



操作系统

操作系统大作业：part I-3
谭舜泉
计算机与软件学院

大作业Part I

- 大作业Part I 由六部分组成。六部分均为阅读xv6中文文档，理解对应的xv6源代码，并回答大作业中提出的代码理解问题。
 1. (第二周) 第0章：操作系统接口；
 2. (第四周) 第1章：第一个进程；
 3. (第六周) 附录A/B：PC硬件及引导加载器；
 4. (第八周) 第2章：页表；
 5. (第十周) 第4章：锁；
 6. (第十二周) 第3章：陷入、中断和驱动程序。



大作业Part I

- 注意上半学期的大作业分为两部分汇总提交。
- 其中1-3提交一份实验报告，而4-6提交另一份实验报告。



深圳大學
Shenzhen University

实验报告

- 实验报告要求使用深大实验报告模板填写。
- 允许两至三人组成一个小组，撰写实验报告。
- 请按照实验要求，完成实验并撰写实验报告。把实验报告电子版上传到**BLACKBOARD**。本实验分组完成。每个实验小组完成一份实验报告即可。但要求小组中每个成员都把实验报告的副本上传到**BLACKBOARD**，并注明同组成员的名字和学号。



深圳大学
Shenzhen University

附录A/B：PC硬件及引导加载器

■ CPU (中央处理器)

- 执行一个**非常简单的循环**：从一个被称为『程序计数器』的寄存器中获取一个内存地址，从该地址读出机器指令，增加程序计数器的值，执行机器指令，不断反复。
- 现代 x86 提供了**8个32位通用寄存器**-- %eax, %ebx, %ecx, %edx, %edi, %esi, %ebp, %esp 和一个程序计数器 %eip (instruction pointer) 。
- **控制寄存器** %cr0, %cr2, %cr3, %cr4，**段寄存器** %cs, %ds, %es, %fs, %gs, %ss；还有**全局和局部描述符表**的**伪寄存器** %gdtr, %ldtr 。



附录A/B：PC硬件及引导加载器

- 通常x86对操作系统隐藏了缓存，所以我们只需要考虑寄存器和主存两种存储器。
- 处理器必须像和主存交互一样同设备交互。x86处理器提供了特殊的 **in, out** 指令来在设备地址（称为'I/O 端口'）上读写。



XV6使用的汇编语言

- XV6使用GNU汇编器，而GNU汇编器使用AT&T指令语法。
 - 立即数前缀为\$ (push \$4)；寄存器前缀为% (%eax)。
 - 源操作数在前，目标操作数在后 (addl \$4, %eax)；
 - 指令显式声明操作数的尺寸 (b/字节, w/字, l/长字)：
 - movb foo, %al
 - 访问C语言变量“foo”：_foo。



附录A/B：PC硬件及引导加载器

- BIOS在启动时进行硬件的准备工作，接着把控制权把控制权交给从**引导扇区**（用于引导的磁盘的第一个512字节的数据区）**加载的代码**。
- 引导扇区中包含**引导加载器**——负责内核加载到内存中。
- BIOS 会把引导扇区加载到内存 **0x7c00**处。
- 引导加载器先处于处于模拟 Intel 8088 处理器的模式下，然后再把处理器设置为**现代**的操作模式。
- xv6 引导加载器：
 - **bootasm.S**
 - **bootmain.c**



代码理解问题(bootasm.S)

- 查找资料，回答以下问题：

1. 为什么主引导记录要存放在0x7C00开始的内存地址？（提示：这是历史遗留问题）
2. L21, “# Physical address line A20 is tied to zero...”这是著名的**Gate-A20**，请介绍一下为什么要设定**Gate-A20**。
3. L21-L38，这是一段让人一头雾水的代码，请查找资料，解释一下这段代码为何和**enable A20**有关。（参考<https://www.win.tue.nl/~aeb/linux/kbd/A20.html>）



代码理解问题

- 这些问题，如果不能理解的话也不用担心。我将在两周后的课堂上（在实验报告提交前）给予解答。

