

操作系统

第四次上机课作业实验报告

一， 实验目的

再简化的 JOS 系统上增加 JOS 的 Shell，主要是把 JOS 的有关 Shell 的文件放进项目中，然后修改 Makefile 文件和 init.c 文件首先编译链接这些新加入的文件，之后调用合适的函数使 Shell 能够正常运行。

二， 实验步骤以及结果

1， 在 JOS 系统中，console.c monitor.c 这两个文件中主要是写有关 shell 的。所以只要把这两个文件引用其他文件的闭包包含进这个项目中，就可以构建项目，然后完成目的。经过查看 include 等，发现必须包括 JOS 系统里面的 inc 文件夹中的几乎所有文件，lib 文件夹的所有文件，kernel 文件夹中的 console,monitor,kdebug 这三组文件

2， 之后修改 Makefile 文件，是新加入的这些.c 文件能够编译成.o 文件，然后链接生成可执行程序。这里 Makefile 我是这样修改的
增加的部分：

```
lib/%.o:lib/%.c
    $(CC) $(CFLAGS) -c -o $@ $<

kern/*.o:kern/*.c
    $(CC) $(CFLAGS) -c -o $@ $<
```

这两句是遇到 kernel 以及 lib 文件夹中的.c 文件编译生成.o 文件。

修改的部分：

```
kern/kernel:kern/entry.o kern/entrypgdir.o kern/init.o kern/console.o kern/monitor.o kern/kdebug.o kern/printf.o lib/readline.o lib/string.o lib/printfmt.o $(LD) -o kern/kernel -m elf_i386 -T kern/kernel.ld -nostdlib kern/entry.o kern/entrypgdir.o kern/init.o kern/console.o kern/monitor.o kern/kdebug.o kern/printf.o lib/readline.o lib/string.o lib/printfmt.o /usr/lib/gcc/i686-linux-gnu/5/libgcc.a -b binary
```

这两句是把新生成的.o 文件链接起来，生成可执行程序。(顺序是

调用者在前，被调用者在后)

3, 修改 init.c 文件，使 shell 能够正常运行。我修改的结果是以下部

分：

```
extern char edata[], end[];
memset(edata, 0, end - edata);

// Initialize the console.
// Can't call cprintf until after we do this!
cons_init();
// mem_init();
// Drop into the kernel monitor.
while (1)
    monitor(NULL);
```

其中前两句不加会乱码输出。

4, 结果如下

```
NO Bochs OBE Support available!

Bochs BIOS - build: 09/02/12
$Revision: 11318 $ $Date: 2012-08-06 19:59:54 +0200 (Mo, 06. Aug 2012) $
Options: apmbios pcibios pnpbios eltorito rombios32

ata0 master: Generic 1234 ATA-6 Hard-Disk ( 4 MBytes)

Press F12 for boot menu.

Booting from Hard Disk...
Welcome to the JOS kernel monitor!
Type 'help' for a list of commands.
K> help
help - Display this list of commands
kerninfo - Display information about the kernel
K> kerninfo
Special kernel symbols:
  _start      0010000c (phys)
  entry      f010000c (virt) 0010000c (phys)
  etext      f0101591 (virt) 00101591 (phys)
  edata      f0112320 (virt) 00112320 (phys)
  end        f0112960 (virt) 00112960 (phys)
Kernel executable memory footprint: 75KB
K> a_
```