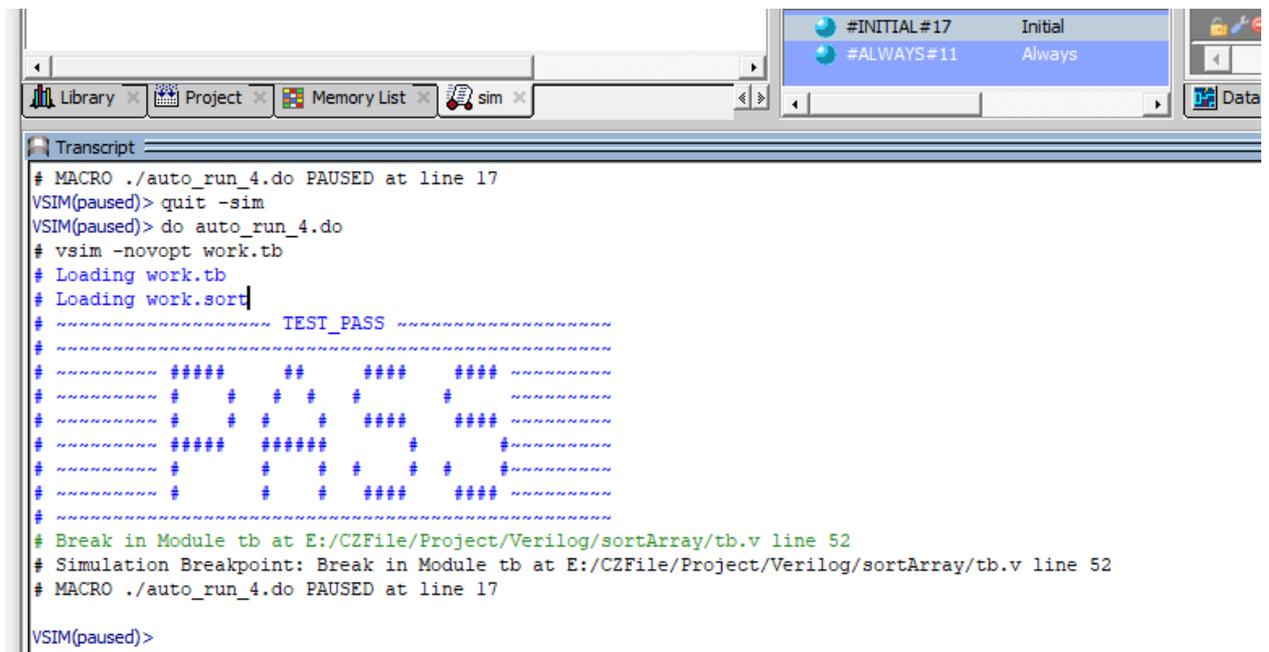


README

1. 将 `sort.v` `tb.v` 加入你的工程。
2. 在 `sort.v` 中实现你的逻辑。

```
1  module sort (  
2      input [3:0] in0,  
3      input [3:0] in1,  
4      input [3:0] in2,  
5      input [3:0] in3,  
6      input clk,  
7      input rst_n,  
8      input enable,  
9      |  
10     output reg [3:0] max0,  
11     output reg [3:0] max1,  
12     output reg [3:0] max2,  
13     output reg [3:0] max3,  
14     output reg valid  
15 );  
16  
17 // 在这里实现你的逻辑  
18  
19  
20 endmodule
```

3. 仿真：在modelsim的tcl命令行输入`do auto_run_4.do`，回车，开始仿真，仿真成功会打印下图中 `TEST PASS`，失败会打印 `TEST FAIL`



The screenshot shows the ModelSim simulation environment. The Transcript window displays the following text:

```
# MACRO ./auto_run_4.do PAUSED at line 17  
VSIM(paused)> quit -sim  
VSIM(paused)> do auto_run_4.do  
# vsim -novopt work.tb  
# Loading work.tb  
# Loading work.sort  
# ~~~~~ TEST_PASS ~~~~~  
# ~~~~~  
# ~~~~~  
# ~~~~~  
# ~~~~~  
# ~~~~~  
# ~~~~~  
# ~~~~~  
# ~~~~~  
# ~~~~~  
# ~~~~~  
# Break in Module tb at E:/CZFile/Project/Verilog/sortArray/tb.v line 52  
# Simulation Breakpoint: Break in Module tb at E:/CZFile/Project/Verilog/sortArray/tb.v line 52  
# MACRO ./auto_run_4.do PAUSED at line 17  
VSIM(paused)>
```

4. 仿真成功以后会生成 `result_4.txt`，分别是 `in0 in1 in2 in3` 输入的4个数据，`max0 max1 max2 max3` 的输出4个数据。

≡ result_4.txt

```
1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11
```

0th input data =	4	3	2	1
output data =	4	3	2	1
1th input data =	4	12	10	7
output data =	12	10	7	4
2th input data =	7	9	15	13
output data =	15	13	9	7
3th input data =	14	15	2	4
output data =	15	14	4	2