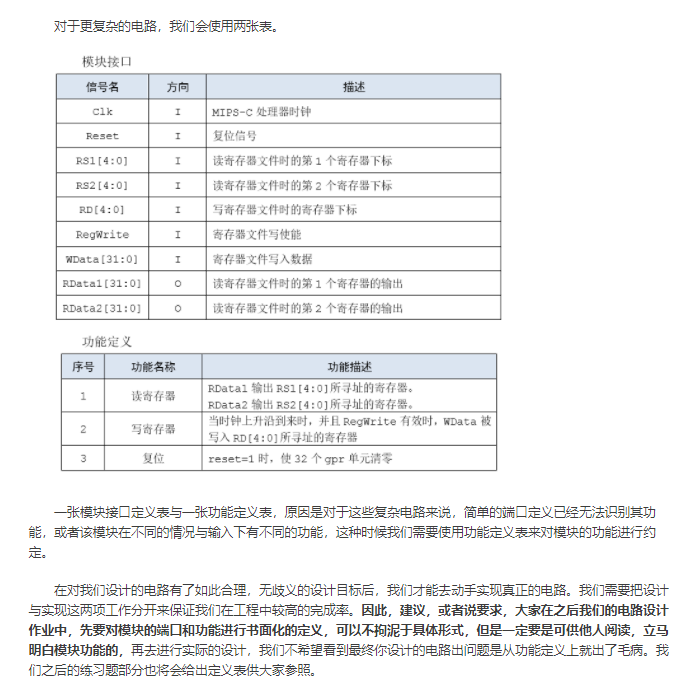
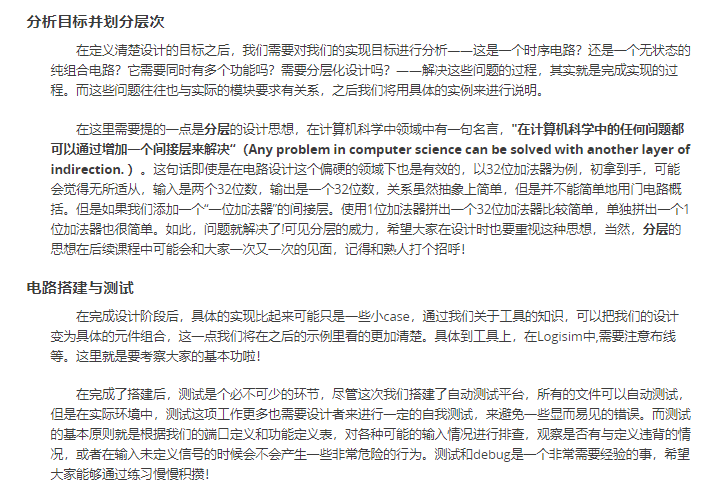
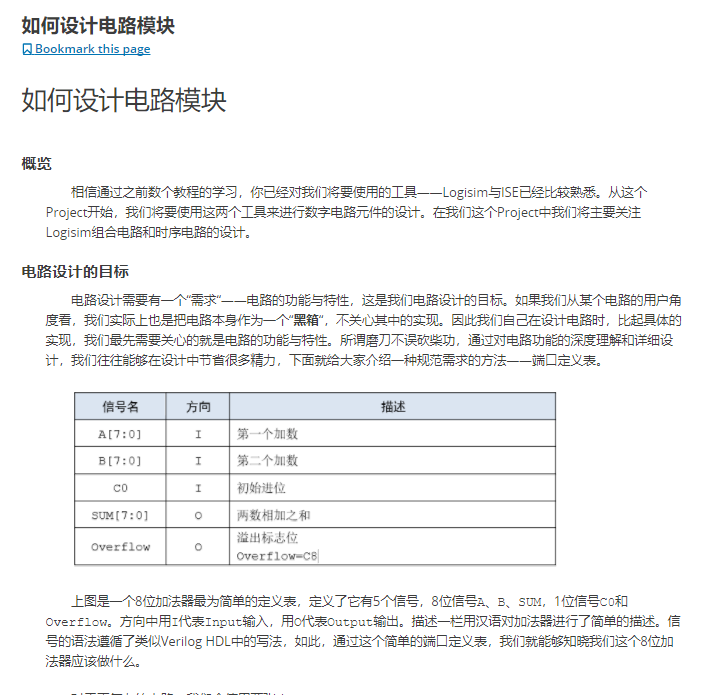
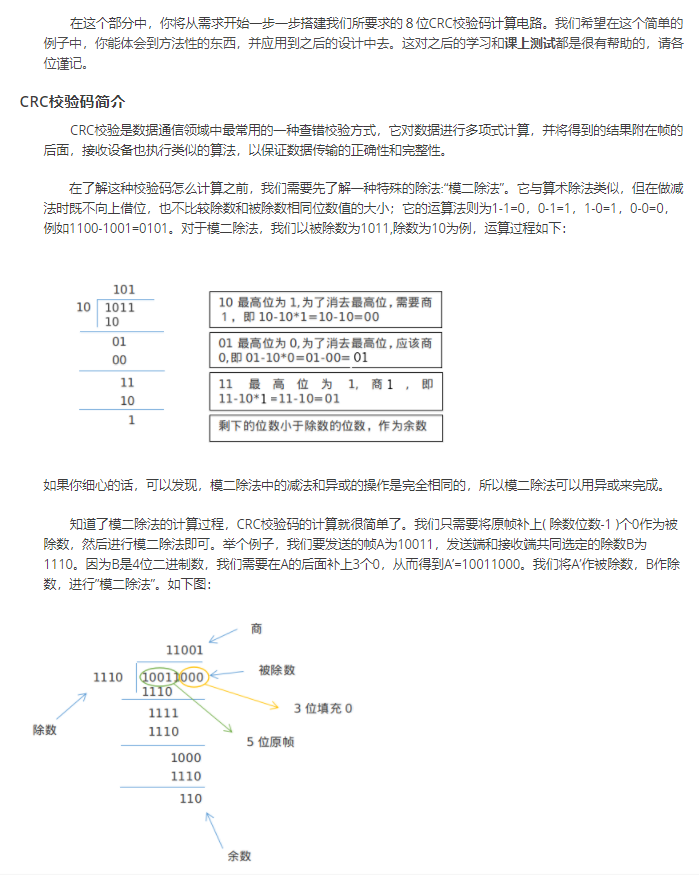
**P0**

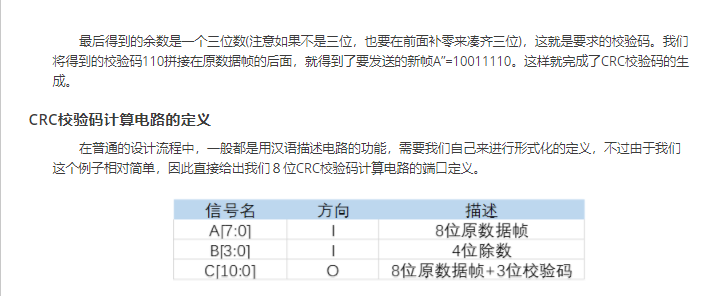
**如何设计电路模块：**

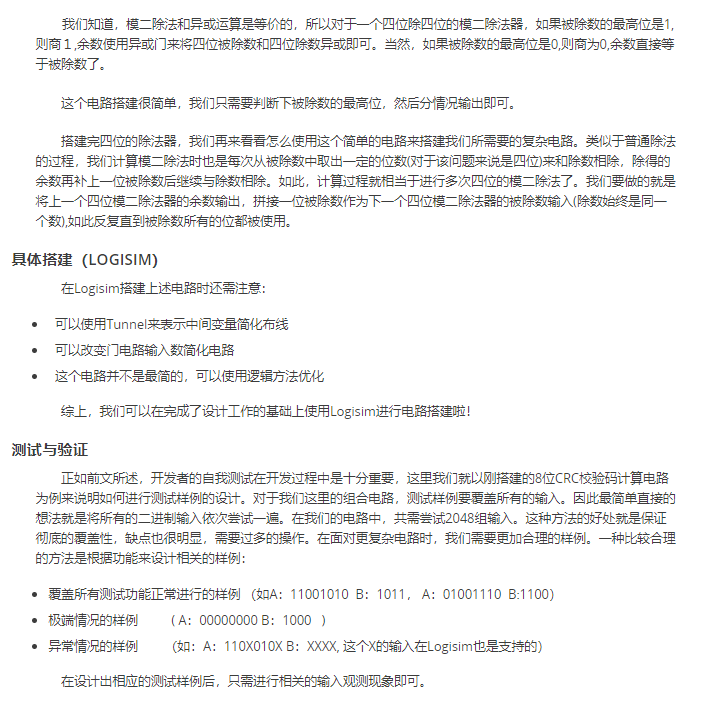
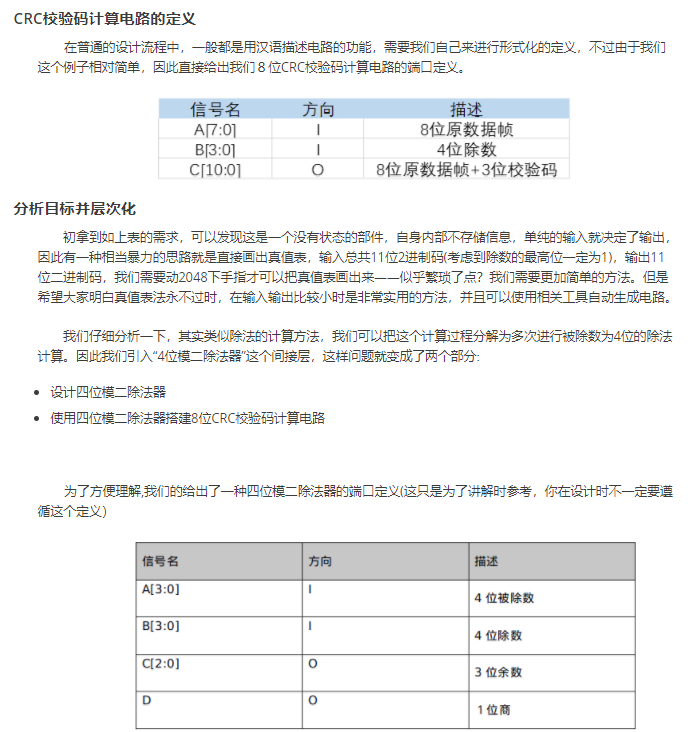


****因此，建议，或者说要求，大家在之后我们的电路设计作业中，先要对模块的端口和功能进行书面化的定义，可以不拘泥于具体形式，但是一定要是可供他人阅读，立马明白模块功能的，**再去进行实际的设计，我们不希望看到最终你设计的电路出问题是从功能定义上就出了毛病。我们之后的练习题部分也将会给出定义表供大家参照。**

### **CRC校验码计算电路 的设计与测试：**







**模二除法：**

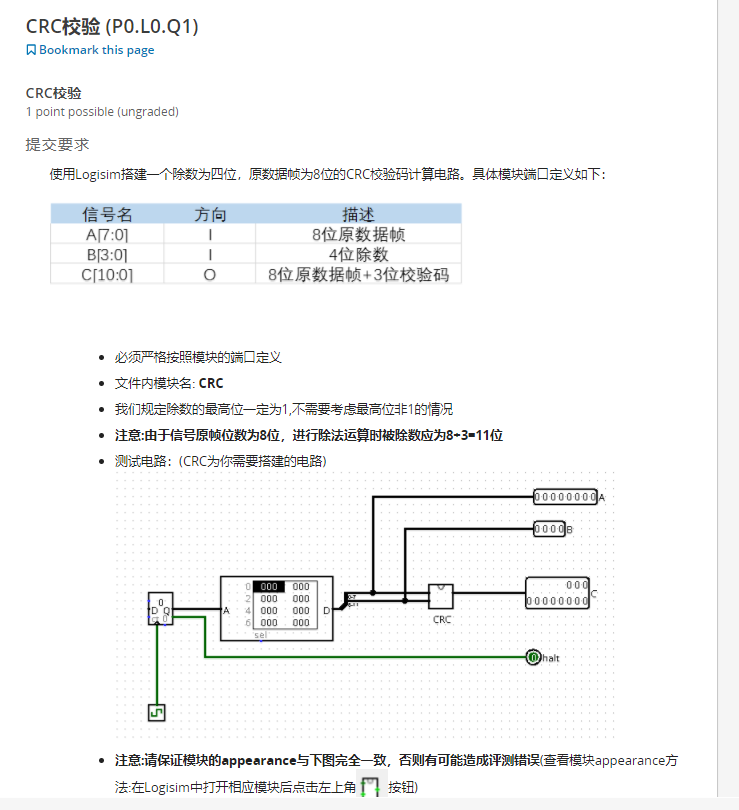
**0-0=0 1-0=1 0-1=1 0-0=0**

**实际上是异或操作**

**在原被除数后面补上（除数位数-1）个0作为新的被除数**

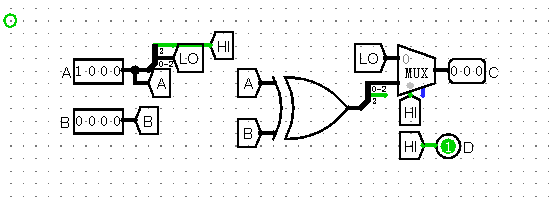
**模二除法和异或运算是等价的，所以对于一个四位除四位的模二除法器，如果被除数的最高位是1,则商１,余数使用异或门来将四位被除数和四位除数异或即可。当然，如果被除数的最高位是0,则商为0,余数直接等于被除数了。**

**将问题分层次，分步骤~**



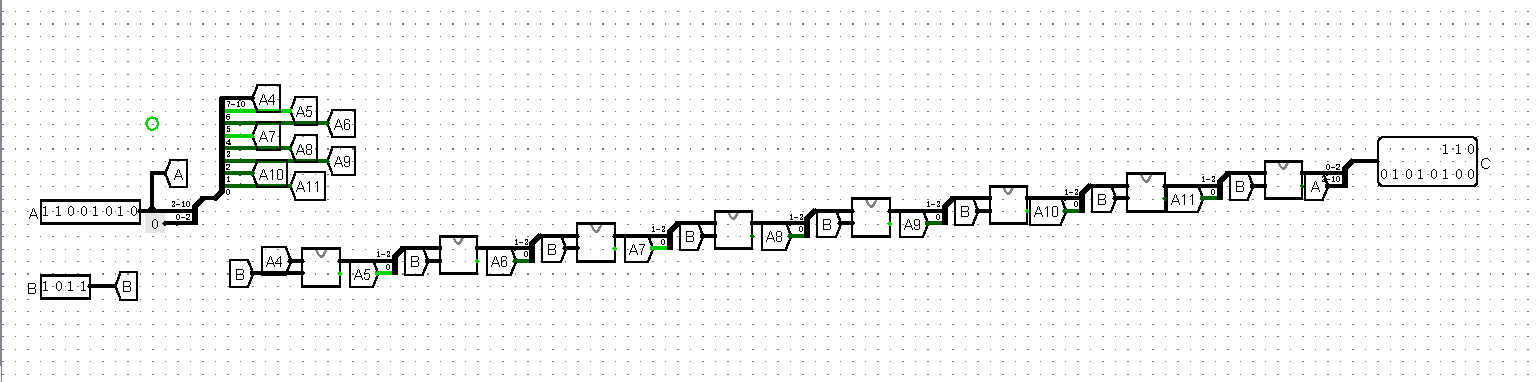
**注：由于保证了输入的B最高位一定是1，所以不用考虑为被除数最高位为1而除数最高位为0的情况，即可以启动这条流水线**

**①先搭建4bit\_mod**



**如果被除数的最高位是1,则商１,余数使用异或门来将四位被除数和四位除数异或即可。当然，如果被除数的最高位是0,则商为0,余数直接等于被除数了。商在此过程中无用。**

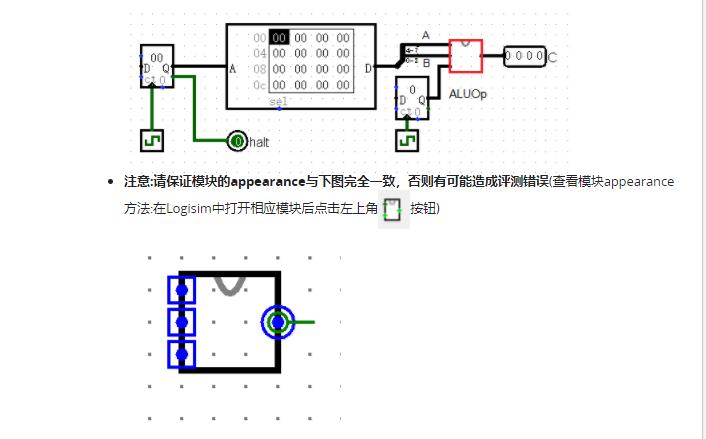
**②用这个直接搭建CRC**



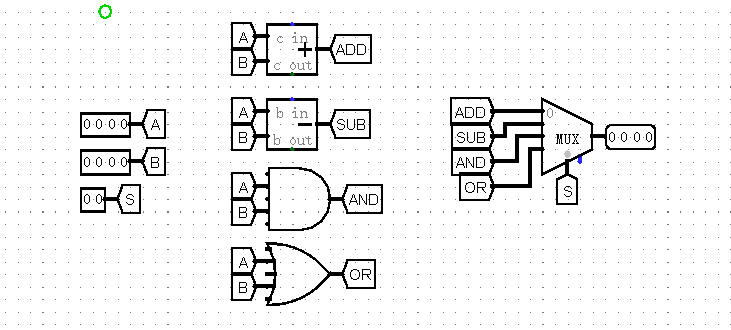
**注意别搞反位数，从最高位开始，注意Tunnel的妙用**

**注意外形-**

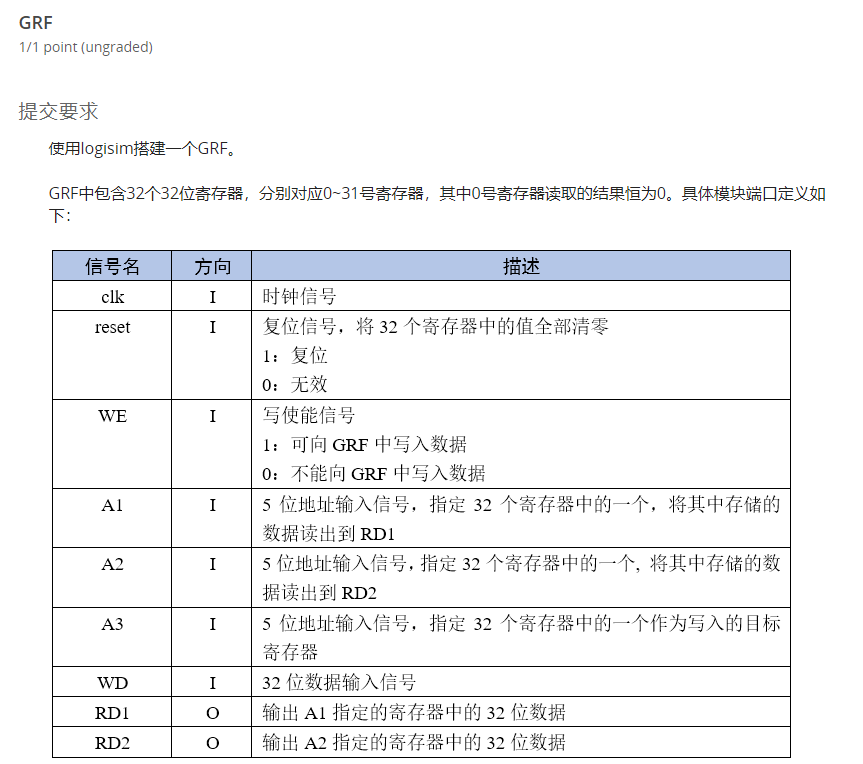


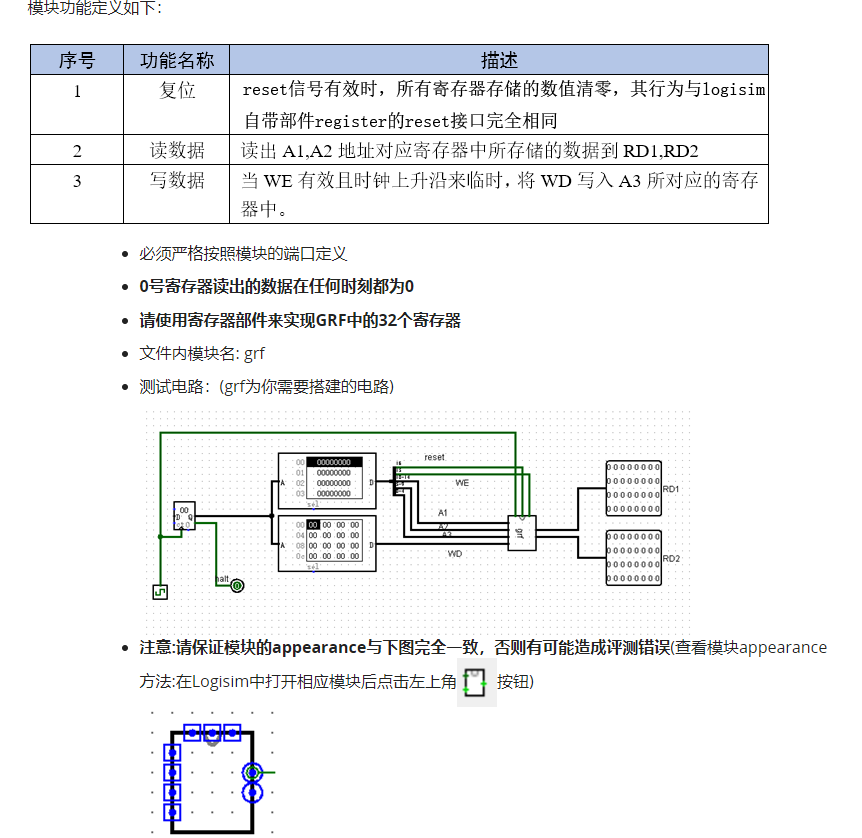


**注意读准题目是加减与或而不是加减乘除，别想当然**



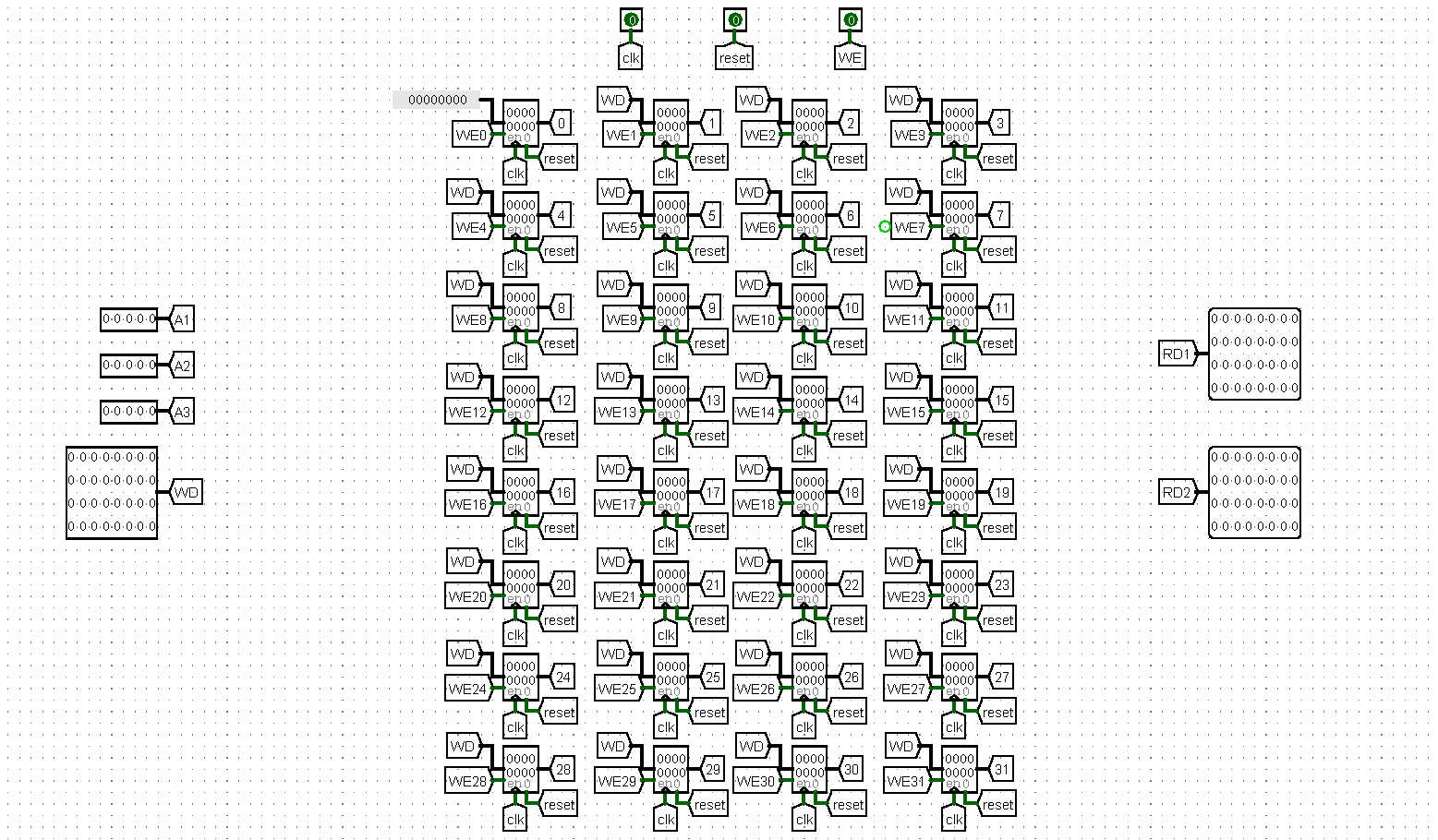
**注意标签和线是否接好**

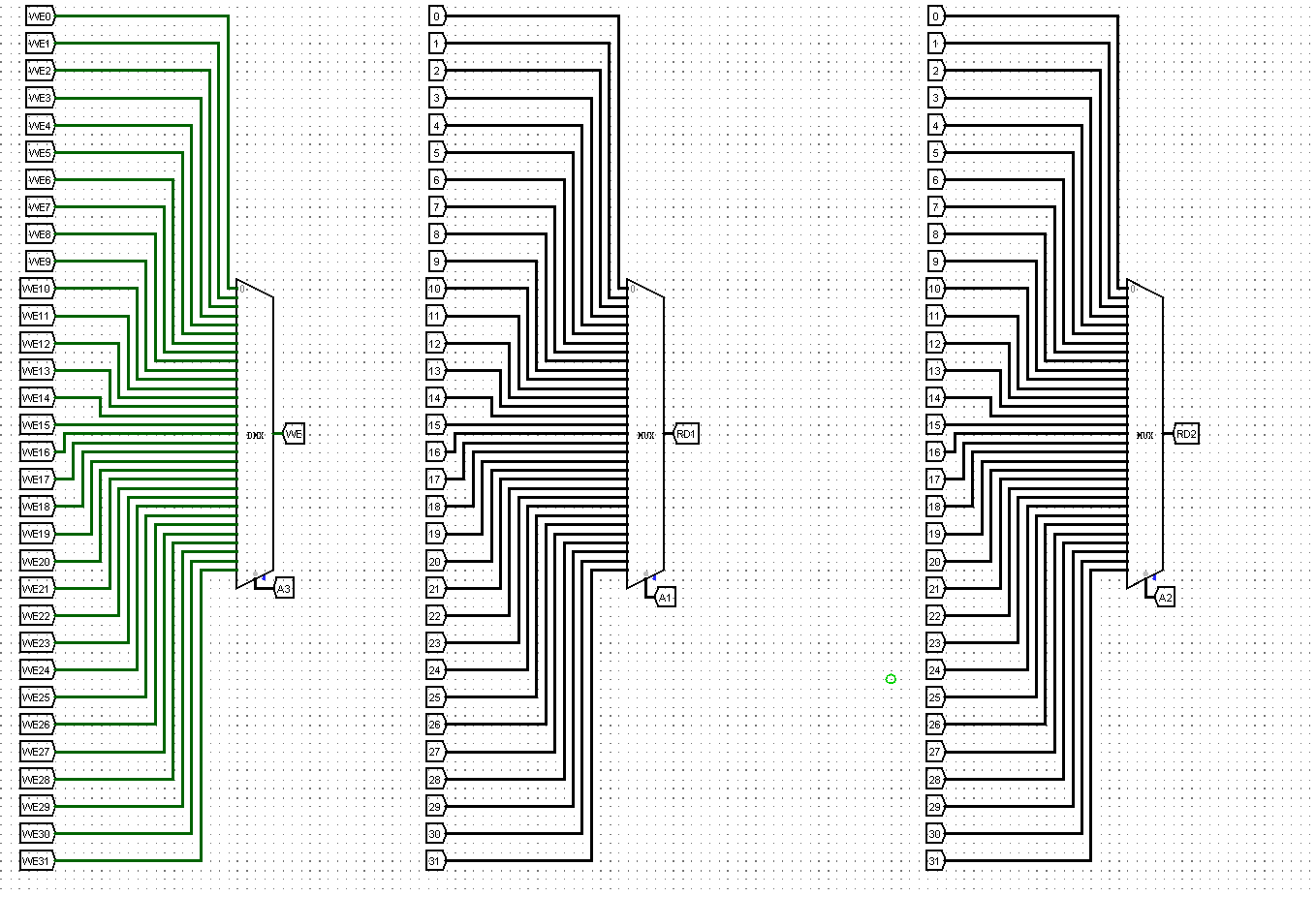




**先不急着开始搭，先搞清楚每个部分的模块再copy**

**注意标签的应用，和注意一定要检查模块的样子！！！**



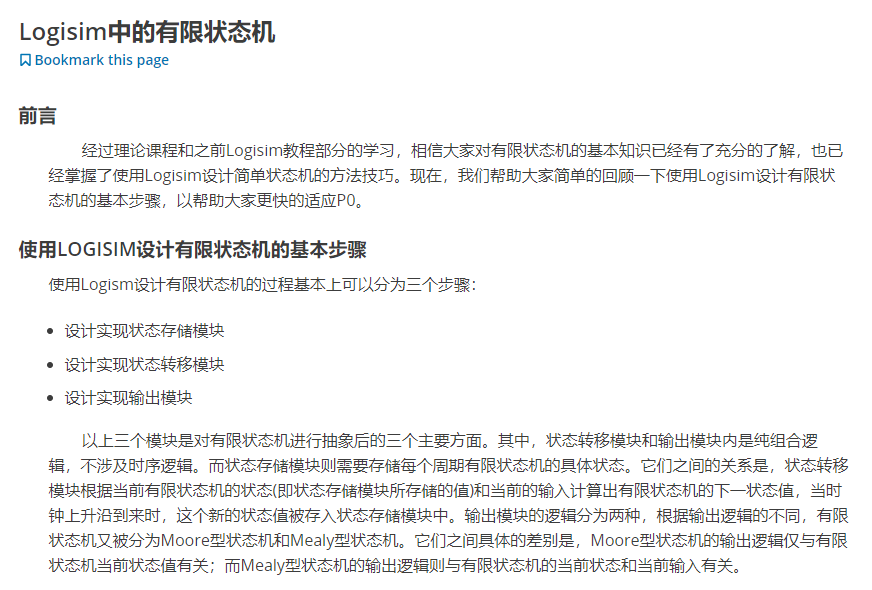


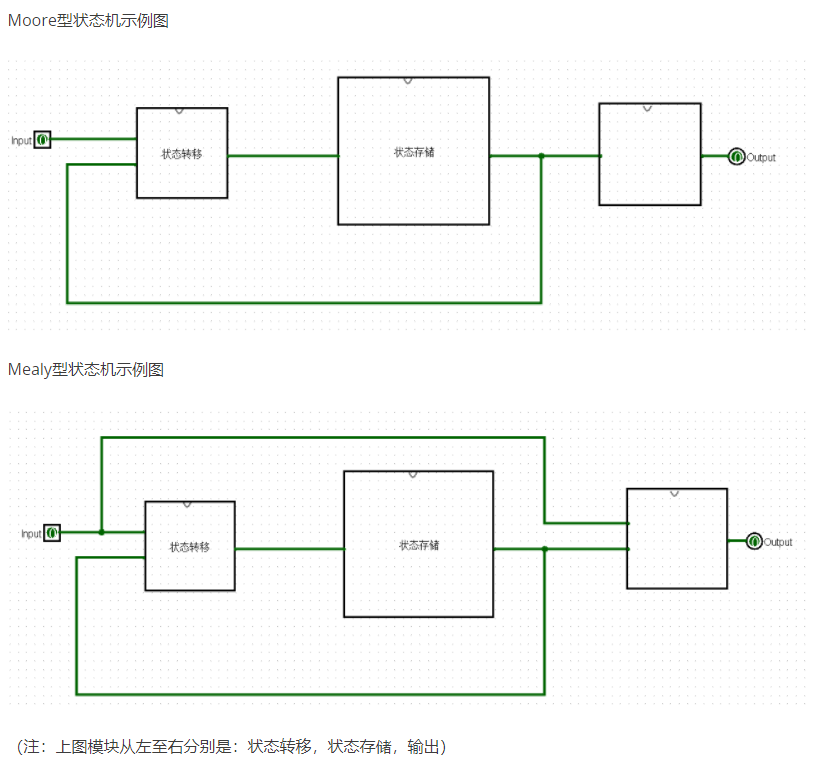
**注意要在子电路中开辟模块！！！！！**

**注意COPY时记得改标签~！！！！！**

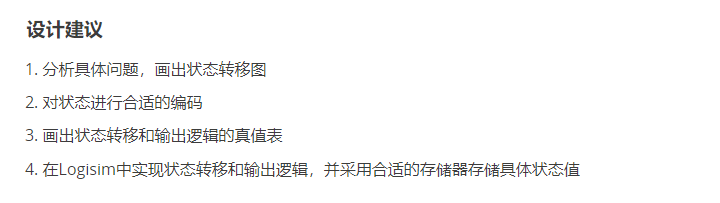
**非常重要的有限状态机！！！**



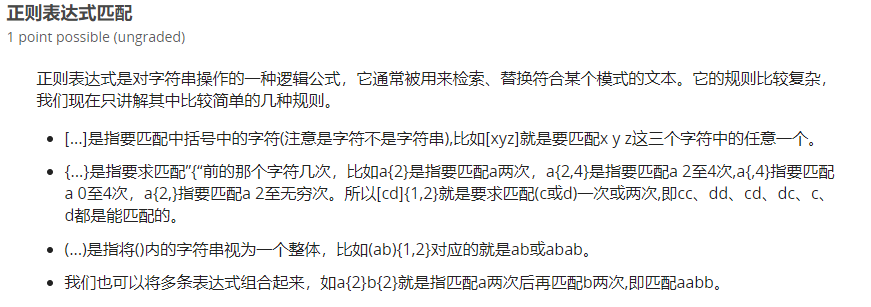


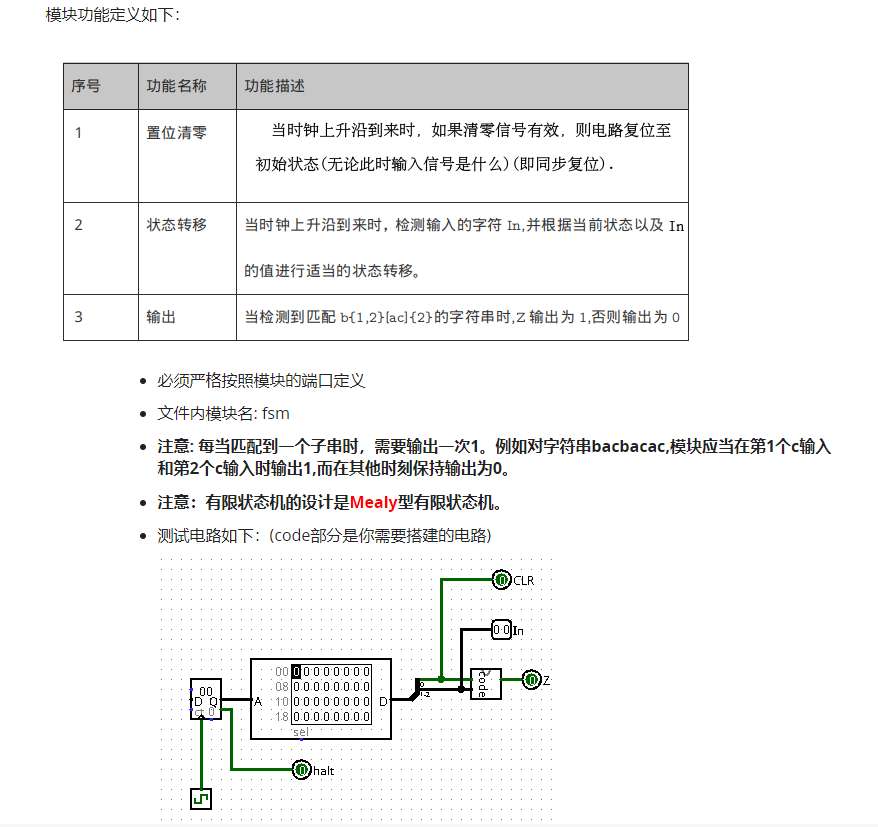


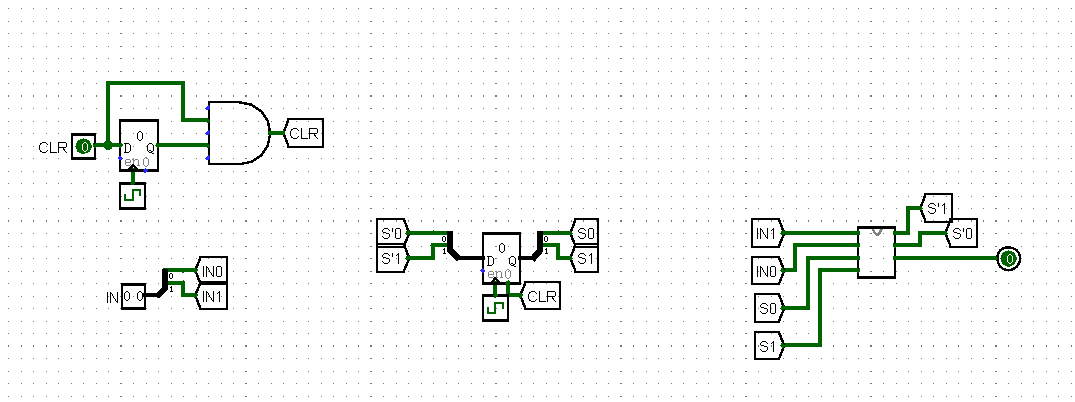
**Mealy型状态机输出还跟输入有关，得在输出模块连多一条输入线。**



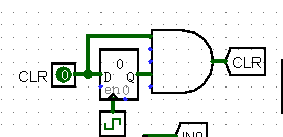
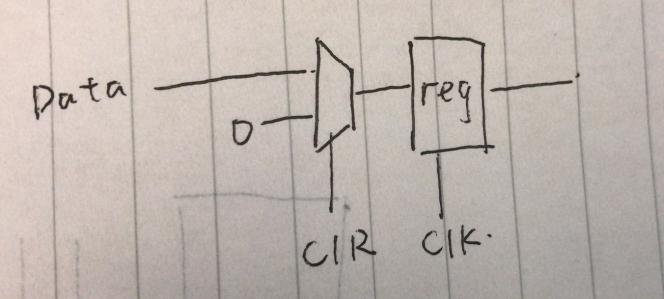
**何为正则表达式**

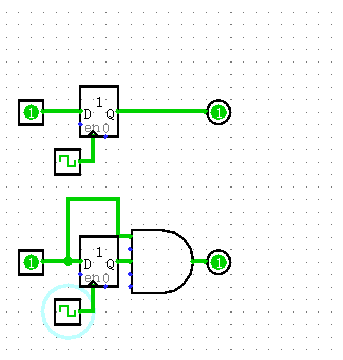






**自己测试的时候感到费解是因为，它的输入并不是说启动时钟才算输入，启动时钟是记录上一个状态，不启动时钟的时候相当于直接输入的了。**

**然后关于reset同步的问题，就是说如果不加寄存器的话，在时钟上升沿未到来时若CLR输入1就直接清零，不同步。如果不加寄存器可能会有这种情况就是说，在CLK低电平的时候CLR震荡，就会造成电路爆了，所以就加个寄存器，判断它时钟到来时刚好是稳定的就RESET。**or****

****

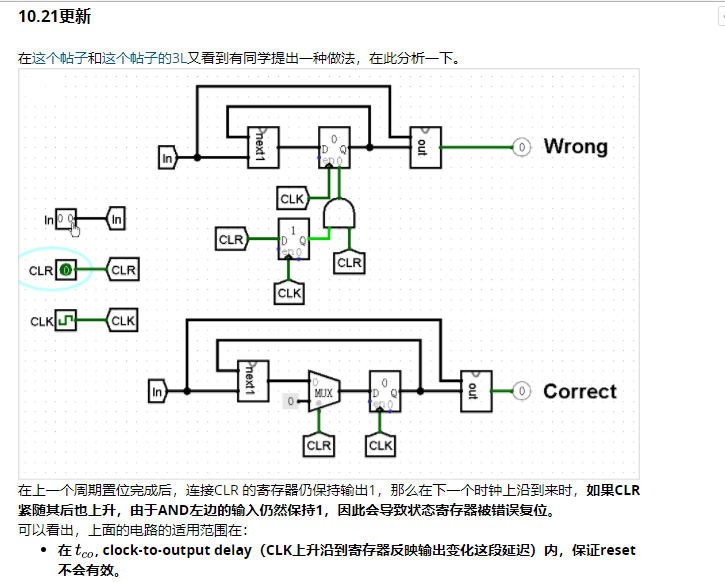
**第一个图，加与门会比不加与门的输出慢两步，可能是在上升沿到来的左右都得保持稳定吧。**

**同步是得等到CLK有效才变化，异步是一旦CLR变，就reset。**

**第二个图意即，任随CLR怎么变化，只选择CLK上升沿的那个时候的CLR决定是否清零。（好理解）**

**用第二个~**

此题中reset只改变状态的值，别连上输出端就好。



**Mealy机：**

**左边是输入部分，中间是状态存储部分，右边是输出部分，注意输出逻辑部分和输入及之前的状态有关。**

**同步和异步：主要是用寄存器来区分当前状态和下一个状态。**

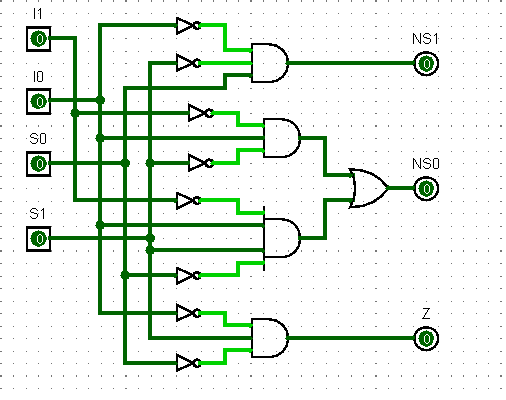
**《数字设计和计算机体系结构》P69提到，一个电路是同步时序电路的话，它需要满足：**

**①每一个电路元件或者是寄存器或者是组合电路。**

**②至少有一个电路元件是寄存器。**

**③所有的寄存器都接收同一个时钟信号。**

**④每个环路至少包含一个寄存器。**





**回到此题：**

1. **确定输入输出，及状态总数**

**输入：IN CLK CLR**

**输出：Z**

**状态总数：初态加b{1,2} × [ac]{2} 实际上可以抽象成两类，所以总共是3种状态**

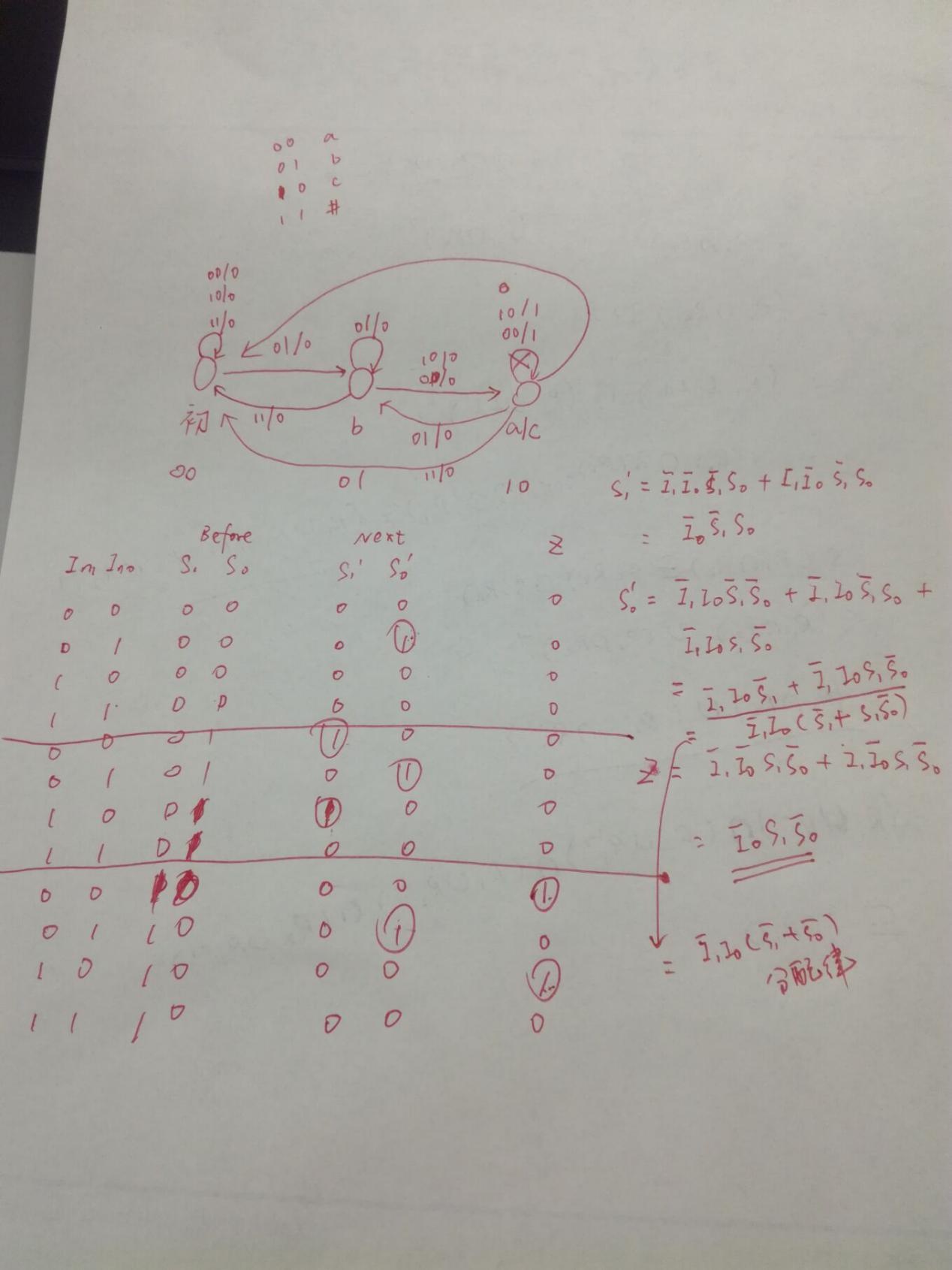
**2、画出状态转换图**

**3、写出真值表**

**4、选择状态编码（一般选择二进制编码）独热编码虽然比二进制编码需要更多的触发器，但是输出逻辑比二进制编码简单，视情况而定。**

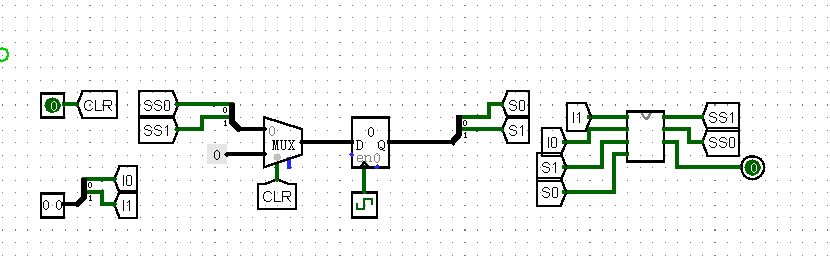
**5、写出状态转移和输出逻辑的表达式**

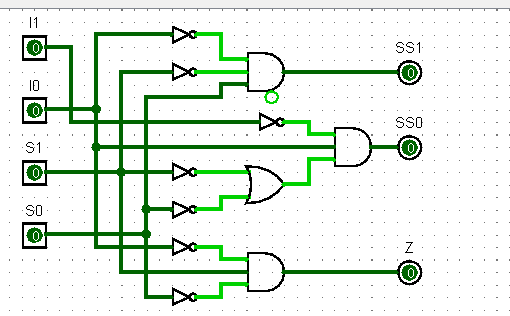
**6、使用Logisim画出电路图**

****

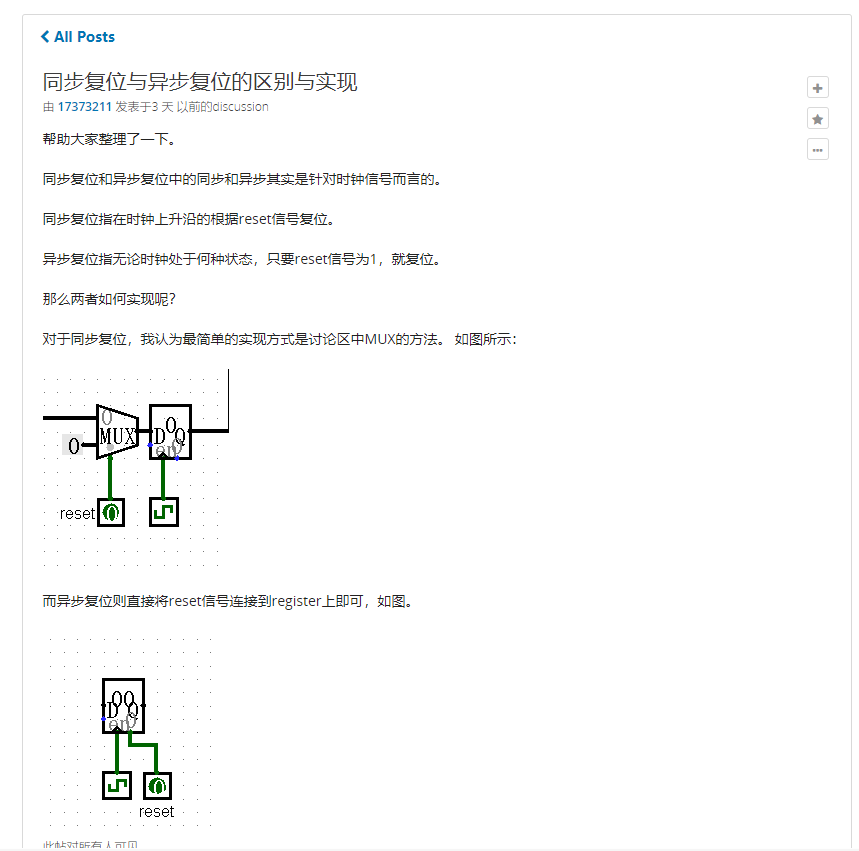
**7、用Verilog进行仿真和测试**

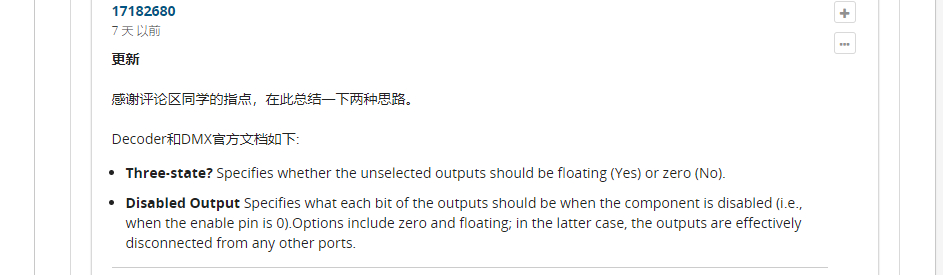
**注意实现同步复位！**





**以后用有限状态机还是分开输入输出和状态转移模块为妙！**

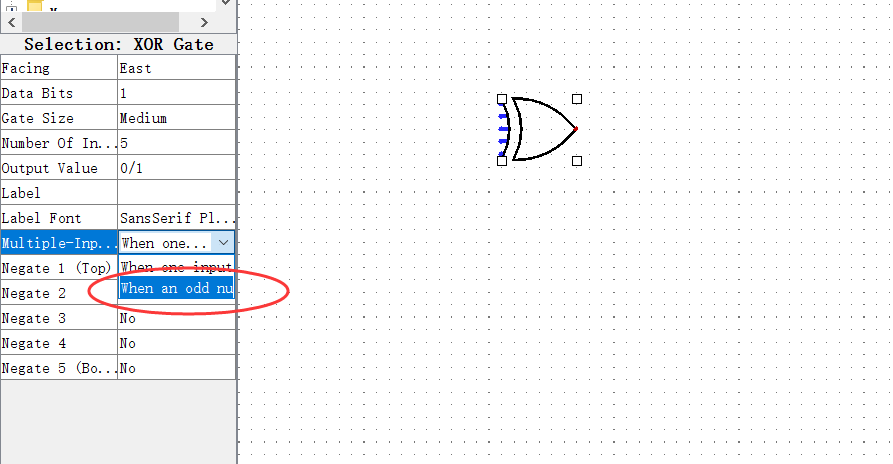




**用DMX控制输入的话可能会有错，但是用DMX控制ENABLE似乎没毛病，但是用decoder控制select则非常合适，因为decoder是把输入的选择信号的二进制编码输出成独热编码。**

**以文档形式打开circ文件，读代码。**





异或门注意调这个，不然会出现1 1 1输出0的情况。