

姓名: Cantjie 学号: 班级:  
实验日期: 2018-05-18 交报告日期: 2018-05-

## 实验题目:

### Lab10 时序逻辑电路设计

## 实验内容:

- 1、 任意模值计数器
- 2、 动态扫描数码管显示
- 3、 串并转换

## 实验任务及完成思路:

### 1、 任意模值计数器

此处完成模 24 的计数器。

为了用 LED 灯和七段数码管分别表示, 需要设计两个电路, 用 5 位二进制表示时, 设计模 24 计数器, 而用七段数码管显示时, 实际上需要设计模 3 和模 10 两个计数器并在 23 时同时归零。

二进制显示时, 利用两个 74LS163 级联, 低位 RCO 接高位 ENT 和 ENP。同时为了在计数达到 23 时, 使计数清零, 只需将 Q0、Q1、Q2、Q4 经与非门后接到两个 CLRN 端即可, 因为 23 的二进制表示为 10111。

十进制显示时, 利用两个 74LS163, 分别作为十位和个位。

①个位清零条件为: 个位达到 9 或十位和个位分别达到 2、3;

$$\text{CLR}_{N0} = (\sim (Q_5 \cdot Q_1 \cdot Q_0)) \cdot (\sim (Q_3 \cdot Q_0))$$

②十位清零条件为: 十位和各位达到 2、3;

$$\text{CLR}_{N1} = \sim (Q_5 \cdot Q_1 \cdot Q_0)$$

③十位进位条件为: 个位达到 9。

$$\text{ENT}_1 = \text{ENP}_1 = Q_3 \cdot Q_0$$

将时钟分频后作为输入 clk, 分别以实验板上的五个 LED 灯和六位动态扫描数码管的低二位作为输出显示。

### 2、 动态扫描数码管显示

考虑普通家用有线电视的制式为每秒 24 帧, 因此六段扫描数码管扫描频率达 25Hz 以上即可, 考虑到响应的时间, 频率也不可过高, 扫描时在每一位停留几毫秒即可。

扫描的实现, 先利用 74LS163 做出一个模 6 计数器, 计数序列 0-5, 然后经过一个 74LS138 译码后取反, 得到 6 位 one-hot 序列的循环, 然后经过和传入数据做与、或运算选出有效输出。

除时钟外, 扫描数码管还需要 6 个 4 位二进制数作为输入, s0-s5、a-g 作为输出, 即  $6 \times 4 = 24$  位输入,  $6 + 7 = 13$  位输出。

### 3、 串并转换

为了实现 1 位入 4 位出的串并转化，输入参数有串口时钟 clk，串口输入 data\_in，输出四位 data[3..0]。

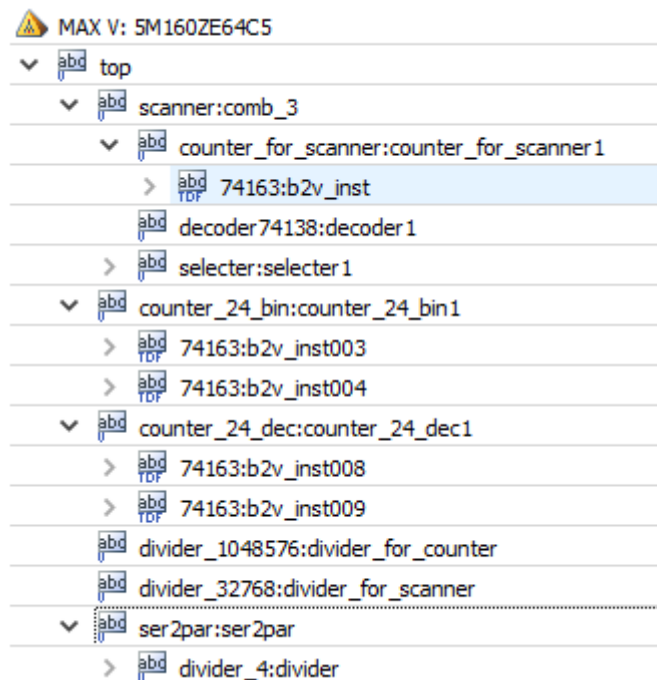
在串口端，每一个时钟上升沿读取一个 data\_in，同时将 4 位移位寄存器左移 1 位，将 data\_in 放入最低位。

在并口端，将串口时钟四分频得到并口时钟，在并口时钟上升沿时，将 4 位寄存器一次性获取。

以两个按键开关分别作为串口时钟 clk 和串口输入 data\_in，以六位动态扫描数码管的高四位作为并口输出。

## 功能实现：

整体结构：



按照自顶向下和模块化的设计思想，先写出顶层实体：

顶层实体包含了三个主要部分：

- ①计数器时钟分频器和计数器的不同进制显示的两个程序
- ②串并转化的程序
- ③扫描器时钟分频器和扫描器

前两个部分获得数据后交给③，由六位七段数码管显示

top.v:

```
module top(  
    clk, //待分配的时钟  
    clk_for_exp3, //以按键开关作为串并转化处的时钟  
    in_data, //串并转化中串口数据  
    clrn_for_bin, //二进制显示的计数器的 clrn  
    clrn_for_dec, //十进制显示的计数器的 clrn  
    sev_out, //数码管的 a-g
```

```

        sel_out,//s0-s5,one-hot
        bin_out,//计数器的二进制输出
    );

input clk,clrn_for_bin,clrn_for_dec,clk_for_exp3,in_data;
output [5:0] sel_out;
output [6:0] sev_out;
output [4:0] bin_out;

wire [3:0] data1,data2,data3,data4,data5,data6;//六位动态扫描数码管的六个数据
wire [4:0] bin_out;
wire clk_count,clk_scan; //计数时钟和扫描时钟

//分频程序
divider_1048576 divider_for_counter(clk,clk_count);
divider_32768 divider_for_scanner(clk,clk_scan);

//串并转化程序
ser2par ser2par(clk_for_exp3,in_data,data3[0],data4[0],data5[0],data6[0]);

//24 位计数器的二进制和十进制显示程序
counter_24_bin
counter_24_bin1(clk_count,clrn_for_bin,bin_out[0],bin_out[1],bin_out[2],bin_out[3],bin_out[4]);
counter_24_dec
counter_24_dec1(clk_count,clrn_for_dec,data1[0],data1[1],data1[2],data1[3],data2[0],data2[1]);

//六位动态扫描数码管扫描和显示
scanner scanner1(clk_scan,data1,data2,data3,data4,data5,data6,sev_out,sel_out);

endmodule

```

分频器程序:

divider\_32768.v

```

module divider_32768(in_clk,out_clk);

input in_clk;
output reg out_clk;

reg [15:0] temp;

always @(posedge in_clk) begin

temp <= temp + 1;
case(temp[15:0])

```

```

        16'b0: out_clk <= ~ out_clk;
    default::
    endcase
end
endmodule

```

串并转化程序:

ser2par.v

```

module ser2par(clk,in,data0,data1,data2,data3);

input clk,in;
output reg data0,data1,data2,data3;

reg [3:0] temp;

wire par_clk;
divider_4 divider(clk,par_clk);

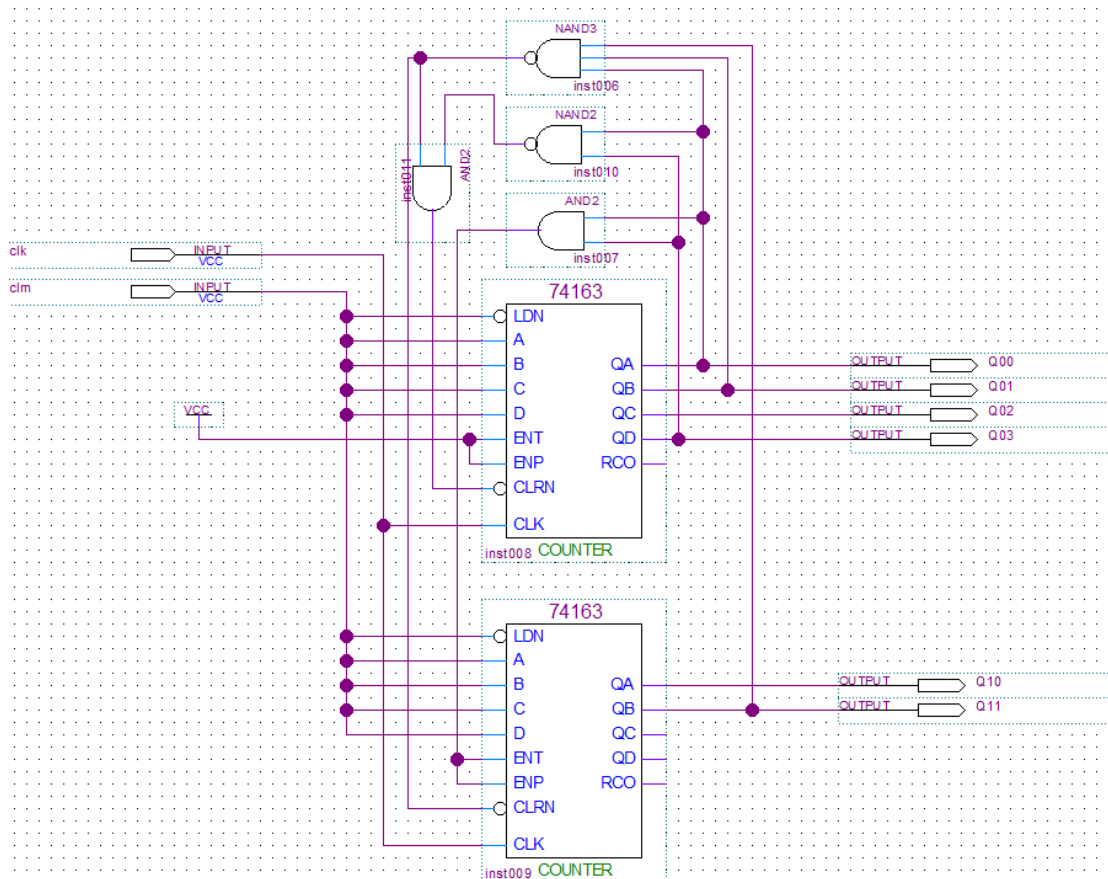
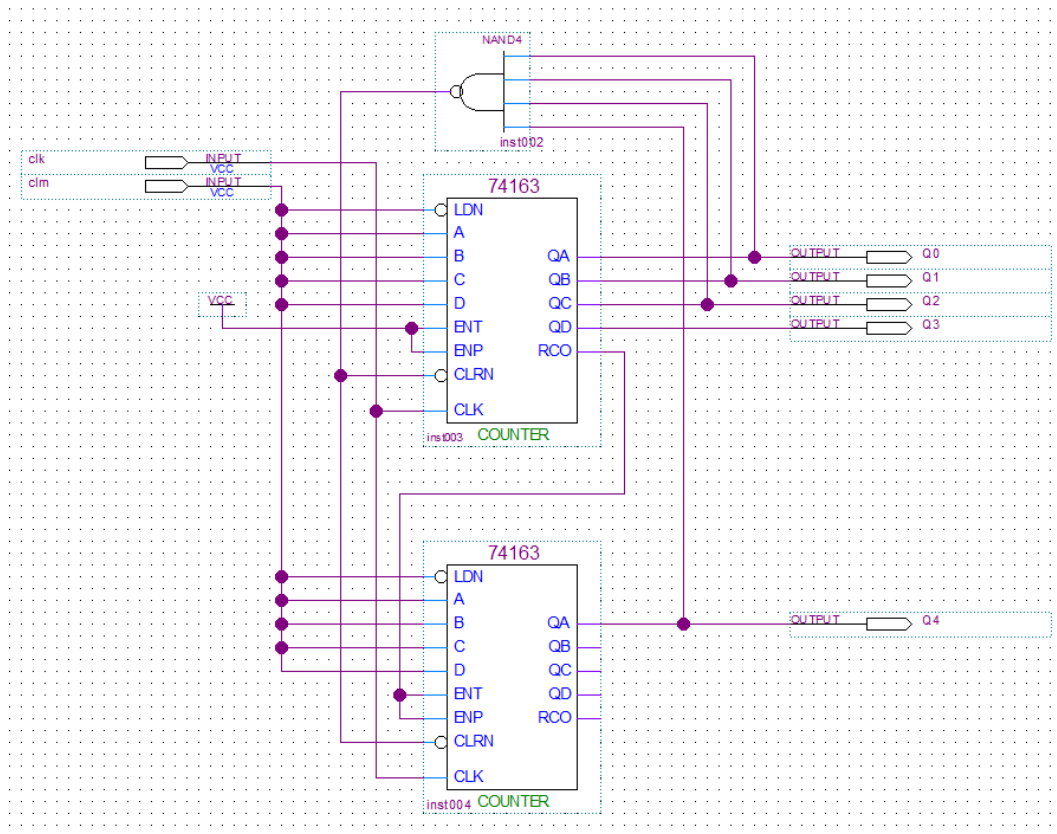
//串口端，移位寄存器
always @(posedge clk) begin
    temp <= temp << 1;
    temp[0] <= in;
end

//并口端，时钟已经四分频
always @(posedge par_clk) begin
    data3 <= temp[3];
    data2 <= temp[2];
    data1 <= temp[1];
    data0 <= temp[0];
end

endmodule

```

模 24 计数器电路（二进制和十进制）:



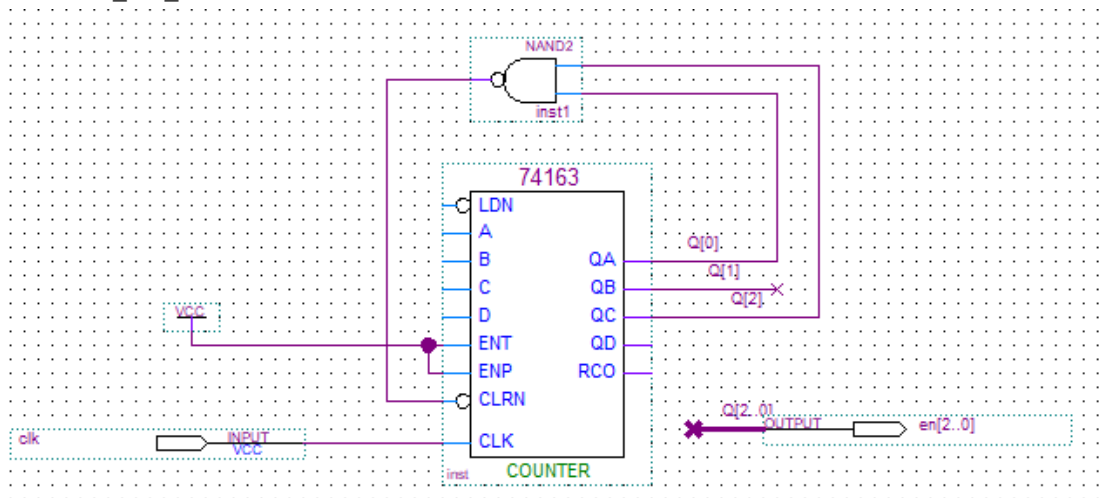
六位动态扫描数码管程序:

scanner.v

```
module scanner(clk,data1,data2,data3,data4,data5,data6,sev_out,sel_out,);  
  
input clk;  
input [3:0] data1, data2, data3, data4, data5, data6;  
output wire [6:0] sev_out;  
output wire [5:0] sel_out;  
  
wire [2:0] en;  
wire [7:0] sel;  
  
counter_for_scanner counter_for_scanner1(clk,en);  
decoder74138 decoder1(en,sel);  
selecter selecter1(sel[5:0],data1,data2,data3,data4,data5,data6,sev_out);  
  
assign sel_out[5:0] = sel[5:0];  
  
endmodule
```

六位动态扫描数码管程序中的模 6 计数器:

counter\_for\_scanner



六位动态扫描数码管程序中的 selecter:

selecter.v

```
module selecter(sel,data1,data2,data3,data4,data5,data6,sev_out);  
  
input [5:0] sel;  
input [3:0] data1,data2,data3,data4,data5,data6;  
output wire [6:0] sev_out;  
  
wire [3:0] data_selected;
```

```

assign data_selected[0] = (data1[0] & (~sel[0])) | (data2[0] & (~sel[1])) | (data3[0] & (~sel[2])) | (data4[0] &
(~sel[3])) | (data5[0] & (~sel[4])) | (data6[0] & (~sel[5]));
assign data_selected[1] = (data1[1] & (~sel[0])) | (data2[1] & (~sel[1])) | (data3[1] & (~sel[2])) | (data4[1] &
(~sel[3])) | (data5[1] & (~sel[4])) | (data6[1] & (~sel[5]));
assign data_selected[2] = (data1[2] & (~sel[0])) | (data2[2] & (~sel[1])) | (data3[2] & (~sel[2])) | (data4[2] &
(~sel[3])) | (data5[2] & (~sel[4])) | (data6[2] & (~sel[5]));
assign data_selected[3] = (data1[3] & (~sel[0])) | (data2[3] & (~sel[1])) | (data3[3] & (~sel[2])) | (data4[3] &
(~sel[3])) | (data5[3] & (~sel[4])) | (data6[3] & (~sel[5]));

seven_segment

seven_segment1(data_selected[0],data_selected[1],data_selected[2],data_selected[3],sev_out[0],sev_out[
1],sev_out[2],sev_out[3],sev_out[4],sev_out[5],sev_out[6]);

endmodule

```

(seven\_segment 在之前实验的基础上取反从而兼容共阴极数码管，74138 为往次实验内容，故在此不再列出。分频器原理类似，仅列出了一个)

## 仿真结果：

仿真需要重新调整分频器分频倍数，且仅二进制显示的计数器易于在仿真中观察结果，而仿真结果又与之前实验相似，故在此报告中略去仿真结果。