****

**DSP原理及应用**

**实验3：PIE中断控制器和**

**CPU TIMER中断**

**实验报告**

**姓 名： 林宇航**

**班 级： 自动化01**

**学 号： 201906060308**

**学 院： 信息工程学院**

**设计日期 2022.3.28**

# 实验三 PIE中断控制器和

# CPU TIMER中断

1. **实验目的**

1. 熟悉如何编写28335的中断服务程序

2. 掌握中断的精准屏蔽与开启

3. 学会PIE中断控制器的执行机制

**二、实验内容**

1. 将老师准备的cputimer工程导入CCS，并下载进DSP

2. 查看主板上灯是否以时间间隔为1s的频率闪烁

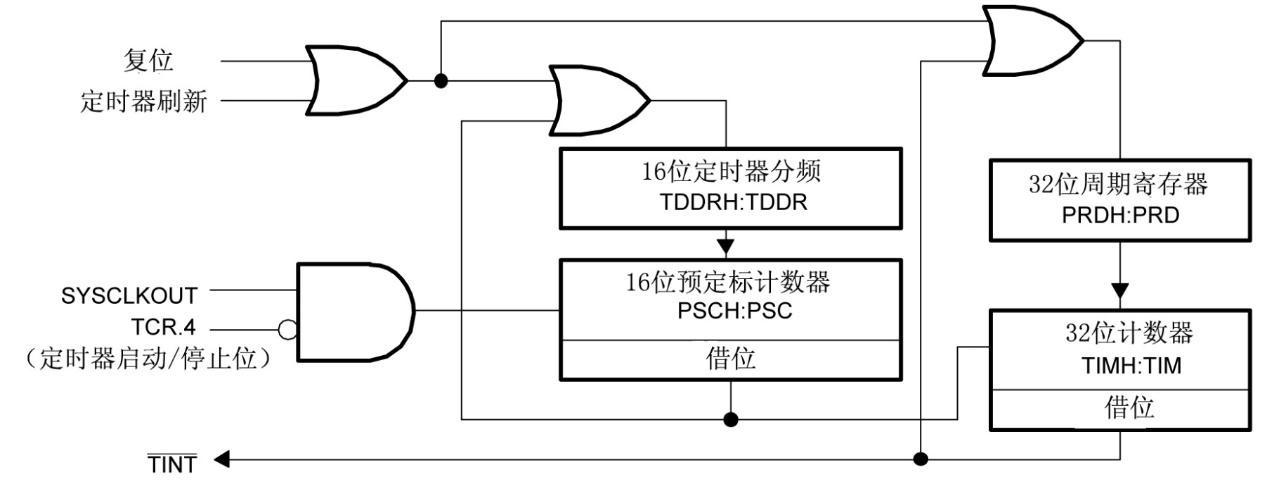
3. 将cputimer0、1、2的定时间隔改为1s、2s、3s

4. 尝试禁用cpu0的定时器中断

**三、实验基本原理**

F2833x系列DSP包含3个CPU定时器，分别是CPU 定时器0、1、2。定时器0、1可以被用户使用，定时器2保留给实时操作系统（DSP-BIOS）。若未用到实时操作系统，用户也可以使用定时器2。

定时器工作原理：32位的计数器（TIMH:TIM）从周期寄存器（PRDH:PRD）中装载数据，每经过（TDDRH:TDDR+1）个SYSCLKOUT周期，（TIMH:TIM）减1，当计数器等于0时将产生一次中断请求信号。



**四、实验步骤以及实验现象：**

1.实验准备

略。

2.导入老师下发的 cputimer 进行相关配置

略。

3. 学习此工程中 cputimer.c 文件中的代码

1)将代码原封不动烧入DSP中，点击运行，观察主板发现有个黄色led灯在闪烁，时间间隔约为1s，看代码发现其实现的基本功能是设置1s定时器中断，没进入一次中断就翻转LED对应IO口的电平，从而实现LED闪烁，开启中断如图1、中断服务函数如图2。

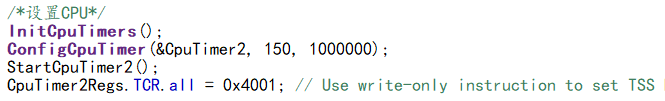


图 1 定时器初始化

CpuTimer2的频率为：，同时TCR寄存器的第15位控制使能位，故置1。

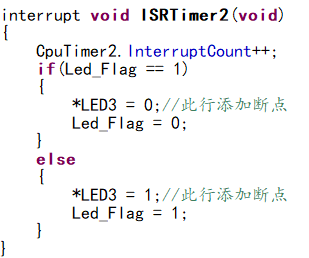


图 2 中断服务函数

根据cpu中断连接图，cputimer2直接连到INT14端口，所以设置IER对应位对其使能，如图3。



图 3 使能CPUtimer2对应的中断INT14

应老师要求，开启timer0和timer1的定时器中断，中断时长分别为2s和3s，仿cputimer2的初始化方法，如图4：

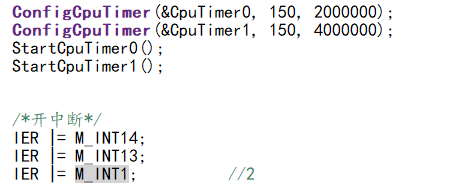


图 4 不同定时器对应的中断通道使能

同时还要初始化PIE中断向量表，外部中断包括12个组，每组有8个中断，共96个。内部中断有19个，将各定时器按照向量表的映射初始化到对应的中断通道，并给出中断服务函数的地址，如图5：

文本

描述已自动生成

图 5 PIE中断向量表初始化

为了方便调试，中断服务函数中设置一个计数器自加，由于cputimer0是挂载到PIE模块上的，根据中断向量表映射到INT1，所以在中断函数中要应答一下，对PIEACK寄存器进行赋值，如图6：

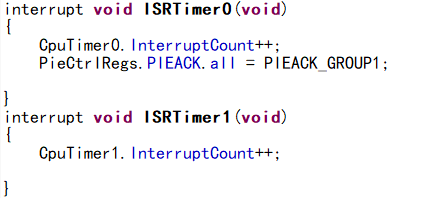


图 6 定时器1和0的中断服务函数

将代码烧入DSP，进入debug界面后在Expression中加入需要观察的变量：CpuTimer0.InterruptCount、CpuTimer1.InterruptCount和CpuTimer2.InterruptCount，发现这三个变量按照1，2，3的倍数递增，如图7：

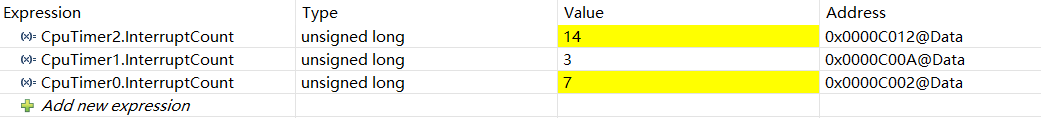


图 7 定时器中的计数器

说明定时器中断成功。

4. 关闭cputimer0的多种方法

1)关闭对应的PIE中断使能标志位 PIEIRx

2)关闭cputimer0对应的INT通道的中断使能标志位IER

3)不赋值应答标志位PIEACK

4)关闭全局中断使能INTM

**五、实验总结：**

通过本次实验我更深入的了解了一个平时经常用的单片机资源——定时器，与平时使用方法不同的是，DSP的定时器有多种类型，必须根据DSP中断连接来进行配置，十分繁琐，但同时也带来很大的灵活性，平时的定时器就是一句函数初始化，根据对应的接口类型写入要开启的定时器几和中断周期即可，跳转到中断服务函数编写需要的程序，用起来十分方便，对于现阶段的知识水平，这完全够用，减少对底层的研究，更注重上层代码的编写。