2022《FPGA应用实验》实验报告

实验编号： lab 1 实验时间： 2022.03.12

实验名称： 利用 8个发光二极管（LED）形成流水灯显示

班级： F1903604 学号： 519021910917 姓名： 费扬

1、实验平台

采用Xilinx 公司的FPGA集成开发环境 Xilinx ISE Design Suite 10.1 sp3，实验开发板为Xilinx Spartan-3E FPGA Starter Kit。

2、实验设计要求：

在 Spartan‐3E FPGA Starter Kit Board 上有 8 个发光二极管（LED7 ～ LED0）。

使用开发板的全局时钟信号 CLK\_50MHz，管脚为 P = C9。产生 1 Hz 的秒脉冲，每秒钟点亮一个 LED。

使用滑动开关SW0，管脚P=L13，作为复位键。

复位后，8 个 LED 都为关闭状态（缺省值为：LEDOut = 8’b0000\_0000）；

即：

（0）LEDOut = 8’b0000\_0000;

（1）LED Out = 8’b0000\_0001;

（2）LED Out = 8’b0000\_0011;

（3）LED Out = 8’b0000\_0111;

（4）LED Out = 8’b0000\_1111;

（5）LED Out = 8’b0001\_1111;

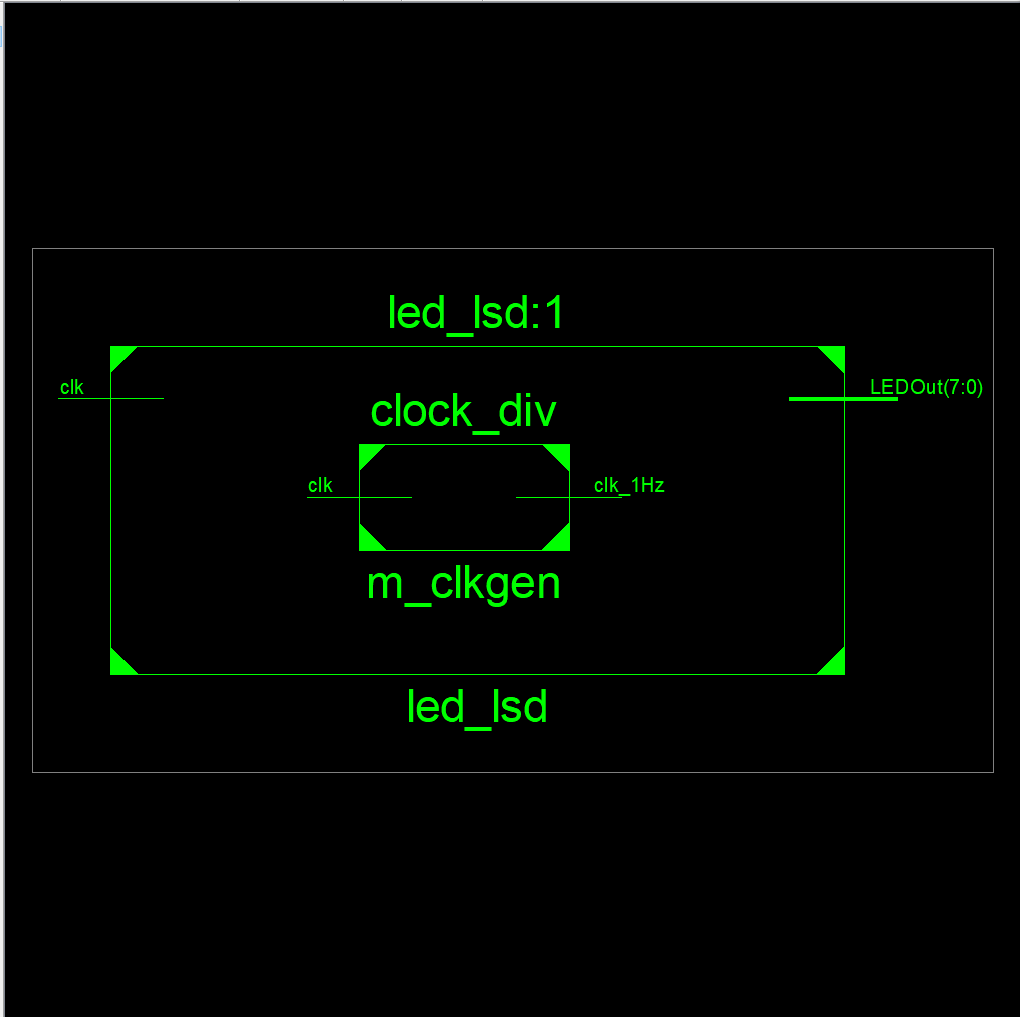
（6）LED Out = 8’b0011\_1111;

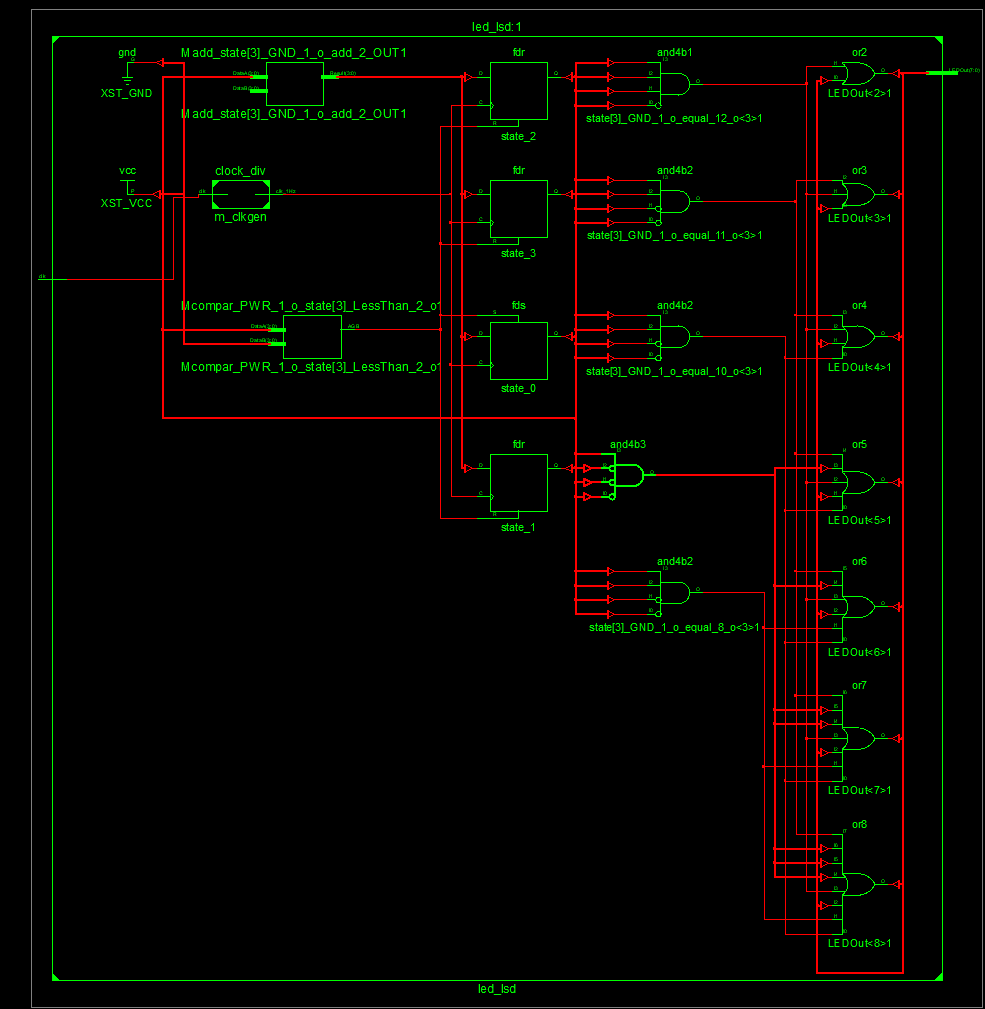
（7）LED Out = 8’b0111\_1111;

（8）LED Out = 8’b1111\_1111;

（9）不断重复（0）～（8）

3、模块设计框图

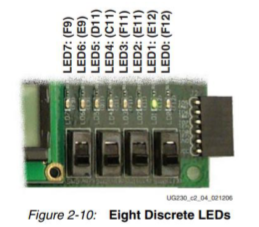




4、实验原理：

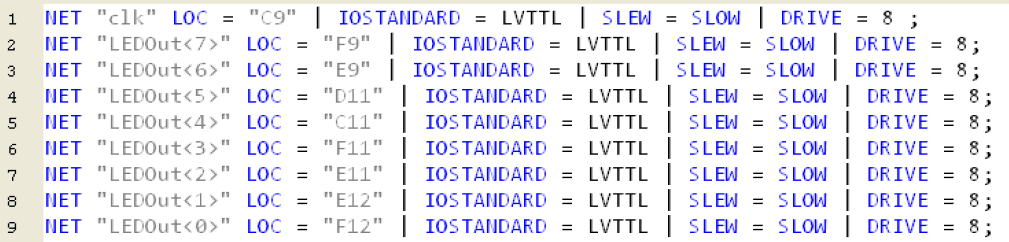
1. LED原理

Spartan-3e FPGA Starter Kit 板有 8 个独立的表面安装 led 位于滑动开关上方，如下图所示。led 标记为LED0 到 LED7。LED7 是最左边的 LED, LED0 是最右边的LED。



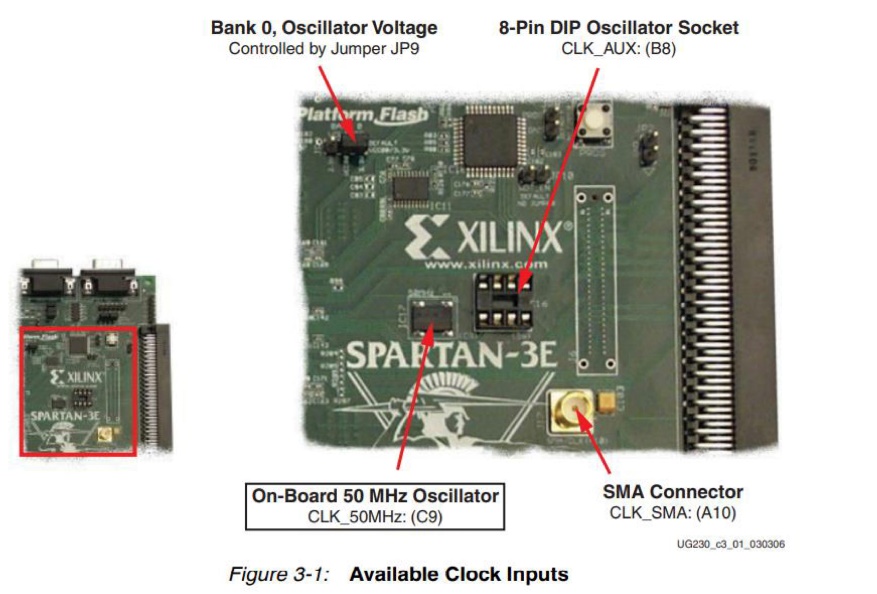
操作：每个LED的一端连接到地，另一端通过 390 欧限流电阻连接到Spartan-3e 设备上的引脚。为了点亮单个LED，需驱动相关的FPGA控制信号为高。

UCF 位置约束：下图提供了四个按钮开关的 UCF 约束，包括 I/O 引脚分配，使用的I/O 标准，输出回转率和输出驱动电流。



2. 50MHz时钟

Spartan-3E FPGA Starter Kit 单板支持三个主时钟输入源，它们都位于Xilinx 标识的下方，靠近 spartan-3e 标识。该板包括一个板载 50 MHz 时钟振荡器。时钟可以通过 SMA风格的连接器提供。FPGA也可以通过 SMA风格的连接器产生时钟信号或其他高速信号。可选安装一个单独的 8 针DIP 风格时钟振荡器在提供的插座。



5、Verilog 模块设计

**//顶层模块**

**//led\_lsd.v**

module led\_lsd(  
    output [7:0] LEDOut,  
    input clk  
    );  
 reg [7:0] led\_connect;  
 reg [3:0] state;  
 wire clk\_1;  
   
 clock\_div m\_clkgen(.clk\_1Hz(clk\_1),.clk(clk));  
   
 always @(posedge clk\_1) begin  
if (state > 8) state <= 1; //下一个状态为1，1号灯亮  
else state <= state + 1;  
 end  
   
 always@(\*)  
case (state)  
1: led\_connect<=8’b0000\_0001;  
2: led\_connect<=8’b0000\_0011;  
3: led\_connect<=8’b0000\_0111;  
4: led\_connect<=8’b0000\_1111;  
5: led\_connect<=8’b0001\_1111;  
6: led\_connect<=8’b0011\_1111;  
7: led\_connect<=8’b0111\_1111;  
8: led\_connect<=8'b1111\_1111;  
default: led\_connect<=8’b0000\_0000;  
endcase  
  
assign LEDOut = led\_connect;  
  
endmodule

**//时钟分频**

**//clock\_div.v**

module clock\_div( output reg clk\_1Hz,     
                      input  clk                    
                    );  
parameter PULSESCOUNT = 18’h3\_D090, // = 11\_1101\_0000\_1001\_0000(bin) (16bits)   
                  RESETZERO = 18’h0;  
  
    reg [17:0] counter;       
                                 
  
 always @(posedge clk) begin          
  if ( counter < PULSESCOUNT )      
   counter <= counter + 1’b1;  
  else begin  
   clk\_1Hz <= ~clk\_1Hz;  
   counter <= RESETZERO;  
  end  
 end  
endmodule

**//引脚约束**

**//led\_lsd.ucf**

NET "clock" LOC="C9" | IOSTANDARD = LVCMOS33;  
NET "LEDOut<7>" LOC = "F9" | IOSTANDARD = LVTTL | SLEW = SLOW | DRIVE = 8;  
NET "LEDOut<6>" LOC = "E9" | IOSTANDARD = LVTTL | SLEW = SLOW | DRIVE = 8;  
NET "LEDOut<5>" LOC = "D11" | IOSTANDARD = LVTTL | SLEW = SLOW | DRIVE = 8;  
NET "LEDOut<4>" LOC = "C11" | IOSTANDARD = LVTTL | SLEW = SLOW | DRIVE = 8;  
NET "LEDOut<3>" LOC = "F11" | IOSTANDARD = LVTTL | SLEW = SLOW | DRIVE = 8;  
NET "LEDOut<2>" LOC = "E11" | IOSTANDARD = LVTTL | SLEW = SLOW | DRIVE = 8;  
NET "LEDOut<1>" LOC = "E12" | IOSTANDARD = LVTTL | SLEW = SLOW | DRIVE = 8;  
NET "LEDOut<0>" LOC = "F12" | IOSTANDARD = LVTTL | SLEW = SLOW | DRIVE = 8;

**//test\_bench**

**//led\_lsd\_tb.v**

`timescale 1ns / 1ps  
module led\_lsd\_tb;  
  
// Inputs  
reg clk;  
  
// Outputs  
wire [7:0] p\_led;  
  
// Instantiate the Unit Under Test (UUT)  
led\_lsd uut(  
.LEDOut(p\_led),   
.clk(clk)  
);  
  
initial begin  
// Initialize Inputs  
clk = 0;  
forever #10 clk=~clk;  
end  
  
initial  
#3000 $stop;  
  
initial  
$monitor("time=%d,ledout=%b",$time,p\_led);  
        
endmodule

6、试验仿真结果和分析

如图：仿真中，LED正确显示流水灯：

