Assignment 005: Lab 5：Linux系统调用

葛现隆 3120102146

一、 实验目的

1 学习Linux内核的配置和编译；

2 深入理解Linux系统调用；

3 理解ARM和x86的CPU模式（系统模式、用户模式等）的不同。

二、 实验器材

硬件

- 树莓派板一块；
- 5V/1A电源一个；
- microUSB线一根；

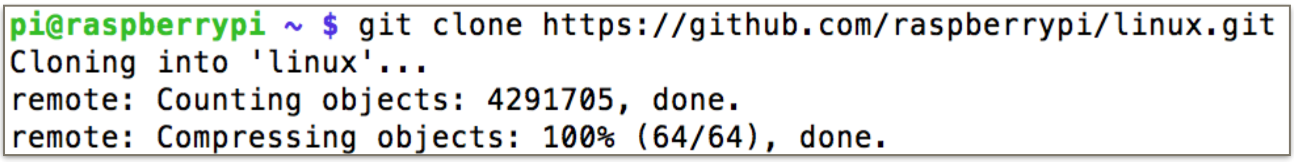
三、 实验步骤

1 下载raspberry pi的源代码

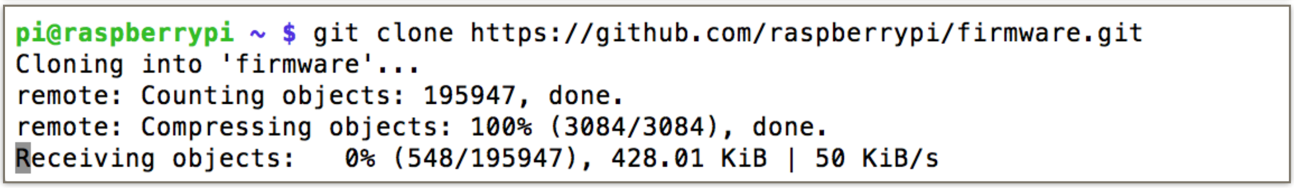
通过 $git clone https://github.com/raspberrypi/linux.git

$git clone https://github.com/raspberrypi/firmware.git

命令获得raspberry的内核代码，耗时较长；



```
pi@raspberrypi ~ $ git clone https://github.com/raspberrypi/linux.git
Cloning into 'linux'...
remote: Counting objects: 4291705, done.
remote: Compressing objects: 100% (64/64), done.
```

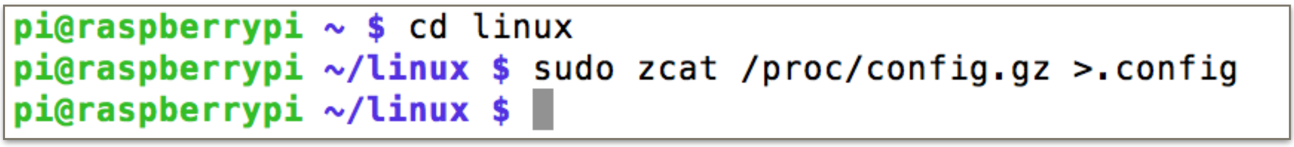


```
pi@raspberrypi ~ $ git clone https://github.com/raspberrypi/firmware.git
Cloning into 'firmware'...
remote: Counting objects: 195947, done.
remote: Compressing objects: 100% (3084/3084), done.
Receiving objects:   0% (548/195947), 428.01 KiB | 50 KiB/s
```

2 提取原有内核配置文件

$cd linux

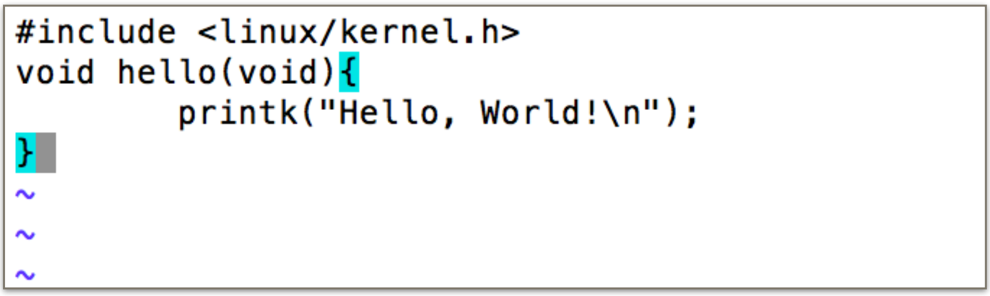
$sudo zcat /proc/config.gz > .config



```
pi@raspberrypi ~ $ cd linux
pi@raspberrypi ~/linux $ sudo zcat /proc/config.gz >.config
pi@raspberrypi ~/linux $ 
```

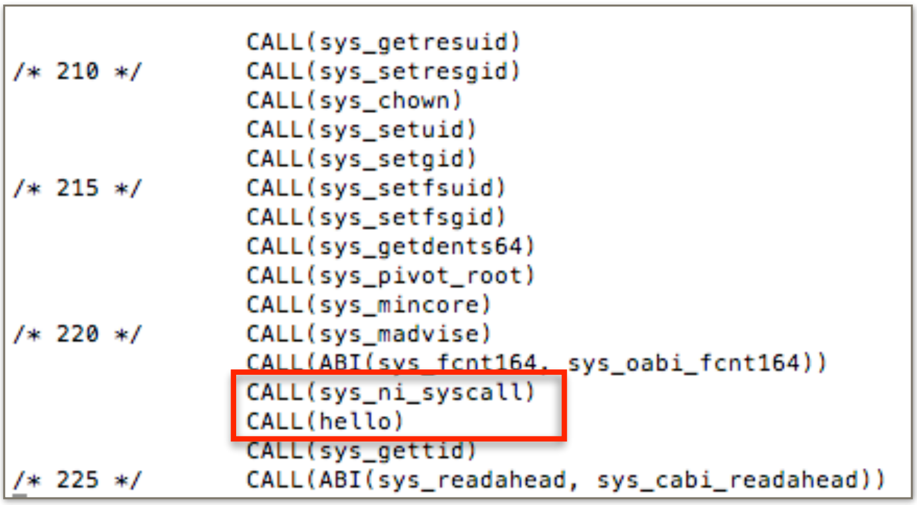
3 建立系统调用文件

在arch/arm/kernel目录下创建mysyscall.c文件，内容如下：



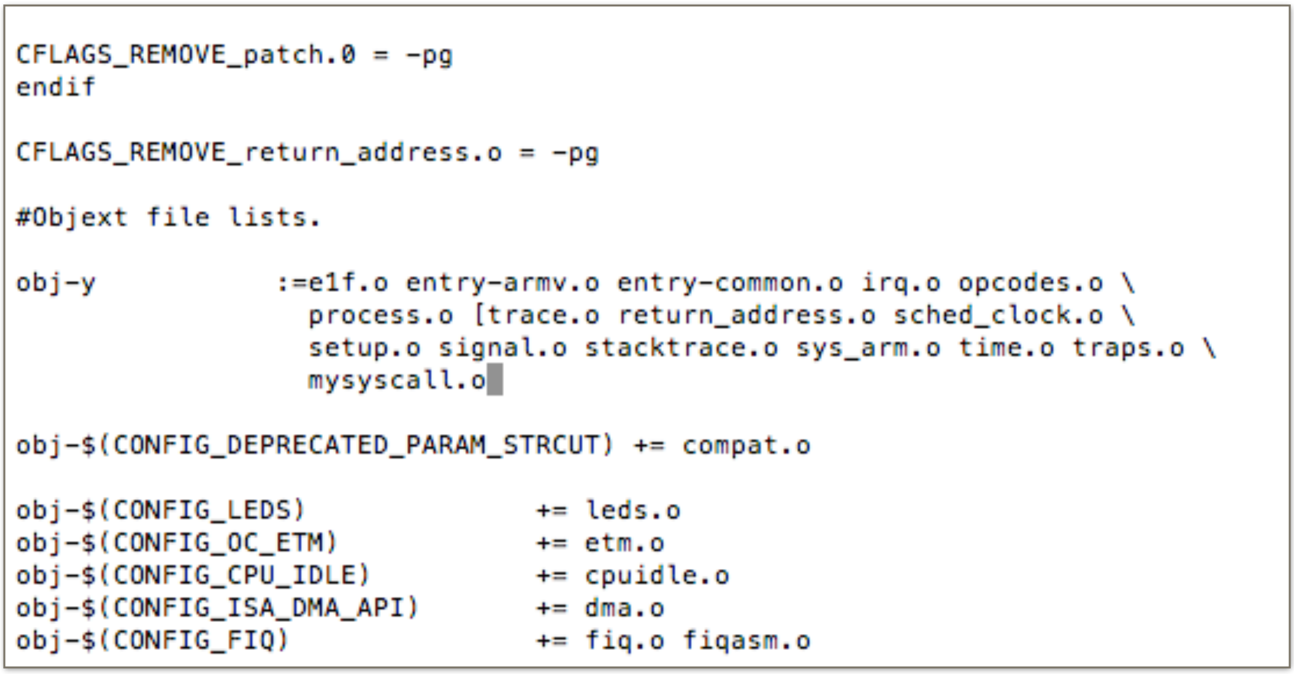
```
#include <linux/kernel.h>
void hello(void){
        printk("Hello, World!\n");
}
~
~
~
```

4 增加系统调用
在223的位置，修改系统调用，新增CALL(hello)；



```
                CALL(sys_getresuid)
/* 210 */       CALL(sys_setresgid)
                CALL(sys_chown)
                CALL(sys_setuid)
                CALL(sys_setgid)
/* 215 */       CALL(sys_setfsuid)
                CALL(sys_setfsgid)
                CALL(sys_getdents64)
                CALL(sys_pivot_root)
                CALL(sys_mincore)
/* 220 */       CALL(sys_madvise)
                CALL(ABI(sys_fcntl64, sys_oabi_fcntl64))
                CALL(sys_ni_syscall)
                CALL(hello)
                CALL(sys_gettid)
/* 225 */       CALL(ABI(sys_readahead, sys_cabi_readahead))
```

5 修改makefile文件
在 obj-y 后面的加上 mysyscall.o ；



```

CFLAGS_REMOVE_patch.0 = -pg
endif

CFLAGS_REMOVE_return_address.o = -pg

#Objext file lists.

obj-y           :=e1f.o entry-armv.o entry-common.o irq.o opcodes.o \
                  process.o [trace.o return_address.o sched_clock.o \
                  setup.o signal.o stacktrace.o sys_arm.o time.o traps.o \
                  mysyscall.o

obj-$(CONFIG_DEPRECATED_PARAM_STRCUT) += compat.o

obj-$(CONFIG_LEDS)              += leds.o
obj-$(CONFIG_OC_ETM)            += etm.o
obj-$(CONFIG_CPU_IDLE)          += cpuidle.o
obj-$(CONFIG_ISA_DMA_API)       += dma.o
obj-$(CONFIG_FIQ)               += fiq.o fiqasm.o
```

6 使用已有配置配置内核

$ make oldconfig

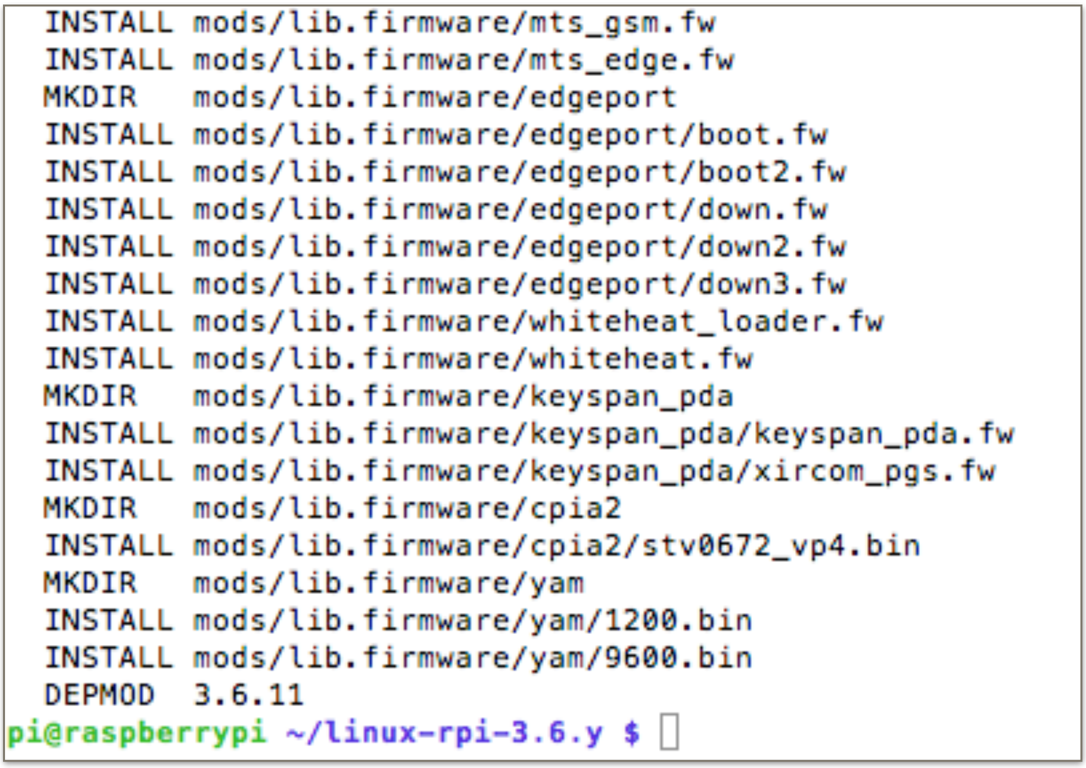
7 编译内核

$ make

8 模块淡妆

$ mkdir mods

$ make moules_install MODULES_INSTALL_PATH=mods



```
  INSTALL mods/lib.firmware/mts_gsm.fw
  INSTALL mods/lib.firmware/mts_edge.fw
  MKDIR   mods/lib.firmware/edgeport
  INSTALL mods/lib.firmware/edgeport/boot.fw
  INSTALL mods/lib.firmware/edgeport/boot2.fw
  INSTALL mods/lib.firmware/edgeport/down.fw
  INSTALL mods/lib.firmware/edgeport/down2.fw
  INSTALL mods/lib.firmware/edgeport/down3.fw
  INSTALL mods/lib.firmware/whiteheat_loader.fw
  INSTALL mods/lib.firmware/whiteheat.fw
  MKDIR   mods/lib.firmware/keyspan_pda
  INSTALL mods/lib.firmware/keyspan_pda/keyspan_pda.fw
  INSTALL mods/lib.firmware/keyspan_pda/xircom_pgs.fw
  MKDIR   mods/lib.firmware/cpia2
  INSTALL mods/lib.firmware/cpia2/stv0672_vp4.bin
  MKDIR   mods/lib.firmware/yam
  INSTALL mods/lib.firmware/yam/1200.bin
  INSTALL mods/lib.firmware/yam/9600.bin
  DEPMOD  3.6.11
pi@raspberrypi ~/linux-rpi-3.6.y $ 
```

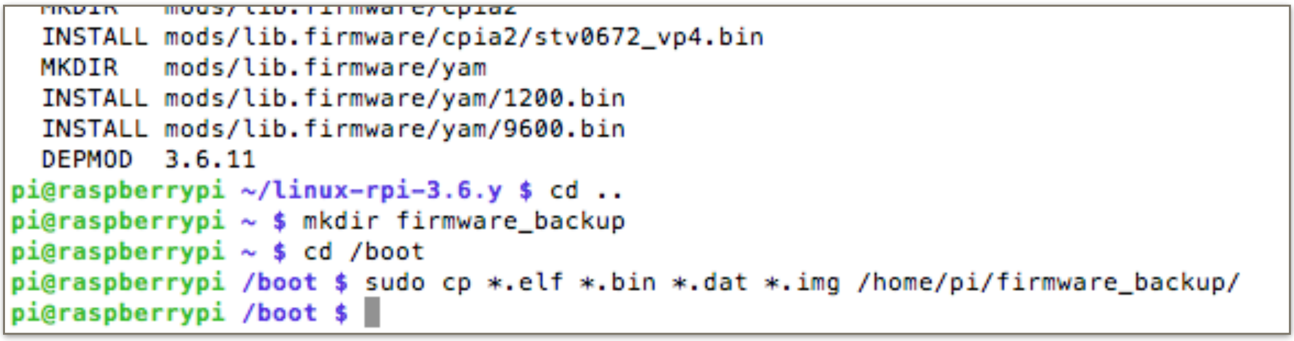
9 备份已有内核和固件

$ cd ..

$ mkdir firmware_backup

$ cd /boot

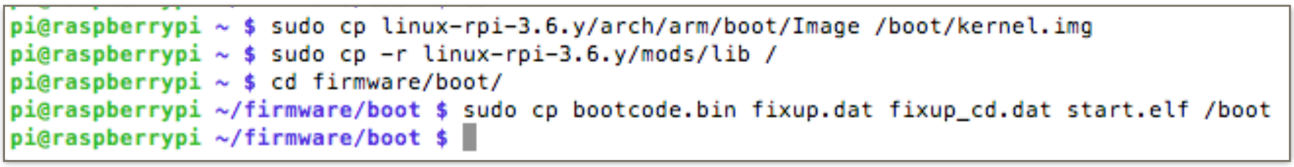
$ cp *.elf *.bin *.img *.dat /home/pi/firmware_backup



```
  MKDIR   mods/lib.firmware/cpia2
  INSTALL mods/lib.firmware/cpia2/stv0672_vp4.bin
  MKDIR   mods/lib.firmware/yam
  INSTALL mods/lib.firmware/yam/1200.bin
  INSTALL mods/lib.firmware/yam/9600.bin
  DEPMOD  3.6.11
pi@raspberrypi ~/linux-rpi-3.6.y $ cd ..
pi@raspberrypi ~ $ mkdir firmware_backup
pi@raspberrypi ~ $ cd /boot
pi@raspberrypi /boot $ sudo cp *.elf *.bin *.dat *.img /home/pi/firmware_backup/
pi@raspberrypi /boot $ 
```

10 更新内核和固件

$ sudo cp linux-rpi-3.6.y/arch/arm/boot/Image /boot/kernel.img
$ sudo cp –r linux-rpi-3.6.y/mods/lib /
$ cd firmware/boot
$ sudo cp bootcode.bin fixup.dat fixup_cd.dat start.elf /boot

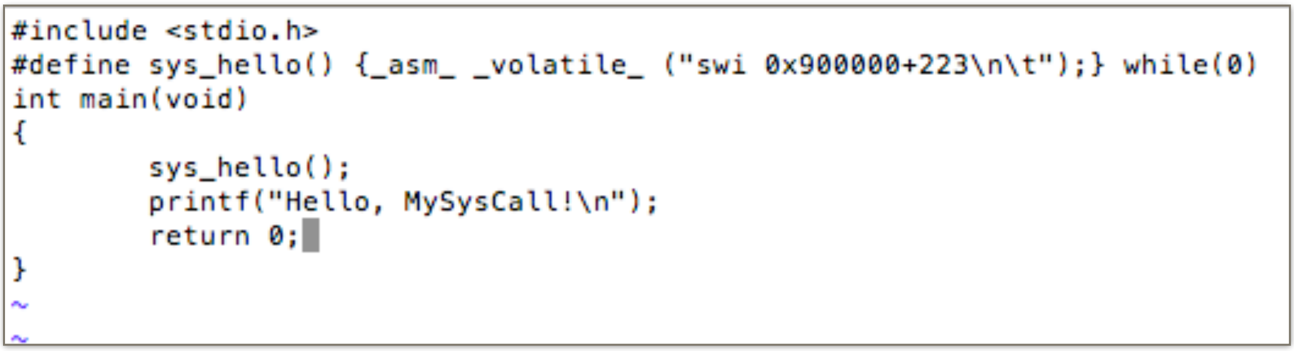


```
pi@raspberrypi ~ $ sudo cp linux-rpi-3.6.y/arch/arm/boot/Image /boot/kernel.img
pi@raspberrypi ~ $ sudo cp -r linux-rpi-3.6.y/mods/lib /
pi@raspberrypi ~ $ cd firmware/boot/
pi@raspberrypi ~/firmware/boot $ sudo cp bootcode.bin fixup.dat fixup_cd.dat start.elf /boot
pi@raspberrypi ~/firmware/boot $ 
```

11 重启

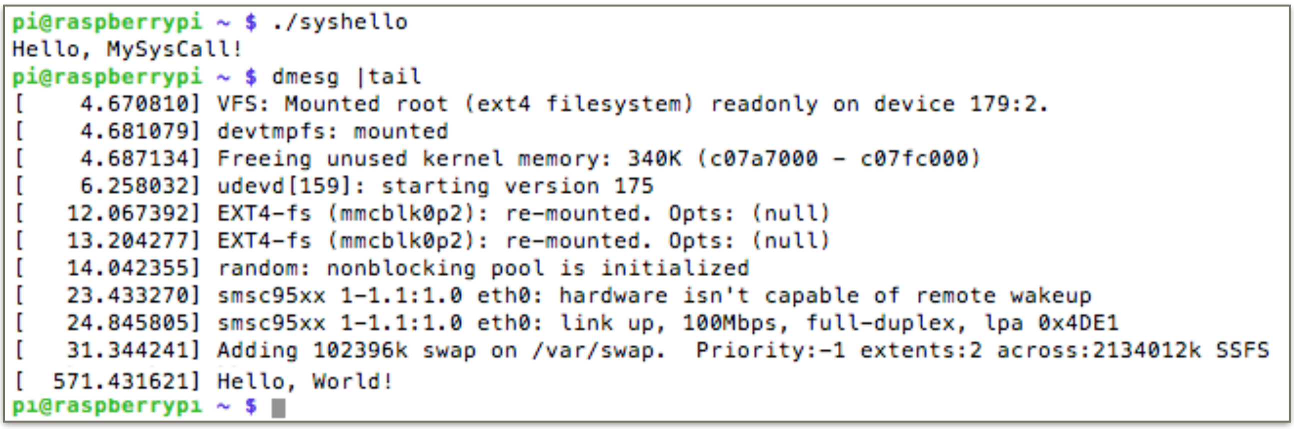
$sudo reboot

12 编写程序测试系统调用，内容如下



```
#include <stdio.h>
#define sys_hello() {_asm_ _volatile_ ("swi 0x900000+223\n\t");} while(0)
int main(void)
{
        sys_hello();
        printf("Hello, MySysCall!\n");
        return 0;
}
~
~
```

13 编译运行，在最后一行获得系统调用结果”Hello World!”



```
pi@raspberrypi ~ $ ./syshello
Hello, MySysCall!
pi@raspberrypi ~ $ dmesg |tail
[    4.670810] VFS: Mounted root (ext4 filesystem) readonly on device 179:2.
[    4.681079] devtmpfs: mounted
[    4.687134] Freeing unused kernel memory: 340K (c07a7000 - c07fc000)
[    6.258032] udevd[159]: starting version 175
[   12.067392] EXT4-fs (mmcblk0p2): re-mounted. Opts: (null)
[   13.204277] EXT4-fs (mmcblk0p2): re-mounted. Opts: (null)
[   14.042355] random: nonblocking pool is initialized
[   23.433270] smsc95xx 1-1.1:1.0 eth0: hardware isn't capable of remote wakeup
[   24.845805] smsc95xx 1-1.1:1.0 eth0: link up, 100Mbps, full-duplex, lpa 0x4DE1
[   31.344241] Adding 102396k swap on /var/swap.  Priority:-1 extents:2 across:2134012k SSFS
[  571.431621] Hello, World!
pi@raspberrypi ~ $ 
```