

DEVICE DRIVERS – LAB EXERCISE 6

Submitted By:

Kiran Thomas Cherian

CED18I028

OBJECTIVE:

Write a simple Char Device Driver Program in C language. Compile it as a kernel module. Insert in the kernel. Check whether the device file is created in proper directory. Check by writing data into the device file and reading data from the device file.

Linux Distribution Used:

MX Linux 19.1 (Running on Virtual machine)

```
kiran@LastNightmare00:~/Desktop
$ cat /etc/*-release
NAME="MX"
VERSION="19.1 (patito feo)"
ID="mx"
VERSION_ID="19.1"
PRETTY_NAME="MX 19.1 (patito feo)"
ANSI_COLOR="0;34"
HOME_URL="https://mxlinux.org"
BUG_REPORT_URL="https://mxlinux.org"
PRETTY_NAME="MX 19.1 patito feo"
DISTRIB_ID=MX
DISTRIB_RELEASE=19.1
DISTRIB_CODENAME="patito feo"
DISTRIB_DESCRIPTION="MX 19.1 patito feo"
PRETTY_NAME="Debian GNU/Linux 10 (buster)"
NAME="Debian GNU/Linux"
VERSION_ID="10"
VERSION="10 (buster)"
VERSION_CODENAME=buster
ID=debian
HOME_URL="https://www.debian.org/"
SUPPORT_URL="https://www.debian.org/support"
BUG_REPORT_URL="https://bugs.debian.org/"
```

Code:

```
#include <linux/init.h>
#include <linux/module.h>
#include <linux/cdev.h>
#include <linux/device.h>
#include <linux/kernel.h>
#include <linux/uaccess.h>
#include <linux/fs.h>

#define MAX_DEV 1

static int mychardev_open(struct inode *inode, struct file *file);
static int mychardev_release(struct inode *inode, struct file *file);
static ssize_t mychardev_read(struct file *file, char __user *buf, size_t count,
loff_t *offset);
static ssize_t mychardev_write(struct file *file, const char __user *buf, size_t
count, loff_t *offset);

static const struct file_operations mychardev_fops = {
    .owner    = THIS_MODULE,
    .open     = mychardev_open,
    .release  = mychardev_release,
    .read     = mychardev_read,
    .write    = mychardev_write
```

```
};
```

```
struct mychar_device_data {  
    struct cdev cdev;  
};
```

```
static int dev_major = 0;  
static struct class *mychardev_class = NULL;  
static struct mychar_device_data mychardev_data[MAX_DEV];
```

```
static int mychardev_uevent(struct device *dev, struct kobj_uevent_env *env)  
{  
    add_uevent_var(env, "DEVMODE=%#o", 0666);  
    return 0;  
}
```

```
static int __init mychardev_init(void)  
{  
    int err, i;  
    dev_t dev;  
  
    err = alloc_chrdev_region(&dev, 0, MAX_DEV, "kiranchardev");  
  
    dev_major = MAJOR(dev);  
  
    mychardev_class = class_create(THIS_MODULE, "kiranchardev");
```

```

mychardev_class->dev_uevent = mychardev_uevent;

for (i = 0; i < MAX_DEV; i++) {
    cdev_init(&mychardev_data[i].cdev, &mychardev_fops);
    mychardev_data[i].cdev.owner = THIS_MODULE;

    cdev_add(&mychardev_data[i].cdev, MKDEV(dev_major, i), 1);

    device_create(mychardev_class, NULL, MKDEV(dev_major, i), NULL,
"kiranchardev-%d", i);
}

    printk(KERN_INFO "Inserting my module..\n");
return 0;
}

static void __exit mychardev_exit(void)
{
    int i;

    for (i = 0; i < MAX_DEV; i++) {
        device_destroy(mychardev_class, MKDEV(dev_major, i));
    }

    class_unregister(mychardev_class);
    class_destroy(mychardev_class);

    unregister_chrdev_region(MKDEV(dev_major, 0), MINORMASK);
}

```

```
    printk(KERN_INFO "Removing my module.bye..\n");  
}
```

```
static int mychardev_open(struct inode *inode, struct file *file)  
{  
    printk("KIRANCHARDEV: Device opened\n");  
    return 0;  
}
```

```
static int mychardev_release(struct inode *inode, struct file *file)  
{  
    printk("KIRANCHARDEV: Device closed\n");  
    return 0;  
}
```

```
static ssize_t mychardev_read(struct file *file, char __user *buf, size_t count,  
loff_t *offset)  
{  
    uint8_t *data = "Greetings from the kernel side\n";  
    size_t datalen = strlen(data);  
  
    printk("Reading from device: %d\n", MINOR(file->f_path.dentry->d_inode->i_rdev));  
  
    if (count > datalen) {  
        count = datalen;
```

```
}
```

```
if (copy_to_user(buf, data, count)) {
```

```
    return -EFAULT;
```

```
}
```

```
return count;
```

```
}
```

```
static ssize_t mychardev_write(struct file *file, const char __user *buf, size_t  
count, loff_t *offset)
```

```
{
```

```
    size_t maxdatalen = 30, ncopied;
```

```
    uint8_t databuf[maxdatalen];
```

```
    printk("Writing to device: %d\n", MINOR(file->f_path.dentry->d_inode->  
i_rdev));
```

```
    if (count < maxdatalen) {
```

```
        maxdatalen = count;
```

```
    }
```

```
    ncopied = copy_from_user(databuf, buf, maxdatalen);
```

```
    if (ncopied == 0) {
```

```
        printk("Copied %zd bytes from the user\n", maxdatalen);
```

```
    } else {
```

```

        printk("Could't copy %zd bytes from the user\n", ncopied);
    }

    databuf[maxdatalen] = 0;

    printk("Data from user side: %s\n", databuf);

    return count;
}

MODULE_LICENSE("GPL");
MODULE_AUTHOR("Kiran Thomas Cherian");

module_init(mychardev_init);
module_exit(mychardev_exit);

```

Makefile Contents:

```

BINARY    := kiranchardev
KERNEL    := /lib/modules/$(shell uname -r)/build
ARCH      := x86
C_FLAGS   := -Wall
KMOD_DIR   := $(shell pwd)
TARGET_PATH := /lib/modules/$(shell uname -r)/kernel/drivers/char

```



```
OBJECTS := CED18I028Lab6.o
```

```
ccflags-y += $(C_FLAGS)
```

```
obj-m += $(BINARY).o
```

```
$(BINARY)-y := $(OBJECTS)
```

```
$(BINARY).ko:
```

```
    make -C $(KERNEL) M=$(KMOD_DIR) modules
```

```
install:
```

```
    cp $(BINARY).ko $(TARGET_PATH)
```

```
    depmod -a
```

```
uninstall:
```

```
    rm $(TARGET_PATH)/$(BINARY).ko
```

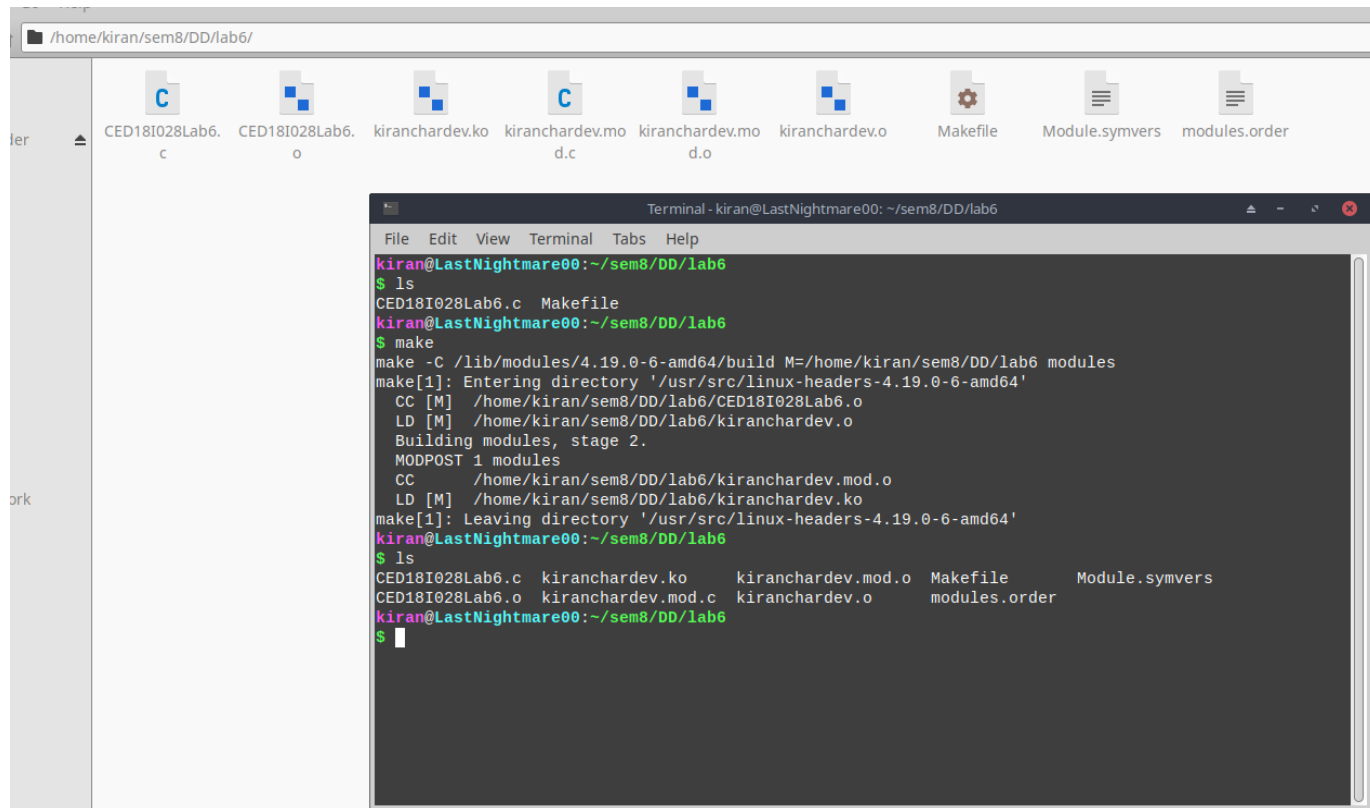
```
    depmod -a
```

```
clean:
```

```
    make -C $(KERNEL) M=$(KMOD_DIR) clean
```

Name given: kiranchardev

Run the make command to compile the source code.



The screenshot shows a file manager window with the path `/home/kiran/sem8/DD/lab6/`. The file list includes `CED18I028Lab6.c`, `CED18I028Lab6.o`, `kiranchardev.ko`, `kiranchardev.mod.c`, `kiranchardev.mod.o`, `kiranchardev.o`, `Makefile`, `Module.symvers`, and `modules.order`. Overlaid on this is a terminal window titled `Terminal - kiran@LastNightmare00: ~/sem8/DD/lab6`. The terminal shows the following commands and output:

```
kiran@LastNightmare00:~/sem8/DD/lab6
$ ls
CED18I028Lab6.c  Makefile
kiran@LastNightmare00:~/sem8/DD/lab6
$ make
make -C /lib/modules/4.19.0-6-amd64/build M=/home/kiran/sem8/DD/lab6 modules
make[1]: Entering directory '/usr/src/linux-headers-4.19.0-6-amd64'
CC [M] /home/kiran/sem8/DD/lab6/CED18I028Lab6.o
LD [M] /home/kiran/sem8/DD/lab6/kiranchardev.o
Building modules, stage 2.
MODPOST 1 modules
CC /home/kiran/sem8/DD/lab6/kiranchardev.mod.o
LD [M] /home/kiran/sem8/DD/lab6/kiranchardev.ko
make[1]: Leaving directory '/usr/src/linux-headers-4.19.0-6-amd64'
kiran@LastNightmare00:~/sem8/DD/lab6
$ ls
CED18I028Lab6.c  kiranchardev.ko      kiranchardev.mod.o  Makefile      Module.symvers
CED18I028Lab6.o  kiranchardev.mod.c  kiranchardev.o      modules.order
kiran@LastNightmare00:~/sem8/DD/lab6
$
```

Then use `insmod` to load the module.

You can get information about the module using the `modinfo` command, which will identify the description, author and any module parameters that are defined. And to see the message, we need to read the `kern.log` in `/var/log` directory

```

CED18I028Lab6. CED18I028Lab6. kiranchardev.ko kiranchardev.mo kiranchardev.mo kiranchardev.o Makefile Module.symvers modules.order
Terminal - kiran@LastNightmare00: ~/sem8/DD/lab6
File Edit View Terminal Tabs Help
MODPOST 1 modules
CC      /home/kiran/sem8/DD/lab6/kiranchardev.mod.o
LD [M]  /home/kiran/sem8/DD/lab6/kiranchardev.ko
make[1]: Leaving directory '/usr/src/linux-headers-4.19.0-6-amd64'
kiran@LastNightmare00:~/sem8/DD/lab6
$ ls
CED18I028Lab6.c  kiranchardev.ko  kiranchardev.mod.o  Makefile  Module.symvers
CED18I028Lab6.o  kiranchardev.mod.c  kiranchardev.o  modules.order
kiran@LastNightmare00:~/sem8/DD/lab6
$ sudo insmod kiranchardev.ko
kiran@LastNightmare00:~/sem8/DD/lab6
$ modinfo kiranchardev.ko
filename:       /home/kiran/sem8/DD/lab6/kiranchardev.ko
author:        Kiran Thomas Cherian
license:       GPL
depends:
retpoline:     Y
name:          kiranchardev
vermagic:      4.19.0-6-amd64 SMP mod_unload modversions
kiran@LastNightmare00:~/sem8/DD/lab6
$ sudo tail /var/log/kern.log
Mar 31 15:33:36 LastNightmare00 kernel: [ 2539.226675] RIP: 0033:0x7f6ac20a3dd7
Mar 31 15:33:36 LastNightmare00 kernel: [ 2539.226676] Code: 73 01 c3 48 8b 0d b9 10 0c 00 f7 d8 64 89 01 48 83 c8 ff c3 66 2e 0f 1f 84 00 00 00
00 00 0f 1f 44 00 00 b8 b0 00 00 00 0f 05 <48> 3d 01 f0 ff ff 73 01 c3 48 8b 0d 89 10 0c 00 f7 d8 64 89 01 48
Mar 31 15:33:36 LastNightmare00 kernel: [ 2539.226677] RSP: 002b:00007ffdc281f378 EFLAGS: 00000206 ORIG_RAX: 00000000000000b9
Mar 31 15:33:36 LastNightmare00 kernel: [ 2539.226678] RAX: ffffffff00000000 RBX: 0000560b3ed87760 RCX: 00007f6ac20a3dd7
Mar 31 15:33:36 LastNightmare00 kernel: [ 2539.226678] RDX: 000000000000000a RSI: 0000000000000000 RDI: 0000560b3ed877c8
Mar 31 15:33:36 LastNightmare00 kernel: [ 2539.226679] RBP: 0000000000000000 R08: 00007ffdc281e2f1 R09: 0000000000000000
Mar 31 15:33:36 LastNightmare00 kernel: [ 2539.226679] R10: 00007f6ac2115ae0 R11: 0000000000000206 R12: 00007ffdc281f5a0
Mar 31 15:33:36 LastNightmare00 kernel: [ 2539.226680] R13: 00007ffdc281f8b2 R14: 0000560b3ed87260 R15: 0000560b3ed87760
Mar 31 15:37:42 LastNightmare00 kernel: [ 2539.226681] ---[ end trace 733dc493b809b0a0 ]---
Mar 31 15:37:42 LastNightmare00 kernel: [ 2784.984221] Inserting my module..
kiran@LastNightmare00:~/sem8/DD/lab6
$
```

Checking the tree structure of the file created, and also verifying that it is a char device:

```

Terminal - kiran@LastNightmare00: ~/sem8/DD/lab6
File Edit View Terminal Tabs Help
Mar 31 15:33:36 LastNightmare00 kernel: [ 2539.226680] R13: 00007ffdc281f8b2 R14: 0000560b3ed87260 R15: 0000560b3ed87760
Mar 31 15:33:36 LastNightmare00 kernel: [ 2539.226681] ---[ end trace 733dc493b809b0a0 ]---
Mar 31 15:37:42 LastNightmare00 kernel: [ 2784.984221] Inserting my module..
kiran@LastNightmare00:~/sem8/DD/lab6
$ ls -l /dev/kiranchardev-0
crw-rw-rw- 1 root root 243, 0 Mar 31 15:37 /dev/kiranchardev-0
kiran@LastNightmare00:~/sem8/DD/lab6
$ tree /sys/devices/virtual/kiranchardev/
/sys/devices/virtual/kiranchardev/
├── kiranchardev-0
│   ├── dev
│   │   └── power
│   │       ├── async
│   │       ├── autosuspend_delay_ms
│   │       ├── control
│   │       ├── runtime_active_kids
│   │       ├── runtime_active_time
│   │       ├── runtime_enabled
│   │       ├── runtime_status
│   │       ├── runtime_suspended_time
│   │       └── runtime_usage
│   └── subsystem -> ../../../../class/kiranchardev
│       └── uevent
3 directories, 11 files
kiran@LastNightmare00:~/sem8/DD/lab6
$
```

Reading data from device file:

```
Terminal - kiran@LastNightmare00: ~/sem8/DD/lab6
File Edit View Terminal Tabs Help
├─ runtime_usage
├─ subsystem -> ../../../../class/kiranchardev
└─ uevent

3 directories, 11 files
kiran@LastNightmare00:~/sem8/DD/lab6
$ head -c31 /dev/kiranchardev-0
Greetings from the kernel side
kiran@LastNightmare00:~/sem8/DD/lab6
$ sudo tail /var/log/kern.log
Mar 31 15:33:36 LastNightmare00 kernel: [ 2539.226678] RAX: ffffffffda RBX: 0000560b3ed87760 RCX: 00007f6ac20a3dd7
Mar 31 15:33:36 LastNightmare00 kernel: [ 2539.226678] RDX: 000000000000000a RSI: 0000000000000000 RDI: 0000560b3ed877c8
Mar 31 15:33:36 LastNightmare00 kernel: [ 2539.226679] RBP: 0000000000000000 R08: 00007ffdc281e2f1 R09: 0000000000000000
Mar 31 15:33:36 LastNightmare00 kernel: [ 2539.226679] R10: 00007f6ac2115ae0 R11: 00000000000000206 R12: 00007ffdc281f5a0
Mar 31 15:33:36 LastNightmare00 kernel: [ 2539.226680] R13: 00007ffdc281f8b2 R14: 0000560b3ed87760 R15: 0000560b3ed87760
Mar 31 15:33:36 LastNightmare00 kernel: [ 2539.226681] ---[ end trace 733dc493b809b0a0 ]---
Mar 31 15:37:42 LastNightmare00 kernel: [ 2784.984221] Inserting my module..
Mar 31 15:44:53 LastNightmare00 kernel: [ 3215.661275] KIRANCHARDEV: Device opened
Mar 31 15:44:53 LastNightmare00 kernel: [ 3215.661279] Reading from device: 0
Mar 31 15:44:53 LastNightmare00 kernel: [ 3215.661294] KIRANCHARDEV: Device closed
kiran@LastNightmare00:~/sem8/DD/lab6
$
```

Writing data to device file:

```
Terminal - kiran@LastNightmare00: ~/sem8/DD/lab6
File Edit View Terminal Tabs Help
Mar 31 15:37:42 LastNightmare00 kernel: [ 2784.984221] Inserting my module..
Mar 31 15:44:53 LastNightmare00 kernel: [ 3215.661275] KIRANCHARDEV: Device opened
Mar 31 15:44:53 LastNightmare00 kernel: [ 3215.661279] Reading from device: 0
Mar 31 15:44:53 LastNightmare00 kernel: [ 3215.661294] KIRANCHARDEV: Device closed
kiran@LastNightmare00:~/sem8/DD/lab6
$ echo "writing to the device by me" > /dev/kiranchardev-0
kiran@LastNightmare00:~/sem8/DD/lab6
$ sudo tail /var/log/kern.log
Mar 31 15:37:42 LastNightmare00 kernel: [ 2784.984221] Inserting my module..
Mar 31 15:44:53 LastNightmare00 kernel: [ 3215.661275] KIRANCHARDEV: Device opened
Mar 31 15:44:53 LastNightmare00 kernel: [ 3215.661279] Reading from device: 0
Mar 31 15:44:53 LastNightmare00 kernel: [ 3215.661294] KIRANCHARDEV: Device closed
Mar 31 15:46:43 LastNightmare00 kernel: [ 3326.330208] KIRANCHARDEV: Device opened
Mar 31 15:46:43 LastNightmare00 kernel: [ 3326.330219] Writing to device: 0
Mar 31 15:46:43 LastNightmare00 kernel: [ 3326.330219] Copied 28 bytes from the user
Mar 31 15:46:43 LastNightmare00 kernel: [ 3326.330220] Data from user side: writing to the device by me
Mar 31 15:46:43 LastNightmare00 kernel: [ 3326.330220]
Mar 31 15:46:43 LastNightmare00 kernel: [ 3326.330221] KIRANCHARDEV: Device closed
kiran@LastNightmare00:~/sem8/DD/lab6
$
```

To unload the module, we run `rmmod`, and verify it is removed by checking tree structure again:

```
File Edit View Terminal Tabs Help
Mar 31 15:46:43 LastNightmare00 kernel: [ 3326.330220]
Mar 31 15:46:43 LastNightmare00 kernel: [ 3326.330221] KIRANCHARDEV: Device closed
kiran@LastNightmare00:~/sem8/DD/lab6
$ sudo rmmod kiranchardev
kiran@LastNightmare00:~/sem8/DD/lab6
$ tree /sys/devices/virtual/kiranchardev/
/sys/devices/virtual/kiranchardev/ [error opening dir]

0 directories, 0 files
kiran@LastNightmare00:~/sem8/DD/lab6
$ sudo tail /var/log/kern.log
Mar 31 15:44:53 LastNightmare00 kernel: [ 3215.661275] KIRANCHARDEV: Device opened
Mar 31 15:44:53 LastNightmare00 kernel: [ 3215.661279] Reading from device: 0
Mar 31 15:44:53 LastNightmare00 kernel: [ 3215.661294] KIRANCHARDEV: Device closed
Mar 31 15:46:43 LastNightmare00 kernel: [ 3326.330208] KIRANCHARDEV: Device opened
Mar 31 15:46:43 LastNightmare00 kernel: [ 3326.330219] Writing to device: 0
Mar 31 15:46:43 LastNightmare00 kernel: [ 3326.330219] Copied 28 bytes from the user
Mar 31 15:46:43 LastNightmare00 kernel: [ 3326.330220] Data from user side: writing to the device by me
Mar 31 15:46:43 LastNightmare00 kernel: [ 3326.330220]
Mar 31 15:46:43 LastNightmare00 kernel: [ 3326.330221] KIRANCHARDEV: Device closed
Mar 31 15:47:55 LastNightmare00 kernel: [ 3398.226665] Removing my module.bye..
kiran@LastNightmare00:~/sem8/DD/lab6
$
```