

# 《数字电路与数字系统实验》实验报告

第0次实验：Quartus 使用初步—双控开关实验

姓名：张逸凯

学号：171840708

院系：物理学院

邮箱：645064582@qq.com

电话：18051988316

## 一.实验目的

以 Quartus18.0 为例，熟悉如何利用 FPGA 开发平台 DE10-Standard 进行数字逻辑设计。课后参考相关资料，在自己的电脑上安装 Lite 版的开发软件套装。

复习数字电路课程中简化表达式等相关知识，从 Truth Table 推 数字逻辑图，前面小实验就要好好做仿真，学会在 Verilog 语言下设计一个双控开关。

## 二.实验原理

双控开关是一个开关同时带常开、常闭两个触点（即为一对）。通常用两个双控开关控制一个灯或其它电器，意思就是可以有两个开关来控制灯具等电器的开关，比如，在楼下时打开开关，到楼上后关闭开关。

很简单可以直接看出表达式：

$$F = A \oplus B = \bar{A}B + A\bar{B}$$

其中  $A$  和  $B$  是开关的输入， $F$  是灯的输出。

真值表：

A	B	F
0	0	0
0	1	1
1	1	0
1	0	1

## 三.实验设备环境

硬件器材：FPGA 开发板。

软件平台：Qaurtus 开发平台。

## 四.实验步骤 / 过程（设计思路、设计代码、测试代码、仿真结果和硬件实现等的截图代码等）

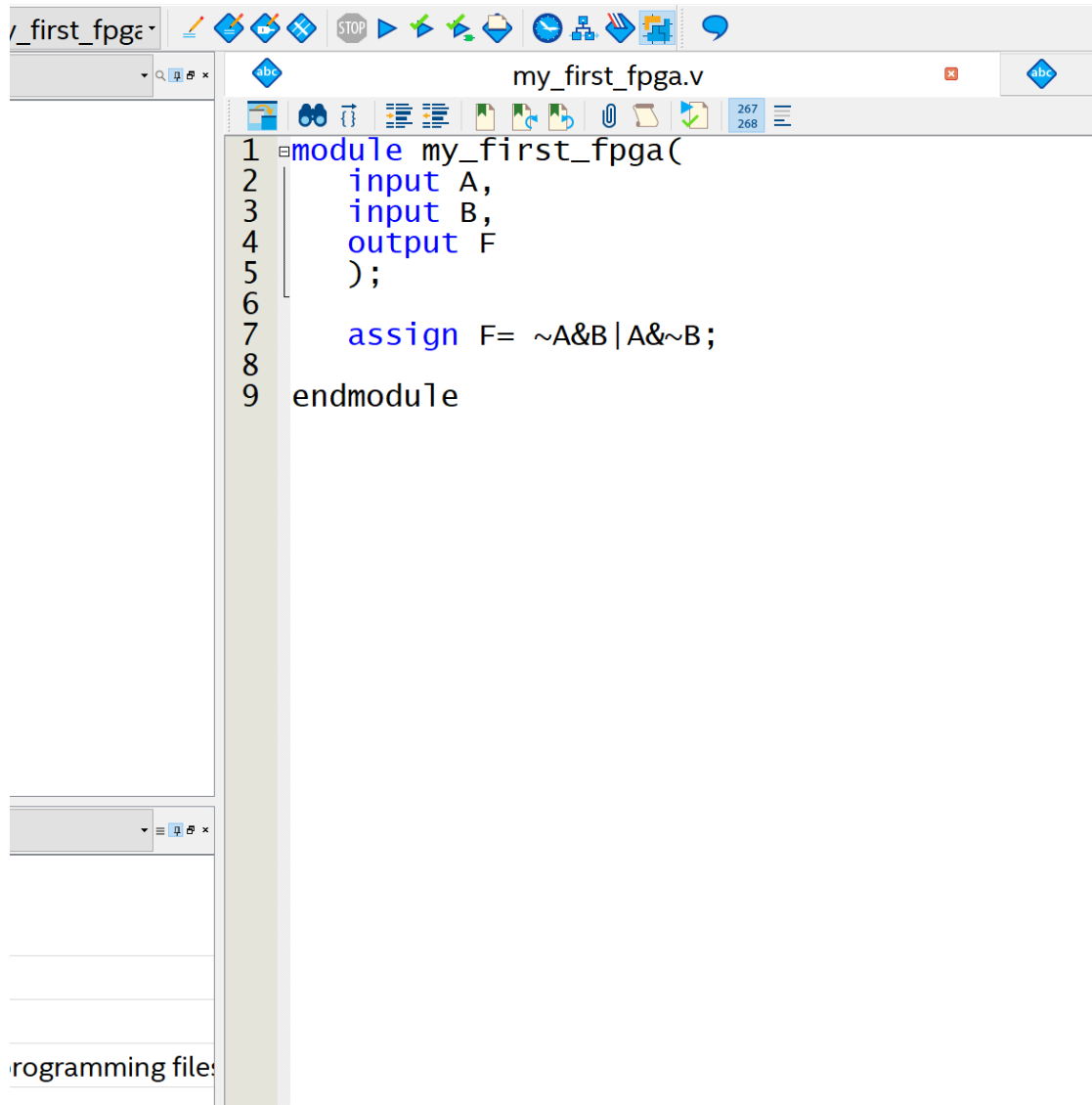
设计思路：

经过几天的对 Verilog 语言的学习，加上老师在第一次实验给的 Tutorial，从最基础的开始，注意每一步的细节即可。

设计代码:

- E:/My\_design/my\_first\_fpga/my\_first\_fpga - my\_first\_fpga

ments Processing Tools Window Help



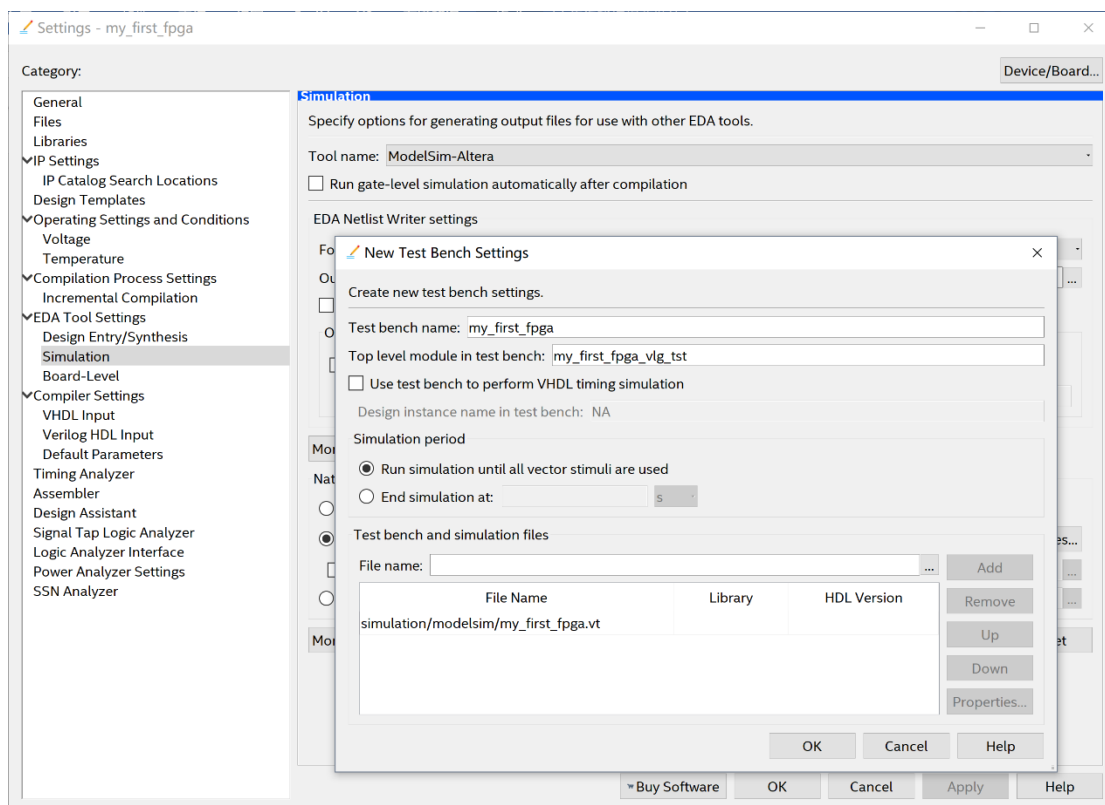
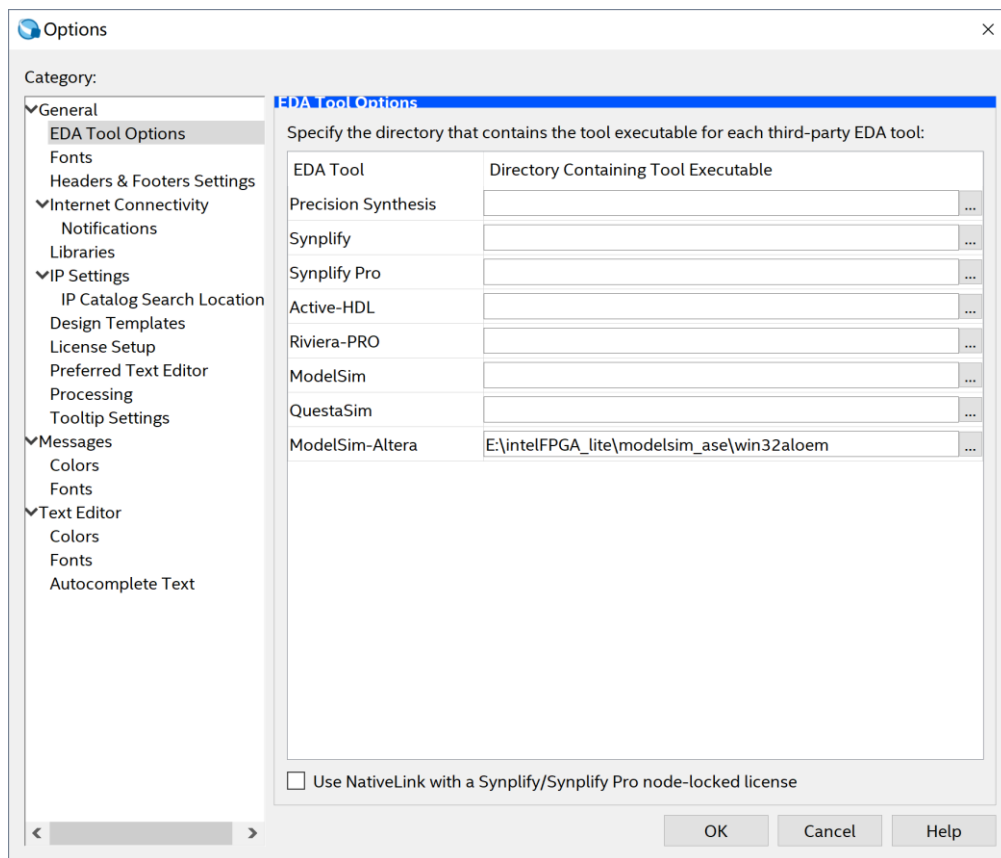
The screenshot shows a software interface for FPGA development. The main window displays a Verilog code file named 'my\_first\_fpga.v'. The code defines a module 'my\_first\_fpga' with two inputs, 'A' and 'B', and one output, 'F'. The logic implemented is the XOR operation:  $F = \sim A \& B \mid A \& \sim B$ . The code is as follows:

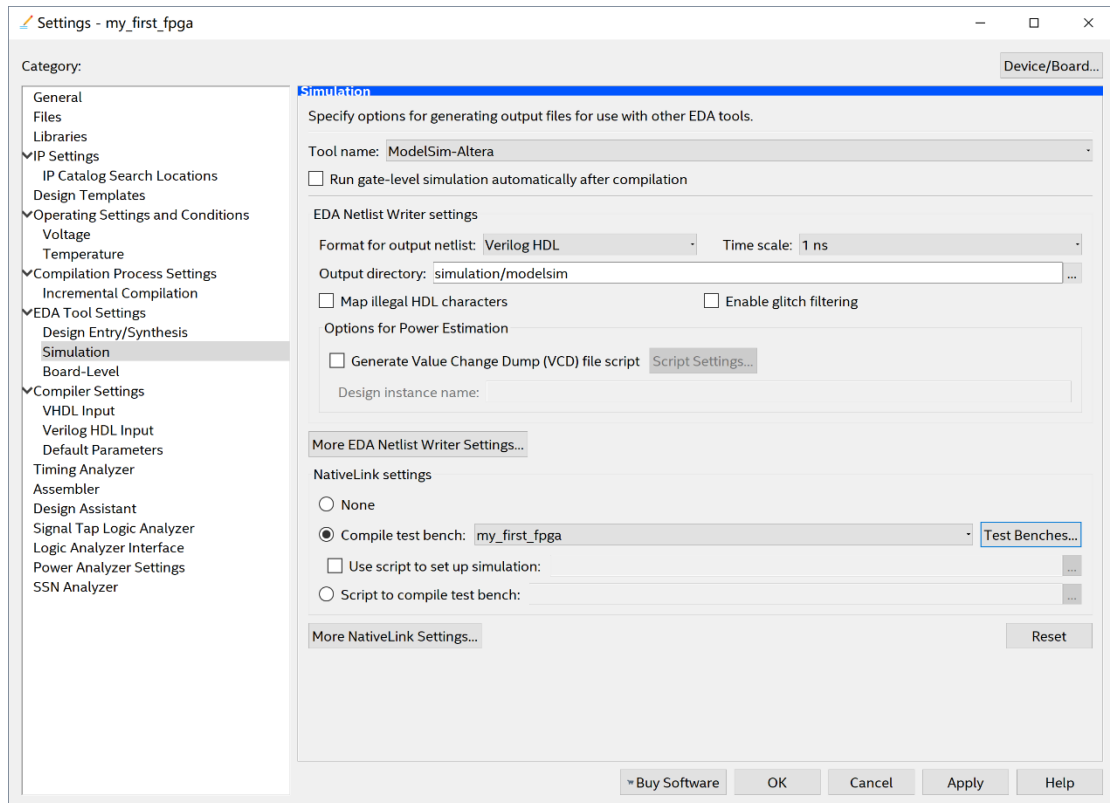
```
1 module my_first_fpga(  
2     input A,  
3     input B,  
4     output F  
5 );  
6  
7     assign F= ~A&B|A&~B;  
8  
9 endmodule
```

At the bottom of the window, there is a label 'programming file:'.

图1 设计代码

## 建立工程操作过程细节图:





## 测试代码:

- E:/My\_design/my\_first\_fpga/my\_first\_fpga - my\_first\_fpga

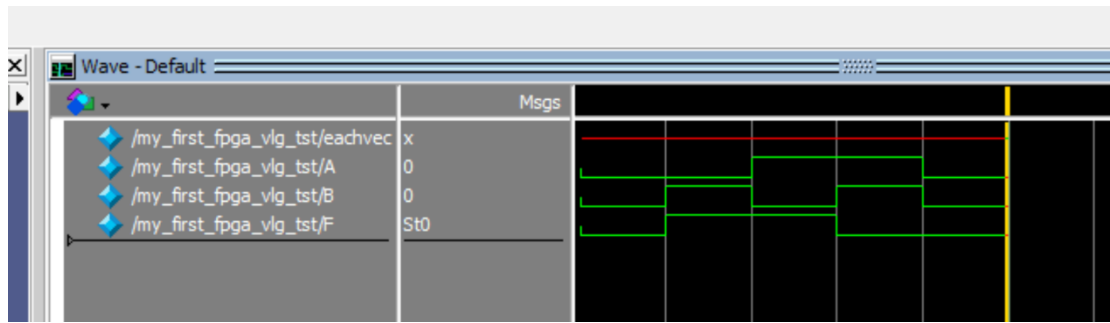
ments Processing Tools Window Help

```

my_first_fpga.vt*
my_first_fpga.v
32 // test vector input registers
33 reg A;
34 reg B;
35 // wires
36 wire F;
37
38 // assign statements (if any)
39 my_first_fpga i1 (
40 // port map - connection between master ports and signals/registers
41 .A(A),
42 .B(B),
43 .F(F)
44 );
45 initial
46 begin
47 // code that executes only once
48 // insert code here --> begin
49 A = 1'b0; B = 1'b0; #20;
50 A = 1'b0; B = 1'b1; #20;
51 A = 1'b1; B = 1'b0; #20;
52 A = 1'b1; B = 1'b1; #20;
53 A = 1'b0; B = 1'b0; #20;
54
55 // --> end
56 $display("Running testbench");
57 end
58 always
59 // optional sensitivity list
60 // @(event1 or event2 or .... eventn)
61 begin
62 // code executes for every event on sensitivity list
63 // insert code here --> begin
64
65 @eachvec;
66 // --> end
67 end
68 endmodule
69
70

```

## ModelSim 仿真波形:

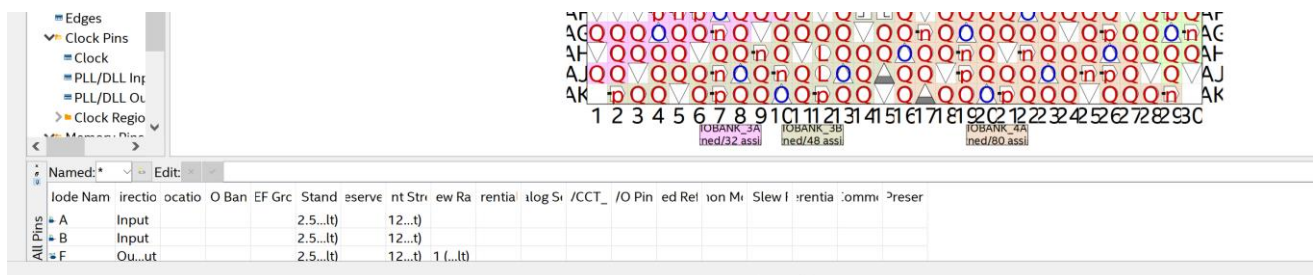


### 注意点:

- 仿真波形图是可以看它哪里错的东西
- 可以添加中间输出.
- 调试波形很重要, 测试文件的设计很重要

## 硬件实现:

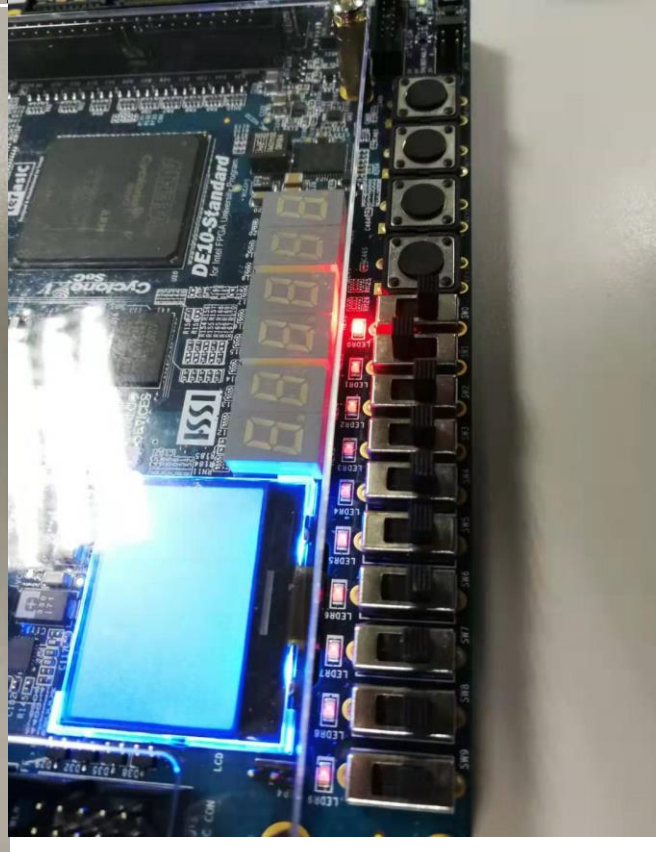
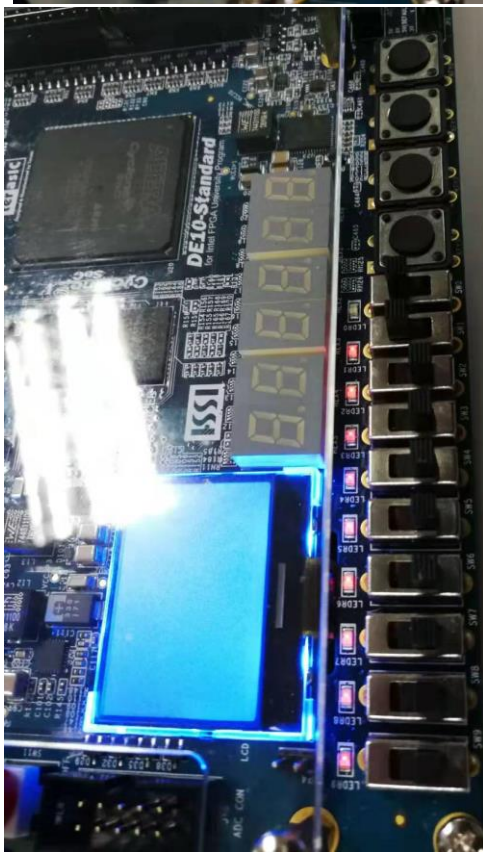
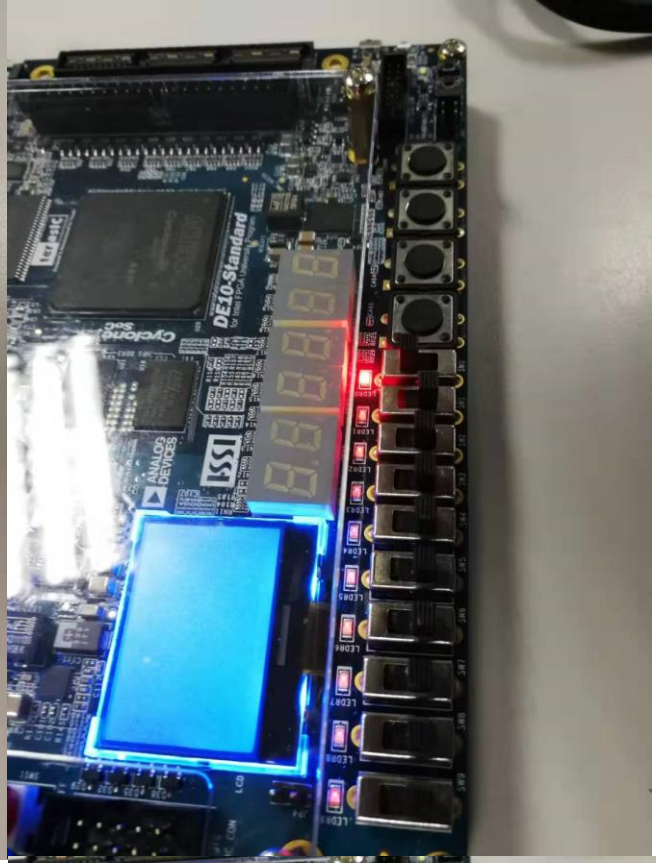
### 引脚分配:



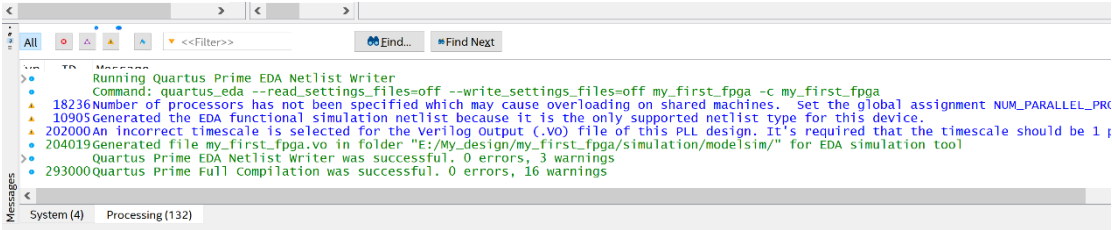
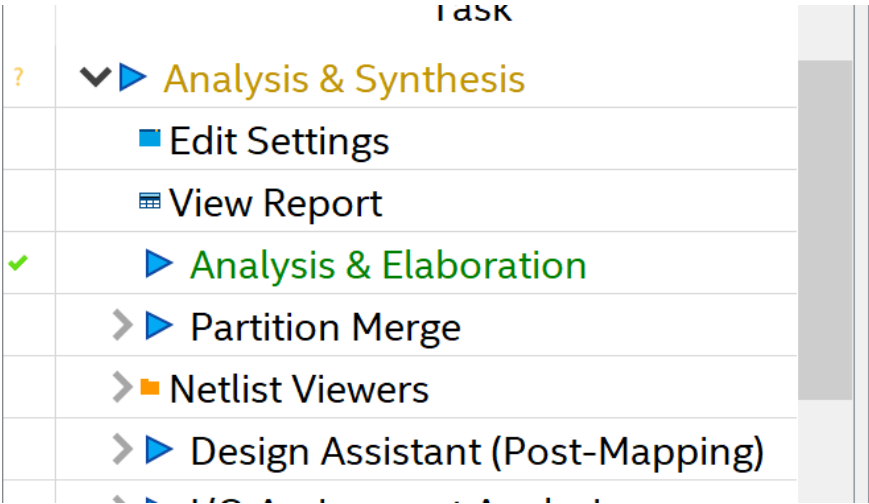
### 下面四幅图中:

- 左上是两个开关都为 0 的状态, 左下是两个开关都为 1 的状态
- 右上和右下是两个开关一个为 0, 一个为 1 的状态.



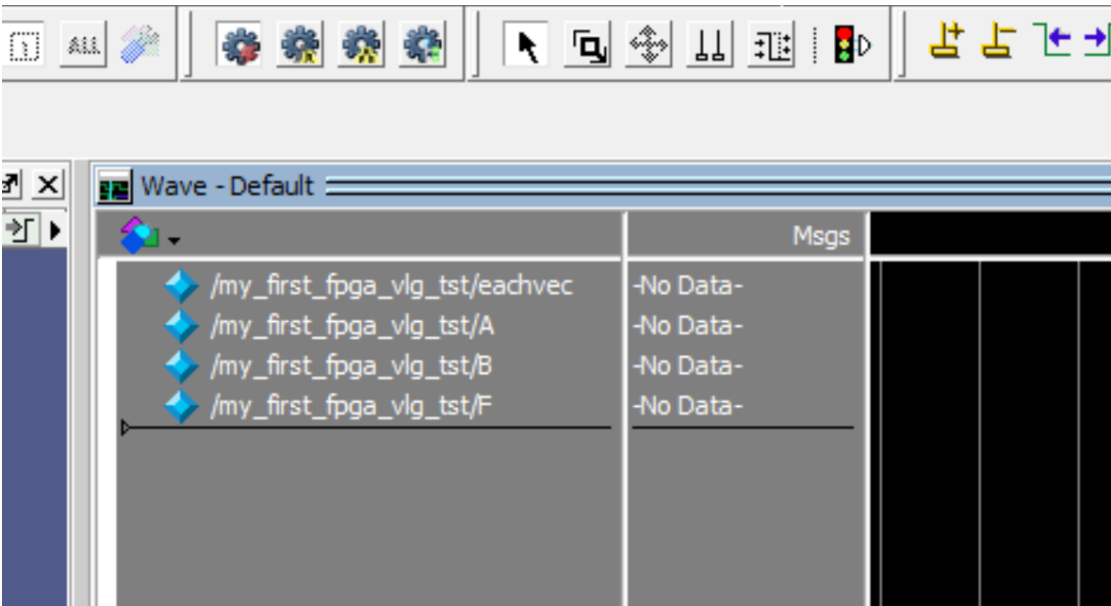


部分成功截图:



## 五.实验中遇到的问题及解决方案（请具体的描述问题和解决方法）

在模拟波形时，出现了模拟文件为空的情况





### 可能情况排查:

1. test bench 没有成功加入, 导致读不到测试文件.
2. 测试代码有问题, 测试代码没有起到测试的作用.
3. 测试文件没有保存

反向排查之后发现是测试文件改过之后没有保存, 保存以后重新添加仿真测试结果就正确了.

## 六.实验得到的启示（请大家积极思考）

进行每一步操作的时候需要细心, 这样能减少程序的错误, 节省很多不必要的时间, 同时出错以后要细心检查反向排查, 这样才能快速定位错误. 同时还要能逐渐熟练 Quartus 软件的使用, 比如某个地方出错以后能够利用此软件的提示信息较快的找出错误源, 只有这样做实验和查错才会更加迅速.

## 七.意见和建议等

老师给的实验手册已经讲的很详细了, 我会跟着老师的步伐, 一步一步提高技能水平.