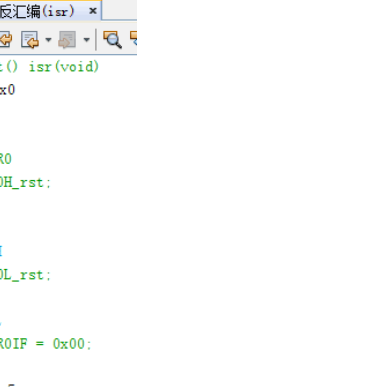


## 工作日志

标题	0913 工作日志		
姓名	汪能志	学号	U201713082
日期	2020年9月 13 日	天气	晴
日记撰写时间			
内容	<p>一、基于定时器中断的闪灯实验</p> <p>通过 c 语言完成了基于 16bit 定时器的定时器中断闪灯。</p> <p>根据定时器的计算公式</p> $pre \times (65536 \times (post - 1) + 65535 - tmr)$ <p>计数器溢出周期为 499990。</p> <p>通过 mplab 进行调试，结果为正好 500ms。</p> <p>设置如下：</p> <pre> T0CON0 = 0xD3; T0CON1 = 0x41; TMR0H = 0x2F; TMR0L = 0x75; </pre>  <pre> 1  !void __interrupt() isr(void) 2  0x4: BSF 0x7E, 0x0 3  0x5: MOVLW 0x0 4  !{ 5  !    // reset TMR0 6  !    TMR0H = TMR0H_rst; 7  0x6: MOVLW 0x2F 8  0x7: MOVLB 0x0 9  0x8: MOVWF TMR0H 10 !    TMR0L = TMR0L_rst; 11 0x9: MOVLW 0x75 12 0xA: MOVWF TMR0L 13 !    PIR0bits.TMR0IF = 0x00; 14 0xB: MOVLB 0xE 15 0xC: BCF PIR0, 0x5 16 !    // interrupt task 17 !    PORTC = ~PORTC; 18 0xD: MOVLB 0x0 19 0xE: COMP PORTC, F 20 !    // flip PORTC 21 !} 22 0xF: BCF 0x7E, 0x0 23 0x10: RETFIE </pre> <p>Stopwatch cycle count = 500000  Stopwatch cycle count = 500000  Stopwatch cycle count = 500000  Stopwatch cycle count = 500000  Stopwatch cycle count = 500000  Stopwatch cycle count = 500000</p> <p>二、中断时间波动</p> <p>使用 8bit 定时器，初值如下：</p> <pre> T0CON0 = 0xCF; T0CON1 = 0x47; TMR0H = 0xF3; </pre> <p>溢出周期为 499712</p> <p>调试结果会在 499711，499712，499713 之间循环。</p> <p>三、通过 MPLAB 进行反汇编操作</p> <p>由于进行中断程序调试时，发现中断触发的间隔超过了定时器设置的值，而且和 datasheet 中所说的中断延时不符，因此我们对程序进行了反汇编，查看可能的原因。</p> <p>在 MPLAB 中选择 windows/PIC 存储器视图/程序存储器，然后选择调试。就可以查</p>		

看编译后的汇编代码。

Line	Address	Opcode	Label	DisAssy
38	025	002E		MOVLB 0xE
39	026	128C		BCF PORTA, 0x5
40	027	0008		RETURN
41	028	3180		MOVLW 0x0
42	029	2012		CALL 0x12
43	02A	3180		MOVLW 0x0
44	02B	3180		MOVLW 0x0

由于单片机的内存架构问题，部分代码需要结合 datasheet 中关于内存结构的部分才能理解。如截图的第一行和第二行，这两行执行的操作是：将 PIR0 的第 5 位（即 TMR0 溢出标志位，TMR0IF）置零。但是由于在不同的 bank 中，PORTA 和 PIR0 的地址均为 0x0C，因此 BCF PIR0, 0x5 和 BCF, PORTA, 0x5 的机器码相同。同时，这也说明了在进行操作前进行正确的 banksel 是非常重要的。

在进行反汇编后，我们仍然不能确定 定时器和触发中断之间的时间关系。推测原因有：中断延时，处理器不能在执行一条指令的过程中切换进入中断

#### 四、实验汇报

由于原计划有 ppt 汇报，为了较为直观的说明中断的操作和意义，我们在 ppt 中准备了如下内容：

## 实验原理

中断的好处：不用刻意地反复检测信号，能第一时间收到信号

### • 不使用中断

```
while(作业没写完)
{
    写一道题目();
    if(去门口看一眼()==有人)
        开门();
}
//既要花不少时间去看有没有人，又有可能让别人等着
```

### • 使用中断

```
写作业();
有人敲门，中断发生！
执行中断函数，开门
{
    走到门口();
    转动门把手();
    拉开门();
}
中断结束，继续写作业
/*接着之前的位置*/
写作业();
```