

第一章作业

题目 1 解答：

1.主机系统或微型计算机：不同的用户通过终端访问同一计算机，共享该计算机资源且可交换数据。所以操作系统需要确保 CPU 时间、内存资源、I/O 设备能够被有效分配，还要确保用户的操作是否超出访问权限。

2.通过服务器连接的工作站：工作站和其他的工作站与服务器相连，即可作为个人计算机为用户服务、也能以一定权限访问服务器。所以操作系统需要确保 CPU 时间、内存资源、网络带宽、文件存储空间充分利用，此类操作系统的目的是在性能和资源利用率之间平衡。

3.手持计算机：手持 PC 为个人使用，由于受电源、内存大小和接口限制，操作系统需要在最大化用户使用感的同时仔细管理功耗、内存资源。

题目 2 解答：

计算机存储层次体系中，离 CPU 越近的存储器速度越快，单位字节的成本也就越高，同时存储器容量也更小：

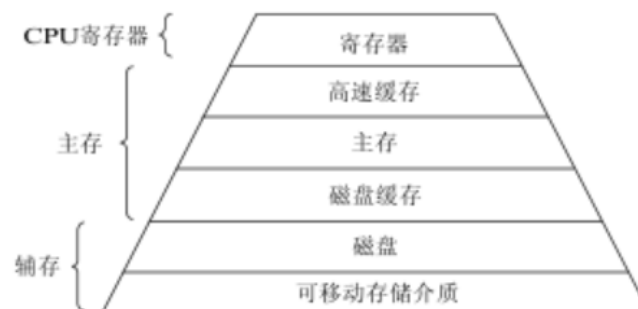


图 4-1 计算机系统存储层次示意

对于软盘来讲,属于辅存中的可移动存储介质,应该具有容量大、访问时间长、价格便宜和非易失的特点。但是常用的软盘容量仅有 1.4MB、存取速度慢。而且对于存储条件有要求——不要划伤盘片,盘片不能变形、不能受高温、不能受潮、不要靠近磁性物质等等,否则将造成数据丢失。这不符合存储器存储层次的规律,并没有使得容量与速度之间达到相对好的平衡,所以当出现了 DVD、U 盘等容量更大、速度更快的存储设备时,软盘便开始走向消亡。

题目 3 解答:

1.中断的作用:

- a) 通过中断机制,当计算机遇到突发故障时,比如存储出错、运算溢出、电源掉电等,能够利用中断处理函数自行处理。
- b) 通过中断机制,可以使得即使是单个 CPU 也能在不同进程间切换作业,从而实现多任务工作,提高运行效率。
- c) 通过中断机制,使得 CPU 能与外界 I/O 设备交换数据的同时还能够进行其他作业。

总结而言,中断的作用就是:在解收到中断请求之后,停止当前的用户进程或者优先级更低的中断处理程序,转而执行优先级更高的中断处理程序,完成之后再切换回引起中断的指令。一言以蔽之,操作系统是由中断驱动的。

2.陷阱和中断的区别:

从中断的不同分类来看:

- a) 陷阱属于同步中断,而一般所说的中断为异步中断;陷阱通常是由于 CPU 正在执行的指令所引起,比如系统调用,属于“有意为之”;异步中断通常由与现执行指令无关的突发中断源引起的,无法预测,比如电源掉电、算术溢出等。

b) 陷阱属于软中断，而一般所说的中断为硬中断；陷阱是由当前进程所产生的，通常是对一些 I/O 的请求；硬中断是由 HardWare 产生，通过中断请求使得 CPU 挂起当前执行程序，执行相应的中断处理函数。所以陷阱并没有直接中断 CPU，因为中断源就是当前程序，只是中断了当前进程的流程，去调用中断处理函数。

3. 用户程序是否可以有意地生成陷阱？目的是？

是；通过陷阱指令，可以使执行流程从用户态陷入内核并且把控制权转移给操作系统，使得用户可以调用内核函数以及 I/O 设备。

题目 4 解答：

1. CPU 如何与设备协作完成数据传输？

①当执行到需要调用 I/O 设备传输数据的指令时，CPU 把准备用来存储输入数据的内存起始地址、以及传送字节数写入 DMA 控制器中的内存地址寄存器和传送字节计数器。②该进程被挂起，等待数据传输完成，其他进程被调度占据 CPU。③此时外接设备持续工作，设备控制器中的数据缓冲寄存器持续写入内存，直到字节数达到所要求传输的字节数。④所有字节传输完成，DMA 通过中断请求触发中断，CPU 执行完中断处理函数之后，返回到被中断的进程继续执行。

此时一轮数据传输完成。

2. CPU 如何知道内存操作已经结束？

DMA 在达到要求传输的字节数后，通过中断请求触发中断，要求 CPU 进行数据处理的方式通知 CPU 内存操作已经结束。

题目 5 解答：

1.CPU 中提供双模式 (User, Monitor) 有什么用途：

系统中既有用户的程序，也有操作系统的程序，为了安全性和稳定性，不能随意访问操作系统的程序，这就是内核态的作用，只能通过内核态才能执行操作系统的程序。用户态下想要执行系统程序就先要转化成内核态，这样保护了对系统资源的使用，也确保用户程序不会访问权限之外的存储空间。

特权指令（当前特权级 CPL=0）：

- a) 对 I/O 设备的调用的指令（如输出语句）；
- b) 访问程序状态的指令（如加载程序状态字 PSW 的指令）；
- c) 清理内存的指令；
- d) 存取特殊寄存器（中断寄存器、始终寄存器）的指令；
- e) 控制中断禁止屏蔽位（用来允许/禁止中断）的指令；
- f) 停机指令（HLT 指令）；
- g) 系统调用指令（INT 0x80 指令）；

非特权指令：

- a) 运算操作的指令（add、sub、imul、idivl）；
- b) 比较指令和跳转指令（cmp、jmp、je、jne.....）；
- c) 不涉及寄存器值的数据传输指令（mov 类指令）；

题目 6 解答：

- a) 操作系统为应用程序提供编程接口；
- b) 对一个程序的执行从 main 或中断为入口；

- c) 操作系统是由中断驱动的，中断、异常、系统调用使操作系统进入管态调用相应的系统函数，中止了当前的进程流，使得操作系统能够处理更多的突发事件，而不是单按照既定的指令序列工作；
- d) 当没有进程占据 CPU 时，会执行 Idle Loop，在闲置循环中等待；

题目 7 解答：

开源组织介绍：

Apache 软件基金会，是专门为运作一个开源软件项目的 Apache 的团体提供支持的非盈利性组织，Apache 所支持的 Apache 项目与子项目中所发行的软件产品都遵循 Apache License。

Linux 基金会是一个非盈利性的联盟，其目的在于协调和推动 Linux 系统的发展，以及宣传、保护和规范 Linux，该组织是 2007 年，由开源码发展实验室 (Open Source Development Labs, OSDL)与自由标准组织 (Free Standards Group, FSG)联合起来成立的。

OpenSymphony 项目为应用开发提供了一套组件，用于快速开发企业 Java 应用。该项目的构架是多层的基于组件的松耦合构架。OpenSymphony 的组件分为 5 个层次: Foundation, Business, Interaction, Presentation, Applications。

开源协议介绍：

GNU GPL (GNU General Public License GNU 通用公共许可证)

- 复制自由：允许把软件复制到任何人的电脑中，并且不限制复制的数量。
- 传播自由：允许软件以各种形式进行传播。

- 收费传播：允许在各种媒介上出售该软件，但必须提前让买家知道这个软件是可以免费获得的；因此，一般来讲，开源软件都是通过为用户提供有偿服务的形式来盈利的。
- 修改自由：允许开发人员增加或删除软件的功能，但软件修改后必须依然基于 GPL 许可协议授权。

BSD(Berkeley Software Distribution, 伯克利软件发布版)许可协议

- 如果再发布的软件中包含源代码，则源代码必须继续遵循 BSD 许可协议。
- 如果再发布的软件中只有二进制程序，则需要在相关文档或版权文件中声明
原
始代码遵循了 BSD 协议。
- 不允许用原始软件的名字、作者名字或机构名称进行市场推广。

Apache 许可证版本(Apache License Version)许可协议

- 该软件及其衍生品必须继续使用 Apache 许可协议。
- 如果修改了程序源代码，需要在文档中进行声明。
- 若软件是基于他人的源代码编写而成的，则需要保留原始代码的协议、商标、专利声明及其他原作者声明的内容信息。
- 如果再发布的软件中有声明文件，则需在此文件中标注 Apache 许可协议及其他许可协议。

开源运动对操作系统的影响：

第一个开源操作系统 Linux 便萌发在开源环境中，遵循 GPL 协议下任何个人和机构都能自由地使用 Linux 源代码，在首创者 Linus Torvalds 和社区贡献者的参与下一起开发和维护 Linux 内核，开源运动大大加快了发展，来自世界各地的 Linux 内核贡献者都参与到这一项目中，开创了高效率的开发方式。基于 Linux 内核的 Android 更是成为了全球最流行的智能手机操作系统，而 Android 的广泛普及也让开源更加深入大家的视线中。开源运动影响了操作系统开发的模式，开放的方式使得所有人都能参与代码的学习、纠错，这减少了操作系统的开发周期，并且 bug 也有更多人来修复，更重要的是开源项目有更好的流传度，更多的人将能够体会到开源带来的便捷之处。