<https://blog.csdn.net/lixianlin/article/details/25604779>

在linux c编程中，setitimer是一个比较常用的函数，可用来实现延时和定时的功能，网上有各种零零散散的用法说明，都只提到了个别用法，今天抽空实践整理了一份比较详细的：

使用时需要引入的头文件：

**[cpp]** [view plain](https://blog.csdn.net/lixianlin/article/details/25604779) [copy](https://blog.csdn.net/lixianlin/article/details/25604779)

1. #include <sys/time.h>

setitimer函数原型：

**[cpp]** [view plain](https://blog.csdn.net/lixianlin/article/details/25604779) [copy](https://blog.csdn.net/lixianlin/article/details/25604779)

1. **int** setitimer(**int** which, **const** **struct** itimerval \*new\_value,
2. **struct** itimerval \*old\_value);

其中which参数表示类型，可选的值有：

**ITIMER\_REAL**：以系统真实的时间来计算，它送出SIGALRM信号。

**ITIMER\_VIRTUAL**：以该进程在用户态下花费的时间来计算，它送出SIGVTALRM信号。

**ITIMER\_PROF**：以该进程在用户态下和内核态下所费的时间来计算，它送出SIGPROF信号。

紧接着的new\_value和old\_value均为itimerval结构体，先看一下itimerval结构体定义：

**[cpp]** [view plain](https://blog.csdn.net/lixianlin/article/details/25604779) [copy](https://blog.csdn.net/lixianlin/article/details/25604779)

1. **struct** itimerval {
2. **struct** timeval it\_interval; /\* next value \*/
3. **struct** timeval it\_value;    /\* current value \*/
4. };
6. **struct** timeval {
7. **time\_t**      tv\_sec;         /\* seconds \*/
8. suseconds\_t tv\_usec;        /\* microseconds \*/
9. };

itimeval又是由两个timeval结构体组成，timeval包含tv\_sec和tv\_usec两部分，其中tv\_se为秒，tv\_usec为微秒(即1/1000000秒)

其中的new\_value参数用来对计时器进行设置，it\_interval为计时间隔，it\_value为延时时长，下面例子中表示的是在setitimer方法调用成功后，延时1微秒便触发一次SIGALRM信号，以后每隔200毫秒触发一次SIGALRM信号。

**settimer工作机制是，先对it\_value倒计时，当it\_value为零时触发信号，然后重置为it\_interval，继续对it\_value倒计时，一直这样循环下去。**

基于此机制，setitimer既可以用来延时执行，也可定时执行。

**假如it\_value为0是不会触发信号的，所以要能触发信号，it\_value得大于0；如果it\_interval为零，只会延时，不会定时（也就是说只会触发一次信号)。**

old\_value参数，通常用不上，设置为NULL，它是用来存储上一次setitimer调用时设置的new\_value值。

以下是一个简单的使用例子：

**[cpp]** [view plain](https://blog.csdn.net/lixianlin/article/details/25604779) [copy](https://blog.csdn.net/lixianlin/article/details/25604779)

1. #include <stdio.h>
2. #include <signal.h>
3. #include <sys/time.h>
5. **void** signalHandler(**int** signo)
6. {
7. **switch** (signo){
8. **case** SIGALRM:
9. printf("Caught the SIGALRM signal!\n");
10. **break**;
11. }
12. }
14. **int** main(**int** argc, **char** \*argv[])
15. {
16. signal(SIGALRM, signalHandler);
18. **struct** itimerval new\_value, old\_value;
19. new\_value.it\_value.tv\_sec = 0;
20. new\_value.it\_value.tv\_usec = 1;
21. new\_value.it\_interval.tv\_sec = 0;
22. new\_value.it\_interval.tv\_usec = 200000;
23. setitimer(ITIMER\_REAL, &new\_value, &old\_value);
25. **for**(;;);
27. **return** 0;
28. }

参考链接：

<http://man7.org/linux/man-pages/man2/setitimer.2.html>

版权声明：本文为博主原创文章，未经博主允许不得转载。 https://blog.csdn.net/lixianlin/article/details/25604779