;

end

endmodule

//74HC74测试平台代码

module test\_zjh\_74HC74;

reg Sd,Rd,Clk,D;

wire Q;

zjh\_74HC74 u9(Sd,Rd,Clk,D,Q);

initial

begin

Clk=0;

repeat(100)

#5 Clk=~Clk;

end

initial

begin Sd=0;

#22 Sd=1;

#20 Rd=0;

#20 Rd=1;

end

initial

begin

D=0;

#22 D=1;

end

endmodule

//74HC112代码

module zjh\_74HC112(Set\_N, Rst\_N, Clk\_N, J, K, Q, Qn);

input Set\_N, Rst\_N, Clk\_N, J, K;

output Q, Qn;

reg Q;

assign Qn = ~Q;

always @(negedge Set\_N or negedge Rst\_N or negedge Clk\_N)

case({Set\_N,Rst\_N})

0: Q <= 1;

1: Q <= 1;

2: Q <= 0;

default:

case({J,K})

0: Q <= Q;

1: Q <= 0;

2: Q <= 1;

default: Q <= ~Q;

endcase

endcase

endmodule

//74HC112测试平台代码

module test\_zjh\_74HC112;

reg set, res, clk, j, k;

wire q, qn;

zjh\_74HC112 u13(set, res, clk, j, k, q, qn);

always

#5 clk = ~clk;

task clock;

repeat(3)

begin

j = 0;k = 1;#20; j = 1;k = 0;#20;

j = 0;k = 0;#20; j = 1;k = 1;#20;

end

endtask

initial

begin

clk = 0;

set = 0;res = 0;clock;

set = 0;res = 1;clock;

set = 1;res = 0;clock;

set = 1;res = 1;clock;

end

endmodule

//74HC161代码

module zjh\_74HC161(MR,Clk,Cep,Cet,PE,D,Q,TC);

input MR,Clk,Cep,Cet,PE;

input [3:0]D;

output [3:0]Q;

output TC;

reg [3:0]Q;

always @(posedge Clk,negedge MR)

if(!MR) Q<=0;

else if(!PE)Q<=D;

else if(Cep & Cet)Q<=Q+1;//else Q<=Q;

assign TC=&{Cet,Q};

endmodule

//74HC161测试平台代码

module test\_zjh\_74HC161;

reg MR,Clk,Cep,Cet,PE;

reg[3:0]D;

wire[3:0]Q;

wire TC;

zjh\_74HC161 u5(MR,Clk,Cep,Cet,PE,D,Q,TC);

initial

begin

Clk=0;

repeat(100)

#5 Clk=~Clk;

end

initial

begin MR=0;

#22 MR=1;

end

initial

begin

PE=1;Cep=1;Cet=1;D=0;

#58 PE=0;

#20 PE=1;

#50 Cep=0;

#20 Cep=1;

#50 Cet=0;

#20 Cet=1;

end

endmodule

//74HC194代码

module zjh\_74HC194(MR\_N,S,D,Clk,In,Out);

input MR\_N,Clk;

input [1:0]S;

input [1:0]D;

input [0:3]In;

output reg [0:3]Out;

always@(posedge Clk or negedge MR\_N)

if(~MR\_N) Out<=0;

else case(S)

2'b00:Out<=Out;

2'b01:if(D[1]) Out<={1'b1,Out[0:2]};

else Out<=Out>>1;

2'b10:if(D[0]) Out<={Out[1:3],1'b1};

else Out<=Out<<1;

default: Out<=In;

endcase

endmodule

//74HC194测试平台代码

module test\_zjh\_74HC194;

reg MR\_N,Clk;

reg [1:0]S;

reg [1:0]D;

reg [0:3]In;

wire [0:3]Out;

zjh\_74HC194 u14(MR\_N,S,D,Clk,In,Out);

always #5 Clk=~Clk;

task s\_clock;

begin

D=0;

repeat(4)

#10 D=D+1;

end

endtask

task clock;

begin

S=2'b11; s\_clock;

S=2'b00; s\_clock;

S=2'b01; s\_clock;

S=2'b10; s\_clock;

end

endtask

initial

begin

Clk=1; In=4'b0110; clock;

end

initial

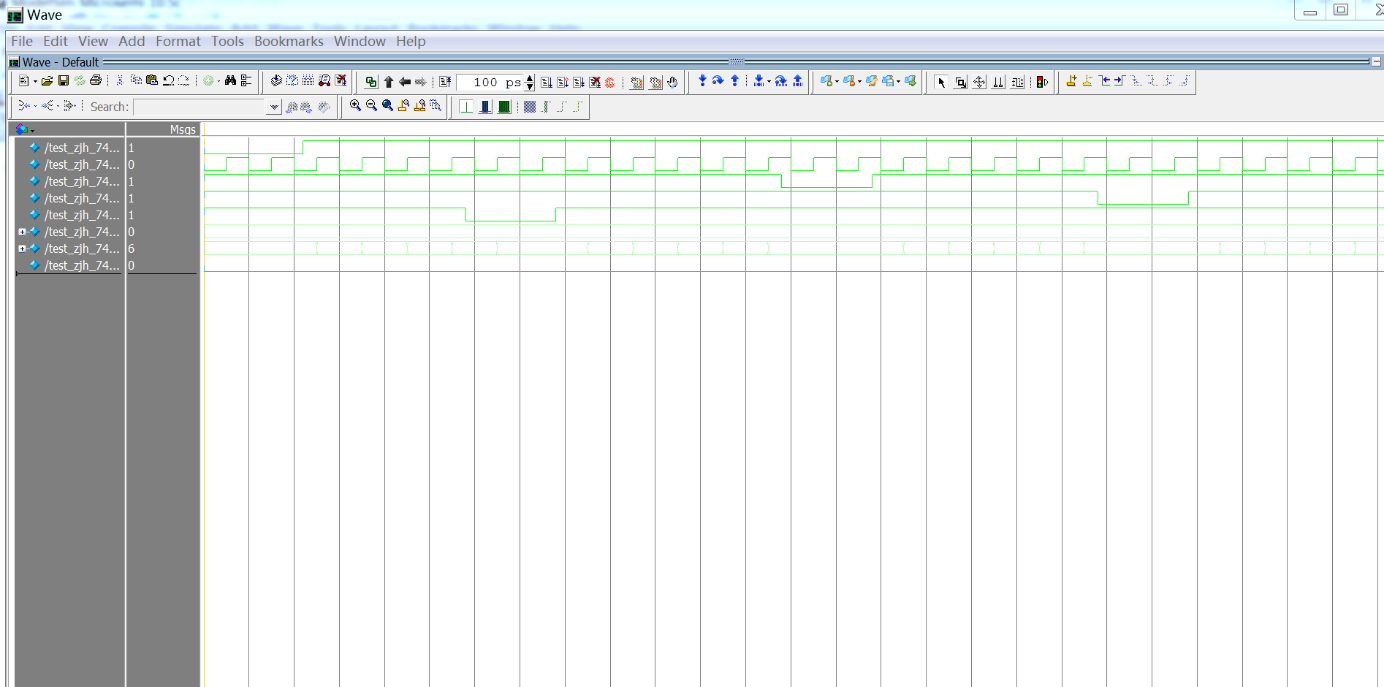
begin

MR\_N=1; #5 MR\_N=0; #10 MR\_N=1;

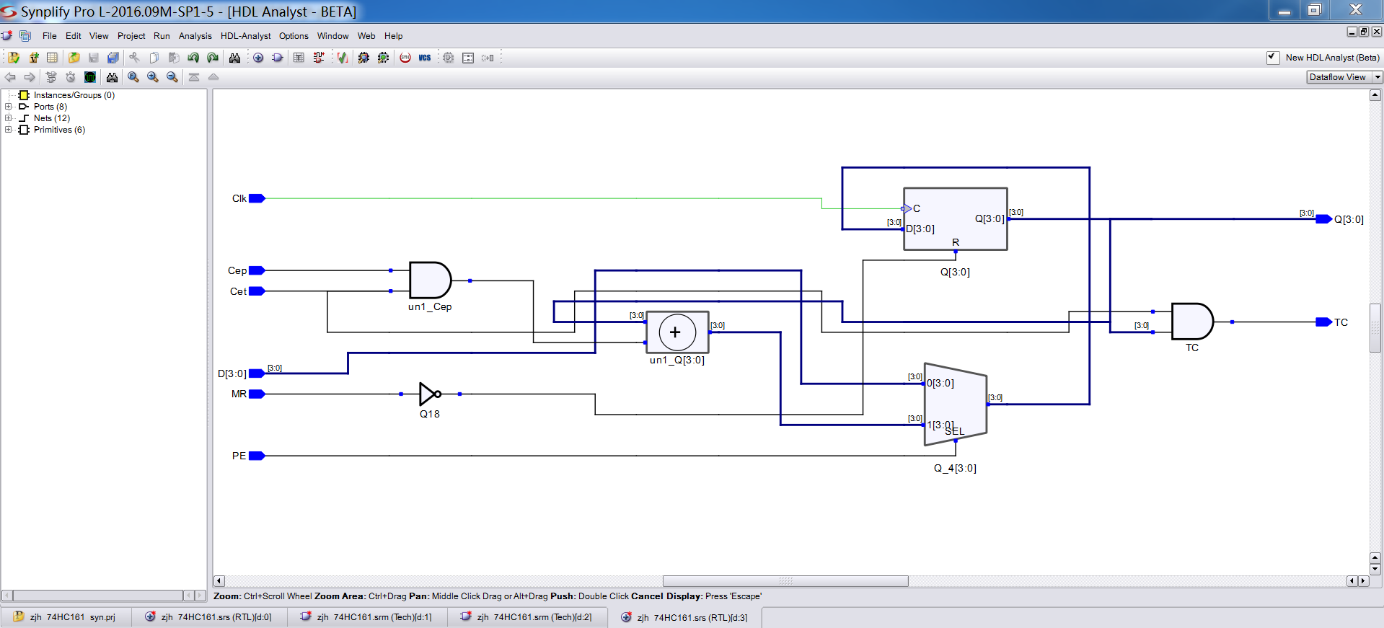
end

endmodule

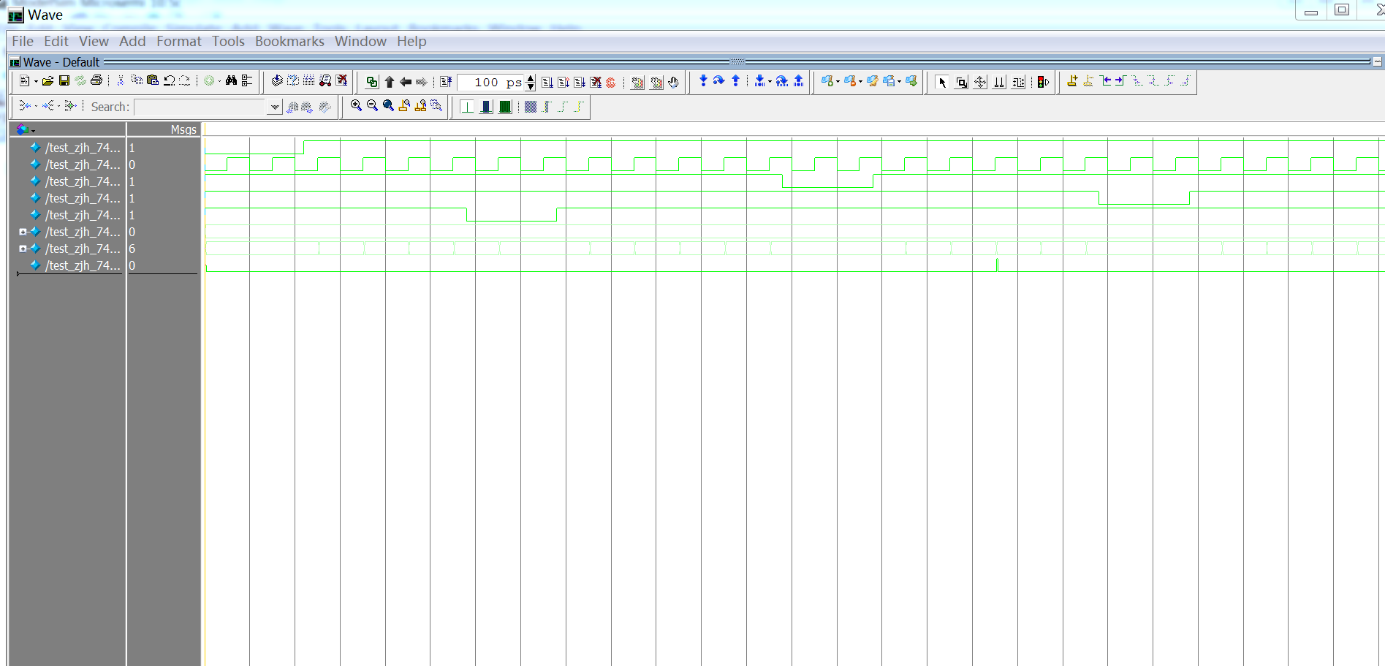
2、第一次仿真结果（电路名称：74HC161）



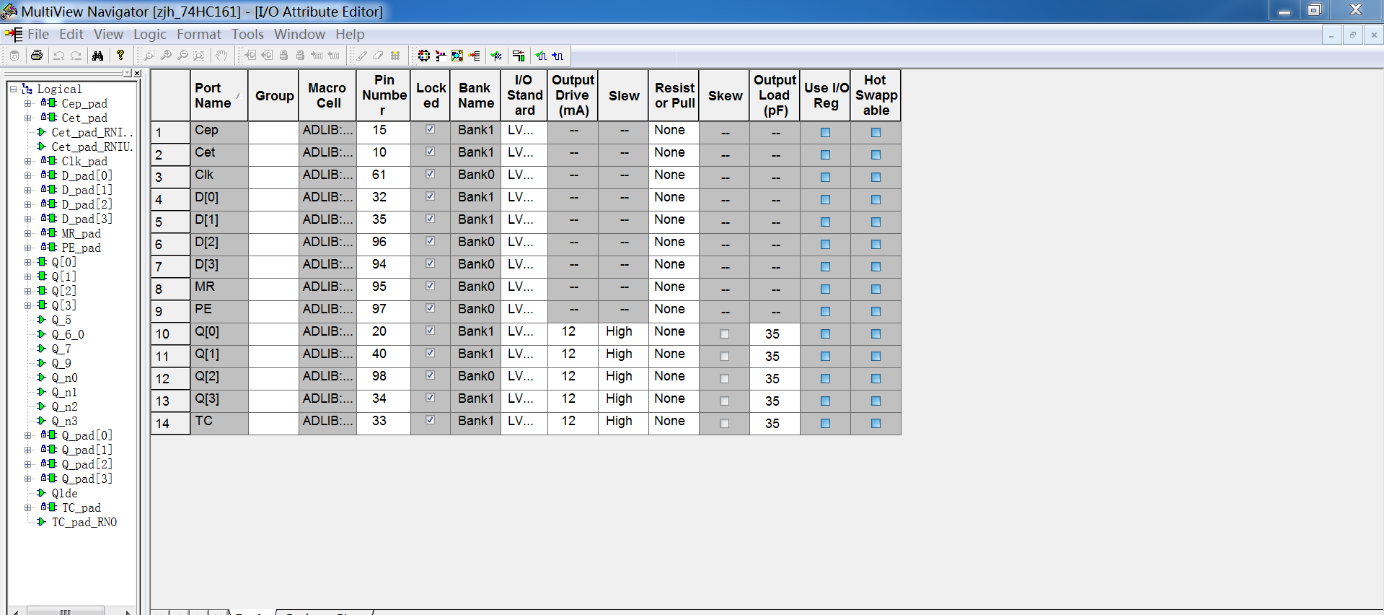
3、综合结果



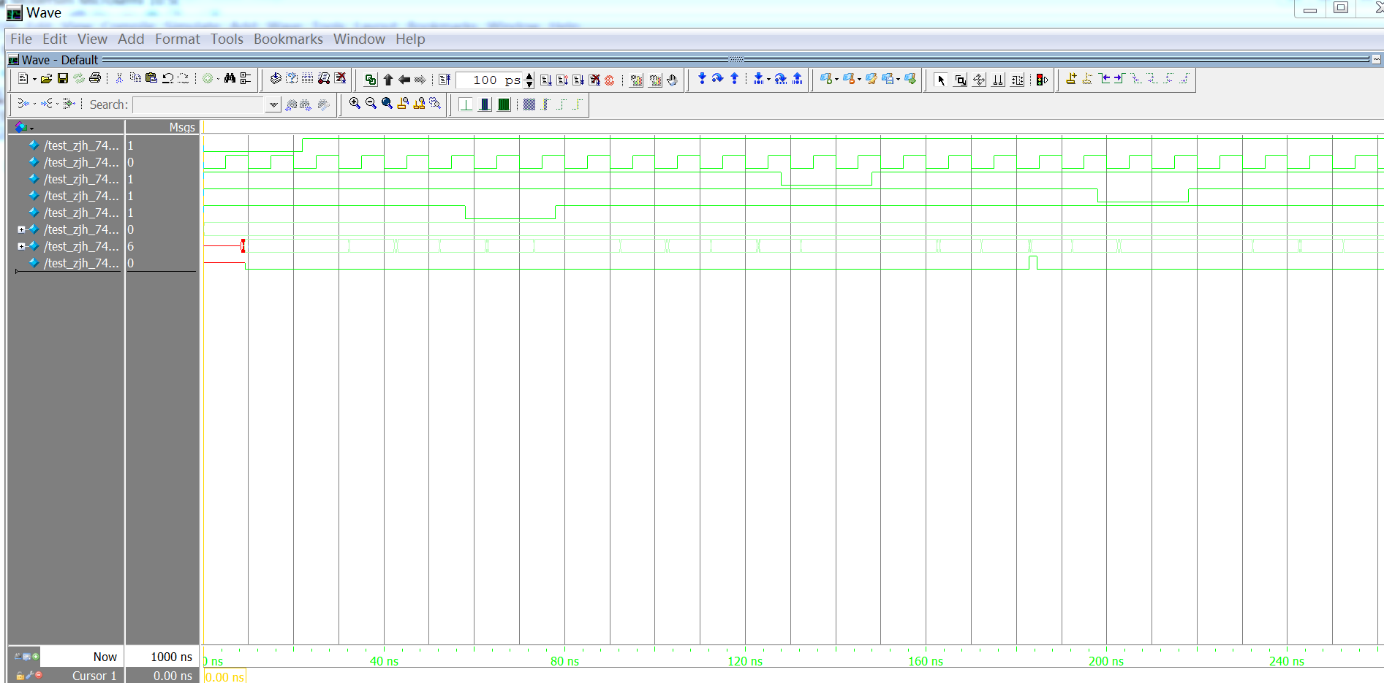
4、第二次仿真结果（综合后）



5、布局布线（引脚分配截图）



6、第三次仿真结果（布局布线后）。



**实验报告**

计算机 学院 计算机科学与技术 专业2020级3班 .

学号 3120005043 姓名 张俊鸿 .

实验题目 基于Libero Soc的数字逻辑综合设计实验

一、实验目的

1、进一步熟悉利用EDA工具进行设计及仿真的流程。

2、熟悉利用EDA工具中的图形化设计界面进行综合设计。

3、熟悉芯片烧录的流程及步骤。

4、掌握分析问题、解决问题的综合能力，通过EDA工具设计出能解决实际问题的电路。

二、实验环境

1、Libero仿真软件。

2、数字逻辑与系统设计实验箱。

3、Actel A3P060 FPGA芯片及Flash Pro5烧录器。

三、实验内容

以下综合实验使用SmartDesign或HDL代码实现，其中设计模块名及测试平台模块名的命名规则如下：

设计模块名举例：**zxt\_CE0**（表示张小童综合题0）

测试平台模块名举例：**test\_ zxt\_CE0**（表示**zxt\_CE0**的测试平台模块）

**0、使用SmartDesign设计N进制计数器。要求如下：**

使用已设计的74HC161模块，及IP核中Macro Library库中的门电路，使用清零法设计N进制计数器。学号尾数与N值的关系如下：

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 学号尾数 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| N | 十 | 十一 | 十三 | 十四 | 四 | 五 | 六 | 七 | 八 | 九 |

要求布线时，将除Clk以外的输入、输出端口布线置小实验箱中的开关及Led灯上。

设计完成后填写以下内容：

（1）编写测试平台代码

module test\_zjh\_CE0;

reg MR,Clk;

wire Q0,Q1,Q2,Q3;

wire C;

zjh\_CE0 u0(.MR(MR),.Clk(Clk),.Q0(Q0),.Q1(Q1),.Q2(Q2),.Q3(Q3),.C(C));

initial

begin

Clk=0;

repeat (100)

#5 Clk=~Clk;

end

initial

begin

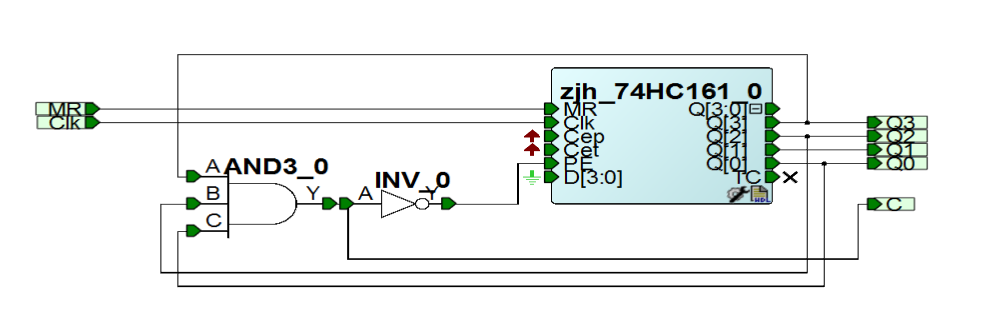
MR=0;

#22 MR=1;

end

endmodule

（2）将SmartDesign画布中的设计截图。



（3）SmartDesign保存后生成的代码

`timescale 1ns / 100ps

module zjh\_CE0(

// Inputs

Clk,

MR,

// Outputs

C,

Q0,

Q1,

Q2,

Q3

);

input Clk;

input MR;

output C;

output Q0;

output Q1;

output Q2;

output Q3;

wire C\_net\_0;

wire Clk;

wire INV\_0\_Y;

wire MR;

wire [0:0] Q0\_net\_0;

wire [1:1] Q1\_net\_0;

wire [2:2] Q2\_net\_0;

wire [3:3] Q3\_net\_0;

wire Q2\_net\_1;

wire Q0\_net\_1;

wire Q3\_net\_1;

wire Q1\_net\_1;

wire C\_net\_1;

wire [3:0] Q\_net\_0;

wire VCC\_net;

wire [3:0] D\_const\_net\_0;

assign VCC\_net = 1'b1;

assign D\_const\_net\_0 = 4'h0;

assign Q2\_net\_1 = Q2\_net\_0[2];

assign Q2 = Q2\_net\_1;

assign Q0\_net\_1 = Q0\_net\_0[0];

assign Q0 = Q0\_net\_1;

assign Q3\_net\_1 = Q3\_net\_0[3];

assign Q3 = Q3\_net\_1;

assign Q1\_net\_1 = Q1\_net\_0[1];

assign Q1 = Q1\_net\_1;

assign C\_net\_1 = C\_net\_0;

assign C = C\_net\_1;

assign Q0\_net\_0[0] = Q\_net\_0[0:0];

assign Q1\_net\_0[1] = Q\_net\_0[1:1];

assign Q2\_net\_0[2] = Q\_net\_0[2:2];

assign Q3\_net\_0[3] = Q\_net\_0[3:3];

AND3 AND3\_0(

// Inputs

.A ( Q3\_net\_0 ),

.B ( Q2\_net\_0 ),

.C ( Q0\_net\_0 ),

// Outputs

.Y ( C\_net\_0 )

);

INV INV\_0(

// Inputs

.A ( C\_net\_0 ),

// Outputs

.Y ( INV\_0\_Y )

);

zjh\_74HC161 zjh\_74HC161\_0(

// Inputs

.MR ( MR ),

.Clk ( Clk ),

.Cep ( VCC\_net ),

.Cet ( VCC\_net ),

.PE ( INV\_0\_Y ),

.D ( D\_const\_net\_0 ),

// Outputs

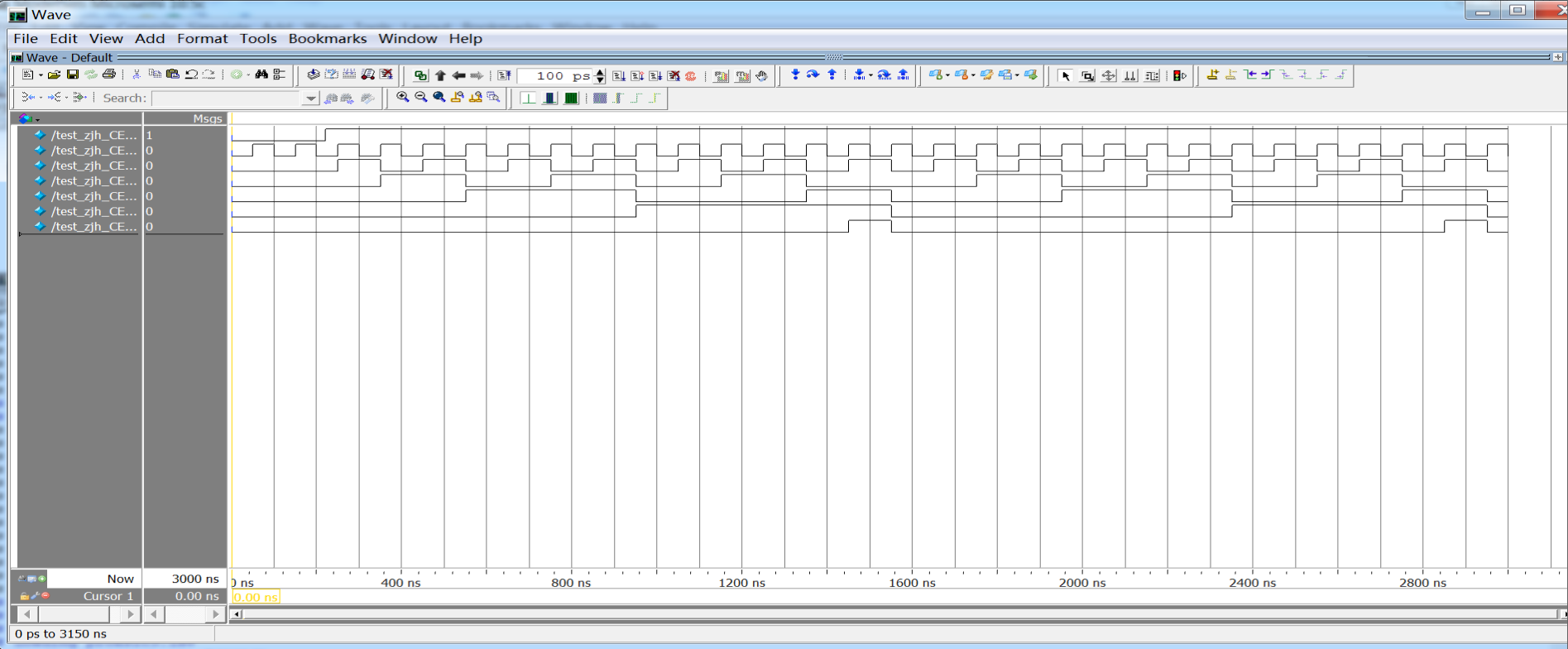
.Q ( Q\_net\_0 ),

.TC ( )

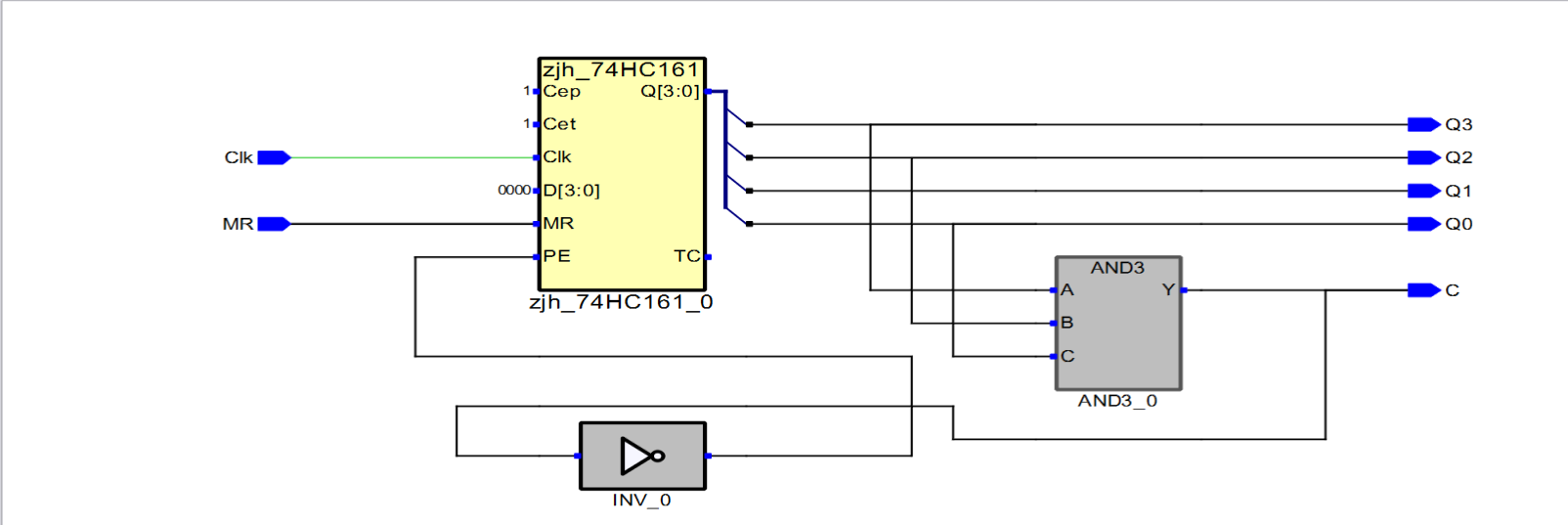
);

endmodule

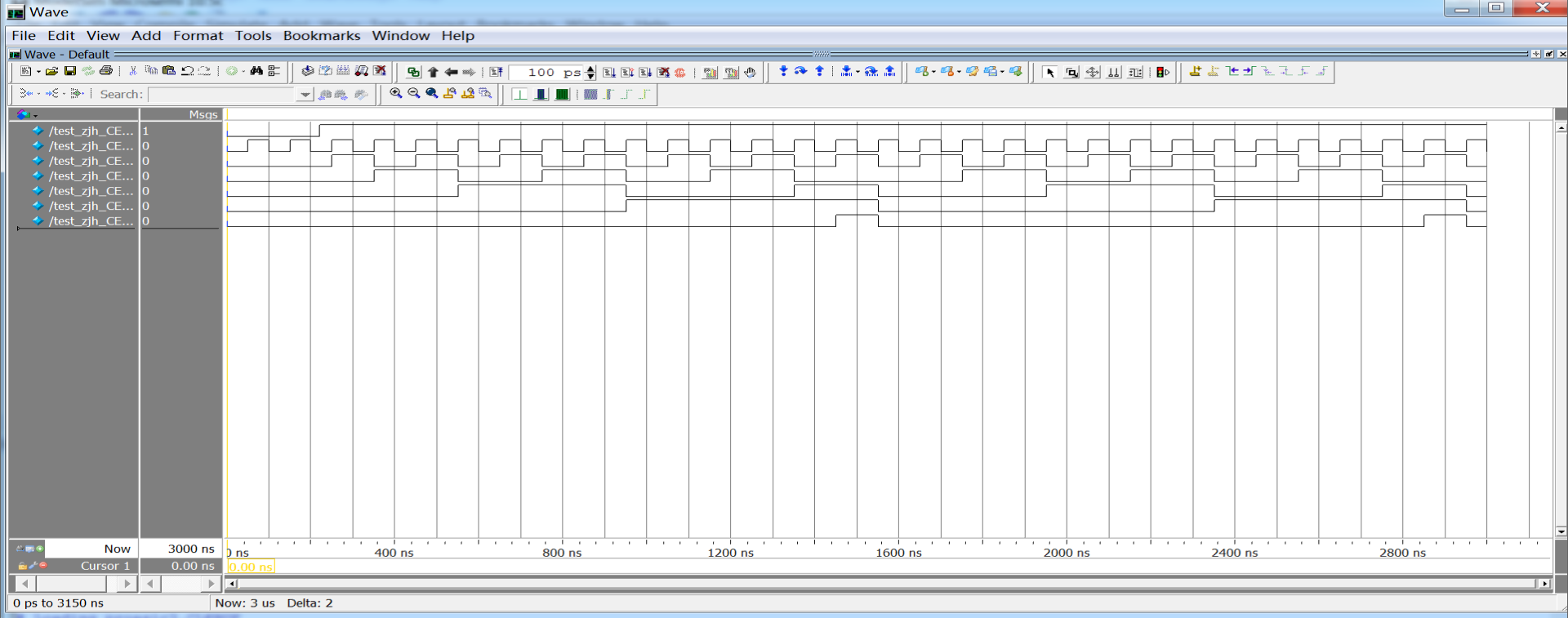
（4）综合前仿真截图



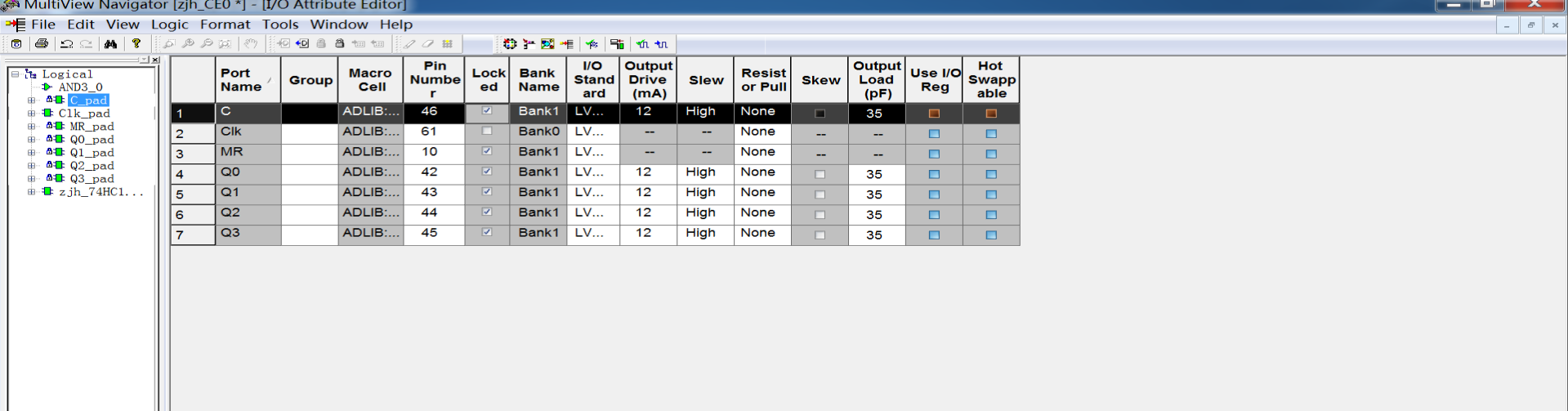
（5）综合截图



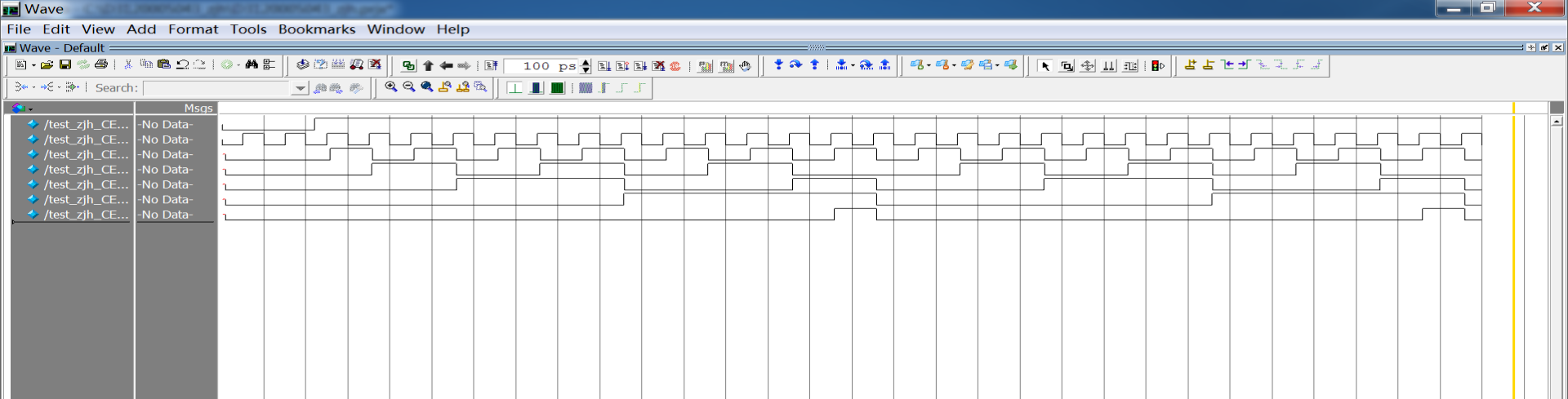
（6）综合后仿真截图



（7）布局布线引脚分配截图



（8）布局布线后仿真截图



**1、SmartDesign设计：使用SmartDesign工具进行设计，要求如下：**

使用已设计的74HC138、74HC4511模块，及IP核中Basic Blocks库中的计数器模块（counter），在SmartDesign画布中设计下图框中的模块。



设计完成后填写以下内容：

（1）编写测试平台代码

module test\_zjh\_CE1;

reg Aclr;

reg Clock;

wire a,b,c,d,e,f,g;

wire Y4,Y3,Y2,Y1;

zjh\_CE1 u1(.Aclr(Aclr),.Clock(Clock),.a(a),.b(b),.c(c),.d(d),.e(e),.f(f),.g(g),.Y4(Y4),.Y3(Y3),.Y2(Y2),.Y1(Y1));

initial

begin

Clock =0;

repeat(100)

#5 Clock=~Clock;

end

initial

begin

Aclr=0;

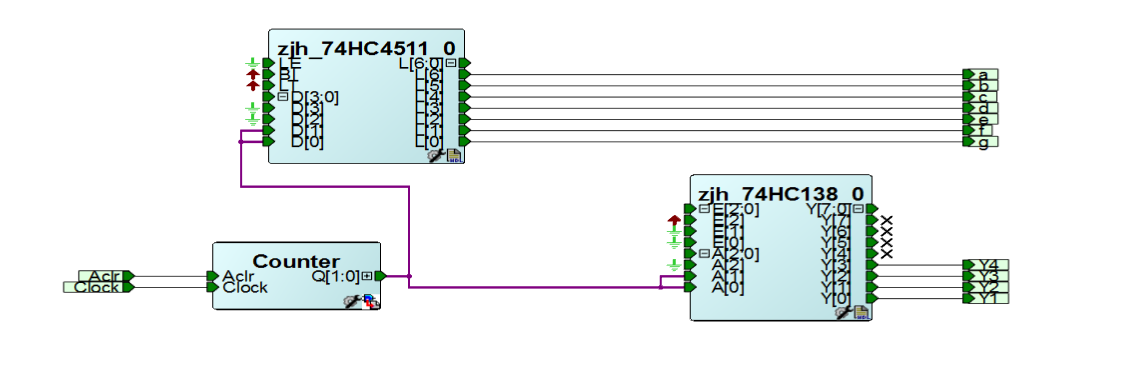
#22 Aclr=1;

#100 ;$finish;

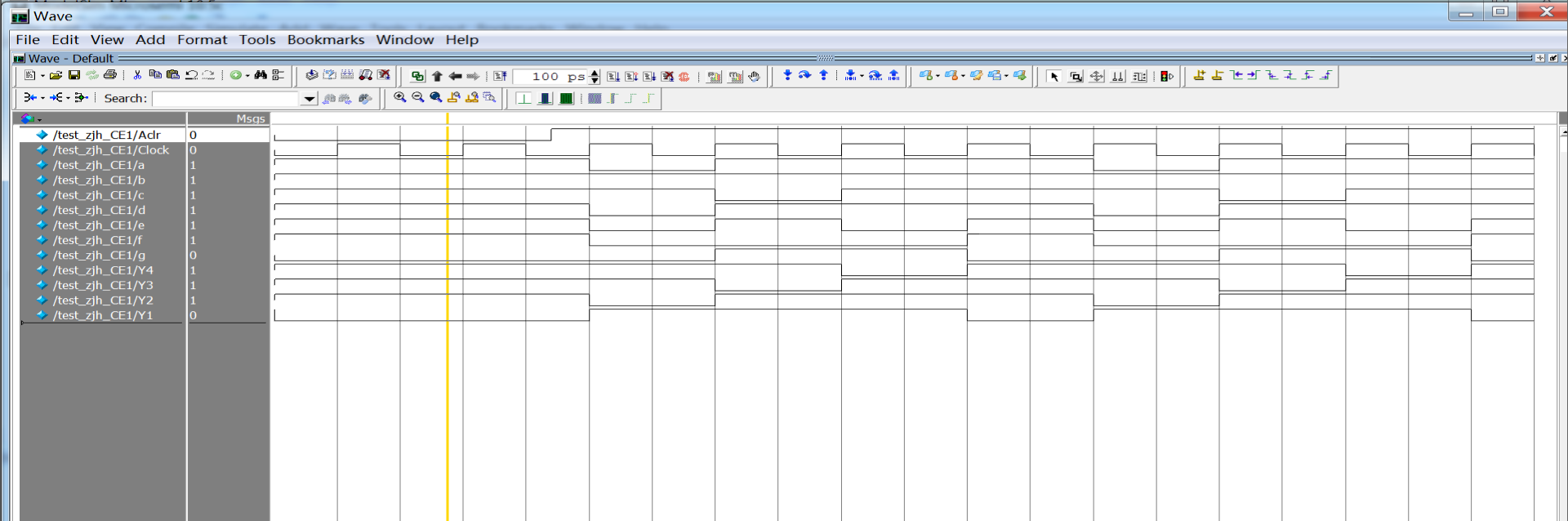
end

endmodule

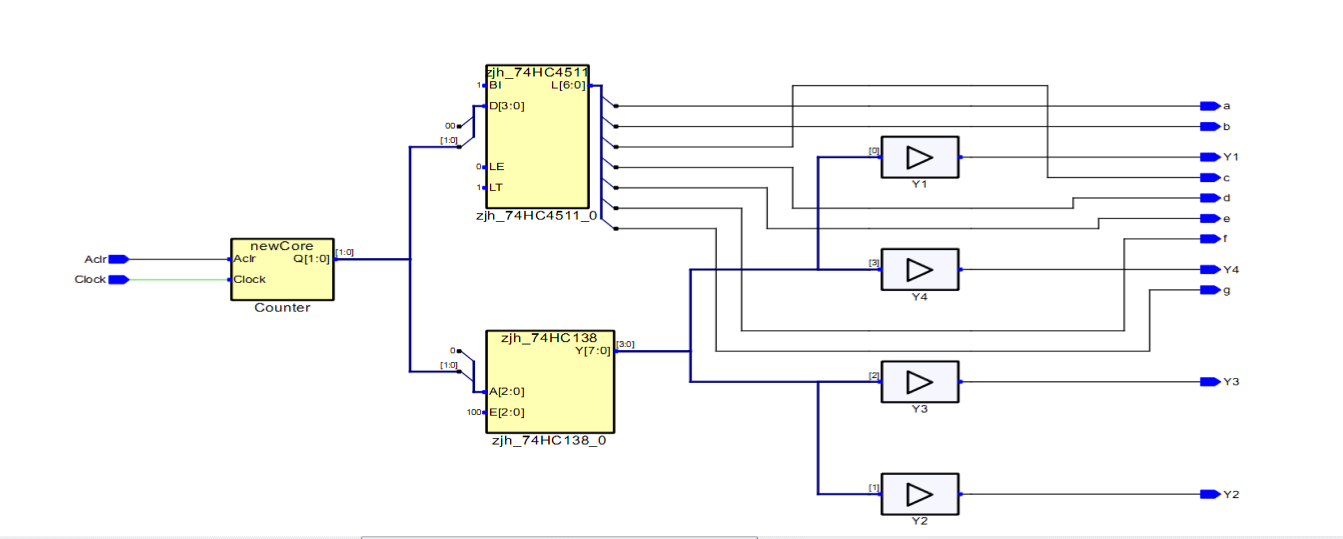
（2）将SmartDesign画布中的设计截图。



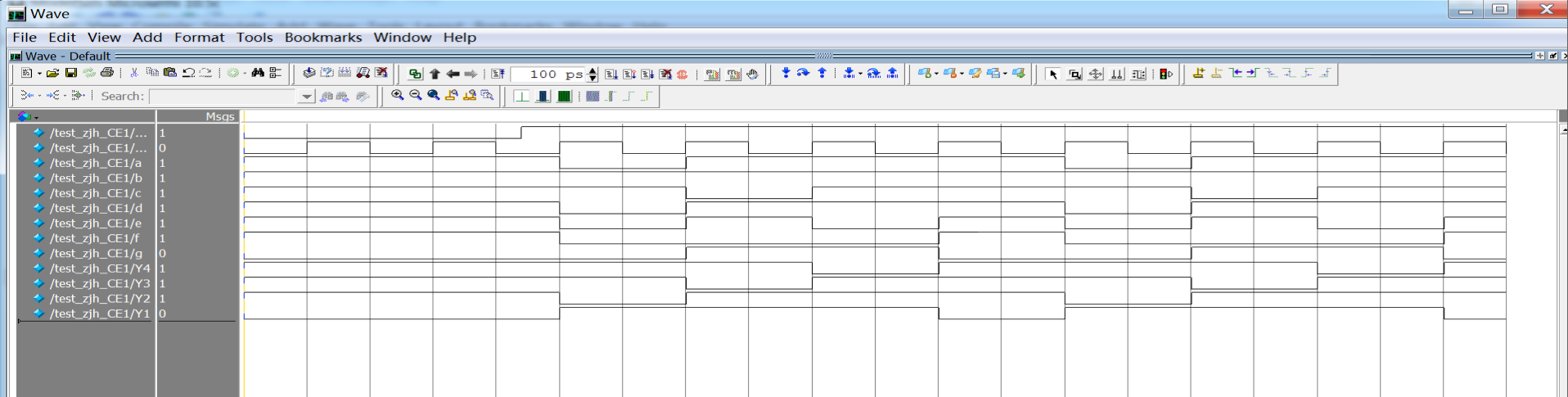
（3）综合前仿真截图



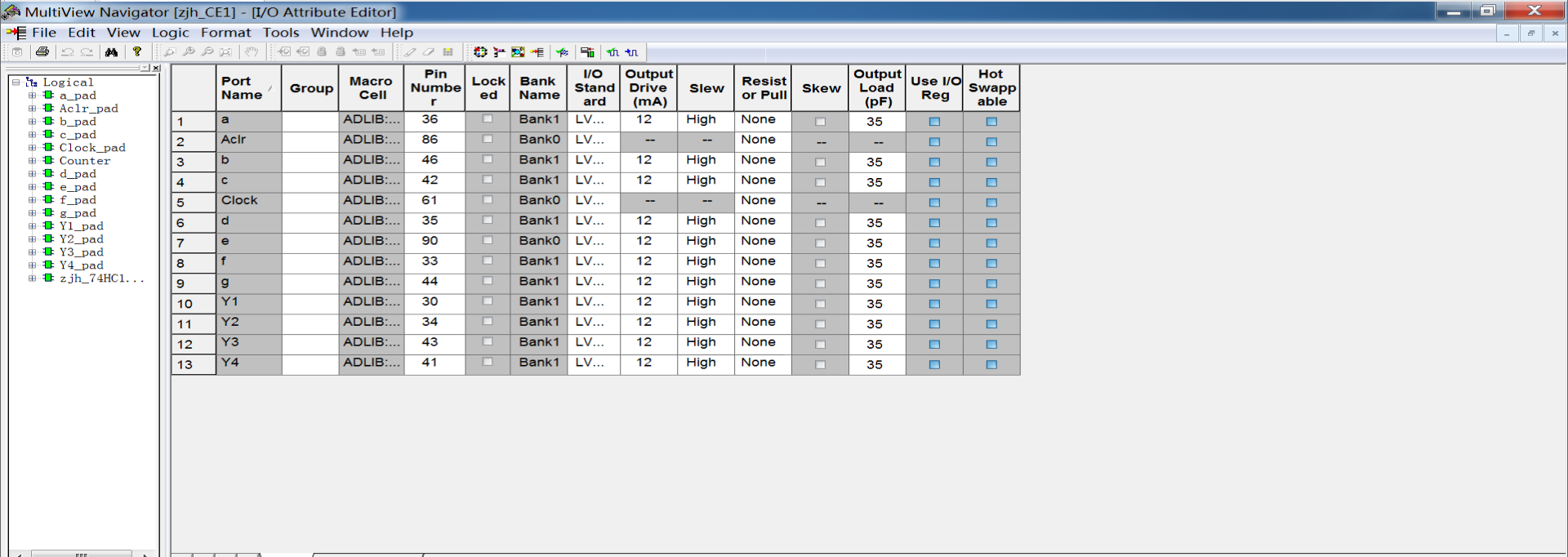
（4）综合截图



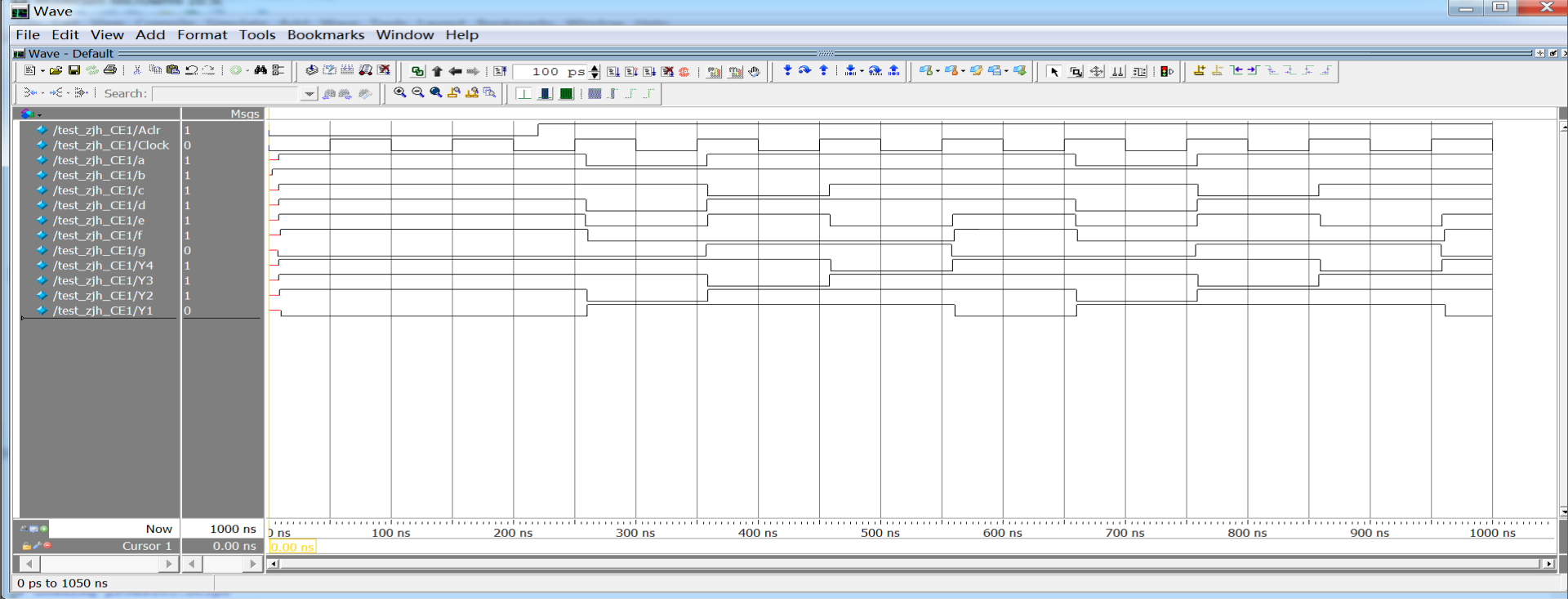
（5）综合后仿真截图



（6）布局布线引脚分配截图



（7）布局布线后仿真截图



**2、SmartDesign设计（选做）：使用SmartDesign工具进行设计，要求如下：**

使用已设计的74HC161、74HC85、74HC4511模块，及IP核中Macro library库中的反相器模块（INV），在SmartDesign画布中设计下图左框中的模块。



设计完成后填写以下内容：

（1）编写测试平台代码

module test\_zjh\_CE2;

reg Clk,MR;

reg A0,A1,A2,A3,B0,B1,B2,B3;

wire a,b,c,d,e,f,g;

zjh\_CE2 u2(.MR(MR),.Clk(Clk),.A0(A0),.A1(A1),.A2(A2),.A3(A3),.B0(B0),.B1(B1),.B2(B2),.B3(B3),.a(a),.b(b),.c(c),.d(d),.e(e),.f(f),.g(g));

initial

begin

Clk=1;

repeat (100)

#5 Clk=~Clk;

end

initial

begin

MR=0;

# 22 MR=1;

#200 ;$finish;

end

initial

begin

A0=0;

A1=1;

A2=0;

A3=0;

B0=1;

B1=0;

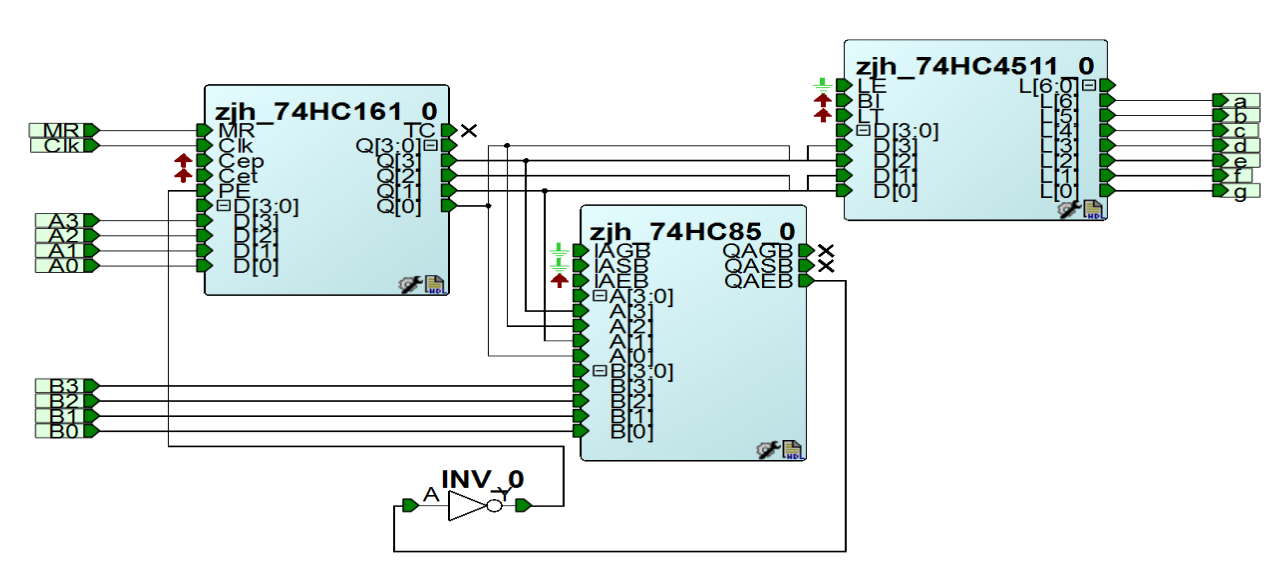
B2=1;

B3=1;

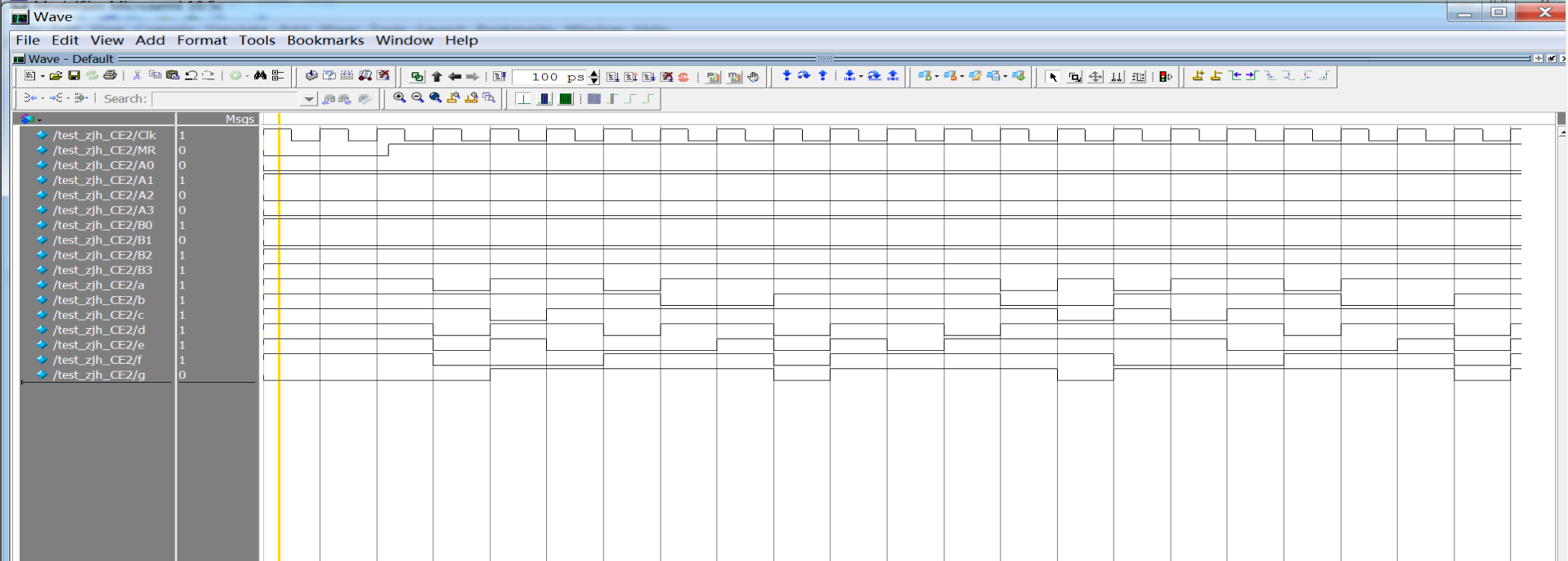
end

endmodule

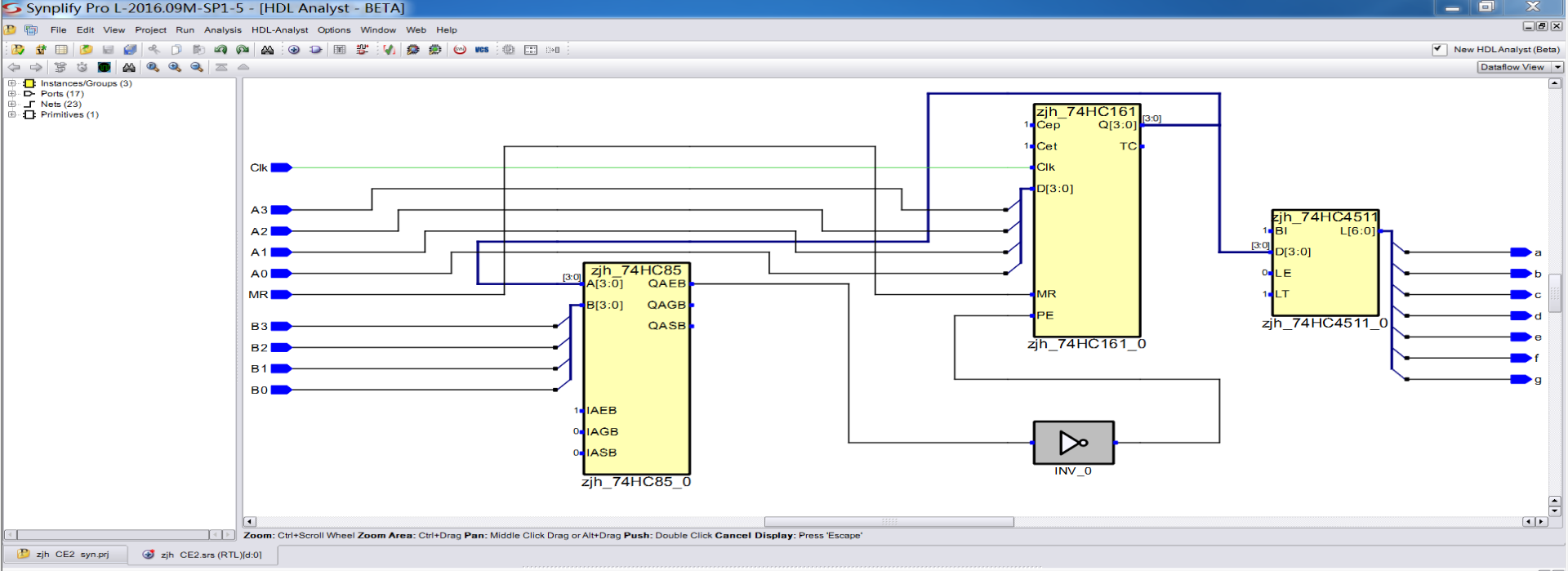
（2）将SmartDesign画布中的设计截图。



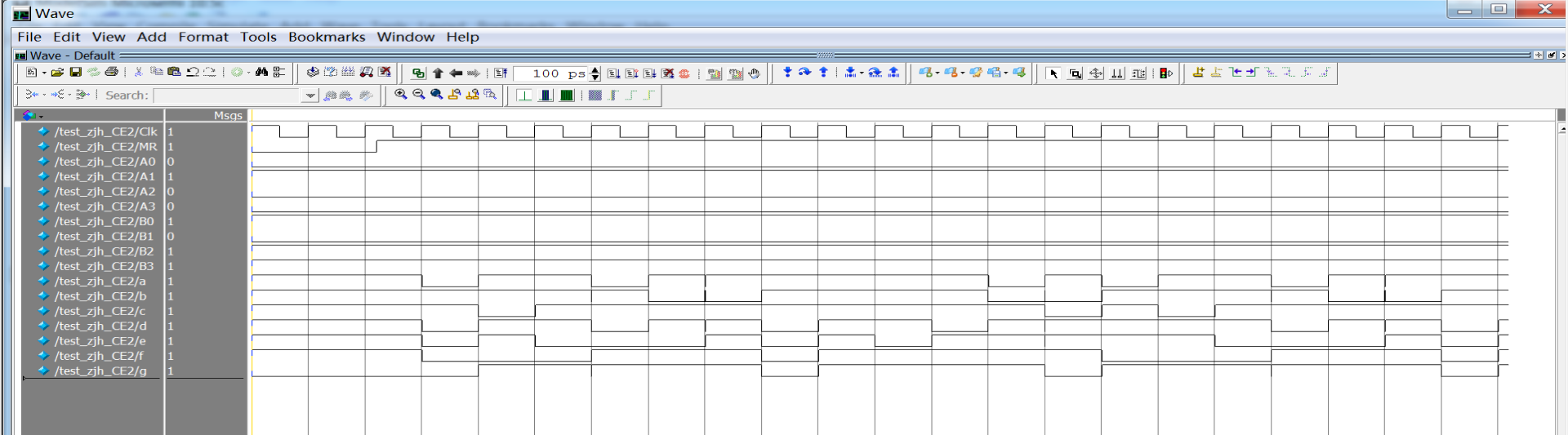
（3）综合前仿真截图



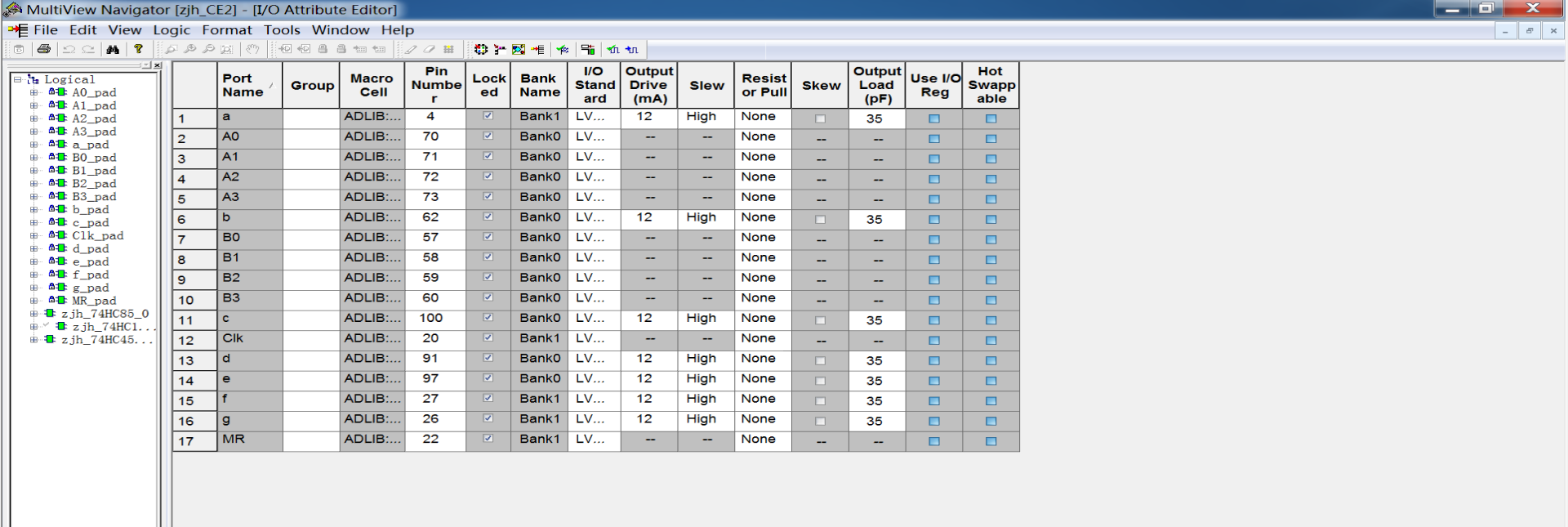
（4）综合截图



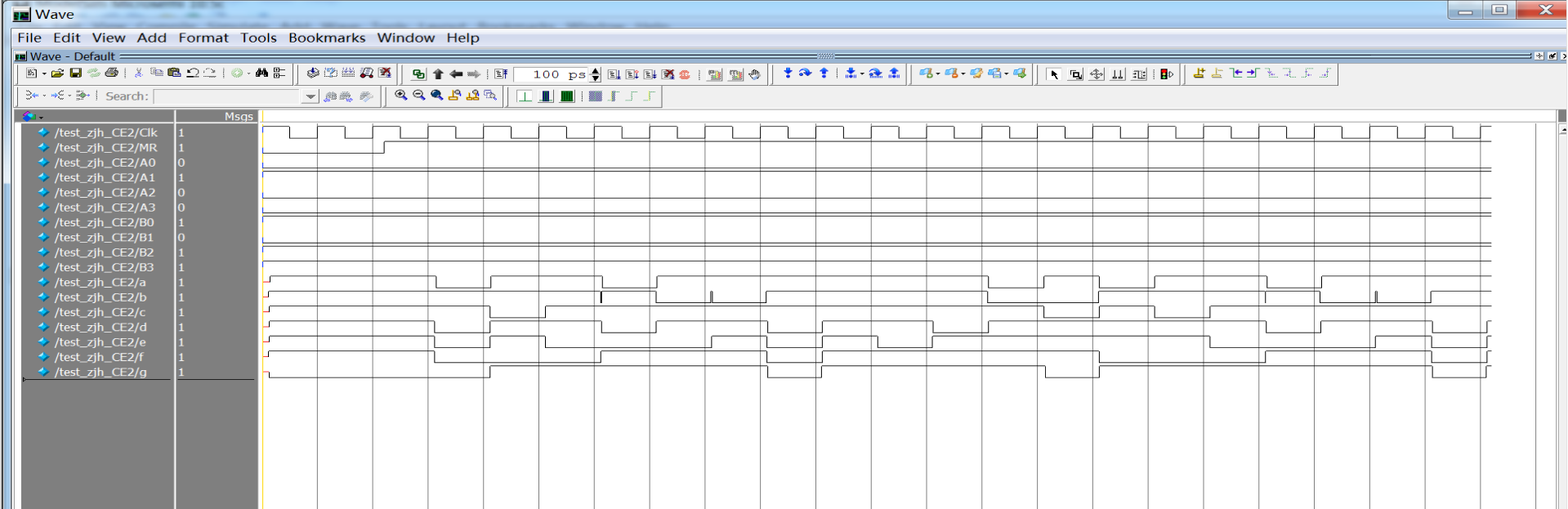
（5）综合后仿真截图



（6）布局布线引脚分配截图



（7）布局布线后仿真截图



**3、学号显示器（选做）**

在以上第1题设计的基础上，增加一个4\*4的寄存器（4组，每组4位），寄存器有2位地址输入端，4位数据输出端。4\*4寄存器的存储内容可以直接写入学号末四位，寄存器的输出值由2位地址输入信号决定。

将寄存器组加入到第1题的设计中，使得在时钟脉冲的控制下，4位显示器依次输出学号末四位。

设计完成后填写以下内容：

（1）编写测试平台代码

module test\_zjh\_CE3;

reg Aclr;

reg Clock;

wire a,b,c,d,e,f,g;

wire Y4,Y3,Y2,Y1;

zjh\_CE3 u3(.Aclr(Aclr),.Clock(Clock),.a(a),.b(b),.c(c),.d(d),.e(e),.f(f),.g(g),.Y4(Y4),.Y3(Y3),.Y2(Y2),.Y1(Y1));

initial

begin

Clock =0;

repeat(100)

#5 Clock=~Clock;

end

initial

begin

Aclr=0;

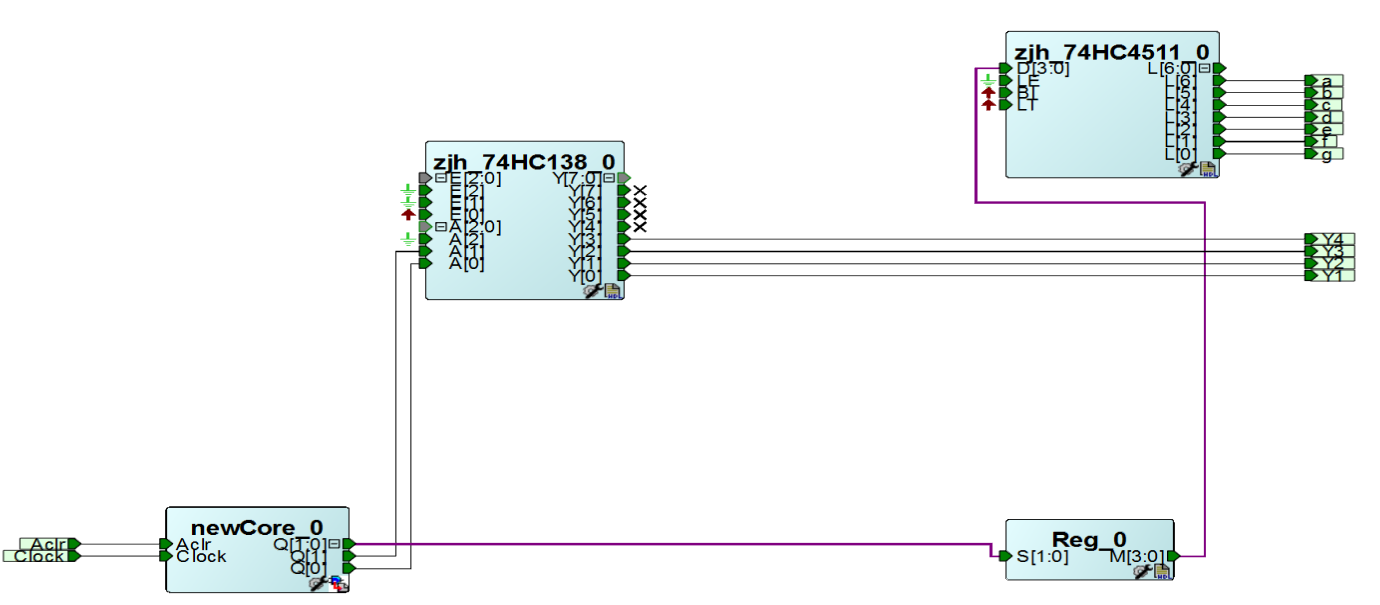
#22 Aclr=1;

#100 ;$finish;

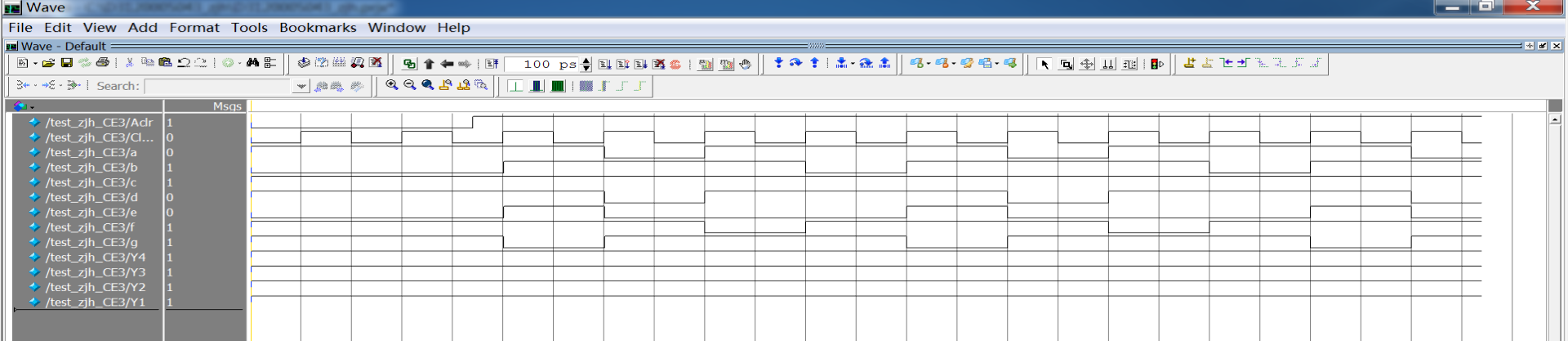
end

endmodule

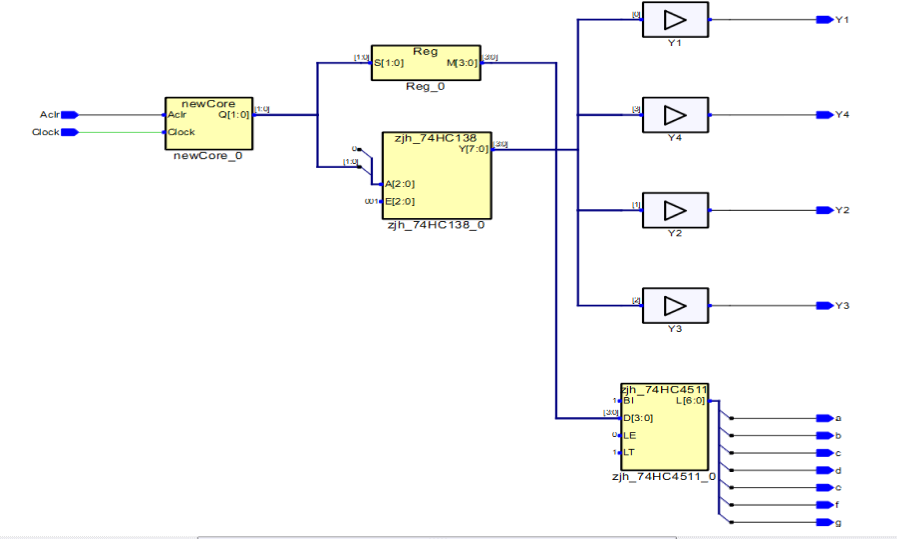
（2）将SmartDesign画布中的设计截图。



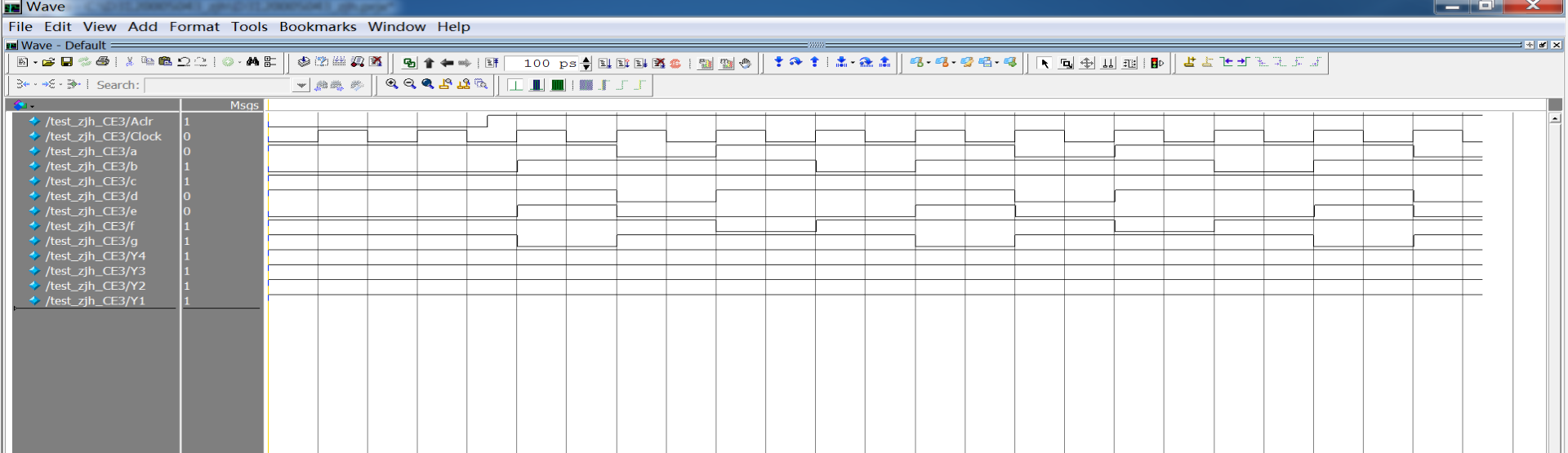
（3）综合前仿真截图



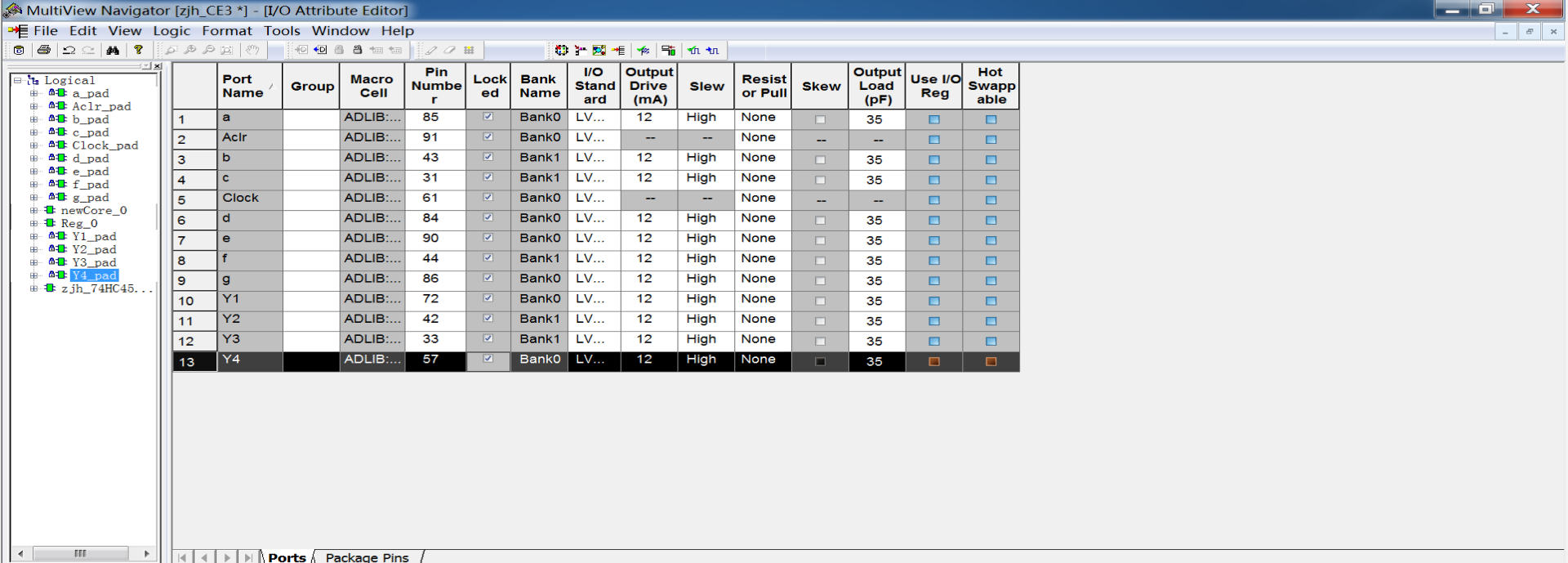
（4）综合截图



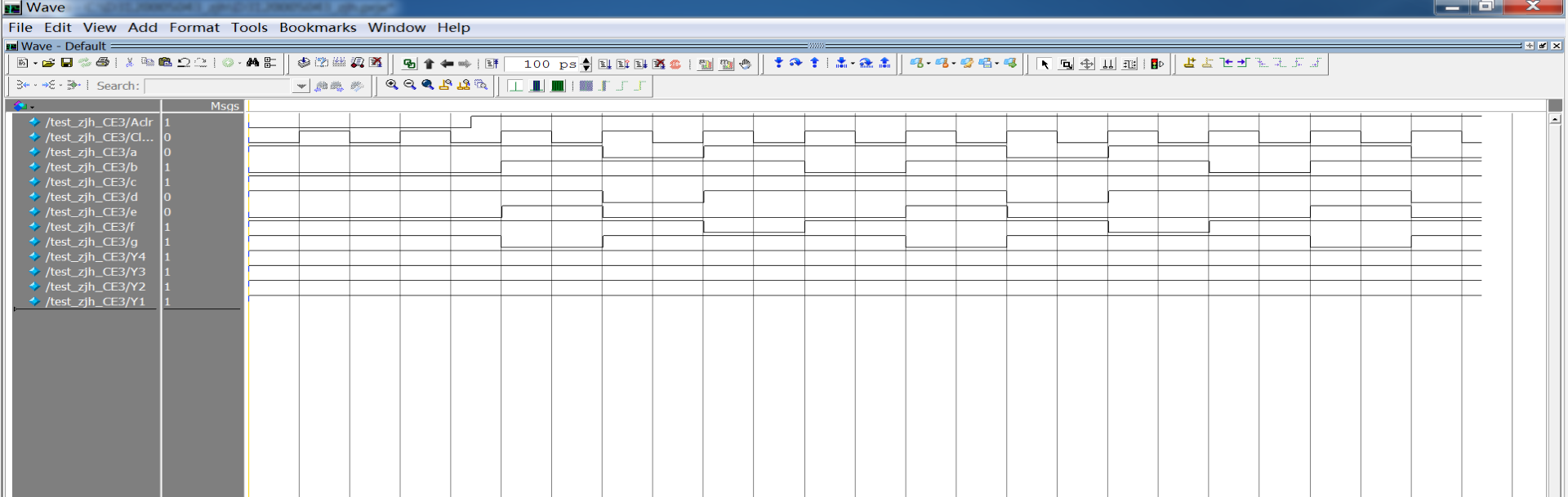
（5）综合后仿真截图



（6）布局布线引脚分配截图



（7）布局布线后仿真截图



**4、有符号数比较器（选做）**

设计一个电路，输入信号是有符号数的原码，输出信号包括A>B、A=B、A<B。要求使用组合逻辑电路实验中设计的原码-补码转换器，以及IP核中Basic Blocks库中的中的比较器（Comparator）。

设计完成后填写以下内容：

（1）编写测试平台代码

module test\_zjh\_CE4;

reg [7:0]Ain,Bin;

wire aeb,alb,agb;

zjh\_CE4 u0(.DataIn(Ain),.DataIn\_0(Bin),.AEB(aeb),.ALB(alb),.AGB(agb));

initial

begin

Ain = 8'b01000000;Bin = 8'b01110000;

#10 Ain = 8'b01100000;

#10 Bin = 8'b11000000;

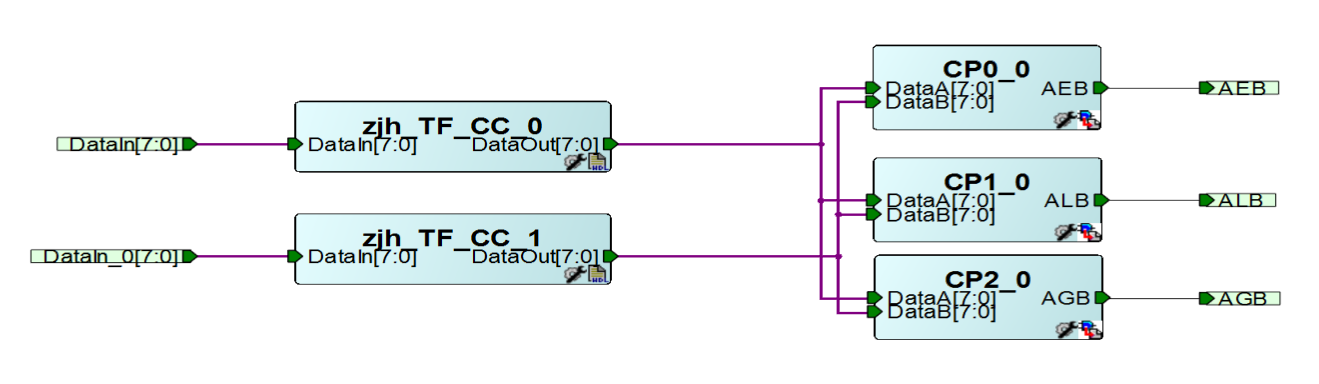
#10 Ain = 8'b10100000;

#10 Bin = 8'b10100000;

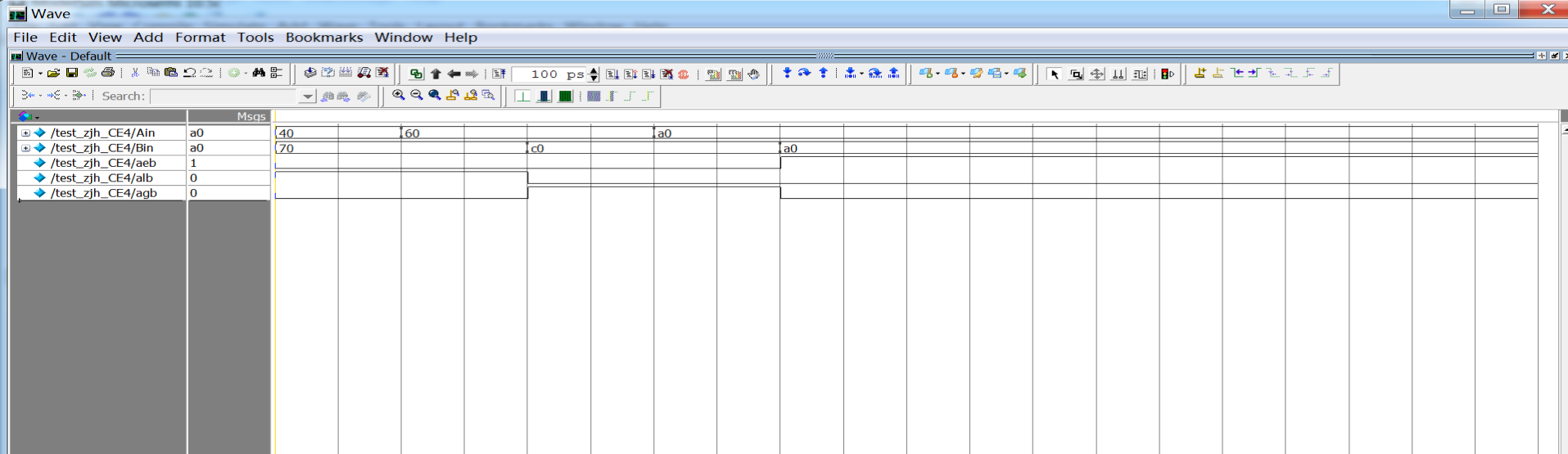
end

endmodule

（2）将SmartDesign画布中的设计截图。



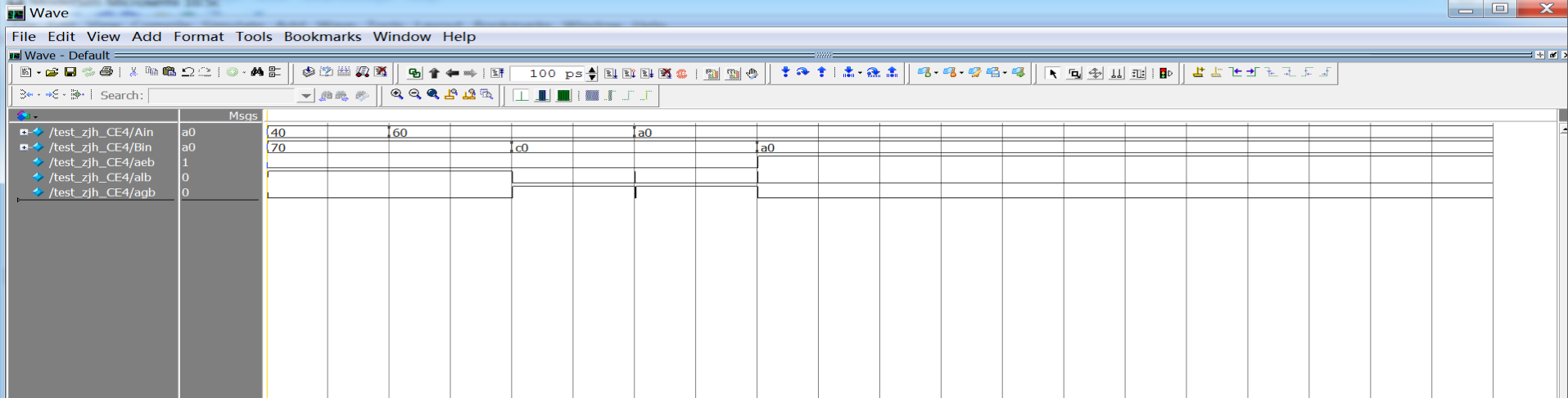
（3）综合前仿真截图



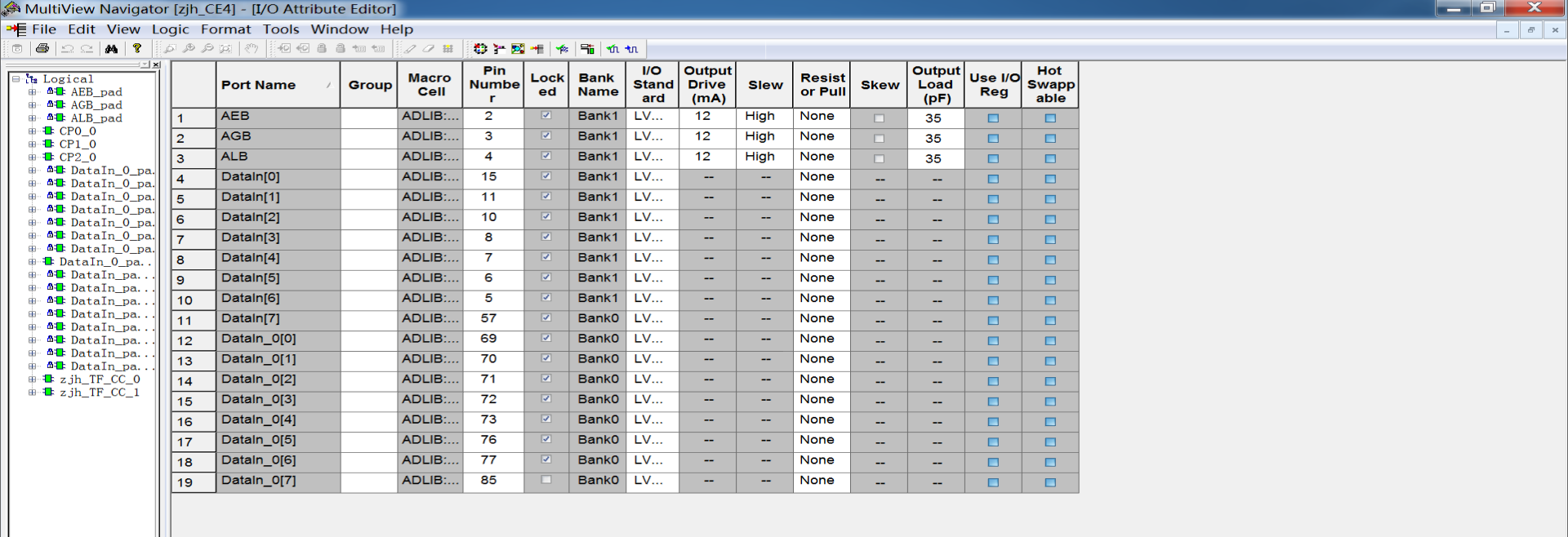
（4）综合截图



（5）综合后仿真截图



（6）布局布线引脚分配截图



（7）布局布线后仿真截图

