

实验 4 报告

第 27 小组
姚子薇、陈乐滢

一、实验任务

编译一段汇编程序，运行在 SoC_Lite 上，调用 Confreg 模块的数码管和按钮开关等外设，实现一个 24 小时进制的电子表，并在实验板上予以演示。

该电子表的显示包含时、分、秒，采用实验箱开发板上的 4 组数码管显示，并通过板上的矩阵键盘完成电子表的设置功能。具体要求是：

- (1) 电子表具有一个暂停/启动键（左上键），具有时、分、秒设置键。
- (2) 电子表复位结束后从 0 时 0 分 0 秒开始计时，按下暂停/启动键一次则计时暂停进入设置模式，此时可以通过时、分、秒的设置键修改时、分、秒的值，再次按下暂停/启动键则推出设置模式并从设置好的时间开始继续计时。
- (3) 时、分、秒设置键的设置方式是每按下一次，对应的时、分、秒值循环加 1，按住不放则按照一定频率不停地循环加 1 直至按键松开。
- (4) 时、分、秒设置键仅在设置模式下操作才有效果。
- (5) 矩阵键盘上非设置键被按下，应当不影响电子表的精确计时。
- (6) 采用硬件中断+时钟中断完成本次实验。

验证方法即上板演示符合上述要求成功。

二、实验设计

（一）为实现硬件中断所做的 verilog 修改

在 soc_lite_top.v 中将 confreg 模块中的 btn_key_r 引出，并取相应位连接输入 ls132r_top cpu 模块的 int_n_i，从而实现硬件中断。其中 btn_key_r[0] 位（即左上按键）为暂停/启动键，btn_key_r[1]-btn_key_r[3] 分别为秒，分，时的设置按键。

（二）精确计时及设置的实现

精确计时通过时钟中断实现，进入设置模式、时分秒设置及退出设置模式均使用硬件中断实现，流程图如下：

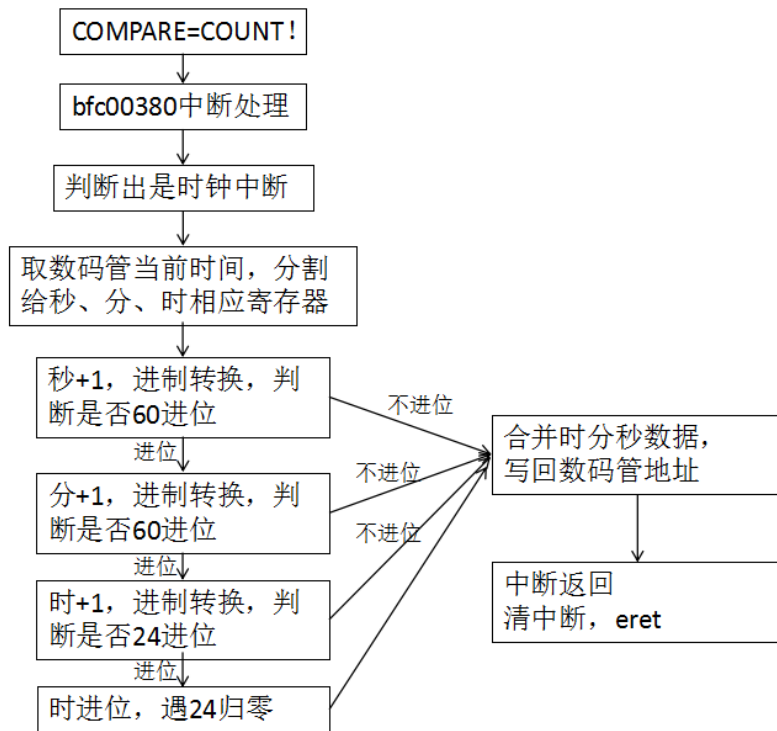


图 1. 时钟中断流程图

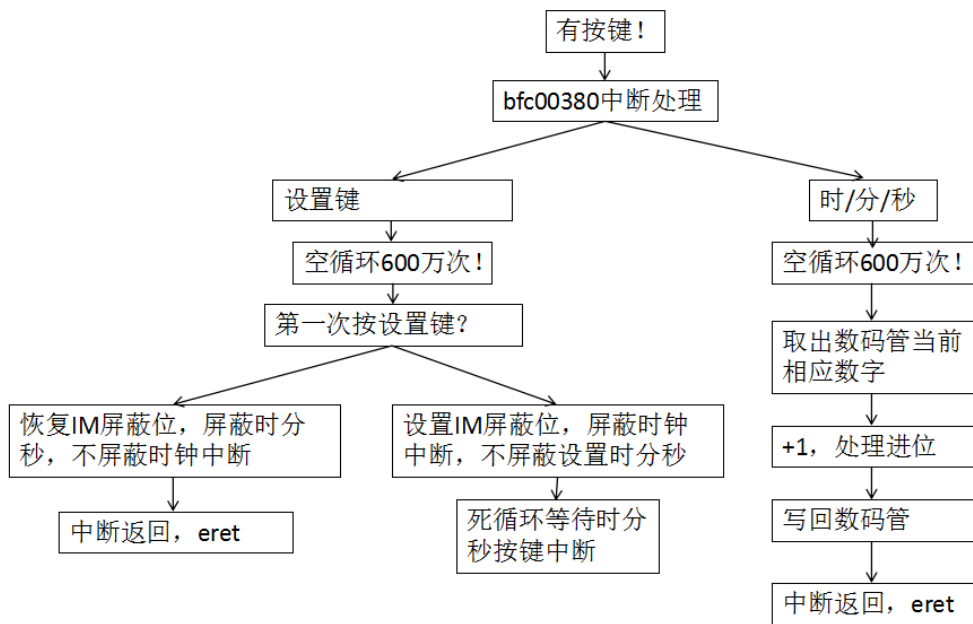


图 2. 硬件中断流程图

三、实验过程

（一）实验流水账

1. 姚子薇&陈乐滢

2017.11.6, 18:00 - 22:00, 写出时钟中断的雏形, 仿真用的 mif 文件不对所以没有进行。

2017.11.7, 9:00 - 16:00, 调整时钟中断、添加设置时间的硬件中断, 仿真现象合理, 上板大体正确。

2017.11.7, 17:00 - 17:30, 撰写实验报告。

(二) 错误记录

1、错误 1

(1) 错误现象

编译错误, error 信息 express out of range。

(2) 分析定位过程

观察出错的指令, 发现都是立即数操作, 然后想起立即数操作只有后 16 位才是立即数。

(3) 错误原因

立即数操作是 16 位的, 而我们在 ori 等指令中用了 32 位的立即数。

(4) 修正效果

将立即数 li 到寄存器中, 再利用寄存器之间的计算实现功能。

2、错误 2

(1) 错误现象

替换后仍然跑的是原来的仿真文件, 即“PC 顺序加 4 将各个寄存器初始化”的 PC 序列, 没有执行 j locate 这一条。

(2) 分析定位过程

将 mif 文件交给同学, 发现在他们的程序中正确的跳转到了 locate, 我们考虑应该不是硬件代码和软件代码的问题。

(3) 错误原因

没有开始仿真以后替换 mif 文件。

(4) 修正效果

在开始仿真之后替换 mif 文件, PC 正常跳转到了 locate。

3、错误 3

(1) 错误现象

num_dat 写回错误, 显示 0x00xx0000, 没有加 1。

(2) 分析定位过程

通过看波形找到 XX 的来源, 第五六位是 X 说明问题来源于 hour 时间的计算, 的确相应寄存器有问题, 检查代码。

(3) 错误原因

hour 计算中有一步 and t3,t0,t5,是为了取出小时对应位的值, t0 保存了当前数码管显示数字。手误将 t0 写成了 t3, t3 没有初始化过, 所以导致了 X。

(4) 修正效果

将第二个 t3 改为 t0 后正确。

4、错误 4

(1) 错误现象

十进制十六进制转化在应该正常+1 的位置+6。

(2) 分析定位过程

判断是否需要+6 的分支跳转选择无误，明明应该跳走，却还是进行了下一条指令+6。因为+6 指令使延迟槽指令，一定会被执行。

(3) 错误原因

bne 后没加 nop 延迟槽。

(4) 修正效果

在所有的跳转指令后面都加了 nop，修改后正确。

5、错误 5

(1) 错误现象

加入硬件中断处理之后中断类型判断错误。

(2) 分析定位过程

的确都是按照条件判断的，但是从 15 号寄存器里取的 IP 位匹配不上。

(3) 错误原因

发现 13 号寄存器才是 Cause，之前可能手滑改成了 15。

改回 15 后发现 IP 位还是对应不上。IP7 是 1 时，CauseIP7 是 0，IP9 是 1，恰好错开。

(4) 修正效果

将 13 号寄存器改为 15 号，将 IP7-IP3 对应到 CauseIP9-IP5 的位上，修改判断中断类型条件后正常。

6、错误 6

(1) 错误现象

上板时钟+1 特别快，仅能隐约看到 hour 的改变，分秒完全看不清。

(2) 分析定位过程

在最开始的初始化写成了 25000000，一时没有找到原因，后来停止上板修改别的代码时发现可能是时钟中断处理返回时重写 COMPARE 没修改造成的。

(3) 错误原因

时钟中断处理返回时重写 COMPARE 没修改为 25000000 导致第一次+1 之后后面都是以 250 为界中断处理，时间非常短。

(4) 修正效果

修改后再次上板正确。

7、错误 7

(1) 错误现象

按 set 的时候有一定几率可以进入 set 模式，一定几率不能进入，且按一下调整时间的键（如秒），会连续调整很多秒。

(2) 分析定位过程

和同学交流后我们发现应该在硬件中断中分别加一个死循环以适应“人按一下该按键的 1 会持续很久”这一事实。

(3) 错误原因

在板子上按一下 set 键实际上相当于机器内部过很多个周期，所以在按下过程中，会不断出现 set_int 和 set_ret 这两个中断，导致这一操作不稳定，按一下会增加很多秒也是类似的问题。

(4) 修正效果

在硬件中加一个循环 6000000 次的死循环后，虽然感觉有点卡，但是可以实现功能。

四、实验总结

(一) 组员：姚子薇

延迟槽！！遇到的百分之七十的奇奇怪怪的问题最后发现原因都是跳转后没加 nop。。。以后估计不会忘了。大体思路很明确，但是小问题细节上的层出不穷，第一次上板的现象可以说是很诡异了，很多上板操作出现的比较特殊的情况也很难仿真出来。一个困难点在于按键时长对中断处理的影响，通过加了六百万次空循环解决了。emmm 总体看这个实验跟前面比起来应该是简单一点，但是这两周...大作业+期中考，时间不是一般的紧张。这周六还有考试，周三还有 DDL，希望能活下去……

(二) 组员：陈乐滢

计算机系的死亡之周。在看过任务书之后认为实验并不难，所以打算在人工智能考试之后再着手准备。

没有想到又做到了实验课前 qwq。写代码的时候无限分不清自己在写硬件还是在写软件。

好了好了要去写操作系统了。