



深圳大学
Shenzhen University

操作系统

操作系统大作业：part I-4
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。



深圳大學
Shenzhen University

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BLCKBOARD，并注明同组成员的名字和学号。



深圳大学
Shenzhen University

第2章：页表

- x86的页表硬件通过映射机制将虚拟地址和物理地址联系起来。
- 一个x86页表就是一个包含 2^{20} 条页表条目(PTE)的数组。每条PTE包含了一个20位的物理页号(PPN)及一些标志位。
- 分页硬件只需使用其高20位来找到该虚拟地址在页表中的索引，然后把其高20位替换为对应PTE的PPN。而低12位是会被分页硬件原样复制的。页表可以为操作系统提供对一块块大小为4096(2^{12})字节的内存片,这样的一个内存片就是一页。



第2章：页表

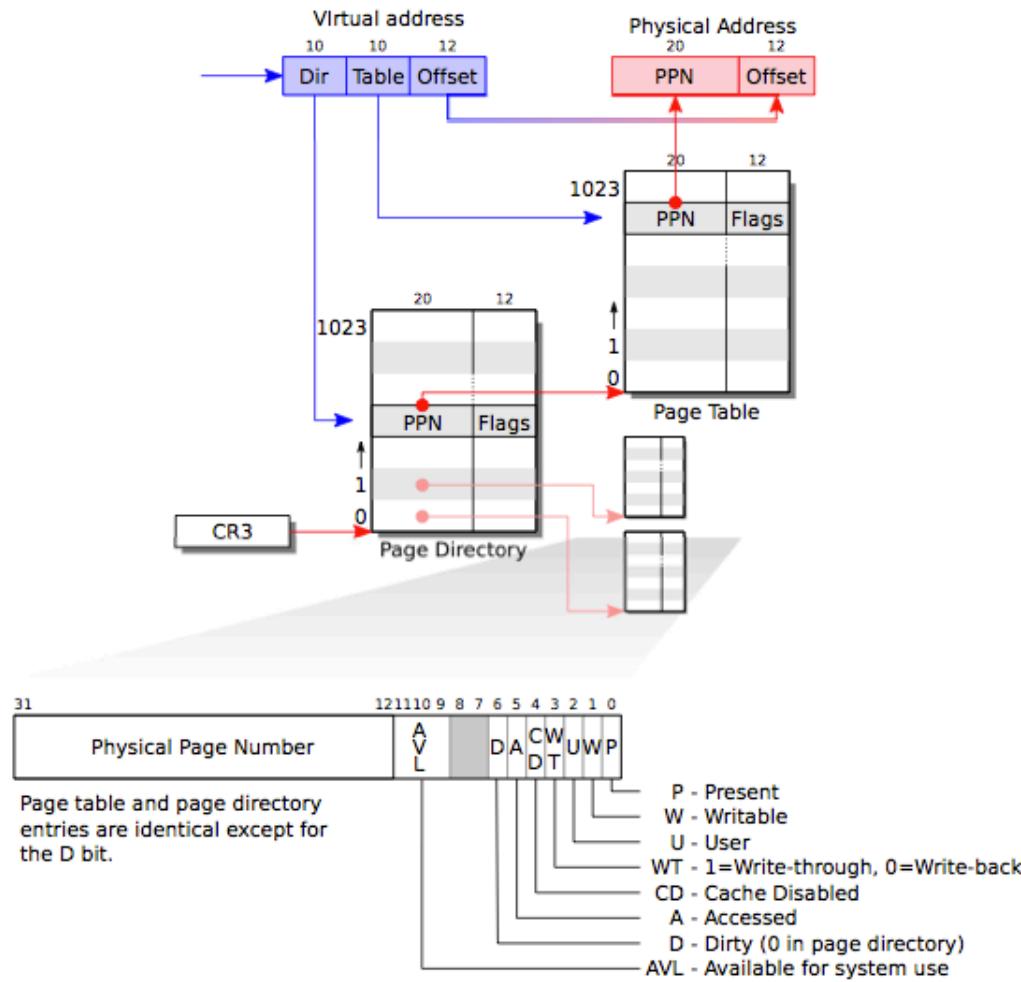


Figure 2-1. x86 page table hardware.

第2章：页表

- 一个页表在物理内存中像一棵 **两层的树**。
- 它的根是一个 4096 字节的页目录，其中包含了 1024 个类似 **PTE** 的条目，但其实每个条目是**指向一个页表页的引用**。
- 每个页表页又是**包含 1024 个 32位PTE** 的数组。
- 虚拟地址：高10位决定对应页目录条目，接下来的 10 位来从页表页中**选择出对应的 PTE**。否则，分页硬件就会**抛出错误**。



第2章：页表

- 每个进程的页表同时包括用户内存和内核内存的映射，这样当用户通过中断或者系统调用转入内核时就不需要进行页表的转换了。
- 大多数情况下，内核都没有自己的页表，所以内核几乎都是在借用用户进程的页表。



深圳大學
Shenzhen University

第2章：代码理解问题

- (vm.c L11) kpgdir被用于创建一个调度器所用的页表。请问理论上这个页表所支持的最大虚拟地址空间有多大？
- (vm.c L124) 在kmap中，DEVSPACE的phys_end为0？！请问在mappages函数中这一段的长度有多大。
- (vm.c L151) kpgdir = setupkvm();
通过setupkvm函数，创建了调度器所用的页表。请深入setupkvm函数内部，确定在创建页表过程中，总共调用了多少次kalloc函数分配4K物理块用于存放页表项？



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。



深圳大學
Shenzhen University