

数字电路与逻辑设计

课程实验报告

|  |  |
| --- | --- |
| 题目： | Quartus II软件的基本操作、译码器 |
| 学生姓名： | 魏子铖 |
| 学生学号： | 201726010308 |
| 专业班级： | 软件1703 |
| 完成时间： | 2018.11.2 |

**实验一 Quartus II软件的基本操作、译码器**

班级 软件1703 姓名 魏子铖 学号 201706010308

1. **实验目的**

熟悉Quartus II软件的基本操作，了解各种设计方法（原理图设计、文本设计、波形设计）

1. **实验内容**
2. 用逻辑图和VHDL语言设计一个异或门。
3. 用逻辑图和VHDL语言设计一个3-8译码器。
4. 用VHDL语言设计模型机指令译码器。
5. **实验方法**
6. **实验方法**

采用基于FPGA进行数字逻辑电路设计的方法。

采用的软件工具是Quartus II软件仿真平台。

1. **实验步骤**
2. **新建工程文件**

* 异或门的VHDL工程文件名为myxor1
* 异或门的逻辑图工程文件名为myxor2
* 3-8译码器的VHDL工程文件名为mydecoder3\_8\_1
* 3-8译码器的逻辑图工程文件名为mydecoder3\_8\_2
* 指令译码器的VHDL工程文件名为command\_decoder

1. **编写源代码或画逻辑图**

完成后保存文件，文件名分别为myxor1.vhd, myxor2.bdf, mydecoder3\_8\_1.vhd, mydecoder3\_8\_2.bdf, command\_decoder.vhd，分别位于五个不同的工程文件下。

1. **编译与调试**

确定文件名与工程文件名相同，点击【processing】-【start compilation】进行文件编译，文件编译成功。

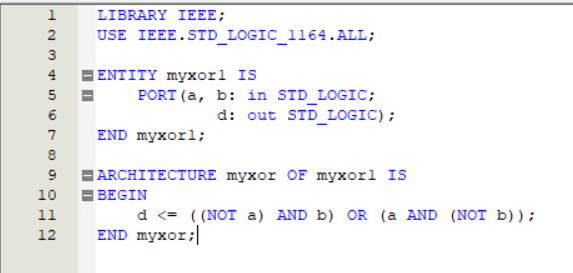
1. **波形仿真及验证**

为每个工程新建一个vector waveform file。

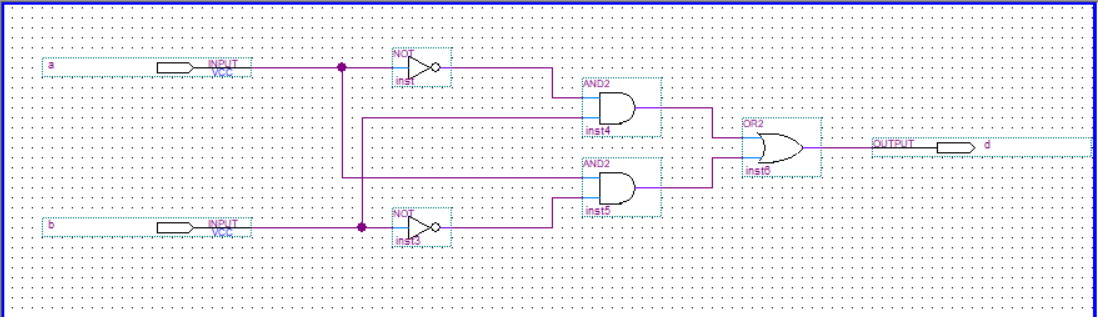
按照程序所述插入所有输入和输出节点。任意设置输入波形，点击保存按钮保存，然后【start simulation】，得到波形仿真图。

1. **查看RTL Viewer：**【Tools】-【netlist viewer】-【RTL viewer】。
2. **实验过程**
3. **编译过程**

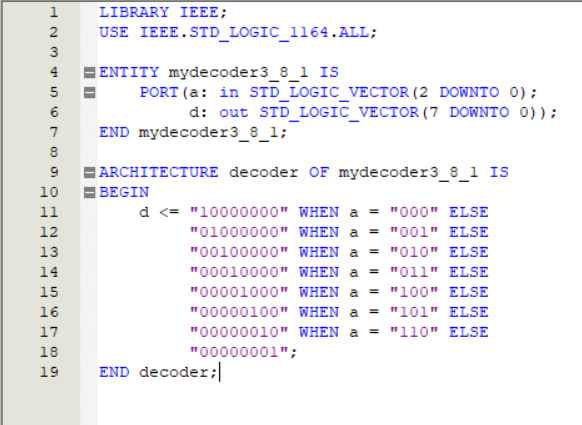
**异或门源代码：**



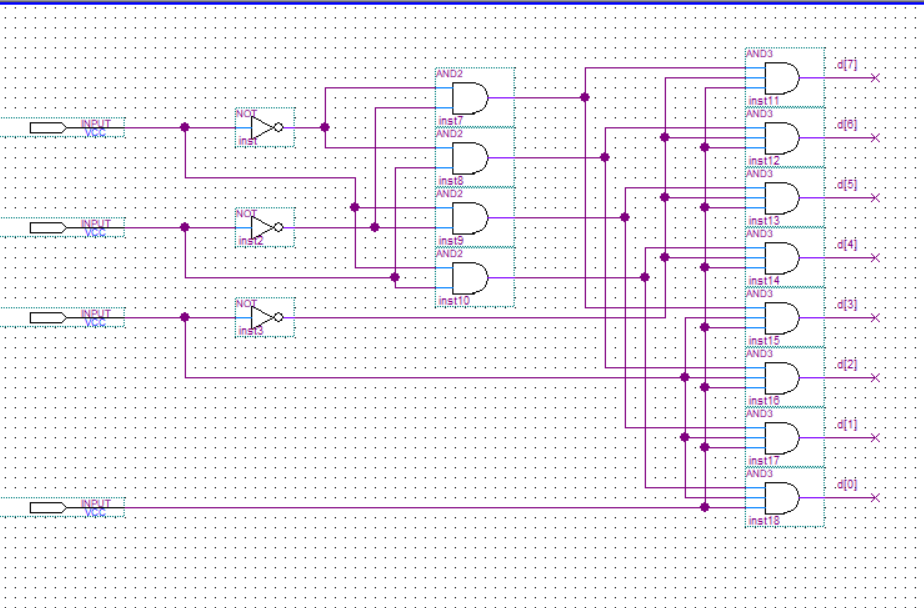
**异或门的逻辑图：**



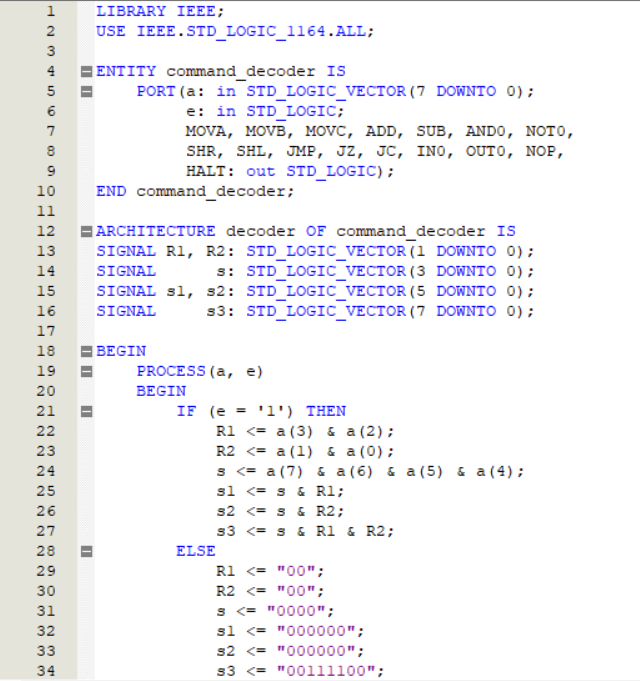
**3-8译码器源代码：**

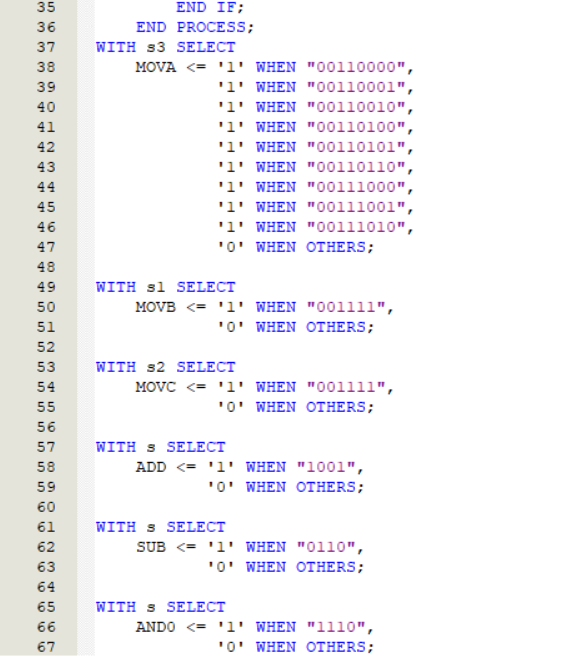


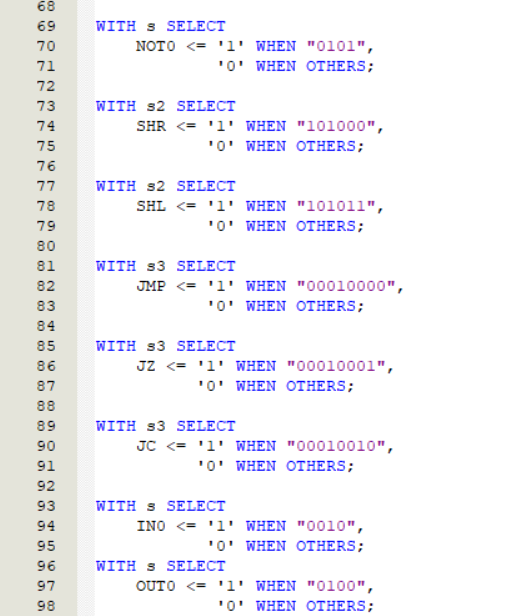
**3-8译码器逻辑图**

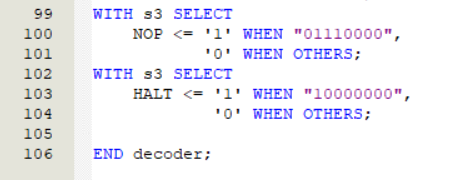


**指令译码器源代码：**





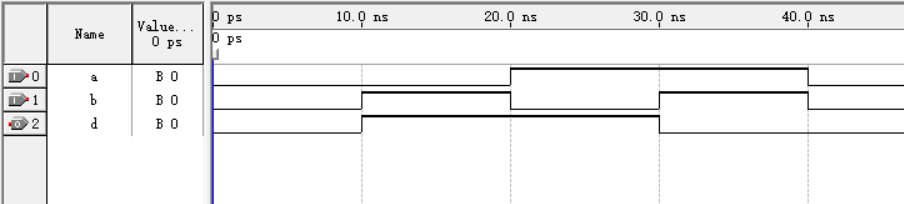




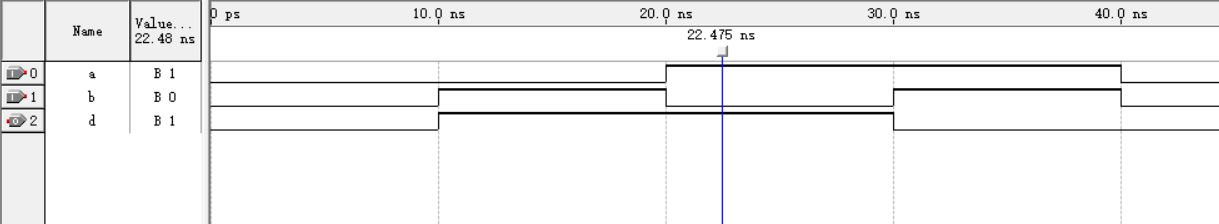
**编译、调试过程：编写完成后进行全编译，编译成功，没有出现错误。**

**结果分析及结论：源代码及原理图没有出现语法或结构上的错误。**

1. **波形仿真**
2. **异或门仿真图：**



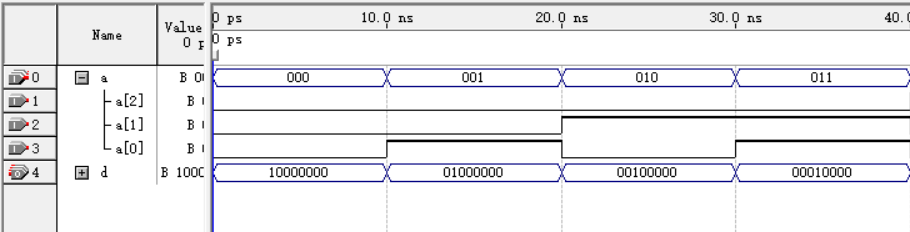
（VHDL）



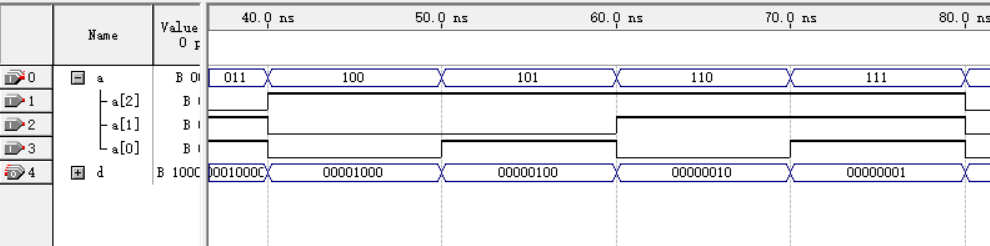
逻辑图

**结果分析及结论：由异或运算的定义知，当输入的a, b相同时，输出d为0，否则，输出d为1。由波形仿真图可以明确地看出结果时正确的。**

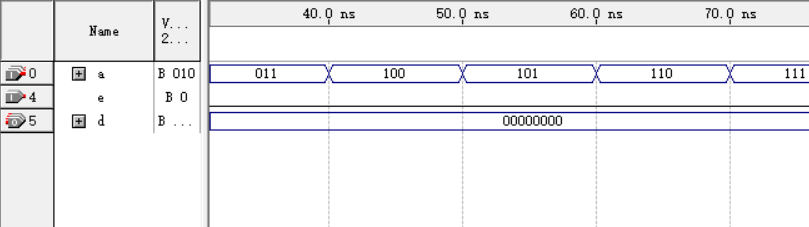
1. **3-8译码器仿真图**



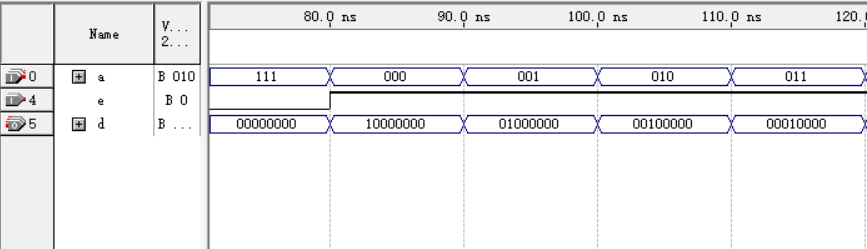
（VHDL）



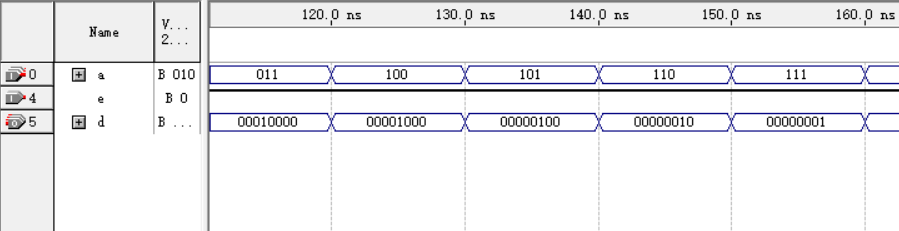
（VHDL）



逻辑图



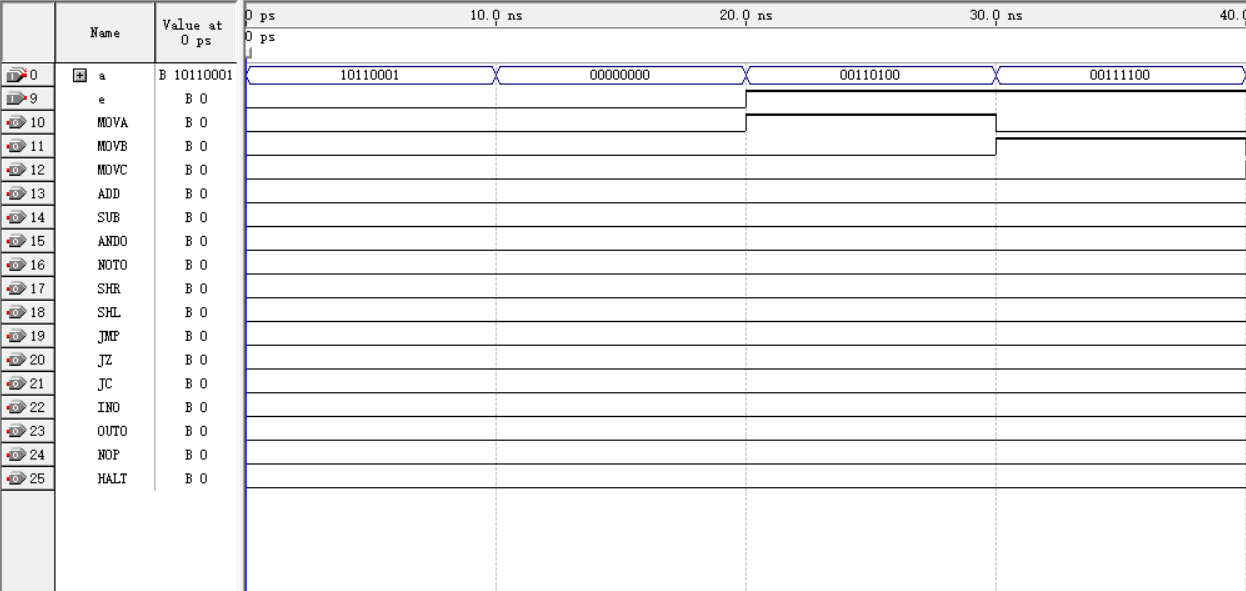
逻辑图

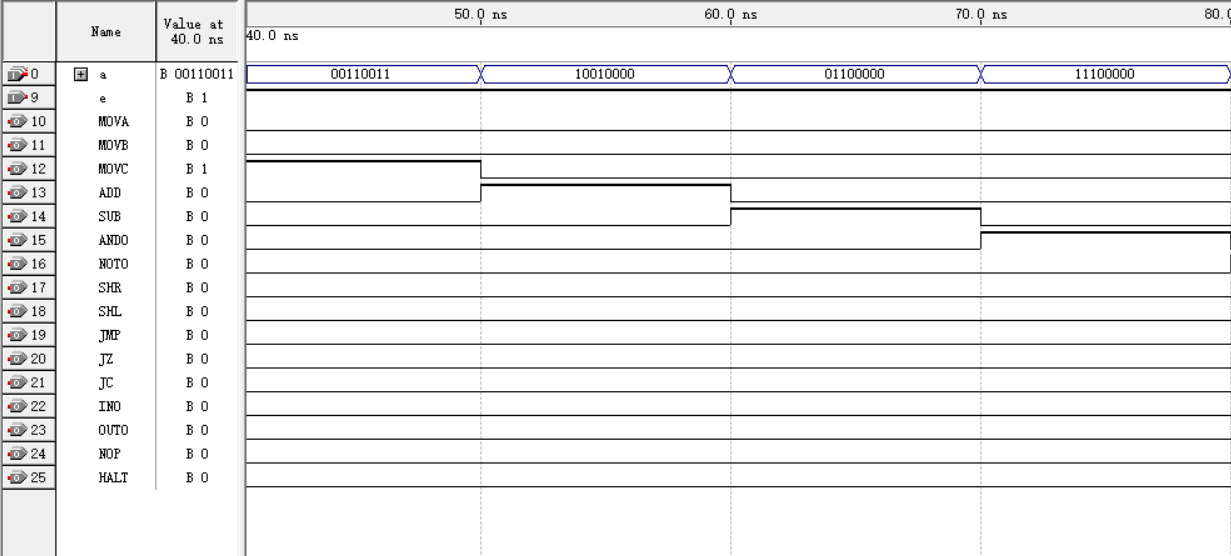


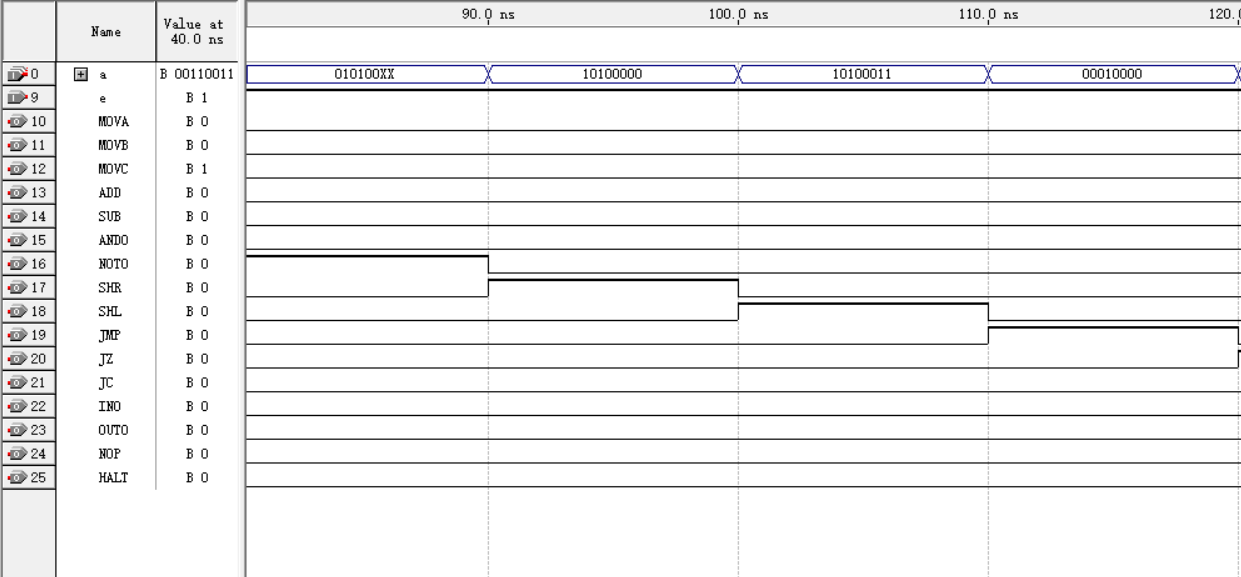
逻辑图

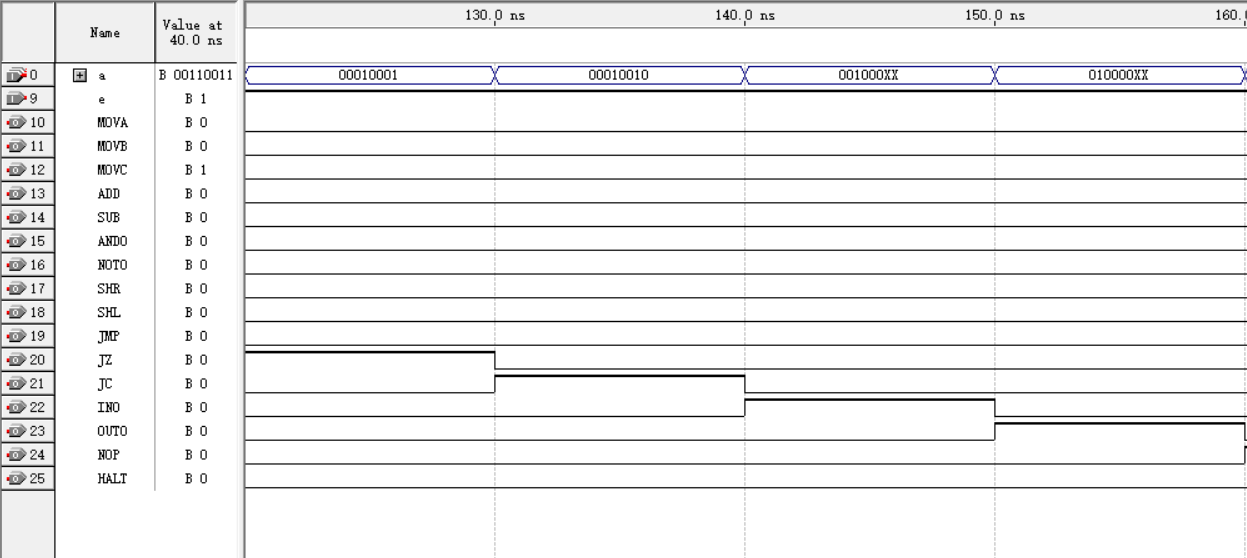
**结果分析及结论：38译码器是指将3位2进制数通过电路转换成八路不同状态的输出，输入000时输出10000000，输入001时输出01000000……在逻辑图的实现时加入了使能信号e，当e为低电平时电路失效，e为高电平时正常工作。由波形仿真图可以看出结果是正确的。**

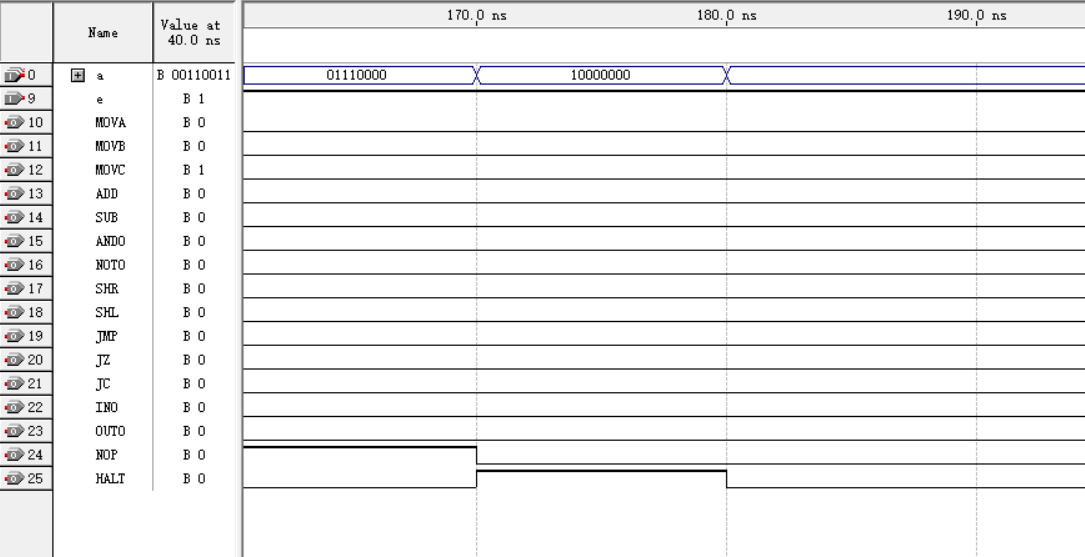
1. **指令译码器仿真图**











**结果分析及结论：**



根据输入的指令的二进制编码，将对应的功能输出高电平，对比波形仿真图可以发现结果是正确的。

1. **实验结论**

本次实验通过对指令译码器的编程，使我更加深刻的了解了计算机内部的执行操作的顺序，首先，用户输入指令代码，然后指令译码器转换成为机器能够识别的机器码，再转交给指令执行部件去执行，指令译码器将复杂的汇编代码简化成为了简单的二进制代码，从而大大增加了cpu的执行能力与运算速度。