

DSP 最终设计文档

13331148 林浩文

13331181 龙嘉伟

设计概述

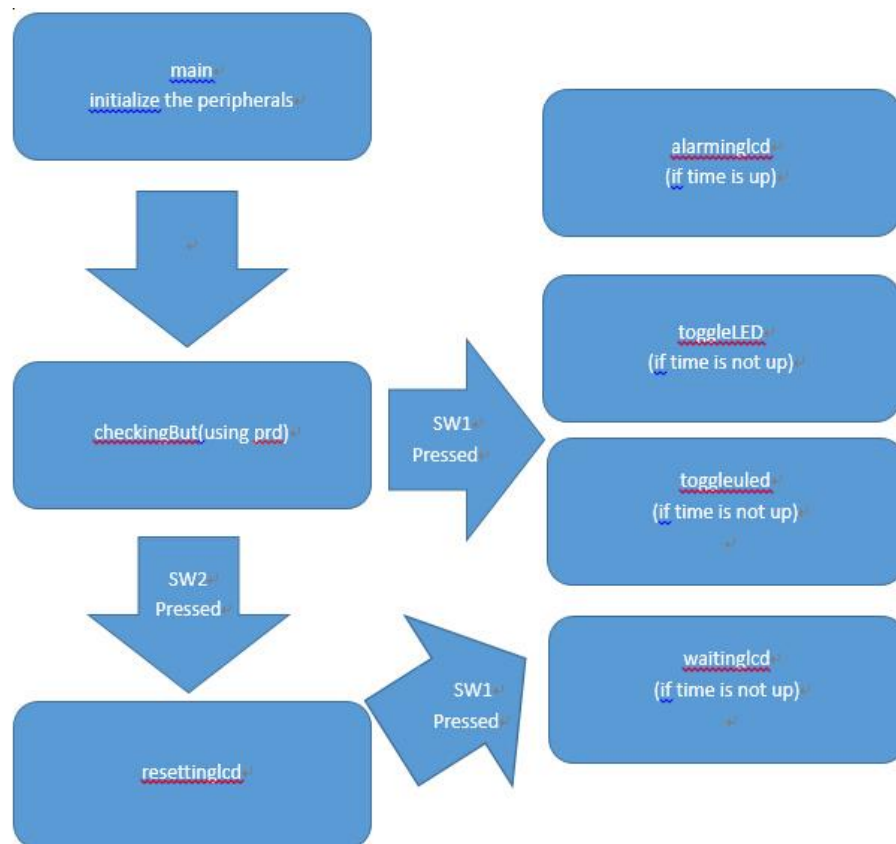
功能介绍:

本设计是一个闹钟程序，能够预先设定闹钟触发时间，通过一号按键开始闹钟程序，进入等待模式，通过二号按键可以重置闹钟。在闹钟等待过程中，led 灯会以一秒一次的速度闪烁，达到读秒的作用。而四盏彩灯会根据等待时间显示当前进度，每一盏灯代表总时间的 25%，每经过 25%就会多亮一盏灯。在等待过程中，LCD 显示屏会显示 ““ZZZZZZZZZZ”” 的字符串，以表示闹钟正在运行。当到达了闹钟触发时间，led 停止闪烁，四盏彩灯全亮，而 LCD 会显示 “TIME IS UP” 的字样，表示闹钟已被触发。此时可以通过二号按键重置闹钟，使得四盏彩灯熄灭，并在 LCD 上显示 “RESET INSTRUMENT” 字样，提示设备已被重置。之后可以通过一号按键重新进行计时。

设计思路:

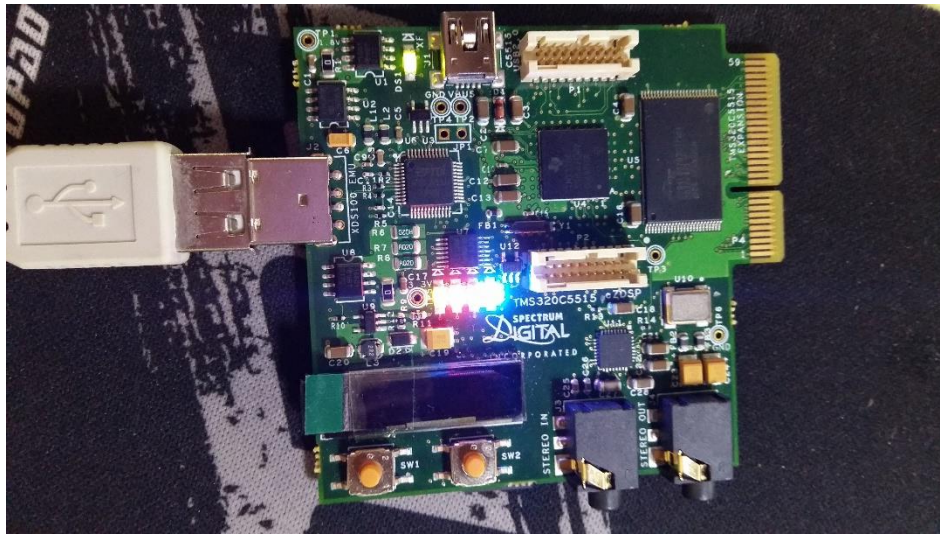
本设计使用到了 C5515 芯片上的 led 灯、四盏彩灯、两个按键以及 LCD 显示屏等外部设备。通过使用 CSL 中的 SAR-ADC 检测按键是否被按下，进而实现开始和重置这两个功能。设计中还使用了 DSP/BIOS 中的周期函数，通过一个以一秒为周期的函数来实现亮灯和显示功能。

程序流程:



设计结果:

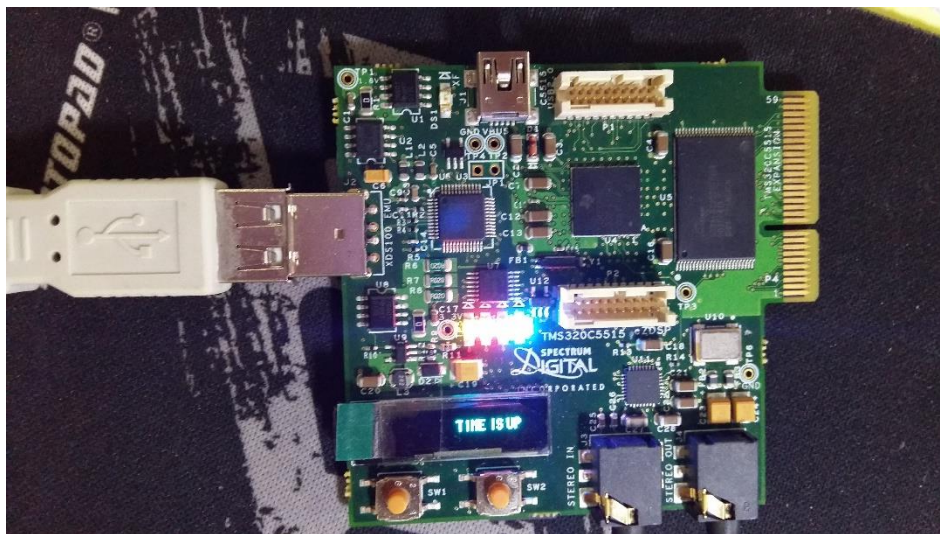
芯片启动后，等待 SW1 被按下，启动闹钟程序：



程序已启动：



闹钟被触发：



按下 SW2 重置闹钟，等待按下 SW1 重新开始：



用户指南

使用说明：

需要计量的时间要预先设置，这个数值是程序中的一个常量，可以自行修改。用户启动 C5515 后，按下一号按键（SW1），就会启动计时器。此时 led 闪烁读秒，而彩灯也会闪烁。每经过总时间的 25%，就会有一盏彩灯由闪烁变为常量。当 led 不再闪烁，四盏彩灯长亮时，闹钟即到达触发时间。此时按下二号按键（SW2）可以重置闹钟，使得彩灯熄灭，等待开始按键（SW1）按下，重新进行计时。

程序说明：

首先通过一个宏定义的常量来控制你需要计时的长度，60 即为一分钟。

```
#define Time 60 // set the alarm
```

然后是 main 函数，主要负责 LED、ULED、SAR 等设备的初始化。

CheckingBut 函数是一个周期函数，周期为 1s，负责检测有无键被按下，并根据按下的键的不同，调用不同的函数，实现后续的不同功能。

toggleLED 函数负责使 LED 产生闪烁效果。

toggleuled 函数负责让彩灯产生闪烁效果，并且在经过总时长的 25%后使一盏彩灯长亮。

OSD9616_send、OSD9616_multiSend、printLetter 函数都是 LCD 显示字符时需要用到的。

waitinglcd 函数负责显示闹铃等待过程中的信息，即“ZZZZZZZZZZ”。

alarminglcd 函数负责显示闹铃触发时的信息，即“TIME IS UP”。

resettinglcd 函数负责显示按下重置按键时的信息，即“RESET INSTRUMENT”。

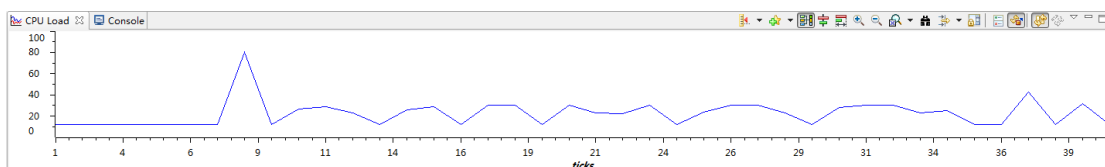
测试报告

芯片主频：100MHz

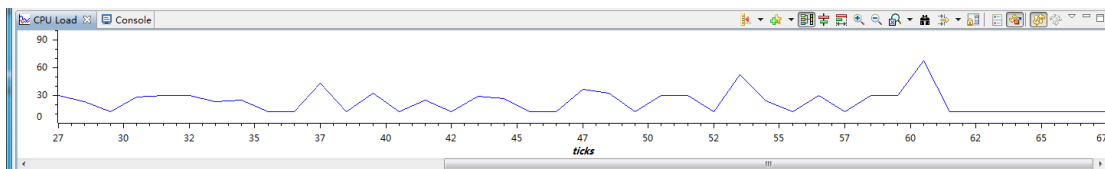
功能测试：

连接好电源并且使设备处于 debug 状态，按下 SW1 启动，显示情况理想。按下 SW2 进行重置时，需要注意按压力度和按压时间。在 LCD 显示屏进行不同的信息切换时，会有花屏现象出现。

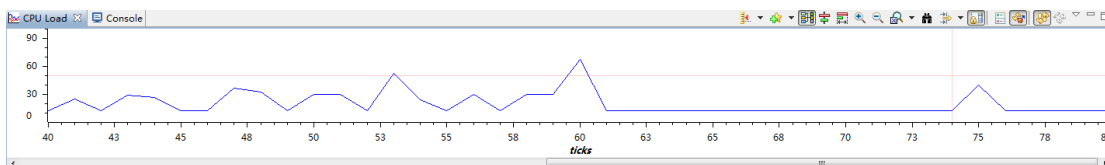
CPU 占用率测试：



第七秒左右启动计时器



第七十四秒左右进行重置



设计总结

本次的设计要同时使用 CSL 和 DSP/BIOS,使得多个外设能够高效地按照设计目的实现对应的功能。

难点在于对 CSL 各个头文件以及 C5515 芯片的不熟悉,例如在使用 LCD 显示字母时,需要了解字符是如何显示的,并且能够将特定参数传给 `printLetter` 函数来显示目标字符。这个例子里面要了解字符是如何编码的,需要我们自己去找文档、查资料。

而另外一个例子是如何判断有无键被按下以及哪个键被按下。按钮下方有一个电阻,如果将按钮按下电阻的值就会发生改变,进而影响到对应电压的值。而我们需要通过检测电压的变化来判断键是否被按下以及哪个键被按下。一开始我是单纯判断对应的电压是否与某个值相等,而经过多次尝试,发现这样的实现会使得按键不灵敏,即按下了也无法被识别。后来,通过判断电压是否落在一个区间里来确定按键是否被按下,灵敏度得到了很大的提高。由于硬件会因为各部件自身或者外界环境因素而产生一定的误差,按键按下后的电压不一定每次都相同,但一定是很接近的,因此通过范围判断更为合适。

平时的课程很少有机会能够给硬件编程,而 DSP 程序设计不仅给我们提供 C5515 这片芯片,还让我们学习到如何控制这块芯片去实现一些有趣的功能。这次的设计虽然比较简陋,但的确是组员们一手一脚地看例子,读文档,动手写代码做出来的一个项目,是对我们五天的课程学习的一个检验和肯定。