

中国科学技术大学计算机学院
《数字电路实验》报告



实验题目：FPGA 原理及 Vivado 综合

学生姓名：陈鸿绪

学生学号：PB21000224

完成日期：11.12

计算机实验教学中心制

2020 年 09 月

【实验题目】

FPGA 原理及 Vivado 综合

【实验目的】

了解 FPGA 工作原理；

了解 Verilog 文件和约束文件在 FPGA 开发中的作用；

学会使用 Vivado 进行 FPGA 开发的完整流程；

【实验环境】

VLAB 平台： vlab.ustc.edu.cn

FPGAOL 实验平台： fpgaol.ustc.edu.cn

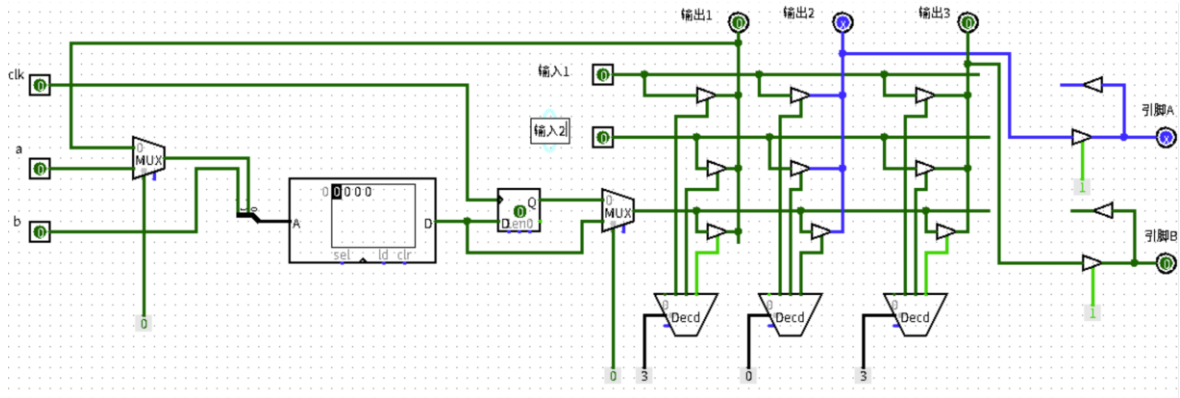
Logisim Vivado 工具

【实验练习】

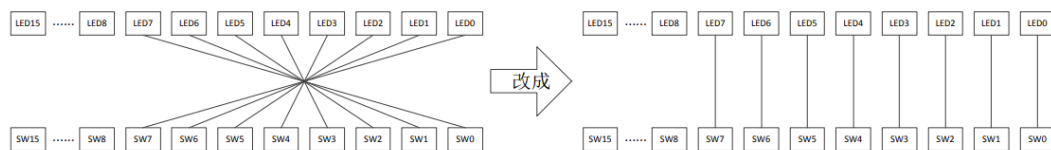
T1. 请通过实验中给出的可编程逻辑单元、交叉互连矩阵及 IOB 电路图，实现如下代码，并将其输出到引脚 B 上。给出配置数据和电路截图。

```
module test(input clk,output reg a);  
always@(posedge clk)  
a <= a ^ 1'b1;  
endmodule
```

[解]：配置数据与电路截图如下图所示：



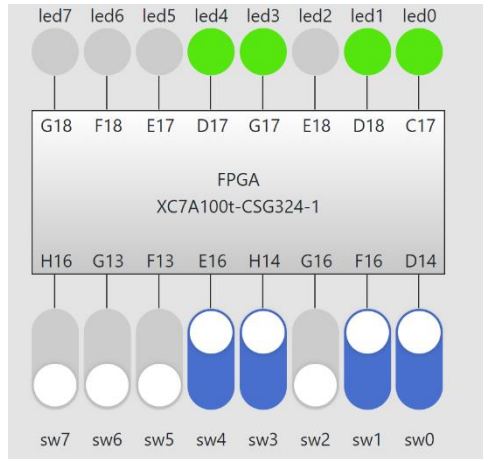
T2. 实验中的开关和 LED 的对应关系是相反的，即最左侧的开关控制最右侧的 LED，最右侧的开关控制最左侧的 LED，请修改实验中给出的 XDC 文件，使开关和 LED 一一对应（最左侧的开关控制最左侧的 LED），如下图所示。



[解]：修改后文件如下图所示：

```
/home/ubuntu/project_9/project_9_srcs/constrs_1/new/test.xdc
1 set_property -dict { PACKAGE_PIN E3 IOSTANDARD LVCMOS33 } [get_ports { clk }];
2 set_property -dict { PACKAGE_PIN B18 IOSTANDARD LVCMOS33 } [get_ports { rst }];
3 set_property -dict { PACKAGE_PIN G18 IOSTANDARD LVCMOS33 } [get_ports { led[0] }];
4 set_property -dict { PACKAGE_PIN F18 IOSTANDARD LVCMOS33 } [get_ports { led[1] }];
5 set_property -dict { PACKAGE_PIN E17 IOSTANDARD LVCMOS33 } [get_ports { led[2] }];
6 set_property -dict { PACKAGE_PIN D17 IOSTANDARD LVCMOS33 } [get_ports { led[3] }];
7 set_property -dict { PACKAGE_PIN G17 IOSTANDARD LVCMOS33 } [get_ports { led[4] }];
8 set_property -dict { PACKAGE_PIN E18 IOSTANDARD LVCMOS33 } [get_ports { led[5] }];
9 set_property -dict { PACKAGE_PIN D18 IOSTANDARD LVCMOS33 } [get_ports { led[6] }];
10 set_property -dict { PACKAGE_PIN C17 IOSTANDARD LVCMOS33 } [get_ports { led[7] }];
11
12 set_property -dict { PACKAGE_PIN D14 IOSTANDARD LVCMOS33 } [get_ports { sw[0] }];
13 set_property -dict { PACKAGE_PIN F16 IOSTANDARD LVCMOS33 } [get_ports { sw[1] }];
14 set_property -dict { PACKAGE_PIN G16 IOSTANDARD LVCMOS33 } [get_ports { sw[2] }];
15 set_property -dict { PACKAGE_PIN H14 IOSTANDARD LVCMOS33 } [get_ports { sw[3] }];
16 set_property -dict { PACKAGE_PIN E16 IOSTANDARD LVCMOS33 } [get_ports { sw[4] }];
17 set_property -dict { PACKAGE_PIN F13 IOSTANDARD LVCMOS33 } [get_ports { sw[5] }];
18 set_property -dict { PACKAGE_PIN G13 IOSTANDARD LVCMOS33 } [get_ports { sw[6] }];
19 set_property -dict { PACKAGE_PIN H16 IOSTANDARD LVCMOS33 } [get_ports { sw[7] }];
```

修改 XDC 文件结束后，生成 bit 文件，烧写到 FPGA 内，拨动 8 个虚拟开关，观察 LED 运行状态如下图所示：



可以发现修改后的开关与 LED 是一一对应的。

T3. 设计一个 30 位计数器, 每个时钟周期加 1, 用右侧的 8 个 LED 表示计数器的高 8 位, 观察实际运行结果。将该计数器改成 32 位, 将高 8 位输出到 LED, 与前面的运行结果进行对比, 分析结果及时钟信号在其中所起的作用。

[解]: 首先 Verilog 写出 30 位计数器的设计文件, 代码如下:

```
module test1(
    input clk,rst,
    output reg [7:0] led
);
    reg [29:0] temp=0;//若为 32 位计数器, 将 29 改为 31
    always@(posedge clk or posedge rst)
        begin
            if(rst)
                begin
```

```

        temp=0;

        led = 8'b0;

    end

    else

    begin

        temp=temp+1;

        led=temp[29:22];

        //32 位计数器需要将 29、22 分别改为 31, 24

    end

end

endmodule

```

编写 XDC 文件，文件代码截屏如下：

```

/home/ubuntu/project_10/project_10.srcs/constrs_1/new/test1.xdc
1 set_property -dict { PACKAGE_PIN E3 IOSTANDARD LVCMOS33 } [get_ports { clk }];
2 set_property -dict { PACKAGE_PIN B18 IOSTANDARD LVCMOS33 } [get_ports { rst }];
3 set_property -dict { PACKAGE_PIN G18 IOSTANDARD LVCMOS33 } [get_ports { led[0] }];
4 set_property -dict { PACKAGE_PIN F18 IOSTANDARD LVCMOS33 } [get_ports { led[1] }];
5 set_property -dict { PACKAGE_PIN E17 IOSTANDARD LVCMOS33 } [get_ports { led[2] }];
6 set_property -dict { PACKAGE_PIN D17 IOSTANDARD LVCMOS33 } [get_ports { led[3] }];
7 set_property -dict { PACKAGE_PIN G17 IOSTANDARD LVCMOS33 } [get_ports { led[4] }];
8 set_property -dict { PACKAGE_PIN E18 IOSTANDARD LVCMOS33 } [get_ports { led[5] }];
9 set_property -dict { PACKAGE_PIN D18 IOSTANDARD LVCMOS33 } [get_ports { led[6] }];
10 set_property -dict { PACKAGE_PIN C17 IOSTANDARD LVCMOS33 } [get_ports { led[7] }];
11

```

生成 bit 文件，烧写到 FPGA 内，观察 LED 的运行状态：

发现 LED 灯首先亮一盏，其次依次亮两盏，然后依次亮三盏（均为从左往右）．．．最后八盏灯依次亮起，不断重复以上过程。

对 32 位的计数器，Verilog 代码只需要按照注释修改以上代码的若干数，而 XDC 文件不需要改变。LED 运行结果与 30 位几乎相同，不同的是 LED 灯亮的时间间隔会稍长一些。

时钟信号作用：1. 由于不同位数的计数器的高位对齐的相同位增加 1 所需要的时钟周期不同，所以时钟信号会使两个计数器对应的 LED 依次亮的时间间隔的不同显现出来。2. 由于每个时钟周期计数器都会加 1，故时钟信号会使 LED 灯亮的频率下降，使得更好观察。

【总结与思考】

总结：我了解了 FPGA 工作原理；并懂得了 Verilog 文件和约束文件在 FPGA 开发中的作用；同时学会了使用 Vivado 进行 FPGA 开发的完整流程；