

数电实验EXP5（扩展见另一份报告）

刘永鹏 191220070 计科 693901492@qq.com 2020.9.29

1. 实验目的

请在 DE10-Standard 开发板上实现一个计时器，在七段数码管上直接以十进制显示。

利用开发板上的频率为 50MHz 的时钟，请先设计一个分频器，其输入为 50MHz 的时钟，输出为一个频率为 1Hz，周期为 1 秒的时钟信号。再用这个新的频率为 1Hz 的时钟信号作为你设计的时钟信号，进行计数。

要求此计时器有开始、暂停和清零功能，要求从 00 计数到 99，计数值到 99 后重新从零开始计数。在数码管上用两位数字显示。

可以在计时结束的时候让某一个发光二极管闪烁，提示计时结束，类似数电教材 8.4.3 节 74x163 计数器中的 RCO 信号高电平一个周期。

2. 实验原理

串行的n个触发器既可以构成简单的n位计数器，同时，也可以利用移位寄存器制作计数器。

在Quartus中，实现计数器相对比较简单，本次实验的清零端位**同步清零**。

关于分频器，已知开发板的时钟为**50MHz**，于是只需要用count_clk == 25000000做判断，每数25000000，程序的时钟翻转一次，即可以构建**1s的时钟周期**。

使用function模块单独编写时钟信号->七段数码管的转换，程序逻辑更加清晰。

3. 实验环境

1. Quartus环境，手动建立项目（没有使用systembuilder）。

```
input clk,
input s_p, // 开始或暂停,sw[1]
input clr, //sw[0]
output reg rco,
output reg [3:0] units,
output reg [3:0] tens,
output reg [6:0] Hex0,
output reg [6:0] Hex1
```

4. 程序代码/流程图

首先，分频器代码如下(原理在上面给出)：

```
1 always @ (posedge clk)
2 begin
3     if(clk_onesec == 25000000)
4     begin
5         clk_t <= ~clk_t;
6         clk_onesec <= 0;
7     end
8     else clk_onesec <= clk_onesec + 1;
9 end
```

使用units/tens分别记录十位和个位，用clr/s_p表示清零端和暂停，使用同步清零。

具体代码如下。

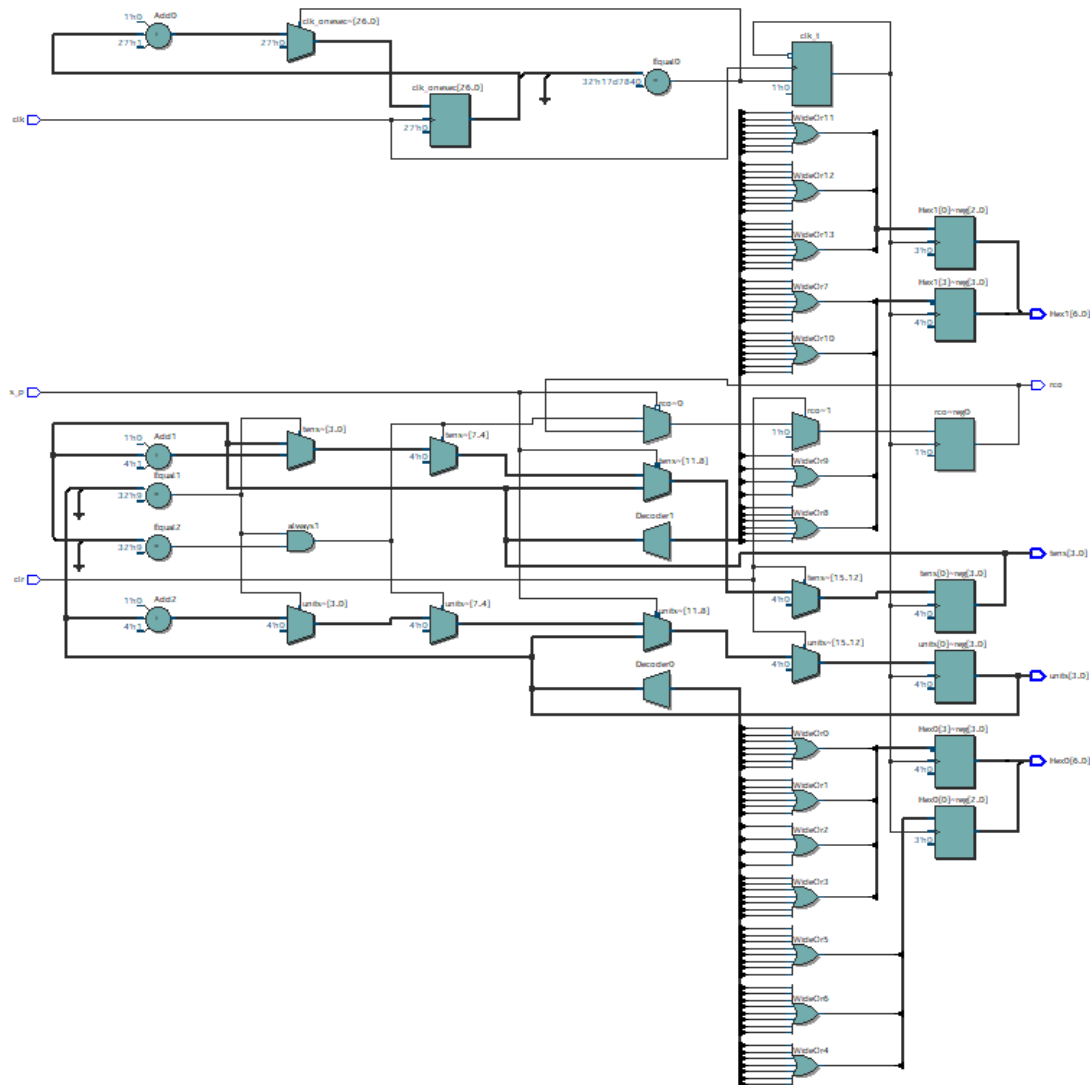
```
1  always @ (posedge clk_t)
2  begin
3      if(clr) // 清零状态
4      begin
5          rco <= 0;
6          units <= 0;
7          tens <= 0;
8      end
9      else if(!s_p) //暂停状态
10     begin
11         rco <= rco;
12         units <= units;
13         tens <= tens;
14     end
15     else begin //正常计数
16         //判断特殊值99
17         if(units == 9 && tens == 9)
18         begin
19             units <= 0;
20             tens <= 0;
21             rco <= 1; //进位，相当于计数器的rco输出端，高电平有效
22         end
23         //各位进1时
24         else if(units == 9)
25         begin
26             rco <= 0;
27             tens <= tens + 1;
28             units <= 0;
29         end
30         else begin
31             rco <= 0;
32             units <= units + 1;
33             tens <= tens;
34         end
35     end
36     {Hex1,Hex0} = to_hex(tens, units);
37 end
```

testbench中生成一段时钟周期：

```
1  repeat(100000000)
2  begin
3      clk = ~clk;
4      #1;
5  end
```

其实在测试中，我采用的是暂时使分频器失效，而直接对clk_t程序时钟进行赋值，这样便于在仿真时查看。

RTL门电路视图：



5. 实验过程

1. 做出初步设计。
2. Quartus编写代码。
3. 编写estbench，进行仿真测试。
4. 烧板测试。

6. 实验结果

已完成验收。

7. 问题与解决方案

1. 使用分频器后不便于仿真（时钟周期过长），因此在加入分配器之前先进行一次仿真测试，确认代码无误。
2. 掌握function语句的用法，使得代码编写起来更加整洁。
3. 开发板再再次出现了错误！（到目前为止，我们的开发板已经发现有**两个坏的按钮KEY1,KEY2**，以及本次发现的坏掉的**时钟clock1**）不过很快在修改引脚后烧板验证成功。

8. 启示/意见与建议

暂无。