

# 实验回顾

- Verilog代码与XDC约束文件的对应关系
- 名称——对应
- 每个信号分配一个IO管脚

```
module test(  
    input                clk,  
    input                rst,  
    input                [7:0] SW,  
    output reg           [7:0] led);  
always@(posedge clk or posedge rst)  
begin  
    if(rst)  
        led <= 8'haa;  
    else  
        led <= {sw[0],sw[1],sw[2],sw[3],sw[4],sw[5],sw[6],sw[7]};  
    end  
endmodule
```

```
6 ## Clock signal  
7 set_property -dict { PACKAGE_PIN E3          IOSTANDARD LVCMOS33 } [get_ports { clk }]; #IO_L12P_T1_I  
8 #create_clock -add -name sys_clk_pin -period 10.00 -waveform {0 5} [get_ports {CLK100MHZ}];  
9 ## FPGAOL BUTTON & SOFT_CLOCK  
10 set_property -dict { PACKAGE_PIN B18        IOSTANDARD LVCMOS33 } [get_ports { rst }];  
11 ## FPGAOL LED (signle-digit-SEGPLAY)  
12 set_property -dict { PACKAGE_PIN C17        IOSTANDARD LVCMOS33 } [get_ports { led[0] }];  
13 set_property -dict { PACKAGE_PIN D18        IOSTANDARD LVCMOS33 } [get_ports { led[1] }];  
14 set_property -dict { PACKAGE_PIN E18        IOSTANDARD LVCMOS33 } [get_ports { led[2] }];  
15 set_property -dict { PACKAGE_PIN G17        IOSTANDARD LVCMOS33 } [get_ports { led[3] }];  
16 set_property -dict { PACKAGE_PIN D17        IOSTANDARD LVCMOS33 } [get_ports { led[4] }];  
17 set_property -dict { PACKAGE_PIN E17        IOSTANDARD LVCMOS33 } [get_ports { led[5] }];  
18 set_property -dict { PACKAGE_PIN F18        IOSTANDARD LVCMOS33 } [get_ports { led[6] }];  
19 set_property -dict { PACKAGE_PIN G18        IOSTANDARD LVCMOS33 } [get_ports { led[7] }];  
20  
21 ## FPGAOL SWITCH  
22 set_property -dict { PACKAGE_PIN D14        IOSTANDARD LVCMOS33 } [get_ports { sw[0] }];  
23 set_property -dict { PACKAGE_PIN F16        IOSTANDARD LVCMOS33 } [get_ports { sw[1] }];  
24 set_property -dict { PACKAGE_PIN G16        IOSTANDARD LVCMOS33 } [get_ports { sw[2] }];  
25 set_property -dict { PACKAGE_PIN H14        IOSTANDARD LVCMOS33 } [get_ports { sw[3] }];  
26 set_property -dict { PACKAGE_PIN E16        IOSTANDARD LVCMOS33 } [get_ports { sw[4] }];  
27 set_property -dict { PACKAGE_PIN F13        IOSTANDARD LVCMOS33 } [get_ports { sw[5] }];  
28 set_property -dict { PACKAGE_PIN G13        IOSTANDARD LVCMOS33 } [get_ports { sw[6] }];  
29 set_property -dict { PACKAGE_PIN H16        IOSTANDARD LVCMOS33 } [get_ports { sw[7] }];  
30  
31  
32 ## FPGAOL HEXPLAY  
33  
34 #set_property -dict { PACKAGE_PIN A14        IOSTANDARD LVCMOS33 } [get_ports { hexplay_data[0] }];  
35 #set_property -dict { PACKAGE_PIN A13        IOSTANDARD LVCMOS33 } [get_ports { hexplay_data[1] }];  
36 #set_property -dict { PACKAGE_PIN A16        IOSTANDARD LVCMOS33 } [get_ports { hexplay_data[2] }];  
37 #set_property -dict { PACKAGE_PIN A15        IOSTANDARD LVCMOS33 } [get_ports { hexplay_data[3] }];  
38 #set_property -dict { PACKAGE_PIN B17        IOSTANDARD LVCMOS33 } [get_ports { hexplay_an[0] }];  
39 #set_property -dict { PACKAGE_PIN B16        IOSTANDARD LVCMOS33 } [get_ports { hexplay_an[1] }];  
40 #set_property -dict { PACKAGE_PIN A18        IOSTANDARD LVCMOS33 } [get_ports { hexplay_an[2] }];
```



**中国科学技术大学**  
University of Science and Technology of China

# **实验07**

## **FPGA实验平台及IP核使用**

**卢建良**

**计算机科学与技术学院**

**11/27/2020**

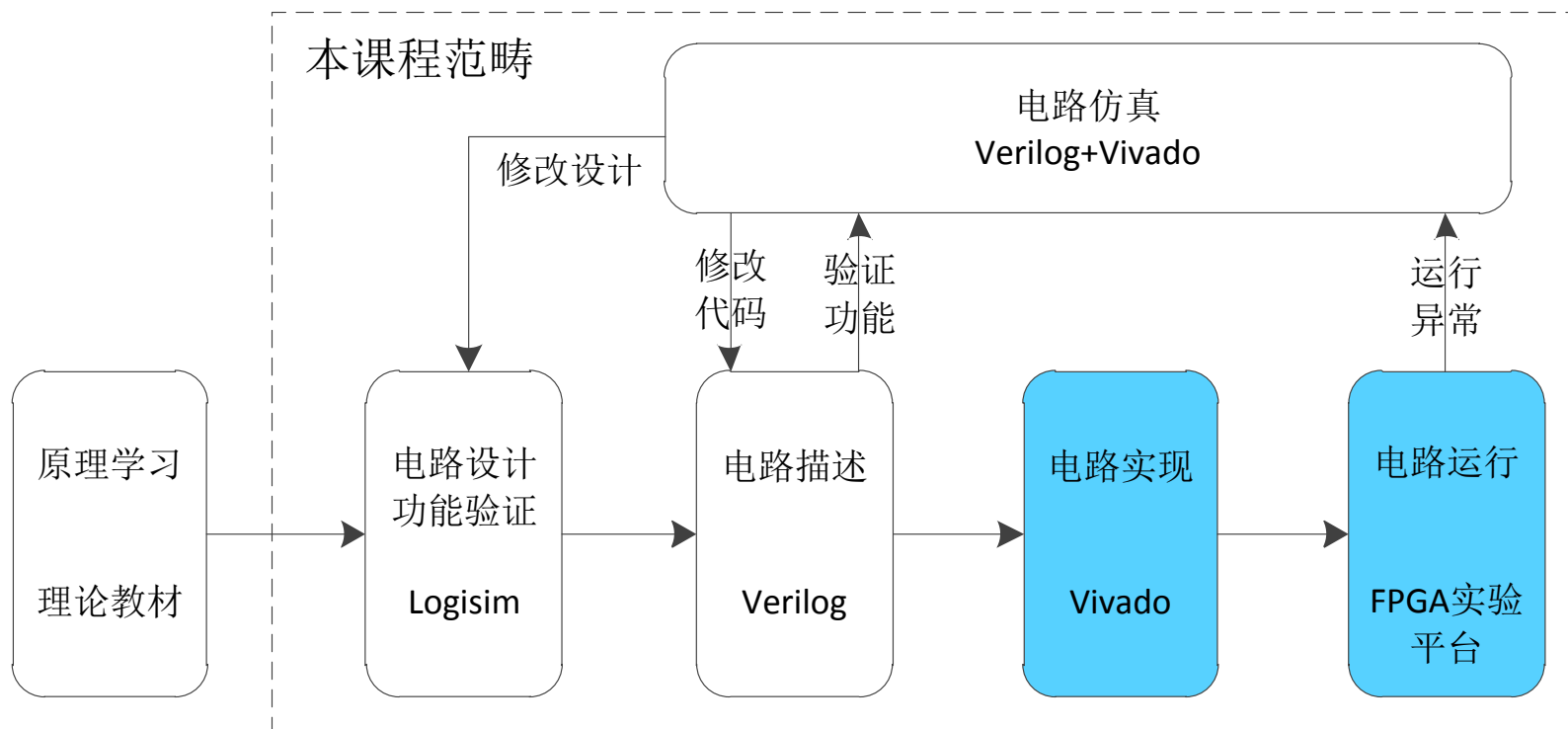
# 实验安排

数字电路实验关键时间节点

时间	实验发布（23：59前）	实验讲解（上课时间）	检查截止（21:30前）	报告提交截止（23:59前）
第四周(10.09)	实验01_Logisim入门			
第五周(10.16)	实验02_简单组合逻辑电路	实验01_Logisim入门		
第六周(10.23)	实验03_简单时序逻辑电路	实验02_简单组合逻辑电路	实验01_Logisim入门	
第七周(10.30)	实验04_Verilog硬件描述语言	实验03_简单时序逻辑电路	实验02_简单组合逻辑电路	实验01_Logisim入门
第八周(11.06)	实验05_使用Vivado进行仿真	实验04_Verilog硬件描述语言	实验03_简单时序逻辑电路	实验02_简单组合逻辑电路
第九周(11.13)	实验06_FPGA原理及Vivado综合	实验05_使用Vivado进行仿真	实验04_Verilog硬件描述语言	实验03_简单时序逻辑电路
第十周(11.20)	实验07_FPGA实验平台及IP核	实验06_FPGA原理及Vivado综合	实验05_使用Vivado进行仿真	实验04_Verilog硬件描述语言
十一周(11.27)	实验08_信号处理及有限状态机	实验07_FPGA实验平台及IP核	实验06_FPGA原理及Vivado综合	实验05_使用Vivado进行仿真
十二周(12.04)	实验09_竞争冒险及流水线技术	实验08_信号处理及有限状态机	实验07_FPGA实验平台及IP核	实验06_FPGA原理及Vivado综合
十三周(12.11)	实验10_综合实验	实验09_竞争冒险及流水线技术	实验08_信号处理及有限状态机	实验07_FPGA实验平台及IP核
十四周(12.18)		实验10_综合实验	实验09_竞争冒险及流水线技术	实验08_信号处理及有限状态机
十五周(12.25)			实验10_综合实验	实验09_竞争冒险及流水线技术
十六周(01.01)				实验10_综合实验

# 实验目的

- 熟悉FPGAOL在线实验平台结构及使用
- 掌握FPGA开发各关键环节
- 学会使用IP核（知识产权核）



# 实验环境

---

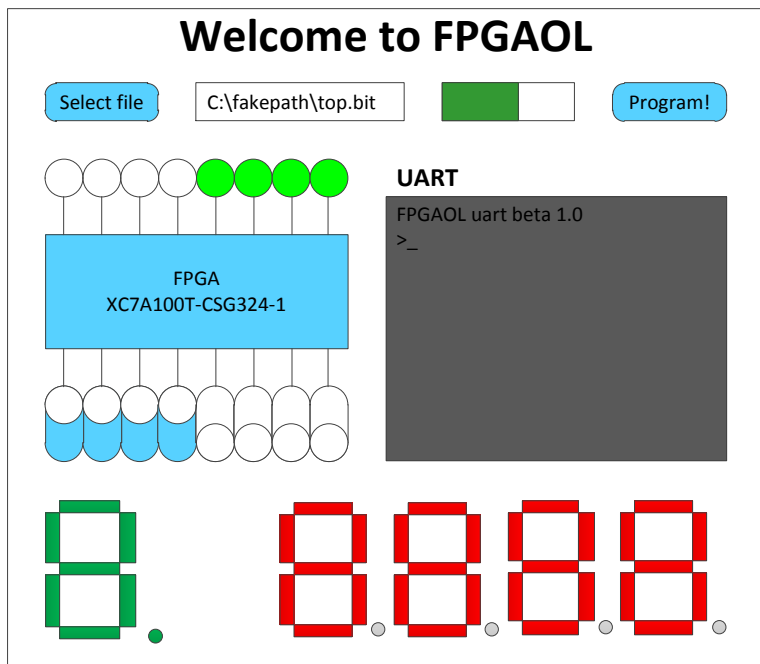
- VLAB平台 : [vlab.ustc.edu.cn](http://vlab.ustc.edu.cn)
- FPGAOL平台 : [fpgaol.ustc.edu.cn](http://fpgaol.ustc.edu.cn)
- Vivado
- Logisim

# 实验原理

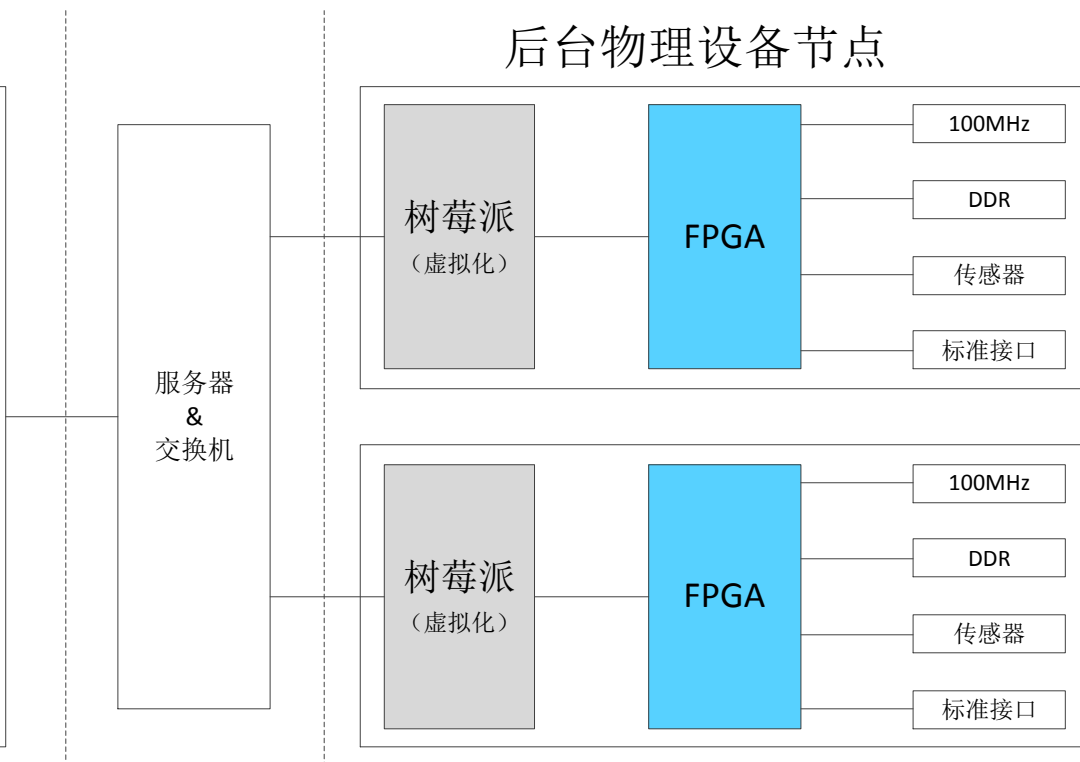
## ■ FPGAOL实验平台

■FPGA online ( 简称FPGAOL ) 是中国科学技术大学计算机学院实验中心自主开发的一套在线实验平台，该平台通过分时复用的方式为用户提供FPGA设备节点的在线服务，用户可通过网址[fpgaol.ustc.edu.cn](http://fpgaol.ustc.edu.cn)访问平台，

网页端虚拟化界面



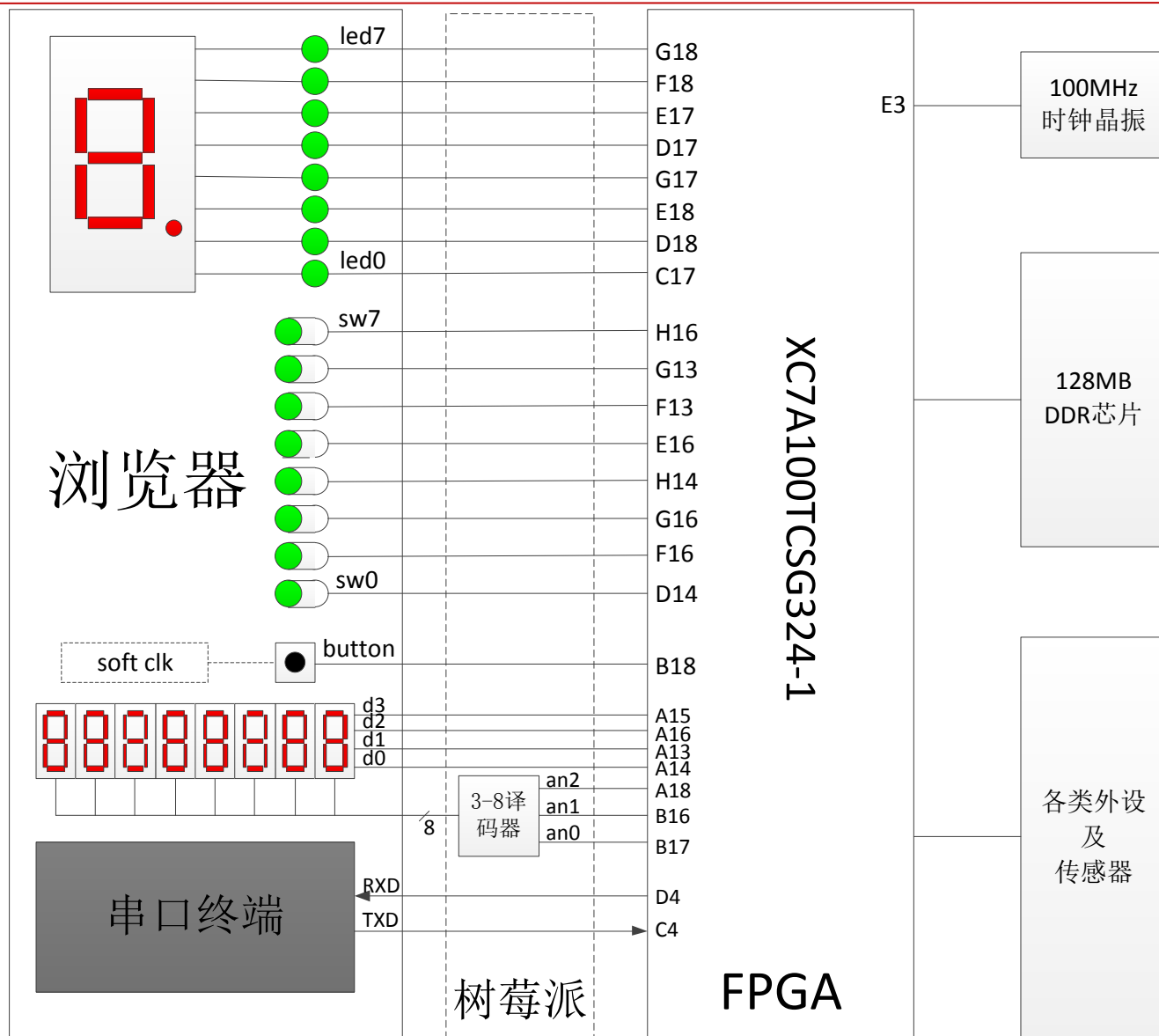
后台物理设备节点



# 实验原理

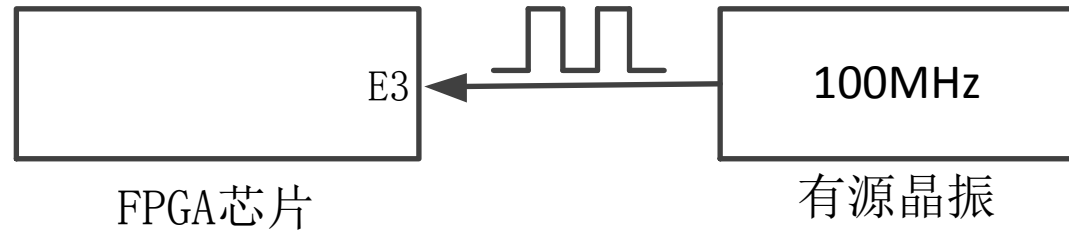
FPGAOL平台的每个设备节点都包含了一个树莓派和一个FPGA板卡，树莓派与FPGA芯片的26个IO管脚直接相连，并最终映射到网页端

--LED \*8  
--开关 \*8  
--数码管 \*8  
--按键 \*1  
--串口 \*1



# 实验原理

- 100MHz时钟信号
- 输入信号
- 为FPGA芯片提供固定的100MHz时钟
- 在XDC文件中将clk信号绑定到E3管脚



```
module test(input clk, .....);  
always@(posedge clk or negedge rst)  
begin  
    ...  
end  
...
```

The screenshot shows a text editor window titled "FPGA0L.xdc". The file path is "E:/code/lab07\_cnt/FPGA0L.xdc". The code content is as follows:

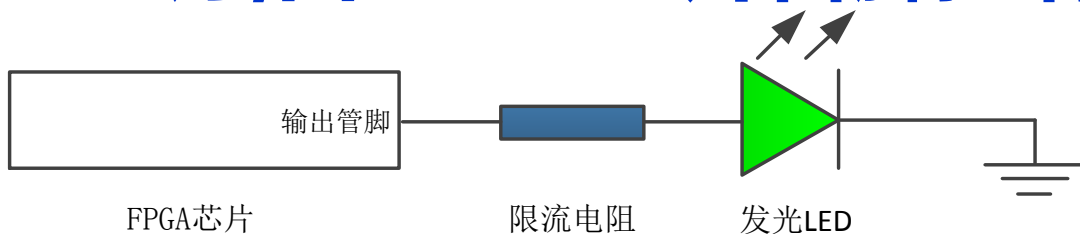
```
6 ## Clock signal  
7 set_property -dict { PACKAGE_PIN E3      IOSTANDARD LVCMOS33 } [get_ports { clk }]; #IO_L12P_  
8 #create_clock -add -name sys_clk_pin -period 10.00 -waveform {0 5} [get_ports {CLK100MHZ}];
```

In the code, the text `{ clk }` in line 7 is highlighted with a red rectangular box.

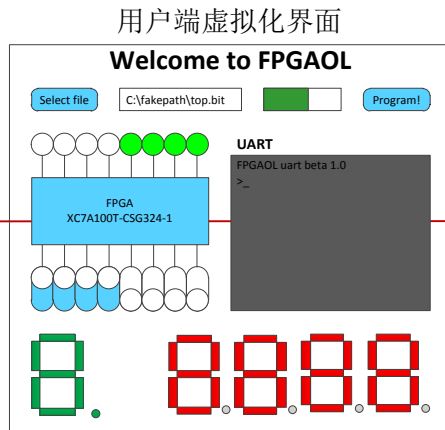
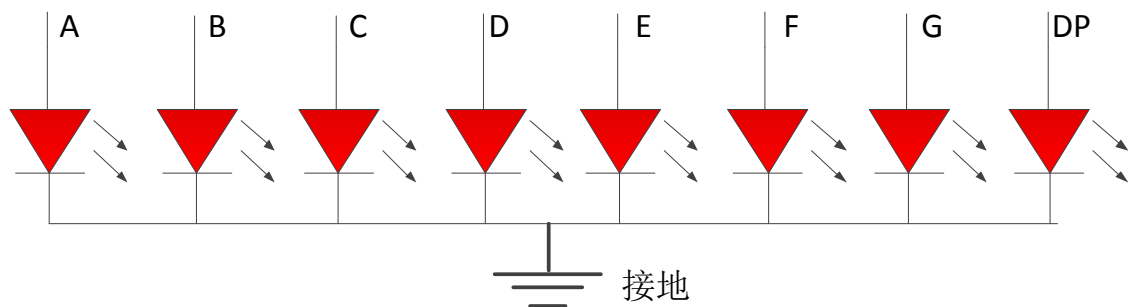


# 实验原理

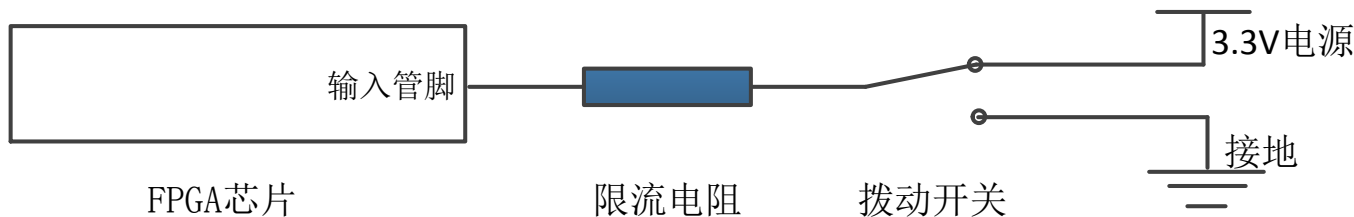
## ■ LED灯,由FPGA芯片管脚控制亮灭



## ■ 七段数码管,与LED灯复用相同的FPGA管脚



## ■ 拨动开关,可向FPGA芯片管脚输入高低电平

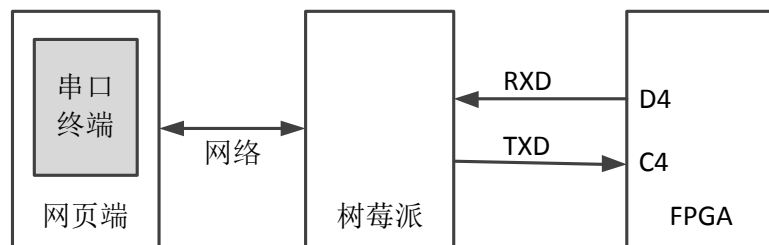
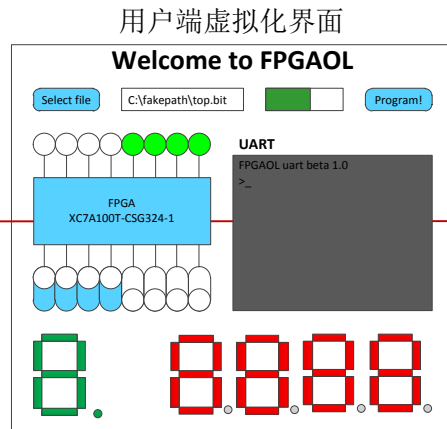


# 实验原理

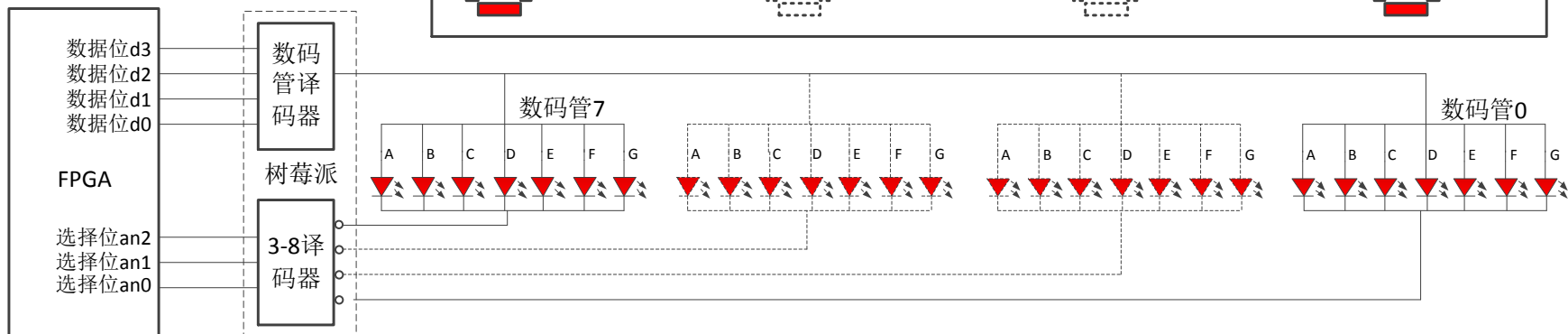
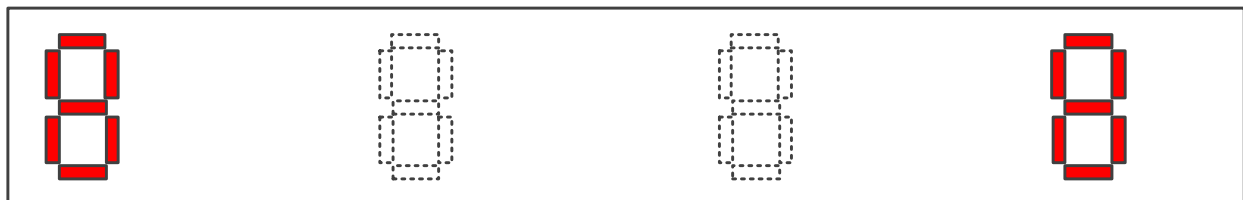
## ■ 串口终端

### ■ 环回测试代码

```
module uart_test(  
    input  uart_tx,  
    output uart_rx);  
    assign uart_rx = uart_tx;  
endmodule
```



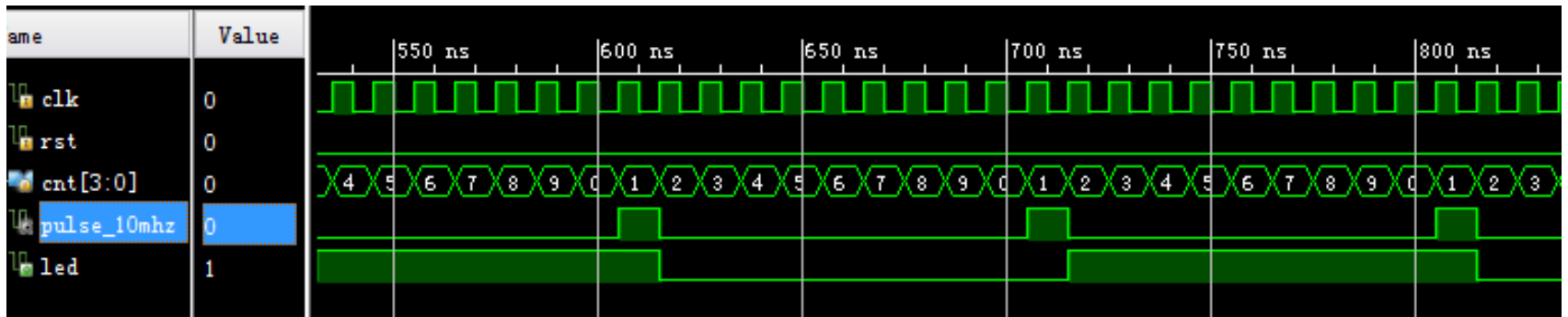
## ■ 十六进制数码管



# 实验原理

## ■ 时钟信号使用

- 通过计数器产生一个低频的脉冲信号，然后再将该脉冲信号控制其他逻辑的控制信号



时钟信号只应该出现在always语句的时序控制部分

正确示例: `always@(posedge clk)` //时钟上升沿触发, 同步复位或无复位

正确示例: `always@(negedge clk)` //时钟下降沿触发, 同步复位或无复位

正确示例: `always@(posedge clk or posedge rst)` //异步复位, 高有效

正确示例: `always@(posedge clk or negedge rst_n)` //异步复位, 低有效

## ■ 时钟管理单元

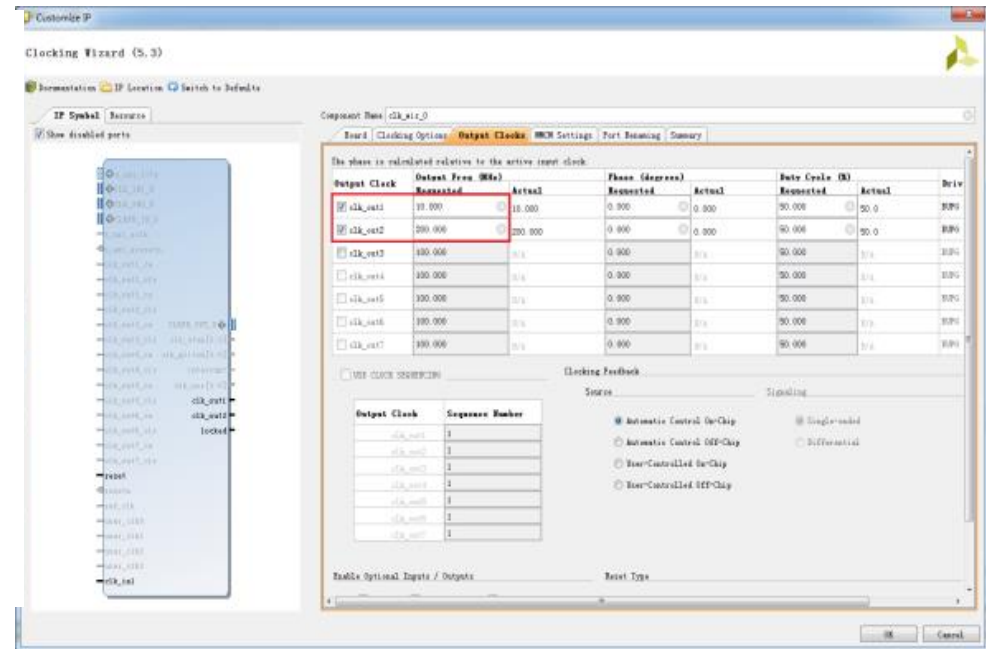
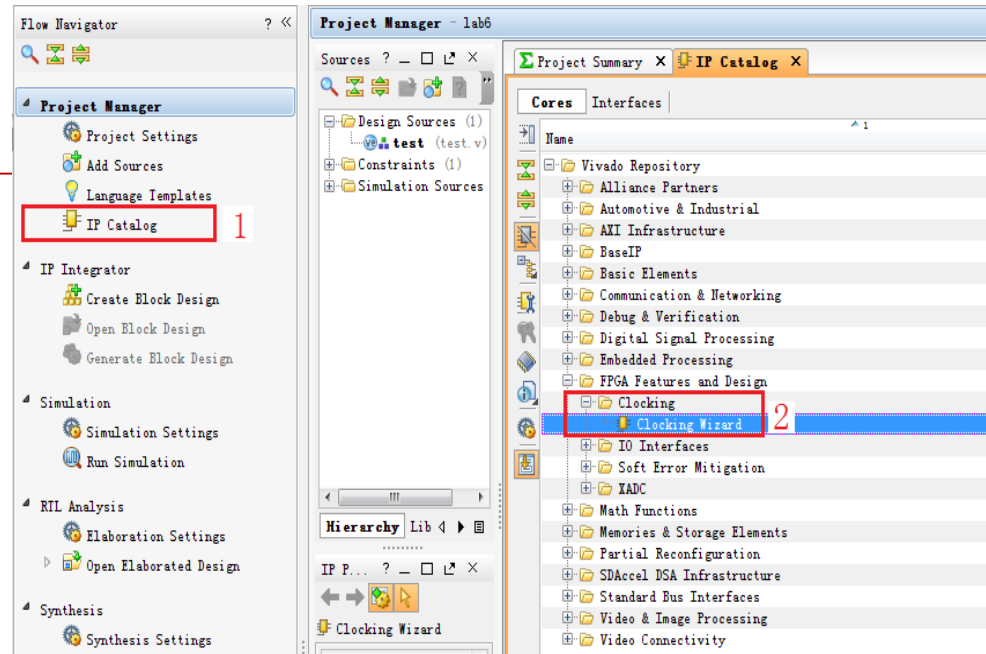
# 实验原理

## ■ 时钟管理单元

工程目录\工程

名.srcs\sources\_1\ip\clk\_wiz\_0\clk\_wiz\_0.v

```
module clk_wiz_0
(
// Clock in ports
input          clk_in1,
// Clock out ports
output         clk_out1,
output         clk_out2,
// Status and control signals
input          reset,
output         locked
);
```

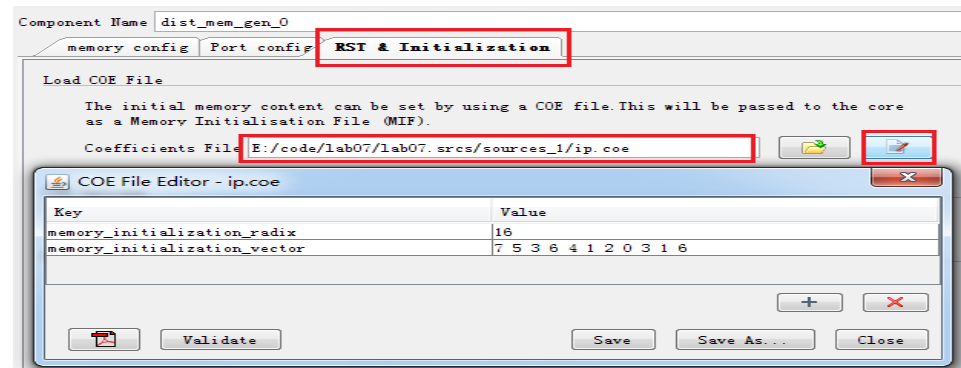
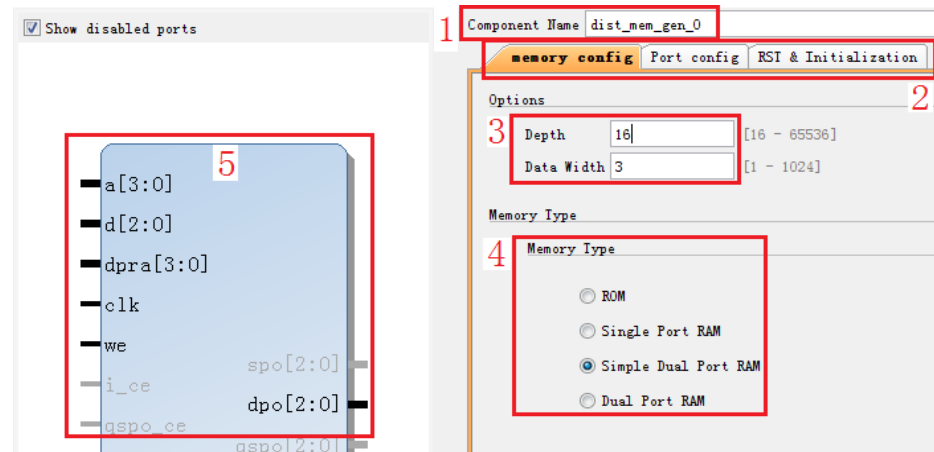
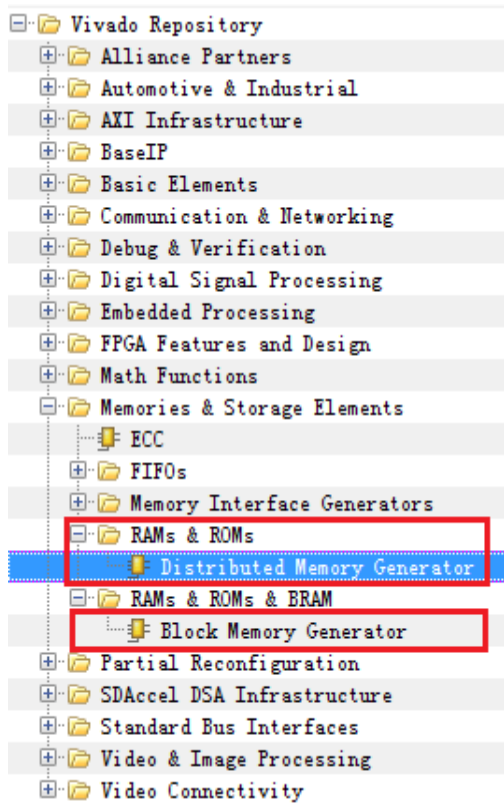


# 实验原理

## ■ 存储器

```
memory_initialization_radix=16;
```

```
memory_initialization_vector=7 5 3 6 4 1 2 0 3 1 6;
```



# 实验内容

---

- 按时完成实验指导书上的所有题目
- 按时提交实验报告

---

**谢谢！**