

第 5 章 设备管理

1. 设备通常有哪几种分类方法？可以分为哪几种类型？

【解答】从不同的角度，可以对设备进行不同的分类。

- (1) 按使用特性分类：存储设备、输入/输出设备
- (2) 按传输速率分类：低速设备、中速设备、高速设备
- (3) 按信息传输单位分类：块设备、字符设备
- (4) 按资源分配方式分类：独占设备、共享设备、虚拟设备

2. 数据传输控制方式有哪几种？各有什么特点？

【解答】数据传输控制方式有 4 种，即程序直接控制方式、中断控制方式、DMA 方式、通道控制方式。

(1) 程序直接控制方式。又称为“忙等”方式或“查询”方式。该方式在启动设备进行一次 I/O 操作前，要执行一个循环查询程序，反复查询外设状态，直至查得外设状态为“空闲”时为止。该方式的优点是控制简单，也不需要多少硬件支持，但 CPU 和外设只能串行工作，使 CPU 的利用率大大降低；CPU 在一段时间内只能和一台外设交换数据信息，从而不能实现设备之间的并行工作；由于程序直接控制方式依靠测试设备的状态来控制数据传送，因此，无法发现和处理因设备或其他硬件所产生的错误。所以，程序直接控制方式只适合于那些 CPU 执行速度较慢且外设较少的系统。

(2) 中断控制方式。该方式在进行 I/O 时，CPU 向控制器发出 I/O 命令后，由控制器控制外设操作，CPU 转去执行其他任务，CPU 与外设并行工作，当外设完成 I/O 操作后向 CPU 发送中断信号，CPU 执行中断处理程序进行 I/O 的善后处理。该方式的优点是能实现 CPU 与设备以及设备与设备之间的并行操作，CPU 利用率较程序直接控制方式大大提高；但由于 I/O 控制器的数据缓冲寄存器装满数据后将会发出中断，且数据缓冲寄存器通常较小，因此在一批数据传送过程中发生中断次数较多而耗去大量 CPU 时间；如果系统中配置的外设数目较多，且都以中断方式进行操作，则可以耗去大量的 CPU 时间或因 CPU 来不及处理而造成数据的丢失。

(3) DMA 方式。该方式用于直接在内存和外设之间传送数据，不需经过 CPU 中转，一般用于内外存之间的数据传输，一次可以传输一个或多个数据块。与中断方式类似，DMA 方式是在一批数据传送完成后中断 CPU，从而大大减少了 CPU 进行中断处理的次数；但 DMA 方式仍有一定的局限，如对外设的管理和某些操作仍由 CPU 控制，多个 DMA 控制器的使用也不经济。

(4) 通道控制方式。通道是一个专管输入/输出控制的处理机。在通道控制方式下，系统预先将控制 I/O 的过程实现为一段通道程序，置于内存的特定位置。CPU 启动通道后，由通道执行通道程序控制外设进行 I/O 操作，CPU 转去执行其他任务，I/O 完成后通道向 CPU 发出中断信号，由 CPU 进行中断处理。与 DMA 方式相比，CPU 对 I/O 过程的干预更少，因此 CPU 与外设的并行度更高；但是，通道价格较高，从经济的角度出发不能过多使用。

3. 引入中断技术的原因有哪些？

【解答】在多任务计算机系统中，多个进程并发执行，各进程要完成自己的任务，必须获得 CPU 的控制权。在分时系统中，采用分时共享 CPU 方式对并发进程进行调度，使它们能公平地使用 CPU，从而完成各自的任务。各并发进程被轮流分配给一定的时间片，当某进程的时间片用完时，操作系统暂停该进程的运行，将处理机分配给下一个进程运行，该进程也运行一个时间片，依次类推，都轮完一遍后，再开始新一轮时间片轮转。要实现这种分时共享 CPU 的方式，关键是要准确把握各进程的运行时间，要能定时。可以采用硬件定时器来定时，预先设置定时器，当时间片用完时便向操作系统发送一个时钟中断信号，通知操作系统当前运行进程的时间片用完，操作系统便可调度另一个进程运行。此时采用了时钟中断技术。

在进程进行输入输出操作时，如果主机采用查询方式来等待慢速外设操作的完成，将会浪费 CPU 这一重要资源，降低计算机系统的工作效率。为了使处理机和外设能够并行工作，现代计算机系统都引入了中断技术。

计算机系统应具有能够自动处理各种事故的能力，例如电源故障、地址越界等。

计算机系统应具有能够响应随机突发事件的能力，如键盘操作、鼠标操作等。

由于以上的各种原因，在现代计算机系统中引入了中断技术。

4. 中断源通常有哪些？

【解答】计算机中的中断源种类有很多，大致可分为：

(1) 强迫性中断源

1) 硬件故障

如电源故障、主存储器故障，等等。

2) 程序性错误

由执行机器指令引起的错误，如除数为零、操作数溢出、非法指令、目态下使用特权指令、地址越界等等。

3) 外部事件

时钟中断、重新启动中断等。

4) I/O 中断事件

外设完成 I/O 操作或 I/O 操作出错，如打印完成、打印缺纸等。

(2) 自愿性中断源

指进程执行访管指令请求 OS 服务，如请求分配外设、请求 I/O 等。由访管指令引起的中断称为访管中断。

5. 引入缓冲技术的原因有哪些？

【解答】引入缓冲技术的原因有：

(1) 减少读块设备的次数

当进程在块设备上执行读操作（输入）时，通过系统调用，由操作系统将数据读入缓冲器，然后送入相应进程的数据存储区，由进程进行处理。缓冲器内保存的信息可能会被再次读到，因此，当再次对此块设备进行读操作时，可以先查看一下所读信息是否在缓冲器内，若在缓冲器内，则可以直接从缓冲器内读取，而不必从设备读取。这种场合要求对缓冲器的读取速度要快于对块设备的读取速度，则当命中率（所读数据在缓冲器内的次数

与读操作的总次数的比值)较高时,会明显提高读操作的速度。CPU 和内存之间的高速缓存的作用就是这样的。

(2) 减少对 CPU 的中断次数,放宽对中断响应时间的限制。

这是硬件缓冲器的作用。例如,在远程通信系统中,如果从远程终端发来的数据仅用一位缓冲来接收,则必须在每收到一位数据时便中断一次 CPU。这样,对于速率为 9.6Kb/s 的数据通信而言,就意味着其中断 CPU 的频率也为 9.6Kb/s,即每 100 微秒就中断 CPU 一次,而且 CPU 必须在 100 微秒内响应,否则缓冲区内的数据将被覆盖。若设置一个具有 8 位的缓冲寄存器,则可使 CPU 被中断的频率降低为原来的 1/8。

(3) 用于无法直接通信的设备间的中转站

如果在两个设备无法直接通信的设备之间传送数据,那么可以通过在内存中开辟缓冲区进行中转。如将从卡片输入机读取的数据存到磁盘上,可以在内存中开辟一个缓冲区,先将卡片内容输入到缓冲区,然后再将缓冲区中的内容传送到磁盘上。

(4) 解决程序所请求的逻辑记录大小和设备的物理记录大小不匹配的问题。

例如,当进程需要从磁盘读取数据时,因为磁盘是块设备,以数据块作为数据传输单位,所以一次必须传输整个磁盘块,而程序仅要求读一条逻辑记录,此逻辑记录包含在一盘块内,这时必须在内存中开辟一个缓冲区,其大小等于盘块大小,先将包含逻辑记录的盘块读到缓冲区中,然后进程再从缓冲区中提取所需的逻辑记录。

(5) 加快进程(作业)的推进速度

假设某进程的功能是每次读取一张卡片,然后对此卡片上的信息进行处理,循环进行。如果在卡片阅读机和进程存储区之间直接传输数据,则在输入时,进程只能等待,当一张卡片输入完毕后才能进行处理,然后输入下一张卡片,再进行处理,依次类推。而如果在卡片阅读机和进程存储区之间设置一个缓冲区,则输入时先将卡片信息输入到缓冲区中,然后将其复制到进程存储区,由进程对其进行处理,同时输入下一张卡片,如果两者速度能匹配,则可使进程能够流畅运行而不必等待,从而加快了进程的推进速度。

6. 单缓冲与双缓冲的区别是什么?

【解答】单缓冲方式由于只有一个缓冲区,这一缓冲区在某个时刻能存放输入数据或输出数据,但不能既是输入数据又是输出数据,否则在缓冲区中的数据会引起混乱。在单缓冲方式下解决输入及输出的情形是这样的:当数据输入到缓冲区时,输入设备在工作,而输出设备空闲;而当数据从缓冲区输出时,输出设备在工作,输入设备空闲。因此单缓冲方式不能使外部设备并行工作。

双缓冲方式可以实现外部设备并行工作。先将数据输入到缓冲区 buffer1,然后进程从 buffer1 中取出数据进行计算,将输出的数据送往 buffer2,然后将 buffer2 中的数据送往输出设备,与此同时,从输入设备将数据输入到 buffer1,如此继续进行,使得输入设备和输出设备并行工作。

7. 分别解释什么是独享设备和共享设备。

【解答】从设备分配的角度看,I/O 设备可以分为独占设备和共享设备两种。有的设备仅适应于某进程独占,有些设备可以被多进程共享。

独占设备:这类设备被分配给一个进程后,就被该进程独占使用,其他任何进程不能使用,直到该进程使用完毕(不再使用)主动释放为止。在进程占有独占设备期间,设备

不一定总在工作，但是，即使在设备空闲时，也不能被其他进程使用。例如打印机，如果两个进程交替使用，就会使得两个进程打印的内容混淆在一起无法辨别，所以必须独占使用打印机，以使得打印内容完整。独占设备是一种临界资源，对独占设备的竞争可能引起死锁。

共享设备：是指可以由多个进程交替使用的设备。如磁盘机。

8. 什么是设备独立性？为什么要引入设备独立性？

【解答】设备独立性又称为设备无关性，是指，当在应用程序中使用某类设备时，不直接指定具体使用哪个设备，而只指定使用哪类设备，由操作系统来为进程分配具体的一个该类设备。

当应用程序（进程）具体指定使用哪个设备时，如果该设备已经分配给其他进程或出了故障，而此时可能还有几台其他的同类设备可用，该进程却被阻塞。若进程只指定使用哪类设备，则可以从该类设备中任选一个分配给它使用，只有当所有此类设备已全部分配完毕时，进程才会阻塞。

设备无关性功能可以使应用程序的运行不依赖于特定设备是否完好、是否空闲，而由系统合理地进行分配，从而保证程序的顺利运行。

9. 什么是虚拟设备技术？

【解答】系统中独占设备的数量有限，且对独占设备的分配往往采用静态分配方式。这样做不利于提高系统效率。这些设备只能分配给一个作业，且在作业的整个运行期间一直占有该设备，直至作业结束时才被释放。而一个作业往往不能充分地利用该设备，在独占设备被某个作业占用期间，往往只有一部分时间在工作，而其余时间处于不工作的空闲状态。因此设备利用率低。而其他申请该设备的作业因得不到该设备而无法运行，降低了系统效率。

另一方面，独占设备往往是低速设备，因此，在作业执行过程中，由于要等待这类设备传输而大大延长了作业的执行时间。

为了克服独占设备的这些缺点，可以采用虚拟设备技术。即用一共享设备——高速存储设备（如高速磁盘机）上的存储区域模拟独占设备。

10. Spooling 系统采用了哪些技术？

【解答】Spooling 系统在硬件上采用了通道和中断机制，这是使 CPU 与外部设备充分并行工作的硬件基础；在软件上采用了多道进程并发执行（多道程序设计）技术和虚拟设备技术；多道进程并发执行是使 CPU 与外设并行工作的软件基础，虚拟设备技术提高了独占设备的利用率。

11. Spooling 系统由哪些部分构成？

【解答】（1）输入井和输出井

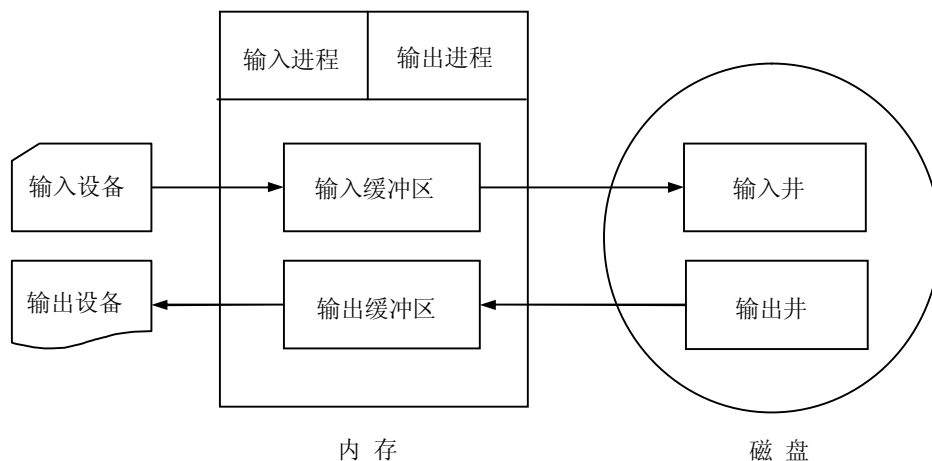
这是在高速磁盘上开辟的两个存储区域。此处采用了虚拟设备技术，用输入井虚拟低速输入设备，用输出井虚拟低速输出设备。输入井用于暂时存储从输入设备预输入的信息，输出井用于暂时存储要缓输出到输出设备的信息。利用虚拟设备技术，可以提高作业（进程）的执行速度，提高了 I/O 设备的利用率。

（2）输入缓冲区和输出缓冲区

I/O 输入设备无法直接和磁盘存储设备进行数据传输，必须经过内存的中转，因此对

应于输入和输出，要在内存中分别开辟输入缓冲区和输出缓冲区。

输入缓冲区用于输入设备和磁盘输入井之间的中转站，输出缓冲区用于磁盘输出井和输出设备之间的中转站。



(3) 输入进程和输出进程

输入进程模拟脱机输入时的卫星输入机，它将数据从输入设备经过输入缓冲区送到输入井中。当 CPU 需要读取数据时，直接从输入井中提取数据到内存。

输出进程模拟脱机输出时的卫星输出机，它将应用进程要输出的数据送到输出井，当输出设备空闲时，将输出井中的数据经过输出缓冲区送到输出设备输出。

12. I/O 控制过程包括哪几个步骤？

【解答】当用户进程通过系统调用提出 I/O 请求时，从系统响应此请求开始，至系统完成用户要求的 I/O 操作，并唤醒相应的等待 I/O 完成的进程为止，这整个过程称为 I/O 控制过程。

I/O 控制过程包括以下步骤：

(1) 响应 I/O 请求，为在具体物理设备上 I/O 操作做准备。包括将逻辑设备名转换为物理设备名（设备分配）、I/O 请求的合法性等。

(2) 设备驱动，控制设备完成 I/O 操作，对每类设备分别设置不同的设备驱动程序。

(3) 中断处理，I/O 操作完成之后，设备控制器向 CPU 发送中断信号，CPU 响应后转向相应的中断处理程序进行善后处理。

13. 访问磁盘的时间由哪几部分构成？

【解答】磁盘访问时间可以分为以下三部分。

(1) 寻道时间 T_s

这是把磁头移到指定磁道上用的时间。

(2) 旋转延迟 T_r

这是指定扇区旋转到磁头下经历的时间。

(3) 传输时间 T_t

这是指把数据从磁盘读出或向磁盘写入数据所经历的时间。由读写的字节数和磁盘旋转速度决定。

$$T_t = \frac{b}{rN}$$

其中， b 为一次读写的字节数， r 为磁盘每秒的转数， N 为每条磁道的字节数

对磁盘的访问时间为三部分时间之和： $T = T_s + T_r + T_t$ 。在这三个时间中，寻道时间和旋转延迟与读写的字节数无关，寻道时间所占比例最大。

14. 磁盘调度算法有哪些？它们之间有何区别？

【解答】常用的磁盘调度算法有先来先服务算法、最短寻道时间优先算法和扫描算法。

（1）先来先服务算法

这种调度算法按进程请求访问磁盘的时间先后次序进行调度。此算法的优点是实现简单，且公平，每个进程的磁盘 I/O 请求都能依次得到处理，不会出现某一进程的请求长时间得不到满足的情况。缺点是未对寻道进行优化，平均寻道时间较长。

（2）最短寻道时间优先算法

该算法选择这样的磁盘 I/O 请求，其要访问的磁道与当前磁头所在的磁道距离最近，以使每次的寻道时间最短。此算法只从当前角度考虑，没有考虑全局，表面看来平均寻道时间应该最短，但是不一定。

最短寻道时间优先算法可能导致某个进程发生“饥饿”现象。因为，只要不断有新进程的请求到达，且其所要访问的磁道与磁头当前所在磁道的距离最近，这种新进程的 I/O 请求必须优先满足，这样就会使某些进程的磁盘 I/O 请求长时间得不到满足。

（3）扫描算法

为防止出现最短寻道时间优先调度算法的“饥饿”现象，对最短寻道时间优先调度算法进行了改进，提出了扫描算法。

扫描算法不仅考虑到要访问的磁道与当前磁道间的距离，更优先考虑的是磁头当前的移动方向。例如，当磁头正在从里向外移动时，扫描算法选择的下一个访问对象，是在当前磁头所在磁道之外的距离当前磁道最近的磁道。这样从里向外地访问，直至再无更外的磁道需要访问时，磁头才返回从外向里移动，同样每次也是选择当前磁道之内的距离最近的磁道，到头后再返回，从里向外移动访问。由于此算法中磁头移动的规律很象电梯的运行，因此又称为电梯调度算法。

（4）循环扫描算法

扫描算法能获得较好的寻道性能，且能防止“饥饿”现象，所以被广泛应用于大、中、小型计算机和网络中的磁盘调度。但是存在一个问题：当磁头刚从里向外移动而越过了某一磁道时，恰好又有一个进程请求访问此磁道，这时，该进程必须等待，待磁头继续从里向外，然后再从外向里扫描完所有要访问的磁道后，才处理该进程的请求，致使该进程的请求被大大推迟。为了减小这种延迟，引入循环扫描算法，规定磁头单向移动。例如，只从里向外移动，当磁头移到最外的磁道并访问后，磁头立即返回到最里的要访问的磁道，再从里向外移动。

15. Linux 系统中磁盘高速缓存的原理是什么？

【解答】对文件系统的一切存取操作，都能通过直接从磁盘上读或往磁盘上写来实现，但磁盘 I/O 的速度较慢，因此使系统性能较低。为了减少对磁盘的存取频率，Linux 使用了磁盘高速缓存技术。

Linux 磁盘缓冲管理策略是试图把尽可能多的有用的数据保存在缓冲区中。磁盘缓冲管理模块位于文件系统与块设备驱动程序之间。

当从磁盘中读数据时，文件系统先从磁盘高速缓存中读，如果数据已在高速缓存中，则可以不必启动磁盘 I/O，如果数据不在高速缓存中，则启动磁盘 I/O，从磁盘读取数据送往高速缓存，进程再从高速缓存中读取数据。

当进程往磁盘上写数据时，先往高速缓存中写，以便随后又读它时，能从高速缓存中读取，而不必启动磁盘读取。Linux 采取了“延迟写”策略，即：如果缓冲区还没有写满，则不急于把缓冲区的内容写到磁盘上，而是在缓冲管理数据结构中对该缓冲区设置延迟写标志，当高速缓存中的数据延迟到必须往磁盘上写的时候才进行写盘操作。

16. Linux 把外部设备分为哪几类？它们的物理特性有何不同？它们的作用有何不同？

【解答】Linux 系统把设备分为两类：

(1) 块设备。用于存储信息，它对信息的存取是以信息块为单位的，如通常使用的磁盘、磁带等。

(2) 字符设备。通常用于输入输出，作为人和计算机之间的接口，它对信息的存取是以字符为单位进行的，如键盘、鼠标、显示器、打印机等。

17. Linux 设备管理的主要特点是什么？

【解答】Linux 系统把 I/O 设备看作文件，称为特别文件。例如，打印机的文件名为 lp，控制台终端的文件名是 console。这些特别文件组织在目录/dev 下。如果要访问打印机就可以使用路径名/dev/lp。

在高级编程语言中，可以使用以下语句打开特别文件/dev/lp。

```
fd=open("/dev/lp",O-WRONLY);
```

fd 为打开文件号，/dev/lp 为打开文件名，O-WRONLY 为打开方式。

Linux 的这一特征使得任何外部设备在用户面前与普通文件完全一样，而不必涉及它的物理特性，这给用户带来极大的方便。在文件系统内部，外部设备和普通文件一样受到存取控制的保护，仅仅在最终驱动设备时才转向各个设备的驱动程序。

块设备上的普通文件和目录文件并不是设备特殊文件，但块设备本身仍可以作为特殊文件来访问。