|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **学 期** | 2020-2021学年第1学期 | | | | **实验日期** | | 2020.10.13 | | |
| **学 院** | 计算机学院 | | | | **专 业** | | 计算机类 | | |
| **班 级** | 190741 | | **学 号** | | 19030419 | | **姓 名** | | 王昕 |
| **组 号** | 24 | | **学 号** | | 19074104 | | **姓 名** | | 王佳玮 |
| **评 阅 内 容** | | | | | | | | | |
| **实验预习** | | **实验原理** | | **详细设计** | | **实验结果** | | **成 绩** | |
|  | |  | |  | |  | |  | |
|  | | | | | | | | | |
| **题 目** | 实验 9：时序逻辑设计——选做题 | | | | | | | | |

1. **实验目的**

1．通过序列检测器、键盘去抖、寄存器堆等电路的设计，进一步学习时序逻辑电路的设计。

2．进一步掌握运用 Verilog 描述基本时序电路的方法。

3．进一步掌握运用状态机设计时序电路的方法。

1. **实验任务及要求**

**以下任务选作一题。**

**任务（一） 设计一个序列检测器**

**1．设计一个1010序列检测器（不考虑序列重叠）**

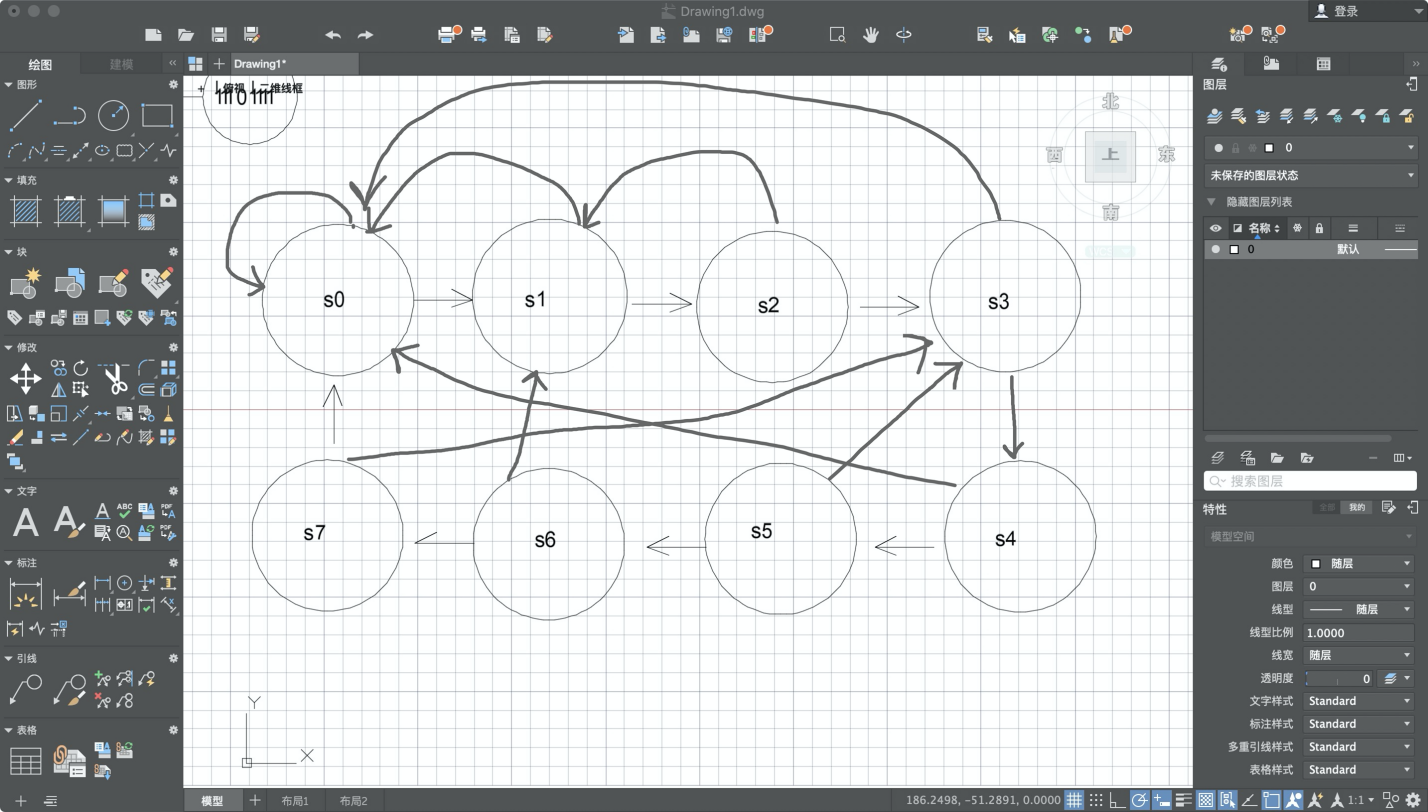
**2．设计一个8位序列的序列检测器。该序列为同组两位同学学号末两位相加，如相加后低于20，需在和的基础上加20（检测序列为结果对应的8421BCD码，如结果为34，检测序列为00110100）。**

**要求：**

**1．以上任务二选一。**

**2．采用Verilog实现状态机。**

**三．实验原理**



本次实验共涉及三个模块。其中输入信号reset, x可控，CLK为时钟脉冲信号。如程序所示，通过控制输入信号x，通过时钟脉冲，当序列00100011出现时，输出z为1。

**四．详细设计**

1、电路实现

module wxwjwjiance(clk,clr,reset,z,in);

input clk,clr,reset;

output z;

output [3:0] in;

wire x;

wxwjwfasheng (clk,clr,x);

wxwjwxljc (clk,x,reset,z,in);

endmodule

module wxwjwfasheng (clk,clr,dout);

input clk,clr;

output dout;

reg [7:0] data;

reg dout;

always@(posedge clk)

begin

if(!clr)

begin

dout<=0;

data<=8'b00100011;

end

else

begin

dout<=data[7];

data<={data[6:0],data[7]};

end

end

endmodule

module wxwjwxljc (clk,x,reset,z,in);

input clk,x,reset;

output z;

output[3:0] in;

reg z;

reg [3:0]in;

parameter s0=0,s1=1,s2=2,s3=3,s4=4,s5=5,s6=6,s7=7;

reg [2:0]cstate,nstate;

always@(posedge clk)

if(!reset)

cstate<=s0;

else

begin

cstate<=nstate;

end

always@(cstate or x)

begin

case(cstate)

s0:if(!x) nstate<=s1; else nstate<=s0;

s1:if(!x) nstate<=s2; else nstate<=s0;

s2:if(x) nstate<=s3; else nstate<=s1;

s3:if(!x) nstate<=s4; else nstate<=s0;

s4:if(!x) nstate<=s5; else nstate<=s0;

s5:if(!x) nstate<=s6; else nstate<=s3;

s6:if(x) nstate<=s7; else nstate<=s1;

s7:if(x) nstate<=s0; else nstate<=s3;

default: nstate<=s0;

endcase

end

always@(posedge clk)

begin

case(cstate)

s7:if(x) begin z=1'b1; in=in+4'b0001; end else z=1'b0;

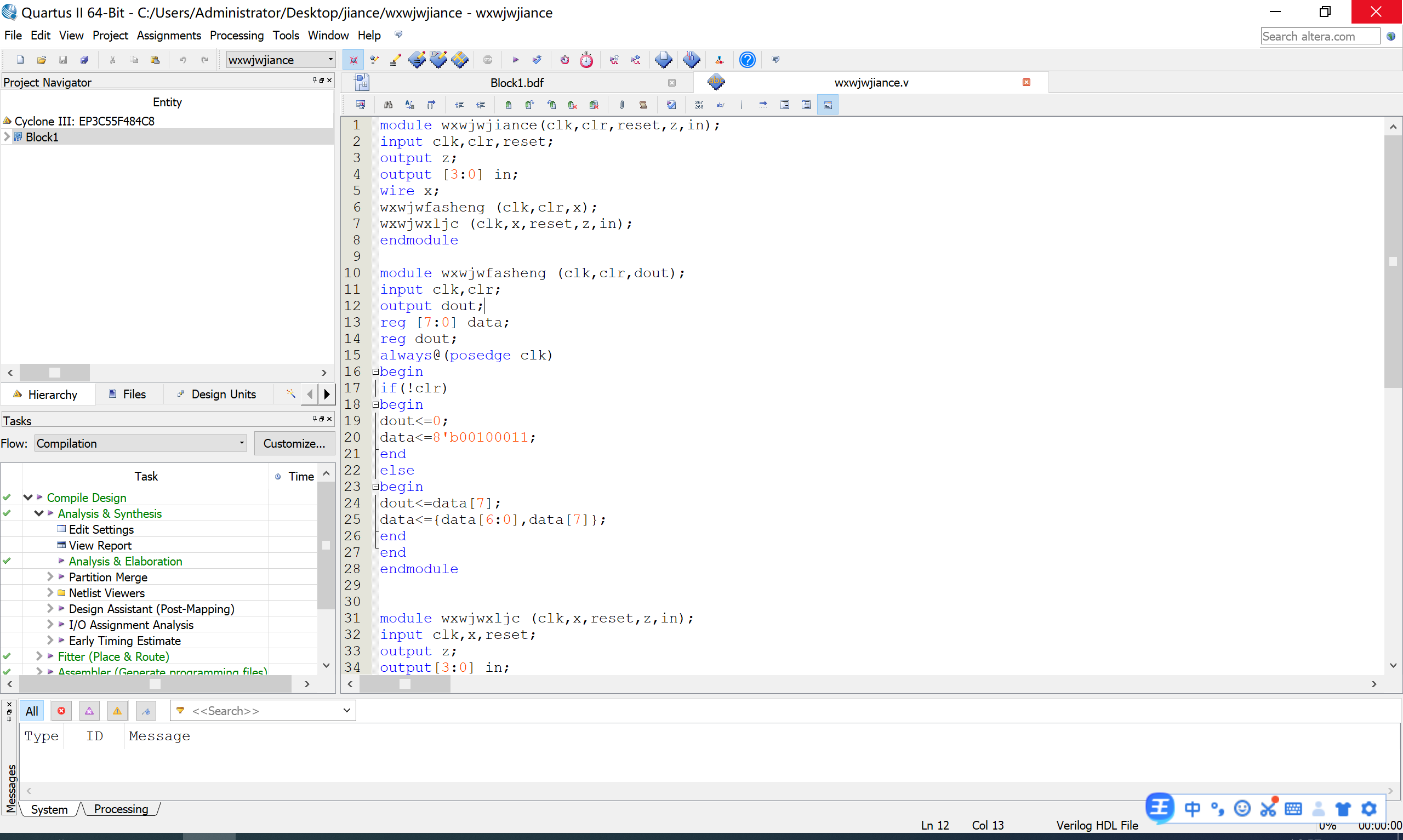
default: z=1'b0;

endcase

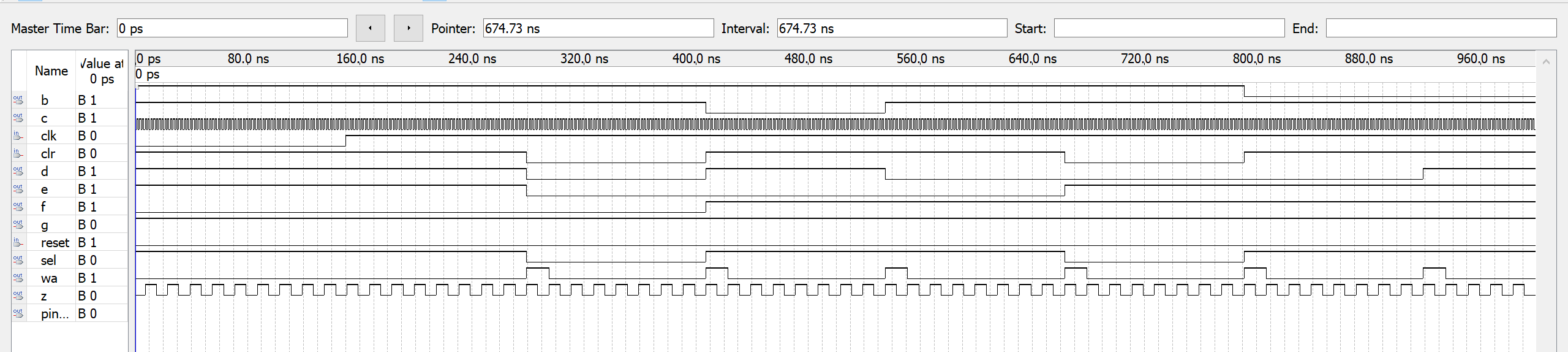
end

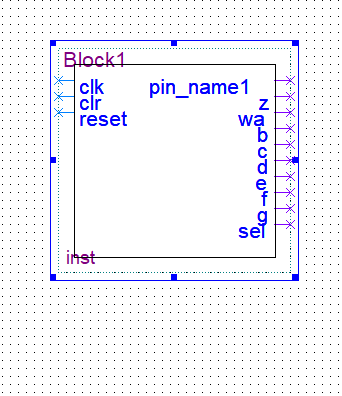
Endmodule

拓展实验：z=1时，计数加一。并显示在数码管上。



1. 仿真



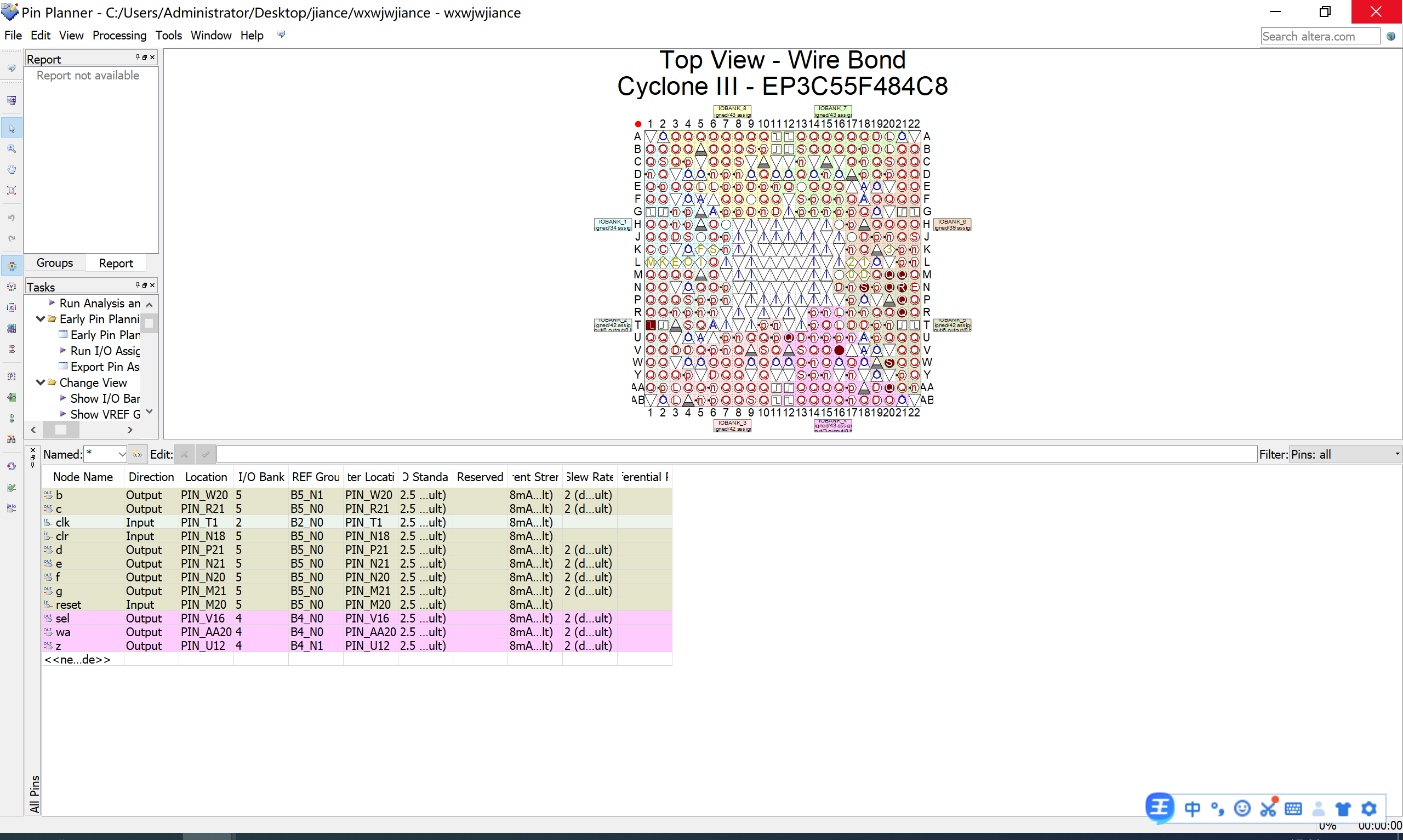
1. 封装图
2. 

**五．实验结果**

（1）引脚分配

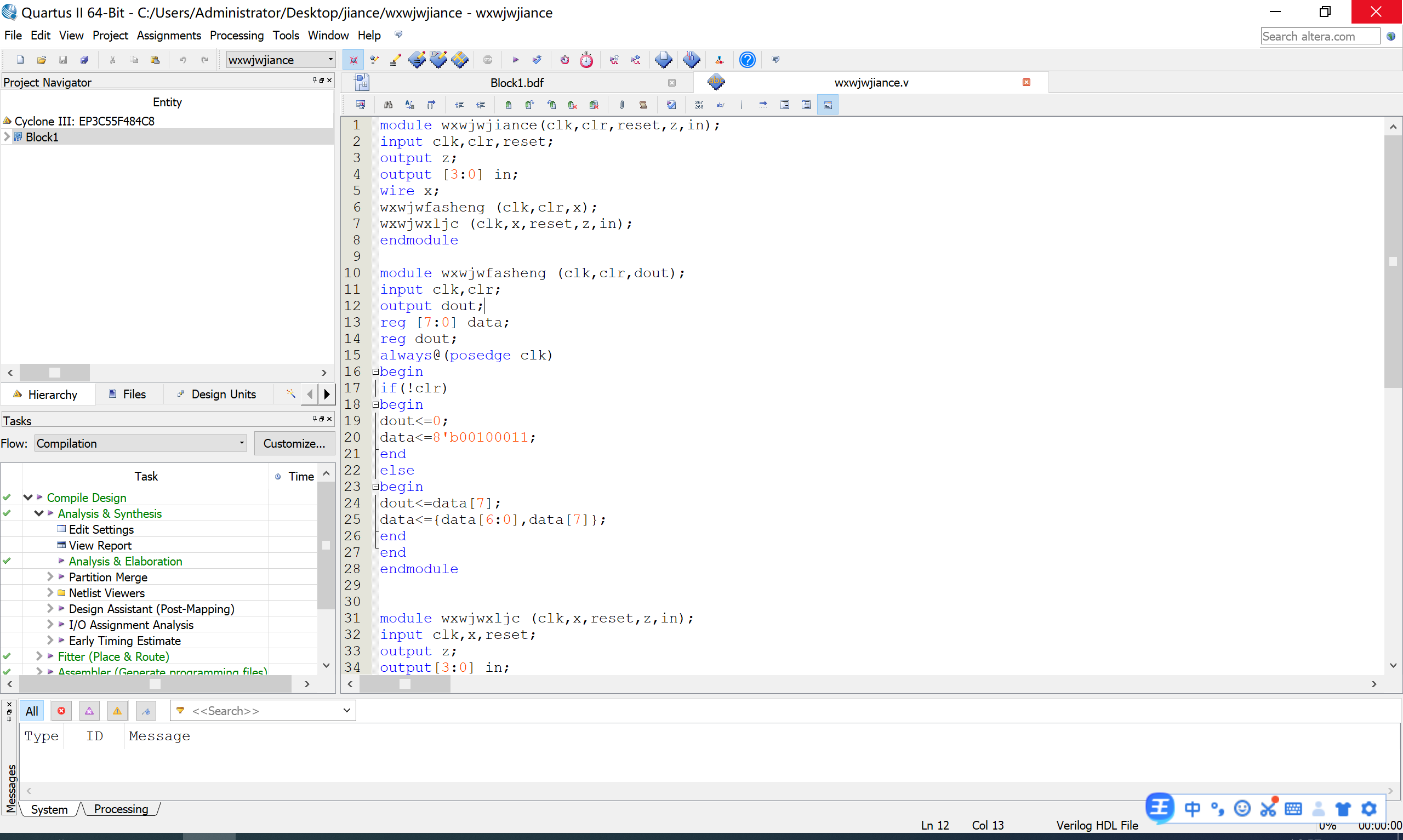
拓展实验：

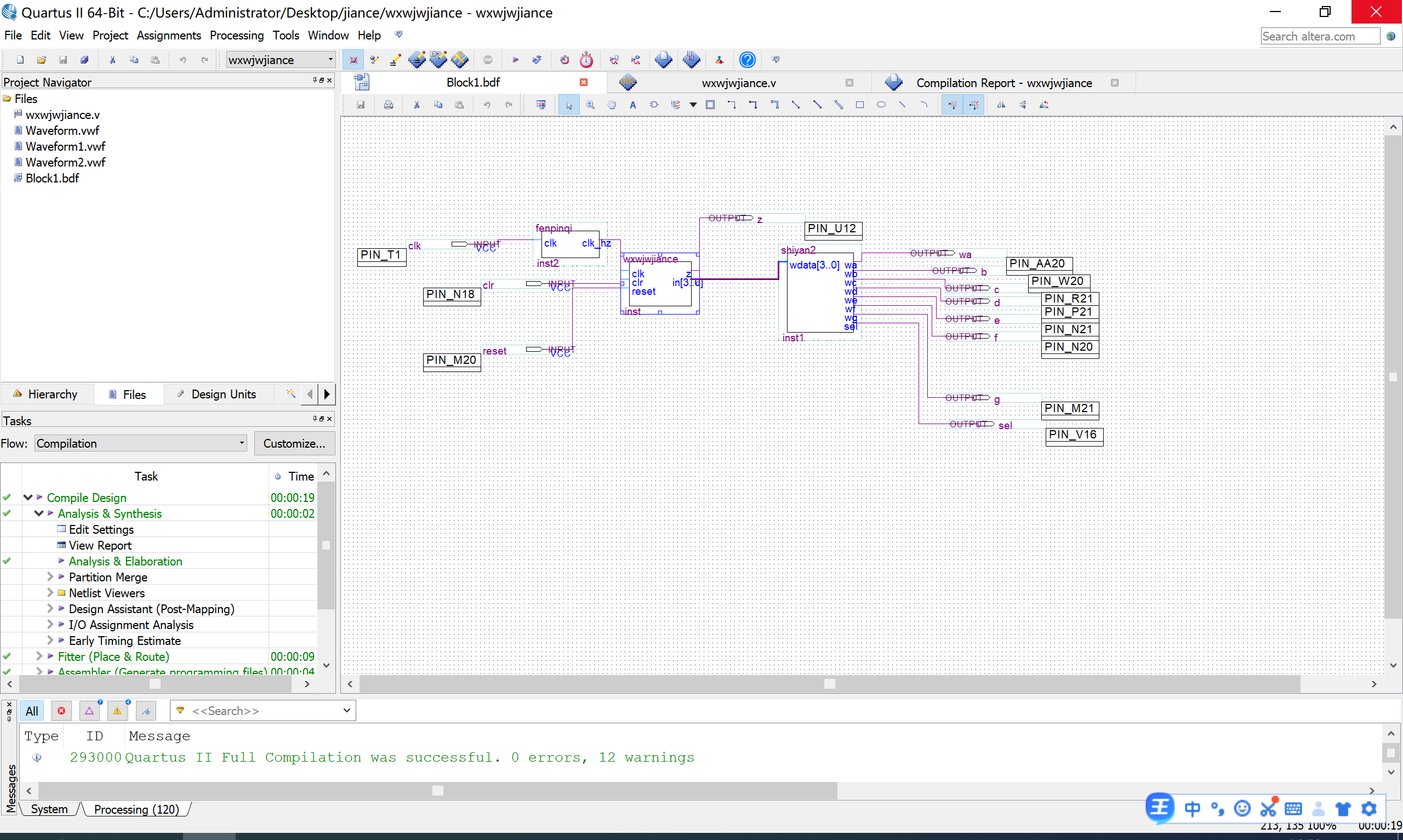
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **平台端口** | LA | LB | LC | LD |
| **引脚编号** | PIN\_AA20 | PIN\_W20 | PIN\_R21 | PIN\_P21 |
| **节点名称** | a | b | c | d |
| **平台端口** | LE | LF | LG | DS8 |
| **引脚编号** | PIN\_N21 | PIN\_N20 | PIN\_M21 | PIN\_V16 |
| **节点名称** | e | f | g | sel |
| **平台端口** | 时钟信号 | SW1 | SW2 | SW3 |
| **引脚编号** | PIN\_T1 | PIN\_N18 | PIN\_M20 | PIN\_U12 |
| **节点名称** | clk | clr | reset | z |



（2）实验现象

基础实验：clr为低有效，需先置0，序列开始发生，再置1，不影响程序运行，随着时钟信号扫描，出现相应序列时，z输出为1，led灭，数码管计数加一。





此图为拓展实验图

附图：顶层模块