**数字逻辑与部件设计实验报告**

实验九：密码锁



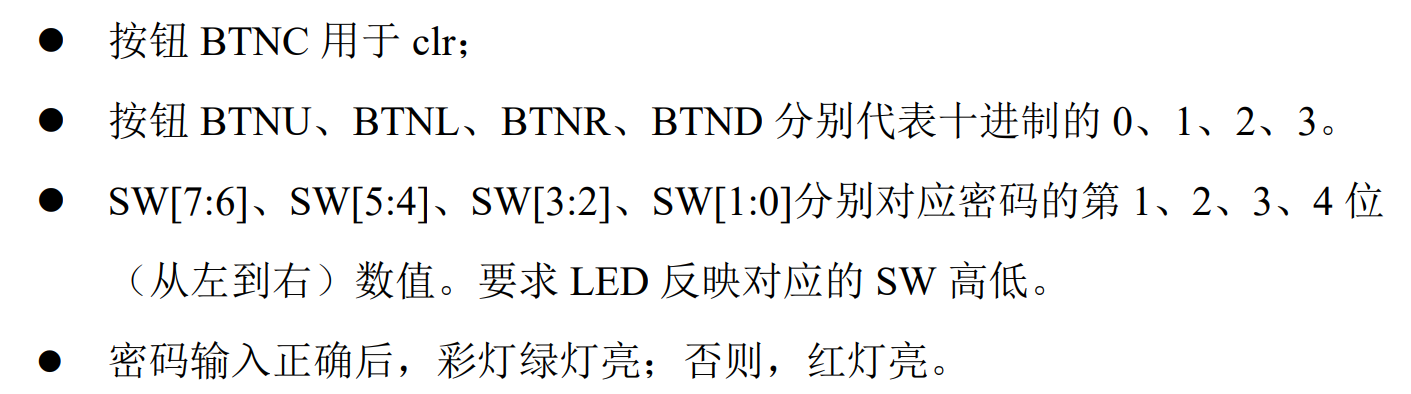
学生姓名： 谢志康

学号：22307110187

日期：2023.11.13

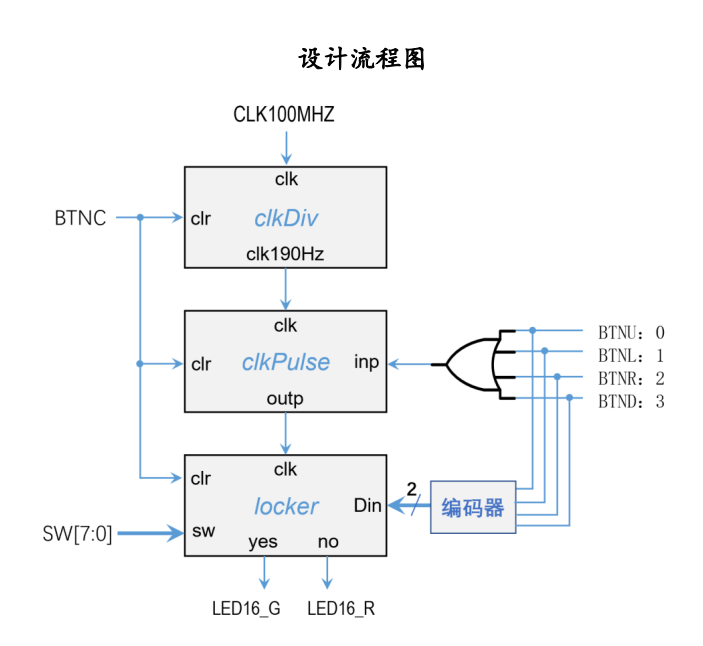
1. 实验内容

设计一个 4 位密码锁电路：利用拨码开关设置初始密码的，通过按钮输入密 码；如果用户输入正确密码，绿灯亮；否则，红灯亮。 在 NEXYS4 开发板上实现验证。



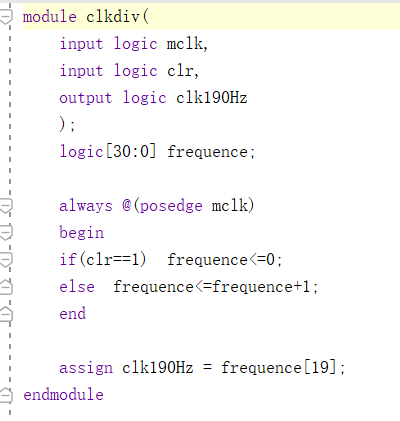
1. 实验方案

首先根据下面的设计流程图我们知道，1是要做一个clkdiv分频，使频率达到要求不至于过快（模仿上次实验内容的div写，稍作修改）。2是要做一个pulse去掉毛刺，同时实现按一下产生一个脉冲的功能（这部分按照老师最后一页ppt写即可）。3是最关键的部分：实现密码锁的功能，用状态机判定的方式实现（这部分得参照实验文档中的几个过程图，在第三部分实验分析仔细说明）

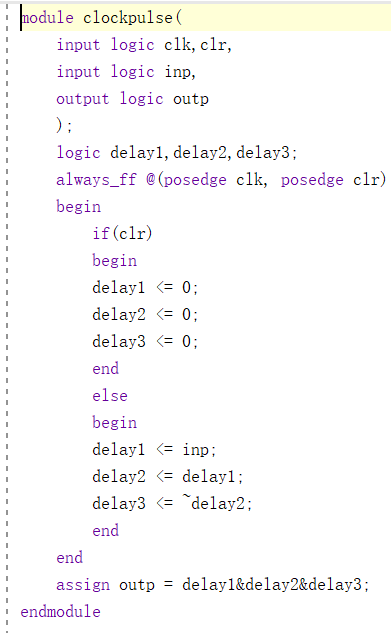


1. 实验分析

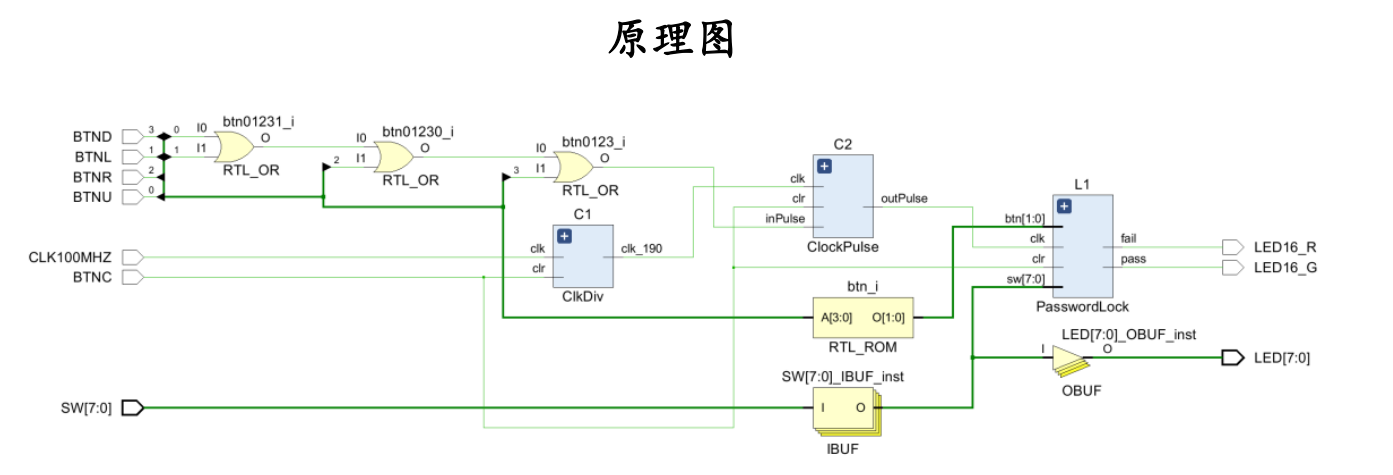
首先是clkdiv分频的代码：取19使之不至于过快，附近的几个数值应该也都可以。



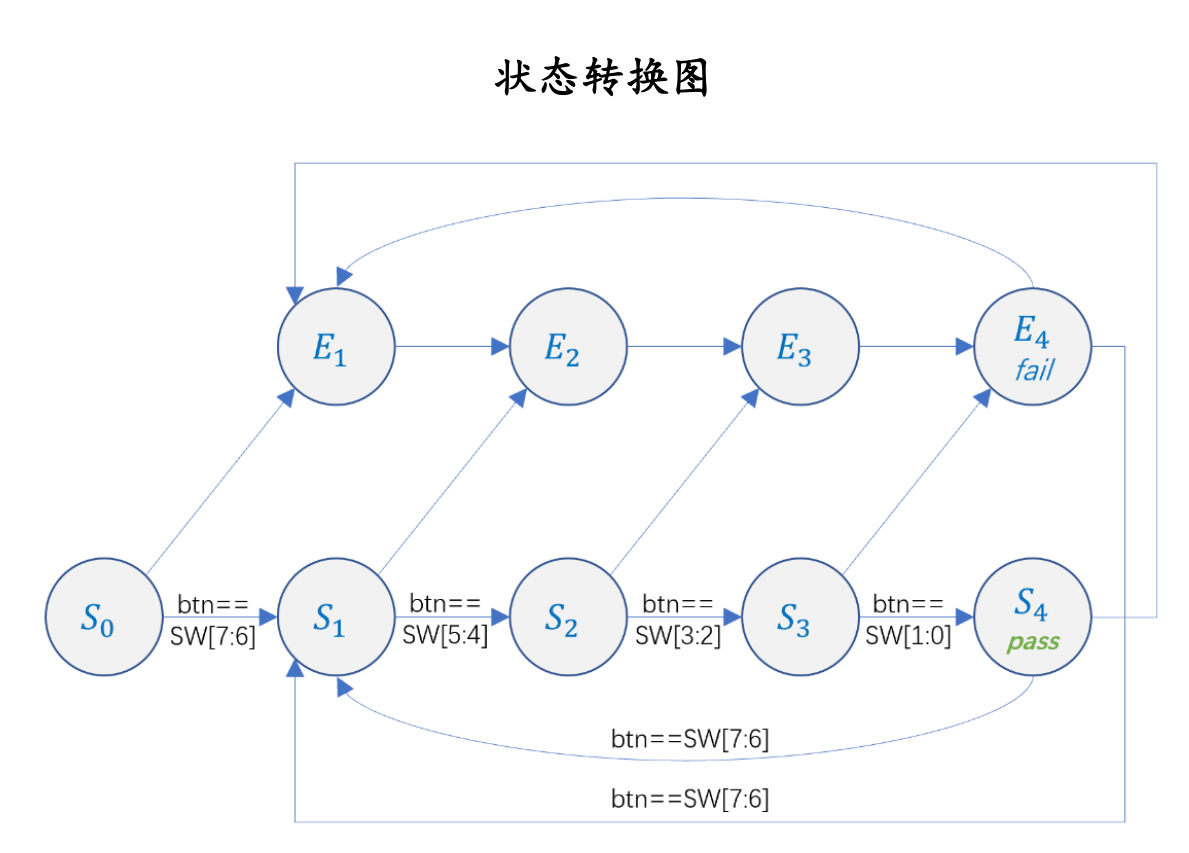
其次是模仿老师最后一页ppt写的脉冲去毛刺的代码：



接下来是实现密码锁的功能：



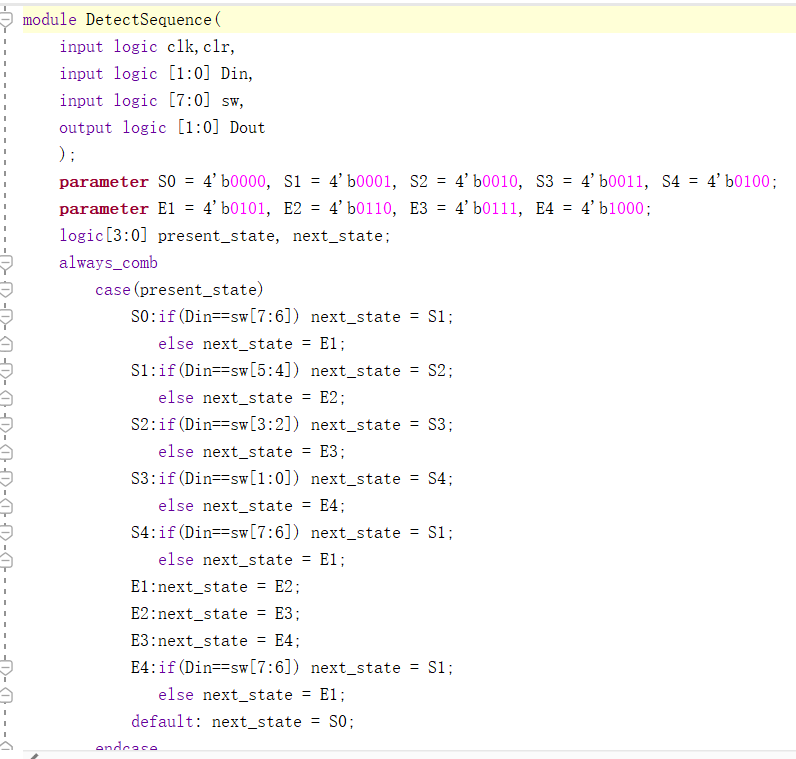
参考这个原理图我们知道，传值首先需要SW八个开关，两位的bnt记录按下的是哪个开关，根据规则：按钮 BTNU、BTNL、BTNR、BTND 分别代表十进制的 0、1、2、3，然后是clk和clr。实现代码部分参考老师的ppt和实验文档的状态机图片：



根据这个状态转换图仔细敲代码即可。

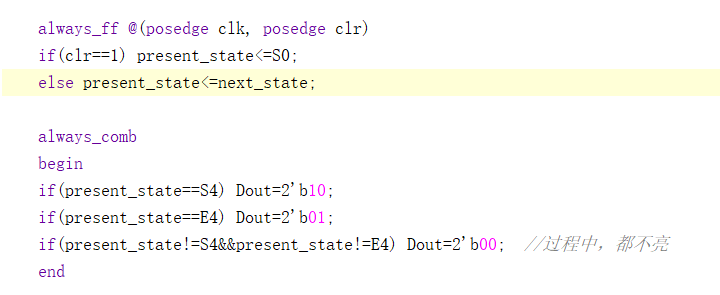
注意到在过程中是不亮的，只有最后走通或者走错才亮绿或红。且我们能保证四次按钮走四步后必然有个结果，因为从S0开始四步后必然要么走到S4（pass），要么走到E4（fail），而其它情况也就是说明在半路上，就置0，灯不亮。

代码如下：每一个状态点对应一个整数，组合逻辑实现每一个状态点的next走法。输入Din（顶层文件后面说明，代表对应的按键）和SW的两位比较有两条路。



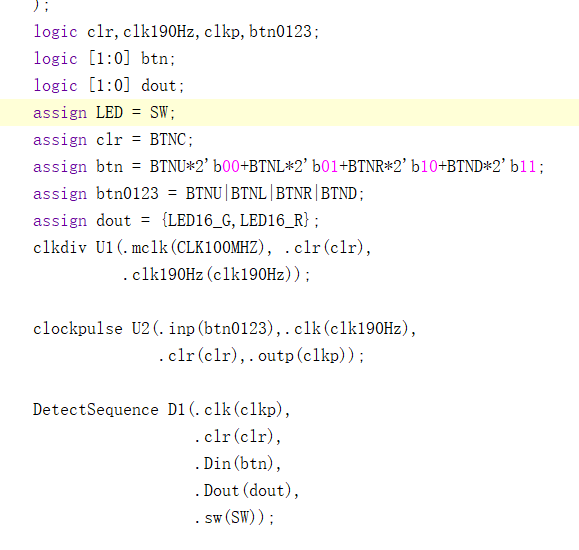
当clr==1时（也就是按下BTNC重置），如今状态强制改变为S0（initialize），其它情况，已经开始走，也就是present状态转移到next状态。 最后定义哪个灯该亮或者都不亮。

代码如下：



这三个功能函数实现后，最后就是需要在顶层文件将他们串起来（这里参考了老师ppt的代码）

Btn0123传入pulse功能函数去毛刺需要将四个或运算后一起传入，其次，最开始卡住我的是如何给btn[1:0]初始值，后面询问老师后马上明白了，btn也就是对应按键的数字取值。由于肯定每次只按一个按键，只有一个按键值是1，其余三个都是0，这样我们就可以逻辑比较清晰的给btn赋值了，见下面代码assign btn那一行（eg：按下BTNR，只有BTNR==1，其余==0，所以btn==1\*2==2（BTNR对应2））这样子轻松完成赋值。4🡪2像是编码器功能，之后将两位的btn传入密码锁的Din。 整体流程按照流程图，先div降频，再去毛刺产生脉冲，最后传到密码锁验证功能。代码如下：



Dout设为两位{G，R}，根据密码锁正确则0’b10，错误则0’b01，过程中则0’b00（都不亮），最终成功完成了实验。

开发板视频如下：（双击播放即可）



四、总结与思考

这次实验深入了解了有限状态机的代码写法，步骤比较多，难度不小，但理解了之后思路还是非常清晰的，在借鉴老师ppt代码的帮助下顺利完成了实验。