

## 第七章 习题及解答

7-1 什么是设备独立性？引入这一概念有什么好处？

答：所谓设备独立性是指，用户在编制程序时所使用的设备同实际使用的设备无关，也就是在用户程序中仅使用逻辑设备。

引入设备独立性，可使应用程序独立于物理设备。此时，用户编程只需用逻辑设备去请求使用某类设备。当系统中有多台该类设备时，系统将其中的任一设备分配给请求进程，而不必局限于某一指定设备。这样，可以显著地提高资源的利用率和可适应性。

独立性还可以使用户程序独立于设备类型。例如，在进行输出时，既可以利用显示终端进行输出，也可以利用打印机进行输出。有了这种适应性，就可以很方便地实现输出重定向，类似地可以实现输入重定向。

7-4 什么是缓冲？引入缓冲的原因是什么？

答：缓冲是两种不同速度的设备之间传输信息时平滑传输过程的常用手段。

引入缓冲技术的原因有如下几点。

(1) 缓和 CPU 和 I/O 设备之间速度不匹配的矛盾。

(2) 减少中断次数和 CPU 的中断处理时间。如果没有缓冲，慢速 I/O 设备每传一个字节就要产生一个中断，CPU 必须处理该中断；如果采用了缓冲，则慢速 I/O 设备将缓冲填满时，才向 CPU 发出中断，减少了中断次数和 CPU 的中断处理时间。

(3) 解决 DMA 或通道方式下数据传输的瓶颈问题。DMA 或通道方式都用于成批数据传输，在无缓冲的情况下，慢速 I/O 设备只能一个字节一个字节的传输信息，成了 DMA 或通道方式数据传输的瓶颈。缓冲的设置适应了 DMA 或通道方式的成批数据传输方式，解决了数据传输的瓶颈问题。

### 7-5 常用的缓冲技术有哪些？

答：常用的缓冲技术有双缓冲、环形缓冲和缓冲池。

引入双缓冲以提高处理机与 I/O 设备之间的并行操作程度，例如，输入设备先将第一个缓冲装满数据，在输入设备向第二个缓冲装数据时，处理机就可以从第一个缓冲中取出数据进行处理。第一个缓冲的数据处理完毕，若第二个缓冲已经装满数据，则处理机又可以从第二个缓冲中取出数据进行行处理，而输入设备又向第一个缓冲装填数据。

为了在 CPU 与 外设对信息的操作速度相差甚远时仍能得到良好并行效果，可以采用环形缓冲技术。环形缓冲技术是在主存中分配一组大小相等的存储区作为缓存区，并将这些缓存区链接起来，每个缓存区中有一个指向下一个缓存区的指针，最后一个缓存区的指针指向第一个缓存区，这样  $n$  个缓存区就成了一个环形缓冲外，系统中有个缓冲链首指针指向第一个缓存区。环形缓冲用于输入输出时，需要两个指针  $in$  和  $out$ ， $in$  指向第一个空缓存区， $out$  指向第一个装满数据的缓存区。输入时，把数据输入到  $in$  所指的缓存区中，然后  $in$  模取后移一位，指向下一个空缓存区。输出时，从  $out$  所指的满缓存区中取出数据，然后  $out$  模取后移一位，指向下一个满缓存区。

缓冲池是由若干个大小相等的缓存区组成的。缓冲池中的每一个缓存区都由系统统一管理和动态分配。若某个进程需要使用缓冲时便提出申请，由系统将缓存区分配给它，进程不再使用缓存区时，就将缓存区交还给缓冲池。这样，就可以用少量的缓存区服务更多的进程。缓冲池通常将缓存区排成 3 个队列：空闲缓存区队列、输入缓存区队列和输出缓存区队列。

### 7-8 什么是独占设备？对独占设备如何分配？

答：独占设备是指在一段时间内只允许一个用户进程访问的设备。系统一旦把这类设备分配给某进程后，便由该进程独占直到使用完后释放。多数低速 I/O 设备都属于独占设备，如打印机等。

独占设备采用独占分配方式，即将一个独占设备分配给某进程后便一直由它独占，直到该进程完成或释放该设备时，系统才能将该设备分配给其他进程。

7-9 什么是共享设备？对共享设备如何分配？

答：共享设备是指在一段时间内允许多个进程同时访问的设备，如磁盘。对共享设备可将其同时分配给多个进程，使用共享分配方式显著提高了设备的利用率，但对设备的访问需进行合理的调度。

7-10 什么是虚拟设备技术？什么是虚拟设备？如何进行虚拟分配？

答：所谓虚拟设备技术，是在一类物理设备上模拟另一个物理设备的技术，是将独占设备转换成共享设备的技术。目前广泛流行的虚拟设备技术是 SPOOLing 技术，网络环境中的虚拟打印机。

虚拟设备是指通过虚拟技术将一独占设备变换成若干台逻辑设备，供若干个用户进程使用，通常把这种经过虚拟技术处理的设备称为虚拟设备。引入虚拟设备的目的是为了克服独占设备速度较慢、资源利用率较低的缺点，以提高设备的利用率。

虚拟分配是针对虚拟设备而言的。当进程申请独占设备时，由系统分配给它共享设备，如磁盘的一部分存储空间。当进程要和设备交换信息，系统就将要交换的信息放到这部分存储空间中，在合适的时候，系统再将存储空间中的信息传到独占设备。