

**Devices**

***Objective: In this lab you will follow a walkthrough to build a character device*** chardev ***and communicate with it via a user mode application called*** chartest***.***

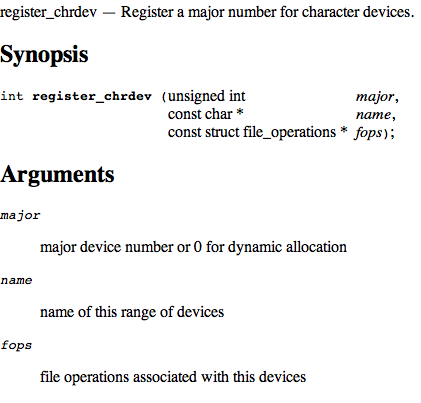
/dev is the location of special or device files. It is a very interesting directory that highlights one important aspect of the Linux filesystem - everything is a file or a directory. Look through this directory and you should hopefully see ex. sda1, sda2 etc.... which represent the various partitions on the first master drive of the system. /dev/cdrom and /dev/fd0 represent your CD-ROM drive and your floppy drive. This may seem strange but it will make sense if you compare the characteristics of files to that of your hardware. Both can be read from and written to.

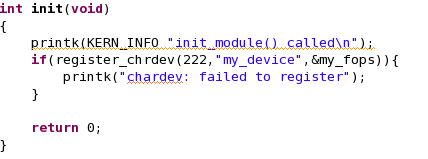
**File(s) for this lab:**

The majority of devices are either block or character devices; however other types of devices exist and can be created. In general, 'block devices' are devices that store or hold data, 'character devices' can be thought of as devices that transmit or transfer data. For example; hard drives and CD-ROM drives are all block devices while serial ports, mice and parallel printer ports are all character devices.

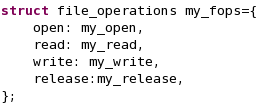
In this lab you will create a character device and communicate with it via it’s /dev entry.

1. To create a /dev entry we first need to use the function register\_chrdev():

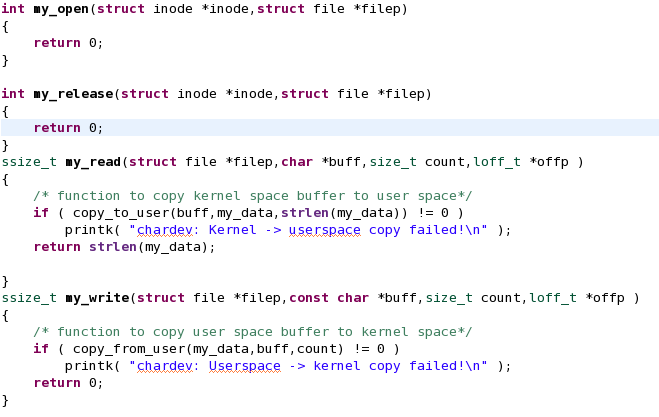




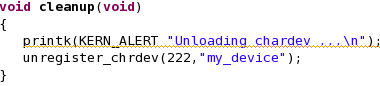
1. We have already been exposed to the file\_operations structure. We need to build our own functions for read and write to communicate with our user mode application.



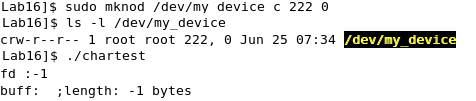
1. For our example open and release will be dummy functions.



1. We’ll use copy\_to\_user() and copy\_from\_user() to read/write with usermode.
2. Finally, we need to unregister the device when we unload the module.



1. To test the module we need to create the usermode device using the mknod command.



1. Initially we get an error when running our user mode application because the module is not loaded yet and it has not registered itself with the device subsystem.

