

Linux系统内核定时器机制详解（上）（4）PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/144/2021\\_2022\\_Linux\\_E7\\_B3\\_BB\\_E7\\_BB\\_c103\\_144888.htm](https://www.100test.com/kao_ti2020/144/2021_2022_Linux_E7_B3_BB_E7_BB_c103_144888.htm)

7 . 6 . 3 . 2 动态定时器的时钟滴答基准timer\_jiffies 由于动态定时器是在时钟中断的Bottom Half中被执行的，而从TIMER\_BH向量被激活到其timer\_bh()函数真正执行这段时间内可能会有几次时钟中断发生。因此内核必须记住上一次运行定时器机制是什么时候，也即内核必须保存上一次运行定时器机制时的jiffies值。为此，Linux

在kernel/timer.c文件中定义了全局变量timer\_jiffies来表示上一次运行定时器机制时的jiffies值。该变量的定义如下所示：

```
static unsigned long timer_jiffies;
```

7 . 6 . 3 . 3 对内核动态定时器链表的保护 由于内核动态定时器链表是一种系统全局共享资源，为了实现对它的互斥访问，Linux定义了专门的自旋锁timerlist\_lock来保护。任何想要访问动态定时器链表的代码段都首先必须先持有该自旋锁，并且在访问结束后释放该自旋锁。其定义如下（kernel/timer.c）：

```
/* Initialize both explicitly - lets try to have them in the same cache line */ spinlock_t timerlist_lock = SPIN_LOCK_UNLOCKED;
```

100Test 下载频道开通，各类考试题目直接下载。详细请访问 [www.100test.com](http://www.100test.com)