****

**计算机组成原理**

**·实验指导书·**

# 目 录

[目 录 2](#_Toc130568471)

[实验要求： 3](#_Toc130568472)

[前言 5](#_Toc130568473)

[实验一 Verilog 与 ModelSim 基础应用 6](#_Toc130568474)

# 实验要求：

1. 前4次实验（实验一~实验四），在自己的电脑上完成，只要在截至日期前，提交仿真结果截图到QQ群的作业里就可以了。

中间4次实验（实验五~实验八），需要到实验室里进行。需要现场验收实验结果，然后在截至日期前提交激励输入文件和仿真结果截图到QQ群的作业里（可以放到一个word文档中再提交）。

每次提交结果到QQ群的作业里时，要在文档里注明你的姓名和学号。以便实验成绩统计。只要将中间4次实验（实验五~实验八）写入一个实验报告文档就可以了。前4次实验（实验一~实验四））不要写实验报告。

实验九和实验十属于选做实验，感兴趣的同学可以选做，鼓励选做。实验成绩满分的名额优先分配给选做了实验九或者实验十的同学。选做的实验需要写实验报告才能实验加分。

1. 所有的仿真都与自己的学号有关，后面的实验指导书会以XH这个变量代表每个同学的学号的后8位。例如：学号为：20210281896同学在做实验时，对应的变量XH=10281896。也可以用学号的后4位或者学号的后两位作为XH变量的值（大家自己决定），相应的实验结果就以这个变量XH的值，作为仿真输入值。由于每个同学的XH值是不同的，所以每个同学获得的结果截图也是不同的。这点需要大家注意啦。
2. 实验总分20分。分配如下：前4次实验只要在QQ群里提交激励输入和相应的仿真结果截图。每个实验最多得分1.5分。后4次实验中，每个实验最多得分3.5分（现场验收1分，提交激励输入和结果仿真截图1分，实验报告1.5分 ）。

4、提交截止时间为见每个实验；每迟交一周扣除获得分数的10%，不足一周按一周计算，迟交时间以截止时间为起点计算，最低扣除到获得分数的70%。

5、一个优秀(81-100)的实验报告（第5~8次实验需要写报告）应该有：

（1）有操作方法与实验步骤、实验结果与分析、讨论与心得三个板块，且每个板块内容完整

（2）其中的操作方法与实验步骤一节：有所有要求的实验步骤的语言描述，对测试文件进行解释，给出必要的自己绘制图形。或者编程代码截图。

（3）其中的实验结果与分析一节：有所有要求的实验步骤对应的结果照片（如波形与下板结果）或者屏幕截图，对波形进行正确充分的解释，对下板结果有正确的解释

（4）其中的讨论与心得一节：能够记录遇到的问题，或对实验提出了三条及以上有意义的建议

实验报告的模板在第五次实验时会发给大家。

6、实验指导书还在不断修订。对实验有任何疑问，欢迎联系ahhfdxq@163.com 。

# 前言

1. **verilog中元件的例化**

就是利用已有的元件，来例化出和它一模一样的新元件，只是管脚的名字不同。元件的例化类似于由类派生出具体的对象。例如：

module add4（A,B,C,V）; //引脚名字分别为A、B、C、V。

input[3:0] A;

input[3:0] B;

output[3:0] C;

output V;

assign {V,C}=A+B;

endmodule

上面的代码定义了add4这个元件，对应的引脚是A[3:0]、B[3:0] 、C[3:0]和V。现在可以通过元件例化方法，生成多个和add4相同的元件。

例如：add4 addone（X,Y,Z,W）; //或者写成add4 addone（ .A(X) , .B(Y), .C(Z), .V(W) ）;

add4 addtwo（M,N,O,P）; //或者写成add4 addtwo（ .A(M) , .B(N), .C(O), .V(P) ）;

add4 addthree（H,I,J,K）; //或者写成add4 addthree（ .A(H) , .B(I), .C(J), .V(K) ）;

就是用add4这个元件例化出来了3个同add4元件结构完全相同，只是引脚名字不同的元件，对应为addone（X,Y,Z,W），addtwo（M,N,O,P），addthree（H,I,J,K）。

1. **verilog中激励输入文件的创建**

激励输入就是给定义好的电路增加输入激励值。在激励输入文件里，通常需要先对电路元件例化，通过例化元件固定引脚名字，然后在initial语句中来对具体的引脚添加激励输入信号。例如：如果设计的电路的顶层文件是topcircute（A,B,C）。我们可以这样来定义激励输入文件。

‘timescale 1ns/1ps; //时间单位是1ns，仿真精度可以达到1ps。

module testcircute\_tb（）;

reg X,Y;

wire Z;

topcircute TOPTEST(X,Y,Z); // topcircute是顶层电路文件，对它实例化，固定下来引脚名字。

//或者topcircute TOPTEST( .A(X), .B(Y), .C(Z) ); 效果同上行。这条例化语句通常放在end后面。

initial

X=0;

Y=1;

always #10 //每隔10个时间单位就进行下面的操作。

begin //begin end在此处功能类似于C语言中的大括号{ }。

X<=~X; //将X的值取反后再赋给X。

Y<=~Y; //将Y的值取反后再赋给Y。

end

endmodule

这个激励输入文件就是先对顶层电路进行例化，从而固定下来例化后元件的引脚名字为X,Y,Z。然后在initial语句中给输入量X、Y进行赋值。

# 实验一 Verilog 与 ModelSim 基础应用

一、实验目的：

熟悉并掌握 Verilog HDL 与 ModelSim 的使用

二、实验环境：

ModelSim

三、实验内容：

学习使用 Verilog 完成 4 选 1 多路选择器的设计和实现,并使用 ModelSim 工具对设计进行仿真和分析验证。

四、实验原理

多路选择器（MUX）是一种在多路数据传送过程中，能够根据需要将其中任意一路选

出来的电路，其原理图和真值表如下图所示。

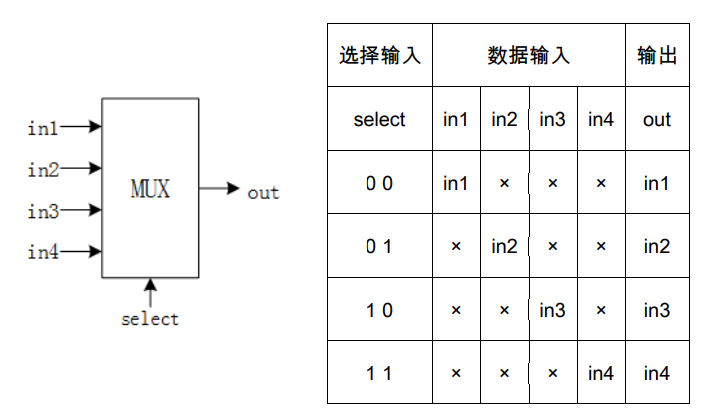
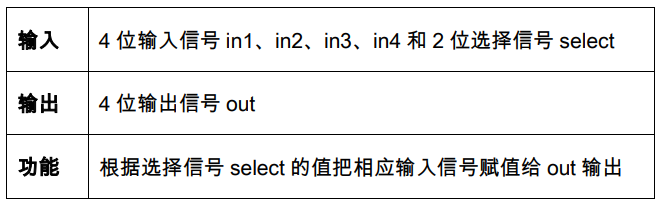


图 1 4 选 1 多路选择器及其真值表

五、实验内容（步骤）

5.1、Verilog 关键代码实现

表 2. MUX 模块功能描述



MUX模块的verilog代码如下∶

module mux41(

input wire [3:0] in1, in2, in3, in4，

input wire [1:0] select，

output reg [3:0] out

);

always@\* begin

case (select)

2'b00: out = in1;

2 'b01: out = in2 ;

2'b10: out = in3;

2 'b11: out = in4;

default: out =4 "bx;

endcase

end

endmodule

5.2、测试文件(TestBench)关键代码描述

‘timescale 1ns/1ps;

module mux41\_tb;

reg [3:0] in1, in2, in3, in4;

reg [1:0] select;

wire [3:0] out;

initial begin

in1 = 4'b0001;

in2 = 4'b0011;

in3 = 4'b0111;

in4 = 4'b1111;

select = 2'b00;

#10 select = 2'b01;

#10 select = 2'b10;

#10 select = 2 'b11;

#10 $stop;

end

mux41 uut (

.in1 (in1) , .in2 (in2) , .in3 (in3) , .in4 (in4) ,.select ( select) , .out (out)

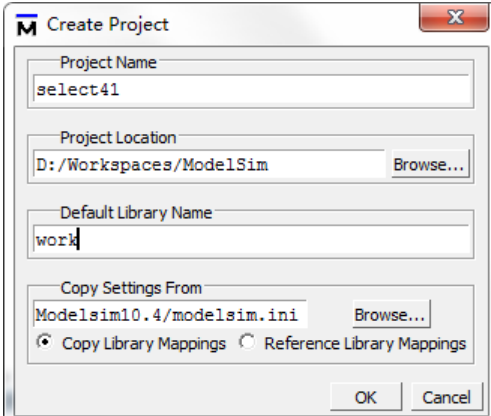
); //元件的例化。

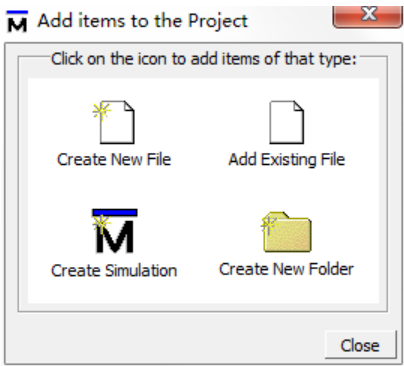
endmodule

5.3、ModelSim 仿真及分析

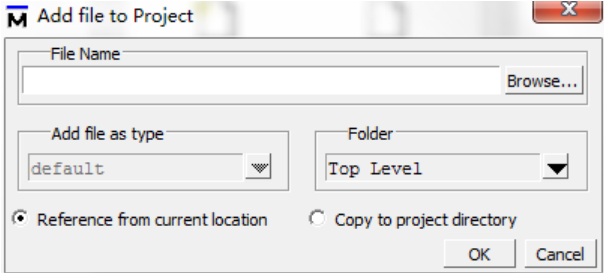
5.3.1建立ModelSim工程

打开ModelSim ,选择File->New->Project ,出现 Create Project对话框﹐填写工程名( Project Name )，选择保存目录(Project Location )，注意保存目录中不要有中文，如下图所示︰

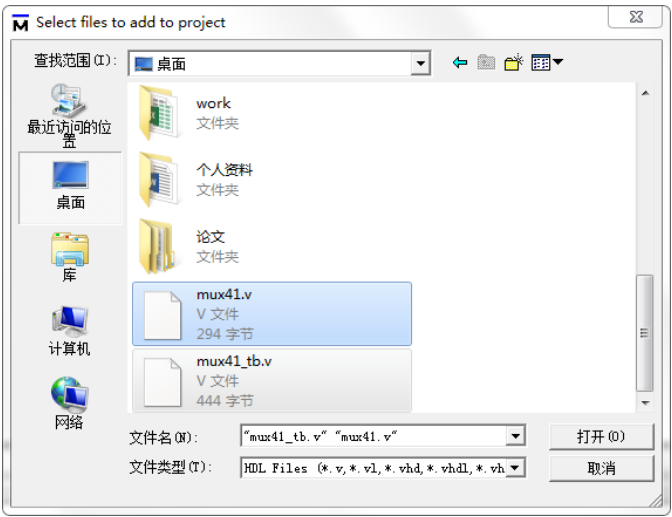


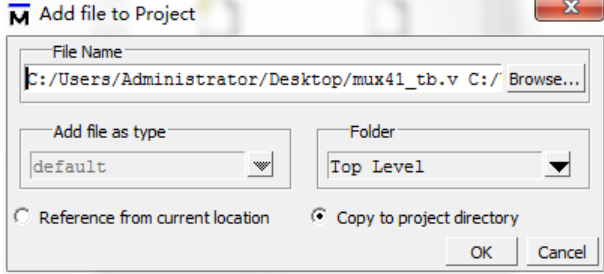
单击 OK 按钮后，会出现下图界面：

现在可以选 Create New File（新建文件）或者 Add Existing File（添加已存在文件）。这里我们选择 Add Existing File，也就是添加 5.1 和 5.2 中的 Verilog 代码，会出现下图界面：

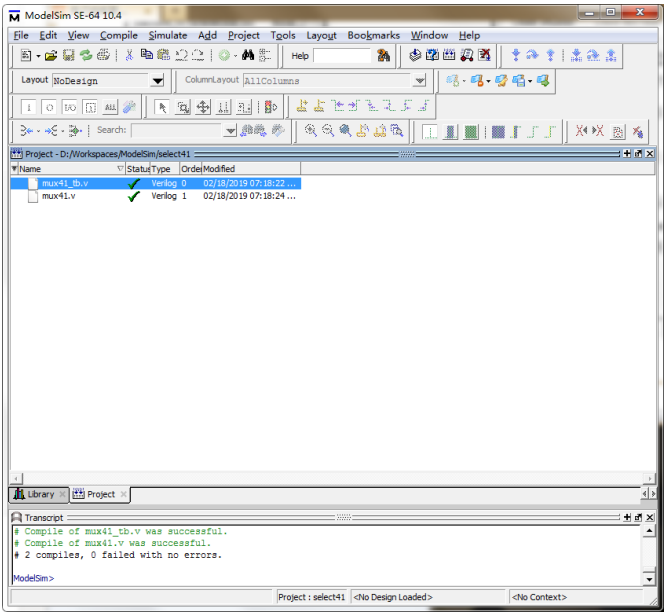


点击 Browse 按钮，添加 5.1 中的 mux42.v 和 5.2 中的测试文件 mux41\_tb.v，会出现下图界面：



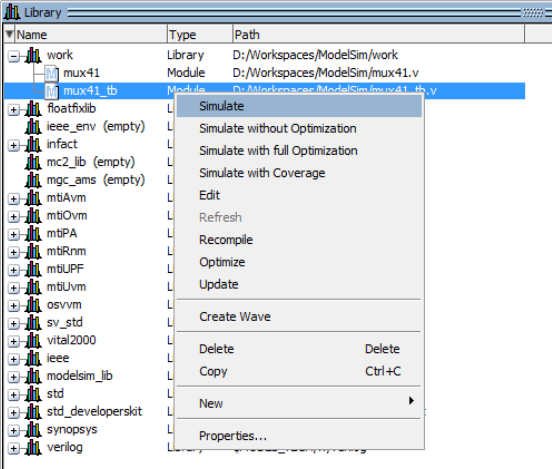
选择要添加的文件后，单击“打开”按钮，即添加完成，会出现下图界面，在其中选择 copy to project directory，这样就会将 mux42.v 和 mux41\_tb.v 文件复制到新的工程目录下，单击OK 按钮。

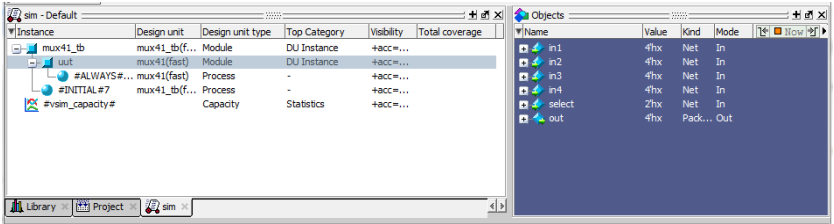
文件添加完成后，ModelSim 主界面会显示所有文件的状态。选中任意一个文件，右键单击，选择 Compile->Compile All，即开始编辑所有文件，会出现下图界面。没有出错，文件状态应该都是绿色的对号，否则点击屏幕下方的 Transcript，查看出错信息，直至无误



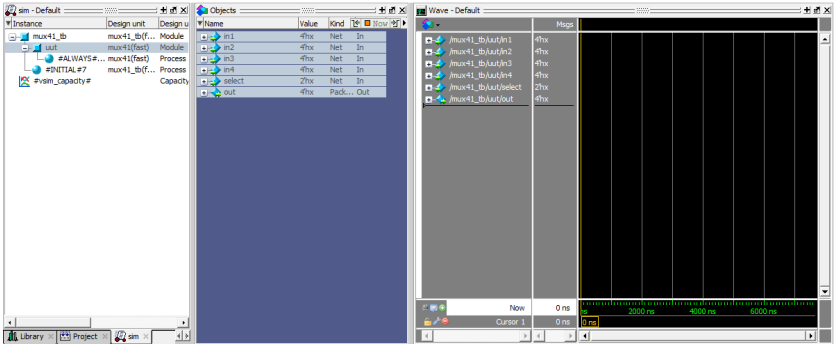
5.3.2 开始仿真

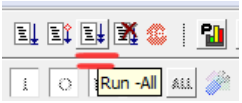
切换到 Library，然后展开 work 目录，在 mux41\_tb.v 文件上单击右键，在弹出菜单中选择 Simulate （without Optimization），如下图界面：



此时会出现一个名为 sim 的界面，展开其中的 mux41\_tb 节点，选择 uut，会在 Objects 窗口显示所有信号，如下图所示：（若没有出现 Objects 窗口，可以通过菜单 View->Objects调出该窗口）

选择 Objects 窗口的所有信号（Ctrl+A），然后单击右键，在弹出菜单中选择 Add

to->Wave->Selected Signals，如下图所示：

单击工具栏中的 Run-All 按钮，便开始仿真，如下图所示：

仿真效果图，如下图所示：

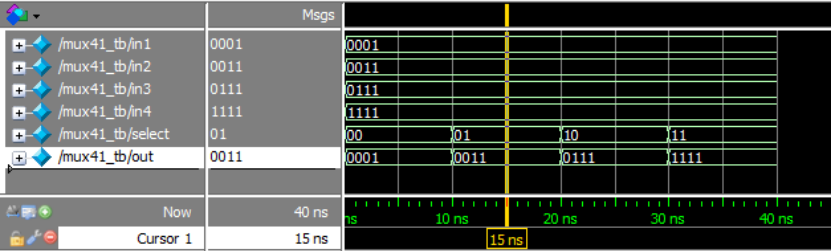
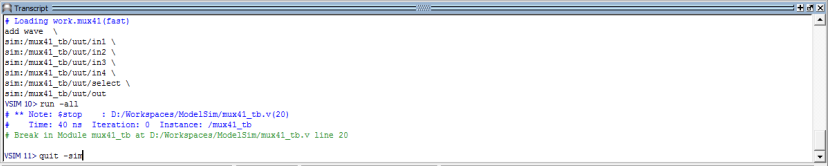


图 X. 多路选择器仿真结果

结束后，请在 Transcript 中输入 quit –sim 命令退出，如下图所示：



编写对应的输入激励文件，进行仿真。编写的输入激励文件和仿真截图需要提交到QQ群里的对应的实验作业提交里。提交截止时间：2023年4月2号晚上9点。

六、感兴趣的同学，可以进一步实验，不需要提交结果截图：

请使用 Verilog 完成 4 位全加器、8 位比较器、74138 译码器等模块设计,然后编写测试文件使用 ModelSim 进行仿真验证。