05 基本定时器

实现功能：通过定时器TIM3 50毫秒中断一次,LED每隔1S移动一位。

（1）屏幕初始化：

第1行：" TIMER DEMO "

第3行：" See The LEDs! "

（2）在LCD上继续显示点亮的灯的序号。

第6行： " -- LED ON:LD\* " （\*表示灯的序号）

说明：用proteus8.6仿真时，tim3定时器不能完成仿真，tim1可以做，但会出现时间问题。

建议用proteus更高级版本，安装时不能安装在C盘。

仿真结果时间会有很大的误差，需要耐心等待。

1. **仿真原理图**

**二、移植说明**

1、硬件引脚说明

2、软件配置说明

**三、实验程序**

.c .h代码

**核心要点分析**

while(1)

{ gpc=~(1<<(leds+7));

GPIO\_Write(GPIOC,gpc);

GPIO\_Write(GPIOA,~(1<<(leds+7)));

sprintf(string,"%s%d","--LED ON:LD",leds);

LCD1602\_Show\_Str(0,4,string);

}

GPIO\_Write(GPIOC,gpc);会改变PORTC.10、PORTC.11、PORTC.12引脚状态

但LCD显示时，对这三个引脚所做的一系列操作，中间不会被打断，因此不会受到该语句的影响。若在中断中使用该语句，则LCD会受到影响，可以只对其中的某些位进行置位或清零，或者增加锁存器以解决，或者改端口。

用下面方法不会对其它位产生影响。

GPIOx->ODR |=(~(1<<(leds-1) ) ) <<8 //eg：1111 1101 0000 0000 ，led=2，高8位(除D9)置1

GPIOx->ODR &=~(1<<(leds+7) // eg：1111 1101 1111 1111，led=2，D9位清0，其余保持

**四、实验现象说明**

**五、练习**

1、开启定时器1，关闭其他定时器，实现例程相同的功能。

2、采用systick的delay延时功能依次实现花样流水灯：

（1）————> （依次点亮，其他灯灭）

（2）<———— （依次点亮，其他灯灭）

（3）————> （依次点亮，其他灯灭）

（4）<———— （依次点亮，其他灯灭）

（5）————> （依次点亮，其他灯灭）

（6）<———— （依次点亮，其他灯灭）

（7）————> （依次点亮，其他灯不灭）

（8）<———— （依次灭掉，直到所有灯灭）

（9）————> （依次点亮，其他灯不灭）

（10）<———— （依次灭掉，直到所有灯灭）

（11）————> （依次点亮，其他灯不灭）

（12）<———— （依次灭掉，直到所有灯灭）

（13）————> （依次点亮，其他灯不灭）

（14）————> （依次点亮，其他灯不灭）

（15）————> （依次点亮，其他灯不灭）

（16）所有灯灭

（17）所有灯亮

（18）所有灯灭

（19）所有灯亮

（20）所有灯灭

（21）所有灯亮

（22）转到（1）、（2）......循环执行

调整delay延时时间，再观察现象。

提示：将花式流水灯的所有状态存放到数组，依次从数组取出数据传给端口，每传一次后做一次延时。

**六、总结**