数字电路实验报告

实验题目: 信号处理及有限状态机

学生姓名: 孔浩宇

学生学号: PB20000113

完成日期: 2022/12/1

1 实验题目

信号处理及有限状态机

2 实验目的

1. 进一步熟悉 FPGA
2. 掌握几种常见的信号处理技巧
3. 掌握有限状态机的设计方法
4. 能够使用有限状态机设计功能电路

3 实验环境

(1) vlab.ustc.edu.cn

(2) fpagol.ustc.edu.cn

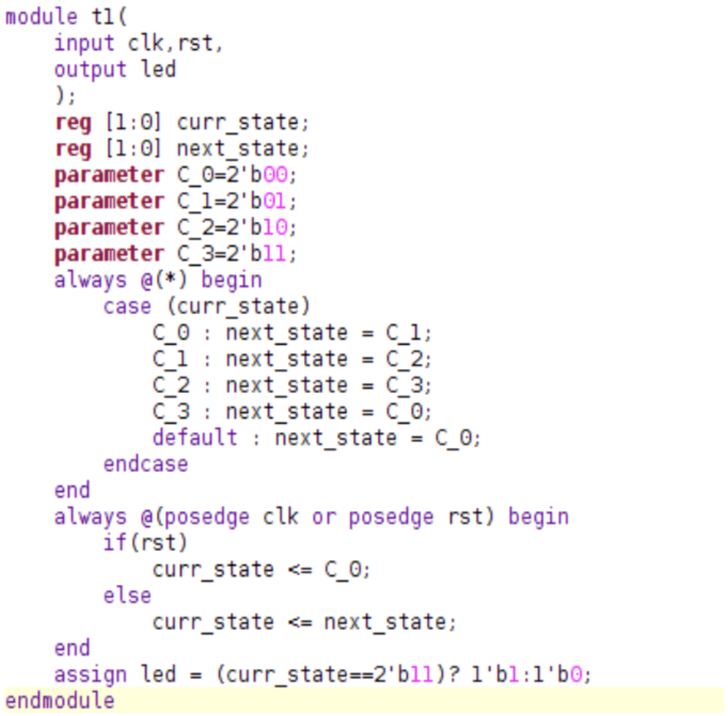
(3) Logisim

(4) Vivado

4 实验练习

题目 1

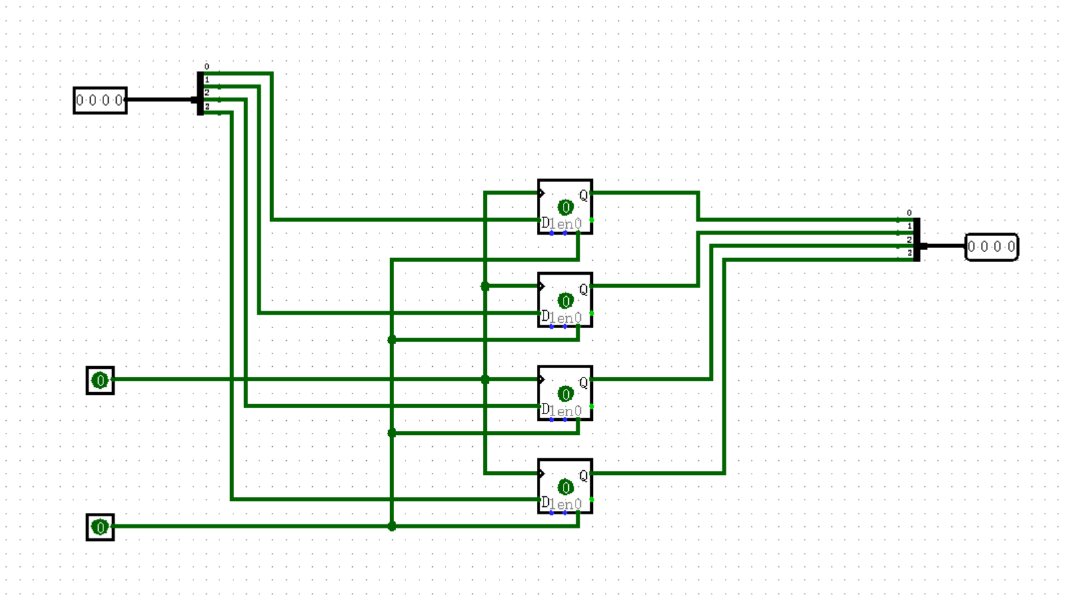
代码如图



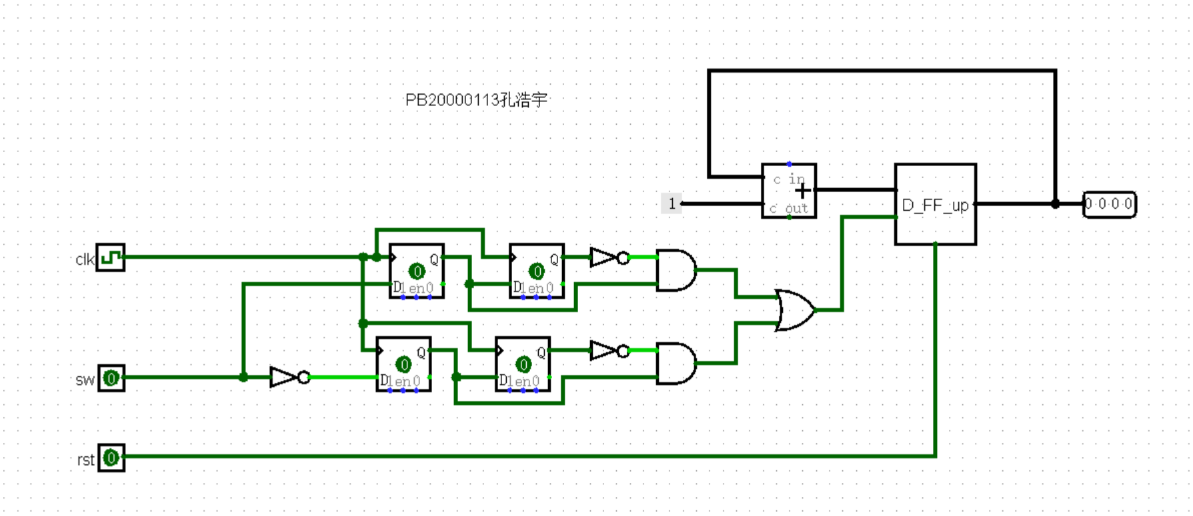
```
module t1(
    input clk,rst,
    output led
    );
    reg [1:0] curr_state;
    reg [1:0] next_state;
    parameter C_0=2'b00;
    parameter C_1=2'b01;
    parameter C_2=2'b10;
    parameter C_3=2'b11;
    always @(*) begin
        case (curr_state)
            C_0 : next_state = C_1;
            C_1 : next_state = C_2;
            C_2 : next_state = C_3;
            C_3 : next_state = C_0;
            default : next_state = C_0;
        endcase
    end
    always @(posedge clk or posedge rst) begin
        if(rst)
            curr_state <= C_0;
        else
            curr_state <= next_state;
    end
    assign led = (curr_state==2'b11)? 1'b1:1'b0;
endmodule
```

题目 2

构造四位 D 触发器如图

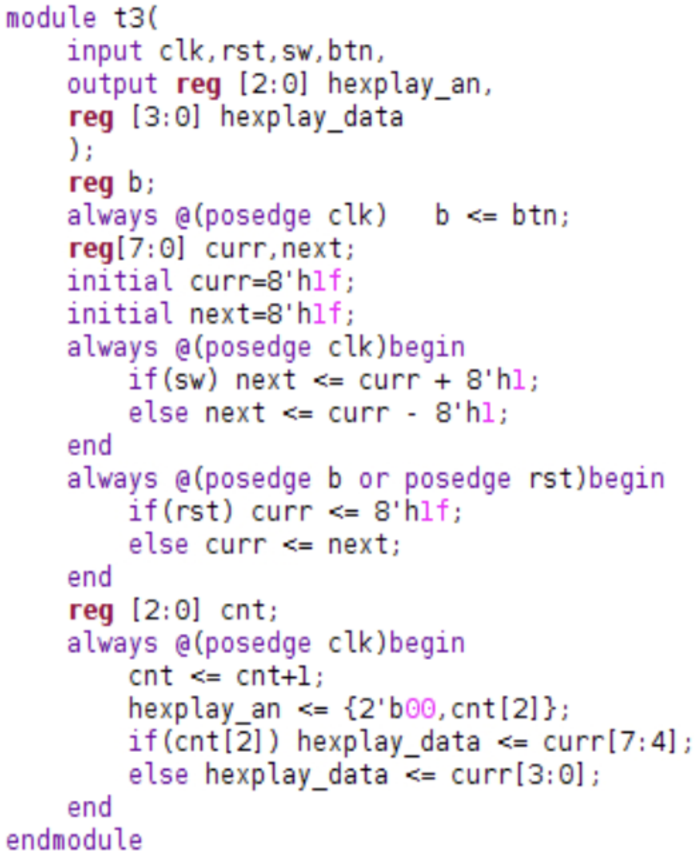


构造计数器如图



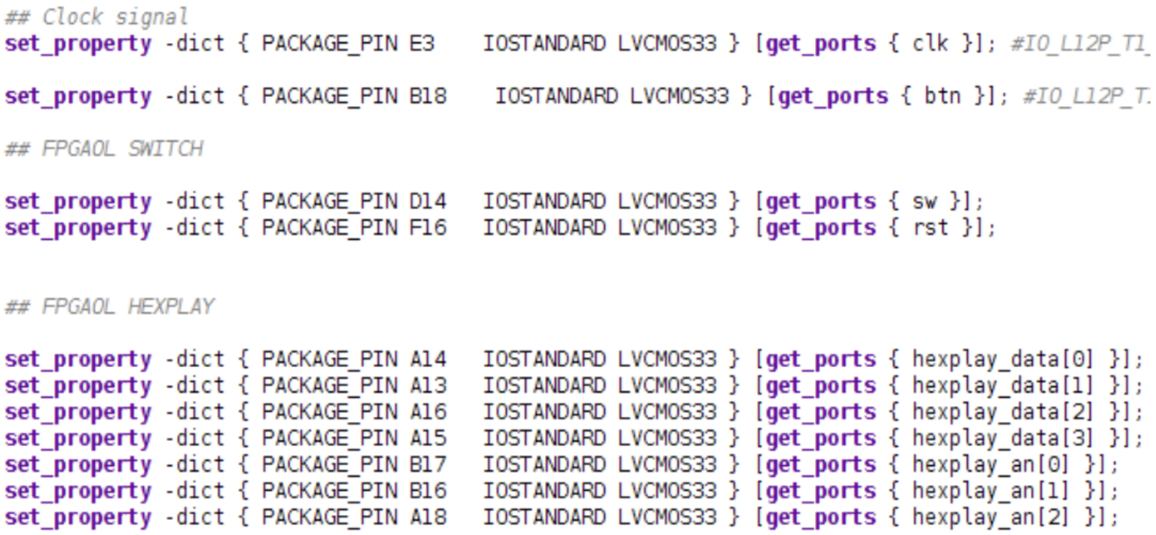
题目 3

设计文件代码如图



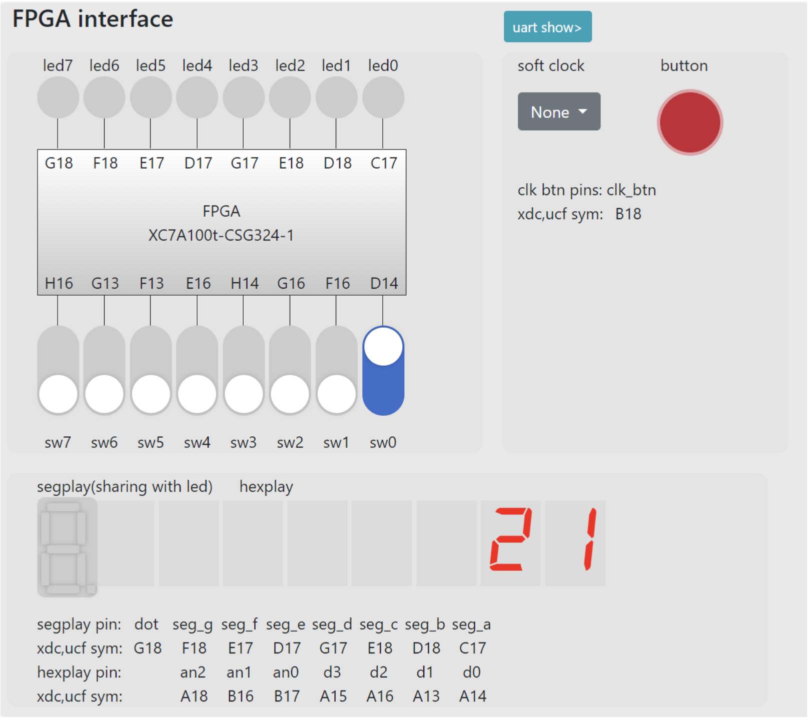
```
module t3(
    input clk,rst,sw,btn,
    output reg [2:0] hexplay_an,
    reg [3:0] hexplay_data
    );
    reg b;
    always @(posedge clk)   b <= btn;
    reg[7:0] curr,next;
    initial curr=8'h1f;
    initial next=8'h1f;
    always @(posedge clk)begin
        if(sw) next <= curr + 8'h1;
        else next <= curr - 8'h1;
    end
    always @(posedge b or posedge rst)begin
        if(rst) curr <= 8'h1f;
        else curr <= next;
    end
    reg [2:0] cnt;
    always @(posedge clk)begin
        cnt <= cnt+1;
        hexplay_an <= {2'b00,cnt[2]};
        if(cnt[2]) hexplay_data <= curr[7:4];
        else hexplay_data <= curr[3:0];
    end
endmodule
```

约束文件如图



```
## Clock signal
set_property -dict { PACKAGE_PIN E3    IOSTANDARD LVCMOS33 } [get_ports { clk }]; #IO_L12P_T1_

set_property -dict { PACKAGE_PIN B18    IOSTANDARD LVCMOS33 } [get_ports { btn }]; #IO_L12P_T.

## FPGAOL SWITCH

set_property -dict { PACKAGE_PIN D14   IOSTANDARD LVCMOS33 } [get_ports { sw }];
set_property -dict { PACKAGE_PIN F16   IOSTANDARD LVCMOS33 } [get_ports { rst }];


## FPGAOL HEXPLAY

set_property -dict { PACKAGE_PIN A14   IOSTANDARD LVCMOS33 } [get_ports { hexplay_data[0] }];
set_property -dict { PACKAGE_PIN A13   IOSTANDARD LVCMOS33 } [get_ports { hexplay_data[1] }];
set_property -dict { PACKAGE_PIN A16   IOSTANDARD LVCMOS33 } [get_ports { hexplay_data[2] }];
set_property -dict { PACKAGE_PIN A15   IOSTANDARD LVCMOS33 } [get_ports { hexplay_data[3] }];
set_property -dict { PACKAGE_PIN B17   IOSTANDARD LVCMOS33 } [get_ports { hexplay_an[0] }];
set_property -dict { PACKAGE_PIN B16   IOSTANDARD LVCMOS33 } [get_ports { hexplay_an[1] }];
set_property -dict { PACKAGE_PIN A18   IOSTANDARD LVCMOS33 } [get_ports { hexplay_an[2] }];
```

烧写结果如图

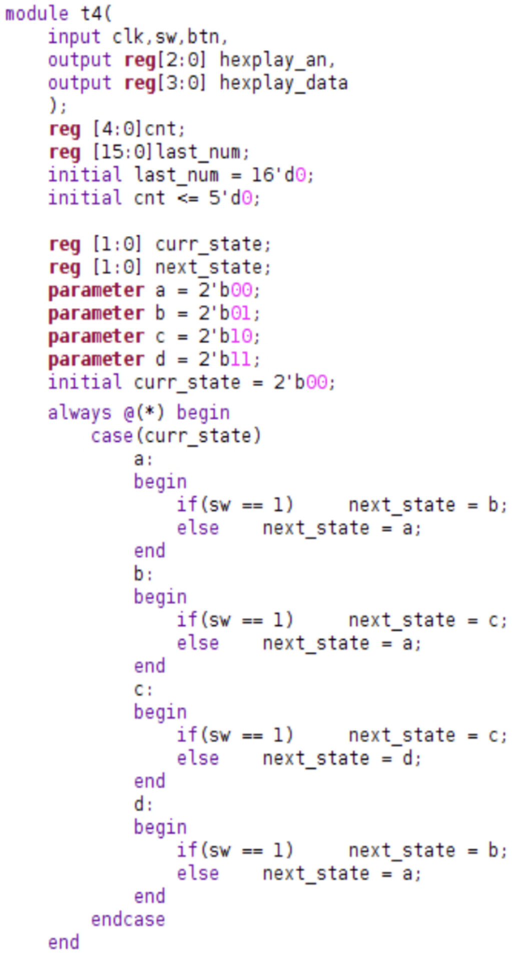

FPGA interface
uart show>
led7 led6 led5 led4 led3 led2 led1 led0
G18 F18 E17 D17 G17 E18 D18 C17
FPGA
XC7A100t-CSG324-1
H16 G13 F13 E16 H14 G16 F16 D14
sw7 sw6 sw5 sw4 sw3 sw2 sw1 sw0
soft clock
button
None
segplay(sharing with led)
hexplay
2 1

clk btn pins: clk_btn
xdc,ucf sym: B18

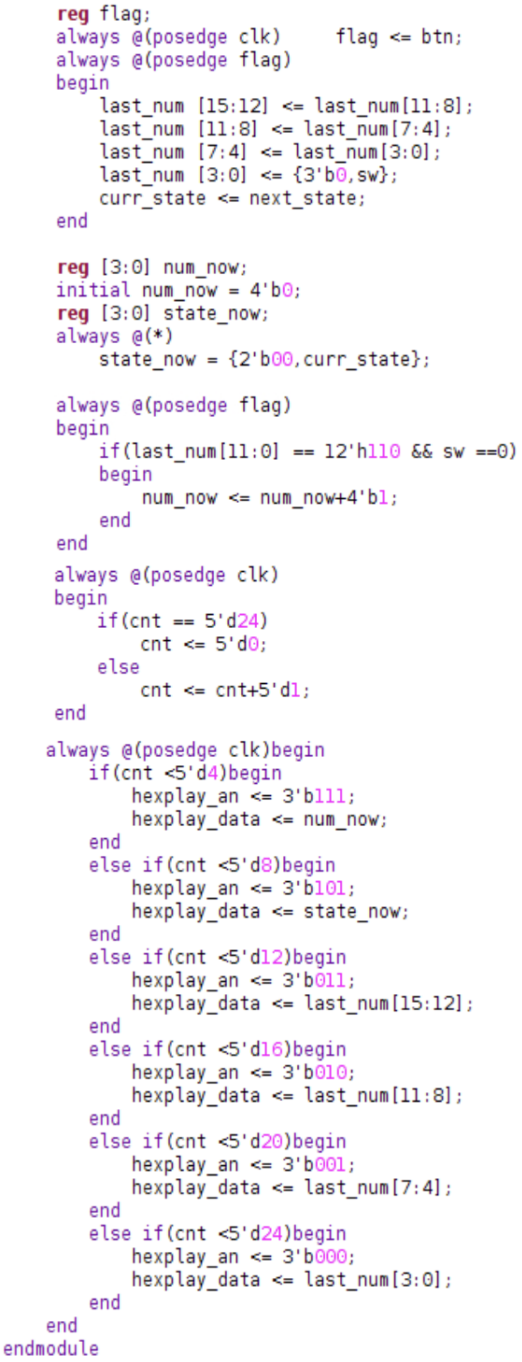
segplay pin:	dot	seg_g	seg_f	seg_e	seg_d	seg_c	seg_b	seg_a
xdc,ucf sym:	G18	F18	E17	D17	G17	E18	D18	C17
hexplay pin:		an2	an1	an0	d3	d2	d1	d0
xdc,ucf sym:		A18	B16	B17	A15	A16	A13	A14

题目 4

设计文件代码如图

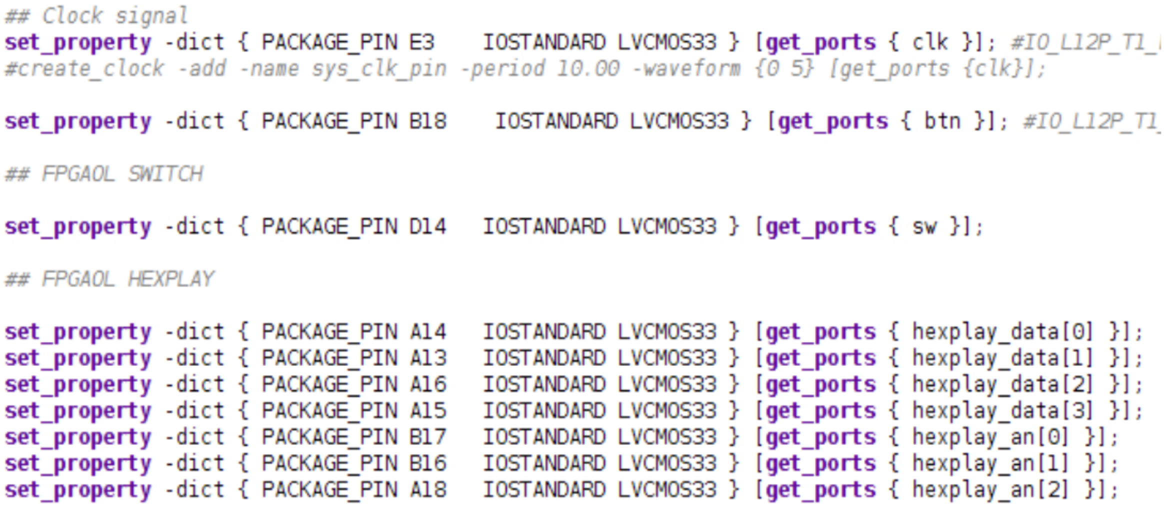


```
module t4(
    input clk,sw,btn,
    output reg[2:0] hexplay_an,
    output reg[3:0] hexplay_data
    );
    reg [4:0]cnt;
    reg [15:0]last_num;
    initial last_num = 16'd0;
    initial cnt <= 5'd0;

    reg [1:0] curr_state;
    reg [1:0] next_state;
    parameter a = 2'b00;
    parameter b = 2'b01;
    parameter c = 2'b10;
    parameter d = 2'b11;
    initial curr_state = 2'b00;
    always @(*) begin
        case(curr_state)
            a:
            begin
                if(sw == 1)     next_state = b;
                else    next_state = a;
            end
            b:
            begin
                if(sw == 1)     next_state = c;
                else    next_state = a;
            end
            c:
            begin
                if(sw == 1)     next_state = c;
                else    next_state = d;
            end
            d:
            begin
                if(sw == 1)     next_state = b;
                else    next_state = a;
            end
        endcase
    end
```



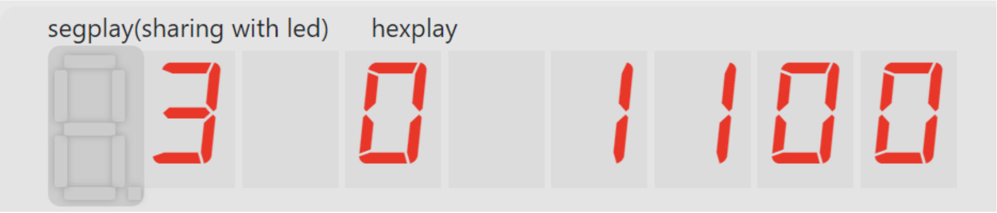
```
    reg flag;
    always @(posedge clk)     flag <= btn;
    always @(posedge flag)
    begin
        last_num [15:12] <= last_num[11:8];
        last_num [11:8]  <= last_num[7:4];
        last_num [7:4]  <= last_num[3:0];
        last_num [3:0]  <= {3'b0,sw};
        curr_state <= next_state;
    end

    reg [3:0] num_now;
    initial num_now = 4'b0;
    reg [3:0] state_now;
    always @(*)
        state_now = {2'b00,curr_state};

    always @(posedge flag)
    begin
        if(last_num[11:0] == 12'h110 && sw ==0)
        begin
            num_now <= num_now+4'b1;
        end
    end

    always @(posedge clk)
    begin
        if(cnt == 5'd24)
            cnt <= 5'd0;
        else
            cnt <= cnt+5'd1;
    end

    always @(posedge clk)begin
        if(cnt <5'd4)begin
            hexplay_an <= 3'b111;
            hexplay_data <= num_now;
        end
        else if(cnt <5'd8)begin
            hexplay_an <= 3'b101;
            hexplay_data <= state_now;
        end
        else if(cnt <5'd12)begin
            hexplay_an <= 3'b011;
            hexplay_data <= last_num[15:12];
        end
        else if(cnt <5'd16)begin
            hexplay_an <= 3'b010;
            hexplay_data <= last_num[11:8];
        end
        else if(cnt <5'd20)begin
            hexplay_an <= 3'b001;
            hexplay_data <= last_num[7:4];
        end
        else if(cnt <5'd24)begin
            hexplay_an <= 3'b000;
            hexplay_data <= last_num[3:0];
        end
    end
endmodule
```

约束文件如图



```
## Clock signal
set_property -dict { PACKAGE_PIN E3    IOSTANDARD LVCMOS33 } [get_ports { clk }]; #IO_L12P_T1_
#create_clock -add -name sys_clk_pin -period 10.00 -waveform {0 5} [get_ports {clk}];

set_property -dict { PACKAGE_PIN B18    IOSTANDARD LVCMOS33 } [get_ports { btn }]; #IO_L12P_T1_

## FPGAOL SWITCH

set_property -dict { PACKAGE_PIN D14   IOSTANDARD LVCMOS33 } [get_ports { sw }];

## FPGAOL HEXPLAY

set_property -dict { PACKAGE_PIN A14   IOSTANDARD LVCMOS33 } [get_ports { hexplay_data[0] }];
set_property -dict { PACKAGE_PIN A13   IOSTANDARD LVCMOS33 } [get_ports { hexplay_data[1] }];
set_property -dict { PACKAGE_PIN A16   IOSTANDARD LVCMOS33 } [get_ports { hexplay_data[2] }];
set_property -dict { PACKAGE_PIN A15   IOSTANDARD LVCMOS33 } [get_ports { hexplay_data[3] }];
set_property -dict { PACKAGE_PIN B17   IOSTANDARD LVCMOS33 } [get_ports { hexplay_an[0] }];
set_property -dict { PACKAGE_PIN B16   IOSTANDARD LVCMOS33 } [get_ports { hexplay_an[1] }];
set_property -dict { PACKAGE_PIN A18   IOSTANDARD LVCMOS33 } [get_ports { hexplay_an[2] }];
```

烧写结果如图



5 总结与思考

1. 进一步熟悉了 FPGA 的开发流程
2. 本次实验难度较大
3. 本次实验任务量较多
4. 无