

**本科实验报告**

|  |  |
| --- | --- |
| 课程名称： | 计算机组成 |
| 姓 名： | 应周骏 |
| 学 院： | 计算机科学与技术学院 |
| 系： | 计算机科学与技术系 |
| 专 业： | 计算机科学与技术 |
| 学 号： | 3200103894 |
| 指导教师： | 杨莹春 |

2021 年 11月 3 日

**浙江大学实验报告**

课程名称： 数字逻辑设计 实验类型： 综合

实验项目名称： 多路选择器设计及应用

学生姓名： 应周骏 专业：计算机科学与技术 学号：3200103894

同组学生姓名： 无 指导老师： 杨莹春

实验地点： 东4-509实验日期：2021 年 11 月 3 日

**一、实验目的和要求**

1. 掌握数据选择器的工作原理和逻辑功能；

2. 掌握数据选择器的使用方法；

3. 掌握四位数码管扫描显示方法；

4. 四位数码管显示应用—记分板设计；

**二、实验内容和原理**

**内容：**

1. 数据选择器设计；

2. 记分板设计；

**原理：**

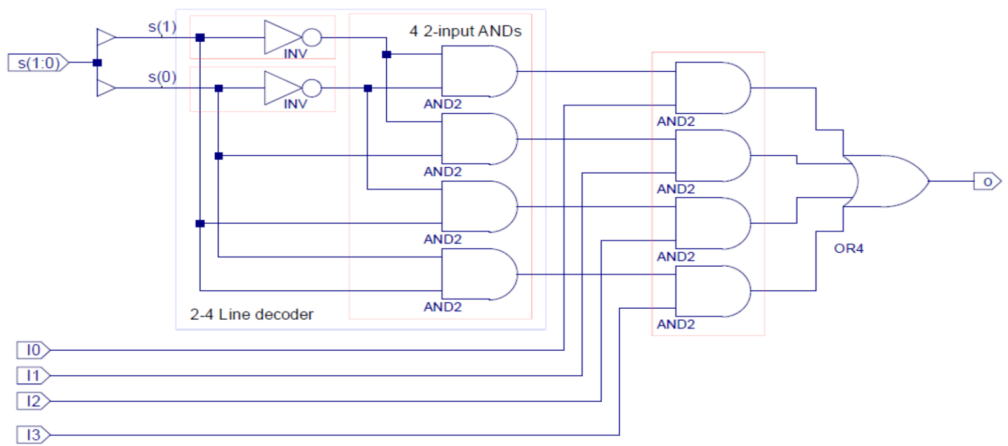
**1. 四选一多路选择器：MUX4to1**

多路选择器是[数据选择器](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E9%80%89%E6%8B%A9%E5%99%A8/4882827" \t "https://baike.baidu.com/item/%E5%A4%9A%E8%B7%AF%E9%80%89%E6%8B%A9%E5%99%A8/_blank)的别称。在多路[数据传送](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E4%BC%A0%E9%80%81/500685" \t "https://baike.baidu.com/item/%E5%A4%9A%E8%B7%AF%E9%80%89%E6%8B%A9%E5%99%A8/_blank)过程中，能够根据需要将其中任意一路选出来的电路，叫做[数据选择器](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E9%80%89%E6%8B%A9%E5%99%A8/4882827" \t "https://baike.baidu.com/item/%E5%A4%9A%E8%B7%AF%E9%80%89%E6%8B%A9%E5%99%A8/_blank)，也称多路选择器或[多路开关](https://baike.baidu.com/item/%E5%A4%9A%E8%B7%AF%E5%BC%80%E5%85%B3/4882878" \t "https://baike.baidu.com/item/%E5%A4%9A%E8%B7%AF%E9%80%89%E6%8B%A9%E5%99%A8/_blank)。

四选一多路选择器将通过控制端S1S0的变化，选择输入I0~I3中的其中一路，输出是控制信号全部最小项与或结构，其真值表如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 信息输入 | 控制端 | 选择输出 | |
| I0 I1 I2 I3 | S1   S0 | o       输出项 | |
| I0 I1 I2 I3 | 0      0 | I0 | I0 |
| I0 I1 I2 I3 | 0      1 | I1 | I1 |
| I0 I1 I2 I3 | 1      0 | I2 | I2 |
| I0 I1 I2 I3 | 1       1 | I3 | I3 |

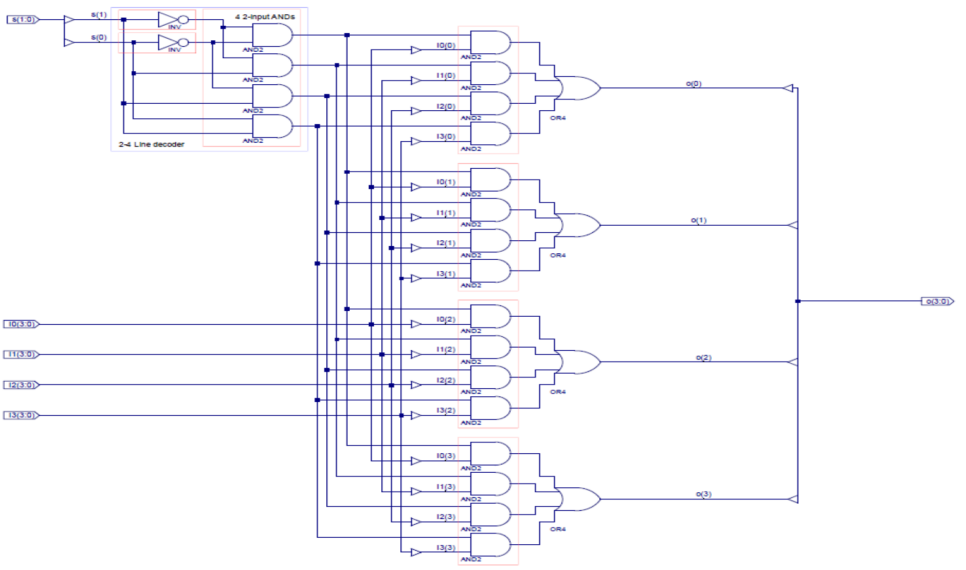
MUX4to1的原理图如下：



**2.位扩展：**

对多路选择器进行位扩展，则可在控制结构不变的情况下，每路输入向量化。即在每一路共享S1S0变量译码器的情况下，对后面的AND-OR通道进行多位复制。例如，对于四位四选一多路选择器MUX4to1b4，可对Mux4to1进行如图所示的拓展：

保留变量译码器



位扩展

**3.记分板的逻辑功能:**

自增控制：用BTNX4Y3～BTNX4Y0这4个按钮，每个按钮按下一次，对应的数码管的值加1。

小数点控制：用SW0～SW3这4个开关控制每个数码管的小数点。

消隐控制：用SW4～SW7这4个开关控制每个数码管的消隐。

**4.动态扫描显示方案:**

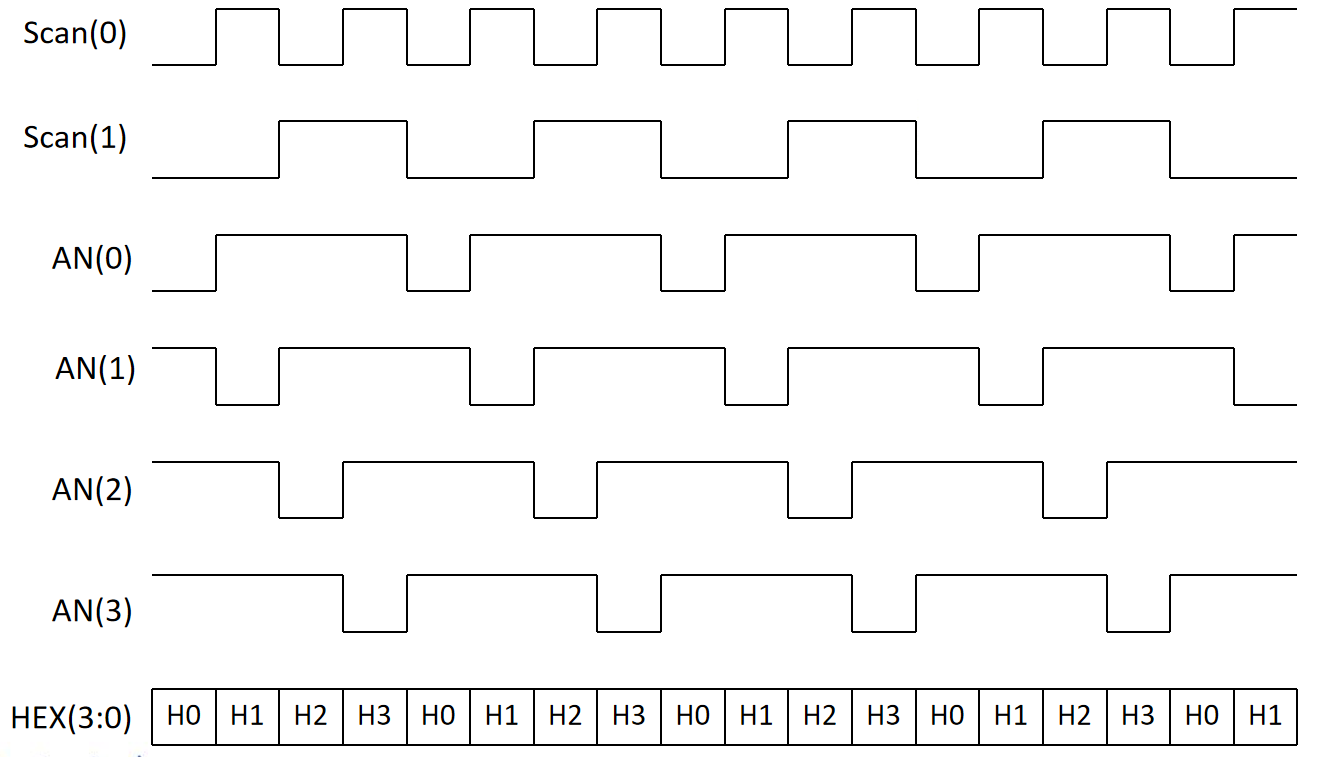
扫描信号来自时钟计数分频器：时序转化为组合电路。

由板载时钟clk(100MHz)作为计数器时钟，分频后的高两位信号（clk\_div[18:17]）作为扫描控制信号，输入2-4译码器产生数码管位选信号，控制哪个数码管显示（位选择），同时输入4选1多路复用器选择需要显示哪个数据（段码选择）。

计数器的分频系数要适当，几ms切换一次，眼睛舒适即可。



四位七段显示扫描控制的波形图：



**5.32位时钟计数分频器：**

 可输出2～232分频信号，可用于一般非同步类时钟信号延时较高，要求不高的时钟也可以用。本实验中用clkdiv(18:17)作为扫描控制信号，控制4 位数码管的动态扫描，每一位显示切换时间为 217 / 100M = 1.3ms。

**三、实验过程和数据记录**

**实验一：数据选择器设计**

1. 工程文件建立：新建工程文件，命名为“MUX4to1b4\_sch”。

2. MUX4to1元件设计：

新建原理图文件“MUX4to1”，依照四选一多路选择器原理，画如下原理图。

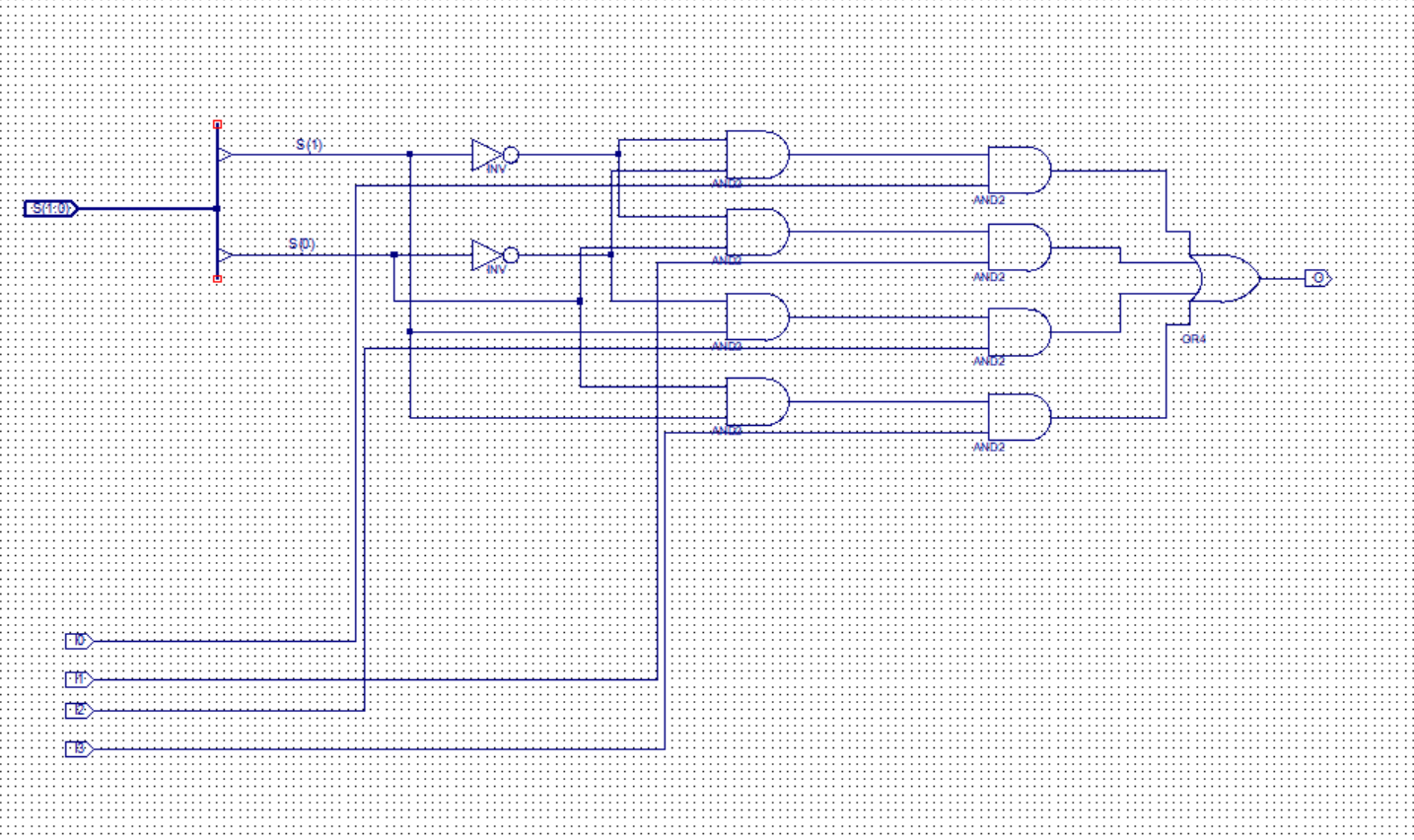


图1.1 四选一多路选择器原理图

对该原理图进行仿真，输入如下仿真代码:

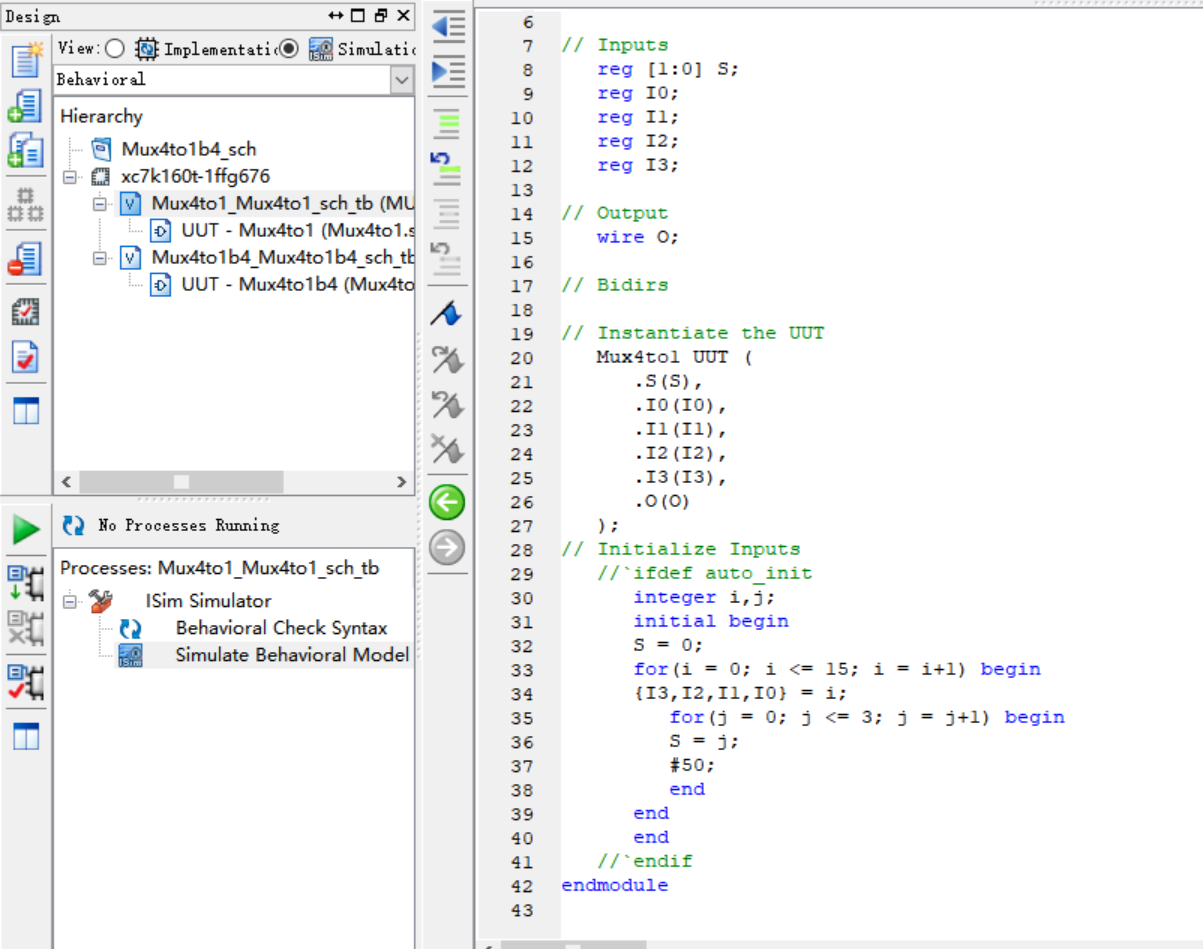


图1.2 四选一多路选择器仿真代码

得到仿真结果图如下（节选）：

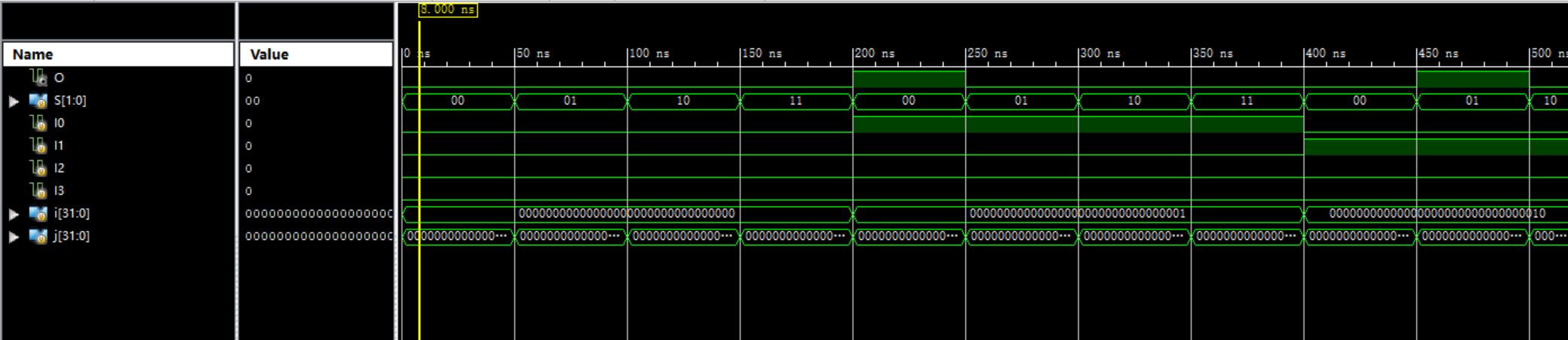


图1.3 四选一多路选择器仿真结果图

点击Create Symbol键，生成MUX4to1.sym文件，在实验二进行调用。

3. MUX4to1b4元件设计：

新建原理图文件“MUX4to1b4”，依照四位四选一多路选择器原理，画如下原理图。

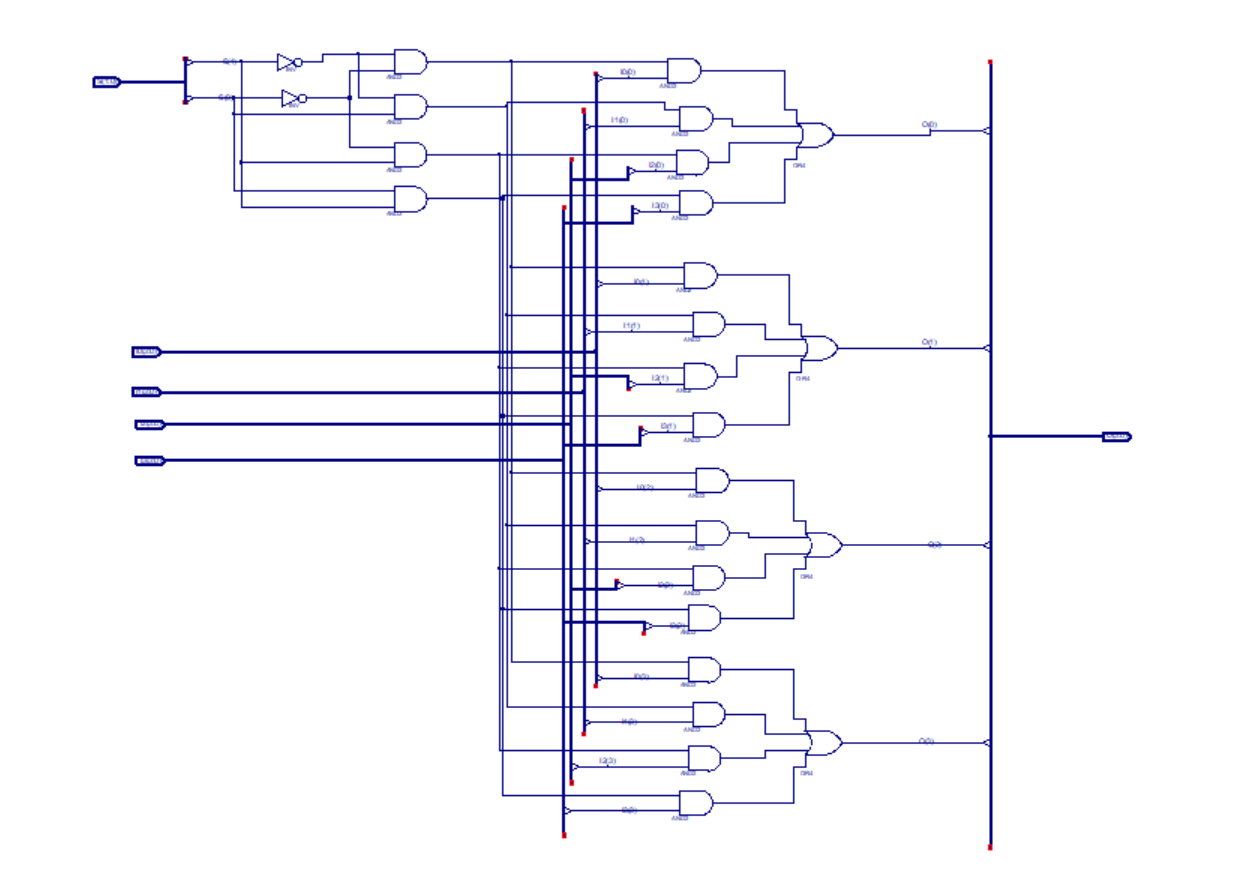


图1.4 四位四选一多路选择器原理图

对该原理图进行仿真，输入如下仿真代码：

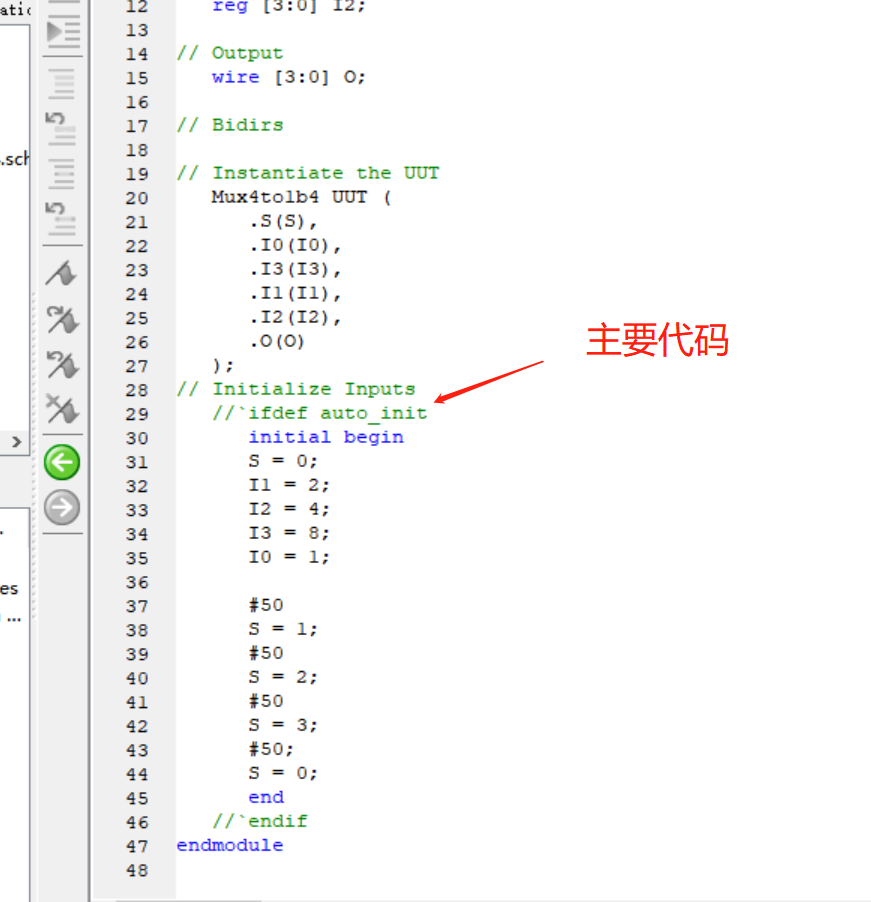


图1.5 四位四选一多路选择器仿真代码

得到仿真结果：

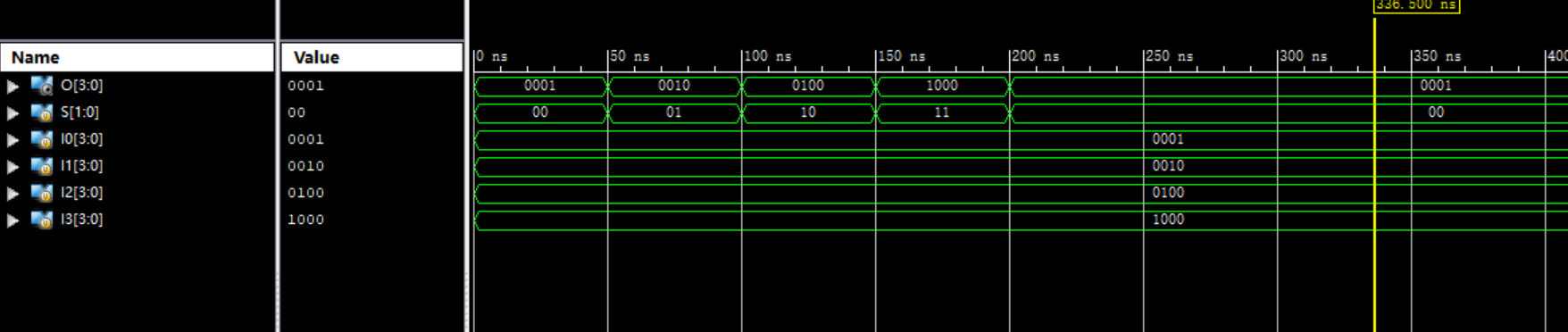


图1.6 四位四选一多路选择器仿真结果图

点击Create Symbol键，生成MUX4to1b4.sym文件，在实验二进行调用。

**实验二：记分板设计**

1. 建立工程文件，命名为ScoreBoard，Top Level设置为HDL。将实验一中生成的sym文件以及原理图文件复制到该工程的目录下。

2. DisplaySync模块设计：

新建原理图文件，命名DisplaySync，绘制原理图如下。

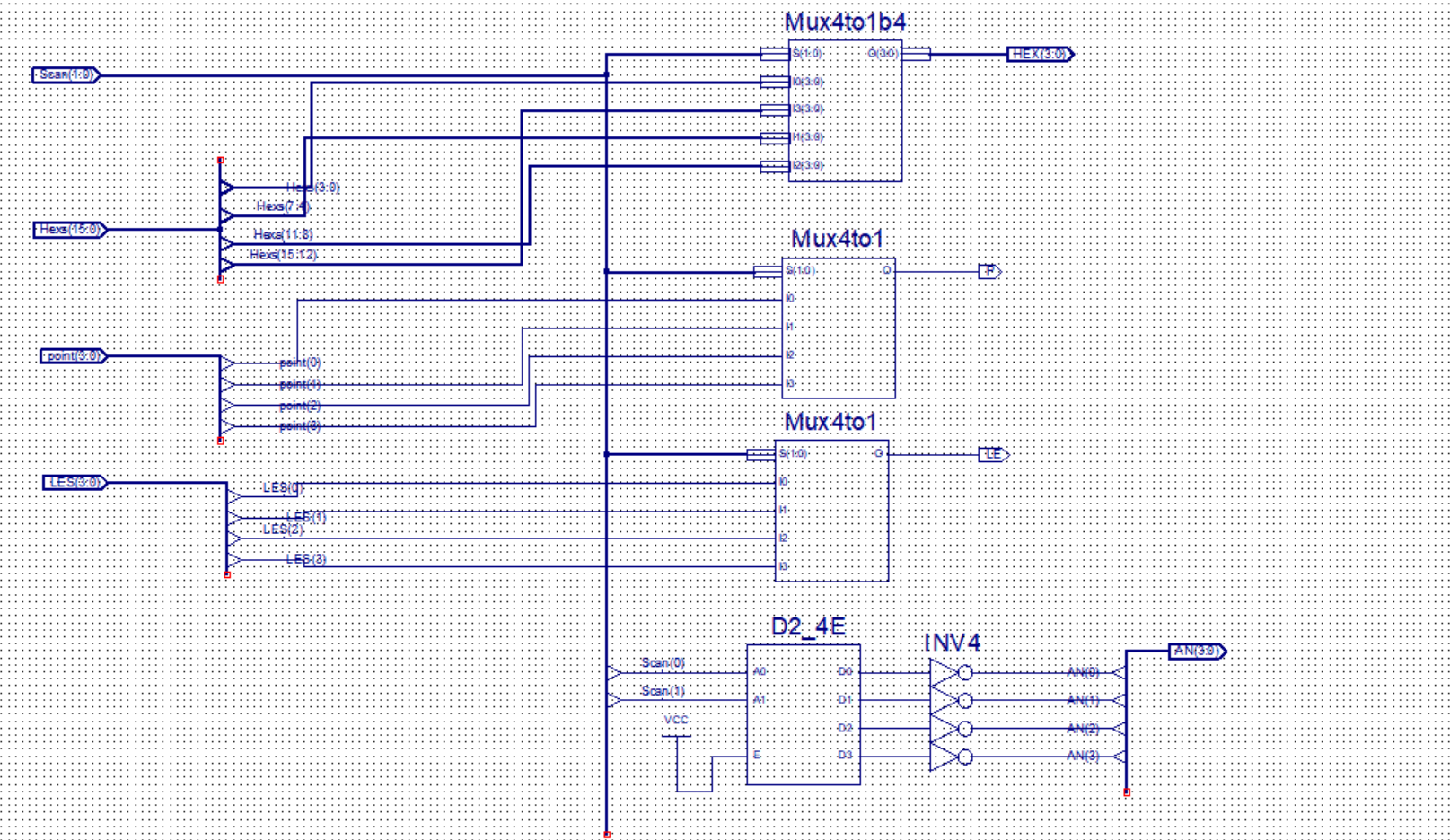


图2.1 DisplaySync模块原理图

其中：

Hexs(15:0)：需要显示的4个4位二进制数；

point(3:0)：每位数码管的小数点；

LES(3:0)：每位数码管是否需要消隐；

Scan(1:0)：扫描控制信号；

HEX(3:0)：当前要显示的4位二进制数；

AN(3:0)：4位数码管的位选择信号（低电平有效）

P、LE：小数点和消隐控制；

点击Create Symbol键，生成DisplaySync.sym符号文件。

3. 时钟计数分频器模块设计：

新建Verilog文件“clkdiv.v”，输入如下代码：

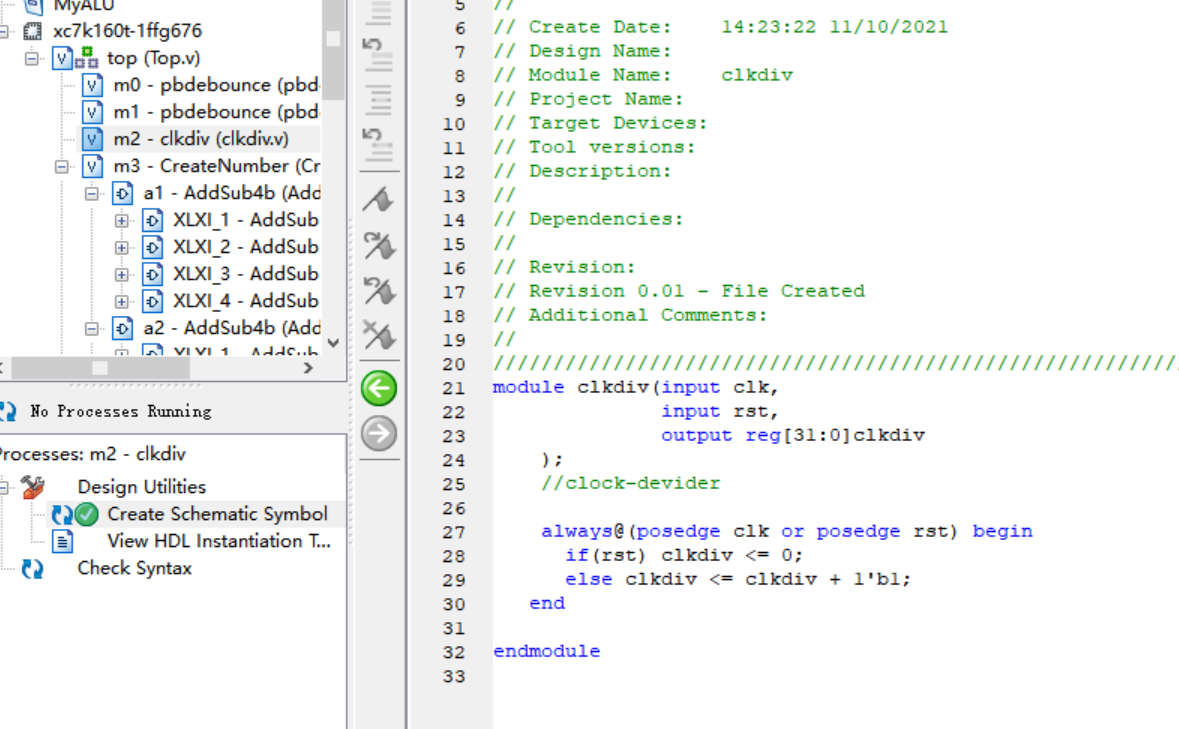


图2.2 clkdiv模块Verilog代码

其中，clk为实验板主时钟，rst为复位信号，clkdiv(31:0)为分频时钟输出。

点击Create Symbol键，生成clkdiv.sym符号文件。

4. CreateNumber模块设计：

新建Verilog文件，命名“CreateNumber.v”，输入如下代码。

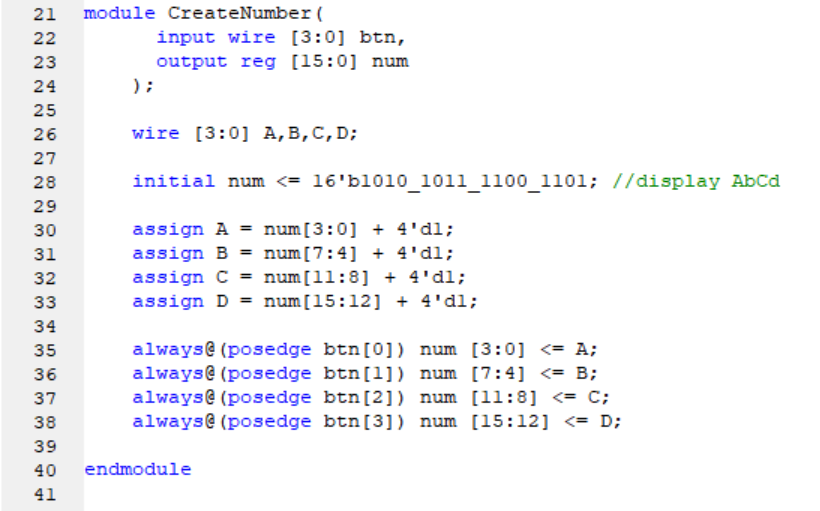


图2.3 CreateNumber模块代码

5. disp\_num模块搭建：

新建原理图文件“disp\_num.sch”，运用生成的符号文件辅助进行绘制：

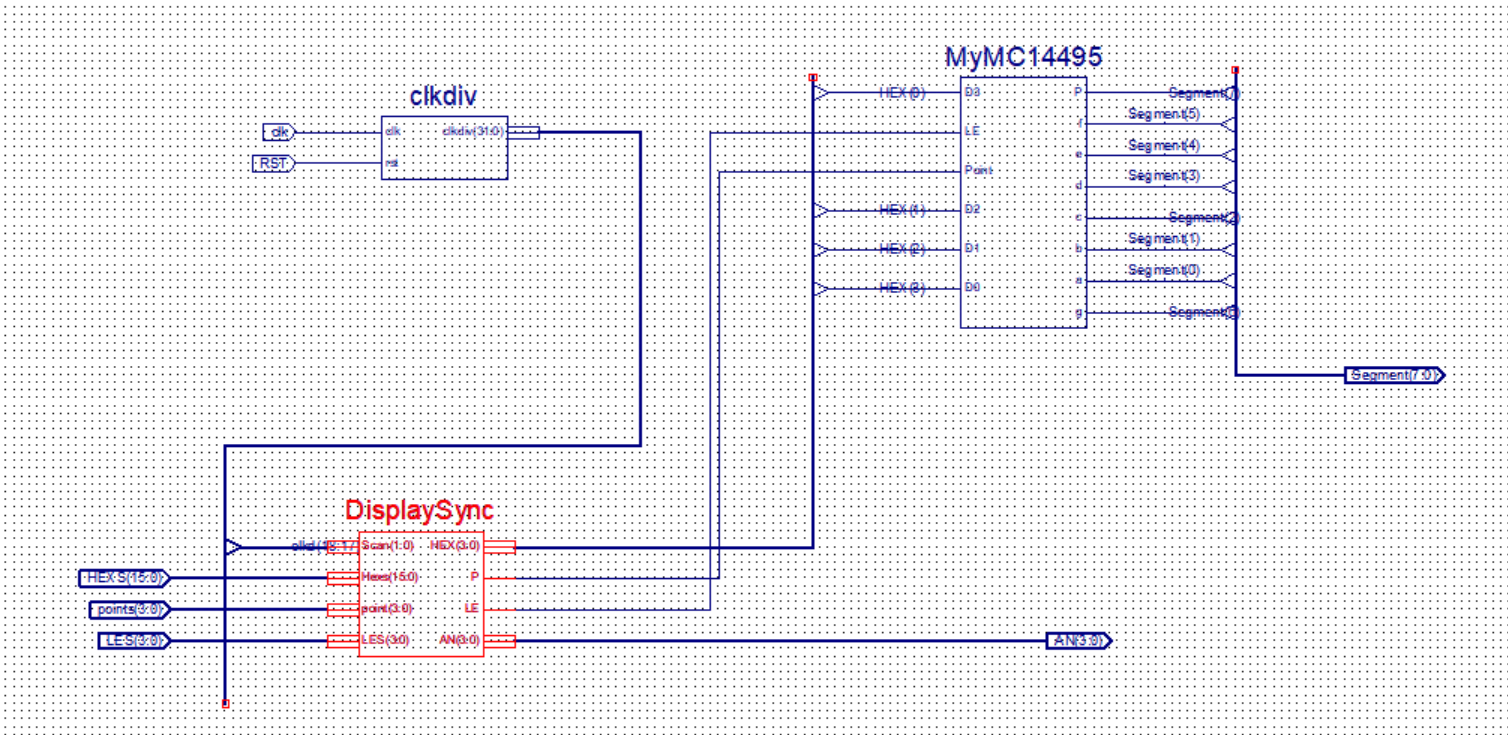


图2.4 disp\_num模块原理图

6. top顶层设计：

新建Verilog文件“top.v”，右键该文件，选择“Set as Top Module”。输入如下Verilog代码：

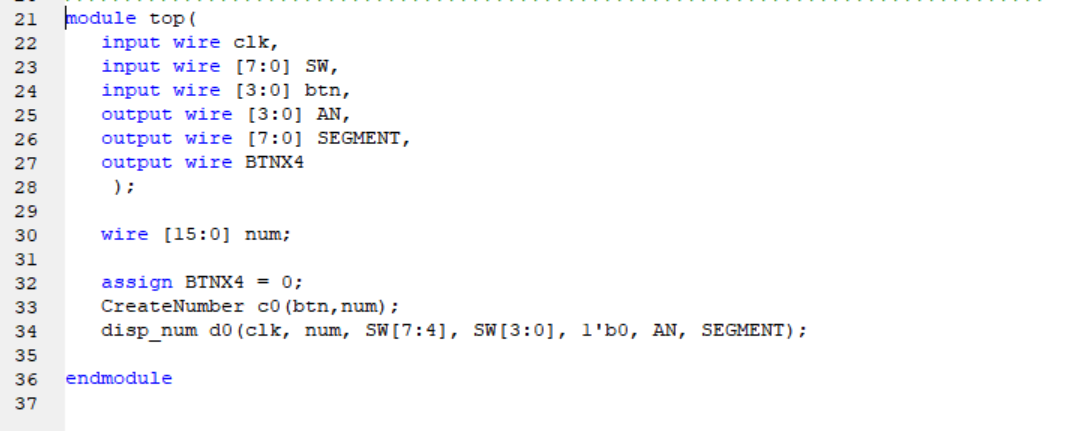


图2.5 top代码

将MyMC14495等运用到的前面实验设计的模块，通过Add Copy of Source加入到工程文件中。

点击View RTL schematic，查看原理图：

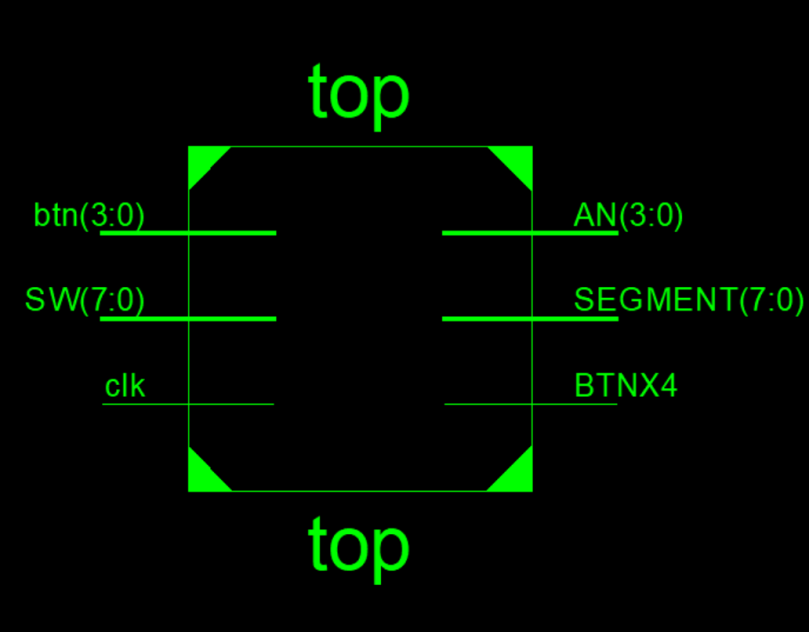


图2.7 顶层模块示意图

双击后查看内部细节：

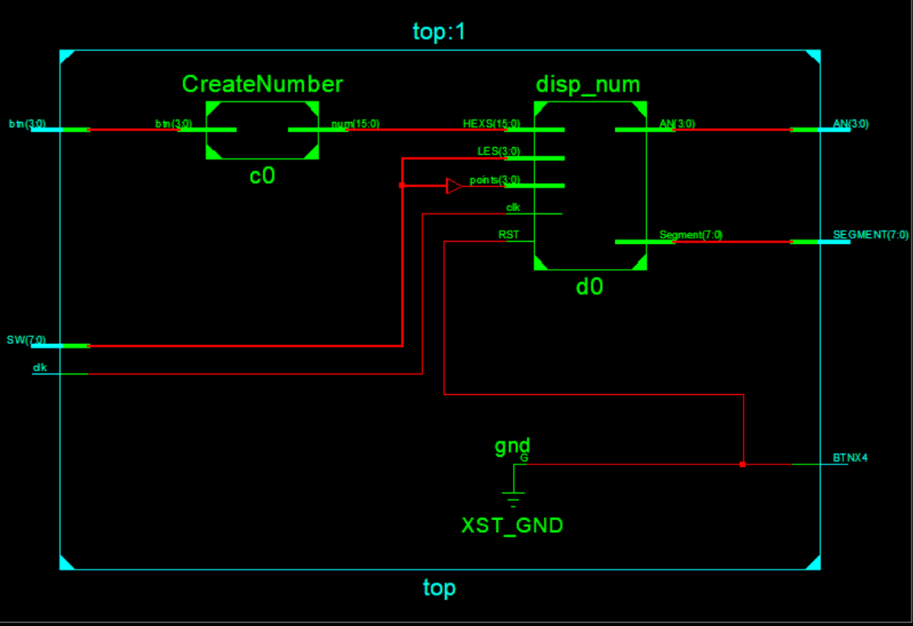


图2.8 顶层模块示意图（细节）

7. 对这一工程进行引脚分配。新建ucf文件K7，在其中编辑如下代码：

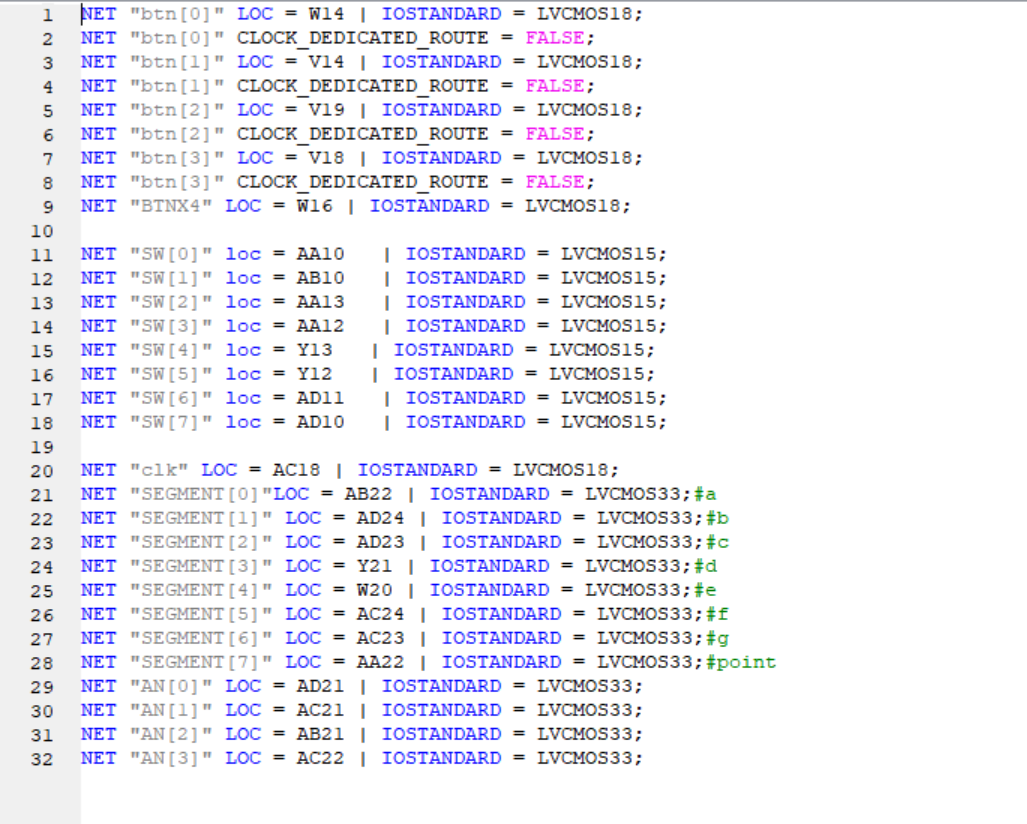


图2.6 引脚约束代码

进行Implement Design，Pinout Report显示如下：

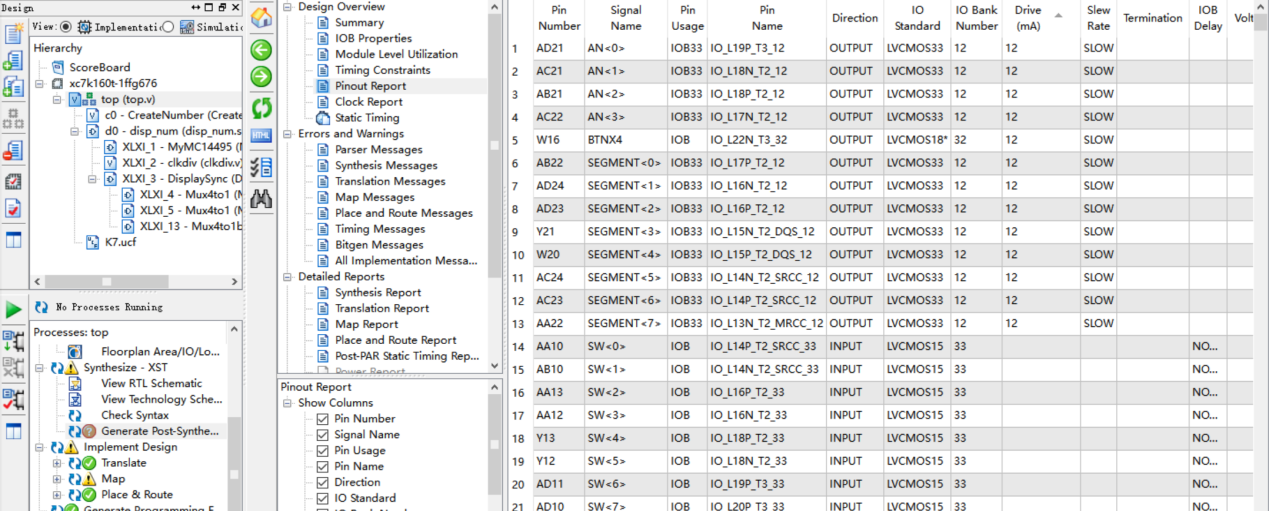


图2.9 pinout report

5.进行Generate Programming File，生成bit文件，并下载结果到SWORD板，操作开关，进行功能验证。验证结果如下。

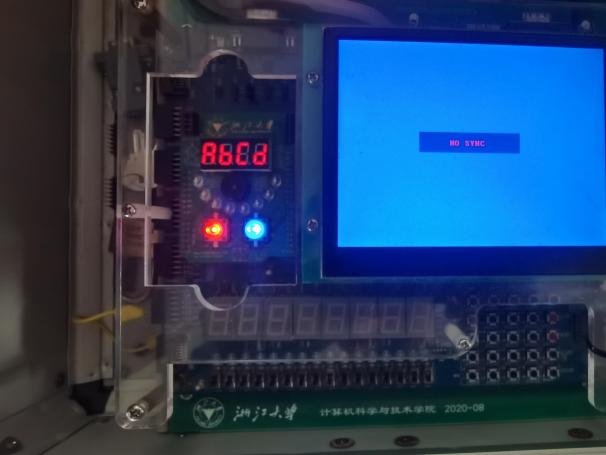


图2.9 初始状态



图2.10 显示全部小数点及数字



图2.11 显示部分小数点与数字



图2.12 全部消隐



图2.13 改变所有数字



图2.14 按下一次开关后第三位的变化

**四、实验结果分析**

**实验一：**

1. 原理图对比分析：

观察MUX4to1和MUX4to1b4的不同，可以看到位扩展不改变变量译码器，仅改变原本的一位的通道，将其扩展成四位，从而能够在选择出一一对应的每一位，而不会出现选出的结果出现位与位的不对应问题，实现多路选择器的功能。

2. 仿真分析：

本次仿真由自己完成，第一次编写相关的代码。参考之前实验的仿真代码，我对于MUX4to1进行了输入变量的遍历。通过for循环，实现I0~I4按照其构成的数字I[4:0]从0到15变化。输入每次变化时，S1S2分别从00变化到11。每组S1S0持续50ns，每组输入则相应持续200ns（由于I0~I4只有1位，若仅仅枚举情况，会失去一般性，因此使用了循环）。结果符合预期。

对于MUX4to1b4，由于组数较多，我采用枚举法，列举了一组具有代表性的情况，即确定I0~I4(各不相同，以验证选择情况），改变S1S0，每组S1S0持续50ns，从而验证了S1S0对四位输入I的选择。

**实验二：**

1. 整体架构以及代码、原理图功能分析：

最顶层的模块为top.v，其代码内容主要为调用CreateNumber模块以及disp\_num模块。

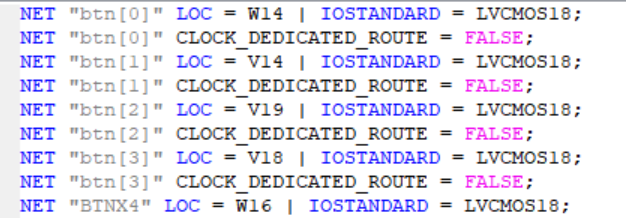
其中CreateNumber模块主要实现数字的生成以及自增，初始化定义对16位的num变量进行赋值，使得最初显示的数字为“AbCd”，并且通过btn变量，响应按钮按下事件，按下一次后实现num对应位数的数字自增。

disp\_num模块主要实现在数码管上通过LED显示数字，通过调用实验六的MyMC14495来实现数字在七段数码管上的显示。同时，使用DisplaySync模块和clkdiv时钟分频器动态的扫描，使得数字能够持续在七段数码管中显示。并且通过DisplaySync中的变量译码器和四选一多路选择器，实现对于小数点、数字的消隐控制以及数字的变化显示。

2. 引脚约束分配分析：

我们将小数点以及数字的使能端（共8位）分别分配到SWORD板上的8个开关，开关下拨显示，上拨则消隐。同时，分别为MyMC14495分配相应的七段数码管以及小数点，从而将CreateNumber以及DisplaySync中的数字反映在硬件上。

此出我们第一次对于按钮进行引脚约束分配。可以看到我们代码格式如下：



回顾前面实验四通过按钮实现楼道灯控制时，我遇到的困惑（不能实现按钮操作，只有开关）。此处可以解释一部分原因，即若缺少NET “btn[i]” CLOCK\_DEDICATED\_ROUTE = FALSE这行语句，得到的结果是按钮按下依然无反应，无法实现功能，只有对相应的输出进行取非操作才可以实现。通过查找资料，此行语句将CLOCK\_DEDICATED\_ROUTE设置为FALSE，相对于允许忽略时钟配置，目前我对此的理解是这一操作给予按键更高的权限，能够实现对程序的中断以及状态的改变。因此若没有这行语句，无法在设置了时钟的情况下实现楼道灯的控制。

3. 实验结果分析：

通过SWORD板上的操作，可以看到消隐的控制可以正确实现，但是按按键后却无法实现自增1，其数值变化较大。此现象的原因是我们未进行软件防抖动。由于板上的按键是机械弹性开关，按下再松开的过程中，开关在闭合时不会立刻接通，松开时不会立刻断开，中间可能会产生几处电压的变化，从而在按下一次按键后会出现自增多于1的情况。此情况避免可以在后续实验中学习。

**五、讨论与心得**

1.本次实验第一次进行了分模块和top文件的搭建，使我对于一整个工程的基本架构有了更深入的认识，也是后续实验操作时，框架搭建的基础。

2. 本次实验我自己编写了仿真的Verilog代码，对于代码的语法规范有了更好地了解，对于其语句的功能有了更深入的理解，能够初步实现相应的功能。

3. 本次实验中，我在完成基本的功能后，发现初始显示并不是AbCd，对此我对比了CreateNumber模块代码和具体的在七段数码管中实现之间需要经过的电路。首先我排除了模块代码编辑的错误，num的初始值定义没有问题。沿着通路进行分析，最后发现是由于MyMC14495输入与DisplaySync模块输出的HEXS变量的连接出现差错，从而导致显示的初始值不符合预期，这也是由于两处的变量定义的差异。解决这个问题后，整个工程能够实现基本的功能，但是对于抖动的问题依然没有解决，需要进一步学习。