

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：李琼

作业题目：电子密码锁的设计

完成人：Youngsc

学号：

班级：

报告日期：2020 年 12 月 18 日

报告成绩	
教师评语	

目 录

1. 设计要求	4
1.1.....	4
1.2.....	4
1.3.....	4
2. 工作原理及系统方框图	4
2.1 工作原理.....	4
2.2 系统方框图.....	5
3. 各部分模块具体功能及设计思路	5
3.1 主程序.....	5
3.1.1 具体功能.....	5
3.1.1 设计思路.....	6
3.2 数码转换模块.....	6
3.2.1 具体功能.....	6
3.2.1 设计思路.....	6
3.3 数码管显示模块.....	6
3.3.1 具体功能.....	6
3.3.1 设计思路.....	6
3.4 数值比较模块.....	6
3.4.1 具体功能.....	6
3.4.1 设计思路.....	6
3.5 状态灯显示模块.....	6
3.5.1 具体功能.....	6
3.5.1 设计思路.....	6
3.6 倒计时模块.....	6
3.6.1 具体功能.....	6
3.6.1 设计思路.....	6
3.7 倒计时显示模块.....	6
3.7.1 具体功能.....	6
3.7.1 设计思路.....	6
3.8 警示灯和流水灯模块.....	7
3.8.1 具体功能.....	7
3.8.1 设计思路.....	7
4. 调试过程	7
4.1 错误.....	7
4.2 主程序.....	7
4.3 数码转换模块.....	8
4.4 显示模块.....	8
4.5 倒计时模块.....	8
4.6 警示灯和流水灯模块.....	8
5. 设计结论	8
5.1 综述.....	8
5.2 附加功能.....	8

6. 设计心得与总结	9
参考文献	9
附录	9
附录 1.....	10
附录 2.....	11
附录 3.....	11

报告正文

1 设计要求

- 1.1 设计一个开锁密码至少为 4 位数字（或更多）的密码锁。
- 1.2 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。
- 1.3 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。

2 工作原理及系统方框图

2.1 工作原理

- 2.1.1 该密码锁有四位密码，每位密码可以为 0 到 7 中的任意一个数字，这八个数字分别与开发板下方的八个开关相对应，从左到一依次为 0 到 7，初始密码为 0000。
- 2.1.2 每次从左向右依次输入密码，对某一个开关，快速向上推动并相下推动为依次输入，若在某开关已经在开状态时推动另一开关，系统无任何反应，直到所有开关均处于关状态时才可输入下一位密码或结束输入。
- 2.1.3 密码锁在位于锁止状态时，红灯亮，绿灯灭，开锁状态时，红灯灭，绿灯亮，修改密码状态时，红灯绿灯同时亮。
- 2.1.4 在输入第一位密码的同时，系统开始进行五秒倒计时，若倒计时结束之后仍未输完所有密码，则已输入密码自动清除，警示灯连续闪烁以示警告。
- 2.1.5 若输入密码错误，则已输入密码自动清除，警示灯连续闪烁以示警告。
- 2.1.6 若输入密码正确，则密码锁开启，下方流水灯开始工作。
- 2.1.7 在密码锁开启的状态下，按动修改密码键，系统进入修改密码状态，指示灯亮时表示当前处于输入新密码状态，指示灯熄灭则新密码输入完毕。
- 2.1.8 在密码锁开启的状态下，按动锁定键，密码锁锁定，若在输入新密码时按动锁定键，则密码不会被修改。
- 2.1.9 输入密码，修改密码，倒计时的数字均会在数码管上显示，期中倒计时精确到小数点后两位。
- 2.1.10 开锁且未进入修改密码状态下，无法进行任何输入。

2.2 系统方框图

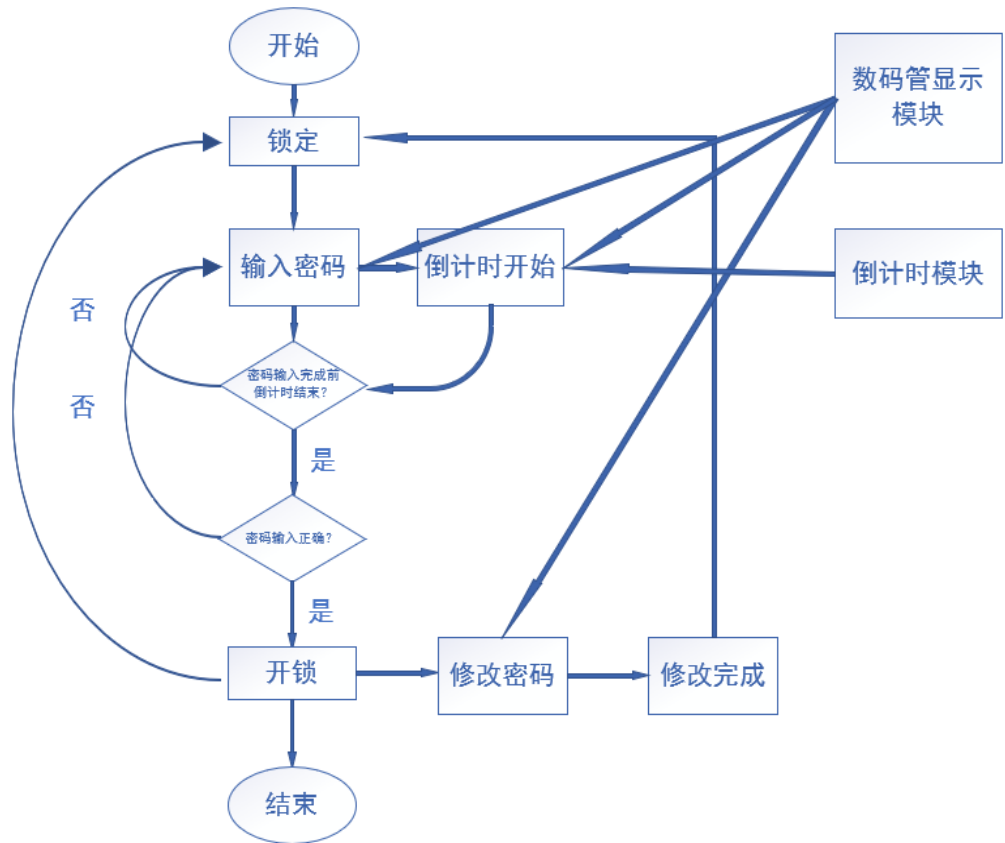


图 1 系统方框图

3 各部分模块具体功能及设计思路

3.1 主程序

3.1.1 具体功能

对时钟信号进行分频，读取输入的开关状态，决定红灯、绿灯、状态灯、指示灯的信号，输入密码解锁，修改密码，锁定密码锁调用各小部分模块。

3.1.2 设计思路

在每一个时钟分频信号的上升沿时刻，判断输入开关的状态是否存在为开的开关，同时，如果锁止信号 reset 为 1，则将密码锁锁定，将 reset 复位为 0。紧接着判断当前是否按动修改密码按钮且是否开锁，若按动修改密码按钮且现在处于开锁状态，则系统进入修改密码状态，同时各状态指示灯进行改变，First 记录是否之前书如果密码，若输入过，则用来决定是否输出密码输入错误的警示灯或密码输入正确的流水灯。如果当前处于锁止状态，则输入的四位密码在输入完毕后进入 Compare 模块进行比较，若输入密码完毕前倒计时结束，则已输入密码全部清除重新输入且重新开始倒计时；如果当前处于修改密码状态，则输入四位密码完毕后，将四位当前密码同时输入到记录正确密码的变量中。

3.2 数码转换模块

3.2.1 具体功能

将当前开关的状态转换为可以让数码管显示出对应数字的状态。

3.2.2 设计思路

转换前的数字仅有一个 1，其所在的位置代表应该显示的数字，再将对应数字在数码管上应亮的部分变为 1，不该亮的部分转换为 0，将结果返回。

3.3 数码管显示模块

3.3.1 具体功能

将输入的密码或修改的新密码通过数码转换模块转换后，放在数码管的前四位输出。

3.3.2 设计思路

由于前四位数码管使用同一个信号输入，为了让其显示不同的数字，我们通过极快地调整使能端和对应的输入信号，从而达到类似四位同时输出不同数字地效果。

3.4 数值比较模块

3.4.1 具体功能

比较输入的密码和正确密码是否相同。

3.4.2 设计思路

将正确密码和当前密码同时传入模块，并一一比较，传回信号表示是否相同。

3.5 状态灯显示模块

3.5.1 具体功能

通过不同的灯情况来表明当前所处的不同状态。

3.5.2 设计思路

传入当前是否开锁以及是否处于修改密码状态，若处于修改密码状态且密码锁为开启状态，红灯绿灯同时亮。否则若锁为开启状态，绿灯亮红灯灭，否则，红灯亮绿灯灭。

3.6 倒计时模块

3.6.1 具体功能

实现五秒钟倒计时，并在倒计时结束后重置密码锁。

3.6.2 设计思路

将三位分别重置为 5、0、0，将当前时钟信号重新分频，每 0.01 秒为一个上升沿，每一个上升沿小数点后第二位减一，并在当该位不够减后，小数点后第一位减一，该位不够减后，小数点前一位减一，直到三位全部为 0 为止，倒计时结束。

3.7 倒计时显示模块

3.7.1 具体功能

将当前的倒计时时间输出到数码管的后三位。

3.7.2 设计思路

同数码管显示模块的设计思路相同。

3.8 警示灯和流水灯显示模块

3.8.1 具体功能

点亮警示灯，流水灯和密码修改指示灯。

3.8.2 设计思路

将时钟信号重新分频，在每一个上升沿时，若是警示灯，则全部取反，若是流水灯，则将使能端右移一位，若当前密码修改状态，则让指示灯取反。

4 调试过程

4.1 错误

4.1.1 在两个 always 里不同同时修改同一个 reg 变量，需要将其整合到同一个 always 里。

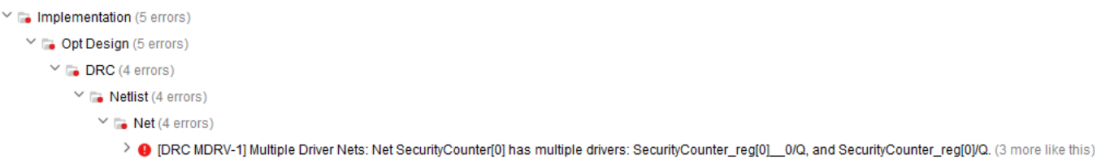


图 2 错误信息

4.1.2 无法修改 wire 变量，需要建立新变量。

4.1.3 reg 变量未初始化

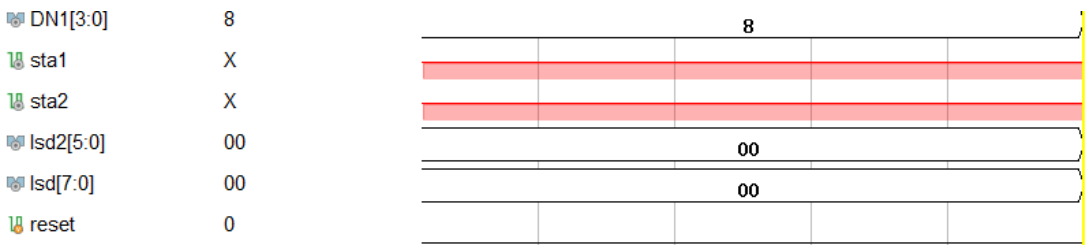


图 4 仿真波形图

4.1.4 时钟速度不能太快，不然在拨动开关时容易出问题。

4.1.5 负责计数器的 reg 变量一定要位数开够，否则会溢出。

4.1.6 仿真截图

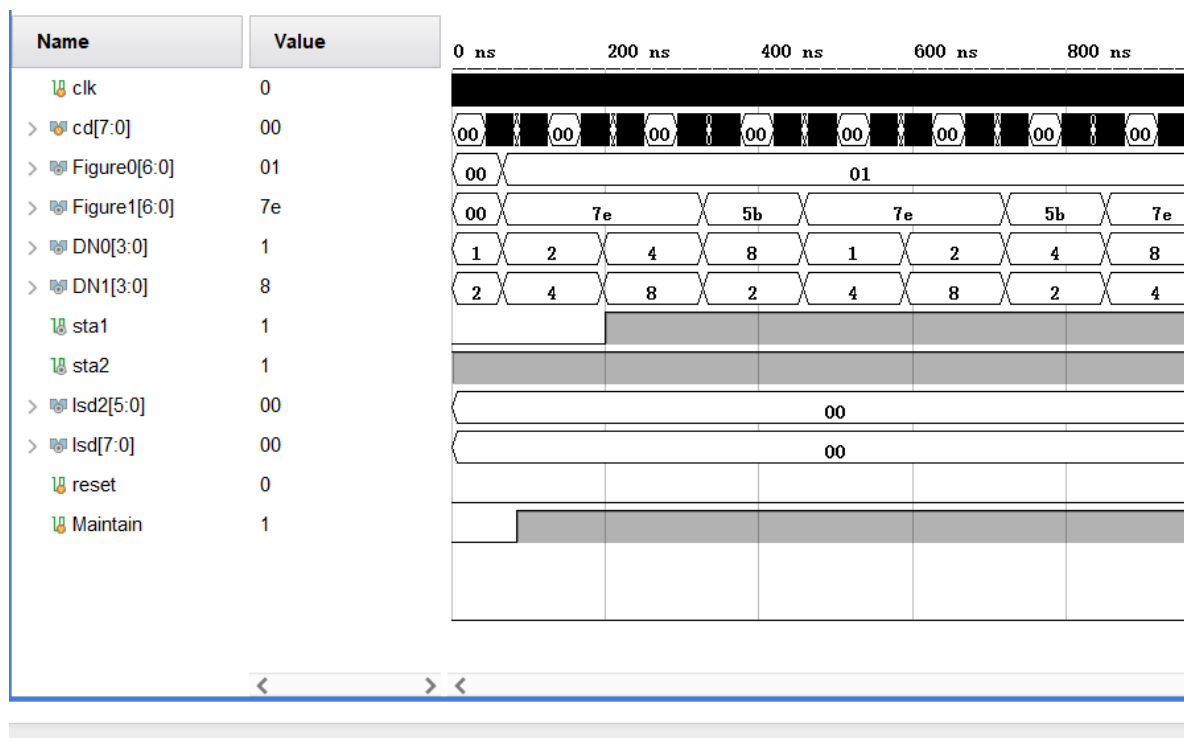


图 5 仿真波形图

4.2 主程序

修改密码时输入最后一个新密码和同时给四个记录位同时赋值新密码不能在同一个上升沿中。

4.3 数码转换模块

注意不同个数的状态下变量位数大小不同，

4.4 显示模块

分频要合适，不然容易出现闪烁的情况。

4.5 倒计时模块

分频计算需要合适，才能精确到秒。

4.6 警示灯和流水灯模块

变量位数需要开够，不然会出现溢出现象。

5 设计结论

5.1 综述

我设计了一个四位密码的电子密码锁，可以通过拨动开关实密码的输入，同时输入的密码可以在数码管的相应位置显示。若密码错误，则警示灯闪烁，并重新输入密码；若密码正确，则密码锁开启，同时流水灯开启。每次输入密码时间限制在五秒内，若倒计时结束密码没有输入完成，输入密码过程结束，全部复位，已输入密码清除。倒计时的时间会在数码管上显示，并精确到小数点后两位。在开锁状态下，按动锁定按钮，密码锁重新锁定。在开锁状态下，点击修改密码按钮，并输入新的四位密码，则密码修改。若此时新密码没有输入完成按动锁定按钮，则系统锁止，密码未修改。

5.2 附加功能

5.2.1 当前输入密码各位数可以在数码管上显示

- 5.2.2 倒计时可以在数码管上显示
- 5.2.3 倒计时精确到小数点后两位
- 5.2.4 密码可以进行修改保存
- 5.2.5 警示灯和各状态表示灯更加丰富

6 设计心得与总结

通过此次大作业的完成，我更了解了时序电路的进行过程和运行原理，以及多模块电路的组合关系以及运行过程，我对使用 Verilog 的使用更加熟练，同时对开发板更加地了解。硬件设计的数字实现方式十分注重细节，我需要仔细理清思路，梳理逻辑，通过不停地调试、仿真、烧板来调整功能，使得功能逐渐完善，最终形成手中的电子密码锁。通过这次经历，锻炼了我的逻辑思维能力和硬件设计能力。

参考文献：

- [1] 叶淑群,陈鸿鹏,梁士坤.实现基于 FPGA 的硬件算法加速器[J].宝鸡文理学院学报(自然科学版),2006(02):154-155+168.
- [2] 丁黄胜,陆生礼,田渊,吴旭凡.基于FPGA的HMAC_SHA1_96算法设计与实现[J].半导体技术,2003(06):25-28+32.
- [3] 房宝光. FPGA 在 OCD 系统中模拟算法的加速应用[D].电子科技大学,2013.
- [4] 严国萍.通信电子线路[M].北京:科学出版社,2015:142-155.

附录

附录 1:

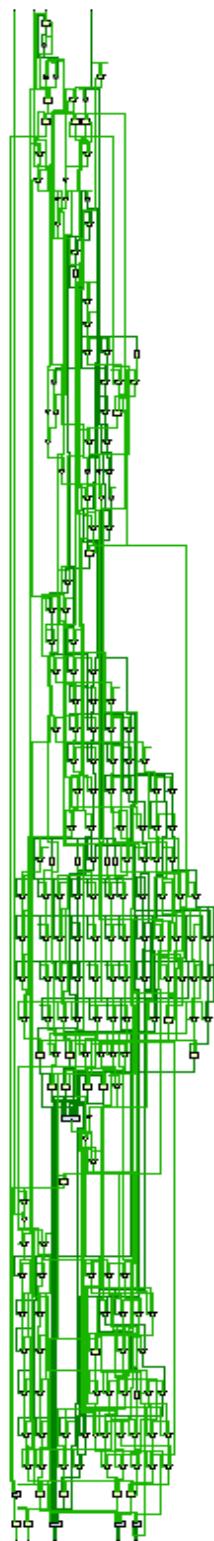


图 6 整体设计图

附录 2:

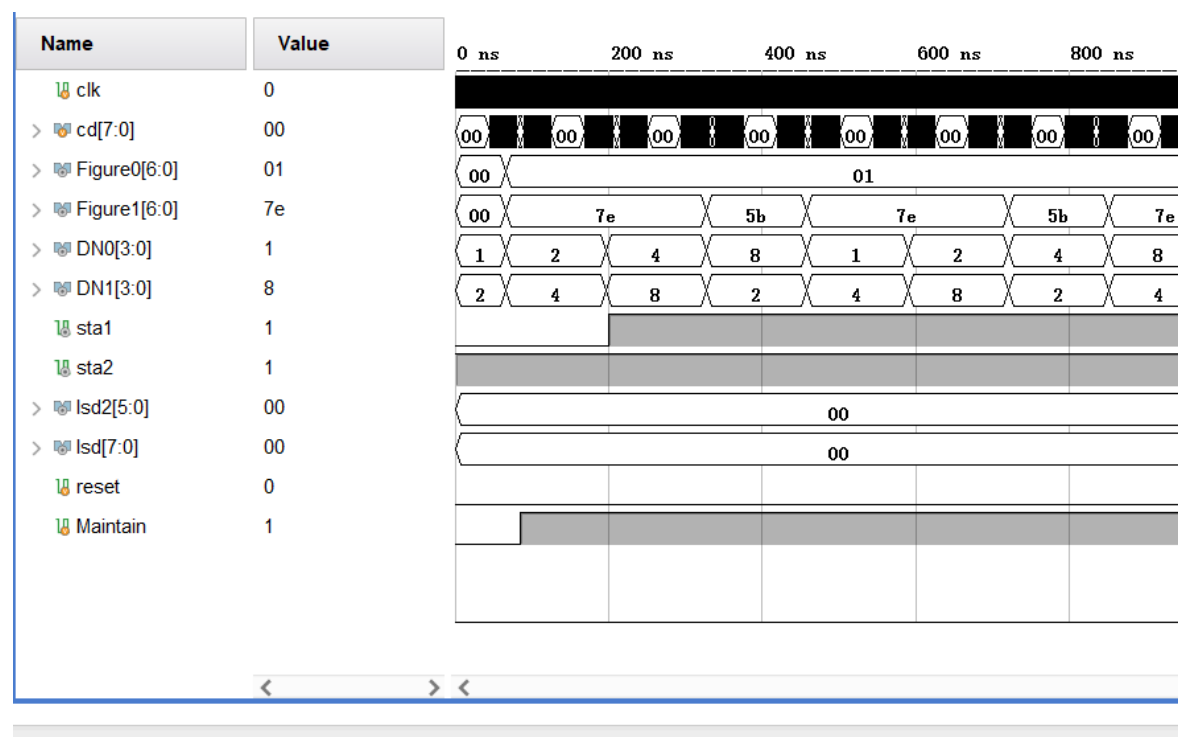


图 7 系统仿真波形图

附录 3:

整个系统设计及报告设计均为袁野独立完成。