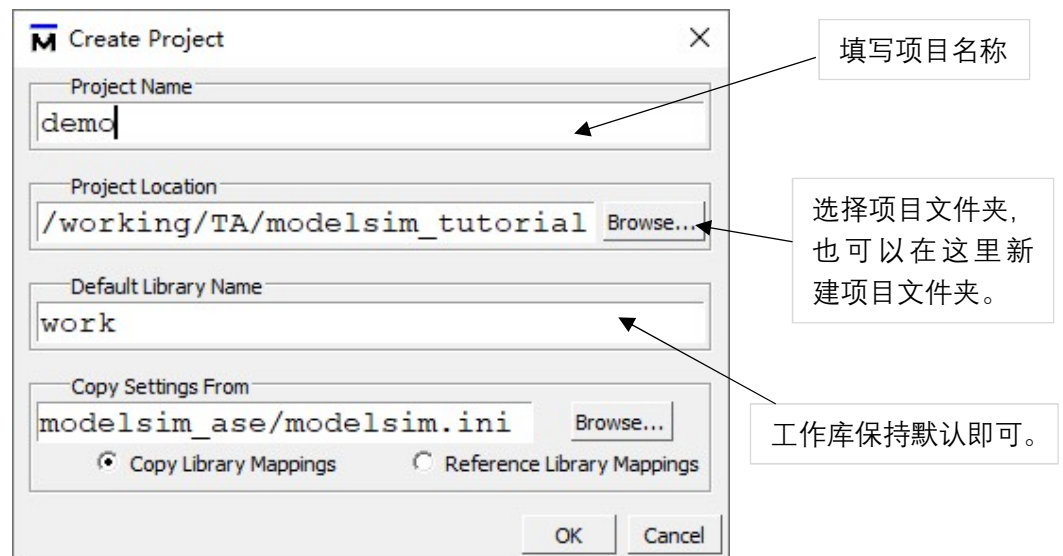


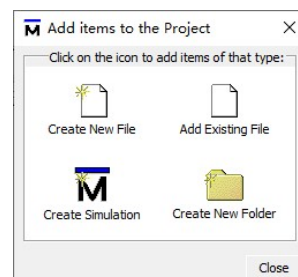
ModelSim 仿真

下面以寄存器组为例，简单地介绍 ModelSim 创建工程与仿真的流程。

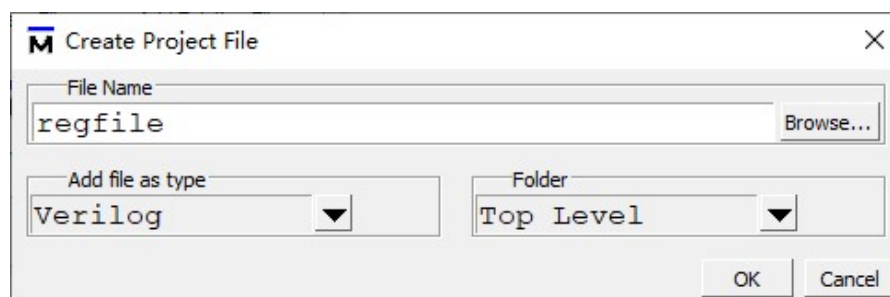
在菜单栏中选择 File > New > Project 创建一个新项目。



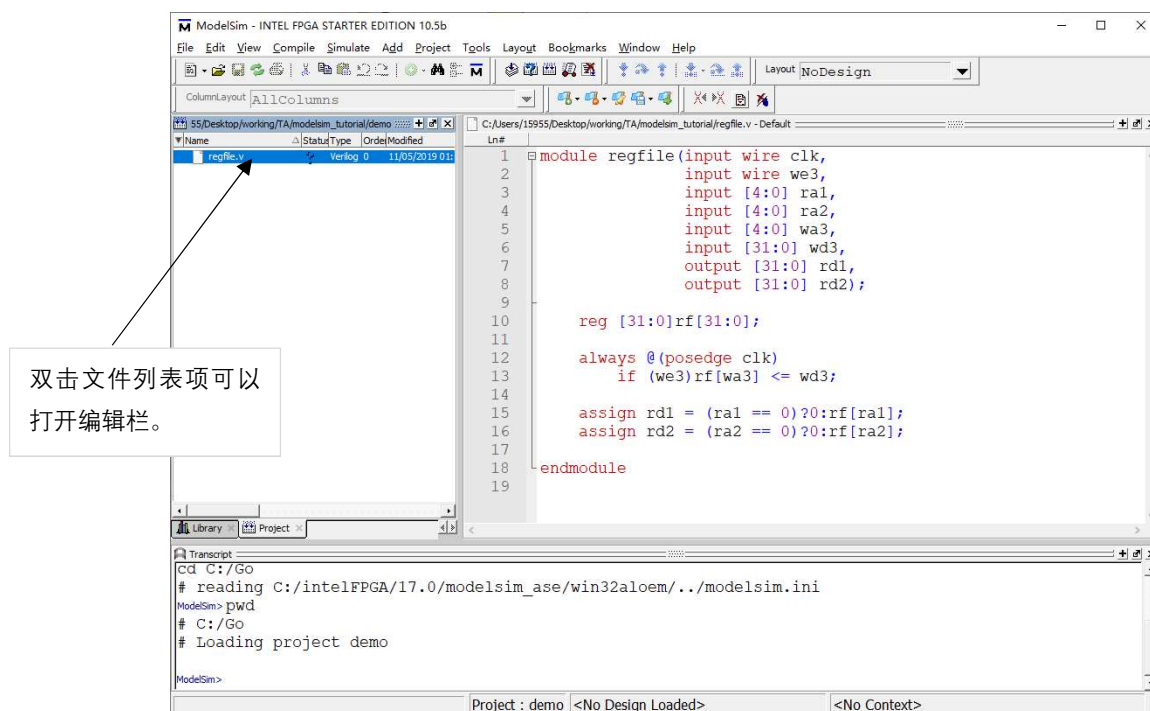
点击 OK 之后，会出现 “Add items to the Project” 对话框，这时可以点击 “Create New File” 创建一个新文件。



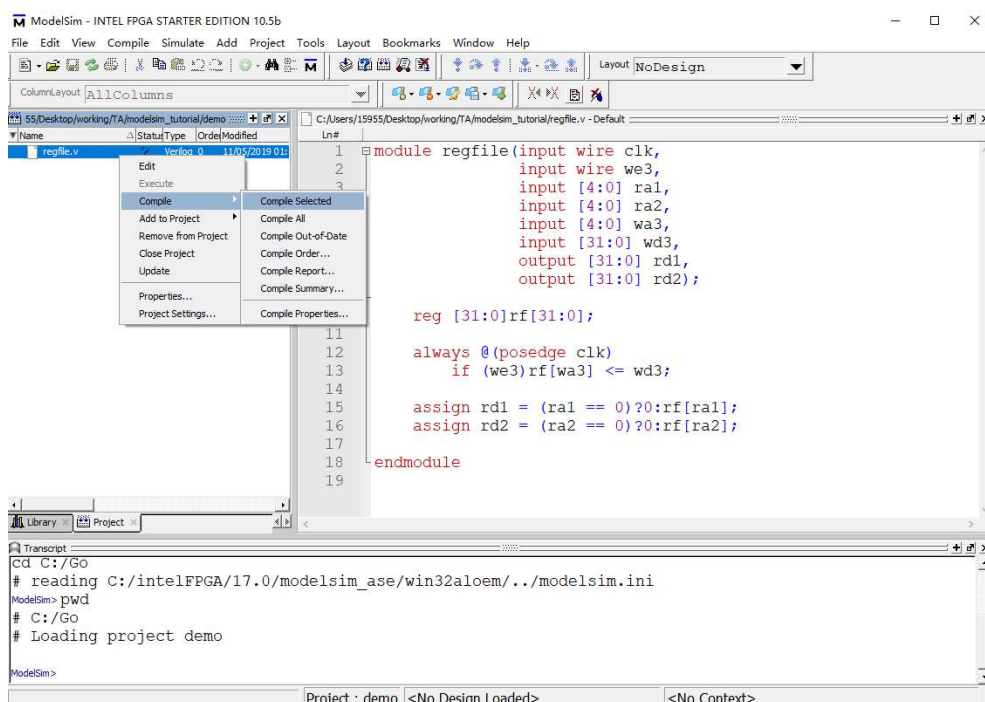
我们将文件命名为 regfile，文件类型选择 Verilog，点击 OK。



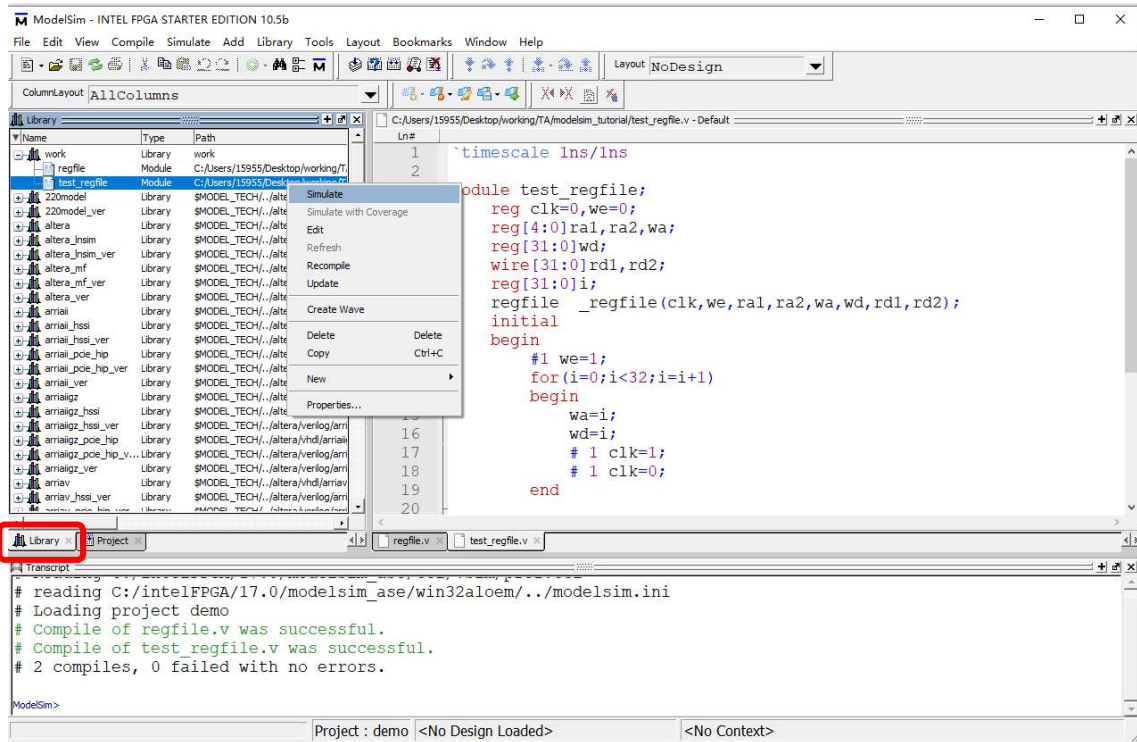
双击文件列表项打开编辑栏，将程序 regfile.v 拷贝进编辑框。



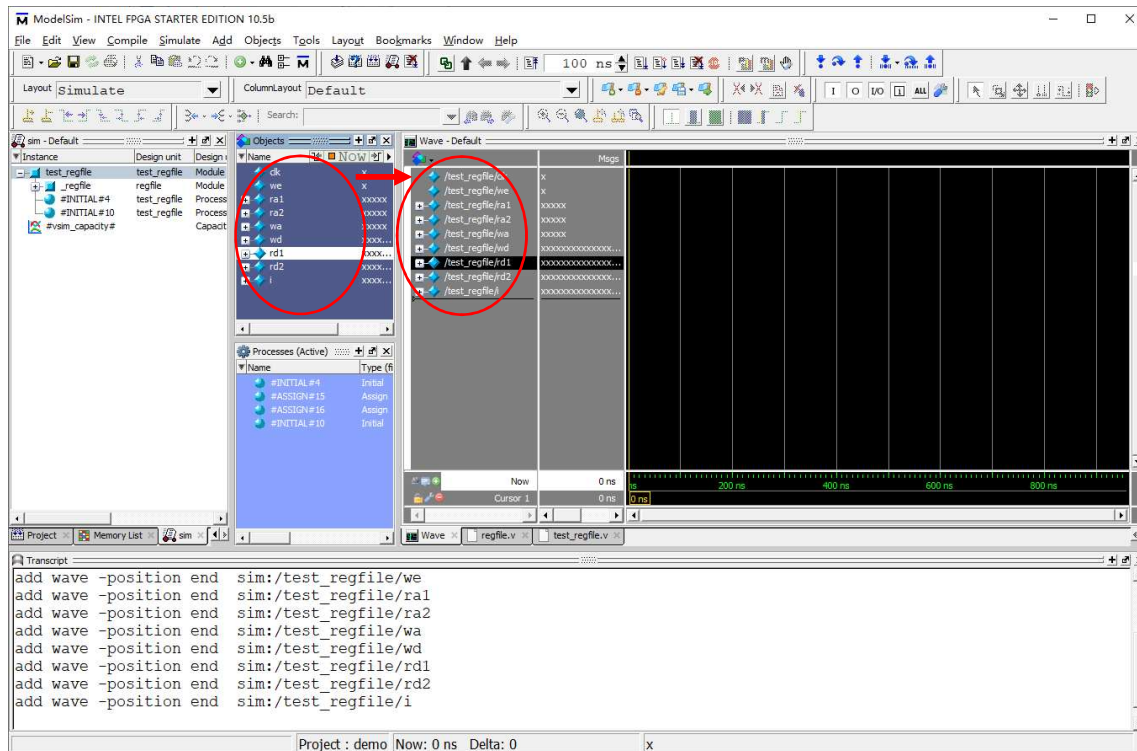
在文件列表项上单击右键，选择 Compile>Compile Selected，可以编译指定的文件，也可以选择 Compile All 编译整个工程。



在项目栏中单击右键，选择 Add to Project > New File，添加 verilog 文件 test_regfile.v，用于测试刚刚编写的寄存器组 regfile。正确编译所有文件后，点击 Library 选项，测试模块 test_regfile 上单击右键，点击 simulate。



测试界面如下所示。如果没有出现如下窗口，可在菜单栏中点击 View>Wave 手动添加。将需要显示的信号从 Object 窗口拖入 Wave 窗口中 (tips: shift+鼠标左键可以批量选择)。



菜单栏中点击 Simulate>Run>Run All，可以运行测试文件。

附录：

Regfile.v

```
module regfile(input wire clk,
               input wire we3,
               input [4:0] ra1,
               input [4:0] ra2,
               input [4:0] wa3,
               input [31:0] wd3,
               output [31:0] rd1,
               output [31:0] rd2);

    reg [31:0] rf[31:0];

    always @(posedge clk)
        if (we3) rf[wa3] <= wd3;

    assign rd1 = (ra1 == 0)?0:rf[ra1];
    assign rd2 = (ra2 == 0)?0:rf[ra2];

endmodule
```

test_regfile.v

```
`timescale 1ns/1ns

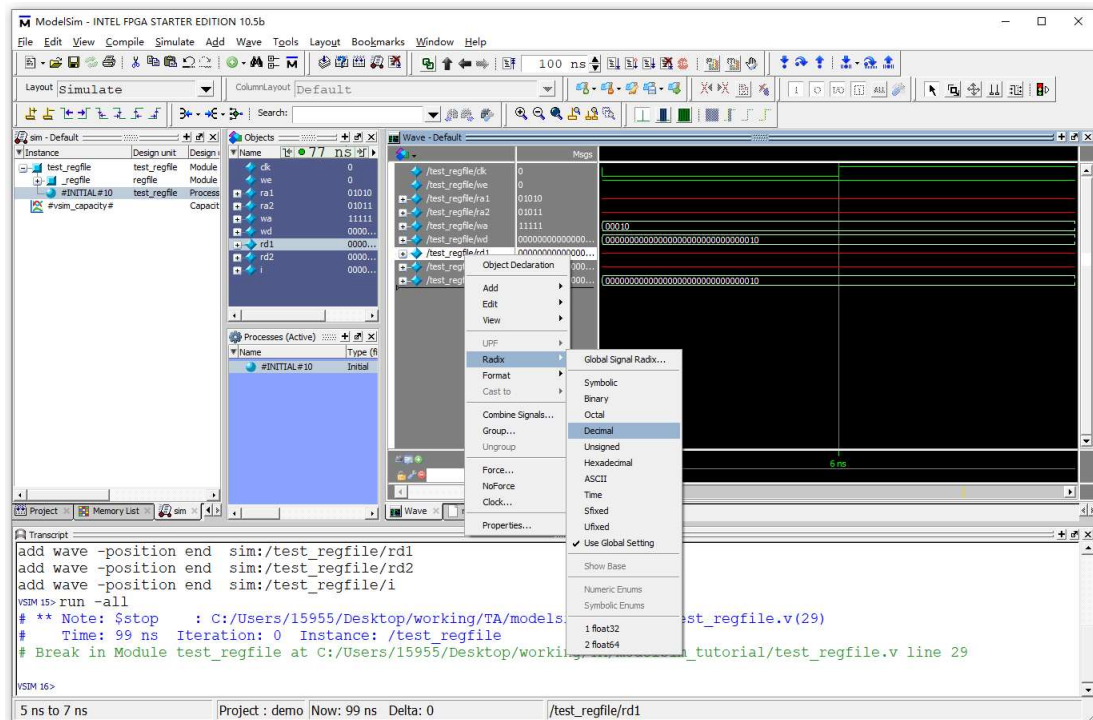
module test_regfile;
    reg clk=0, we=0;
    reg [4:0] ra1, ra2, wa;
    reg [31:0] wd;
    wire [31:0] rd1, rd2;
    reg [31:0] i;
    regfile _regfile(clk, we, ra1, ra2, wa, wd, rd1, rd2);
    initial
    begin
        #1 we=1;
        for(i=0; i<32; i=i+1)
            begin
                wa=i;
                wd=i;
                #1 clk=1;
                #1 clk=0;
            end
    end
endmodule
```

```

#1 we=0; #1
for(i=0;i<16;i=i+1)
begin
    ra1=2*i;
    ra2=2*i+1;
    # 1 clk=1;
    # 1 clk=0;
end
$stop;
end
endmodule

```

怎样更改显示的数据格式？



怎样快速的查看寄存器数组？

The screenshot shows the ModelSim interface with the Memory List window open. A red arrow points to the selected memory instance in the list. The Memory List window displays the following data:

| Instance | Range | Name | Value |
|--------------------------|--------|------|-------|
| /test_regfile/regfile... | [31:0] | 77 | 77 |

The Memory Data window shows the memory contents in hexadecimal and binary format. The data is organized into columns, with the first column showing the address and the subsequent columns showing the data values.

The Transcript window shows the simulation commands and output:

```
add wave -position end sim:/test_regfile/rd1
add wave -position end sim:/test_regfile/rd2
add wave -position end sim:/test_regfile/i
VSI1M 15> run -all
# ** Note: $stop : C:/Users/15955/Desktop/working/TA/modelsim_tutorial/test_regfile.v(29)
# Time: 99 ns Iteration: 0 Instance: /test_regfile
# Break in Module test_regfile at C:/Users/15955/Desktop/working/TA/modelsim_tutorial/test_regfile.v line 29
VSI1M 16>
```

The status bar at the bottom indicates the project name is 'demo', the current time is 99 ns, and the delta time is 0. The selected memory instance is shown as [31:0].