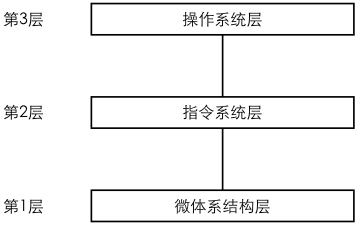
**一、操作系统层**

**1操作系统层的架构**

概述：在指令系统层的基础上实现的程序(Linux由C语言编译而成)



功能：管理和控制硬件与软件资源

**2操作系统层指令集**

系统调用：由操作系统定义的系统服务程序

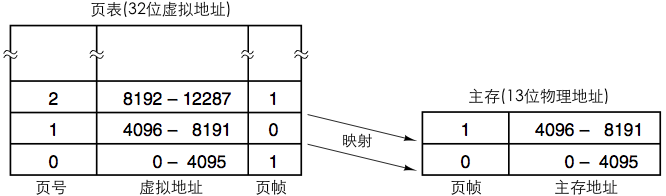
操作系统指令集：应用程序可用指令集（部分指令系统层指令和系统调用）

操作系统内核指令集：内核可用指令集（所有指令系统层指令和系统调用）

**二、内存管理**

**1虚拟地址空间(分页)**

概念：程序使用地址域(如32位)对应的存储空间，该空间被分页映射到主存使用

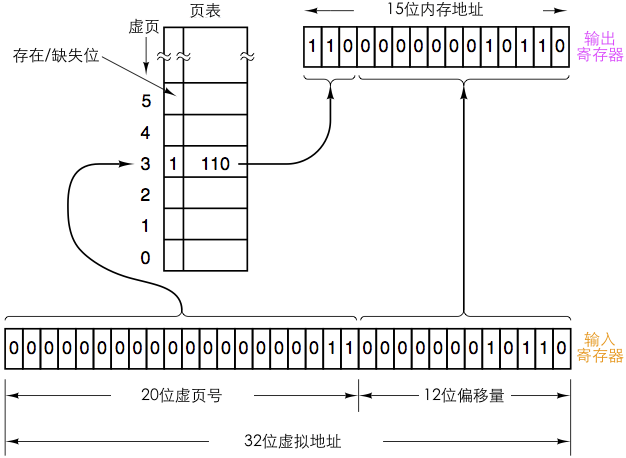


·页表 虚页号与页帧的映射表

·虚页 虚拟地址被分割的项（4K的地址域）

·页帧 内存地址被分割的项（4K的地址域）

虚拟地址寻址：由内存管理单元(MMU)负责映射32位虚拟地址到15位物理地址



页置换：当访问虚拟地址对应页不在内存时，将触发调页操作

·将内存的页帧保存到磁盘

·从磁盘中找出虚拟地址对应的页并调入上述页帧

·修改页表中存在/缺失位(被调出页置为0，被调入页置为1)

页置换策略

·最近最少使用

·先进先出

**2逻辑地址空间(分段)**

概念：程序可访问的独立地址空间(地址域0~Max)，该空间又称段



·段表 描述各个段在内存的分布

·段号 被分段项的标示

特性

·地址域独立(0~Max)

·每个空间都是一个逻辑实体的(过程、数组、栈等)

·可动态改变空间长度

逻辑地址寻址



段交换：当访问逻辑地址对应段不在内存时，将触发段交换(同调页操作)

段交换策略：段大小不固定，交换会在段之间产生内存碎片，有如下策略

·最佳匹配

·最先匹配

**3段页式内存**

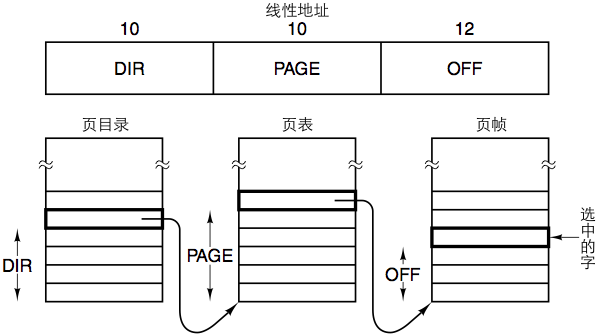
概念：在虚拟地址空间中分段

逻辑(段)地址空间的访问：

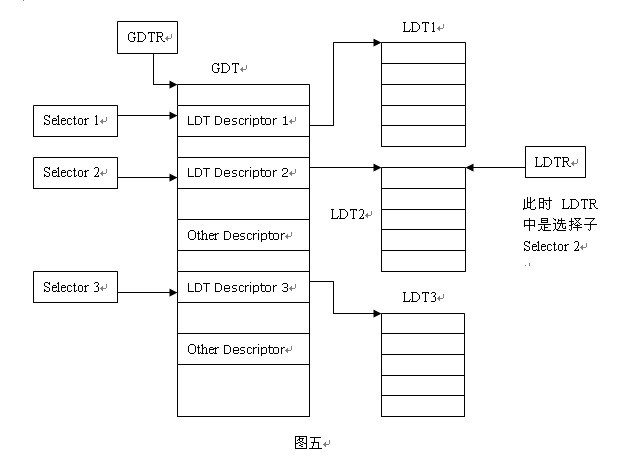
·用段号和段内地址进行逻辑地址寻址，获得段的线性地址

·用线性地址进行虚拟地址寻址，获得物理地址

带页目录的虚拟地址空间：地址被分为页目录、页表、页帧三个层级



Core i7的虚拟内存



·GDT 全局描述符表，描述系统用到的段，系统只有一张表（多CPU待定）

·LDT 局部描述符表，描述每个程序用到的段，每个程序都有自己的LDT

**三、操作系统层I/O指令**

**1产生背景**

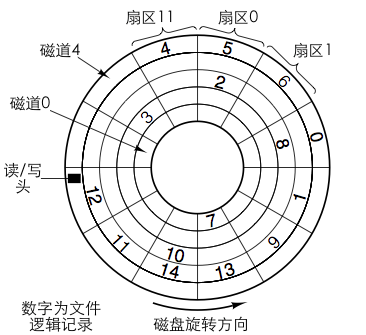
安全性：指令系统层I/O权限过高，可修改其他用户数据、实现复杂等问题

复杂度：指令系统层I/O实现过于复杂

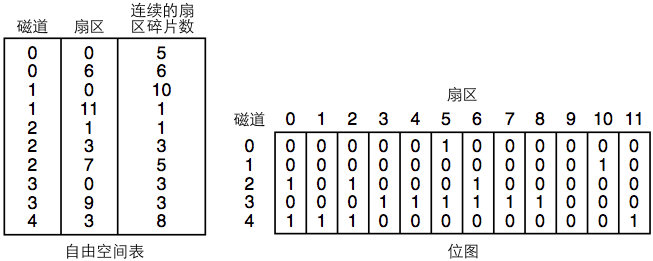
**2文件存储的结构**

文件索引表：文件逻辑记录的磁盘地址表，unix用磁盘分配单元地址组成

文件的磁盘存储结构：将文件的逻辑记录存储在磁盘不连续的分配单元



为文件分配磁盘空间：通过查询磁盘自由空间表和位图表给出可分配单元



·自由空间易于分配指定长度的磁盘

·位图可方便记录使用和修改使用状态

**3文件操作指令**

open、read、write、close等系统调用

**2目录操作指令**

创建文件并放入目录

从目录中删除一个文件

重命名一个文件

改变文件的保护状态

**四、并行处理指令**

**1进程管理**

**2进程同步**