

# Omos (Our Mini Operating System) 软件需求报告

2009 软件工程 dota 组

撰写人: 彭兆盟 无 67 2006010910

同组人: 陶帅 无 67 2006011170

王光回 无 67 2006011161

王逸天 无 67 2006011171

易硕 无 67 2006011178

## 一、引言

### 1、目的

21 世纪是信息时代, 随着 IT 行业迅速发展, 我们的生活已经离不开计算机等电子设备。而计算机离不开操作系统——一个管理硬件设备, 分配硬件资源的软件。操作系统是各种电子产品的灵魂, 大到巨型机, 小到手机、pda, 都离不开操作系统。因此对于信息科学的学生来说, 掌握一些操作系统的知识是十分必要的。

当今比较成功的操作系统有 windows、linux、Mac OS、freeBSD 等等, 但这些操作系统都十分复杂, 对初学者来说要想深入了解其原理十分困难。因此需要一个简单的但具有现代操作系统核心功能的小型操作系统作为教学平台使用。国外在这方面有一些专用的教学操作系统, NACHOS (Not Another Completely Heuristic Operating System) 是一个使用得较多的操作系统教学平台, 但是, 它只模拟 MIPS 处理器, 跟主流的 X86 平台不一样。并且, 它与硬件相隔离的特点也让它在教学的目的上显得力不从心。MINIX, Andrew S. Tanenbaum 编写的方便自己教学的一个小型操作系统, 也是国外操作系统教学的范例。这是一个实际的操作系统, 可以运行在实际的硬件平台上或 X86 模拟器 Bochs。然而, 它的庞大性却使得教师们在使用它的时候陷入两难——相对于教学课时, 它的代码量显得实在是过于庞大。

正是基于这个原因, 我们需要有一个能够为学习操作系统服务的平台。对于我们编写者自己来说也是一个通过实践进行学习的过程, 学习操作系统如果只是学习理论知识而不亲自实践的话, 不仅枯燥而且难于学的精通。通过开发 omos (Our Mini Operating System) 操作系统, 我们不仅自己可以学到操作系统的知识, 同时也为其他希望学习掌握操作系统知识的同仁们提供了方便。

### 2、预期读者

本文档的预期读者为本软件的开发、测试人员, 和对操作系统的原理感兴趣并希望利用我们的操作系统进行学习的软件用户。

### 3、产品范围

Omos 操作系统是一个遵从 GPL 协定的开源操作系统, 以方便希望通过实践学习操作系统的同仁参考, 代码发布在 <http://code.google.com/p/omos/>。

### 4、参考文献

- [1]Anderw S. Tanenbaum. 现代操作系统(英文版第 3 版), 机械工业出版社, 2009
- [2]于渊. 自己动手写操作系统. 电子工业出版社, 2005
- [3]杨文博. 使用开源软件自己动手写操作系统, 2008
- [4]谢睿. MOMOS——一种教学操作系统的设计与实现, 清华大学综合论文训练, 2009
- [5]nasm manual: <http://www.nasm.us/doc/>

## 二、综合描述

### 1、软件前景

虽然目前各种操作系统百花齐放，而且还有各种开源操作系统，但真正为学习操作系统而服务的代码简单而又包含操作系统核心功能的操作系统却寥寥无几。对于从事 IT 行业的工作者来说，了解操作系统的原理是必不可少的。本软件可以作为学习或教学服务的操作系统的一个例子，通过实践为操作系统的理论学习提供帮助。因此 omos 操作系统的应用前景还是十分广阔的。

### 2、软件功能

由于 omos 操作系统是为学习操作系统服务而不是实现一个复杂的功能强大的操作系统，因此这就决定了它的两个特点：第一，要是真实的操作系统，能在实实在在的硬件上运行，并能实现操作系统的核心功能。第二，这个操作系统不宜过于复杂，过于复杂对于初学者来说学习起来很困难。具体而言，应包含以下功能：OMOS 平台至少需要对传统的冯·诺依曼体系结构中的基本硬件都有涉及，包括 CPU、主存、硬盘、时钟等；能在运行在真实的 X86 平台保护模式下，实现系统启动、多进程管理、文件管理以及键盘输入显示器输出等功能，实现类似 DOS 的命令行方式的操作。对于用户来说，它为用户或者说是软件提供了系统调用接口，为上层软件屏蔽具体的硬件参数。

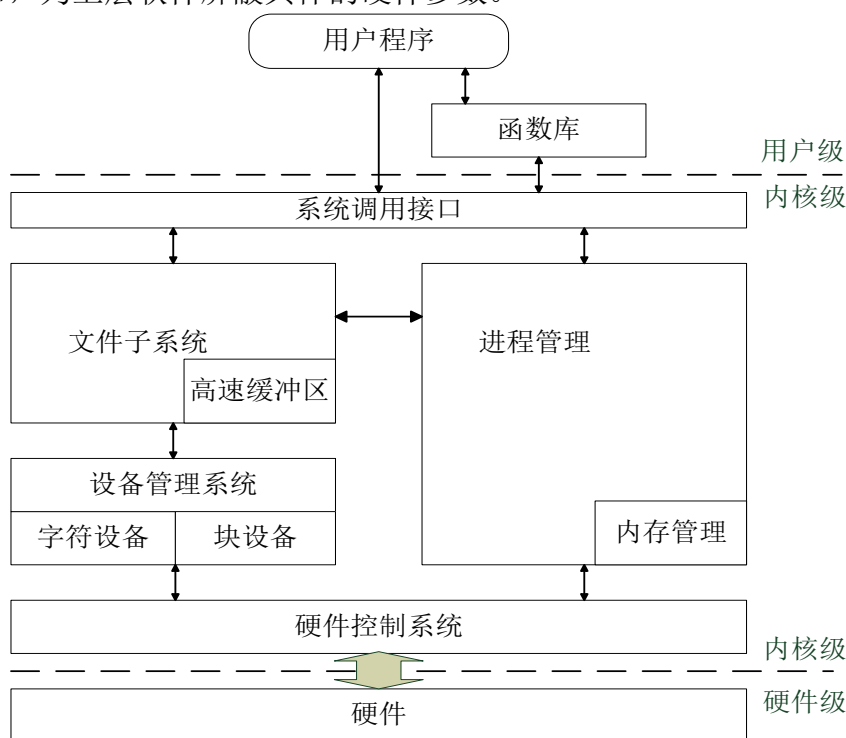


图 1 omos 体系结构

### 3、运行环境

我们希望 omos 操作系统能运行在主流的真实 X86 体系电脑上。但由于条件限制，我们只需在虚拟机如 virtual PC 或 Bochs 上进行验证即可。因此运行环境是正确安装并配置了 virtual PC 或 Bochs（windows 或 Linux 均可）的拥有键盘、显示器的完整的 X86 电脑，并有足够的空间划分给 omos 运行。

### 4、用户类和特征

本软件初步定位对操作系统感兴趣、希望学习操作系统的 IT 工作者。

## 5、设计和实现限制

由于 omos 操作系统的设计思想——为学习操作系统服务而不是实现一个复杂的功能强大的操作系统，因此这个操作系统不宜过于复杂，最好能以最简单的代码实现最核心的功能。同时由于时间关系，我们也只实现一些简单的功能如进程调度、键盘输入、显示器输出等，不支持鼠标、扬声器等复杂的设备，更没有 GUI。

### 三、外部接口需求

#### 1、用户界面

用户界面为类似 DOS 的命令行界面。

#### 2、硬件接口

X86 主机，PS/2 口键盘，和显示器。

#### 3、软件接口

**开发、编译环境：**

由于我们编写的是操作系统，任何库函数、API 都不能使用，我们有的只有编程语言。要与硬件交互，我们需要使用汇编语言，而对于内核部分，我们可以使用高级语言。由于 C 语言是公认的最适合开发操作系统的语言，因此 Omos 的开发使用的是 C 语言与汇编语言，同时尽量使用 C 而不用汇编。

由于需要平台无关性，我们选择 gcc 和 nasm 作为编译器，LD 作为链接器，用 Bochs 或 virtual PC 作为模拟运行环境。

### 四、系统特性

#### 1、说明和优先级

一个完整的计算机系统通常包括 4 个部分：硬件系统、操作系统内核、操作系统服务和应用程序。Omos 设计与实现的主要是操作系统内核和部分系统服务。操作系统内核与硬件进行交互，实现对硬件部件的编程控制和接口操作，调度对硬件资源的访问。

#### 2、激励/响应序列

计算机用户通过系统调用申请操作系统为其完成一项工作，操作系统由用户态切换到内核态，然后操作系统根据用户提供的参数调用相应的服务处理程序，这些服务处理程序再调用底层的相关函数完成用户的要求，然后操作系统返回用户态。

#### 3、功能需求

Omos 操作系统首先要能正确启动，启动程序的任务是将为计算机加载 Omos 的内核映像文件，为 Omos 运行做准备工作。内核从模块上来分，主要有 4 个模块：进程管理模块、内存管理模块、文件系统模块、设备管理模块。进程管理模块主要负责控制进程对 CPU 资源的使用、调度原则是根据进程的优先级和时间片轮转法，使得进程能够合理有序地进行轮流执行。内存管理模块用主要负责所有进程能够合理有效地利用有限的内存，即不造成内存争抢的冲突，又不造成内存死角和浪费。文件系统模块主要负责文件的管理与存储，保证文件在硬盘合理有序地存放，既不浪费磁盘空间又使得访问文件快速方便。设备管理模块，主要负责与硬件交互，并向上层提供抽象的封装的硬件功能，使得上层模块不需要与具体的硬件参数打交道。其中，进程管理模块和文件系统模块向用户程序提供系统

调用接口，用户程序将请求通过系统调用发送给这两个模块，由这两个模块调用相关函数完成用户程序的要求。由于进程在运行过程中可能会使用到文件，所以进程管理模块与文件系统模块需要通讯。进程管理模块同时也会与时钟硬件交互，以公平地将 CPU 分配给每一个进程。文件系统则不直接与硬件交互，而是通过设备管理系统间接地与字符设备和块设备进行交互。

## 五、其他非功能要求

### 1、性能需求

由于软件定位于学习操作系统，对性能没有太多要求，能在常见 X86 电脑上正常运行即可。

### 2、软件质量属性

能稳定运行，最大限度减少崩溃概率。

## 附录 A 词汇表

符号	全称	意义
BPB	BIOS Parameter Block	FAT 文件系统 BIOS 参数块
CPL	Current Privilege Level	当前代码的特权级
DPL	Descriptor Privilege Level	描述符特权级
DRQ	Data Request	数据请求服务
ECC	Error Checking and Correcting	一种校验和纠错方式
ELF	Executable and Linking Format	一种可执行文件格式
FAT	File Allocation Table	文件分配表
GDT	Global Descriptor Table	全局描述符表
IDT	Interrupt Descriptor Table	中断描述符表
ICW	Initialization Command Word	初始化命令字
IRQ	Interrupt Request	中断请求
MBR	Master Boot Record	主引导纪录
NASM	Net-wide Assembler	用到的汇编编译器
OMOS	Our Mini Operating System	所设计的操作系统的名字
PD	Page Directory	页目录
PT	Page Table	页表
RPL	Request Privilege Lever	请求特权级
TTY	Teletype	终端
WIN_SECIFY		建立驱动器参数命令
WIN_RESTORE		校正硬盘控制器命令