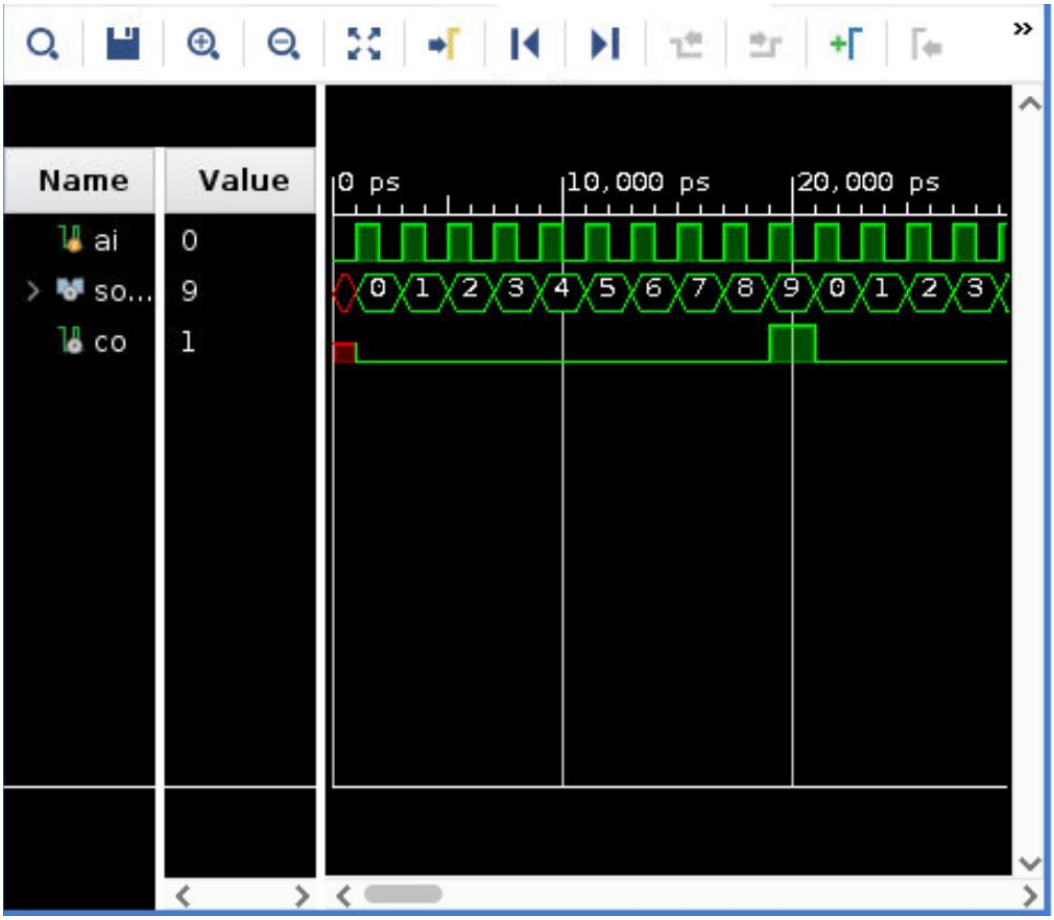
1. **修改模 10 计数器，使进位信号产生在最大那个数字的周期。**

模 10 计数器仿真波形：




源码截图：

```
1  `timescale 1ns / 1ps
2
3  module count10(ai, so, co);
4      input ai;
5      output[3:0] so;
6      reg [3:0] so;
7      output co;
8      reg co;
9
10     always@(posedge ai)
11     begin
12         if(so < 4'b1000)
13             begin
14                 so <= so+1'b1;
15                 co = 0;
16             end
17         else if(so == 8)
18             begin
19                 so <= so+1'b1;
20                 co = 1;
21             end
```

```
    begin
        if(so < 4'b1000)
            begin
                so <= so+1'b1;
                co = 0;
            end
        else if(so == 8)
            begin
                so <= so+1'b1;
                co = 1;
            end
        else
            begin
                so = 4'b0000;
                co = 0;
            end
    end

endmodule
```

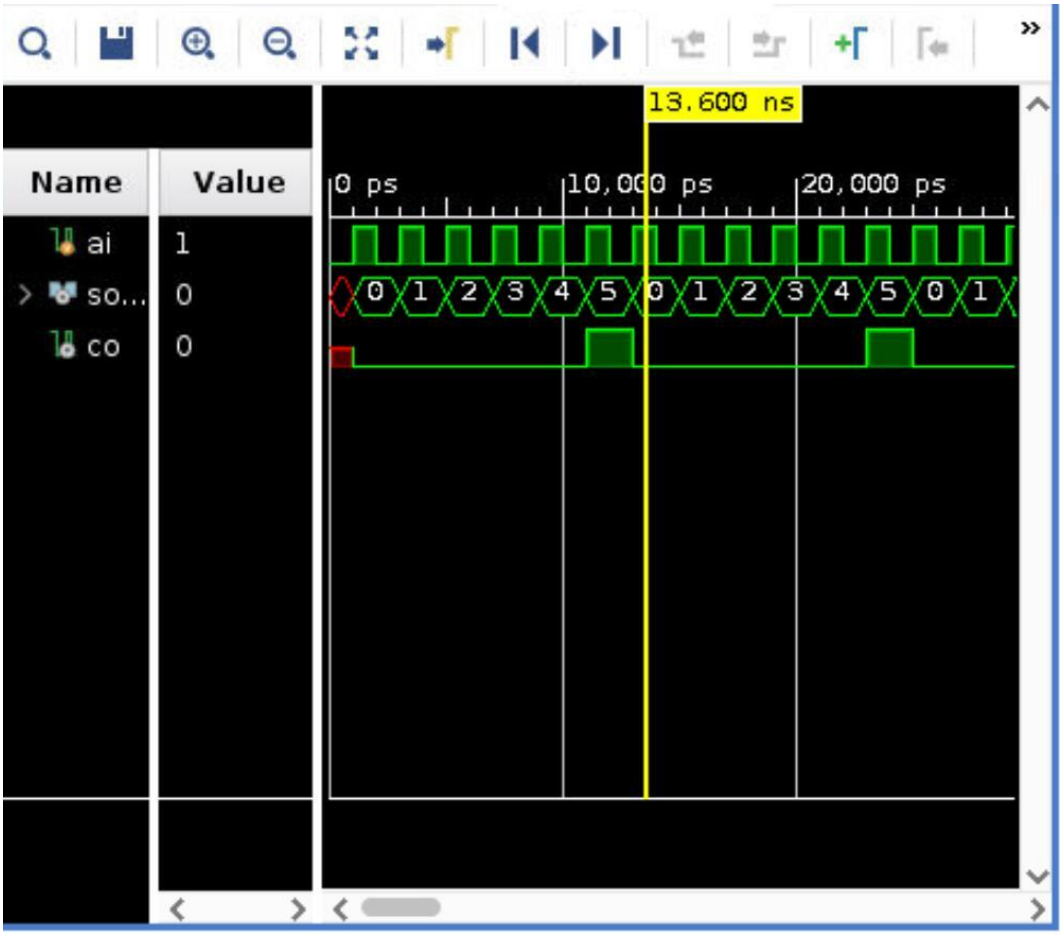
测试文件:

```
`timescale 1ns / 1ps

module test_10;
reg ai;
wire[3:0] so;
wire co;

always
begin
    #1 ai = ~ai;
end

initial
begin
    ai = 1'b0;

end

count10 u1(
    .ai(ai),
    .so(so)
```

```
 6  wire co;
 7
 8  always
 9  begin
10      #1 ai = ~ai;
11  end
12
13  initial
14  begin
15      ai = 1'b0;
16
17  end
18
19  count10 u1(
20      .ai(ai),
21      .so(so),
22      .co(co)
23      );
24
25  endmodule
```

2. **模 6 计数器。**

模 6 计数器仿真波形:



源码截图:

```
1  `timescale 1ns / 1ps
2
3  module count6(ai, so, co);
4      input ai;
5      output[3:0] so;
6      reg [3:0] so;
7      output co;
8      reg co;
9
10     always@(posedge ai)
11     begin
12         if(so < 4'b0100)
13             begin
14                 so <= so+1'b1;
15                 co = 0;
16             end
17         else if(so == 4)
18             begin
19                 so <= so+1'b1;
20                 co = 1;
21             end
```

```
11     begin
12         if(so < 4'b0100)
13             begin
14                 so <= so+1'b1;
15                 co = 0;
16             end
17         else if(so == 4)
18             begin
19                 so <= so+1'b1;
20                 co = 1;
21             end
22         else
23             begin
24                 so = 4'b0000;
25                 co = 0;
26             end
27     end
28
29 endmodule
30
```

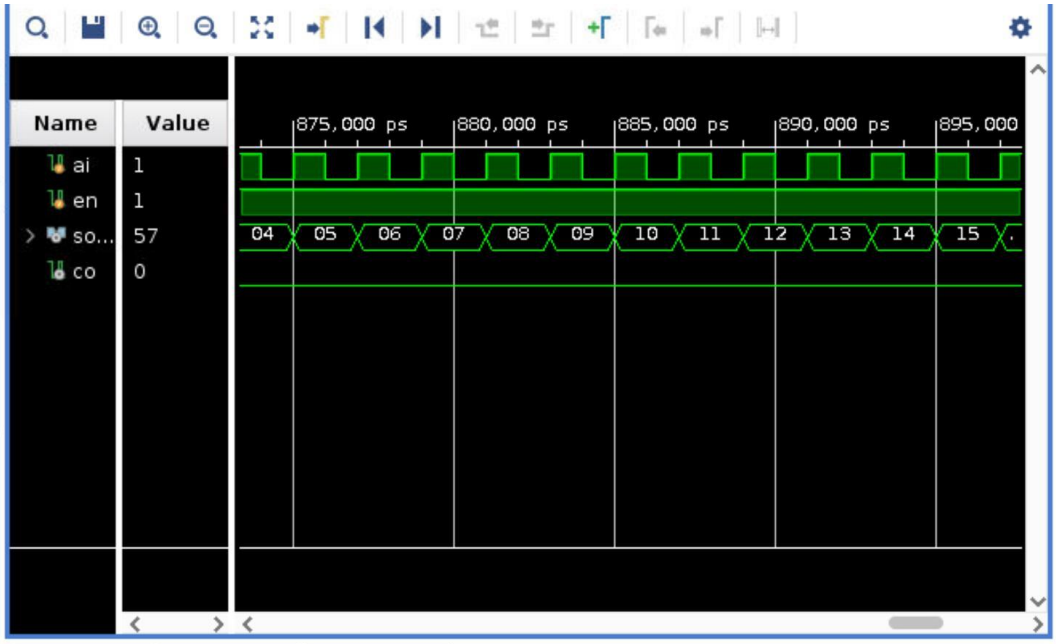
测试文件:

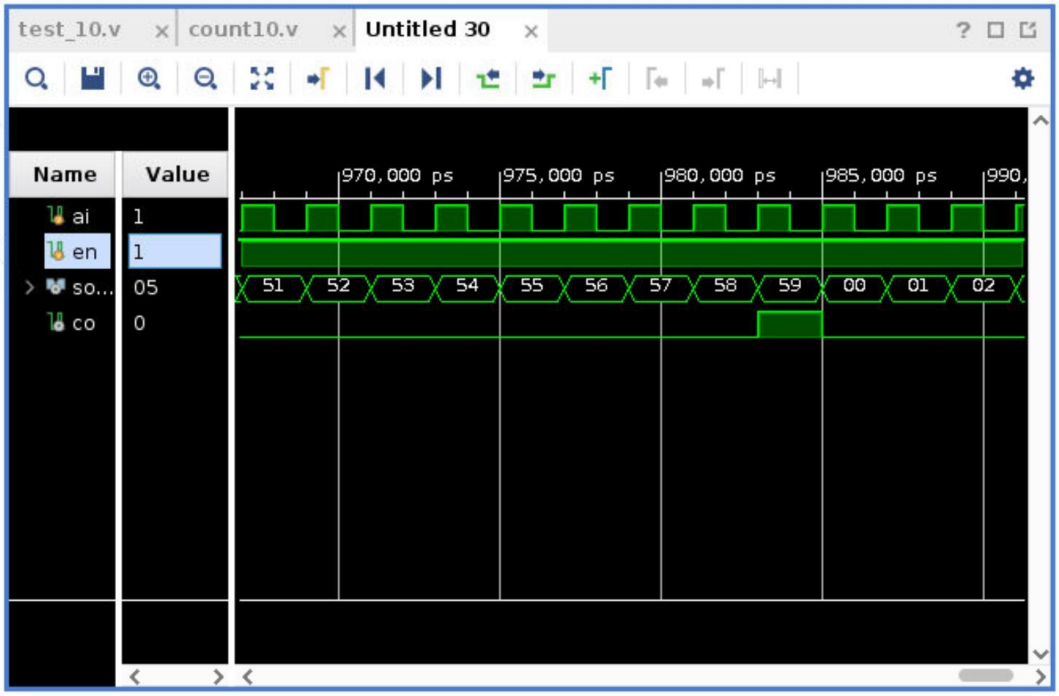
```
`timescale 1ns / 1ps

module test_6;
reg ai;
wire[3:0] so;
wire co;

always
begin
    #1 ai = ~ai;
end

initial
begin
    ai = 1'b0;

end

count6 u1(
    .ai(ai),
    .so(so)
```

```
wire co;

always
begin
    #1 ai = ~ai;
end

initial
begin
    ai = 1'b0;

end

count6 u1(
    .ai(ai),
    .so(so),
    .co(co)
    );

endmodule
```

**3. 级联模 10 计数器和模 6 计数器，生成模 60 计数器。**

模 60 计数器仿真波形：



源码截图：

/mnt/cgshare/project_1/project_1.srcs/sources_1/new/count10.v

```
1  `timescale 1ns / 1ps
2  module count60(ai,so,co,en);
3      input ai,en;
4      output[7:0] so;
5      output co;
6      wire co10,co6,en_6;
7      wire[3:0] so10,so6;
8
9      count10 u1(
10     .ai(ai),
11     .so(so10),
12     .co(co10),
13     .en(en)
14     );
15     and u2(en_6,en,co10);
16     count6 u3(
17     .ai(ai),
18     .so(so6),
19     .co(co6),
20     .en(en_6)
21     );
```

```
20     .en(en_6)
21     );
22     and u4(co,en_6,co6);  //simultaneous_carry
23
24     assign so = {so6,so10};  //connect
25 endmodule
26
27 module count6(ai, so, co, en);
28     input ai,en;
29     output[3:0] so;
30     reg [3:0] so;
31     output co;
32     reg co;
33
34     always@(posedge ai)
35     begin
36         if(en)
37             begin
38                 if(so < 4'b0100)
39                     begin
40                         so <= so+1'b1;
```

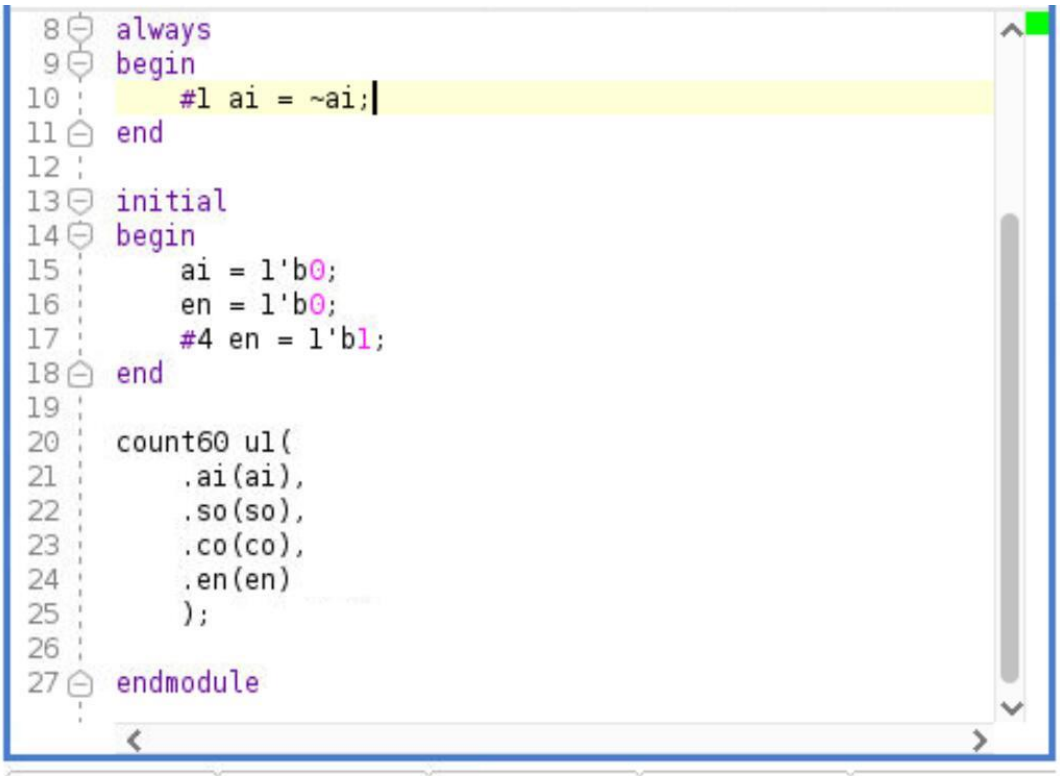
```
                            so <= so+1'b1;
                            co = 0;
                        end
                    else if(so == 4)
                        begin
                            so <= so+1'b1;
                            co = 1;
                        end
                    else
                        begin
                            so = 4'b0000;
                            co = 0;
                        end
                end
            else
                so = so;
        end

endmodule

module count10(ai, so, co,en);
```

```
module count10(ai, so, co,en);
    input ai,en;
    output[3:0] so;
    reg [3:0] so;
    output co;
    reg co;

    always@(posedge ai)
    begin
        if(en)
            begin
                if(so < 4'b1000)
                    begin
                        so <= so+1'b1;
                        co = 0;
                    end
                else if(so == 8)
                    begin
                        so <= so+1'b1;
```

```
                so <= so+1'b1;
                co = 0;
            end
        else if(so == 8)
            begin
                so <= so+1'b1;
                co = 1;
            end
        else
            begin
                so = 4'b0000;
                co = 0;
            end
    end
else
    so = so;
end

endmodule
```

测试文件:

```
`timescale 1ns / 1ps

module test_60;
reg ai,en;
wire[7:0] so;  //8bit
wire co;

always
begin
    #1 ai = ~ai;
end

initial
begin
    ai = 1'b0;
    en = 1'b0;
    #4 en = 1'b1;
end

count60 u1(
    .ai(ai)
```



```
always
begin
    #1 ai = ~ai;
end

initial
begin
    ai = 1'b0;
    en = 1'b0;
    #4 en = 1'b1;
end

count60 u1(
    .ai(ai),
    .so(so),
    .co(co),
    .en(en)
    );

endmodule
```