

导航

博客园

首 页

新随笔

联 系

订 阅

管 理

<2013年1月>

日一 二 三 四 五 六

30 31 1 2 3 4 5

6 7 8 9 10 11 12

13 14 15 16 17 18 19

20 21 22 23 24 25 26

27 28 29 30 31 1 2

3 4 5 6 7 8 9

公告

有事您Q我!!

外刊IT评论

鸟哥的Linux私房菜

HelloGCC

刘未鹏

OCaml官方网站

Frama-C

070714

今日IP[19]

今日PV[36]

昨日IP[44]

51.la 站长统计

站长统计昨日PV[82]

当前在线[0]

站长统计-当前在线[0]

昵称：小交响曲

园龄：7年3个月

粉丝：8

关注：12

+加关注

搜索

找找看

谷歌搜索

我的标签

VIM(1)

随笔分类(111)

.NET(7)

C/C++(7)

Java(4)

Linux(13)

Lua(8)

php(3)

Python(7)

Shell Script(10)

啊哈！灵机一动(8)

大众计算机(17)

待分类(6)

函数式编程(4)

开发那些事儿(5)

求职历程(6)

数据库(3)

系统安全(3)

最新评论

1. Re:HDU 2504 又见GCD

@Bourbon引用我第一次也以为是2b，呵呵没想到自己2b了...

--Barcelona&&cl

2. Re:安装Phoenix-SDK-June-2008-RC1

你好，请问能提供一份Phoenix-SDK-June-2008-RC1.msi文嘛，我项研究这个东西，但MS已经不提供下载了。其他版本也行。谢谢。

--BinSys

3. Re:Ubuntu10.04下安装GCC4.1.2

按照你的方法还是提示依赖关系的错误？怎么办

--海南默然

4. Re:跳过优酷视频的广告链接已经失效。

--垂钧玄武

5. Re:T公司的一道机试题

说明：1、上面我贴出的代码有错，factor的第0项多乘了一个25，应该是：int factor[] = {1+25+25*25+25*25*25, 1+25+25*25, 1+25, 1};2、上面.....

--莫莫4

linux的下两种定时器实现

定时器是比较常用的功能。一直很好奇底层的实现方法。我想最终是使用了CPU的硬件定时器中断机制。有两种实现方法，第一种精度较低，实际使用中并不多。

alarm方式

如果不要求很精确的话，用 alarm() 和 signal() 就够了

unsigned int alarm(unsigned int seconds)

专门为SIGALRM信号而设，在指定的时间seconds秒后，将向进程本身发送SIGALRM信号，又称为闹钟时间。进程调用alarm后，任何以前的alarm()调用都将无效。如果参数seconds为零，那么进程内将不再包含任何闹钟时间。如果调用alarm（）前，进程中已经设置了闹钟时间，则返回上一个闹钟时间的剩余时间，否则返回0。参考示例如下：



```
1 #include<iostream>
2 #include<unistd.h>
3 #include<signal.h>
4 using namespace std;
5
6 void signal_fun(int sig)
7 {
8     cout<<"alarm, sig="<<sig<<endl;
9     alarm(2);
10    return;
11 }
12
13 int main(void)
14 {
15     signal(SIGALRM, signal_fun);
16     alarm(2);
17
18     while(1) pause();
19
20     return 0;
21 }
```



settimer方式

int setitimer(int which, const struct itimerval *value, struct itimerval *ovalue));

setitimer()比alarm功能强大，支持3种类型的定时器：

ITIMER_REAL：以系统真实的时间来计算，它送出SIGALRM信号。

ITIMER_VIRTUAL：以该行程真正有执行的时间来计算，它送出SIGVTALRM信号。

ITIMER_PROF：以行程真正有执行及在核心中所费的时间来计算，它送出SIGPROF信号。

Setitimer()第一个参数which指定定时器类型（上面三种之一）；第二个参数是结构itimerval的一个实例；第三个参数可不作处理。

Setitimer()调用成功返回0，否则返回-1。

参考示例如下：

按 Ctrl+C 复制代码

```
#include <stdio.h>
#include <time.h>
#include <sys/time.h>
#include <stdlib.h>
#include <signal.h>
int count = 0;
void set_timer()
{
    struct itimerval itv, oldtv;
    itv.it_interval.tv_sec = 5;
    itv.it_interval.tv_usec = 0;
    itv.it_value.tv_sec = 5;
    itv.it_value.tv_usec = 0;

    setitimer(ITIMER_REAL, &itv, &oldtv);
}

void sigalrm_handler(int sig)
{
    count++;
    printf("timer signal.. %d\n", count);
}

int main()
{
    printf("%d,%d,%d\n", SIGSEGV, SIGALRM, SIGRTMIN);
    signal(SIGALRM, sigalrm_handler);
    set_timer();
    while (count < 1000)
    {}
    exit(0);
}
```

按 Ctrl+C 复制代码

分类: C/C++,Linux

好文要顶

关注我

收藏该文







小交响曲

关注 - 12

粉丝 - 8

+加关注

« 上一篇：linux中的信号处理

» 下一篇：【lua】table是否为空的判断