

数字逻辑与数字系统设计

实验大作业报告

(2020 年)

课程名称：数字逻辑与数字系统设计

任课教师：张英涛

作业题目：电子密码锁的设计

完成人：_

学号：

班级：

报告日期：2020 年 12 月 12 日

| | |
|------|--|
| 报告成绩 | |
| 教师评语 | |

目 录

| | |
|--------------------------|----------|
| 1. 设计要求 | 3 |
| 1.1 基本要求 | 3 |
| 1.2 附加功能 | 3 |
| 2. 工作原理及系统方框图 | 3 |
| 2.1 密码输入 | 3 |
| 2.2 计时器 | 3 |
| 2.3 计数器 | 3 |
| 2.4 存储器 | 3 |
| 2.5 比较器 | 3 |
| 2.6 系统方框图 | 4 |
| 3. 各部分模块具体功能及设计思路 | 4 |
| 3.1 密码输入模块 | 4 |
| 3.2 计时器模块 | 5 |
| 3.3 计数器模块 | 5 |
| 3.4 存储器模块 | 6 |
| 3.5 比较器模块 | 6 |
| 4. 调试过程 | 6 |
| 5. 设计结论 | 7 |
| 6. 设计心得与总结 | 7 |
| 参考文献 | 7 |
| 附录 | 8 |
| 附录 1 | 8 |
| 附录 2 | 9 |
| 附录 3 | 11 |

报告正文

1.设计要求

1.1 基本要求

- (1) 设计一个开锁密码至少为 4 位数字（或更多）的密码锁。
- (2) 当开锁按钮开关（可设置 8 位或更多，其中只有 4 位有效，其余位为虚设）的输入代码等于所设密码时启动开锁控制电路，并且用绿灯亮、红灯灭表示开锁状态。
- (3) 从第一个按钮触动后的 5 秒内若未能将锁打开，则电路自动复位并发出报警信号，同时用绿灯灭、红灯亮表示关锁状态。

1.2 附加功能

- (1) 记录输入密码的次数（最多可以输入 10 次）
- (2) 每一次输入密码时都进行倒计时，每一次允许的输入时间最多为 5 秒
- (3) 密码锁中的 4 位密码可以更改
- (4) 在一轮输入失败后（即输入十次均错误后）显示“EOF”

2.工作原理及系统方框图

2.1 密码输入

密码输入与修改密码使用相同四位输入端，采用将 4 位 8421 二进制码转换为 1 位十进制数，密码总计 4 位十进制数的方式工作（即需要分四次输入二进制码，一次输入四位二进制代码），采用按键不同区分输入密码与修改密码的工作模式。

2.2 计时器

计时器采用 FPGA 上的数码管显示，每次输入密码倒计时 5 秒（表示每次输入时最长时限是 5 秒），时间结束标志这次输入失败，计数器加一。

2.3 计数器

计数器采用 FPGA 上的数码管显示，从 0 开始计数，每匹配一次密码数字加一，最多可以输入 10 次（即计数器最大显示为 9），超出此界限标志输入密码失败，电子锁处于关锁状态。

2.4 存储器

存储器主要目的是储存密码以及暂存输入需要对比正确性的密码。

2.5 比较器

比较器主要目的是比较输入的密码与正确的密码是否匹配，根据匹配进行相应的操作。

2.6 系统方框图

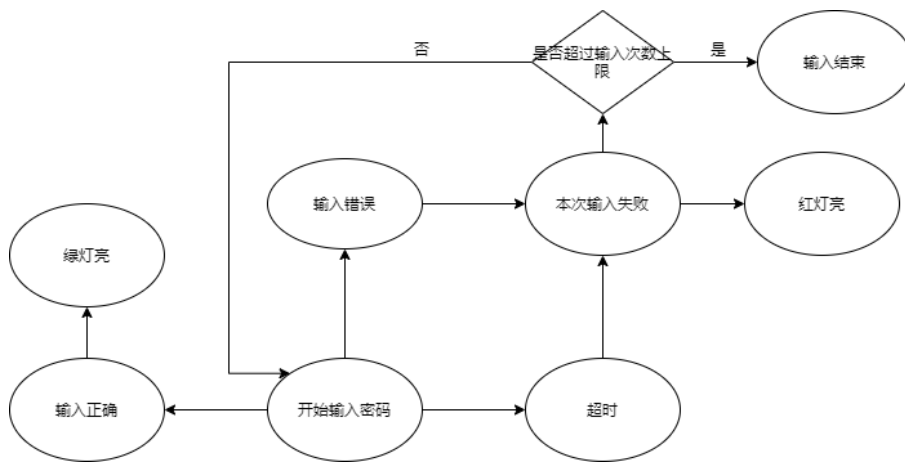


图 1.输入过程方框图

3.各部分模块具体功能及设计思路

3.1 密码输入模块

密码输入过程中采用一个四位的寄存器变量储存每一次输入的四位二进制数，使用一个标志位 flag 表示四位二进制数输入完成，当四位二进制数输入完成一次后，将该四位二进制数转换为 1 位十进制数，并按位次储存在另一个四位寄存器变量中。若此时输入密码过程还未完成则继续进行上述过程，直到储存十进制的寄存器变量存满（即存满四位）标志输入密码过程完成，此时进行比较模块。总结来看，就是将 16 位二进制数分四次转换为 4 位 10 进制数。同时，修改密码时也是采用这种输入方式，具体过程不再赘述，区别仅在于修改密码时不需要进行最后的比较以及储存的寄存器变量不同，其余工作方式完全相同。^[1]

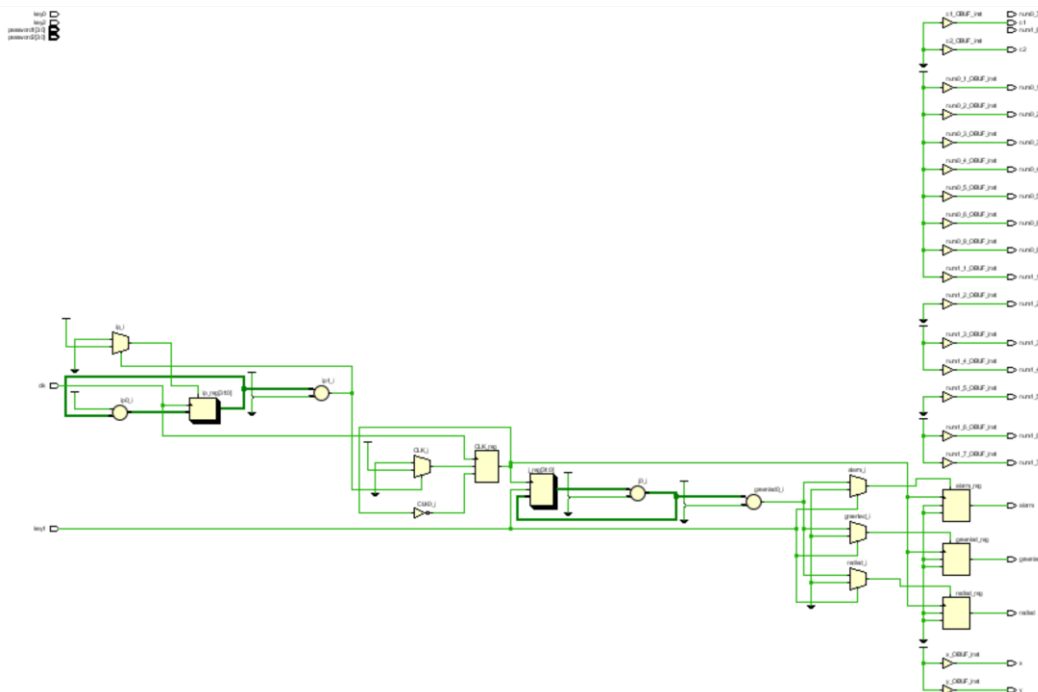


图 2.密码输入模块设计图

3.2 计时器模块

在密码输入过程中，一旦按下标志开始输入密码的按键（本实验中设置为 key2 变量），计时器模块开始工作。为了能让计时器模块的工作顺利进行，使用

系统中的时钟分频产生的时钟信号进行计时。^[2]采用的实验板时钟频率为 100M, 进行一定的分频后由于系统分频产生的时钟信号为每 0.2s 一个周期, 因此每接收到 5 个时钟信号计时器的数减一, 因此设计过程中设置了两个变量 a 与 b, a 用于计时器计数使用, b 用于标志一秒时间的变化 (使用系统分频产生的时钟信号计数), a 的初始值为 5, b 的初始值为 0, 每当 b 标志 1s 过去 (表现为 b 从 0 累加至 4), a 减一, 并将 b 置为 0, 表现出的效果即为显示出的时间减一。当时间从 5 减至 0 时标志本次输入失败, 时间重新置为 5, 计数器加一, 判断本轮输入是否结束。

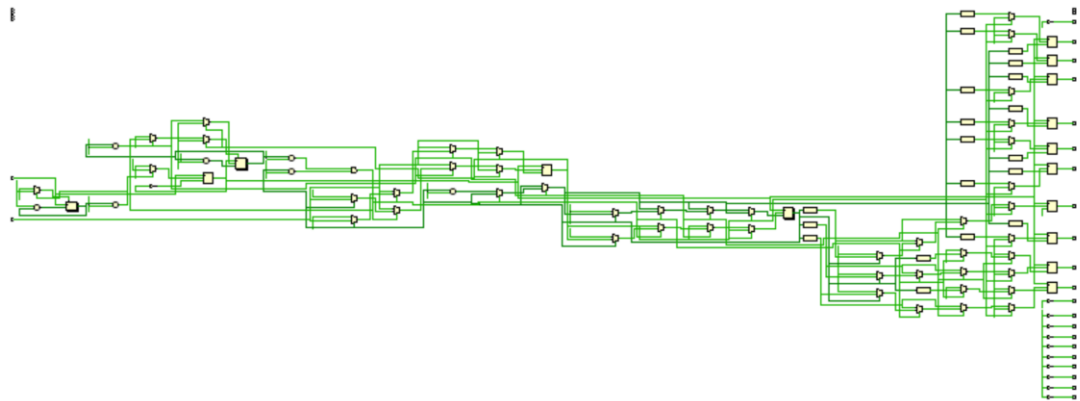


图 3.计时器设计图

3.3 计数器模块

计数器的功能是表示输入密码这一事件进行了多少次。每当输入一次密码就将计数器的数字加一, 当计数器的数达到 9 时表示输入次数达到上限, 如果这一次输入密码仍然不成功, 那么本轮输入密码失败, 电子锁锁死, 在 FPGA 上的表示为无法更改密码, 且红灯与警报灯亮并在数码管上显示 EOF 字样。计数器采用一个变量记录输入的次数, 每次输入完成时将此数加一。在显示过程中采用 FPGA 上的数码管, 对于 0~9 给出不同的显示方式, 则将数码管对应的管脚使能后显示需要显示的数字。

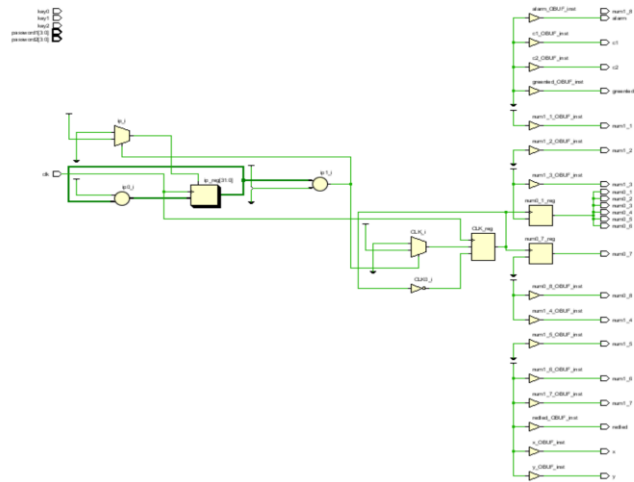


图 4.计数器设计图

3.4 存储器模块

存储器主要是为了暂存输入的密码以及存储系统中存在的密码, 各采用四位

寄存器变量储存。在实现过程中由于采用的是将 4 位 8421 码转换为 1 位十进制密码的方式，因此实现过程中还需要一个辅助的四位寄存器变量来储存每一次输入的四位二进制代码，每一次输入四位二进制密码完成时（需要一个标志位），将这一二进制数字转换为 1 位十进制密码储存在前述的寄存器变量中。在其中，标志位采用的是按键下降沿的形式，每当按下按键一次表示四位二进制数输入完成，开始执行将其转换为一位十进制数的任务。

3.5 比较器模块

比较器主要目的是比较输入的密码与原本储存在系统中的密码是否匹配，在代码设计过程中表现为将输入的密码与储存的密码逐位进行比较。如果密码匹配则绿灯亮，密码锁锁死；如果不匹配则红灯与警报灯亮，绿灯灭，计数器加一。如果进行下一次匹配则比较器继续工作。

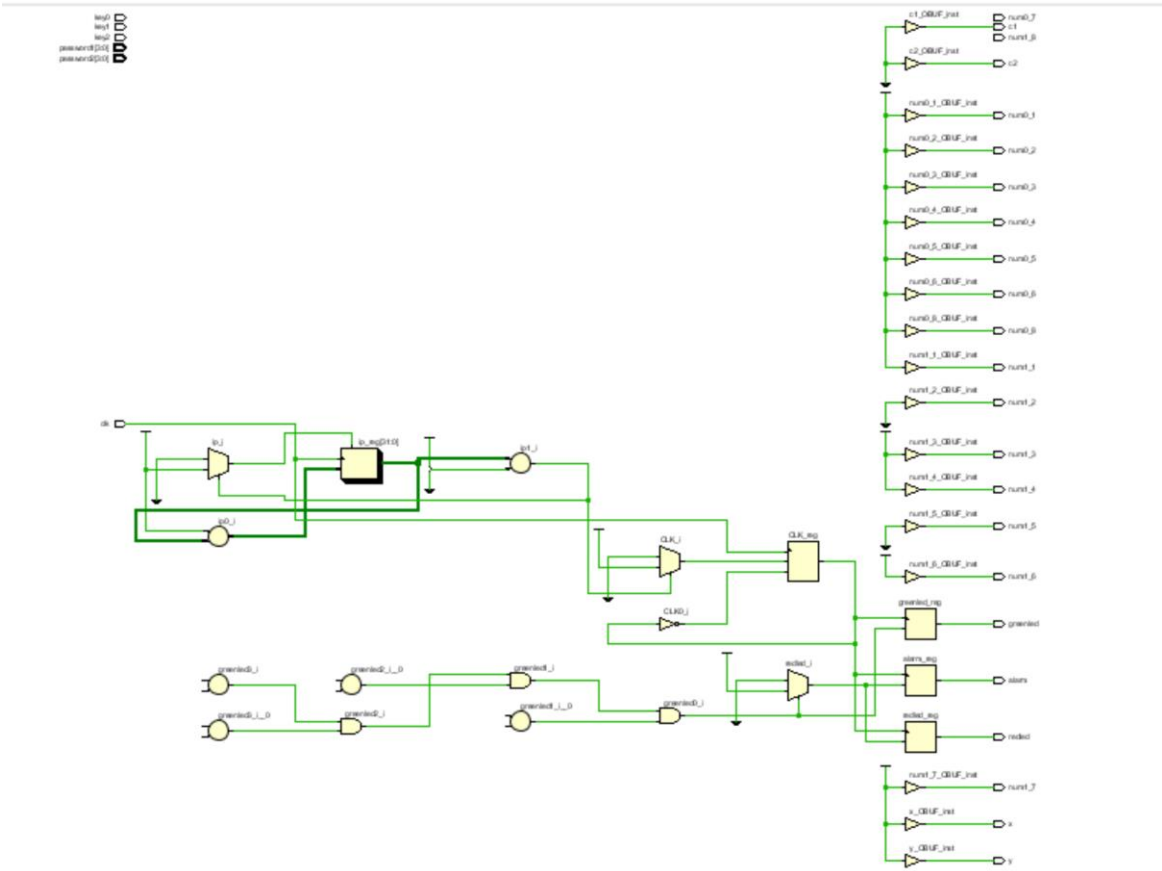


图 5.比较器设计图

4.调试过程

调试过程主要是将每一个模块设计完成后单独进行测试，但是将模块合起来的时候出现的最严重的问题就是在进行综合的时候代码报错，经过对报错信息的阅读以及查阅一定的资料，发现报错的原因是软件认为在设计的过程中出现了同步时序与异步时序混合使用的现象，完整报错信息如下：

```

1 [Place 30-574] Poor placement for routing between an IO pin and BUFG.
2   If this sub optimal condition is acceptable for this design,
3   you may use the CLOCK_DEDICATED_ROUTE constraint in the .xdc file
4   to demote this message to a WARNING.
5   However, the use of this override is highly discouraged.
6   These examples can be used directly in the .xdc file to override this clock rule.
7       < set_property CLOCK_DEDICATED_ROUTE FALSE [get_nets reset_IBUF] >
8
9   reset_IBUF_inst (IBUF.0) is locked to IOB_X0Y82
10  and reset_IBUF_BUFG_inst (BUFG.I) is provisionally placed by clockplacer on BUFGCTRL_X0Y1

```

图 6.报错完整信息

这一现象主要是因为最初的设计过程中，系统使用了时钟信号（同步时序），同时在写入密码的过程中使用了按键下降沿控制输入的方式（异步时序），在查阅了官方手册以及相关资料^[3]后发现这样的设计是有问题的，这种报错标志着 I/O 引脚与 BUFG 之间的次佳路由连接。这是因为此 I/O 并非具有时钟功能的引脚，因此 I/O 与 BUFG 之间没有专用的时钟路由^[4]。因此在后续过程中将按键下降沿更改为 key 变量等于 1 时表示按键下降沿，由于实验板的设计为按键按下时输出为 1，且本实验中设计的时钟周期为 0.2s 为一个时间周期，时间周期较为合适，既不容易出现一次按下识别为多次按下的问题，也不容易出现按下时无法识别的问题，较好解决了这一问题。同时，在调试过程中还曾出现的问题有在需要数码管显示数字时数码管没有使能、在计时器达到 0 时没有对密码锁进行暂时的锁死导致始终识别到时间变量为 0，计数器不停加一，与实验需求不符等错误，最终都一一改正。

5.设计结论

本设计主要实现了输入密码进行匹配，限制输入次数与每次输入时间的功能，并在这些基本功能上进行了一定的优化，形成了一个基本完善的电子密码锁。但是与事实上可以使用的密码锁还是存在很大的差距，以后有时间可以尝试将本设计制作成一个真实可用的电子密码锁。

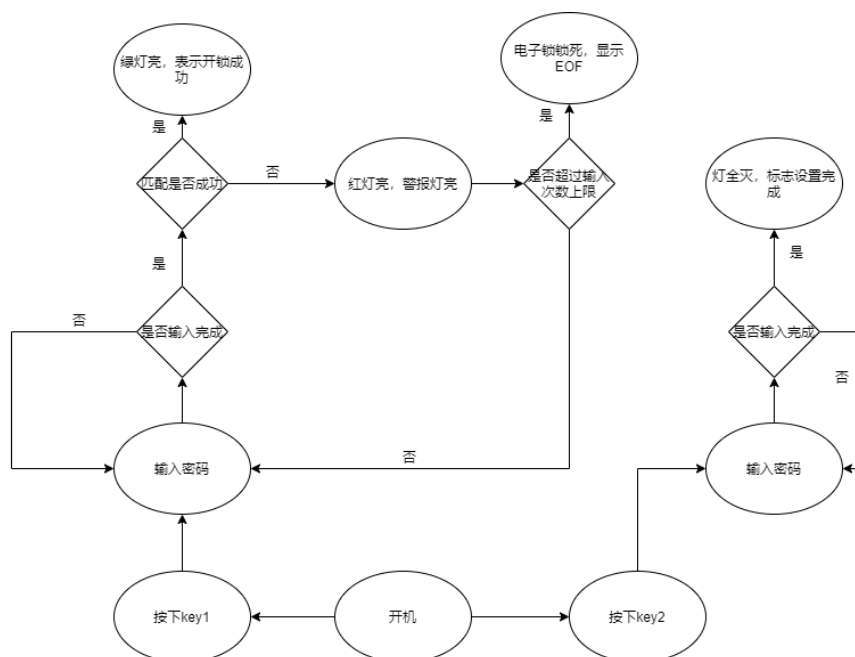


图 7.主要工作流程

6.设计心得与总结

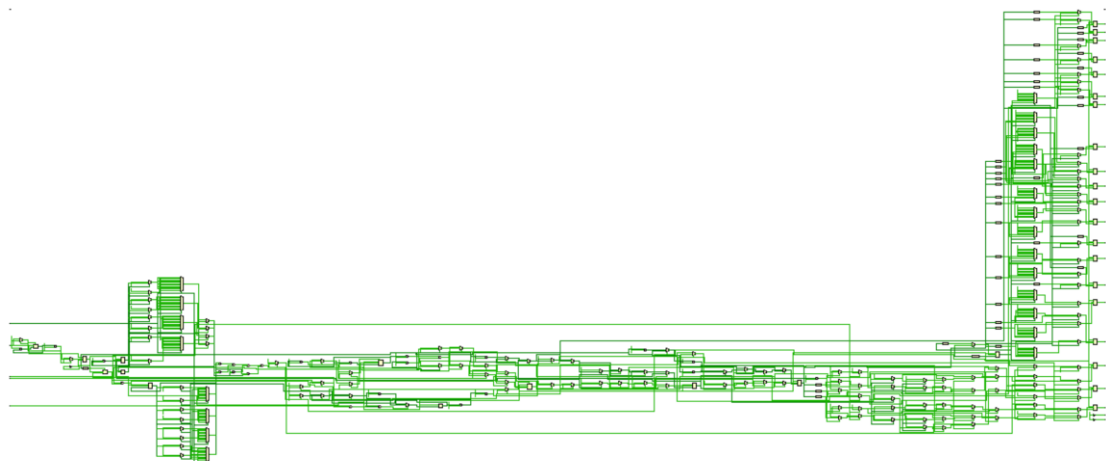
- 1.设计过程中一定要分模块测试功能是否正常,当测试到每一个模块功能都正常之后再各模块整合为一个模块。否则若将所有模块一起测试时容易因为某一模块的错误引起另一模块的错误,在修改的时候定位问题出现的位置出现一定的困难。本次设计规模并不是很大体现不是很明显,如果设计规模再大一些很可能出现无法寻找到出现的错误究竟是哪一个模块的现象。^[5]
- 2.要善于使用网络资源,当编译器出现一些原因不明的报错时可以直接将报错信息复制到搜索引擎中进行检索,这样更容易找到错误出现的原因,从而定位到具体的模块中进行修改,比自行寻找错误原因显得更加高效^[6]。特别是当编译器出现一些比较不常见的报错时自行寻找错误可能需要很长时间或者根本无法寻找到错误,一些由于忽略设计需要遵从的细节的报错在网络上可以很快找到出现问题的原因,但是如果不知道需要遵守这些细节的话有可能根本不会发现是这个部分出现的问题。
- 3.设计过程要基本遵循自顶向下的设计过程,同时需要在设计开始之前有一个基本完整的规划,不至于使得设计显得比较混乱。本次设计过程中出现过多次当设计到某一模块的时候才想起来还需要写另一个模块的某一个功能,这样就导致设计的过程中较为混乱。在设计之初就要基本设想好每一个模块基本需要实现哪些功能,尽量避免想到哪里设计到哪里的设计方法。

参考文献:

- [1] <https://blog.csdn.net/qinglingLS/article/details/74278856>
- [2] <https://www.cnblogs.com/zhangxianhe/p/11083208.html>
- [3] https://blog.csdn.net/weixin_43162818/article/details/89434960
- [4] <https://www.xilinx.com/support/answers/64452.html>
- [5]严浦洲.基于 Verilog HDL 的多周期 CPU 设计与实现[J].数码世界,2020(12):31-32.
- [6]刘焰强,戚正伟,管海兵.FPGA 加速系统开发工具设计:综述与实践[J].软件学报,2020,31(10):3087-3099.

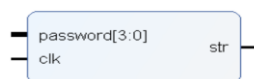
附录

附录一.总体设计图

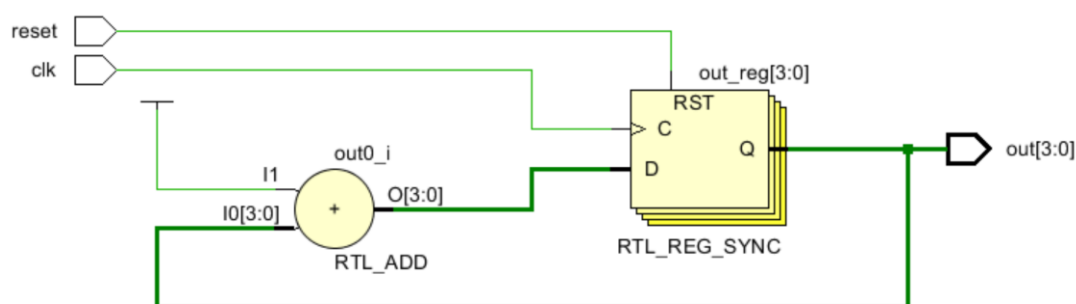


分模块设计图如下

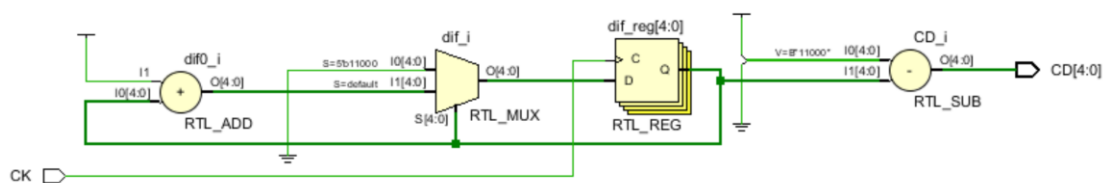
1.1 编码器 在输入密码时使用，采用同步控制方式，输入端为 4 位，输出端为 1 位，在每一次输入密码中输入 4 次 4 位二进制代码，此编码器能将每一个 4 位二进制数转换为 1 位 10 进制数，实现进制的转换。



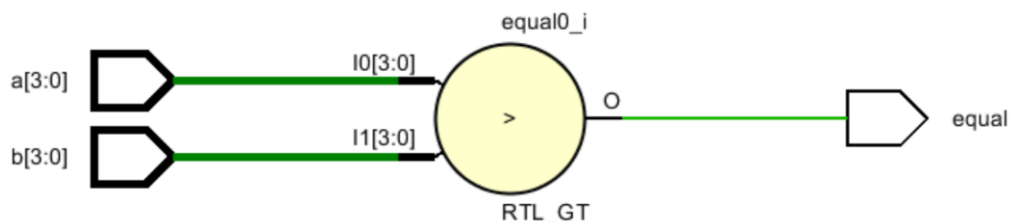
1.2 计数器 一个使能端，一个时钟端，输出端为 4 位二进制数，在本实验的设计中主要是将其从 0 计数到 9 后将密码锁锁死。



1.3 计时器 在实验中每一次输入密码时从 5 倒计时计数到 0，输入为时钟信号，输出为 5 位二进制数，但是在本实验中只使用其中的 3 位表示 0-5 的数。

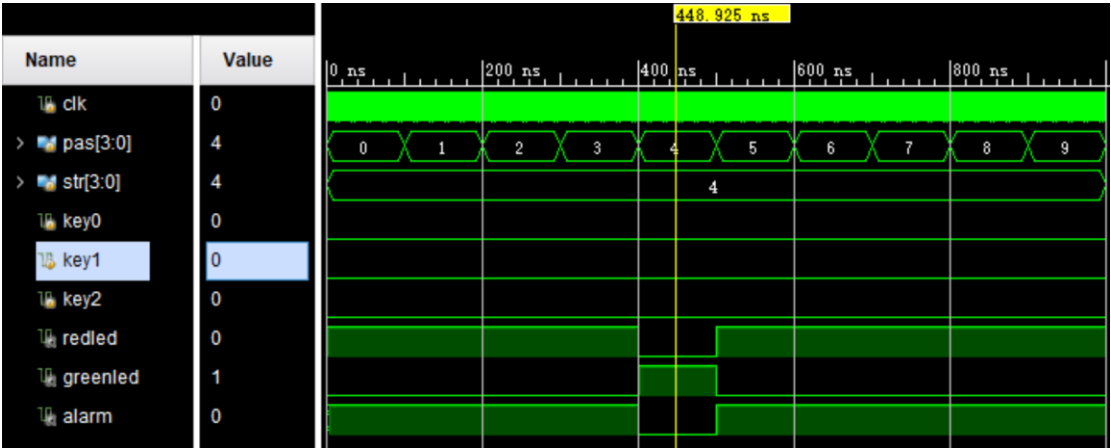


1.4 比较器 输入为两个四位变量，输出为一位二进制数，表示四位变量是否相等。

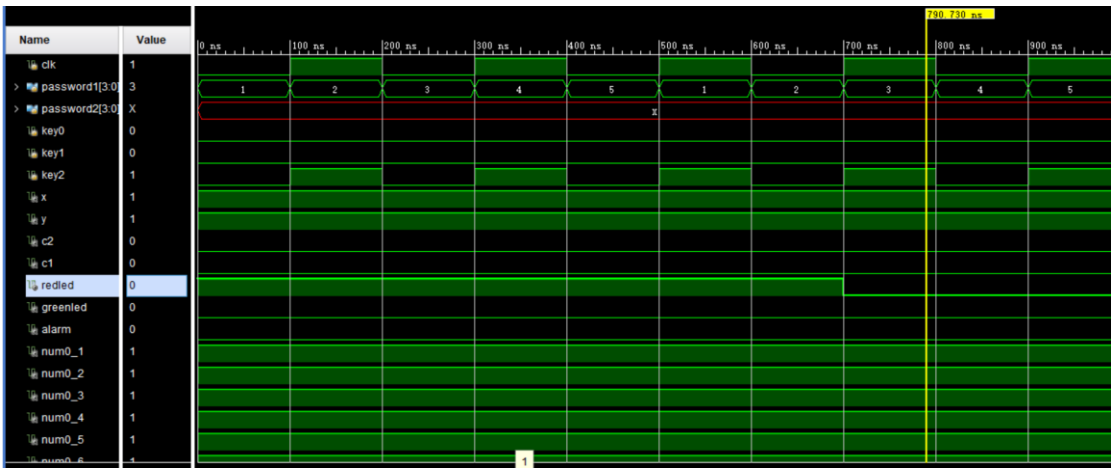


附录二.仿真图

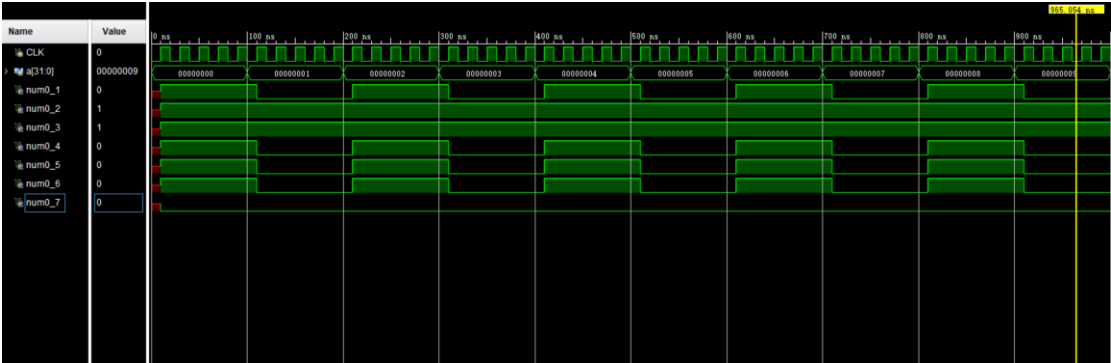
2.1 比较器部分仿真



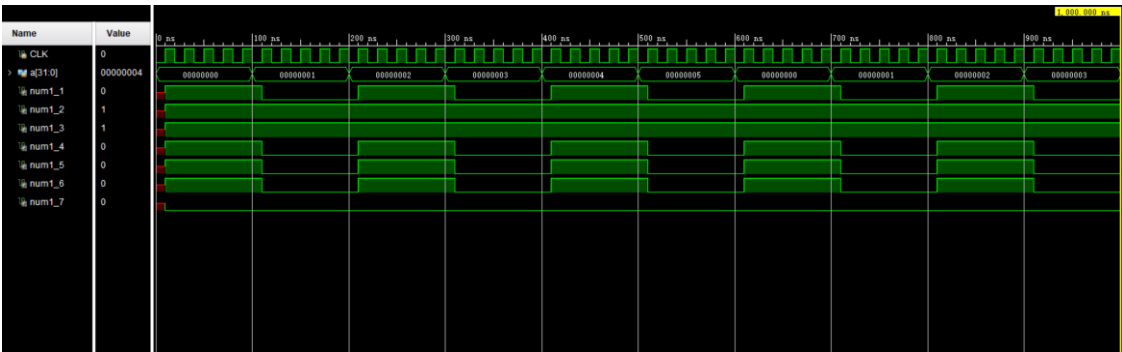
2.2 密码输入部分仿真



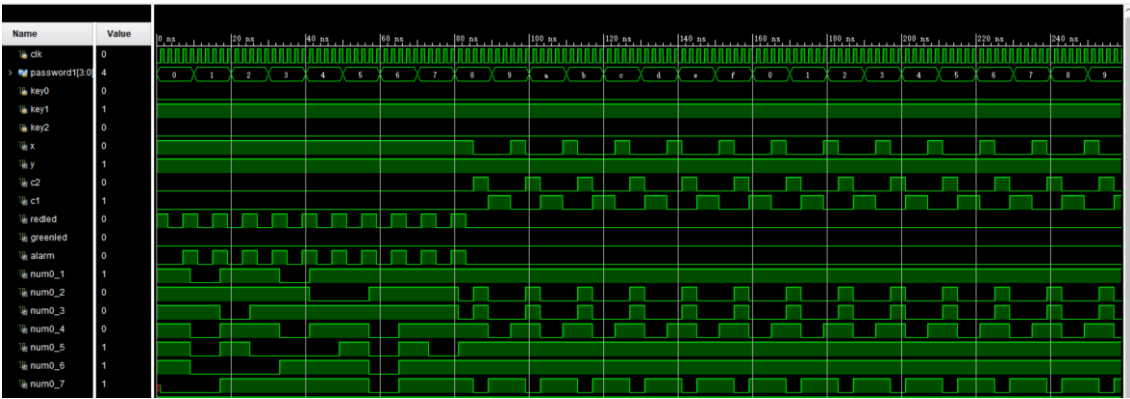
2.3 计数器模块仿真



2.4 计时器模块仿真



2.5 总体设计仿真



附录三：小组各成员所做工作说明

小组成员：郑晟赫

工作说明：小组只有一人，因此所有工作为一人完成。