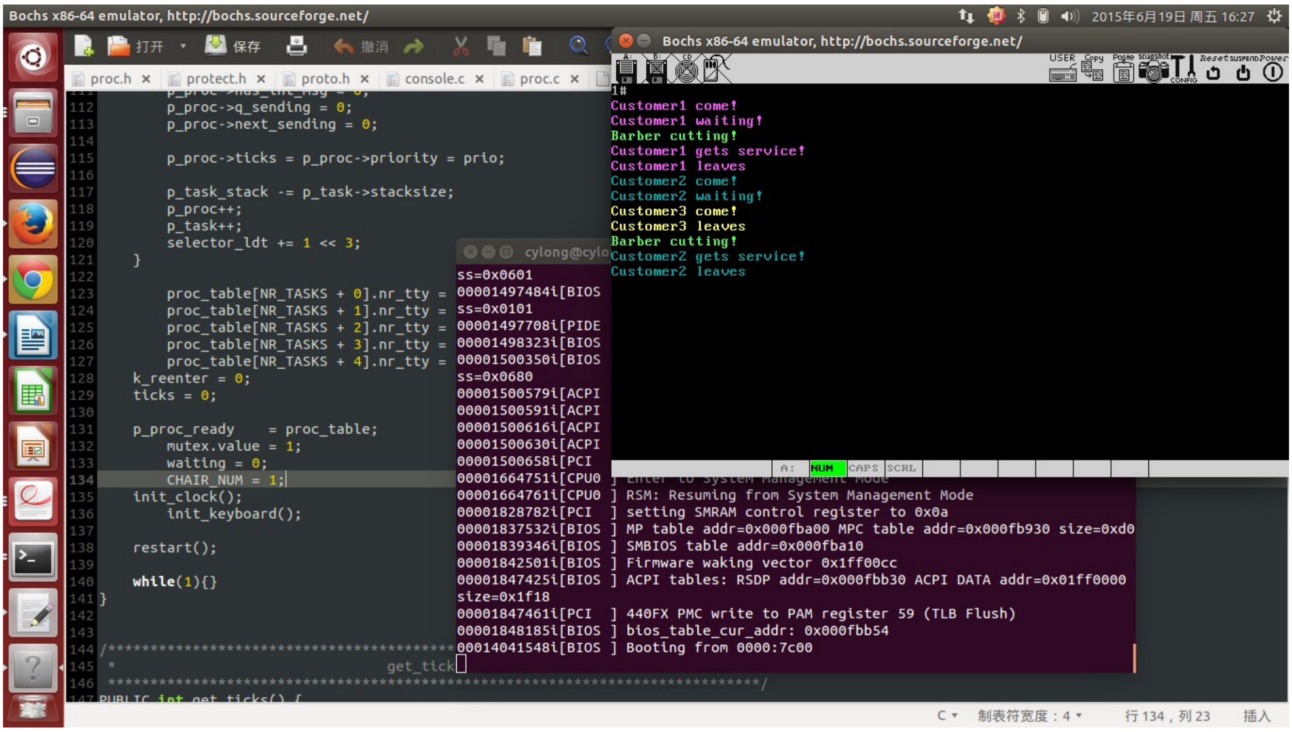
实验四报告

131250181-陈云龙

本代码在 orange’s 的第八章的代码基础上修改，添加了实验要求的 sys_process_sleep，sys_sem_p，sys_tem_v

1. 一个椅子截图

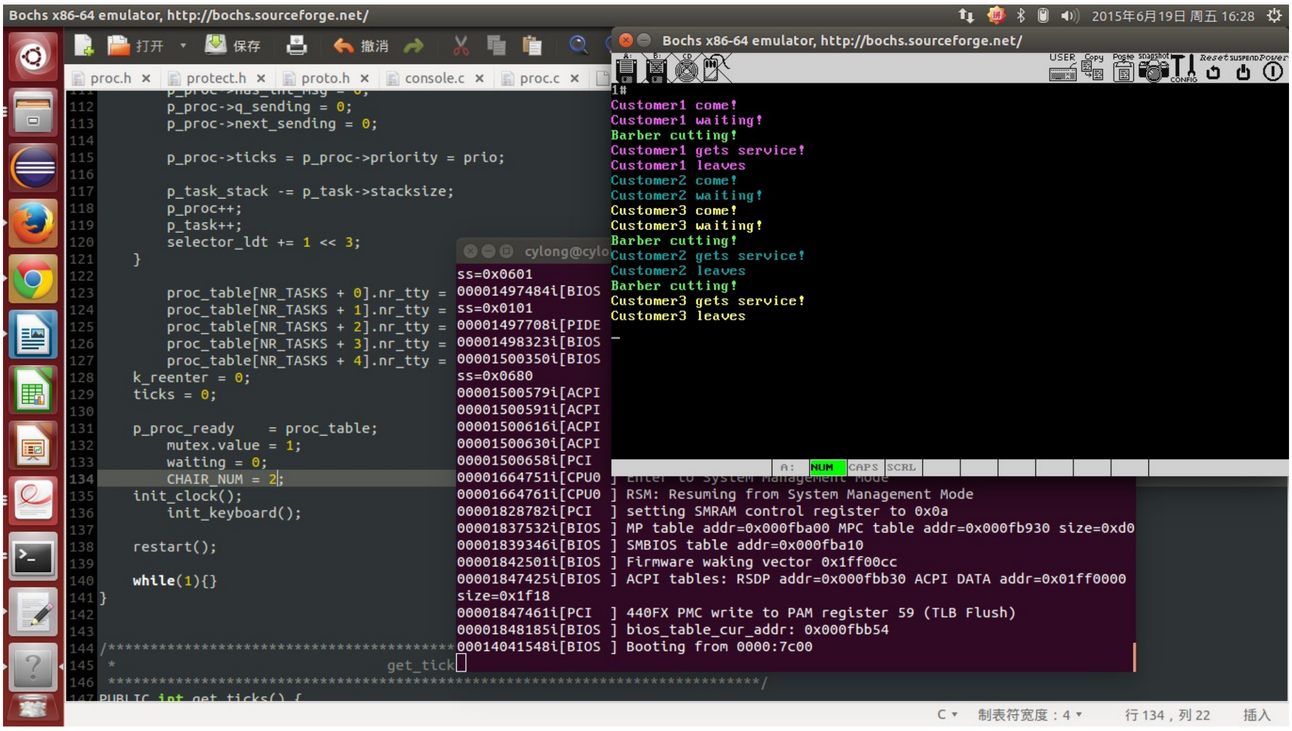


```
        p_proc->has_int_msg = 0;
        p_proc->q_sending = 0;
        p_proc->next_sending = 0;

        p_proc->ticks = p_proc->priority = prio;

        p_task_stack -= p_task->stacksize;
        p_proc++;
        p_task++;
        selector_ldt += 1 << 3;
    }

        proc_table[NR_TASKS + 0].nr_tty =
        proc_table[NR_TASKS + 1].nr_tty =
        proc_table[NR_TASKS + 2].nr_tty =
        proc_table[NR_TASKS + 3].nr_tty =
        proc_table[NR_TASKS + 4].nr_tty =
    k_reenter = 0;
    ticks = 0;

    p_proc_ready     = proc_table;
        mutex.value = 1;
        waiting = 0;
        CHAIR_NUM = 1;
    init_clock();
        init_keyboard();

    restart();

    while(1){}
}

/*****************************************************************
 *                               get_tick
 *****************************************************************/
PUBLIC int get_ticks() {
```

```
1#
Customer1 come!
Customer1 waiting!
Barber cutting!
Customer1 gets service!
Customer1 leaves
Customer2 come!
Customer2 waiting!
Customer3 come!
Customer3 leaves
Barber cutting!
Customer2 gets service!
Customer2 leaves
```

```
ss=0x0601
00001497484i[BIOS
ss=0x0101
00001497708i[PIDE
00001498323i[BIOS
00001500350i[BIOS
ss=0x0680
00001500579i[ACPI
00001500591i[ACPI
00001500616i[ACPI
00001500630i[ACPI
00001500658i[PCI
00001664751i[CPU0 ] Enter to System Management Mode
00001664761i[CPU0 ] RSM: Resuming from System Management Mode
00001828782i[PCI  ] setting SMRAM control register to 0x0a
00001837532i[BIOS ] MP table addr=0x000fba00 MPC table addr=0x000fb930 size=0xd0
00001839346i[BIOS ] SMBIOS table addr=0x000fba10
00001842501i[BIOS ] Firmware waking vector 0x1ff00cc
00001847425i[BIOS ] ACPI tables: RSDP addr=0x000fbb30 ACPI DATA addr=0x01ff0000
size=0x1f18
00001847461i[PCI  ] 440FX PMC write to PAM register 59 (TLB Flush)
00001848185i[BIOS ] bios_table_cur_addr: 0x000fbb54
00014041548i[BIOS ] Booting from 0000:7c00
```

2. 两个椅子的截图

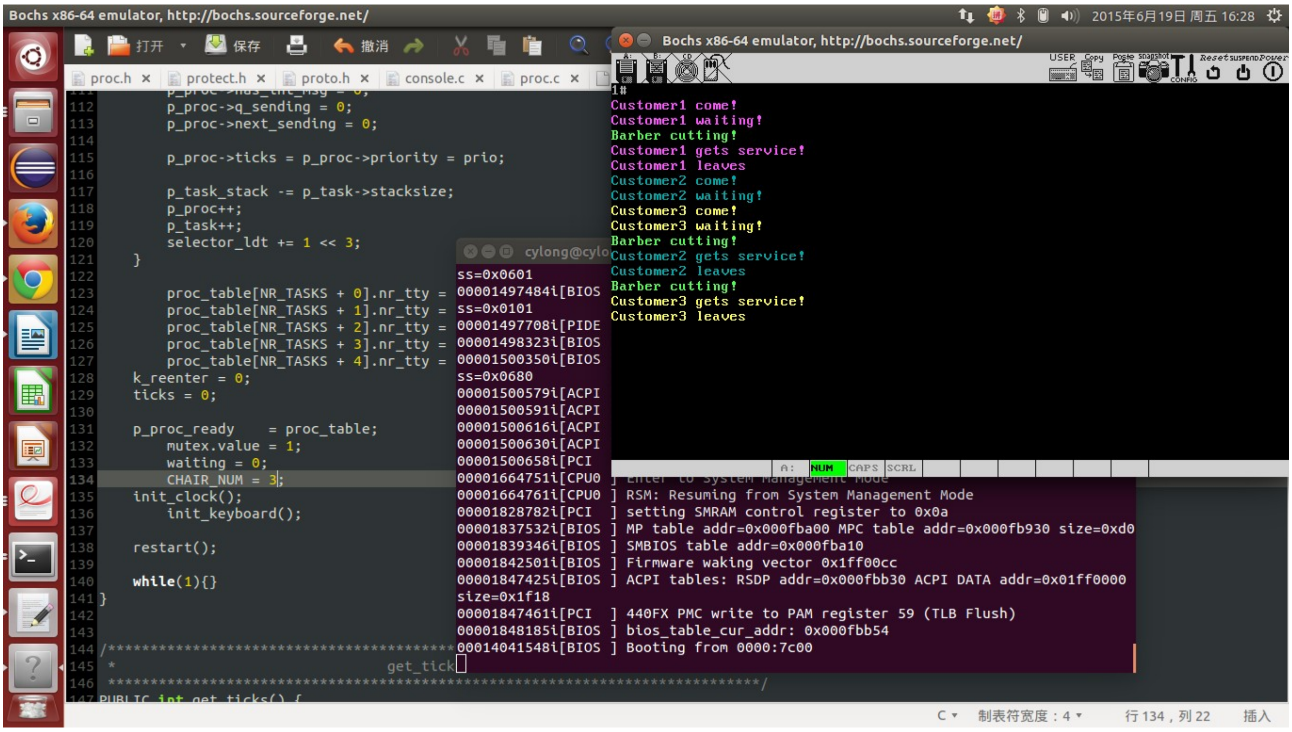


```
        p_proc->has_int_msg = 0;
        p_proc->q_sending = 0;
        p_proc->next_sending = 0;

        p_proc->ticks = p_proc->priority = prio;

        p_task_stack -= p_task->stacksize;
        p_proc++;
        p_task++;
        selector_ldt += 1 << 3;
    }

        proc_table[NR_TASKS + 0].nr_tty =
        proc_table[NR_TASKS + 1].nr_tty =
        proc_table[NR_TASKS + 2].nr_tty =
        proc_table[NR_TASKS + 3].nr_tty =
        proc_table[NR_TASKS + 4].nr_tty =
    k_reenter = 0;
    ticks = 0;

    p_proc_ready     = proc_table;
        mutex.value = 1;
        waiting = 0;
        CHAIR_NUM = 2;
    init_clock();
        init_keyboard();

    restart();

    while(1){}
}

/*****************************************************************
 *                               get_tick
 *****************************************************************/
PUBLIC int get_ticks() {
```

```
1#
Customer1 come!
Customer1 waiting!
Barber cutting!
Customer1 gets service!
Customer1 leaves
Customer2 come!
Customer2 waiting!
Customer3 come!
Customer3 waiting!
Barber cutting!
Customer2 gets service!
Customer2 leaves
Barber cutting!
Customer3 gets service!
Customer3 leaves
_
```

```
ss=0x0601
00001497484i[BIOS
ss=0x0101
00001497708i[PIDE
00001498323i[BIOS
00001500350i[BIOS
ss=0x0680
00001500579i[ACPI
00001500591i[ACPI
00001500616i[ACPI
00001500630i[ACPI
00001500658i[PCI
00001664751i[CPU0 ] Enter to System Management Mode
00001664761i[CPU0 ] RSM: Resuming from System Management Mode
00001828782i[PCI  ] setting SMRAM control register to 0x0a
00001837532i[BIOS ] MP table addr=0x000fba00 MPC table addr=0x000fb930 size=0xd0
00001839346i[BIOS ] SMBIOS table addr=0x000fba10
00001842501i[BIOS ] Firmware waking vector 0x1ff00cc
00001847425i[BIOS ] ACPI tables: RSDP addr=0x000fbb30 ACPI DATA addr=0x01ff0000
size=0x1f18
00001847461i[PCI  ] 440FX PMC write to PAM register 59 (TLB Flush)
00001848185i[BIOS ] bios_table_cur_addr: 0x000fbb54
00014041548i[BIOS ] Booting from 0000:7c00
```

3. 三个椅子的截图



```c
        p_proc->q_sending = 0;
        p_proc->next_sending = 0;

        p_proc->ticks = p_proc->priority = prio;

        p_task_stack -= p_task->stacksize;
        p_proc++;
        p_task++;
        selector_ldt += 1 << 3;
    }

        proc_table[NR_TASKS + 0].nr_tty =
        proc_table[NR_TASKS + 1].nr_tty =
        proc_table[NR_TASKS + 2].nr_tty =
        proc_table[NR_TASKS + 3].nr_tty =
        proc_table[NR_TASKS + 4].nr_tty =
    k_reenter = 0;
    ticks = 0;

    p_proc_ready    = proc_table;
        mutex.value = 1;
        waiting = 0;
        CHAIR_NUM = 3;
    init_clock();
        init_keyboard();

    restart();

    while(1){}
}

/****************************************
 *                              get_tick
 *************************************************************************/
PUBLIC int get_ticks() {
```

```
ss=0x0601
00001497484i[BIOS
ss=0x0101
00001497708i[PIDE
00001498323i[BIOS
00001500350i[BIOS
ss=0x0680
00001500579i[ACPI
00001500591i[ACPI
00001500616i[ACPI
00001500630i[ACPI
00001500658i[PCI
00001664751i[CPU0 ] Enter to System Management Mode
00001664761i[CPU0 ] RSM: Resuming from System Management Mode
00001828782i[PCI  ] setting SMRAM control register to 0x0a
00001837532i[BIOS ] MP table addr=0x000fba00 MPC table addr=0x000fb930 size=0xd0
00001839346i[BIOS ] SMBIOS table addr=0x000fba10
00001842501i[BIOS ] Firmware waking vector 0x1ff00cc
00001847425i[BIOS ] ACPI tables: RSDP addr=0x000fbb30 ACPI DATA addr=0x01ff0000
size=0x1f18
00001847461i[PCI  ] 440FX PMC write to PAM register 59 (TLB Flush)
00001848185i[BIOS ] bios_table_cur_addr: 0x000fbb54
00014041548i[BIOS ] Booting from 0000:7c00
```

```
1#
Customer1 come!
Customer1 waiting!
Barber cutting!
Customer1 gets service!
Customer1 leaves
Customer2 come!
Customer2 waiting!
Customer3 come!
Customer3 waiting!
Barber cutting!
Customer2 gets service!
Customer2 leaves
Barber cutting!
Customer3 gets service!
Customer3 leaves
```

4. 新增 sys_process_sleep 所在文件
1) kernel/proc.c:PUBLIC void sys_process_sleep(int unused1,int unused2,int milli_sec,struct proc * p)
2) kernel/global.c:PUBLIC   system_call sys_call_table[NR_SYS_CALL] = {sys_printx, sys_sendrec,sys_process_sleep,sys_sem_p,sys_sem_v};
3) include/proto.h:PUBLIC void sys_process_sleep(int milli_sec,struct proc * p);

5. 新增 sys_sem_p;sys_sem_v 所在文件
1) kernel/main.c:PUBLIC void sys_sem_p(int unused1,int unused2,struct semaphore * s,struct proc * p)
2) PUBLIC   system_call sys_call_table[NR_SYS_CALL] = {sys_printx, sys_sendrec,sys_process_sleep,sys_sem_p,sys_sem_v};
3) include/proto.h:PUBLIC void sys_sem_p(int unused1,int unused2,struct semaphore * s,struct proc * p)
sys_sem_p 与 sys_sem_v 主要算法和思想参照课本中 3.3 信号量和 PV 操作

6. 在 kernel/main.c 中 CHAIR_NUM 是椅子数量，在 kernel/global.c:PUBLIC int CHAIR_NUM = 3;中定义

7. 在 kernel/main.c 中 TestB 是理发师，TestC、TestD、TestE 是顾客
TestD、TestE 添加顺序：参考 ORANGE’S 第 6 章 6.4.6(207 页)
a.在 task_table 中增加一项(global.c).
b.让 NR_TASK 加 1(proc.h).
c.定义任务堆
栈(proc.h)
d.修改 STACK_SIZE_TOTAL(proc.h).
e.添加新任务执行体的函数声明(proc.h)

8. 不同进程变色: 在 kernel/console.c 中 out_char 函数
sys_printx 会调用 out_char 输出在 console.c 中添加

```
char ch_color = DEFAULT_CHAR_COLOR;
    switch(p_proc_ready->pid) {
            case 3:
                    ch_color = 0x0A;
                    break;
            case 4:
                    ch_color = 0X0D;
                    break;
            case 5:
                    ch_color = 0X03;
                    break;
            case 6:
                    ch_color = 0X0E;
                    break;
        }
```

不同的进程改变其输出颜色

9. 此为 TTY 代码基础上改的,理发师问题的输出在 2 号窗口上,在 kernel/tty.c 设置 select_console(1)默认显示第二个窗口