

Lab1 实验报告

毛陈诚 PB20111694

一、实验目标

1. 了解multiboot协议与过程
2. 安装qemu, 了解qemu对multiboot的支持
3. 根据multiboot协议写出multiboot header
4. 在VGA显存输出
5. 利用uart串口输出

二、实验原理

- multiboot启动协议介绍

Multiboot是一种用于引导操作系统的规范, 它定义了如何在x86计算机上加载内核镜像。Multiboot规范由GNU组织制定, 其目的是为了使操作系统内核能够在多种不同的引导环境中运行。

- qemu / grub介绍, 在本实验的角色 vga 介绍/串口介绍

QEMU是一个开源的虚拟机监视器和仿真器, 它可以模拟多个硬件架构。VGA是一种显示设备, 它可以将计算机内存中的图像数据显示在屏幕上。串口是一种用于与计算机通信的接口。

三、源代码说明

- 例如 VGA中 `movl $0x2f652f68, 0xB8000` 是在干什么?
- 串口 `movb $0x7a, %al` 又是在干什么?

在VGA显存输出中, `movl $0x2f652f68, 0xB8000`是将字符“he”写入屏幕左上角。

在串口输出中, `movb $0x7a, %al`是将字符“z”写入串口。

四、代码布局

- .S中 `multiboot_header` 在内存占多少字节, 对齐方式

`multiboot_header`在内存中占16个字节(实际占12字节), 对齐方式为8字节对齐。

- VGA内存部分如何分配的, 为什么两个`movl`指令中间目标地址相差为4

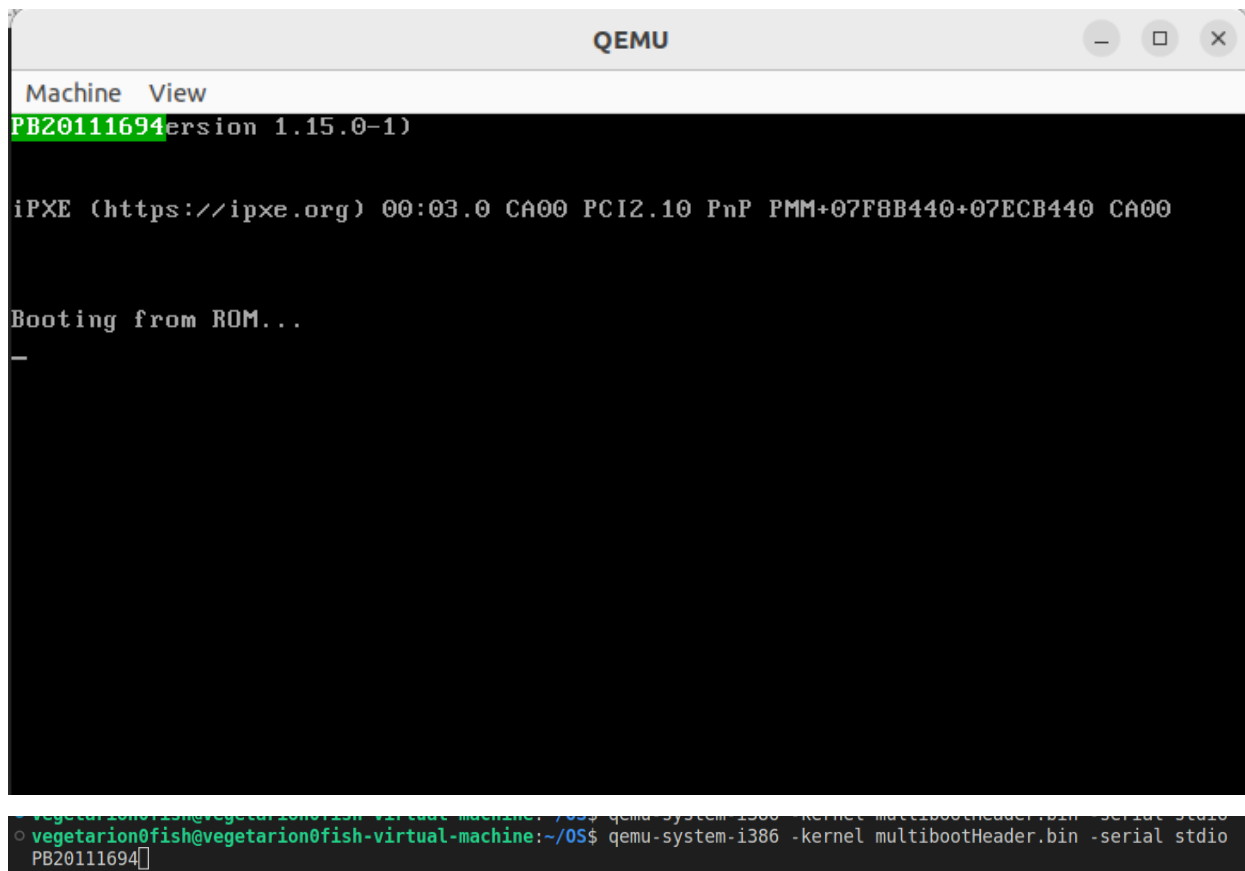
VGA显存被分配在物理地址范围0xB8000到0xBFFFF之间。两个`movl`指令中间目标地址相差为4是因为VGA显存每个字符占用2个字节, 一个字节表示属性, 另一个表示字符的ASCII码。

五、编译过程说明

生成bin文件 `ld -n -T multibootHeader.ld multibootHeader.o -o multibootHeader.bin`

运行qemu `qemu-system-i386 -kernel multibootHeader.bin -serial stdio`

六、实验结果 将输出界面截图



The screenshot shows a QEMU window titled "QEMU" with a "Machine View" tab. The main display area is black with white text. The text reads: "PB20111694 version 1.15.0-1)", "iPXE (https://ipxe.org) 00:03.0 CA00 PCI2.10 PnP PMM+07F8B440+07ECB440 CA00", and "Booting from ROM...". Below this, there is a horizontal line. At the bottom of the window, a terminal window is visible with the command "qemu-system-i386 -kernel multibootHeader.bin -serial stdio" and the output "PB20111694".

```
Machine View
PB20111694 version 1.15.0-1)

iPXE (https://ipxe.org) 00:03.0 CA00 PCI2.10 PnP PMM+07F8B440+07ECB440 CA00

Booting from ROM...
—

vegetarion0fish@vegetarion0fish-virtual-machine: /OS$ qemu-system-i386 -kernel multibootHeader.bin -serial stdio
PB20111694
```