# 一、Linux内核定时器原理

## 1.1、内核时间管理

1、Cortex-M内核使用systick作为系统定时器。

2、硬件定时器、软件定时器，原理是依靠系统定时器来驱动。

3、linux内核频率可以配置，图形化界面配置。

4、重点，HZ表示系统节拍率，

jiffies，

## 1.2、节拍率高低的缺陷

## 1.3、jiffies

## 1.4、内核定时器

1、软件定时器不像硬件定时器一样，直接给周期值。设置期满以后的时间点。

2、定时处理函数。

3、内核定时器不是周期性的，一次定时时间到了以后就会关闭，除非重新打开。

# 二、编写试验驱动

1、定义一个定时器，结构体timer\_list

2、应用ioctl函数-》unlocked\_ioctl和compat\_ioctl。

long (\*unlocked\_ioctl) (struct file \*, unsigned int, unsigned long)

long (\*compat\_ioctl) (struct file \*, unsigned int, unsigned long);

ioctl的命令是自己定义的，但是要符合linux规则。构建命令：

#define \_IO(type,nr) /没有参数的命令

#define \_IOR(type,nr,size) //该命令是从驱动读取数据

#define \_IOW(type,nr,size) //该命令是向驱动写入数据

#define \_IOWR(type,nr,size) //双向数据传输

type是幻数，nr是序号，size是大小。

# 三、测试