## 实验七 分页内存管理算法模拟

### 一、实验目的

1、熟悉基本分页存储管理。

2、建立描述分页内存管理中的页目录表、页表结构。

3、实现进行虚拟内存到物理内存的映射算法。

### 二、实验理论基础及教材对应关系

1、操作系统中内存管理。

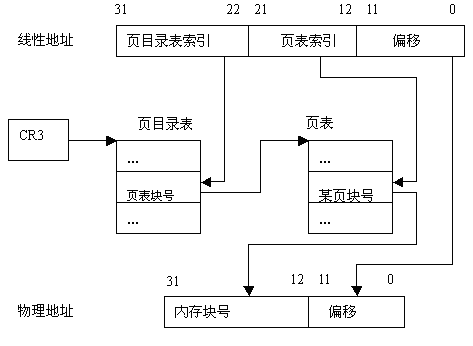
2、基本分页内存、分段内存管理。

3、页目录表、页表的作用，以及虚拟地址到物理地址的映射关系。

### 三、实验内容与步骤

题目：分页存储管理的设计与实现。

某系统采用了两级页表机制，可使页表所占用内存尽量少，分页地址变换机构如下图所示：

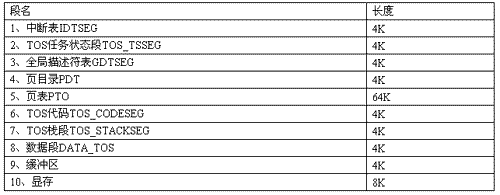


分页地址变换机构

页目录表共1024项，每个页表1024项，每页的大小是4K个字节。地址转换时，先由分段部件生成线性地址，再由上面所述的分页部件，根据线性地址中的页目录索引在页目录表中找相应的项，该项值为所需页表在内存的块号，找到该页表后，然后按第21－12位的页表索引找到所需页的物理内存起始地址，把它与12位偏移直接相加得到32位的物理地址。

设系统有如表1中所示的10个段，已知：1－8段从内存的200000H处开始由低地址到高地址连续存放，映射到3G＋4M开始的线性地址空间；9段（缓冲区）放在400000H开始的内存，映射的线性地址同物理地址；显存从B8000H开始，映射到3G开始的线性地址空间。

表1



（1）、请设计并填写页目录表和页表（需说明每张表的内存地址）

（2）、线性地址为：C0401010H、C0404010H、C0414010H，则物理地址是多少，所在段的段名是什么？（需写出计算的详细步骤）

实验步骤：

1、定义页目录表、页表的数据结构，以及必要的数据。

#define Page\_Size 4096 // 页面大小

#define Pages 26 // 本题定义的总的页面个数

#define FirstLinearAddr 0xC0000000+0x400000// 线性地址3G + 4M

#define SecondLinearAddr 0x400000 // 线性地址0x400000

#define ThirdLinearAddr 0xC0000000// 线性地址3G

#define IDT 0

#define TSS 1

#define GDT 2

#define PDT 3 // 页目录表的下标

#define PT1 4 // 第1 个页表的下标

#define PT2 5 // 第2 个页表的下标

#define PT3 6 // 第3 个页表的下标

#define PT4 7 // 第4 个页表的下标

// ......省略其它页表

#define CODE 20

#define STACK 21

#define DATA 22

#define BUFFER 23

#define DISPLAYMEM 24

2、初始化页目录表、页表中的数据

p = (unsigned int \*)PysicalMemAddr[PDT]; // p 指向页目录表

p[FirstLinearAddr>>22] = (unsigned int)PysicalMemAddr[PT1];// 将第1 个页表的地址填入页目录表中

p = (unsigned int \*)PysicalMemAddr[PT1]; // p 指向第1 个页表

p[(FirstLinearAddr+4096\*IDT)>>12 & 0x3FF] = (unsigned int)PysicalMemAddr[IDT];// 将 IDT 页的起始地址填入页表

p[(FirstLinearAddr+4096\*TSS)>>12 & 0x3FF] = (unsigned int)PysicalMemAddr[TSS];// 将 TSS 页的起始地址填入页表

p[(FirstLinearAddr+4096\*GDT)>>12 & 0x3FF] = (unsigned int)PysicalMemAddr[GDT];// 将 GDT 页的起始地址填入页表

p[(FirstLinearAddr+4096\*PDT)>>12 & 0x3FF] = (unsigned int)PysicalMemAddr[PDT];// 将 PDT 页的起始地址填入页表

p[(FirstLinearAddr+4096\*PT1)>>12 & 0x3FF] = (unsigned int)PysicalMemAddr[PT1];// 将 PT1 页的起始地址填入页表

p[(FirstLinearAddr+4096\*PT2)>>12 & 0x3FF] = (unsigned int)PysicalMemAddr[PT2];// 将 PT2 页的起始地址填入页表

p[(FirstLinearAddr+4096\*PT3)>>12 & 0x3FF] = (unsigned int)PysicalMemAddr[PT3];// 将 PT3 页的起始地址填入页表

p[(FirstLinearAddr+4096\*PT4)>>12 & 0x3FF] = (unsigned int)PysicalMemAddr[PT4];// 将 PT4 页的起始地址填入页表

3、虚拟地址到物理地址的变换

linear = 0xC0401010;

p = (unsigned int \*)PysicalMemAddr[PDT]; // p 指向页目录表

pTable = (unsigned int \*)p[linear>>22]; // pTable 指向页表

pChar = (char \*)pTable[linear>>12 & 0x3FF]; // pChar 指向物理内存

printf("Linear: 0x%X is in %s\n",linear,pChar);

自行变换线性地址：C0404010H、C0414010H

### 四、实验材料的提交与成绩评定

本实验的实验报告一份（电子版一份，格式参考学校实验报告）