

奋斗的小孩之 altera 系列

第二十八篇 数码管

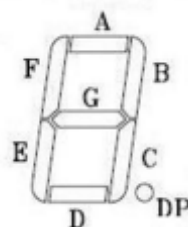
对于每一个的小实验，我们都可以把它看作是一个小项目，逐步的去分析，设计，调试，最后完成功能。下面我们就开始我们的“小项目”。

项目名称：数码管

具体要求：使用数码管显示任意数字。

项目分析：

1. 什么是数码管



数码管也称 LED 数码管，不同行业人士对数码管的称呼不一样，其实都是同样的产品。

数码管按段数可分为七段数码管和八段数码管，八段数码管比七段数码管多一个发光二极管单元，也就是多一个小数点（DP）这个小数点可以更精确的表示数码管想要显示的内容；按能显示多少个（8）可分为 1 位、2 位、3 位、4 位、5 位、6 位、7 位等数码管。



按发光二极管单元连接方式可分为共阳极数码管和共阴极数码管。共阳数码管是指将所有发光二极管的阳极接到一起形成公共阳极 (COM) 的数码管, 共阳数码管在应用时应将公共极 COM 接到 +5V, 当某一字段发光二极管的阴极为低电平时, 相应字段就点亮, 当某一字段的阴极为高电平时, 相应字段就不亮。共阴数码管是指将所有发光二极管的阴极接到一起形成公共阴极 (COM) 的数码管, 共阴数码管在应用时应将公共极 COM 接到地线 GND 上, 当某一字段发光二极管的阳极为高电平时, 相应字段就点亮, 当某一字段的阳极为低电平时, 相应字段就不亮。

例如: 如果想要显示数字“1”, 则 B、C 段亮, 其他段不亮就可以了。

2. 驱动的方式

数码管要正常显示, 就要用驱动电路来驱动数码管的各个段码, 从而显示出我们要的数字, 因此根据数码管的驱动方式的不同, 可以分为静态式和动态式两类。

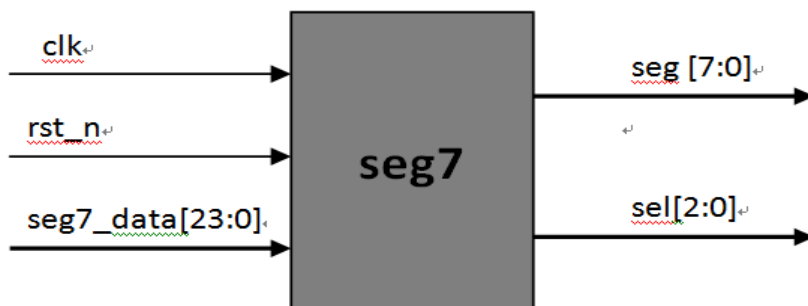
静态显示驱动

静态驱动也称直流驱动。静态驱动是指每个数码管的每一个段码都由一个 FPGA 的 I/O 端口进行驱动。静态驱动的优点是编程简单, 显示亮度高, 缺点是占用 I/O 端口多, 如驱动 6 个数码管静态显示则需要 $6 \times 8 = 48$ 个 I/O 端口来驱动。

动态显示驱动

数码管动态显示接口是单片机中应用最为广泛的一种显示方式之一，动态驱动是将所有数码管的 8 个显示笔划“a, b, c, d, e, f, g, dp”的同名端连在一起，另外为每个数码管的公共极 COM 增加位选通控制电路，位选通由各自独立的 I/O 线控制，当 FPGA 输出字形码时，所有数码管都接收到相同的字形码，但究竟是哪个数码管会显示出字形，取决于 FPGA 对位选通 COM 端电路的控制，所以我们只要将需要显示的数码管的选通控制打开，该位就显示出字形，没有选通的数码管就不会亮。通过分时轮流控制各个数码管的的 COM 端，就使各个数码管轮流受控显示，这就是动态驱动。在轮流显示过程中，每位数码管的点亮时间为 1~2ms，由于人的视觉暂留现象及发光二极管的余辉效应，尽管实际上各位数码管并非同时点亮，但只要扫描的速度足够快，给人的印象就是一组稳定的显示数据，不会有闪烁感，动态显示的效果和静态显示是一样的，能够节省大量的 I/O 端口，而且功耗更低。

架构图如下：



上述架构的设计，基于笔者所使用开发板。开发板的数码管为共阳数码管，共有 6 个，并且位选信号也是经过 3-8 译码器之后连接到数码管硬件电路上。我们选择动态显示驱动。

seg7_data[23:0]: 数码管的正常显示数据为 0~9，二进制为 4' b0000~4' b1001。由于数码管有六个，故而显示的数据应该是 6x4=24（位）。

seg[7:0]: 段选信号。

sel[2:0]: 6 个数码管应该有 6 个位选信号，为了节省 I/O，通过 3-8 译码器与数码管的硬件电路相连接，故而位选信号 3 位。

系统设计:

1. 工程的名称: seg7

2. 实现方法:

动态显示要求扫描的频率足够快 (1-2ms)，笔者定义扫描的频率为 1KHz (每个数码管亮 1ms)。

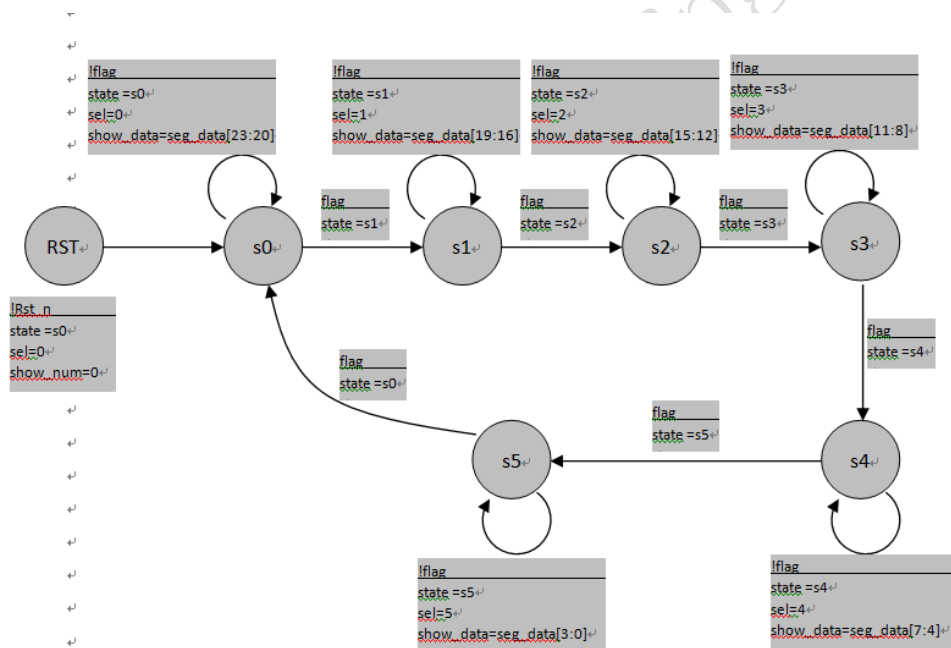
由一个计数器产生频率为 1KHz 的高脉冲，利用高脉冲控制状态机的跳转，进而完成动态驱动。在每一个状态中，控制点亮的那个数码管 (sel)，并且输出想要显示的数字 (show_data)。笔者开发板的数码管共有 6 个，对应的数码管显示对应位的数字，例如: sel=0: 选择第一个数码管。

show_data=seg7_data[23;20]: 最高的四位。



使用组合逻辑将想要显示的数字翻译成段选信号。如果使用时序逻辑，位选信号“sel”和段选信号“seg”就不是同时达到数码管的硬件电路。分析：“sel”和“show_data”同时输出，而“show_data”只是想要显示的数字，并不能够直接输出给数码管硬件电路，还需要翻译成段选信号，若使用时序逻辑，那么 seg 也将被综合成为寄存器，导致段选信号“seg”将比段选信号“sel”晚到一个时钟周期。

3. 状态转移图



设计代码如下：

/*

模块名称：seg7

模块功能：使用数码管显示任意数字



编写时间：2016-09-04

作者：至芯科技----奋斗的小孩

邮箱：zxopenhxs@126.com

*/

```
module seg7 (clk, rst_n, seg7_data, seg, sel);
```

```
    input clk;//50MHz
```

```
    input rst_n;
```

```
    input [23:0] seg7_data;
```

```
    output reg [7:0] seg;
```

```
    output reg [2:0] sel;
```

```
    parameter T1ms = 50_000;//1ms 的周期数
```

```
    reg [15:0] cnt;//1ms 计数器
```

```
    always @ (posedge clk or negedge rst_n)
```

```
        begin
```

```
            if (!rst_n)
```

```
                begin
```



```
        cnt <= 16' d0;

    end

else

    begin

        if (cnt < T1ms - 1)

            cnt <= cnt + 1' b1;

        else

            cnt <= 16' d0;

        end

    end

end

wire flag;

assign flag = (cnt == T1ms - 1) ? 1' b1 : 1' b0; //1ms 的高脉冲

localparam s0 = 3' b000,

            s1 = 3' b001,

            s2 = 3' b010,

            s3 = 3' b011,

            s4 = 3' b100,

            s5 = 3' b101;
```



```
reg [2:0] state;

reg [3:0] show_data;

always @ (posedge clk or negedge rst_n)
begin
    if (!rst_n)
        begin
            state <= s0;
            show_data <= 4' d0;
            sel <= 3' b000;
        end
    else
        begin
            case (state)
                s0 : begin
                    if (!flag)
                        begin
                            state <= s0;
                            sel <= 3' b000;
                            show_data <= seg7_data[23:20];
                        end
                    else
                        begin
                            state <= s1;
                            sel <= 3' b001;
                            show_data <= seg7_data[23:20];
                        end
                end
            endcase
        end
end
```




```
        end

        else

            begin

                state <= s1;

            end

        end

    end

s1 : begin

    if (!flag)

        begin

            state <= s1;

            sel <= 3' b001;

            show_data <= seg7_data[19:16];

        end

    else

        begin

            state <= s2;

        end

    end

end

s2 : begin
```



```
        if (!flag)
            begin
                state <= s2;
                sel <= 3' b010;
                show_data <= seg7_data[15:12];
            end
        else
            begin
                state <= s3;
            end
        end
    end

s3 : begin
    if (!flag)
        begin
            state <= s3;
            sel <= 3' b011;
            show_data <= seg7_data[11:8];
        end
    else
```



```
begin
    state <= s4;
end
end

s4 : begin
    if (!flag)
        begin
            state <= s4;
            sel <= 3' b100;
            show_data <= seg7_data[7:4];
        end
    else
        begin
            state <= s5;
        end
    end
end

s5 : begin
    if (!flag)
        begin
```



```
        state <= s5;

        sel <= 3'b101;

        show_data <= seg7_data[3:0];

    end

else

    begin

        state <= s0;

    end

end

end

default : state <= s0;

endcase

end

end

always @ (*)

begin

    if (!rst_n)

        begin

            seg = 8'b1111_1111;

        end

    end

end
```



```
        end

    else

        begin

            case (show_data)

                0 : seg = 8'b1100_0000;

                1 : seg = 8'b1111_1001;

                2 : seg = 8'b1010_0100;

                3 : seg = 8'b1011_0000;

                4 : seg = 8'b1001_1001;

                5 : seg = 8'b1001_0010;

                6 : seg = 8'b1000_0010;

                7 : seg = 8'b1111_1000;

                8 : seg = 8'b1000_0000;

                9 : seg = 8'b1001_0000;

                default : seg = 8'b0000_0000;

            endcase

        end

    end

end

endmodule
```



代码解析:

段选信号 (seg) 的最高位对应 dp (小数点), 依次是 g, f, e, d, c, b, a。例如: 数字“1”, 则 seg=8' b1111_1001;

激励代码如下:

```
/*
```

```
模块名称: seg7_tb
```

```
模块功能: 为 seg7 模块提供激励信号
```

```
编写时间: 2016-09-04
```

```
作者: 至芯科技---奋斗的小孩
```

```
邮箱: zxopenhxs@126.com
```

```
*/
```

```
`timescale 1ns/1ps
```

```
module seg7_tb;
```

```
reg clk;
```

```
reg rst_n;
```

```
reg [23:0] seg7_data;
```

```
wire [7:0] seg;
```



```
wire [2:0] sel;
```

```
parameter T1ms = 5;//仿真时的参数
```

```
initial begin
```

```
    clk = 1'b1;
```

```
    rst_n = 1'b0;
```

```
    seg7_data = 24'h123456;//显示数据为 123456
```

```
    # 200.1
```

```
    rst_n = 1'b1;
```

```
    # 2000
```

```
    $stop;
```

```
end
```

```
always # 10 clk = ~clk;//50MHz
```

```
seg7
```

```
    # (
```

```
        .T1ms(T1ms)//传递参数
```

```
    )
```

```
    seg7_dut (
```

```
. clk(clk),  
. rst_n(rst_n),  
. seg7_data(seg7_data),  
. seg(seg),  
. sel(sel)  
);
```

endmodule

仿真波形如下：



通过仿真，我们可以清楚的看到，当sel为“0”时，seg为数字“1”的段码；当sel为“1”时……，证明设计正确。

如果设计要求或者本地晶振与笔者的设计不同，请自行更改设计，以保证设计的正确性。如果还是有不明白的读者可以发邮件到我邮箱



至芯科技
ZHI XIN TECHNOLOGY

FPGA 培训专家 www.zxopen.com

或者加群询问。

制作人: 奋斗的小孩

fpga 交流群: 282124839

至芯科技论坛 www.fpgaw.com

至芯科技论坛 www.fpgaw.com