FPGA 实验报告

至芯科技就业班

2016.03.22

马博

实验五　四块模块构建跑马灯

一、 实验预习：

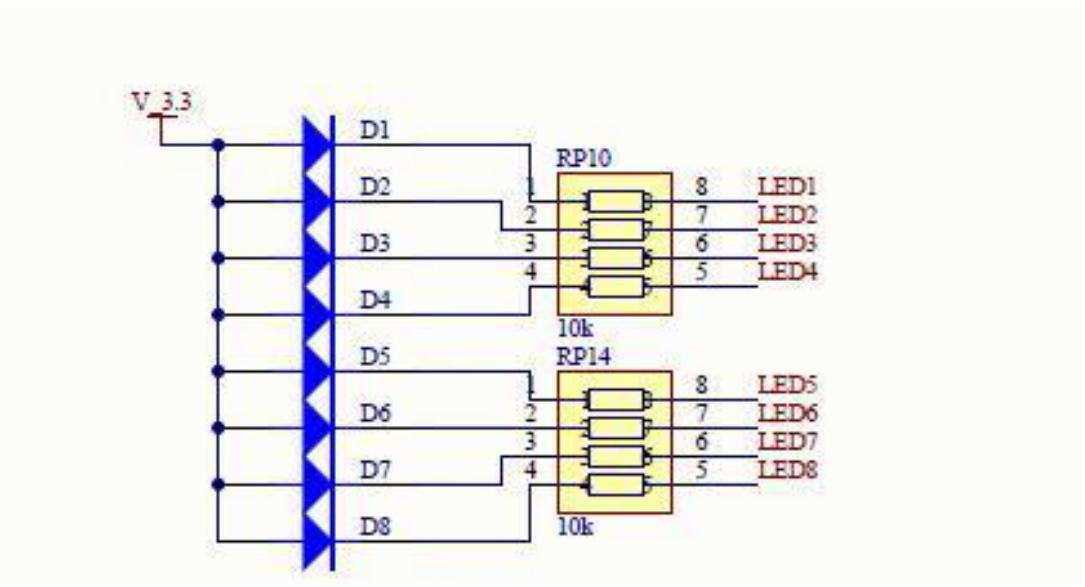
首先 **LED** 是英文 light emitting diode（发光二极管）的缩写，它是由一块电致发光的半导体材料，放在一个有引线的架子上，然后四周用环氧树脂密封。因为 LED 外部用环氧树脂密封，所以 **LED** 的抗震性能好。

二、 实验目的：

四块模块构建跑马灯。

三、 LED 原理：

LED 最基本的构成是二极管。二极管的工作原理是正向导通反向截止。当它的外加正向电压小于 **0.5**V 时，二极管处于截至状态;当外加正向电压大于 **0.7V** 时，处于饱和/导通状态。而 **LED** 的原理是在导通情况下发光，也就是在 **LED** 两边一边加低电压（低电平），另外一边加高电压（高电平）时，**LED** 发光。

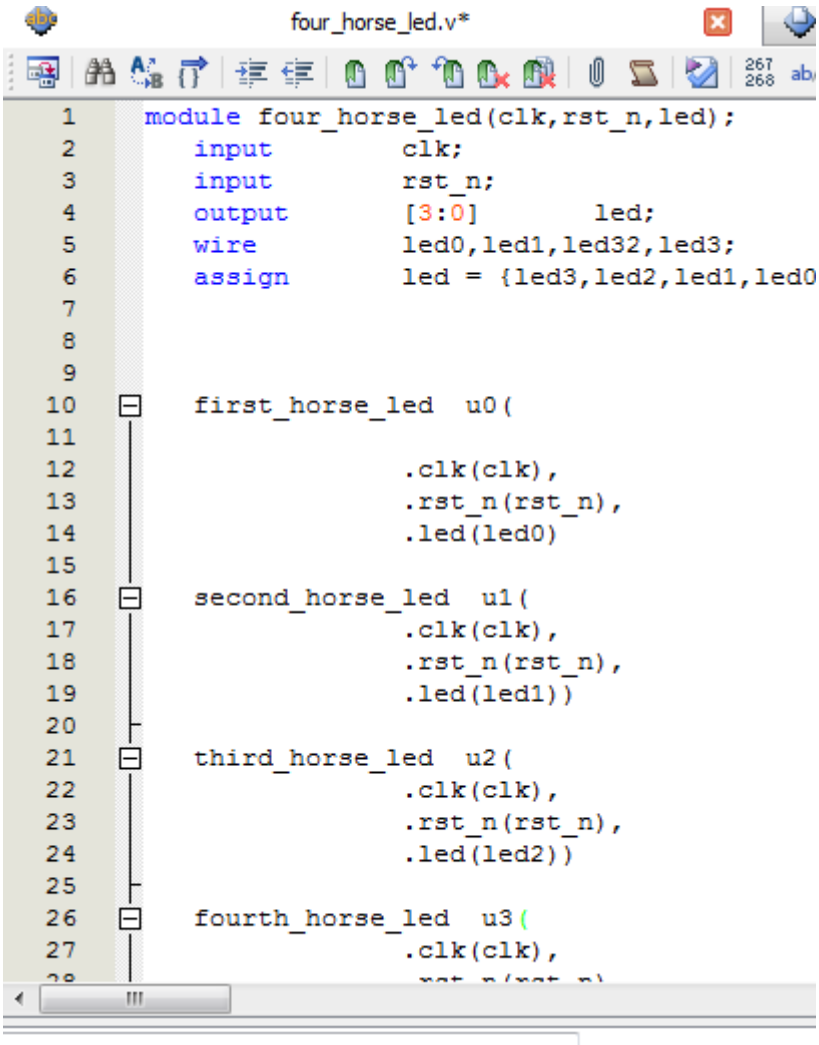


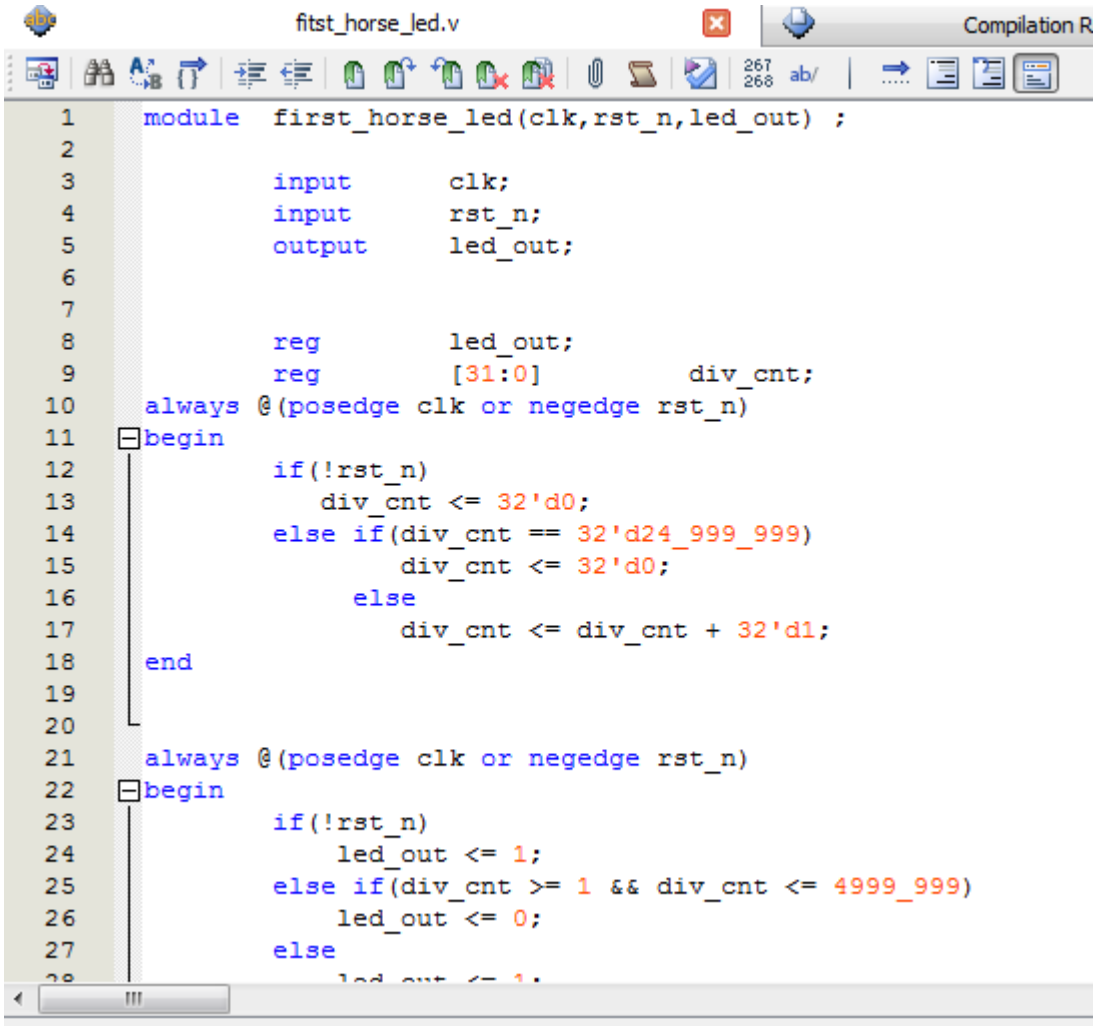
四、实验原理：

从上面的原理图可看出，**LED** 左边为高电平，当 **LED** 右边为低电平时，二极管导通，则该 **LED** 被点亮。

五、编写代码：

1 编写代码：


four_horse_led.v*
1 module four_horse_led(clk,rst_n,led);
2 input clk;
3 input rst_n;
4 output [3:0] led;
5 wire led0,led1,led32,led3;
6 assign led = {led3,led2,led1,led0
7
8
9
10 first_horse_led u0(
11
12 .clk(clk),
13 .rst_n(rst_n),
14 .led(led0)
15
16 second_horse_led u1(
17 .clk(clk),
18 .rst_n(rst_n),
19 .led(led1))
20
21 third_horse_led u2(
22 .clk(clk),
23 .rst_n(rst_n),
24 .led(led2))
25
26 fourth_horse_led u3(
27 .clk(clk),




```
1  module  first_horse_led(clk,rst_n,led_out) ;
2
3          input       clk;
4          input       rst_n;
5          output      led_out;
6
7
8          reg         led_out;
9          reg         [31:0]          div_cnt;
10 always @(posedge clk or negedge rst_n)
11 begin
12         if(!rst_n)
13            div_cnt <= 32'd0;
14         else if(div_cnt == 32'd24_999_999)
15                 div_cnt <= 32'd0;
16              else
17                 div_cnt <= div_cnt + 32'd1;
18 end
19
20
21 always @(posedge clk or negedge rst_n)
22 begin
23         if(!rst_n)
24             led_out <= 1;
25         else if(div_cnt >= 1 && div_cnt <= 4999_999)
26             led_out <= 0;
27         else
```

second_horse_led.v*

Compilation Report - fitst_horse_led

```
1  module  second_horse_led(clk,rst_n,led_out) ;
2
3          input       clk;
4          input       rst_n;
5          output      led_out;
6
7
8          reg         led_out;
9          reg         [31:0]          div_cnt;
10 always @(posedge clk or negedge rst_n)
11 begin
12         if(!rst_n)
13            div_cnt <= 32'd0;
14         else if(div_cnt == 32'd24_999_999)
15                 div_cnt <= 32'd0;
16              else
17                 div_cnt <= div_cnt + 32'd1;
18 end
19
20
21 always @(posedge clk or negedge rst_n)
22 begin
23         if(!rst_n)
24             led_out <= 1;
25         else if(div_cnt >= 5_000_000 && div_cnt <= 9_999_999)
26             led_out <= 0;
27         else
```

third_horse_led.v*

Compilation Report - fits

```
1  module  third_horse_led(clk,rst_n,led_out) ;
2
3          input       clk;
4          input       rst_n;
5          output      led_out;
6
7
8          reg         led_out;
9          reg         [31:0]          div_cnt;
10 always @(posedge clk or negedge rst_n)
11 begin
12         if(!rst_n)
13            div_cnt <= 32'd0;
14         else if(div_cnt == 32'd24_999_999)
15                 div_cnt <= 32'd0;
16              else
17                 div_cnt <= div_cnt + 32'd1;
18 end
19
20
21 always @(posedge clk or negedge rst_n)
22 begin
23         if(!rst_n)
24             led_out <= 1;
25         else if(div_cnt >= 10_000_000 && div_cnt <= 14_999_999)
26             led_out <= 0;
27         else
```

fourth_horse_led.v | Compilation Report - fitst_horse_led

```
1   module  fourth_horse_led(clk,rst_n,led_out) ;
2
3           input       clk;
4           input       rst_n;
5           output      led_out;
6
7
8           reg         led_out;
9           reg         [31:0]          div_cnt;
10  always @(posedge clk or negedge rst_n)
11  begin
12          if(!rst_n)
13             div_cnt <= 32'd0;
14          else if(div_cnt == 32'd24_999_999)
15                  div_cnt <= 32'd0;
16               else
17                  div_cnt <= div_cnt + 32'd1;
18  end
19
20
21  always @(posedge clk or negedge rst_n)
22  begin
23          if(!rst_n)
24              led_out <= 1;
25          else if(div_cnt >= 15_000_000 && div_cnt <= 19_999_999)
26              led_out <= 0;
27          else
```

2、管脚分配：

Named: * Edit:

Node Name	Direction	Location	I/O Bank	VRE
clk	Input	PIN_E1	1	B1_N0
led_out[3]	Output	PIN_M10	4	B4_N0
led_out[2]	Output	PIN_M8	3	B3_N0
led_out[1]	Output	PIN_P8	3	B3_N0
led_out[0]	Output	PIN_T12	4	B4_N0
rst_n	Input	PIN_K1	2	B2_N0
<<new node>>				

六、实验心得：

在此次实验中，通过运用四个模块的插入，构建了跑马灯。这次实验通过复杂问题简单化。这次实验不仅对我在前几天学习的加强，也给我开阔了新的思路，通过构建多的模块来完成所需的目标，让我对 led 的构造已经完全了解，希望我在接下来的实验里再接再厉。