**实验3报告**

学号：2016K8009929029

姓名：张丽玮

一、实验任务（10%）

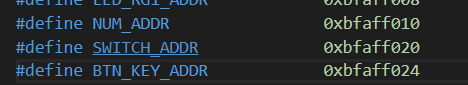
用MIPS汇编写一个软件编程电子表，运行在soc\_lite上，需要点用一些外设，从而在板子上实现一个12/24的时钟功能，最好能够包括闹钟功能等。可以考虑一下精确计时，采用中断、轮询等方法。

1. 实验设计（30%）

总的来说实现几个方面：

1. 与外设的连接
2. 时分秒的累加进位操作
3. 复位操作
4. 时分秒设置功能
5. 利用中断尽量提高计时精确性
6. 与外设的连接

阅读ucas\_cde.h，里面包含了外设的对应地址，这一次主要用到的是这两个宏定义：



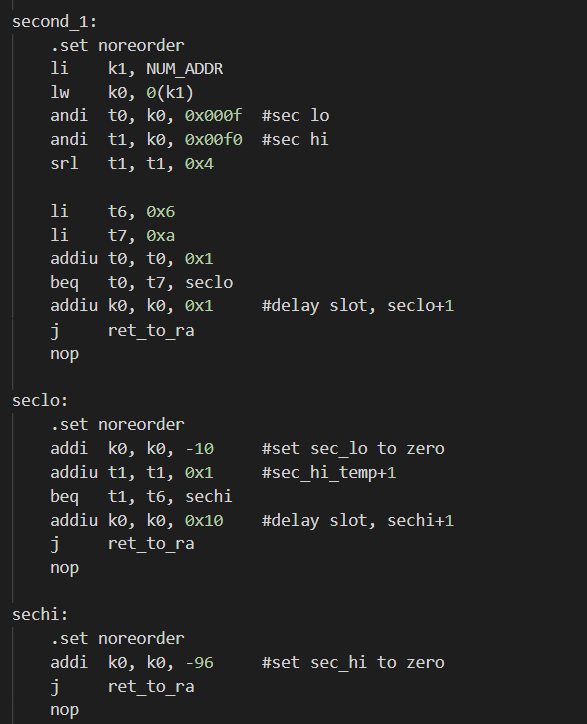
NUM\_ADDR是用来显示数值的，而BIN\_KEY\_ADDR是按键按下时的传入信号地址。

将数值传入NUM\_ADDR地址，也可以从NUM\_ADDR读出现在的时间，而从BIN\_KEY\_ADDR读取是否有按键按下。

1. 时分秒的累加进位操作

从seconds开始每次加一，一旦到了要进位处就置零，改写高位（min和hour）。主要要考虑一个16进制时分秒每个各占2位，分为低位和高位处理，以及在12小时制和24小时制下，hour的进位标准不同。

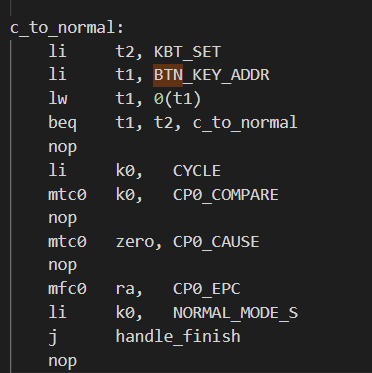
基本步骤大致如下（Min和hour相似）



1. 复位操作

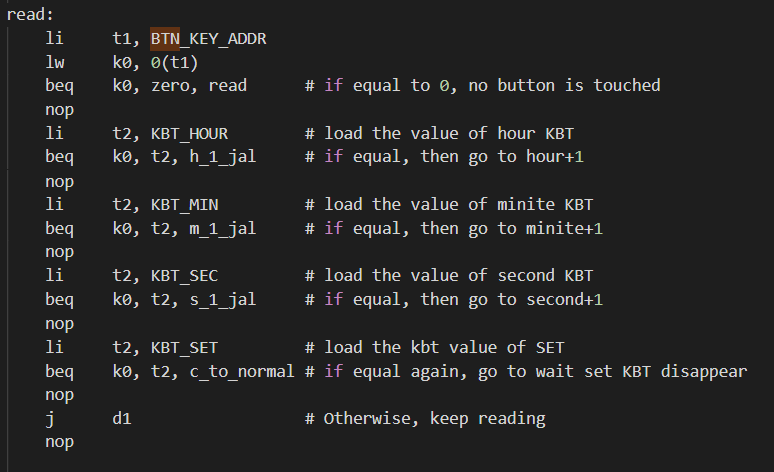
实际上就是获取复位按键是否按下，如果按下就后从 23 (12 进制下为 11)时 59 分 55 秒开始计时。

否则正常进行计时。这里也采用了轮询的方式，来判断按键是否按下。这里注意用cp0的寄存器实现，count是自带时钟，status和cause用来判断中断，compare是人为设置，表示多少时间片count==compare的时候进入时钟中断。



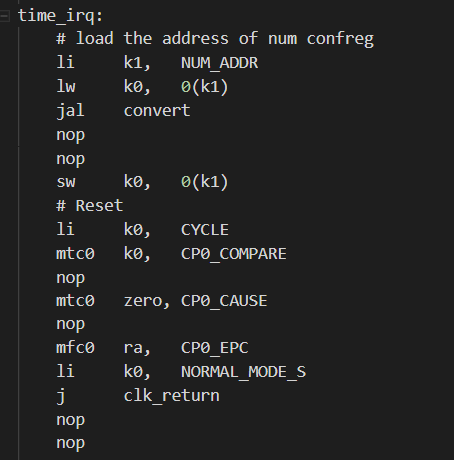
1. 时分秒设置功能

实际上还是一个读取按键的操作，根据按键情况，写入NUM\_ADDR里显示出来。这里也是轮询来判断是否设置了时分秒，从而可以达到及时输出的效果。



1. 时钟中断

时钟中断实际上就是compare==count的时候进入一个Interrupt操作，



实际上就是从NUM\_ADDR里取数，进行校准的一个过程。校准之后再将寄存器重置，准备下一个时钟中断。

三、实验过程（60%）

（一）实验流水账

10月29日 读任务书。

10月30日 阅读相关资料。

10月31日 做操作系统时钟中断相关，顺便思考体系结构实验的时钟中断。

11月1日 还在写操作系统和人工智能，草草看了几眼具体的仿真实现过程。

11月2日 开始动手写代码，写了外设连接部分。

11月3-4日 基本完成代码，但是无法上板测试。

11月5日 测试通过。

（二）错误记录

具体描述实验过程中的错误，环境问题、仿真阶段、上板阶段的都可以记录。

1、错误1

（1）错误现象

显示结果为func2-3的测试结果

（2）分析定位过程

定位具体生成操作有什么错误。

（3）错误原因

并没有实际读懂任务书，obj中的文件并非自己start.s生成的。

（4）修正效果

正常编译，不再显示func2-3结果

（5）归纳总结（可选）

仔细搞清楚修改start文件如何影响bit文件生成，多读几遍并理解任务书。

1、错误2

（1）错误现象

显示结果一直为0.

（2）分析定位过程

先分析逻辑是否有错，并未将结果传入NUM\_ADDR中还是传入不及时还是模式不对。

（3）错误原因

按键连接地址出错。

（4）修正效果

正常输出结果。

（5）归纳总结（可选）

自习阅读.h文件并且搞清楚那些宏定义的意义。

（这次没有图，也没有详细的论据。原因是我自己连不上板子，然后对于仿真调试没有理解清楚，只能勉强请同学帮我看一下bit跑的情况的样子……基本只能一句句筛查debug）

四、实验总结（可选）

为什么会这样呢，明明原本是好的，这是为什么呢？

上一次vivado炸了之后，这一次连不上板子，之前连不上倒是没事，基本仿真PASS了上板问题不大，但是这一次不行啊啊啊啊啊啊啊。

我最大的敌人可能是我自己的电脑。

（顺带一提我操作系统早就连不上了然后因为TLB例外验收出事了orz）