磁盘臂的位置，为了执行寻道，它发出一系列脉冲给磁盘臂电机，每个柱面一个脉冲，这样将磁盘臂移到新的柱面。当磁盘臂移到其目标位置时，控制器从下一个扇区的前导码中读出实际的柱面号。如果磁盘臂在错误的位置上，则发生寻道错误。

大多数硬盘控制器可以自动纠正寻道错误，但是大多数软盘控制器（包括Pentium的）只是设置一个错误标志位而把余下的工作留给驱动程序。驱动程序对这一错误的处理办法是发出一个recalibrate（重新校准）命令，让磁盘臂尽可能地向最外面移动，并将控制器内部的当前柱面重置为0。通常这样就可以解决问题了。如果还不行，则只好修理驱动器。

正如我们已经看到的，控制器实际是一个专用的小计算机，它有软件、变量、缓冲区，偶尔还出现故障。有时一个不寻常的事件序列，例如一个驱动器发生中断的同时另一个驱动器发出recalibrate命令，就可能引发一个故障，导致控制器陷入一个循环或失去对正在做的工作的跟踪。控制器的设计者通常考虑到最坏的情形，在芯片上提供了一个引脚，当该引脚被置起时，迫使控制器忘记它正在做的任何事情并且将其自身复位。如果其他方法都失败了，磁盘驱动程序可以设置一个控制位以触发该信号，将控制器复位。如果还不成功，驱动程序所能做的就是打印一条消息并且放弃。

重新校准一块磁盘会发出古怪的噪音，但是正常工作时并不让人烦扰。然而，存在这样一种情形，对于具有实时约束的系统而言重新校准是一个严重的问题。当从硬盘播放视频时，或者当将文件从硬盘烧录到CD-ROM上时，来自硬盘的位流以均匀的速率到达是必需的。在这样的情况下，重新校准会在位流中插入间隙，因此是不可接受的。称为AV盘（Audio Visual disk，音视盘）的特殊驱动器永远不会重新校准，因而可用于这样的应用。

5.4.5 稳定存储器

正如我们已经看到的，磁盘有时会出现错误。好扇区可能突然变成坏扇区，整个驱动器也可能出乎意料地死掉。RAID可以对几个扇区出错或者整个驱动器崩溃提供保护。然而，RAID首先不能对将坏数据写下的写错误提供保护，并且也不能对写操作期间的崩溃提供保护，这样就会破坏原始数据而不能以更新的数据替换它们。

对于某些应用而言，决不丢失或破坏数据是绝对必要的，即使面临磁盘和CPU错误也是如此。理想的情况是，磁盘应该始终没有错误地工作。但是，这是做不到的。所能够做到的是，一个磁盘子系统具有如下特性：当一个写命令发给它时，磁盘要么正确地写数据，要么什么也不做，让现有的数据完整无缺地留下。这样的系统称为*稳定存储器*（stable storage），并且是在软件中实现的（Lampson和Sturgis，1979）。目标是不惜一切代价保持磁盘的一致性。下面我们将描述这种最初思想的一个微小的变体。

在描述算法之前，重要的是对于可能发生的错误有一个清晰的模型。该模型假设在磁盘写一个块（一个或多个扇区）时，写操作要么是正确的，要么是错误的，并且该错误可以在随后的读操作中通过检查ECC域的值检测出来。原则上，保证错误检测是根本不可能的，这是因为，假如使用一个16字节的ECC域保护一个512字节的扇区，那么存在着2^{4096}个数据值而仅有2^{144}个ECC值。因此，如果一个块在写操作期间出现错误但是ECC没有出错，那么存在着几亿亿个错误的组合可以产生相同的ECC。如果某些这样的错误出现，则错误不会被检测到。大体上，随机数据具有正确的16字节ECC的概率大约是2^{-144}。该概率值足够小以至于我们可以视其为零，尽管它实际上并不为零。

该模型还假设一个被正确写入的扇区可能会自发地变坏并且变得不可读。然而，该假设是：这样的事件非常少见，以至于在合理的时间间隔内（例如1天）让相同的扇区在第二个（独立的）驱动器上变坏的概率小到可以忽略的程度。

该模型还假设CPU可能出故障，在这样的情况下只能停机。在出现故障的时刻任何处于进行中的磁盘写操作也会停止，导致不正确的数据写在一个扇区中并且后来可能会检测到不正确的ECC。在所有这些情况下，稳定存储器就写操作而言可以提供100%的可靠性，要么就正确地工作，要么就让旧的数据原封不动。当然，它不能对物理灾难提供保护，例如，发生地震，计算机跌落100m掉入一个裂缝并且陷入沸腾的岩浆池中，在这样的情况下用软件将其恢复是勉为其难的。

稳定存储器使用一对完全相同的磁盘，对应的块一同工作以形成一个无差错的块。当不存在错误时，在两个驱动器上对应的块是相同的，读取任意一个都可以得到相同的结果。为了达到这一目的，定义了