COMPSCI 3SH3 Winter, 2020
Student name: Frank Su
Student ID: 001411435
Date: January 22, 2020

Lab 1 Report

Q1 - Jiffies

Module description (1-2 sentences)

This module writes the value of the jiffies global variable that was imported as part of “linux/jiffies.h”. The buffer is saved to the linux kernel pseudo-file system under the directory /proc/jiffies, as defined by PROC_NAME global variable.

Source Code: *jiffies.c*[1]

```
#include <linux/init.h>
#include <linux/kernel.h>
#include <linux/module.h>
#include <linux/proc_fs.h>
#include <linux/jiffies.h>
#include <linux/uaccess.h>

#define BUFFER_SIZE 128
#define PROC_NAME "jiffies"

ssize_t proc_read(struct file *file, char __user *usr_buf, size_t count, loff_t *pos);

static struct file_operations proc_ops = {
    .owner = THIS_MODULE,
    .read = proc_read,
};

int proc_init(void)
{
  printk(KERN_INFO, "Loading jiffies kernel module...");
  proc_create(PROC_NAME, 0666, NULL, &proc_ops);
  return 0;
}

void proc_exit(void)
{
  printk(KERN_INFO, "Removing jiffies kernel module...");
  remove_proc_entry(PROC_NAME, NULL);
}

ssize_t proc_read(struct file *file, char __user *usr_buf, size_t count, loff_t *pos)
{
  int rv = 0;
  char buffer[BUFFER_SIZE];
  static int completed = 0;

  if (completed)
  {
    completed = 0;
    return 0;
  }

  completed = 1;

  rv = sprintf(buffer, "%s%d", "The value of jiffies is: ", jiffies);
  copy_to_user(usr_buf, buffer, rv);

  return rv;
}

module_init(proc_init);
module_exit(proc_exit);

MODULE_LICENSE("GPL");
MODULE_DESCRIPTION("Hello Module");
MODULE_AUTHOR("Frank");
```

Q2 - Seconds

Module description (1-2 sentences)

This module creates a proc file named /proc/seconds and writes to the file the number of seconds that have passed since the kernel module was loaded. The seconds is calculated by first storing the value of jiffies to a global variable when the module is loaded, then calculating the difference in the newly read jiffies value when the file is read (since proc_read() will be called) and then dividing that value by the HZ value.

Source Code: seconds.c

```
#include <linux/init.h>
#include <linux/kernel.h>
#include <linux/module.h>
#include <linux/proc_fs.h>
#include <linux/jiffies.h>
#include <linux/uaccess.h>
#include <asm/param.h>

#define BUFFER_SIZE 128
#define PROC_NAME "seconds"
unsigned long currentJiffies;

ssize_t proc_read(struct file *file, char __user *usr_buf, size_t count, loff_t *pos);

static struct file_operations proc_ops = {
    .owner = THIS_MODULE,
    .read = proc_read,
};

int proc_init(void)
{
  printk(KERN_INFO, "Loading seconds kernel module...");
  currentJiffies = jiffies;
  proc_create(PROC_NAME, 0666, NULL, &proc_ops);
  return 0;
}

void proc_exit(void)
{
  printk(KERN_INFO, "Removing jiffies kernel module...");
  remove_proc_entry(PROC_NAME, NULL);
}

ssize_t proc_read(struct file *file, char __user *usr_buf, size_t count, loff_t *pos)
{
  int rv = 0;
  char buffer[BUFFER_SIZE];
  static int completed = 0;

  if (completed)
  {
    completed = 0;
    return 0;
  }

  completed = 1;

  rv = sprintf(buffer, "%s%d", "The number of seconds that have passed since module was loaded is: ", (jiffies -
currentJiffies)/HZ);
  copy_to_user(usr_buf, buffer, rv);

  return rv;
}

module_init(proc_init);
module_exit(proc_exit);

MODULE_LICENSE("GPL");
MODULE_DESCRIPTION("Hello Module");
MODULE_AUTHOR("Frank");
```