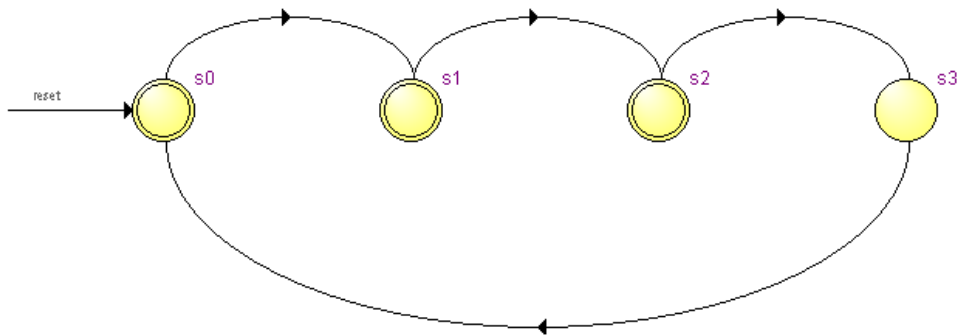


EDA 第二次作业报告

一. 预习任务

(一) 任务一:

1. 数码管扫描状态转换图 (状态机)



(二) 任务二

1. 矩阵键盘工作原理

key_c[3:0]循环输出: 0111、1011、1101、1110, 同时从 key_r[3:0]开始输入, 这样的配合之下 key_c 和 key_r 同时出现 0 的位置就是按键被按下的位置。

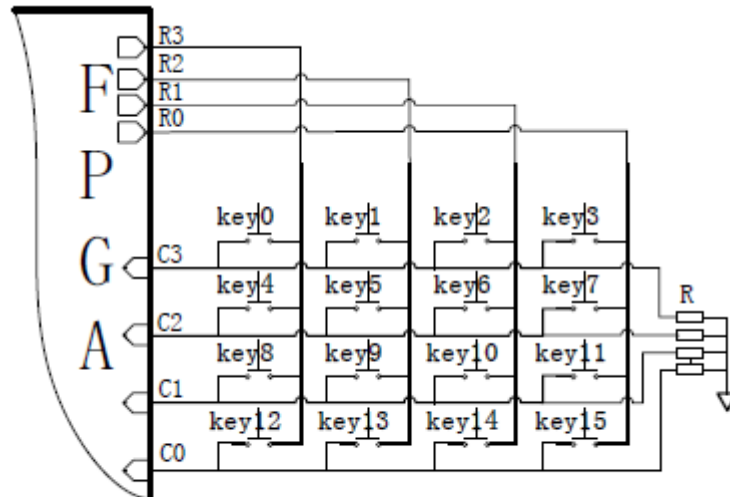


图 1.4 4*4 矩阵键盘原理图

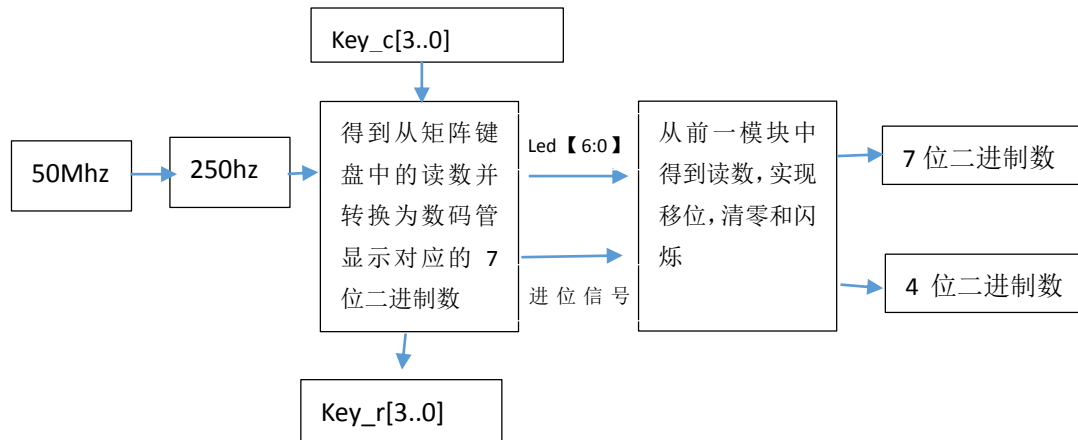
2. 任务分析及电路总体框图

任务分析:

- 1) 任务一需用分频器和状态机完成扫描电路
- 2) 任务二首先需要完成从矩阵键盘读数, 第二步利用任务一的扫描电路完成四个数码管的分别显示, 再完成数码管移位, 紧接着是清零, 最后利用再一次

分频完成闪烁。具体模块设计想法见设计思路。

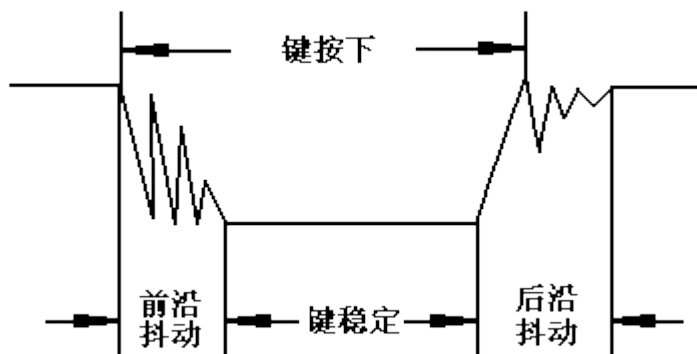
电路总体框图



3. 各功能模块的频率

- 1) 系统时钟：从晶振得到 50MHz 经过第一个分频器变为 250Hz 作为该系统时钟
- 2) 按键读取模块采取系统时钟运行，另内部有对系统时钟的 10 分频设置用于防抖，所以按键输入应为 25Hz。
- 3) 移位和去清零部分都采取系统时钟，但在确认输入信号是亦采用了对系统时钟的 10 分频。
- 4) 数码管的正常扫描采用系统时钟 250Hz，在闪烁是采用将其二分频的 125Hz，且占空比由 50%变为为 40%。

4. 按键抖动



按键所用开关为机械弹性开关，当机械触点断开、闭合时，一个按键开关在闭合时不会马上稳定地接通，在断开时也不会一下子断开。因而在闭合及断开的瞬间均伴随有一连串的抖动。抖动时间的长短由按键的机械特性决定，一般为 5ms~10ms。

消抖原因：如果采用边沿触发作为信号，会检测到一系列的非必要的信号，系统误读多余信号，影响正常的电路设计。

防抖方法：

硬件：可用 D 触发器作为防抖。

软件：在十倍时钟信号的计数是分别读取两次进位的七位二进制数和进位信号，若都确

认按键且输入相同才认为按键成功。

二. 设计思路

(一) 任务一

1. 分频器

系统时钟为 50Mhz，而使用的时钟频率应该在几百 hz 左右，我的分频器在每个系统时钟 clk_50M 的上升沿将 counter 变量的计数加一，在 counter>199999 时将它置零。而在 counter>999999 时将分频后所用的时钟 clk_250 置为高，在 counter<999999，这样就实现了一个 200000 分频。

2. 状态机实现数码管扫描

在第一次作业中已经完成了全加器和数码管的选通，只是当时使用两个拨码开关实现选通，而这次作业中的数码管扫描我利用一个 4 个状态的状态机来实现对每个管的选通，而状态使用分频后的 clk_250 的上升沿作为转换到下一状态的判定，这样每个时钟信号都会对一个数码管实现选通，在 250hz 时钟下就实现了扫描。

(二) 任务二

1. 从矩阵键盘中读数：用 boardkey 模块从矩阵键盘中读数，并是实现防抖，将从键盘中读出的数以总线 out_led[6..0]发出给下一模块，同时输出确认按键按下的信号 flag；

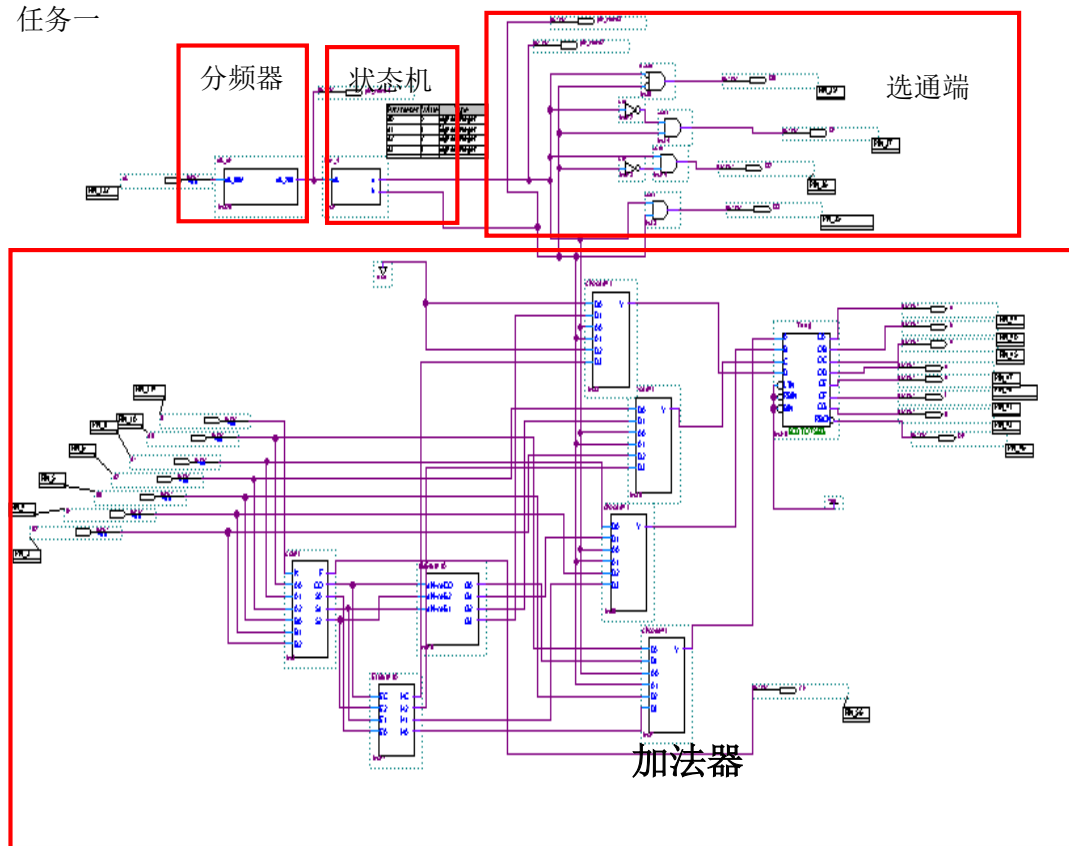
2. 实现数码管移位和闪烁：这一部分都由 forward 模块完成；

- 1) 用 clk_250 作为模块的系统时钟，在此之外用 counter 再做 10 分频用于读数和闪烁
- 2) 移位：设有四个七位数字寄存器 out_led0, out_led1, out_led2, out_led3, 分别对应一位数码管的实际输出信号，在输出该数码管时将对应的数值赋给 out_led 输出即可；从 boardkey 中接受到进位信号 flag 并确认从矩阵键盘中的新读数与上一次读数不同时（防止长按跑马）开始移位，即为：out_led3<= out_led2; out_led2<= out_led1; out_led1<= out_led0; 并将新的读数赋值给 out_led0；这样就实现了移位，移位动作对应于 flag 的上升沿。
- 3) 闪烁：当却仍按键输入为 nume 后将 clkflag 置为高电平，而平时这一信号处于低电平，在数码管扫描时写有判断语句，若 clkflag 为低电平正常扫描；若 clkflag 为高电平在扫描时需确认 counter 的计数大于 5，否则数码管不亮，在这样再一次分频下就是实现了闪烁。
- 4) 清零：当却仍按键输入为 nume 后将 out_led0, out_led1, out_led2, out_led3 全部置为 num0。

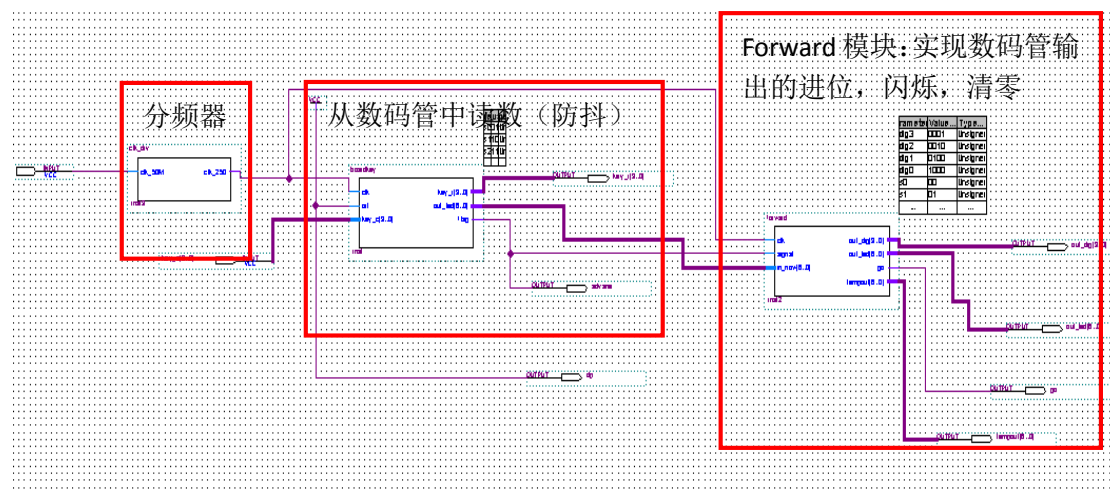
三. 顶层原理图

（见下页）

1. 任务一

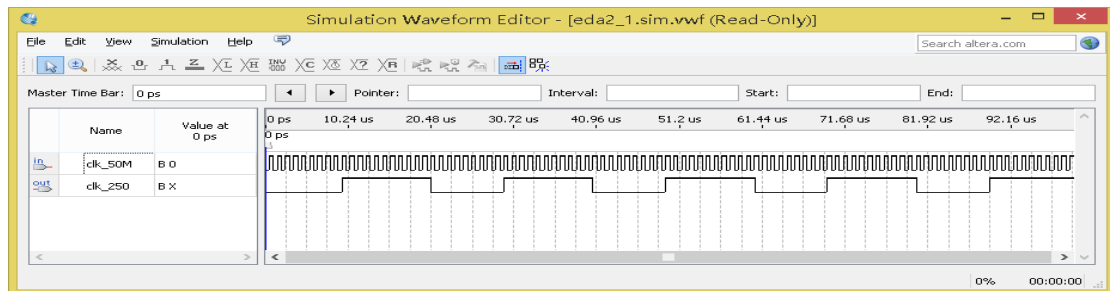


1. 任务二



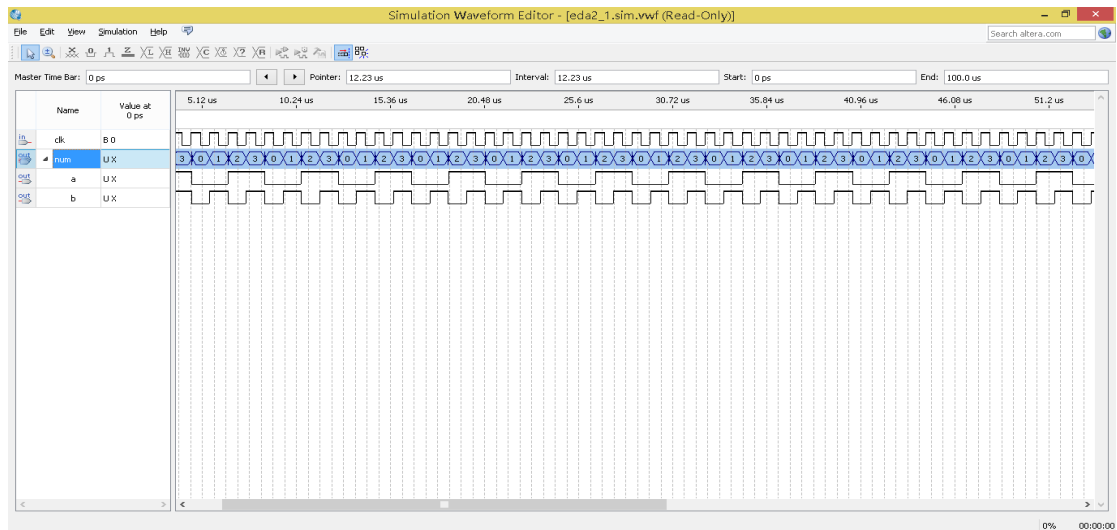
四. 仿真波形

1. 分频器

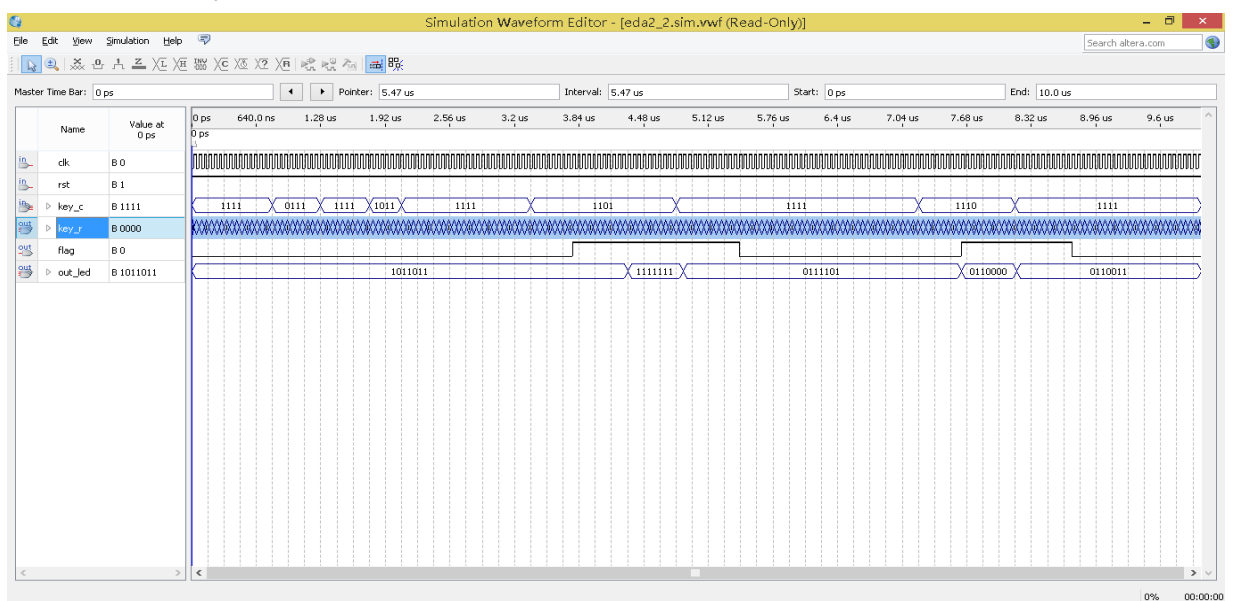


因为仿真时无法显示出 200000 分频的仿真，所以采用了 20 分频作为仿真测试

2. 数码管扫描（状态机）

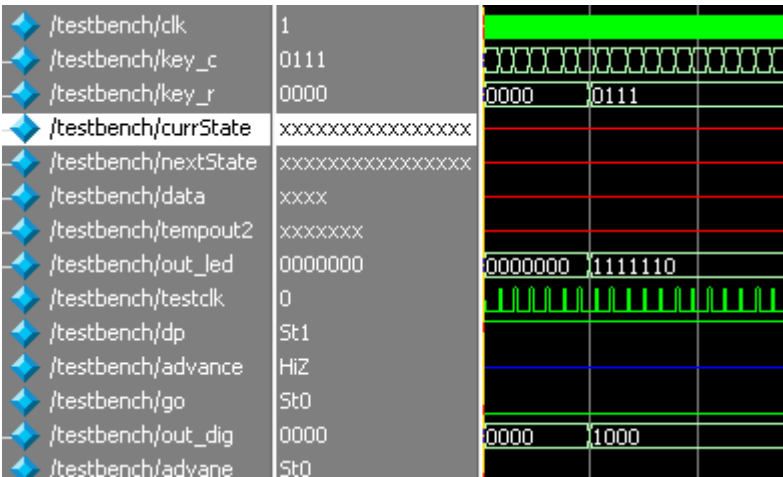


3. 矩阵键盘（quartersII）



如图可以看出，在按键时间过短（key_c 的输入时间过短）该模块不进行相应，作为防抖用途；在长按时也只响应一次，图中分别读出了 8 和 1.

4. 矩阵键盘（testtENCH）



如图可以看出 key_c 和 key_r 两个信号分别的输入和输出实现对矩阵键盘的扫描过程以及在相应位置读出输入数字 0。

五. 问题及解决方法

1. 矩阵键盘的防抖问题

在确认需要再次分频作为延时来进行防抖后我也进行了计算防抖延时时间，但在实际电路中还是有偶尔的抖动现象。我将这个时钟时间扩大了 10 倍可以确认无抖动，但按键需要延时才能被响应。

我为了解决这个问题在两个时钟周期间进行了反复测试后得到了现在的将系统时钟 10 分频可以完成较好的防抖。

2. 仿真的问题

在对分频器进行仿真时我直接进行了仿真，将仿真时间设为 100us，但依然没有看到波形。一开始认为自己完成的有问题。后来发现是在 2000000 分频后时钟周期却远大于仿真周期所致，于是再仿真是暂时修改了分频得到仿真。

3. Key_c 和 key_r 的输入输出问题

这两个变量一个是想矩阵键盘输入一个是读取输出，我在一开始错误的接反时电路也正常工作了。只是防抖的问题没有解决，按一次按键会多次输入，且无论怎样扩大防抖的时间都不能解决。后来发现了这个问题交换了输入输出之后的到了较好的防抖效果。