

Lab1: REG 实验

基于 Digilent Nexys3 FPGA 平台的原理图方式设计

Lab 1: 运算器实验

实验简介

本实验旨在使读者学会 Xilinx 的 ISE 工具的使用，以原理图的方式完成一个简单的单片(4 位)双端口存储器设计。

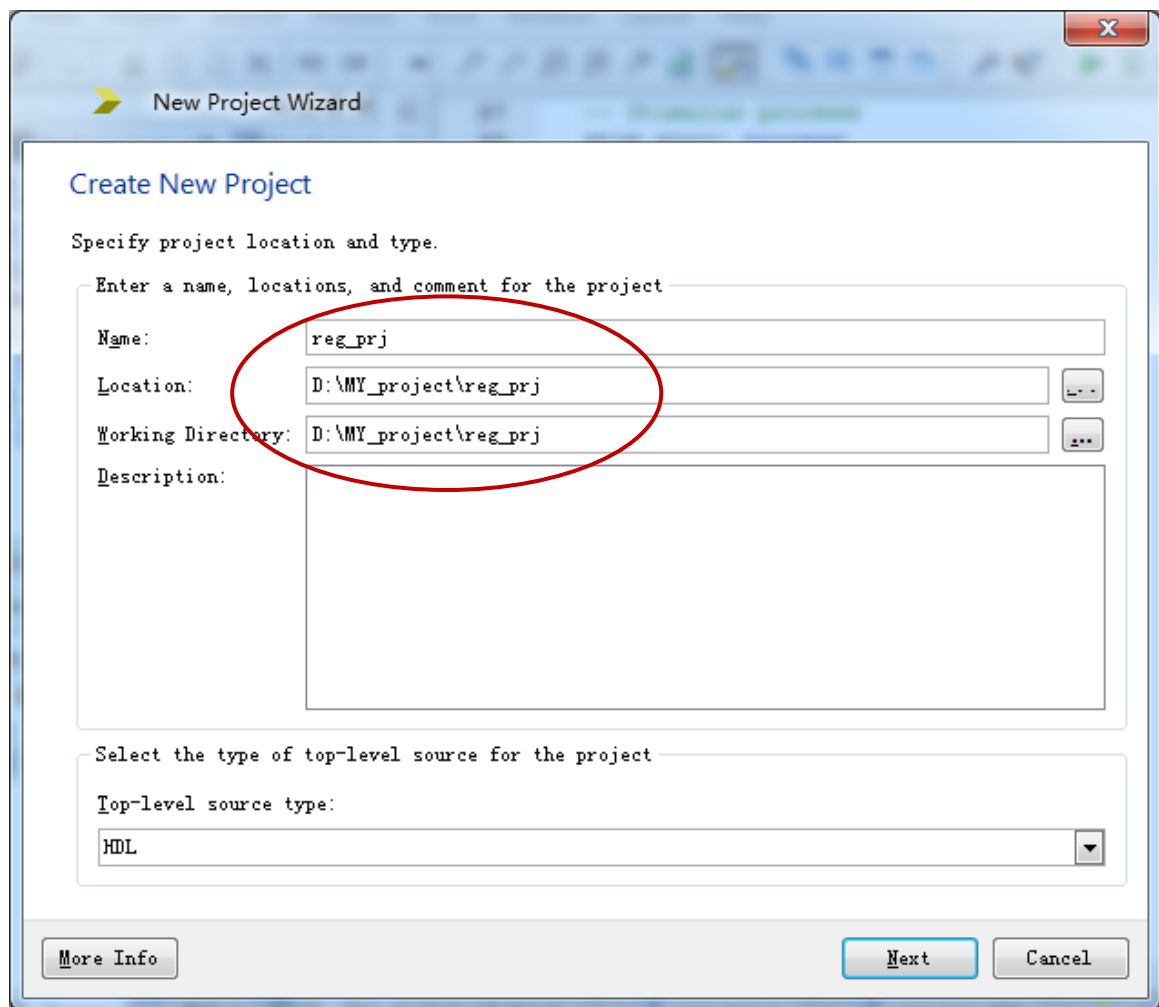


图 1：新工程建立向导

- 1-1-1. 如图 1，在新工程建立向导对话框的 **Location** 栏选择工程建立后存放的路径，存放路径不能有中文字符和空格否则会报错。
- 1-1-2. 新出现的是关于工程的一些参数设置的对话框，设置如下的参数后，点击 **Next**，如图 2。

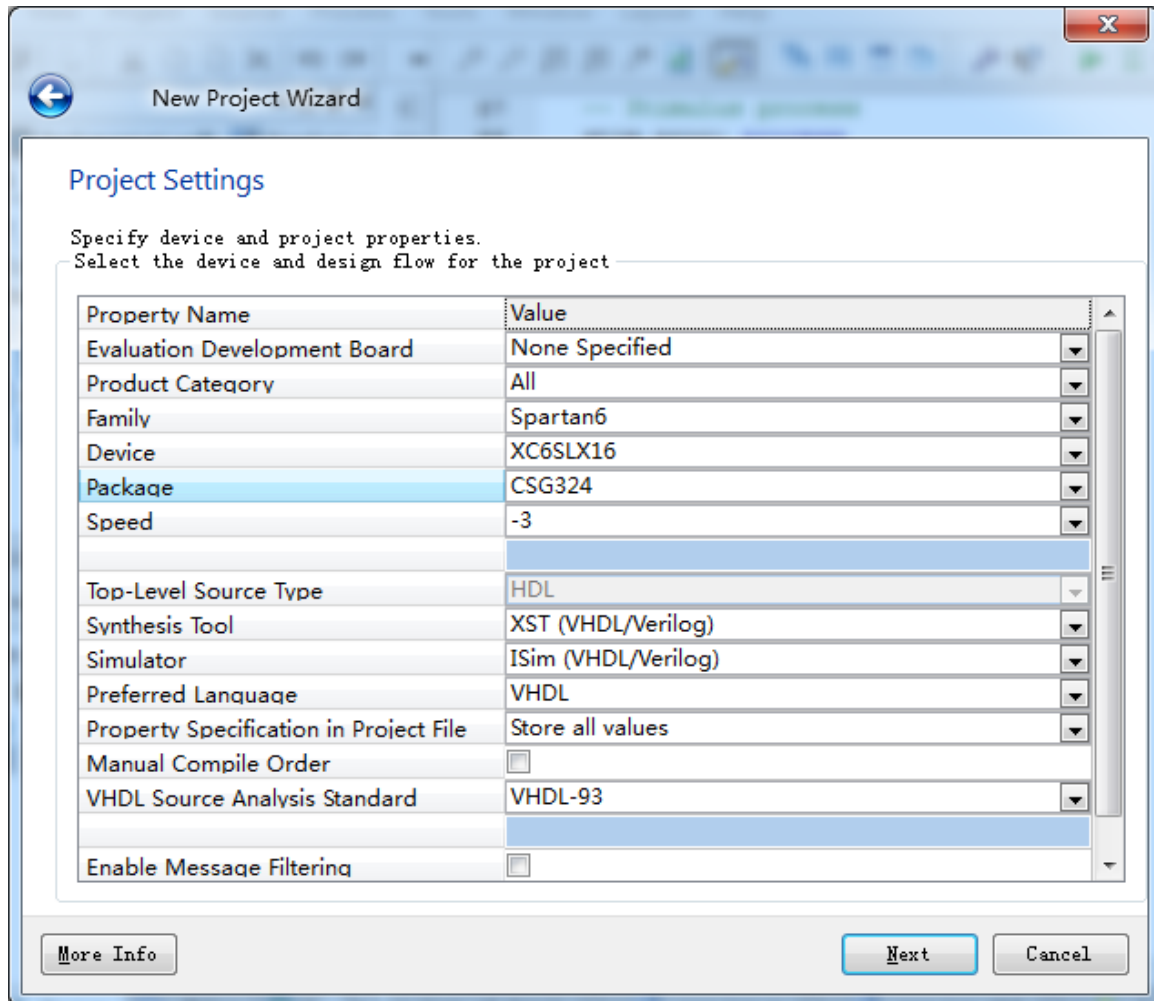


图 2：新工程参数设置

1-1-3. 点击 **Finish**。新的空白工程建立结束。

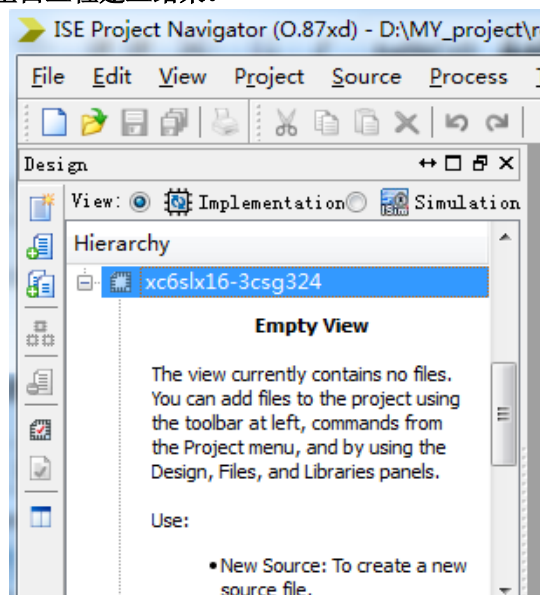


图 3：ISE 用户界面

第二步 添加源文件

2-1. 通过 ISE 工具里的 VHDL Module 模板新建一个文件。

2-1-1. 在 ISE 的用户界面中，左键单击选中 **xc6slx16-3csg324**，右键单击会出现一些选项，选择 **New Source...**，然后在 **New Source wizard(新文件向导)**对话框中选择 **VHDL Module**。在 **file name** 里取一个名字，这里可以取 **top**(名字中不要有中文和空格)，如图 4。点击 **Next**。

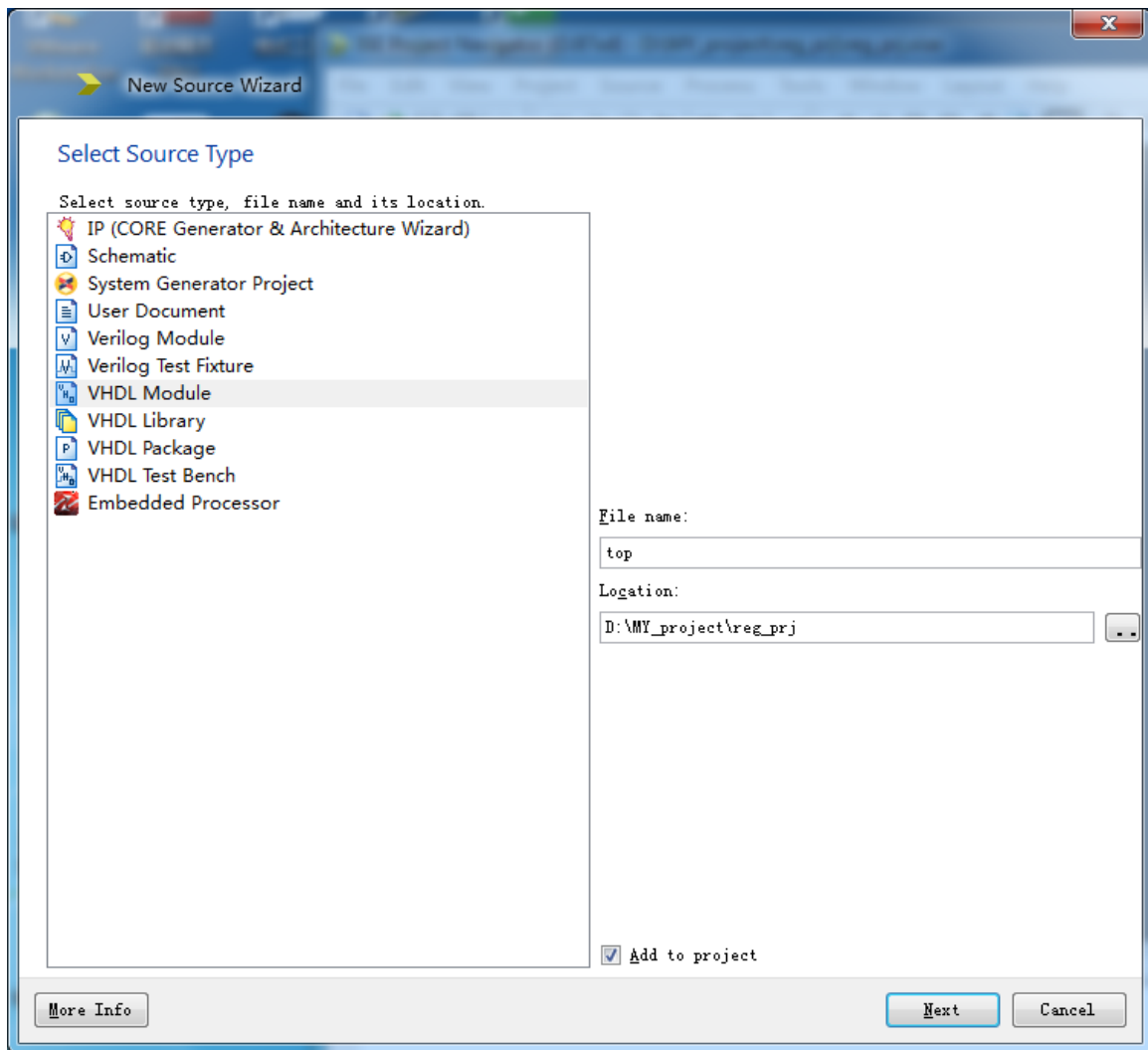
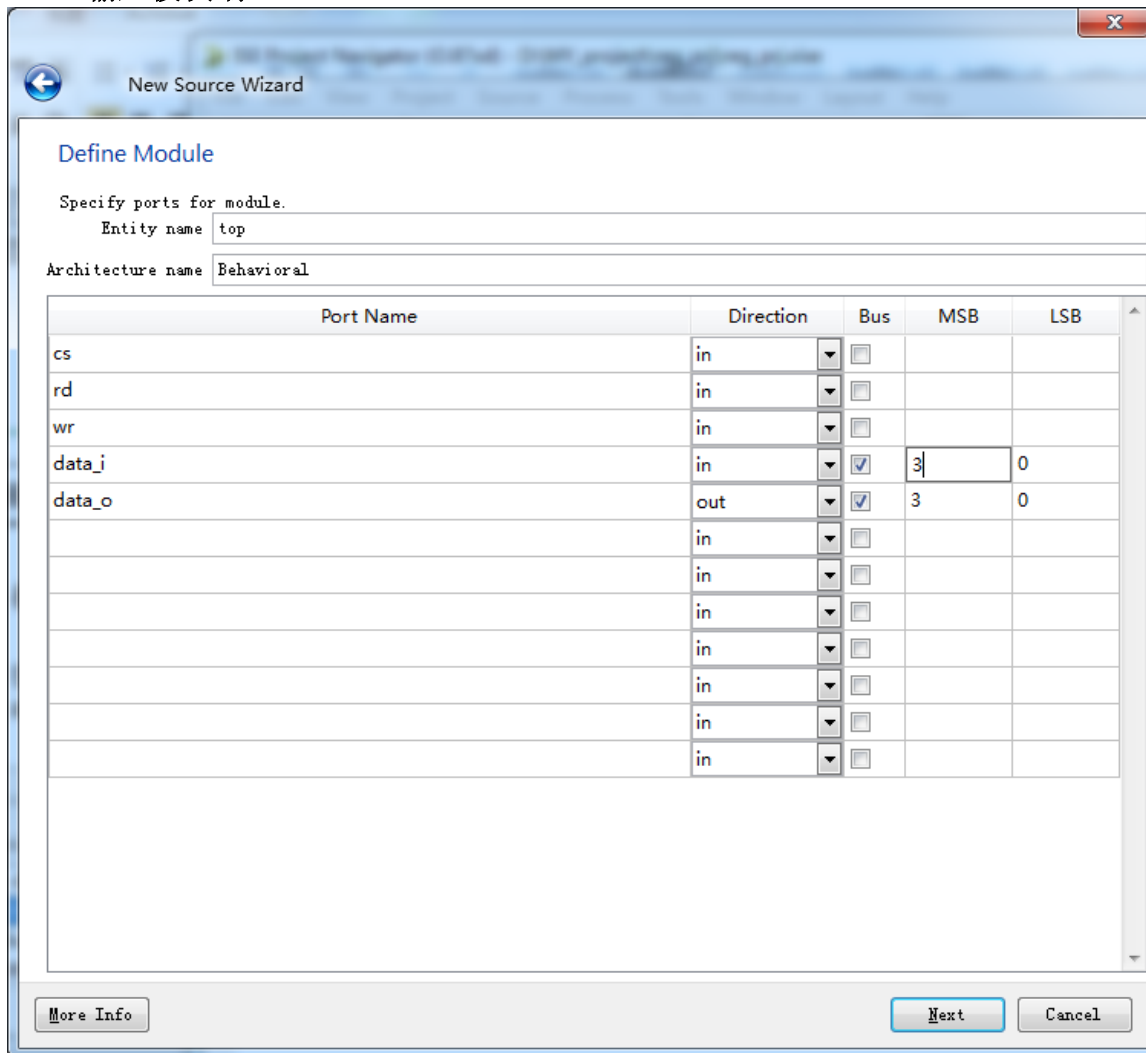


图 4：新文件向导

点击 **Finish**。

第三步 VHDL 代码输入

3-1. 输入模块端口



The image shows the "New Source Wizard" dialog box, specifically the "Define Module" step. The "Entity name" is set to "top" and the "Architecture name" is set to "Behavioral". Below this is a table for defining module ports.

Port Name	Direction	Bus	MSB	LSB
cs	in	<input type="checkbox"/>		
rd	in	<input type="checkbox"/>		
wr	in	<input type="checkbox"/>		
data_i	in	<input checked="" type="checkbox"/>	3	0
data_o	out	<input checked="" type="checkbox"/>	3	0
	in	<input type="checkbox"/>		
	in	<input type="checkbox"/>		
	in	<input type="checkbox"/>		
	in	<input type="checkbox"/>		
	in	<input type="checkbox"/>		
	in	<input type="checkbox"/>		
	in	<input type="checkbox"/>		

At the bottom of the dialog, there are three buttons: "More Info", "Next", and "Cancel".

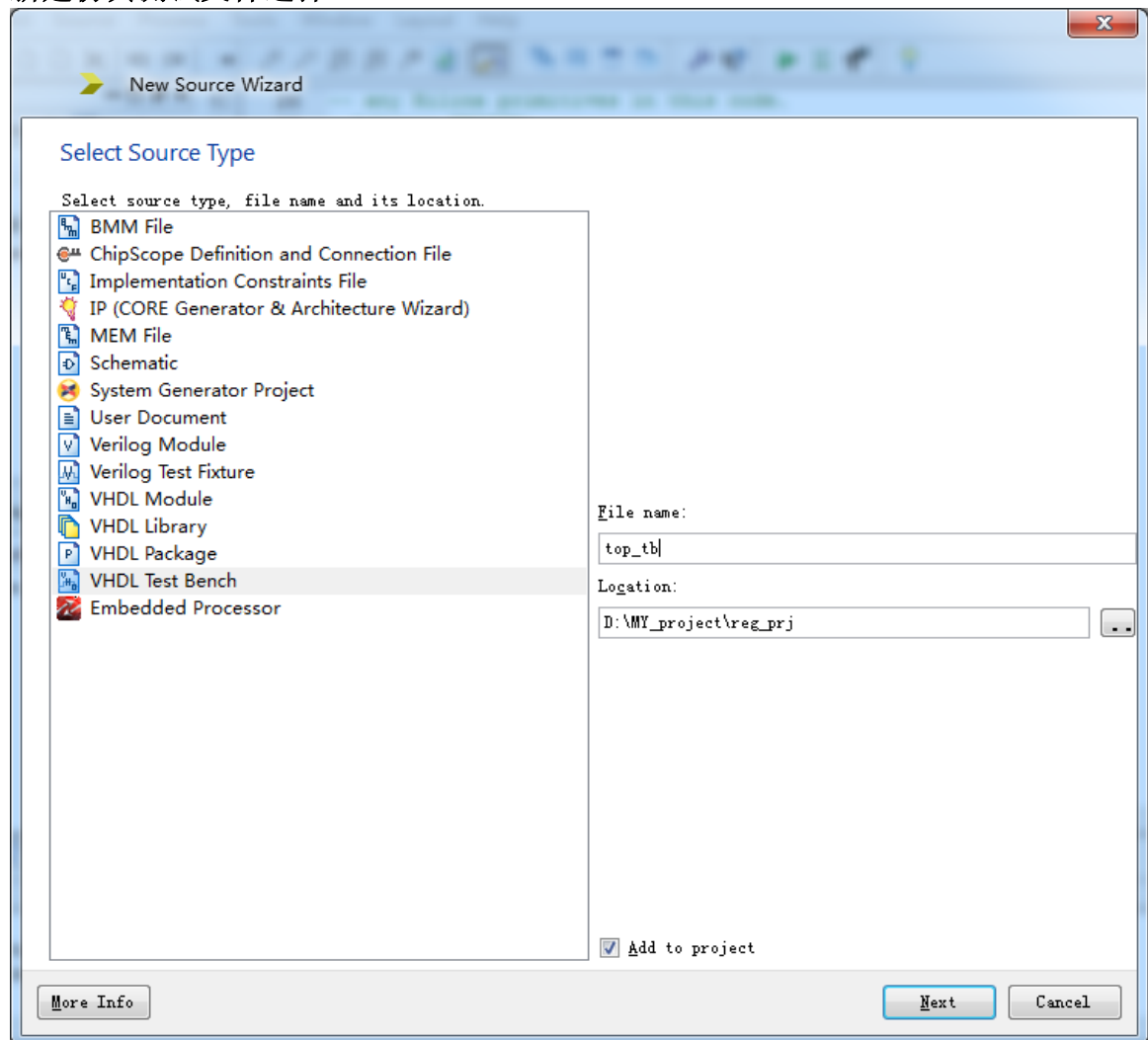
3-1-1. 工具自动生成 VHDL 的模版。

```
19 -----
20 library IEEE;
21 use IEEE.STD_LOGIC_1164.ALL;
22
23 -- Uncomment the following library declaration if using
24 -- arithmetic functions with Signed or Unsigned values
25 --use IEEE.NUMERIC_STD.ALL;
26
27 -- Uncomment the following library declaration if instantiating
28 -- any Xilinx primitives in this code.
29 --library UNISIM;
30 --use UNISIM.VComponents.all;
31
32 entity top is
33     Port ( cs : in  STD_LOGIC;
34           rd : in  STD_LOGIC;
35           wr : in  STD_LOGIC;
36           data_i : in  STD_LOGIC_VECTOR (3 downto 0);
37           data_o : out STD_LOGIC_VECTOR (3 downto 0));
38 end top;
39
40 architecture Behavioral of top is
41
42 begin
43
44
45 end Behavioral;
46
47
```

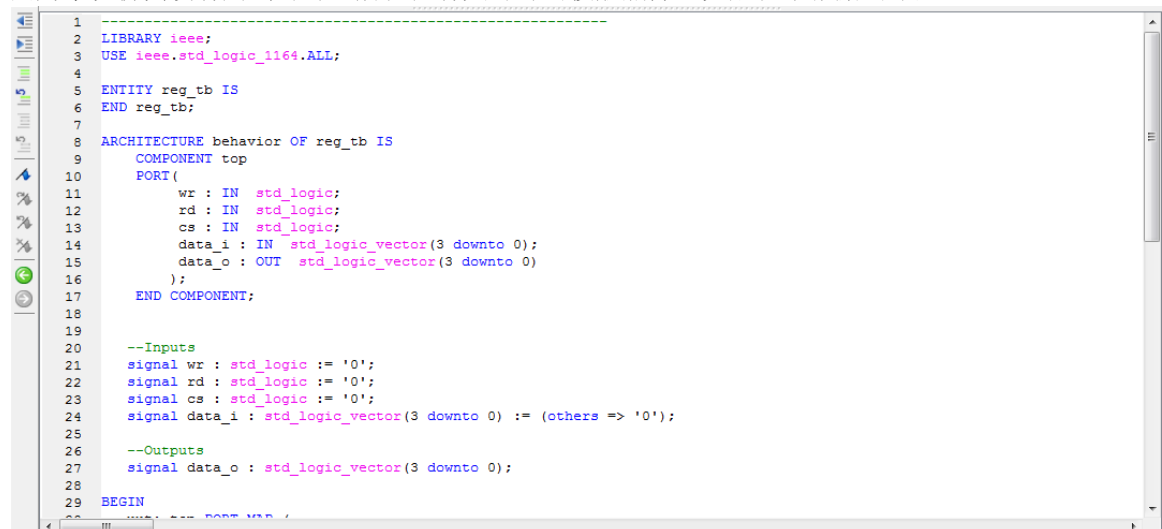
3-1-2. 完成以下代码输入

```
18 --
19 -----
20 library IEEE;
21 use IEEE.STD_LOGIC_1164.ALL;
22 entity top is
23     Port ( cs : in  STD_LOGIC;
24           rd : in  STD_LOGIC;
25           wr : in  STD_LOGIC;
26           data_i : in  STD_LOGIC_VECTOR (3 downto 0);
27           data_o : out STD_LOGIC_VECTOR (3 downto 0));
28 end top;
29
30 architecture Behavioral of top is
31     signal temp : std_logic_vector (3 downto 0);
32 begin
33     C1: process (wr ,rd , cs )
34     begin
35         if cs = '1' and wr = '1' then
36             temp <= data_i;
37         else if cs = '1' and rd = '1' then
38             data_o <= temp;
39         else
40             temp <= temp;
41         end if;
42     end if;
43     end process;
44
45 end Behavioral;
46
```

3-2. 新建仿真测试文件选择 VHDL Test Bench。



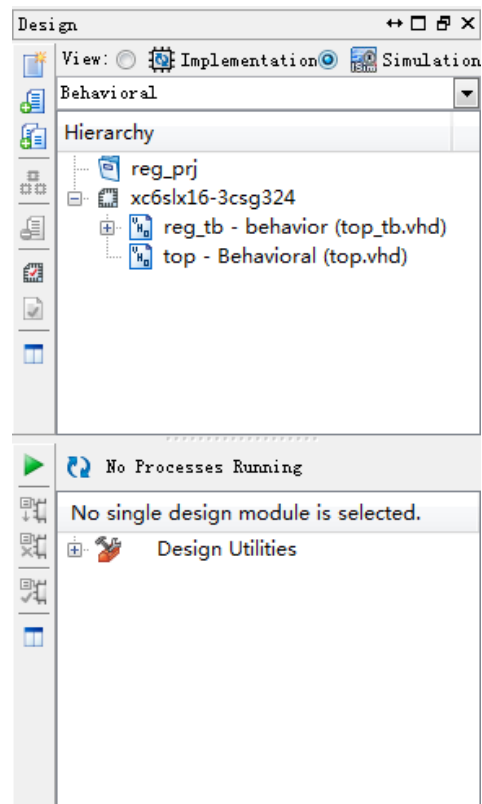
3-2-1. 因为本次仿真没有用到时钟，所以把有关时钟的模版删除，完成以下激励的添加。



```
29 BEGIN
30     uut: reg_top PORT MAP (
31         wr => wr,
32         rd => rd,
33         cs => cs,
34         data_i => data_i,
35         data_o => data_o
36     );
37
38     stim_proc: process
39     begin
40         wait for 100 ns;
41         cs <= '1';
42         wr <= '1';
43         rd <= '0';
44         data_i <= "1010";
45         wait for 100 ns;
46         cs <= '1';
47         wr <= '0';
48         rd <= '1';
49         data_i <= "0000";
50         wait for 100 ns;
51         cs <= '1';
52         wr <= '1';
53         rd <= '0';
54         data_i <= "1111";
55         wait for 100 ns;
56         cs <= '1';
57         wr <= '0';
58         rd <= '1';
59         data_i <= "0000";
60         wait;
61     end process;
62
63 END;
```

图 15:

3-2-2. 切换到仿真。



双击打开 Isim 仿真工具

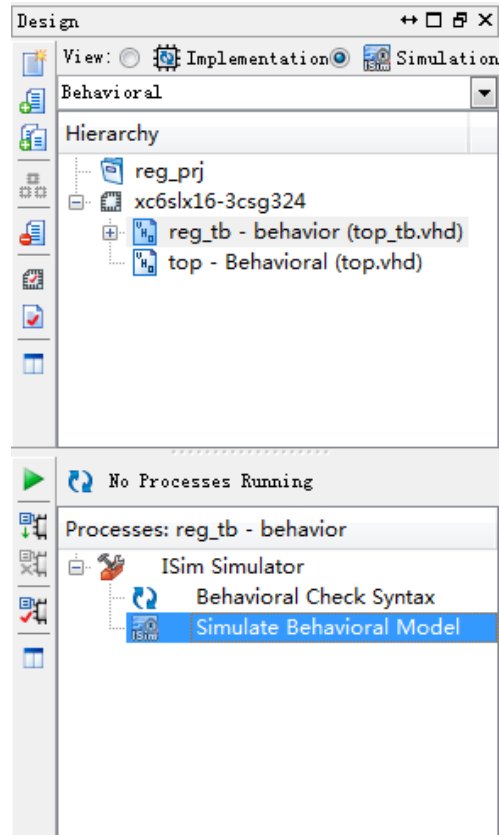


图 17:

第四步 观察仿真波形

