

# 数字电路实验报告：实验 08 信号处理及有限状态机

姓名：曾舒立；学号：PB19000200；日期：2021/11/28。

## 实验目的

进一步熟悉 FPGA 开发的整体流程

掌握几种常见的信号处理技巧

掌握有限状态机的设计方法

能够使用有限状态机设计功能电路

## 实验环境

VLAB 平台：vlab.ustc.edu.cn

FPGAOL 平台：fpgaol.ustc.edu.cn

Vivado

Logisim

## 实验过程

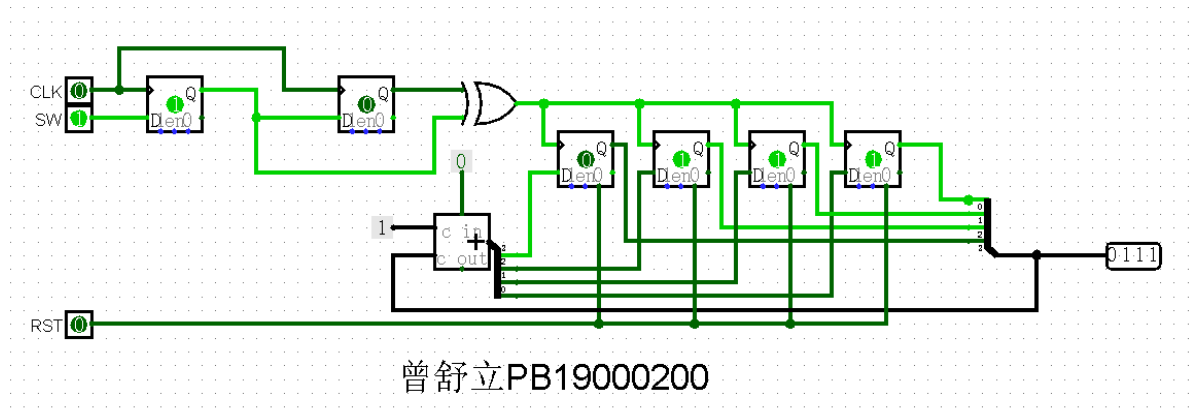
### 题目1

改为有限状态机的代码为：

```
1  module test(input clk,rst,output led);
2  reg [1:0] cnt_curr;
3  reg [1:0] cnt_next;
4
5  always@(*)
6  begin
7      cnt_next = cnt_curr + 1;
8  end
9
10 always@(posedge clk, posedge rst)
11 begin
12     if(rst)
13         cnt_curr <= 2'b0;
14     else
15         cnt_curr <= cnt_next;
16 end
17
18 assign led = (cnt_curr==2'b11) ? 1'b1 : 1'b0;
19
20 endmodule
```

## 题目2

计数器电路如图：



## 实验3

设计文件代码：

```
1 module d_out(input clk,[31:0] data,output reg [3:0] d,reg [2:0] an);
2 wire c50hz0,c50hz1,c50hz2,c50hz3,c50hz4,c50hz5,c50hz6,c50hz7;
3 reg [20:0] count;
4
5 always @(posedge clk)
6 begin
7     count <= count + 21'b1;
8 end
9 assign c50hz0 = (count == 21'b0_0011111111_1111111111);
10 assign c50hz1 = (count == 21'b0_0111111111_1111111111);
11 assign c50hz2 = (count == 21'b0_1011111111_1111111111);
12 assign c50hz3 = (count == 21'b0_1111111111_1111111111);
13 assign c50hz4 = (count == 21'b1_0011111111_1111111111);
14 assign c50hz5 = (count == 21'b1_0111111111_1111111111);
15 assign c50hz6 = (count == 21'b1_1011111111_1111111111);
16 assign c50hz7 = (count == 21'b1_1111111111_1111111111);
17
18 always @(posedge clk)
19 begin
20     if (c50hz0)
21     begin
22         d<=data[3:0];
23         an<=3'b000;
24     end
25     if (c50hz1)
26     begin
27         d<=data[7:4];
28         an<=3'b001;
29     end
30     if (c50hz2)
31     begin
32         d<=data[11:8];
33         an<=3'b010;
34     end
35     if (c50hz3)
36     begin
37         d<=data[15:12];
```

```

38         an<=3'b011;
39     end
40     if (c50hz4)
41     begin
42         d<=data[19:16];
43         an<=3'b100;
44     end
45     if (c50hz5)
46     begin
47         d<=data[23:20];
48         an<=3'b101;
49     end
50     if (c50hz6)
51     begin
52         d<=data[27:24];
53         an<=3'b110;
54     end
55     if (c50hz7)
56     begin
57         d<=data[31:28];
58         an<=3'b111;
59     end
60 end
61
62 endmodule
63
64
65
66 module week8_3(input clk,sw,button,rst_sw,output [3:0] d, [2:0] an);
67 reg [31:0] curr_state,next_state_p,next_state_n;
68
69 always@(*)
70 begin
71     next_state_p = curr_state + 1;
72     next_state_n = curr_state - 1;
73 end
74
75 always@(posedge button)
76 begin
77     if (rst_sw == 1) curr_state<=32'h1F;
78     else
79     begin
80         if (sw == 1) curr_state<=next_state_p;
81         else curr_state<=next_state_n;
82     end
83 end
84
85 d_out d_out(clk,curr_state,d,an);
86
87 endmodule

```

管脚约束文件:

```

1 # This file is a general .xdc for FPGAOL_BOARD (adopted from Nexys4 DDR Rev.
  C)
2 # To use it in a project:
3 # - uncomment the lines corresponding to used pins

```

```

4  # - rename the used ports (in each line, after get_ports) according to the
   top level signal names in the project
5
6  # Clock signal
7  set_property -dict { PACKAGE_PIN E3      IOSTANDARD LVCMOS33 } [get_ports {
   clk }];
8
9  # FPGA BUTTON
10 set_property -dict { PACKAGE_PIN B18      IOSTANDARD LVCMOS33 } [get_ports {
   button }];
11
12 # FPGAOLED (single-digit-SEGPLAY)
13
14 set_property -dict { PACKAGE_PIN A14      IOSTANDARD LVCMOS33 } [get_ports {
   d[0] }];
15 set_property -dict { PACKAGE_PIN A13      IOSTANDARD LVCMOS33 } [get_ports {
   d[1] }];
16 set_property -dict { PACKAGE_PIN A16      IOSTANDARD LVCMOS33 } [get_ports {
   d[2] }];
17 set_property -dict { PACKAGE_PIN A15      IOSTANDARD LVCMOS33 } [get_ports {
   d[3] }];
18 set_property -dict { PACKAGE_PIN B17      IOSTANDARD LVCMOS33 } [get_ports {
   an[0] }];
19 set_property -dict { PACKAGE_PIN B16      IOSTANDARD LVCMOS33 } [get_ports {
   an[1] }];
20 set_property -dict { PACKAGE_PIN A18      IOSTANDARD LVCMOS33 } [get_ports {
   an[2] }];
21
22
23 # FPGAOLED SWITCH
24
25 set_property -dict { PACKAGE_PIN D14      IOSTANDARD LVCMOS33 } [get_ports { sw
   }];
26 set_property -dict { PACKAGE_PIN F16      IOSTANDARD LVCMOS33 } [get_ports {
   rst_sw }];
27
28 set_property CLOCK_DEDICATED_ROUTE FALSE [get_nets button_IBUF];

```

## 实验4

设计文件代码：

```

1  module d_out4(input clk,[3:0] seq,cnt,state,output reg [3:0] d,reg [2:0]
   an);
2  wire c50hz0,c50hz1,c50hz2,c50hz3,c50hz5,c50hz7;
3  reg [20:0] count;
4
5  always @(posedge clk)
6  begin
7      count <= count + 21'b1;
8  end
9
10 assign c50hz0 = (count == 21'b0_0011111111_1111111111);
11 assign c50hz1 = (count == 21'b0_0111111111_1111111111);
12 assign c50hz2 = (count == 21'b0_1011111111_1111111111);
13 assign c50hz3 = (count == 21'b0_1111111111_1111111111);
14 //assign c50hz4 = (count == 21'b1_0011111111_1111111111);

```

```

15 assign c50hz5 = (count == 21'b1_011111111_111111111);
16 //assign c50hz6 = (count == 21'b1_101111111_111111111);
17 assign c50hz7 = (count == 21'b1_111111111_111111111);
18
19 always @(posedge clk)
20 begin
21     if (c50hz0)
22     begin
23         d<={3'b000,seq[0]};
24         an<=3'b000;
25     end
26     if (c50hz1)
27     begin
28         d<={3'b000,seq[1]};
29         an<=3'b001;
30     end
31     if (c50hz2)
32     begin
33         d<={3'b000,seq[2]};
34         an<=3'b010;
35     end
36     if (c50hz3)
37     begin
38         d<={3'b000,seq[3]};
39         an<=3'b011;
40     end
41
42
43     if (c50hz5)
44     begin
45         d<=cnt;
46         an<=3'b101;
47     end
48
49
50     if (c50hz7)
51     begin
52         d<=state;
53         an<=3'b111;
54     end
55 end
56
57 endmodule
58
59
60
61 module week8_4(input clk,sw,button,output [3:0] d, [2:0] an);
62 reg [3:0] curr_seq,next_seq,curr_cnt,next_cnt,curr_state,next_state;
63
64 always@(*)
65 begin
66     next_seq[3:1]=curr_seq[2:0];
67     next_seq[0]=sw;
68     next_cnt=curr_cnt+1'b1;
69     if (curr_state==4'h0)
70     begin
71         if (sw==0) next_state=4'h0;
72         else next_state=4'h1;

```

```

73     end
74     else if (curr_state==4'h1)
75     begin
76         if (sw==0) next_state=4'h0;
77         else      next_state=4'h2;
78     end
79     else if (curr_state==4'h2)
80     begin
81         if (sw==0) next_state=4'h3;
82         else      next_state=4'h1;
83     end
84     else if (curr_state==4'h3)
85     begin
86         if (sw==0) next_state=4'h0;
87         else      next_state=4'h1;
88     end
89     else next_state=4'h0;
90 end
91
92 always@(posedge button)
93 begin
94     curr_seq<=next_seq;
95     curr_state<=next_state;
96 end
97
98 always@(negedge button)
99 begin
100     if (curr_seq==4'b1100) curr_cnt<=next_cnt;
101 end
102
103 d_out4 d_out4(clk,curr_seq,curr_cnt,curr_state,d,an);
104
105 endmodule

```

管脚约束文件:

```

1  # This file is a general .xdc for FPGAOL_BOARD (adopted from Nexys4 DDR Rev.
   C)
2  # To use it in a project:
3  # - uncomment the lines corresponding to used pins
4  # - rename the used ports (in each line, after get_ports) according to the
   top level signal names in the project
5
6  # Clock signal
7  set_property -dict { PACKAGE_PIN E3      IOSTANDARD LVCMOS33 } [get_ports {
   clk }];
8
9  # FPGA BUTTON
10 set_property -dict { PACKAGE_PIN B18     IOSTANDARD LVCMOS33 } [get_ports {
   button }];
11
12 # FPGAOL LED (single-digit-SEGPLAY)
13
14 set_property -dict { PACKAGE_PIN A14     IOSTANDARD LVCMOS33 } [get_ports {
   d[0] }];
15 set_property -dict { PACKAGE_PIN A13     IOSTANDARD LVCMOS33 } [get_ports {
   d[1] }];

```

```

16 set_property -dict { PACKAGE_PIN A16   IOSTANDARD LVCMOS33 } [get_ports {
   d[2] }];
17 set_property -dict { PACKAGE_PIN A15   IOSTANDARD LVCMOS33 } [get_ports {
   d[3] }];
18 set_property -dict { PACKAGE_PIN B17   IOSTANDARD LVCMOS33 } [get_ports {
   an[0] }];
19 set_property -dict { PACKAGE_PIN B16   IOSTANDARD LVCMOS33 } [get_ports {
   an[1] }];
20 set_property -dict { PACKAGE_PIN A18   IOSTANDARD LVCMOS33 } [get_ports {
   an[2] }];
21
22
23 # FPGAOL SWITCH
24
25 set_property -dict { PACKAGE_PIN D14   IOSTANDARD LVCMOS33 } [get_ports { sw
   }]];
26
27 set_property CLOCK_DEDICATED_ROUTE FALSE [get_nets button_IBUF];

```

(即是题目3中管脚约束文件删去rst\_sw的一行)

按照“0011001110011”输入后的显示如下

FPGA interface

uart

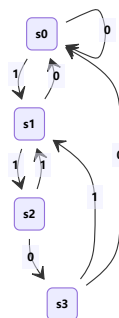
```
FPGAOL uart beta 1.0
>
```

uart pins: cts rts rxd txd  
xdc,ucf sym: D3 E5 D4 C4  
baud rate: 115200

segplay(sharing with led)    hexplay

soft clock    button

None ▾



## 总结与思考

1. 请总结本次实验的收获

学会了用有限状态机的方式编写verilog, 以及信号去抖动和去边沿的方法。

2. 请评价本次实验的难易程度

略困难。

3. 请评价本次实验的任务量

偏大。

4. 请为本次实验提供改进建议

暂无。