

## 数码管上流动的 HS

该电路使用到数码管和时钟电路，有关电路原理参考“3.1.6 七段数  
这一节我们使用动态扫描的方式驱动数码管

**功能描述：**通过 CPLD 用动态扫描驱动的方式来控制 4 个数码管，让他们显示从右向左走动的 HS(代表 HuSoon)

**源程序：**(GUIDE 光盘/samples/qt42/WALKSMG/WALKSMG.V)

/数码管上流动显示 HS，WALKSMG.v

//DOWNLOAD FROM WWW.HUSOON.COM

```
module WALKSMG(SMG, W, CLK);           //定义模块结构
    output [8:1] SMG;                   //定义数码管段输出引脚,SMG=DP G F E D C B A
    output [4:1] W;                     //定义数码管选择输出引脚 ,W=W4 W3 W2 W1
    input CLK;                           //定义输入时钟引脚,CLK=11.0592MHz OR
    CLK=2048Hz
```

```
    reg [8:1] SMG_REG;                  //定义数码管段输出寄存器
    reg [4:1] W_REG;                    //定义数码管选择输出寄存器
    reg [8:1] DISP_DAT;                 //定义显示数据代号寄存器
    reg [36:0] COUNT;                  //定义计数器寄存器
```

```
always @(posedge CLK)                  //定义 CLK 信号下降沿触发
begin
    COUNT=COUNT+1;                    //计数器值加 1
end
```

```
always @(COUNT[14:13])                //定义显示数据触发事件
begin
    case (COUNT[14:13])                //选择扫描显示数据
        3'h3: DISP_DAT = 8'h3;          //显示 SMG4 数值
        3'h2: DISP_DAT = 8'h2;          //显示 SMG3 数值
        3'h1: DISP_DAT = 8'h1;          //显示 SMG2 数值
        3'h0: DISP_DAT = 8'h0;          //显示 SMG1 数值
    endcase
```

```
    case (COUNT[14:13])                //选择哪个数码管
        3'h3: W_REG = 8'b0111;          //选择数码管 SMG4
        3'h2: W_REG = 8'b1011;          //选择数码管 SMG3
        3'h1: W_REG = 8'b1101;          //选择数码管 SMG2
        3'h0: W_REG = 8'b1110;          //选择数码管 SMG1
```

```

        endcase
    end

    always @(COUNT[25:23])          //显示译码输出
    begin
        if (COUNT[25:23]==0) SMG_REG=8'B00000000;
        else if (COUNT[25:23]==1)
        begin
            case (DISP_DAT)          //选择输出数据
                8'h0: SMG_REG = 8'b01110110;    //显示 H
            default: SMG_REG = 8'b00000000;    //关掉所有数码管
            endcase
        end
        else if (COUNT[25:23]==2)
        begin
            case (DISP_DAT)          //选择输出数据
                8'h1: SMG_REG = 8'b01110110;    //显示 H
                8'h0: SMG_REG = 8'b01101101;    //显示 S
            default: SMG_REG = 8'b00000000;    //关掉所有数码管
            endcase
        end

        else if(COUNT[25:23]==3)
        begin
            case (DISP_DAT)          //选择输出数据
                8'h2: SMG_REG = 8'b01110110;    //显示 H
                8'h1: SMG_REG = 8'b01101101;    //显示 S
            default: SMG_REG = 8'b00000000;    //关掉所有数码管
            endcase
        end

        else if(COUNT[25:23]==4)
        begin
            case (DISP_DAT)          //选择输出数据
                8'h3: SMG_REG = 8'b01110110;    //显示 H
                8'h2: SMG_REG = 8'b01101101;    //显示 S
            default: SMG_REG = 8'b00000000;    //关掉所有数码管
            endcase
        end

        else if (COUNT[25:23]==5)
        begin
            case (DISP_DAT)          //选择输出数据
                8'h3: SMG_REG = 8'b01101101;    //显示 S
            endcase
        end
    end

```

```

default: SMG_REG = 8'b00000000;    //关掉所有数码管
endcase
end
else
    SMG_REG = 8'b00000000;    //关掉所有数码管
end

assign SMG=SMG_REG;            //输出数码管译码结果
assign W=W_REG;                //输出数码管选择

endmodule

```

**操作：**在 QUARTUS 中建立工程，并用上面的语句建立 verilog-HDL 文件，保存、编译，连后选定芯片 EPM7128SLC84-15,并按下表指定管脚：

|    | To                                                                                         | Location | General Function ▲ |
|----|--------------------------------------------------------------------------------------------|----------|--------------------|
| 1  |  CLK    | PIN_83   | Global Clock       |
| 2  |  SMG[1] | PIN_6    | I/O                |
| 3  |  SMG[2] | PIN_79   | I/O                |
| 4  |  SMG[3] | PIN_81   | I/O                |
| 5  |  SMG[4] | PIN_8    | I/O                |
| 6  |  SMG[5] | PIN_9    | I/O                |
| 7  |  SMG[6] | PIN_4    | I/O                |
| 8  |  SMG[7] | PIN_80   | I/O                |
| 9  |  SMG[8] | PIN_5    | I/O                |
| 10 |  W[1]   | PIN_76   | I/O                |
| 11 |  W[2]   | PIN_77   | I/O                |
| 12 |  W[3]   | PIN_10   | I/O                |
| 13 |  W[4]   | PIN_11   | I/O                |
| 14 | <<new>>                                                                                    | <<new>>  |                    |

再编译、仿真、下载，并把排针 JP1 对应脚用跳冒插上，J2 上把跳冒插到 11.0592MHz 上，你将看到实验结果

**注意：**

1. 编译时出现警告一些警告，他们不影响电路，因为设计就是的思  
路就是这样

2. 这里 HS 都采用近似的形状，具体见静态数码管控制部分的注释部分