

计算机系统体系结构 Project1

517030910326 王孝诚

2019.9.6

1 设计思路

根据教材给出的 hello.c 的代码进行设计。

1.1 jiffies 模块

需要创建一个名为 /proc/jiffies 的 /proc 文件，当被执行 read 操作时，jiffies 的值会被打印出来；当移除 jiffies 模块时，/proc/jiffies 文件也被移除。所以该模块需要 linux/jiffies.h 头文件，需要 proc_init 来创建 /proc/jiffies 文件，proc_exit 函数用于移除该文件，以及 proc_read 函数，其中是实现获得当前 jiffies 值并将其存入 buffer、复制 buffer 到用户空间等操作。

1.2 seconds 模块

需要创建一个名为 /proc/seconds 的 /proc 文件，当被执行 read 操作时，此刻距模块被加载的时刻度过的秒数将被打印出来；当移除 seconds 模块时，/proc/seconds 文件也被移除。该模块的设计和 jiffies 模块类似，不同在于除了 linux/jiffies.h 头文件还需要 asm/param.h 头文件加载模块创建 /proc/seconds 时，即执行 proc_init 时，需要用一个全局变量存储当前的 jiffies 值，然后在执行 proc_read 时，读取实时 jiffies 值并将其与存储的 jiffies 相减并除以 HZ 数，从而得到秒数，将其以类似 jiffies 中的操作打印出来即可。

2 实验环境

在 windows 10 下，使用 VMWare Workstation 15 Player 创建和运行虚拟机，虚拟机环境是 liunx 发行版 Ubuntu16.04.6 LTS。

3 核心代码解释

3.1 jiffies 模块

首先需要 include <linux/jiffies.h> 头文件，并且修改 PROC_NAME 为 jiffies。

```
1 #include <linux/jiffies.h>
2 #define PROC_NAME "jiffies"
```

然后在 proc_init 中调用 proc_create 创建 /proc/jiffies 文件。

```
1 /* This function is called when the module is loaded. */
2 static int proc_init(void)
3 {
4     // creates the /proc/ jiffies entry
```

```

5      // the following function call is a wrapper for
6      // proc_create_data() passing NULL as the last argument
7      proc_create(PROC_NAME, 0, NULL, &proc_ops);
8      printk(KERN_INFO "/proc/%s_created\n", PROC_NAME);
9      return 0;
10 }

```

在 `proc_exit` 中调用 `remove_proc_entry` 来移除 `/proc/jiffies` 文件。

```

1  /* This function is called when the module is removed. */
2  static void proc_exit(void) {
3      // removes the /proc/jiffies entry
4      remove_proc_entry(PROC_NAME, NULL);
5      printk( KERN_INFO "/proc/%s_removed\n", PROC_NAME);
6  }

```

每次输入 `cat /proc/jiffies` 指令，将调用 `proc_read` 函数，定义如下。

```

1  static ssize_t proc_read(struct file *file , char __user *usr_buf, size_t count, loff_t *
    pos)
2  {
3      int rv = 0;
4      char buffer[BUFFER_SIZE];
5      static int completed = 0;
6      if (completed) {
7          completed = 0;
8          return 0;
9      }
10     completed = 1;
11
12     //将 jiffies 存入kernel memory下的buffer
13     rv = sprintf( buffer , "%lu\n", jiffies );
14     //将信息从buffer移动至用户空间
15     copy_to_user(usr_buf, buffer, rv);
16     return rv;
17 }

```

3.2 seconds 模块

与 `jiffies` 模块类似，多了几处操作。首先要 `include <asm/param.h>` 头文件，并修改 `PROC_NAME` 为 `seconds`。

```

1  #include <asm/param.h>
2  #define PROC_NAME "seconds"

```

要在 `proc_init` 中存储开始时的 `jiffies`。

```

1  /* This function is called when the module is loaded. */
2  static int proc_init(void)
3  {
4      proc_create(PROC_NAME, 0, NULL, &proc_ops);
5      start = jiffies ;

```

```

6      printk(KERN_INFO "/proc/%s_created\n", PROC_NAME);
7      return 0;
8  }

```

在 proc_read 中打印秒数。

```

1  rv = sprintf(buffer, "%lu\n", ( jiffies -start)/HZ);

```

4 实验结果

```

xcwang@ubuntu:~/Documents/osc10e-master/ch2$ sudo insmod jiffies.ko
[sudo] password for xcwang:
xcwang@ubuntu:~/Documents/osc10e-master/ch2$ sudo insmod seconds.ko
xcwang@ubuntu:~/Documents/osc10e-master/ch2$ lsmod
Module                  Size  Used by
seconds                 16384  0
jiffies                 16384  0

```

Figure 1: 加载模块

```

xcwang@ubuntu:~/Documents/osc10e-master/ch2$ ls /proc
1      1494  191  202  214  226  239  2698  280  2968  3052  3113  3225  3346  37  740  87  asound      fb          kpagecgroup  pagetypeinfo  sysvipc
10     15    192  203  215  227  24  2699  282  2971  3057  3143  3242  3364  372  742  88  buddyinfo   filesystems  kpagecount   partitions     thread-self
1017   1507  193  204  216  228  240  27  2877  2975  3058  3169  3264  3369  379  744  89  bus          fs           kpageflags   sched_debug    timer_list
105    16   194  205  217  229  241  2703  2883  2987  306  3173  3270  3384  380  749  9  cgroups     interrupts   loadavg      schedstat      tty
11     18   195  206  218  230  242  2704  2895  2993  3066  3181  3273  34  4  791  98  cmdline     lsm          locks        secal         uptime
1185   184  196  207  219  231  243  276  29  30  3067  3188  3280  341  402  8  911  consoles   lsm          ndstat       seconds        version
12     185  197  208  22  232  244  277  2922  3003  3068  3196  3286  3446  41  806  931  cpuinfo     meminfo     nls           version_signature  self
122    186  198  209  220  233  245  2786  2925  3005  3069  3197  3291  3471  42  807  940  crypto      jiffies     nls           vmallocinfo      slabinfo
13     187  199  21  221  234  246  2788  2933  3018  3070  32  33  3499  43  808  946  devices     kallsyms    modules       softirqs        vmstat
1318   188  2  210  222  235  247  2789  2938  3019  3071  3205  3305  35  6  822  952  diskstats   kcore       nmounts       stat            zoneinfo
14     189  20  211  223  236  25  2794  2947  3028  31  3209  3308  3506  643  823  96  dna          keys         net           sys            swaps
1447   19  200  212  224  237  26  2796  2952  3029  3102  3200  3311  3555  7  85  985  driver      key-users   ntr           sysrq-trigger
1487   190  201  213  225  238  2697  28  2854  305  3112  3224  3323  36  739  86  acpi         execdomains knsg          net

```

Figure 2: 加载后/proc 下的文件

```

xcwang@ubuntu:~/Documents/osc10e-master/ch2$ cat /proc/jiffies
4295138776
xcwang@ubuntu:~/Documents/osc10e-master/ch2$ cat /proc/jiffies
4295139226
xcwang@ubuntu:~/Documents/osc10e-master/ch2$ cat /proc/jiffies
4295139657

```

Figure 3: 对/proc/jiffies 进行 read

```

xcwang@ubuntu:~/Documents/osc10e-master/ch2$ cat /proc/seconds
109
xcwang@ubuntu:~/Documents/osc10e-master/ch2$ cat /proc/seconds
111
xcwang@ubuntu:~/Documents/osc10e-master/ch2$ cat /proc/seconds
115

```

Figure 4: 对/proc/seconds 进行 read

实验中遇到了一些问题，比如将教材上代码中的

```

1  #include <asm/uaccess.h>

```

替换成了

```

1  #include <linux/uaccess.h>

```