

数据选择器

实验介绍

本实验将使用 Verilog 语言设计方法实现一个 4 位 2 选 1 多路选择器。

实验目标

1. 学习设计一个多路选择器
2. 学习设计仿真工具的使用方法
3. 学习如使用开发板

实验原理

数据选择器是一个多输入、单输出电路。数据选择器在地址码（或叫选择控制）电平的控制下，从几个数据输入中选择一个，并将其送到输出端。

接口定义，可以直接复制到文件中

（请务必按照接口定义编写代码，在将来的实验中也是如此，模块名也请按照给出的定义命名）

```
module mux(  
  
    input [3:0] a, // 4 位输入 a  
  
    input [3:0] b, // 4 位输入 b  
  
    input s,          // 1 位输入，当 s=0 时 r 的值等于 a，s=1 时 r 的值等于 b  
  
    output [3:0] r // 4 位输出，相应位的值由 a b s 的值确定  
  
);
```

实验步骤

1. 新建 ISE 工程

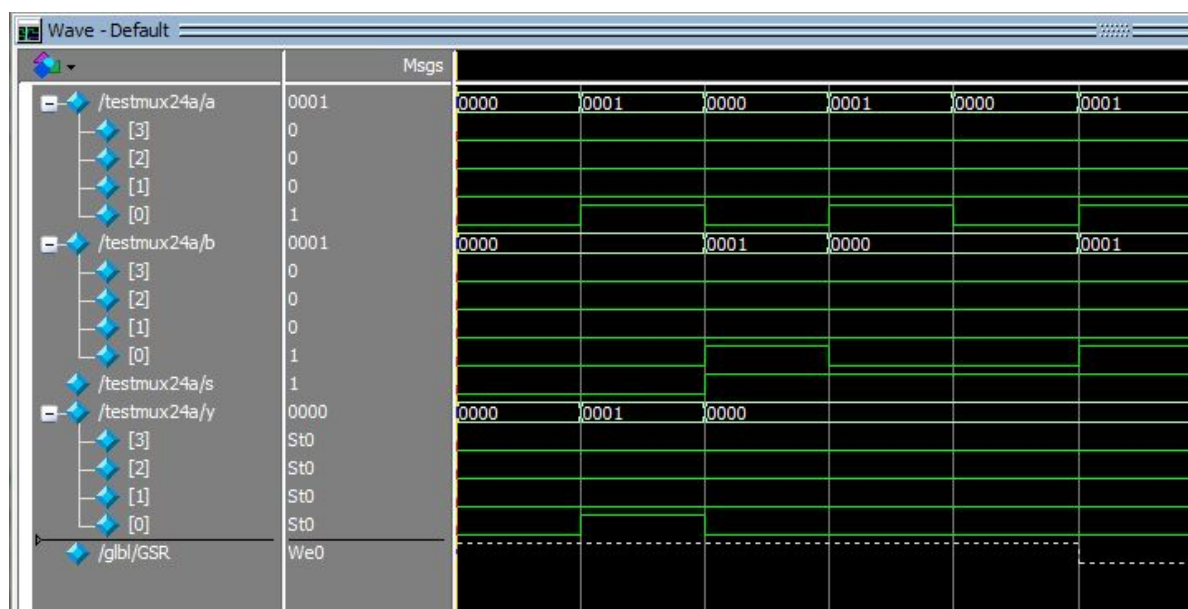
- 在 ISE 中点击 File -> New Project，按照弹出的对话框提示新建工程。
- 在 Project Setting 页面，我们选用的开发版为 Spartan6 系列，XC6SLX16；
Simulator 使用 Modelsim-SE Verilog；Preferred language 选择 Verilog。

2. 编写模块代码

- 点击 New Source，在弹出的对话框中选择 Verilog Module 添加模块，并且输入 Verilog 文件名。
- 单击 Next 进入 Define Module。略过这一步，在程序中定义端口。
- 编写代码

3. 对代码进行行为仿真以及购房真

- 完成代码编写后，对模块进行测试。在工程管理区勾选 Simulation。
- 点击 new source，选择 Verilog Test Fixture
- 选择要进行测试的模块名称，本实验中只有 mux2x4 模块。
- 完成测试代码的编写后，双击 Simulate Behavioral Model，进行仿真。
- 观察波形，验证程序的正确性。



4.完成选择器的下板

- a) 右键点击工程，选择 **New Source**，选择 **Implementation Constraints File**
- b) 编写 ucf 文件，将选择器的输出 r 连接到 LED 灯，输入 a 和输入 b 连接到开发板上的 switch 开关，输入 s 连接到开发板的某个按钮上

eg: 下面演示将 a[0],b[0],s,r[0] 的 ucf 文件编写，剩下的请自行补全

1、从 mips 网站上下载 ucf 文件的压缩包，根据自己的需要修改 ucf 相应选项

2、找到 led，switch，和 button 在文件中的位置

Leds

```
#NET "Led<0>"      LOC = "U16" | IOSTANDARD = "LVCMOS33"; #Bank = 2, Pin name =  
IO_L2P_CMPCLK,      Sch name = LD0
```

Switches

```
#NET "sw<0>"      LOC = "T10" | IOSTANDARD = "LVCMOS33"; #Bank = 2, Pin name =  
IO_L29N_GCLK2,      Sch name = SW0
```

```
#NET "sw<4>"      LOC = "N8" | IOSTANDARD = "LVCMOS33"; #Bank = 2, Pin name = IO_L40N,  
Sch name = SW4
```

Buttons

```
#NET "btn<0>"      LOC = "B8" | IOSTANDARD = "LVCMOS33"; #Bank = 0, Pin name = IO_L33P,  
Sch name = BTNS
```

3、将 NET 后的信号改成模块与之相应的输入与输出

Leds

```
#NET "r<0>"      LOC = "U16" | IOSTANDARD = "LVCMOS33"; #Bank = 2, Pin name =  
IO_L2P_CMPCLK,      Sch name = LD0
```

Switches

```
#NET "a<0>"      LOC = "T10" | IOSTANDARD = "LVCMOS33"; #Bank = 2, Pin name =  
IO_L29N_GCLK2,      Sch name = SW0
```

```
#NET "b<0>"      LOC = "N8" | IOSTANDARD = "LVCMOS33"; #Bank = 2, Pin name = IO_L40N,  
Sch name = SW4
```

Buttons

```
#NET "s"      LOC = "B8" | IOSTANDARD = "LVCMOS33"; #Bank = 0, Pin name = IO_L33P,  
Sch name = BTNS
```

4、去掉“#”注释

- c) 双击 Configure Target Device
- d) 双击 boundary scan
- e) 右键点击右边空白区域，选择 Add Xilinx Device
- f) 选择生成的 bit 文件
- g) 右键点击红色区域，选择 program
- h) 观察实验 现象