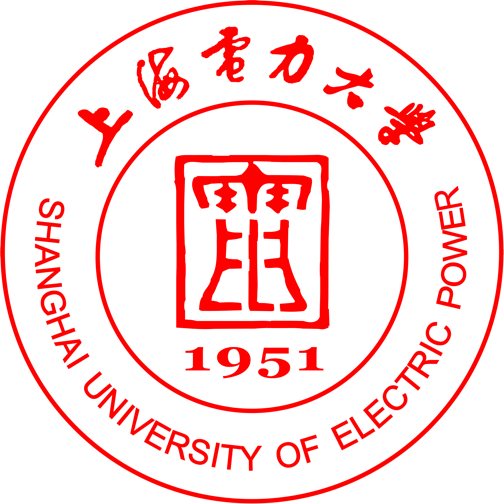
**上 海 电 力 大 学**

《FPGA应用开发》实验报告



**实验题目：**  计数器与时钟电路设计

**专 业：**

**班 级**  **学号**

**姓 名**

**时 间**  2023-03-17

1. 实验目的

(1) 掌握计数器硬件描述语言设计方法。

(2) 掌握计数器LPM实现方法。

(3) 掌握层次化设计方法实现时钟电路。

1. 实验任务及要求

(1) 完成实验内容中要求的各项任务。

(2) 记录模为100、24、6、10的计数器代码

(3) 记录时钟电路完整的电路图

(4) 记录每个实验内容的波形仿真结果、硬件验证结果，并总结调试过程中出现的问题和解决方案

1. 实验内容及步骤

1.计数器

实现一个8位计数器。计数器从“00000000”开始计到“11111111”，计数器的模是256。计 数器模块还需要包含一个时钟clock、一个使能信号en、一个异步清0信号aclr和一个同步数 据加载信号sload。

24

module counter\_24(q1,q0, clk,en,clr);

input clk,en,clr;

output[3:0] q0,q1;

reg[3:0] q0,q1;

always @ (posedge clk or negedge clr)

begin

if (!clr)

begin

q0<=0;q1<=0;

end

else if (en)

begin

if ((q1>2) || (q0>9) || ((q1==2) && (q0>=3)))

begin

q0<=0;

q1<=0;

end

else if (q0==9)

begin

q0<=0;

q1<=q1+1'b1;

end

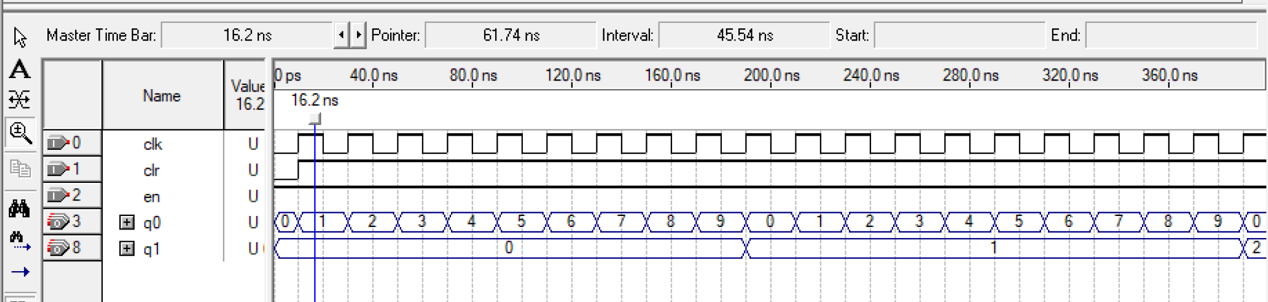
else q0<=q0+1'b1;

end

end

endmodule

模为24计数器功能仿真



10

module counter\_10(q,clk,en,clr,cout);

input clk,en,clr;

output[3:0] q;

output cout;

reg[3:0] q;

reg cout;

always @ (posedge clk or negedge clr)

begin

if (!clr)

begin

q<=0;

end

else if (en)

begin

if (q==9)

begin

q<=0;

cout = ~cout;

end

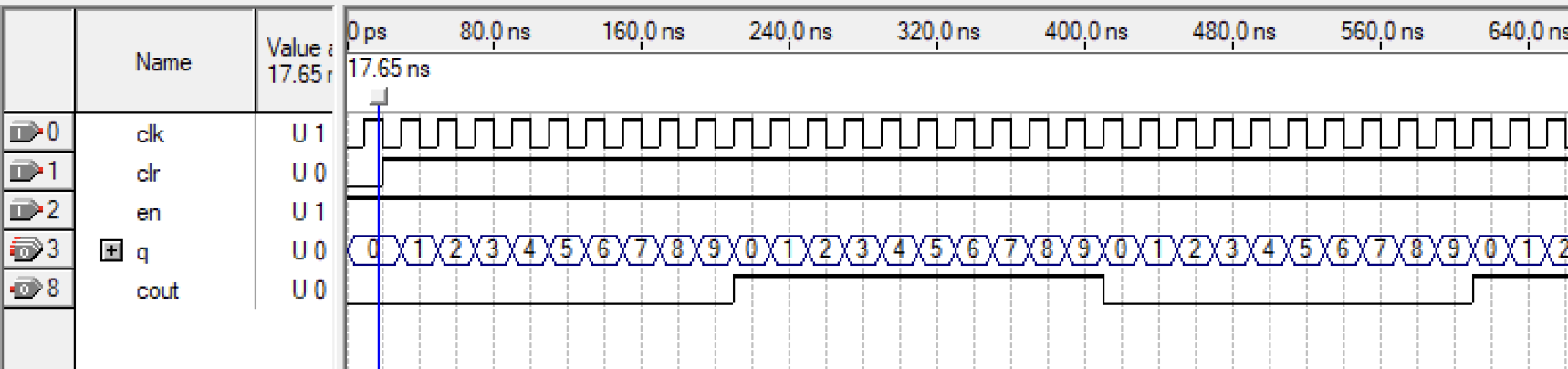
else q <= q+1'b1;

end

end

endmodule

模为10计数器功能仿真



6

module counter\_6(q,clk,en,clr,cout);

input clk,en,clr;

output[3:0] q;

output cout;

reg[3:0] q;

reg cout;

always @ (posedge clk or negedge clr)

begin

if (!clr)

begin

q<=0;

end

else if (en)

begin

if (q==5)

begin

q<=0;

cout = ~cout;

end

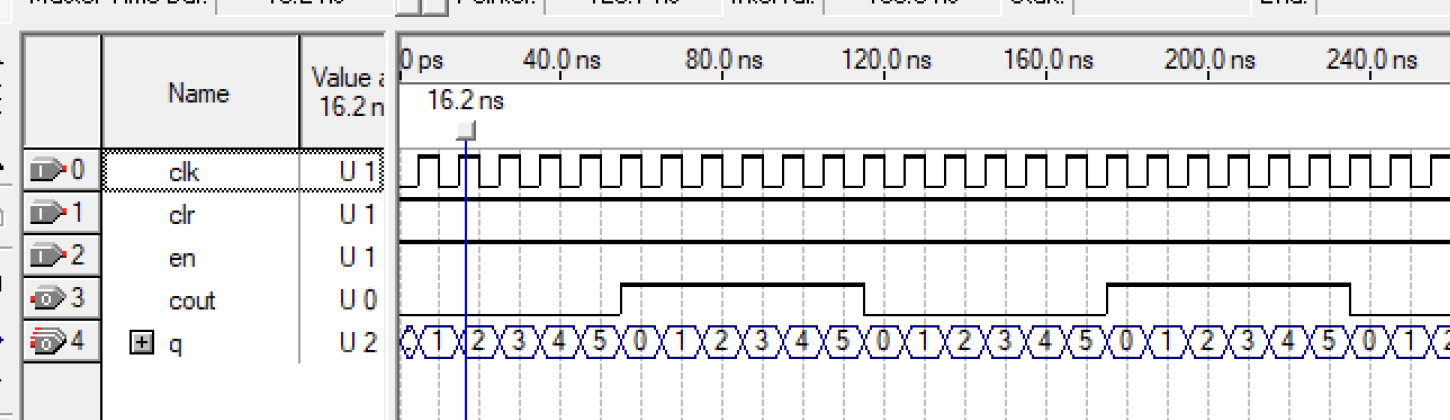
else q <= q+1'b1;

end

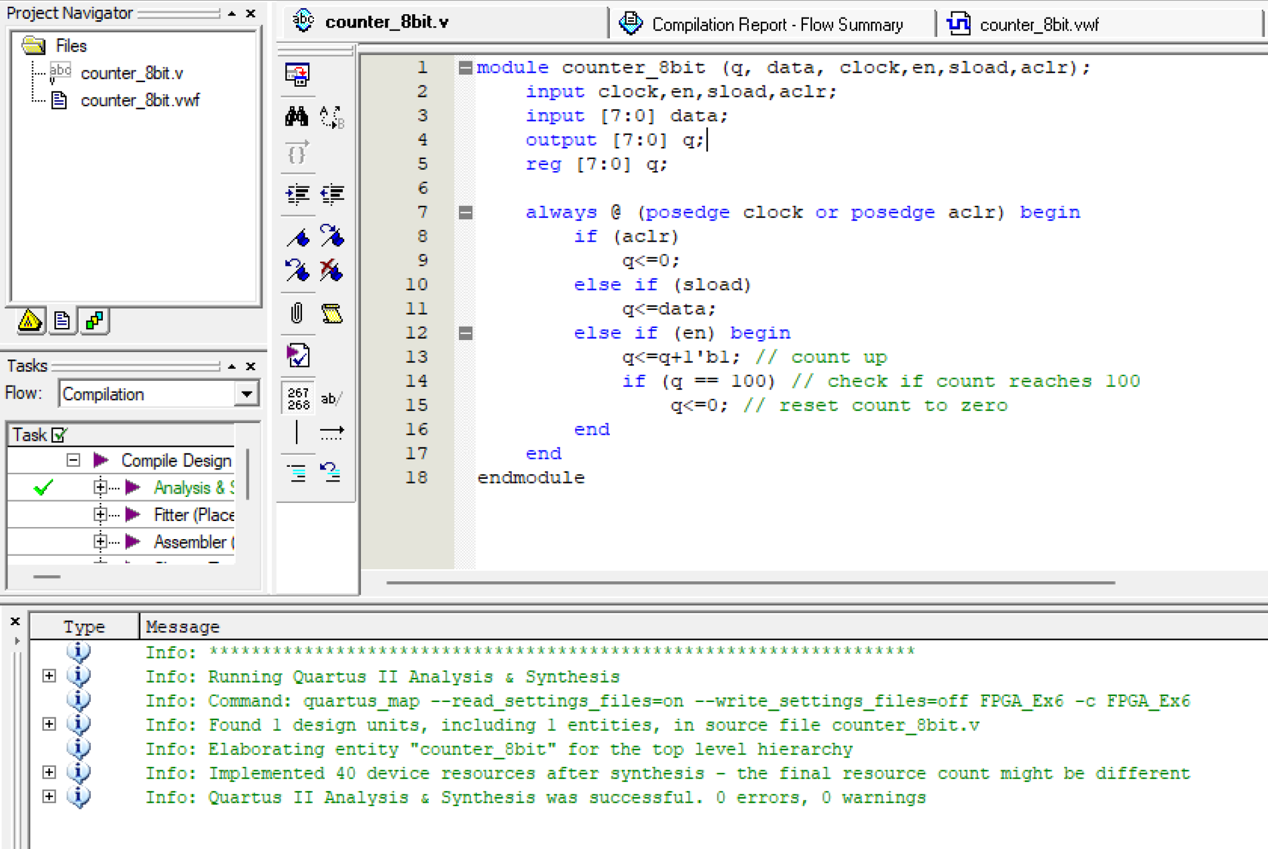
end

endmodule

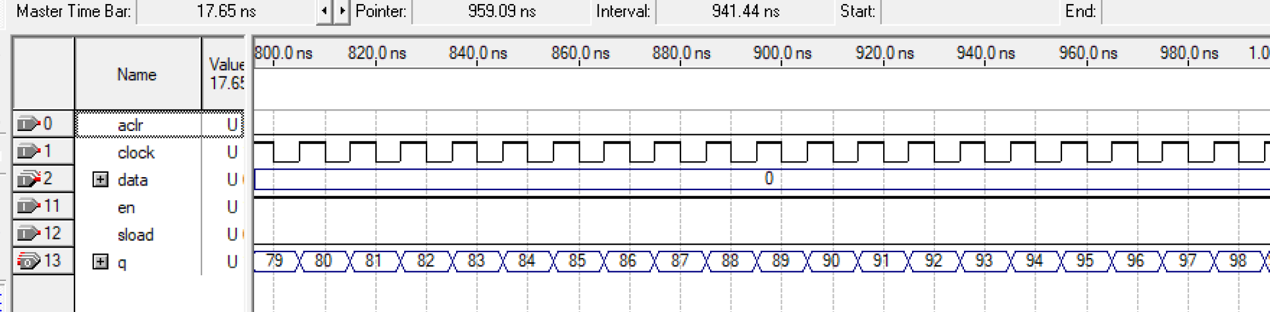
模为6计数器功能仿真



设计要求：修改代码，把计数器的模更改为100，并进行验证。



编译通过

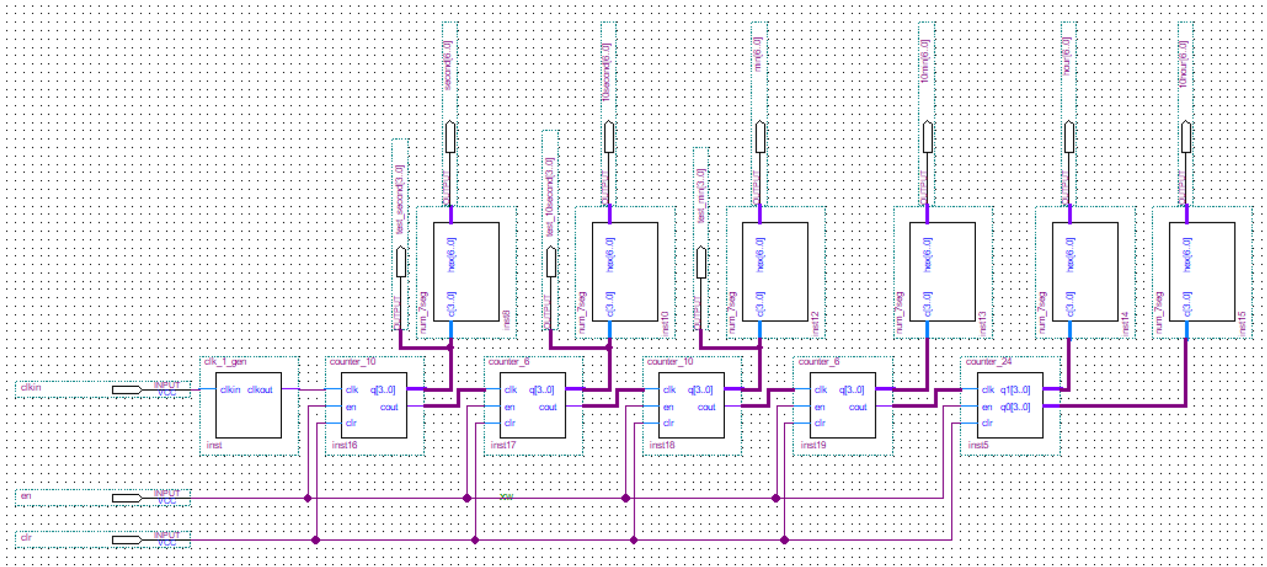


功能仿真

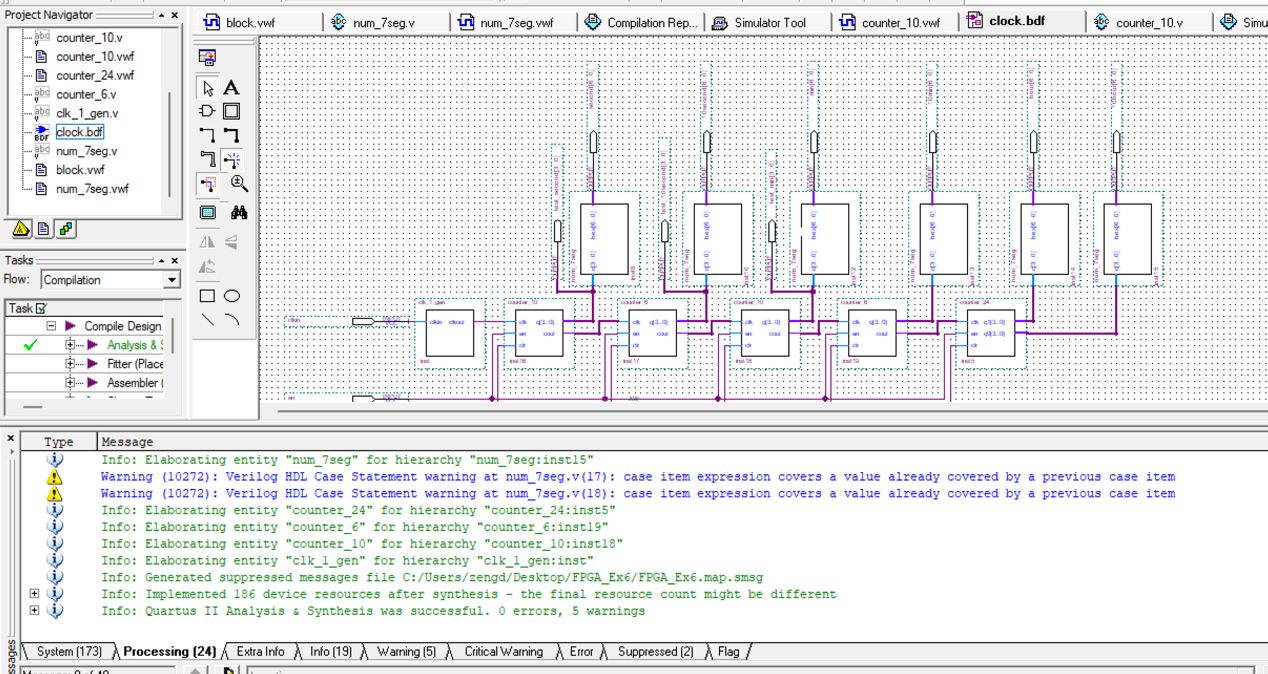
3.时钟电路

利用上面设计好的计数器和分频器设计一个实时的时钟。一共需要1个模24计数器、2 个模6计数器、2个模10计数器、一个生成1Hz的分频器和6个数码管解码器。 最终用 HEX5~HEX4显示小时（0~23），用HEX3~HEX2显示分钟（0~59），用HEX1~HEX0显示秒钟（0~59）。

Block原理图



Block.bdf 编译通过



1. 实验总结

> Error: Width mismatch in port "clk" of instance "inst5" and type counter\_24 -- source is ""cout[3..0]" (ID counter\_6:inst11)"

这个错误信息表明在实例化 `counter\_24` 类型的模块时，端口 `clk` 的宽度与源信号 cout[3..0] 的宽度不匹配。通常情况下，时钟信号 clk 应该是一个单比特信号，而不是一个多比特的总线。你可以检查一下你的代码，确保在实例化 counter\_24 模块时，为 clk 端口提供了正确的单比特信号。