

Report 6.2

System and device programming
Prof. Pietro Laface

Nicola Sabino

mat:253839
Date: April 10, 2019

Laboratory 6

/proc file system and character device driver.

Exercise 2

```
ssize_t
device_read(struct file *filp, char *buff, size_t count, loff_t *offp) {
    unsigned long ret;
    ret = copy_to_user(buff, char_dev_buf, count);
    return count;
}
```

Since the `count` variable contains the amount of bytes to transfer in user space, the `cat` command will loop indefinitely. My solution is the following one

```
ssize_t
device_read(struct file *filp, char *buff, size_t count, loff_t *offp) {
    unsigned long ret;

    int lenght = strlen(char_dev_buf);
    if(count >= lenght) count = lenght;
    ret = copy_to_user(buff, char_dev_buf, count);
    memset(char_dev_buf, 0, BUF_LEN);
    bytes = 0;
    return count;
}
```

The `memset(char_dev_buf, 0, BUF_LEN)` command erase the buffer in order to break `cat` at the following reading.

Regarding the possibiity to insert several strings with the `echo` command, my solution is the following

```
ssize_t
device_write(struct file *filp, const char *buff, size_t count, loff_t *offp) {
    unsigned long ret;
    char_dev_buf[strcspn(char_dev_buf, "\n")] = ' ';
    ret = copy_from_user(char_dev_buf + bytes, buff, count);
    bytes += count;
    return count;
}
}
```

At each writing operation the number of bytes is append to the global variable `bytes`, and emptied at each reading operation. The command

```
char_dev_buf[strcspn(char_dev_buf, "\n")] = ' '
```

substitutes the final new line with a simple space in the previous version of the buffer, before the new `copy_from_user(...)`.

Examples of usage

```
root@VMdebian6:/home/laface/Desktop/chardev_SDP_lab# make
make -C /lib/modules/2.6.32-5-686/build M=/home/laface/Desktop/chardev_SDP_lab modules
make[1]: Entering directory `/usr/src/linux-headers-2.6.32-5-686'
  Building modules, stage 2.
  MODPOST 1 modules
make[1]: Leaving directory `/usr/src/linux-headers-2.6.32-5-686'
root@VMdebian6:/home/laface/Desktop/chardev_SDP_lab# insmod chardev_SDP_lab.ko
root@VMdebian6:/home/laface/Desktop/chardev_SDP_lab# mknod /dev/chardev_SDP_lab c 251 0
root@VMdebian6:/home/laface/Desktop/chardev_SDP_lab# echo something > /dev/chardev_SDP_lab
root@VMdebian6:/home/laface/Desktop/chardev_SDP_lab# cat /dev/chardev_SDP_lab
something
root@VMdebian6:/home/laface/Desktop/chardev_SDP_lab# echo xxx > /dev/chardev_SDP_lab
root@VMdebian6:/home/laface/Desktop/chardev_SDP_lab# cat /dev/chardev_SDP_lab
xxx
root@VMdebian6:/home/laface/Desktop/chardev_SDP_lab# echo "First and" > /dev/chardev_SDP_lab
root@VMdebian6:/home/laface/Desktop/chardev_SDP_lab# echo "second string" > /dev/chardev_SDP_lab
root@VMdebian6:/home/laface/Desktop/chardev_SDP_lab# cat /dev/chardev_SDP_lab
First and second string
root@VMdebian6:/home/laface/Desktop/chardev_SDP_lab# rmmod chardev_SDP_lab
root@VMdebian6:/home/laface/Desktop/chardev_SDP_lab# rm /dev/chardev_SDP_lab
root@VMdebian6:/home/laface/Desktop/chardev_SDP_lab# □
```