

OS MP1 Report

Saikat Roychowdhury(rychwdh2)
Shyam Rajendran(srajend2)
Abhinav Sharma(sharma55)
Ajay Nair(anair10)

Module initialization

In module init we first make a directory "mp1" directory and a file under this heirarchy "status" using `proc_mkdir` and `proc_create`. "status" is given 0666 permssion. Then a timer is initialized using `setup_timer` function, and a callback handler is registered for this timer. (`_periodic_timer_callback()`). This will be the top half interrupt handler which needs to be short and cannot sleep or block. The timer expiry period is set up using `mod_timer` call. Its set to 5 seconds.

Datastructures

To maintain a list of process, a linkedlist is create to store the process meta data viz process id, cpu time, and a pointer to nodes of same type. The process meta data is stored in a custom struct of type "`_process_meta_node`".

The head of the linked list (`plist_head`) is initialized using `LIST_HEAD` macro. This linkedlist will be modified and read from asynchronously. So we define a mutex to achieve a serialized access to the linked list. The mutex is initialized using `DEFINE_MUTEX` macro.

File operations

We expose read write capability to the mp1/status file. We register the handlers using a `file_operations` struct. Whenever user does a cat in the file, read handler "`status_read()`" gets called. When a pid is echod to the mp1/status file, "`status_write`" gets called. These hanlers can asynchronous access the linked list.

Timer interrupt handler

When the timer expires, the registered handler `_periodic_timer_callback` gets called. Since this is a top half handler we cannot acquire a mutex lock on the linkedlist of process meta. So this function initializes a work using `INIT_WORK` macro and queues it to the workqueue. The timer is the reset again to next 5 seconds using `mod_timer` call.

Bottom half

The bottom half is implemented in "`queue_work_bottom_half()`" function. In this function we first acquire lock to access the linkedlist. The helper function "`get_cpu_use()`" as provided in `mp1_given.h`, is used to get CPU time for the process id. If the function returns -1, then it means the process has ended and so this metadata node for this process is deleted from the linked list.

File operations

Process Registration

User space process can register with kernel driver by writing their process id to the mp1/status file. copy_from_user() is used to copy contents from user buffer to a kernel buffer. kstrtoul() is used to convert to an integer. If this call succeeds success status is returned to so user process can know it has successfully registered. If process is already registered error code is returned.

Status Read

When user performs a cat command on the "status" file, "status_read()" gets called. In this function a pointer to the user buffer is provided. Lock is acquired on the process meta data list and traversed. The process id and cpu tick time is appended to a buffer of type char. This content of this kernel buffer is then copied to the user buffer using copy_to_user(). status_read() handler will be called multiple times so the offset pointer is updated. In subsequent calls the offset is checked, and if offset is non zero the handler returns 0.

Details of how to run your program

1. make clean
2. make
3. sudo insmod mp1.ko
4. ./userapp &
5. cat /proc/mp1/status
6. sudo rmmod mp1

ScreenShot with two userapp instances running

```
cs423@cs423-vm:~/OS/MP1/OS_MPS/MP1_sol$ cat /proc/mp1/status
30791: 10244000000
30787: 11308000000
```

Log

```
cs423@cs423-vm:~/OS/MP1/OS_MPS/MP1_sol$ make
rm -f userapp *~ *.ko *.o *.mod.c Module.symvers modules.order
make -C /lib/modules/3.13.0-44-generic/build
M=/home/cs423/OS/MP1/OS_MPS/MP1_sol modules
make[1]: Entering directory
`/usr/src/linux-headers-3.13.0-44-generic'
CC [M] /home/cs423/OS/MP1/OS_MPS/MP1_sol/mp1.o
/home/cs423/OS/MP1/OS_MPS/MP1_sol/mp1.c: In function
`update_process_cpu_time':
/home/cs423/OS/MP1/OS_MPS/MP1_sol/mp1.c:50:5: warning: ISO C90
forbids mixed declarations and code [-Wdeclaration-after-statement]
    unsigned long cputime = 0;
```

```

^
/home/cs423/OS/MP1/OS_MPS/MP1_sol/mp1.c: In function 'status_read':
/home/cs423/OS/MP1/OS_MPS/MP1_sol/mp1.c:143:5: warning: ISO C90
forbids mixed declarations and code [-Wdeclaration-after-statement]
    char *temp=(char*)kmalloc(PAGE_SIZE*sizeof(char), GFP_KERNEL);
^

/home/cs423/OS/MP1/OS_MPS/MP1_sol/mp1.c:148:5: warning: ISO C90
forbids mixed declarations and code [-Wdeclaration-after-statement]
    int size = get_process_info_list_as_string(temp);
^

/home/cs423/OS/MP1/OS_MPS/MP1_sol/mp1.c:139:16: warning: unused
variable 'finished' [-Wunused-variable]
    static int finished = 0;
            ^

/home/cs423/OS/MP1/OS_MPS/MP1_sol/mp1.c: In function 'status_write':
/home/cs423/OS/MP1/OS_MPS/MP1_sol/mp1.c:175:5: warning: ISO C90
forbids mixed declarations and code [-Wdeclaration-after-statement]
    char* input=(char*)kmalloc((count+1) * sizeof(char),
GFP_KERNEL);
^

/home/cs423/OS/MP1/OS_MPS/MP1_sol/mp1.c:186:5: warning: ISO C90
forbids mixed declarations and code [-Wdeclaration-after-statement]
    long int pid = 0;
^

Building modules, stage 2.
MODPOST 1 modules
CC      /home/cs423/OS/MP1/OS_MPS/MP1_sol/mp1.mod.o
LD [M]  /home/cs423/OS/MP1/OS_MPS/MP1_sol/mp1.ko
make[1]: Leaving directory `/usr/src/linux-headers-3.13.0-44-generic'
gcc -o userapp userapp.c
cs423@cs423-vm:~/OS/MP1/OS_MPS/MP1_sol$ lsmod | grep mp1
cs423@cs423-vm:~/OS/MP1/OS_MPS/MP1_sol$ sudo insmod mp1.ko
[sudo] password for cs423:
cs423@cs423-vm:~/OS/MP1/OS_MPS/MP1_sol$ lsmod | grep mp1
mp1                13108  0
cs423@cs423-vm:~/OS/MP1/OS_MPS/MP1_sol$ ls /proc/mp1/status
/proc/mp1/status
cs423@cs423-vm:~/OS/MP1/OS_MPS/MP1_sol$ cat /proc/mp1/status
cs423@cs423-vm:~/OS/MP1/OS_MPS/MP1_sol$ ./userapp &
cs423@cs423-vm:~/OS/MP1/OS_MPS/MP1_sol$ cat /proc/mp1/status
30480: 0
cs423@cs423-vm:~/OS/MP1/OS_MPS/MP1_sol$ cat /proc/mp1/status
30480: 4208000000

```

```

cs423@cs423-vm:~/OS/MP1/OS_MPS/MP1_sol$ cat /proc/mp1/status
30480: 4208000000
cs423@cs423-vm:~/OS/MP1/OS_MPS/MP1_sol$ cat /proc/mp1/status
30480: 4208000000
cs423@cs423-vm:~/OS/MP1/OS_MPS/MP1_sol$ cat /proc/mp1/status
30480: 4208000000
cs423@cs423-vm:~/OS/MP1/OS_MPS/MP1_sol$ dmesg | tail
[699547.555182] bottom half calledbottom half called
[699557.580766] bottom half calledbottom half called
[699567.606262] bottom half calledbottom half called
[699577.631795] bottom half calledregister_process::pid 30476 added
[699579.923198] pid 30476 registeredbottom half called
[699582.644571]          remove          pid30476          from          reg          process
listregister_process::pid 30480 added
[699583.439983] pid 30480 registeredbottom half called
[699587.657297] cpu time for 30480 is 4208000000
[699592.670061] bottom half called
[699592.670067] cpu time for 30480 is 9212000000
cs423@cs423-vm:~/OS/MP1/OS_MPS/MP1_sol$ cat /proc/mp1/status
30480: 29236000000
cs423@cs423-vm:~/OS/MP1/OS_MPS/MP1_sol$ cat /proc/mp1/status
30480: 29236000000
cs423@cs423-vm:~/OS/MP1/OS_MPS/MP1_sol$ cat /proc/mp1/status
30480: 34244000000
cs423@cs423-vm:~/OS/MP1/OS_MPS/MP1_sol$ fg
./userapp
^C
cs423@cs423-vm:~/OS/MP1/OS_MPS/MP1_sol$ cat /proc/mp1/status
30480: 39252000000
cs423@cs423-vm:~/OS/MP1/OS_MPS/MP1_sol$ cat /proc/mp1/status
30480: 39252000000
cs423@cs423-vm:~/OS/MP1/OS_MPS/MP1_sol$ cat /proc/mp1/status
cs423@cs423-vm:~/OS/MP1/OS_MPS/MP1_sol$ cat /proc/mp1/status
cs423@cs423-vm:~/OS/MP1/OS_MPS/MP1_sol$ sudo rmmod mp1
cs423@cs423-vm:~/OS/MP1/OS_MPS/MP1_sol$ dmesg | tail
[699607.708375] bottom half called
[699607.708381] cpu time for 30480 is 24228000000
[699612.721146] bottom half called
[699612.721152] cpu time for 30480 is 29236000000
[699617.733921] bottom half called
[699617.733927] cpu time for 30480 is 34244000000
[699622.746680] bottom half called
[699622.746686] cpu time for 30480 is 39252000000

```

```
[699627.759528] bottom half called
[699627.759533]         remove      pid30480      from      reg      process
list<6>[699634.910528] Goodbye, world 2
```