自动调节亮度的实时时钟

计科1802 孙浩然 201801130227

## 基本原理

1. 使用实时时钟模块获取时间：通过获取当前的时间戳，在复位时将时间写入实时时钟，之后每过100ms将读取实时时钟的数据保存在定义的数据结构中，通过数码管部分进行扫描显示。
2. 使用光敏电阻通过AD转换获取当前的光照值：光敏电阻测得的模拟信号高8位放在AD\_RES，低2位放在AD\_RESL，通过公式AD\_RES \* 256 + AD\_RESL可转换为10进制数，通过AD转换，将光敏电阻测得的模拟信号转换为数字信号，多次测量取平均值。
3. 使用按键检测切换显示：通过检测按键是否被按下并进行消抖处理，切换需要显示的信息。
4. 使用振动传感器显示数码管：通过振动传感器检测是否发生振动，在特定的显示模式下显示当前的时间。
5. 使用数码管动态扫描技术改变数码管显示的亮度：通过根据光照改变数码管扫描的位数，从而达到通过不同亮度显示数码管的功能。

## 设计思路

1. 完成数码管显示模块的功能，建立整体框架
2. 加入实时时钟显示部分，使用数码管正确显示当前的时间。
3. 加入按键检测功能，用于切换数码管的显示。首先加入按键K1切换是否显示实时时钟的时间，再加入按键K2切换显示时间或显示光照值。
4. 加入振动传感器部分，在不显示时间的情况下，检测振动，发生振动则短暂显示时间。
5. 加入光照测量模块，通过光敏电阻测得当前的光照值的模拟信号，使用AD转换将其转换为数字信号，并通过数码管在光照值显示时进行显示。
6. 添加根据光照修改数码管的显示亮度，设定不同的范围值，在某个范围内以一种特定的亮度显示数码管，即修改其扫描的位数。
7. 整合测试，修改已经检测到或可能存在的bug。

## 实现功能

1. 可以通过按键选择三种显示模式：显示当前时间，显示当前光照值，熄屏时振动则显示时间。
2. 熄屏时可以检测振动以短暂的亮屏，实现“敲击亮屏”功能。
3. 可以根据当前的光照值修改数码管的亮度，达到在不同光照环境下有不同的亮度的效果。

## 主要工作

1. 理解实时时钟以及AD转换部分的工作原理，学会自己去使用这一部分，了解其使用的引脚以及各个寄存器的功能，阅读技术手册。
2. 结合之前按键检测和数码管显示案例的学习，写出自己所想要的功能，并进行分模块的检测，构建整体框架。
3. 加入其他扩展功能，如振动亮屏，切换光照显示功能，分模块完成设计，验证模块没有问题之后再进行整合。
4. 完成最终的光照控制数码管显示亮度的功能。
5. 整体测试，查找bug并修复。完成函数重构部分，使得代码可读性增强。

## 实现中的难点

1. AD转换使用的引脚与实时时钟相冲突，二者只能启用一个。
2. 数码管扫描频率问题，使得有些情况下出现闪烁的现象，展示效果不好。

## 解决方法

1. 通过查看相关数据手册，修改AD转换时使用的P1ASF对应的引脚，仅使用相关的部分，将与实时时钟相关的P1.5和P1.6留出。
2. 发现数码管闪烁是由于在变换亮度时发生，所以将其放入定时器模块中，每隔一段时间进行检测来切换亮度，从而减少了闪烁频率的发生。

## 实训总结

“纸上得来终觉浅，绝知此事要躬行。”

从拿到学习板第一天的感觉到的新鲜感，感觉的他的神奇，再到现在自己完成一个大设计，从完全不懂是如何实现、只知道怎么去验证，到现在可以动手来实现一个较为简单的功能，理解一些单片机的原理，是一个很神奇的过程。

这个过程中必不可少的自然就是自己动手去做。记得在第一天看案例的代码时，完全不懂是如何工作的，对于定时器，AD中断等功能的使用也是完全无法理解，直到准备开始写大设计时，去自己尝试修改给出的标程，再到根据标程修改，来实现自己想要的功能，在这个过程之中，我自认为学到的东西远比只去看代码学习要多。当然，某些地方的疑惑的解决，如AD转换部分，也少不了老师的答疑。