CS 401 - ASSIGNMENT 3
User Space - Kernel Space Communication

Name: Arun Babu
Roll No: 143050032

Aim

Communication between user and kernel space. The aim is to create a character device using a kernel module. User space programs can read from or write to it. The data associated with the character device is stored inside kernel space as a circular queue. Also, a proc file system entry is made available to check the status of the character device.

Components

1. Circular Queue : Allows enqueue and dequeue of items along with different operations. It is implemented in the *circular_buffer.h* file

2. Character Device : Allows read and write functions. Implemented in *chardev.h* file.

3. Proc FS Entry: Allows reading and writing statistics of the circular queue. Implemented in *procfile.h* file.

4. Kernel Module : Initializes the circular queue, registers character device, proc fs etc. Implemented in *devmodule.c* file.

High Level Algorithm

1. Create a circular queue with reader() and writer() functions
2. Create a Character Device . Implement read and write methods to read to and write from circular queue.
3. Create a proc fs entry and implement read method to read the parameters of circular queue like number of reads,writes, size, pointers, contents etc
4. Register character device and proc fs entry
5. Use mknod to create a device file by giving the Major number given while registering device
6. Read and write to character device from user space.

How to compile module?

1. Change directory to ./module/
2. `make`
3. The file `devmodule.ko` will be generated and it is the module file

How to create character device?

1. Goto directory where `devmodule.ko` resides
2. Execute `sudo insmod devmodule.ko`
3. Use `dmesg` to see the Major number output by module.
4. Execute `sudo mknod /dev/chardev c MAJORNUM 0`
   where MAJORNUM is obtained from step 3
5. You may need to change permission or owner of the file using `chown` and `chmod`
6. Read and write to the character device `/dev/chardev` .

How to access Proc FS entry?

- Access it using command
  `cat /proc/chardev`.

How to test it?

- You can write to the device using the following command.
  `echo 'Message to be written' | tee /dev/chardev`

- You can read from device using the following command.
  `cat /dev/chardev`

- You can also create normal user space programs and open it as a file and read or write.

User-space testing program

- Compile the user space program using the command `gcc device-tester.c -o test`
- Run the program as `./test`

SAMPLE OUTPUTS

User space program

```
Wrote 25 bytes to device
Read 25 bytes from device : Hello from user space..

Writing until queue becomes full..
Queue full.. Capacity : 512
```

Proc FS

```
CHAR DEVICE STATS

        Items           :- 512
        Capacity        :- 512
        Reads           :- 562
        Writes          :- 1074
        Front           :- 50
        Rear            :- 50
        Content         :- 'aaaaaaaaaaaa..' (more content)
```

Proc FS

```
CHAR DEVICE STATS

        Items           :- 0
        Capacity        :- 512
        Reads           :- 1074
        Writes          :- 1074
        Front           :- 50
        Rear            :- 50
        Content         :- ''
```

Dmesg

```
[ 1279.731778] Character device was assigned major number 248. To talk to
[ 1279.731785] the driver, create a dev file with
[ 1279.731790] 'mknod /dev/chardev c 248 0'.
[ 1279.731794] Remove the device file and module when done.
[ 1281.774648] Device opened
[ 1281.775601] Device released
```