

COD Lab0

❧ 代码

```
1 module Clock (  
2     input clk,  
3     input rstn, // asynchronous reset  
4     output [2:0] hour,  
5     output [3:0] min,  
6     output [4:0] sec  
7 ); // you should not change code upon this line  
8     // your code here  
9     // you may need to add some extra signals  
10    wire sec_c, min_c;  
11    // think the ports needed and how to connect them  
12    Sec sec1(clk, rstn, sec, sec_c);  
13    Min min1(clk, rstn, sec_c, min, min_c);  
14    Hour hour1(clk, rstn, min_c, hour);  
15 endmodule  
16  
17 // implement the three modules here  
18 module Sec (  
19     input clk,  
20     input rstn,  
21     output reg [4:0] sec_reg,  
22     output reg sec_c  
23 );  
24     always @(posedge clk or negedge rstn) begin  
25         if (!rstn)  
26             sec_reg <= 0;  
27         else if (sec_reg == 5'd19)  
28             sec_reg <= 0;  
29         else  
30             sec_reg <= sec_reg + 1;  
31     end  
32     always @(*) begin  
33         if (sec_reg == 5'd19)  
34             sec_c <= 1;  
35         else  
36             sec_c <= 0;  
37     end  
38 endmodule  
39
```

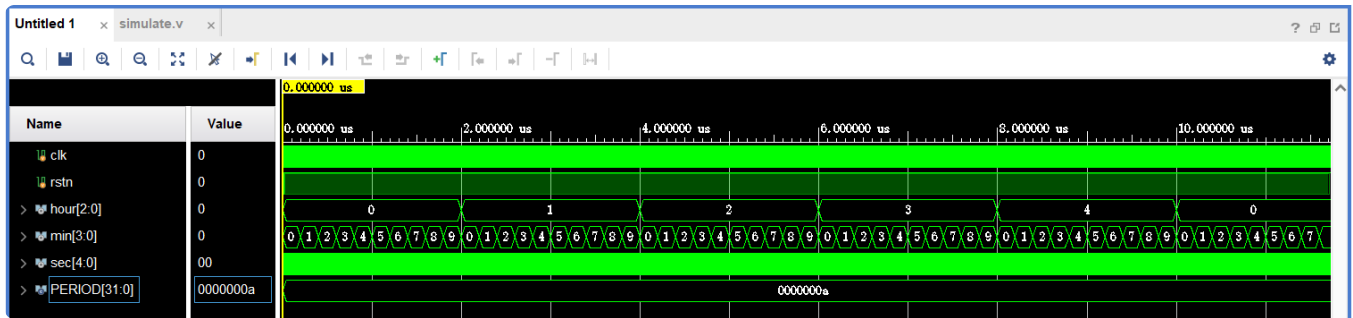
```

40 module Min (
41     input clk,
42     input rstn,
43     input sec_c,
44     output reg [3:0] min_reg,
45     output reg min_c
46 );
47     always @(posedge clk or negedge rstn) begin
48         if (!rstn)
49             min_reg <= 0;
50         else if (sec_c) begin
51             if (min_reg == 4'd9)
52                 min_reg <= 0;
53             else
54                 min_reg <= min_reg + 1;
55         end
56         else
57             min_reg <= min_reg;
58     end
59     always @(*) begin
60         if (sec_c && min_reg == 4'd9)
61             min_c <= 1;
62         else
63             min_c <= 0;
64     end
65 endmodule
66
67 module Hour (
68     input clk,
69     input rstn,
70     input min_c,
71     output reg [2:0] hour_reg
72 );
73     always @(posedge clk or negedge rstn) begin
74         if (!rstn)
75             hour_reg <= 0;
76         else if (min_c) begin
77             if (hour_reg == 3'd4)
78                 hour_reg <= 0;
79             else
80                 hour_reg <= hour_reg + 1;
81         end
82         else
83             hour_reg <= hour_reg;
84     end

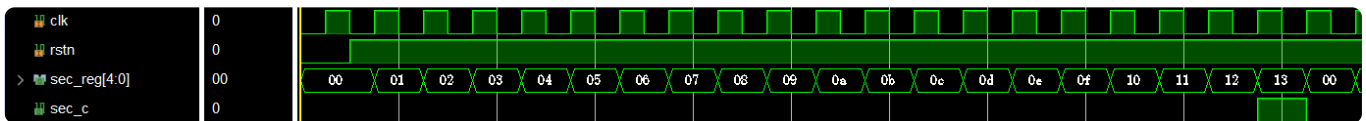
```

仿真波形

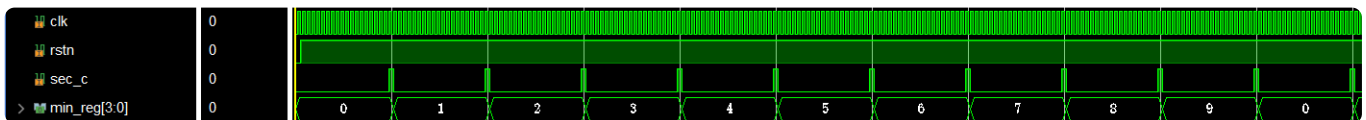
Clock



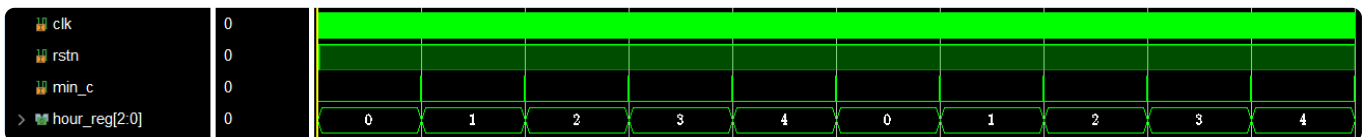
Sec



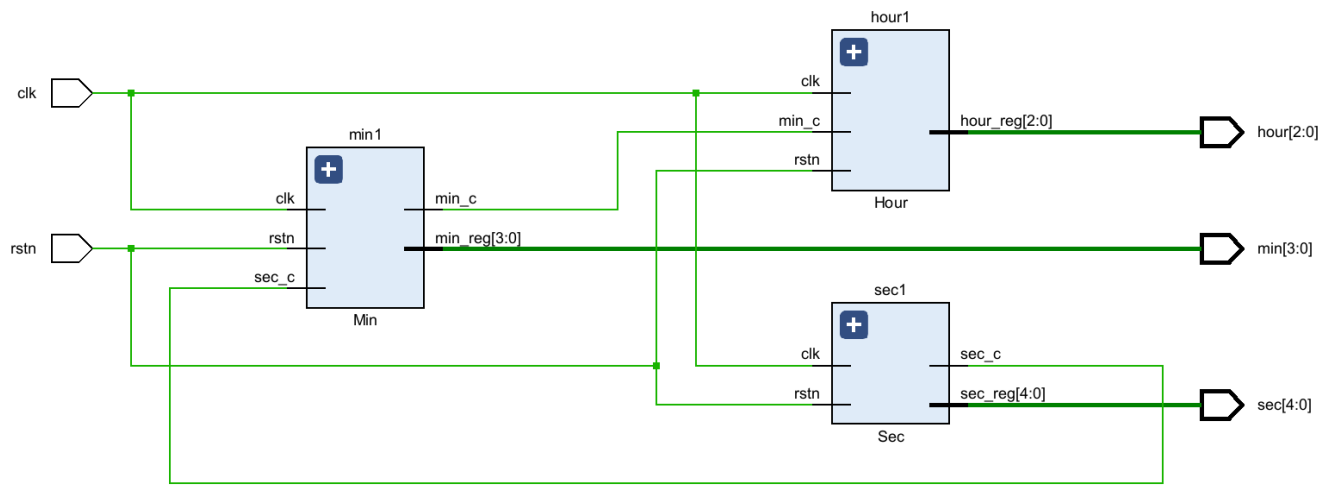
Min



Hour



电路图



实验反馈

文档写的很清晰，很详细，希望以后的实验文档能保持。