

浙江大学

本科实验报告

课程名称：数字逻辑设计

姓 名：王伟杰

学 院：计算机学院

系：软件工程

专 业：软件工程

学 号：3210106034

指导教师：马德

2022 年 12 月12 日

浙江大学实验报告

课程名称： 数字逻辑设计 实验类型：

实验项目名称： 移位寄存器设计与应用

学生姓名： 王伟杰 专业： 软件工程 学号： 3210106034

同组学生姓名： 王熠、陈苇远 指导老师： 马德

实验地点： 东 4-509 实验日期： 2022 年 12 月 12 日

一、实验目的

1.1 掌握支持并行输入的移位寄存器的工作原理。

1.2 掌握支持并行输入的移位寄存器的设计方法。

二、实验内容

- 任务1：设计8位带并行输入的右移移位寄存器
- 任务2：设计主板LED灯驱动模块
- 任务3：设计主板七段数码管驱动模块

三、操作方法与实验步骤

3.1 设计8位带并行输入的右移移位寄存器

1.新建工程 ShfitReg8b，新建 Verilog 文件 ShiftReg8b，编写代码如下：

```
////////////////////////////////////  
module ShfitReg8b(  
    input wire clk, S_L, s_in,  
    input wire [7:0] p_in,  
    output wire [7:0] Q  
);  
    reg [7:0] rev;  
    initial rev = 0;  
    // S_L = 0, 串行移位  
    // S_L = 1, 并行输入  
    always@(posedge clk)  
    begin  
        if(S_L)  
            rev = p_in;  
        else begin  
            rev = rev >> 1;  
            rev[7] = s_in;  
        end  
    end  
    assign Q = rev;  
endmodule
```

2.设计仿真激励代码

```
1  initial begin
2      // Initialize Inputs
3      clk = 0;
4      L = 0;
5      s_in = 0;
6      p_in = 0;
7
8      #100;
9
10     // Add stimulus here
11     S_L = 0;
12     s_in = 1;
13     p_in = 0;
14     #200;
15     S_L = 1;
16     s_in = 0;
17     p_in = 8'b0101_0101;
18     #500;
19 end
20
21 always begin
22     clk = 0; #20;
23     clk = 1; #20;
24 end
```

3.2 设计主板LED灯驱动模块

1.新建工程，利用ShfitReg8b模块，设计LED灯驱动模块LED_DRV

```
1  module ShfitReg9b(
2      input wire L,
3      input wire clk,
4      input wire shfit_in,
5      input wire [8:0] par_in,
6      output wire [8:0] Q
7  );
8
9      wire [8:0] Q_out;
10     wire S;
11     assign Q [8:0] = Q_out [8:0];
12     wire or0, or1, or2, or3, or4, or5, or6, or7, or8;
13     wire and00, and01, and02, and03, and04, and05, and06, and07, and08;
14
15     wire and10, and11, and12, and13, and14, and15, and16, and17, and18;
16
17     INV inv0(.O(S), .I(L));
18
19     FD #(.INIT(1'b0)) fd0(.D(or0), .C(clk), .Q(Q_out[8]));
20     FD #(.INIT(1'b0)) fd1(.D(or1), .C(clk), .Q(Q_out[7]));
21     FD #(.INIT(1'b0)) fd2(.D(or2), .C(clk), .Q(Q_out[6]));
22     FD #(.INIT(1'b0)) fd3(.D(or3), .C(clk), .Q(Q_out[5]));
```

```

22     FD #(.INIT(1'b0)) fd4(.D(or4), .C(clk), .Q(Q_out[4]));
23     FD #(.INIT(1'b0)) fd5(.D(or5), .C(clk), .Q(Q_out[3]));
24     FD #(.INIT(1'b0)) fd6(.D(or6), .C(clk), .Q(Q_out[2]));
25     FD #(.INIT(1'b0)) fd7(.D(or7), .C(clk), .Q(Q_out[1]));
26     FD #(.INIT(1'b0)) fd8(.D(or8), .C(clk), .Q(Q_out[0]));
27
28     OR2 or2_0(.O(or0), .I0(and00), .I1(and10));
29     OR2 or2_1(.O(or1), .I0(and01), .I1(and11));
30     OR2 or2_2(.O(or2), .I0(and02), .I1(and12));
31     OR2 or2_3(.O(or3), .I0(and03), .I1(and13));
32     OR2 or2_4(.O(or4), .I0(and04), .I1(and14));
33     OR2 or2_5(.O(or5), .I0(and05), .I1(and15));
34     OR2 or2_6(.O(or6), .I0(and06), .I1(and16));
35     OR2 or2_7(.O(or7), .I0(and07), .I1(and17));
36     OR2 or2_8(.O(or8), .I0(and08), .I1(and18));
37
38     AND2 and2_00(.O(and00), .I0(s), .I1(shfit_in));
39     AND2 and2_01(.O(and01), .I0(s), .I1(Q_out[8]));
40     AND2 and2_02(.O(and02), .I0(s), .I1(Q_out[7]));
41     AND2 and2_03(.O(and03), .I0(s), .I1(Q_out[6]));
42     AND2 and2_04(.O(and04), .I0(s), .I1(Q_out[5]));
43     AND2 and2_05(.O(and05), .I0(s), .I1(Q_out[4]));
44     AND2 and2_06(.O(and06), .I0(s), .I1(Q_out[3]));
45     AND2 and2_07(.O(and07), .I0(s), .I1(Q_out[2]));
46     AND2 and2_08(.O(and08), .I0(s), .I1(Q_out[1]));
47
48     AND2 and2_10(.O(and10), .I0(s), .I1(par_in[8]));
49     AND2 and2_11(.O(and11), .I0(s), .I1(par_in[7]));
50     AND2 and2_12(.O(and12), .I0(s), .I1(par_in[6]));
51     AND2 and2_13(.O(and13), .I0(s), .I1(par_in[5]));
52     AND2 and2_14(.O(and14), .I0(s), .I1(par_in[4]));
53     AND2 and2_15(.O(and15), .I0(s), .I1(par_in[3]));
54     AND2 and2_16(.O(and16), .I0(s), .I1(par_in[2]));
55     AND2 and2_17(.O(and17), .I0(s), .I1(par_in[1]));
56     AND2 and2_18(.O(and18), .I0(s), .I1(par_in[0]));
57
58 endmodule

```

2.新建top模块

```

1 module Top(
2     input clk,
3     input start,
4     input [15:0] LED,
5     output ser_out,
6     output finish,
7     output [8:0] temp_h,
8     output [7:0] temp_l
9 );
10
11 wire q, q_n, s, r, fin;
12 assign finish = fin;
13 wire [8:0] par_high = {1'b0, LED[15:8]};
14 wire [7:0] par_low = {LED[7:0]};
15 wire [8:0] q_high;

```

```

16     wire [7:0] q_low;
17
18     AND2 is_convert(.O(s), .I0(start), .I1(fin));
19     INV is_end(.O(r), .I(fin));
20     NOR2 sr1(q_n, s, q);
21     NOR2 sr2(q, r, q_n);
22
23     assign temp_h = q_high;
24     assign temp_l = q_low;
25
26     ShfitReg9b m0(.clk(clk), .Q(q_high), .shfit_in(1'b1),
27 .par_in(par_high), .L(q));
28     ShfitReg8b m1(.clk(clk), .Q(q_low), .shfit_in(q_high[0]),
29 .par_in(par_low), .L(q));
30
31     assign ser_out = q_low[0];
32     assign fin = q_high[8] & q_high[7] & q_high[6] & q_high[5] &
33 q_high[4] & q_high[3] & q_high[2] & q_high[1] &
34 q_high[0] & q_low[7] & q_low[6] & q_low[5] &
35 q_low[4] & q_low[3] & q_low[2] & q_low[1] & q_low[0];
36
37 endmodule

```

3.激励仿真

```

1 module top_tb;
2
3     reg clk;
4     reg start;
5     reg [15:0] LED;
6
7     wire ser_out;
8     wire finish;
9     wire [8:0] temp_h;
10    wire [7:0] temp_l;
11
12    Top uut (
13        .clk(clk),
14        .start(start),
15        .LED(LED),
16        .ser_out(ser_out),
17        .finish(finish),
18        .temp_h(temp_h),
19        .temp_l(temp_l)
20    );
21
22    initial forever begin
23        #20; clk = ~clk;
24    end
25
26    initial begin
27        clk = 0;
28        start = 0;
29        LED = 0;
30
31        #100;

```

```

31     LED = 16'b0101_0010_0100_0010; // 5242
32     start = 1;
33     end
34
35 endmodule

```

3.3 设计主板七段数码管驱动模块

1.利用ShfitReg8b模块，设计主板七段数码管驱动模块SEG_DRV

```

1  module Top(
2      input clk,
3      input start,
4      input [31:0] num,
5      input CR,
6      output SEGDT,
7      output SEGCLK,
8      output SEGCLR,
9      output [63:0] temp_cnum
10 );
11
12     wire [63:0] cnum;
13     wire p2s_fin;
14     assign temp_cnum = cnum;
15
16     OR2 seg_clk(.O(SEGCLK), .I0(p2s_fin), .I1(clk));
17     INV clear(.O(SEGCLR), .I(CR));
18
19     MC14495 m0(.num(num[31:28]), .cnum(cnum[63:56]));
20     MC14495 m1(.num(num[27:24]), .cnum(cnum[55:48]));
21     MC14495 m2(.num(num[23:20]), .cnum(cnum[47:40]));
22     MC14495 m3(.num(num[19:16]), .cnum(cnum[39:32]));
23     MC14495 m4(.num(num[15:12]), .cnum(cnum[31:24]));
24     MC14495 m5(.num(num[11:8]), .cnum(cnum[23:16]));
25     MC14495 m6(.num(num[7:4]), .cnum(cnum[15:8]));
26     MC14495 m7(.num(num[3:0]), .cnum(cnum[7:0]));
27
28     P2S m8(.clk(clk), .cnum(cnum), .start(start), .finish(p2s_fin),
29         .ser_out(SEGDT));
30 endmodule

```

2.激励仿真

```

1  module top_tb;
2
3      reg clk;
4      reg start;
5      reg [31:0] num;
6      reg CR;
7
8      wire SEGDT;
9      wire SEGCLK;
10     wire SEGCLR;

```

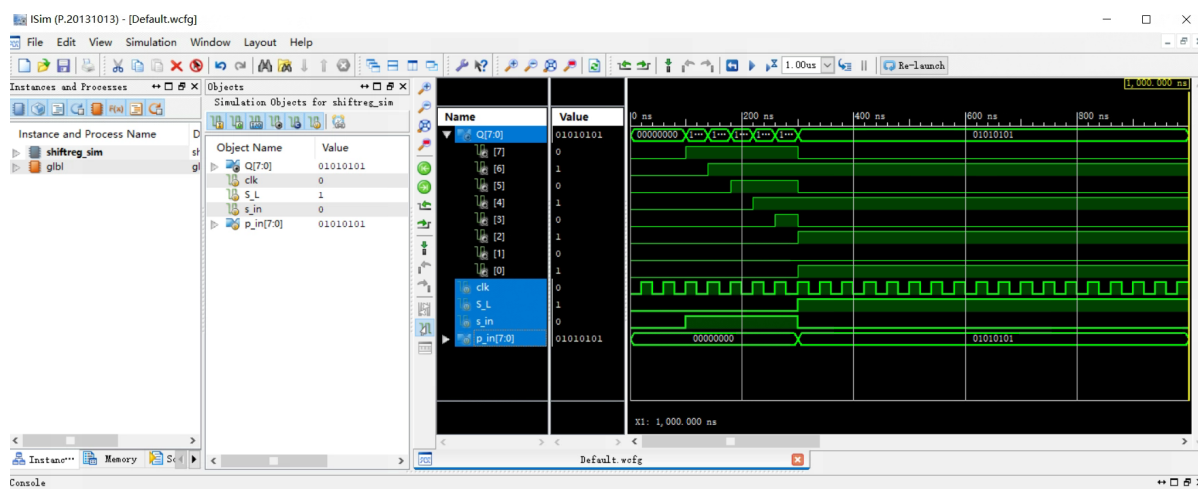
```

11     wire [63:0] temp_cnum;
12
13     Top uut (
14         .clk(clk),
15         .start(start),
16         .num(num),
17         .CR(CR),
18         .SEGDT(SEGDT),
19         .SEGCLK(SEGCLK),
20         .SEGCLR(SEGCLR),
21         .temp_cnum(temp_cnum)
22     );
23
24     initial begin
25         clk = 0;
26         start = 0;
27         num = 0;
28         CR = 0;
29
30         #1000;
31         num = 32'b0001_0000_0001_0000_0101_0010_0100_0010; // 10105242
32         start = 1;
33
34         #100;
35         start = 0;
36     end
37
38     always begin
39         #20; clk = ~clk;
40     end
41
42 endmodule

```

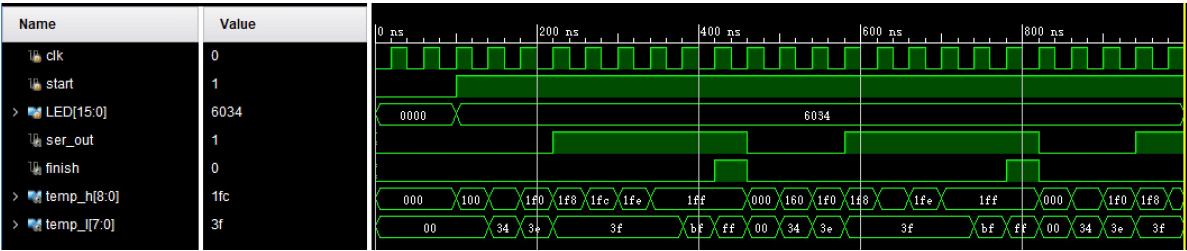
四、实验结果与分析

4.1 设计8位带并行输入的右移移位寄存器



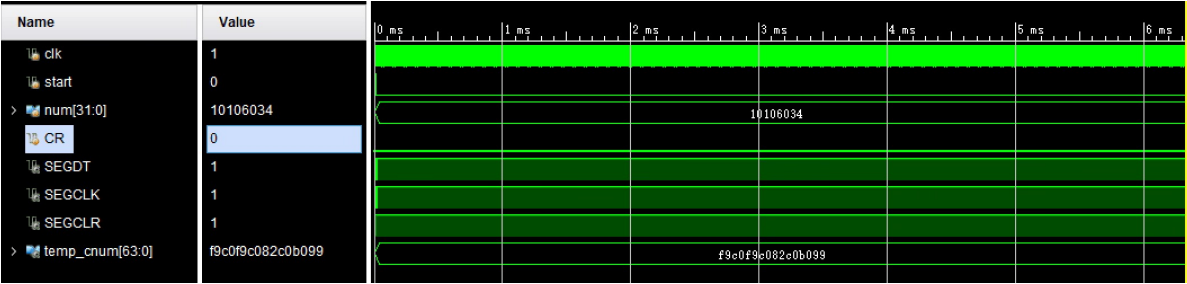
符合预期。

4.2 设计主板LED灯驱动模块



正确显示了学号后四位6034。

4.3 设计主板七段数码管驱动模块



正确显示了学号后八位10106034。

五、讨论、心得

本次实验的内容比较丰富，难度也较大，花费时间比较多。本次实验我们在部分模块中使用了行为描述进行实现，这样的实现更加简单，也可以在一定程度上忽略具体的电路连接逻辑。

本次实验中，我根据实验需求，对一些模块的编写逻辑和实现功能作了一些调整和简化；同时，也使用了之前实验中实现的一些模块。这让我们对以往编写的模块的功能有了更为清晰的认知，也进一步提高了相关的应用能力。