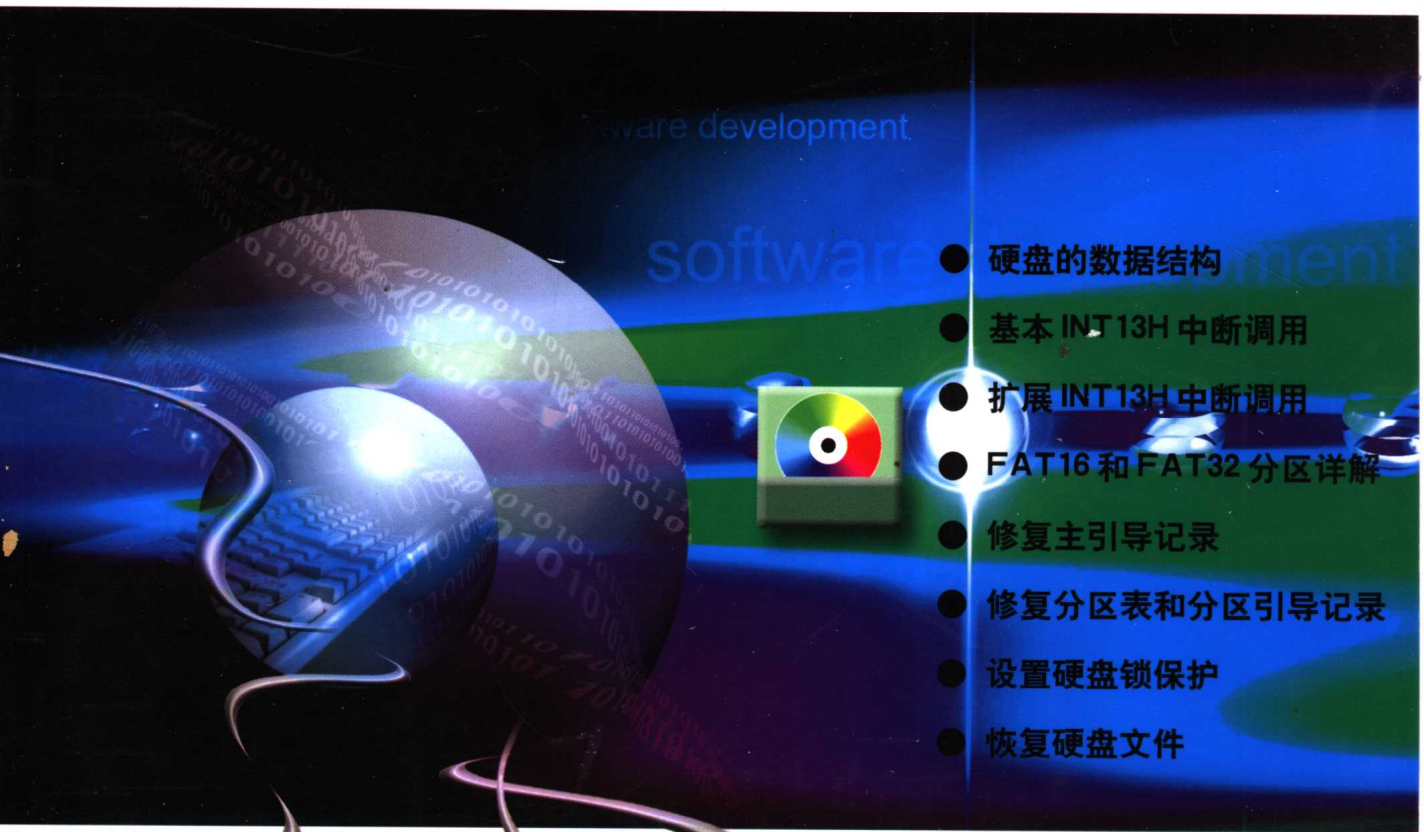
信息科学与技术丛书

程序设计系列

● 宋群生　宋亚琼　编著

硬盘扇区读写技术
—修复硬盘与恢复文件



机械工业出版社
CHINA MACHINE PRESS

硬盘扇区读写技术

——修复硬盘与恢复文件

信息科学与技术丛书
程序设计系列

硬盘扇区读写技术

——修复硬盘与恢复文件

宋群生　宋亚琼　编著

机械工业出版社

本书主要内容包括对硬盘物理扇区的读写技术；使用该技术编写的 40 多个工具程序；使用工具程序修复硬盘和恢复文件的方法。全书分三篇，共计 37 章。第 1 章至第 8 章是“基础篇”，介绍了有关硬盘和两种 FAT 文件系统的基础知识；第 9 章至第 31 章是“工具篇”，介绍了对硬盘扇区进行各种操作的工具程序；第 32 章至第 37 章是“应用篇”，介绍了硬盘扇区读写技术和工具程序的典型应用范例。

本书附送的光盘收录了工具程序的全部编译文件。书中对这些工具程序的运行和应用，进行了详细介绍。读者可以按照书中介绍的方法，使用这些工具程序对硬盘进行各种操作。

本书既可供编程人员参考，也可供不懂编程，但需要对硬盘进行维护的人员参考。

图书在版编目（CIP）数据

硬盘扇区读写技术——修复硬盘与恢复文件 / 宋群生，宋亚琼编著. —北京：机械工业出版社，2004.4
（信息科学与技术丛书　程序设计系列）
ISBN 7-111-14284-5

Ⅰ. 硬…　Ⅱ. ①宋…②宋…　Ⅲ. 磁盘存储器—修复—基本知识
Ⅳ. TP333.3

中国版本图书馆 CIP 数据核字（2004）第 026904 号

机械工业出版社（北京市百万庄大街 22 号　邮政编码 100037）
策　　划：胡毓坚
责任编辑：孙　业
责任印制：李　妍

北京机工印刷厂印刷 · 新华书店北京发行所发行
2004 年 4 月第 1 版 · 第 1 次印刷
787mm×1092mm 1/16 · 23.25 印张 · 574 千字
0 001—5 000 册
定价：40.00 元（含 1CD）

出 版 说 明

随着信息科学与技术的迅速发展，人类每时每刻都会面对层出不穷的新技术、新概念。毫无疑问，在节奏越来越快的工作和生活中，人们需要通过阅读和学习大量信息丰富、具备实践指导意义的图书，来获取新知识和新技能，从而不断提高自身素质，紧跟信息化时代发展的步伐。

众所周知，在计算机硬件方面，高性价比的解决方案和新型技术的应用一直备受青睐；在软件技术方面，随着计算机软件的规模和复杂性与日俱增，软件技术受到不断挑战，人们一直在为寻求更先进的软件技术而奋斗不止。目前，计算机在社会生活中日益普及，随着因特网延伸到人类世界的层层面面，掌握计算机网络技术和理论已成为大众的文化需求。也正是由于信息科学与技术在电工、电子、通信、工业控制、智能建筑、工业产品设计与制造等专业领域中已经得到充分、广泛的应用，所以这些专业领域中的研究人员和工程技术人员将越来越迫切需要汲取自身领域信息化所带来的新理念和新方法。

针对人们对了解和掌握新知识、新技能的热切期待，以及由此促成的人们对语言简洁、内容充实、融合实践经验的图书迫切需要的现状，机械工业出版社适时推出了“信息科学与技术丛书”。这套丛书涉及计算机软件、硬件、网络、工程应用等内容，注重理论与实践相结合，内容实用，层次分明，语言流畅，是信息科学与技术领域专业人员不可或缺的图书。

现今，信息科学与技术的发展可谓一日千里，机械工业出版社欢迎从事信息技术方面工作的科研人员、工程技术人员积极参与我们的工作，为推进我国的信息化建设作出贡献。

机械工业出版社

前　言

自从 1956 年蓝色巨人 IBM 发明硬盘以来，它一直是计算机系统中存储数据的主要设备。在计算机技术高速发展的今天，还找不到更好的存储设备来替代硬盘。据业内权威人士预测，十年以内不可能有替代产品。这就是说，在未来很长一段时间里，凡是使用计算机的人就必须和硬盘打交道。

计算机在正常工作时，是按照“逻辑盘”对硬盘进行数据存取的，这就是我们通常说的 C 盘、D 盘……。这种数据存取的前提条件是硬盘要有正常的主引导记录、分区表、分区引导记录、文件分配表、文件目录表。如果其中有一项不正常，计算机就不能正常工作。然而硬盘在使用中出现不正常的情况是经常发生的，如遭到病毒攻击、使用者误操作、人为破坏等等。这时有没有办法能够挽救硬盘，恢复数据，将损失尽可能减小呢？答案是肯定的，这就是本书要介绍的“硬盘扇区读写技术及硬盘修复和恢复文件的方法”。

目前关于计算机基础教程和介绍软件使用方法的图书可谓百花齐放，但关于硬盘扇区读写技术和应用的书籍却很少。本书提供了笔者编写的 20 多个类别，40 多个读写扇区的工具程序，并对其使用方法进行了详细的分析和说明。对基本 INT13H 中断调用和扩展 INT13H 中断调用；对 C 和汇编的混合编程；对硬盘主引导记录、分区表、分区引导记录的修复；对从不能启动的硬盘上恢复文件的方法；对设置硬盘锁保护；对 FAT16 和 FAT32 两种文件系统，都从编程和应用两个方面进行了深入剖析。

如果读者具有 C 语言和汇编语言基础，就可以阅读本书的全部内容。如果读者没有编程基础，可以只阅读过程类和应用类的内容。

为了照顾那些不会编程，但又需要对硬盘进行修复和从硬盘恢复文件的读者，笔者对过程和应用方面的内容进行了详细的分析和说明。工具程序的目标文件和编译过程中生成的中间文件，都收录在随书附送的光盘中，任何人都可以使用。

硬盘数据的丢失，有时会造成不可弥补的损失，丢失数据的代价远远超过硬盘本身的价值。在这种情况下使用本书介绍的工具程序，进行修复硬盘或恢复文件的操作，有时能收到意想不到的效果。

读者如果掌握了本书的内容，那么硬盘对读者来说，就好像一本打开的书。硬盘的每一个扇区就好像书的每一页，任读者翻阅。

目前在“硬盘扇区读写技术及应用”的研究方面，可供借鉴的资料很少，再加之笔者水平不高，书中难免出现错误，敬请读者谅解并给予指正。读者可以将意见或建议发到笔者的电子邮箱：jnsqsheng@163.com中与笔者进行讨论。

笔　者

目　录

应 用 篇

基 础 篇

“基础篇”的内容，是学习硬盘物理扇区读写编程技术的预备知识，是下一步学习“工具篇”和“应用篇”必不可少的前期准备课程。这些预备知识涉及的范围很广，其每一部分都属于相对独立的研究领域。如果对这些预备知识进行详细的介绍，将会占用本书的很大篇幅，也会消耗读者的很多时间和精力。本书的主要内容，一是介绍硬盘物理扇区读写编程技术，二是介绍使用工具程序修复硬盘和恢复数据。为了突出这两个重点，对一些预备知识进行了最大限度的精简。

本书不是学习编程语言的基础教材，而是介绍如何使用基本的编程知识，去开发一些实用的工具程序和处理一些硬盘的技术故障。所以要全面了解“基础篇”内容的读者，应该具备一定的 C 语言和汇编语言基础。

为了照顾那些没有编程基础的读者，笔者特作如下建议：

（1）“基础篇”中有关编程的内容可以不看，只看有关硬盘和文件系统的一些技术名称和解释，以期能大体了解硬盘的数据结构。

（2）了解了硬盘的各个组成部分之后，转去学习本书的第二部分“工具篇”。“工具篇”中所有的工具程序都可以直接拿来使用，不懂编程的人也可以做到。

（3）不懂编程的技术人员可以不看程序源代码，只学习有关过程的操作方法，等有了一定的编程基础以后再学习全部内容。

（4）暂时还不会编程的读者，在学习过程类内容时，可以逐步进入到编程领域。其实学习编程并不难，不要将编程看得很神秘，认为高不可攀。在笔者看来，编程其实是一项技能，一门艺术，只不过其入门的门槛比较高罢了。

关于本书使用的编程语言，读者可能会提出疑问，为什么非要涉及到两种编程语言呢？这是由以下原因决定的：

（1）读写硬盘的物理扇区，关键的技术问题是调用 INT13H 中断。INT13H 中断有两种类型，一种是较早期的基本 INT13H 中断，另一种是后期以及现在使用最多的扩展 INT13H 中断。而扩展 INT13H 中断的调用规范，是在汇编语言里定义的。据笔者所知，到目前为止，还没有哪一种高级语言能够支持对扩展 INT13H 中断的调用，所以汇编语言是必须使用的编程语言。

（2）如果书中全部工具程序都使用汇编语言来写，那冗长的源程序代码足以让每一个编程人员感到头痛。以本书目前的篇幅，恐怕光书写源程序代码，就要占满全部空间了。因此笔者在编写工具程序和应用程序时，除了必需使用汇编语言编写的程序内容外，其他部分都使用 C 语言来写。

学习“基础篇”的读者，还应该对 FAT16 和 FAT32 两种文件系统有所了解。本书虽然也重点介绍了这两种文件系统，但介绍的内容是从编程需要的角度进行编排的。一些很基本

的操作过程，如怎样对硬盘进行分区、如何格式化逻辑驱动器等内容，由于篇幅所限没有收录进来。

读写硬盘物理扇区的数据，是通过调用主板 BIOS 的磁盘服务程序进行的，这种读写操作不用考虑硬盘上安装的是什么操作系统。但是要修复硬盘以及从硬盘上恢复文件，却必须知道被操作的这部分扇区属于哪一种文件系统。因为不同的文件系统，其存储文件数据的规律是不一样的。在当前的主流文件系统中，FAT16 和 FAT32 使用最为普遍，所以在“基础篇”中详细介绍了这两种文件系统。

除了 FAT16 和 FAT32，当前广泛使用的其他主流文件系统还有 NTFS 和供 Linux 操作系统使用的 ext2。据有关媒体介绍，微软公司在下一代操作系统中，又开发使用了新的文件数据系统，该操作系统预计在 2005 年发布。

只要掌握了本书的基本内容，不管硬盘使用什么样的文件系统，读者都可以在获取了该文件系统的技术资料以后，使用本书介绍的工具程序自行分析研究。因为本书介绍的工具程序所操作的对象是物理硬盘，只要知道了某一个文件系统的数据存取原理，就可以使用这些工具程序进行系统的修复和数据的恢复。

第 1 章　硬盘的物理结构

硬盘的硬件组成主要有两大部分：

（1）磁头-盘片组件 HAD（HEAD DISK ASSEMBLY），其中包括读写磁头、磁头驱动机构、磁盘片、盘片驱动电动机等。

（2）印制电路板组件 PCBA（PRINTED CIRCUIT BOARD ASSEMBLY），其中包括电阻、电容、半导体集成电路等元器件，负责对各种信号进行处理。

因本书是以讲编程及其应用为主的，所以硬盘的硬件组成不是本书的重点内容，读者若想了解更详细的硬件知识，请参阅其他书籍。

硬盘是依靠磁盘片上的磁性介质记录信息的，对盘片表面的磁介质必须进行某些有序的磁化处理，使这些磁介质能够符合进行数据存取的要求。对磁介质的处理分三个过程，就是通常所说的低级格式化（也称物理格式化以下简称低格）、分区和高级格式化（也称逻辑格式化）。

1.1　低级格式化

对于一块新硬盘，这个过程已经由生产厂家在产品出厂前完成了。低级格式化对磁介质的读写功能有一定影响，所以使用者一般不要进行该项操作。除非你的硬盘出现了坏道，不得不进行低格操作。

低级格式化的目的，是将盘面划分成磁道、扇区和柱面。

下面分别介绍磁盘的各组成部分。

1．磁道

硬盘加电正常工作以后，磁盘片由主轴电动机带动高速旋转，磁头在驱动机构的作用下沿盘片径向移动。当磁头停在一个位置时，盘片旋转一周，磁头就在盘片表面画出一个圆形轨迹，这个圆形轨迹称为磁道。随着磁头的径向移动，磁盘就被画出许多封闭的同心圆形磁道。磁道从盘片外缘开始编号，起始号为0。

2．扇区

磁盘上的每个磁道被划分成许多弧段，弧段之间有间隔，这些弧段称为扇区。扇区也进行编号，起始号为1。每个磁道分成63个扇区，每个扇区可存储的数据是512个字节。实际上现在的大容量硬盘，为了提高磁介质的存储利用率，使用了等密度的存储方式，也就是说外圈磁道的扇区数比内圈磁道多。为了与老的制式兼容，由硬盘控制器的驱动程序将参数进行转换。

3．柱面

硬盘一般由多张重叠的盘片组成，每个盘面都被划分成数目相等的磁道。具有相同编号的磁道形成一个圆柱，这就是硬盘的柱面。每个盘面上有多少个磁道，也就有多少个柱面，柱面的编号和磁道编号相同。

4．磁头

一张磁盘有两个盘面，每个盘面都有一个读写磁头。将盘面进行编号，起始号为0，磁头的编号和盘面的编号相同。

目前对扇区寻址有两种称呼，为了今后分析程序时不至于混淆，需要预先对这两种称呼明确一下。以硬盘主引导记录存储的扇区为例，一种叫法称作“0面0头1扇区”，这里的“0面”指的是柱面，“0头”指的是磁头。另一种叫法称作“0道0面1扇区”，这里的“0道”指的是磁道，“0面”指的是盘面。这两种叫法在使用时很容易产生混淆，混淆的根本原因是“面”的含义不一样。前者的“面”与后者的“道”编号相同，而后者的“面”与前者的“头”编号相同。

在本书中统一使用前一种叫法，因为在对扇区编号进行计算的过程中，是沿着“柱面－磁头－扇区”进行的，这就是物理扇区的CHS（柱面数、磁头数、每磁道扇区数）寻址方式。

有了CHS参数，很容易计算硬盘的容量。只要将这三个数相乘，就可得到总的扇区数，再乘上512，就是硬盘的容量。

1.2 分区

硬盘在使用时，是按照不同的区域存储数据的，硬盘分区就是划分区域的过程。划分好的每一个区域都称作一个分区，最多可划分为四个分区。这项工作由分区程序来完成，通常使用FDISK。

有时也可以将硬盘只分成一个分区C，但这和没有分区的硬盘是不一样的。因为只有经过分区，才能使硬盘的管理系统知道这块硬盘有哪些区域可以使用。一般情况下是将硬盘分成一个主分区C和一个扩展分区，激活主分区准备安装操作系统。然后再将扩展分区分成若干个逻辑驱动器，符号依次为D、E、F……。通常说的盘符如E:\，指的就是逻辑驱动器E，主分区C也是一个逻辑驱动器。

在分区的过程中，分区程序向0柱面0磁头1扇区写入主引导记录MBR(MASTER BOOT

RECORD）和分区表记录 DPT（DISK PARTITION TABLE），并建立一个分区表链，向所有的逻辑驱动器写入链表记录。

硬盘的分区格式有很多种，选用什么样的分区格式是由选用哪一种操作系统所决定的。以目前流行的操作系统来说，常用的分区格式有四种，分别是 FAT16，FAT32，NTFS 和 Linux。在这四种分区格式中，使用最多的是 FAT16 和 FAT32。特别是 FAT16，因为它能被目前所有的操作系统所识别，因此用的最多。

本书讨论的“硬盘物理扇区读写编程技术”，是以 FAT16 和 FAT32 这两种分区格式作为操作对象来进行分析的。

1.3 高级格式化

硬盘分区以后还不能直接使用，要在每个分区内建立起完整的存储系统才能正常使用。建立存储系统的工作由 FORMAT 程序来完成，这个过程称作高级格式化。高级格式化的目的是在分区内建立分区引导记录 DBR（DOS BOOT RECORD）、文件分配表 FAT（FILE ALLOCATION TABLE）、文件目录表 FDT（FILE DIRECTORY TABLE）和数据区 DATA。

第 2 章　硬盘的数据结构

硬盘只有建立起完整的数据结构，才能正常使用。数据结构由六部分组成，分别是主引导记录、主分区表和分区表链、分区引导记录、文件分配表、文件目录表以及数据区。了解硬盘的数据结构，是对硬盘物理扇区进行读写编程的基础。

特别是本书介绍的通过读取扇区数据来恢复文件的方法，已经完全不用考虑被操作硬盘安装的是什么操作系统，甚至不用考虑硬盘上还有没有操作系统。因为这种恢复文件的方法，是通过调用 BIOS 磁盘服务程序来完成的，而 BIOS 对硬盘的管理级别高于所有的操作系统。但是这种操作必须了解有关的数据结构和文件的存储方式，才能按照文件的存储规律将它们恢复出来。因此，了解硬盘的数据结构，是学习本书编程技术必须具备的重要基础知识之一。

从本章开始要为以后学习物理扇区读写编程做一些必要的准备，介绍一些必备的常识。这期间为了对物理扇区寻址，经常要进行繁琐的数学运算。为了使运算过程更简明、直观，容易被读者理解，笔者以自己使用的一块 IBM 生产的 18G 硬盘作为标本，为全书所有的数学运算和编程提供参照。

笔者使用的硬盘分为一个主分区 C 和一个扩展分区，扩展分区又分成 D、E、F、G 四个逻辑驱动器，C 也是一个逻辑驱动器。从 C 到 F 四个逻辑驱动器的容量都是 2047M，采用 FAT16 分区格式，逻辑驱动器 G 的容量是 9170M，采用 FAT32 分区格式。这块硬盘虽然容量小点，但因为是 SCSI 硬盘，所以速度很快。硬盘上安装了两个操作系统，C 盘安装 WINDOWS 98 第二版；D 盘安装 WINDOWS 2000 PROFESSIONAL，采用双启动方式安装。

通常将逻辑驱动器看作一个独立的盘，所以笔者硬盘中的五个逻辑驱动器，今后在叙述中就简称为 C 盘、D 盘、E 盘、F 盘和 G 盘。

2.1　主引导记录

硬盘的主引导记录也称 MBR，位于 0 柱面 0 磁头 1 扇区。该扇区的 512 个字节有三部分内容，除了主引导记录外，还有分区表和结束标志 55 AA。

主引导记录的作用非常重要，它是硬盘启动时最先加载的扇区数据。下面通过分析硬盘的启动过程，来说明它的重要性：

1．计算机系统接通电源以后，主板 BIOS 加电进行自检。自检的内容很多，是一个很复杂的过程，这里只介绍与硬盘有关的部分。

2．将硬盘第一个扇区，也就是 0 柱面 0 磁头 1 扇区读入内存。

3．检查结束标志，也就是扇区最后两个字节的值是否等于 aa55H（存储顺序是低字节在前，高字节在后）。若不等则打印屏幕提示，然后死机。

4．执行主引导记录中的程序，将控制权转交给主引导程序。

5．主引导程序首先将自己读入内存，然后查找在分区表中是否有活动分区。找到活动分区以后，将分区引导记录读入内存。

6．检查结束标志是否等于 aa55H，然后执行分区引导记录中的启动程序，将控制权交给操作系统。

7．操作系统加载系统文件，计算机启动。

通过对以上过程的分析可以看出，如果主引导记录不正常，后面所有的启动过程都不可能正常执行。

有一种特殊情况，使计算机启动过程的前两步与上面介绍的不一样。如果硬盘上安装了多系统引导软件，如 PartitionMagician 分区软件，则该软件将主引导记录替换成自己的一段程序。这段程序将 BIOS 引向软件设置的专用分区，然后根据操作者的选择激活某一个分区，再进入正常的启动过程。类似 PartitionMagician 这样的分区软件还有很多，它们各有自己的特点，这些都不在本书的讨论范围之内。

通常情况下，一块硬盘上只有一个主引导记录。

主引导记录扇区所在的磁道，通常被称为 0 磁道，它属于隐藏磁道，这个磁道的 63 个扇区属于隐藏扇区。操作系统的所有命令，除了 FDISK 以外都不能访问它们。就连格式化程序 FORMAT，对它们也无能为力。

正因为如此，0 磁道的 63 个扇区就成了一些病毒程序代码、操作系统的引导代码、应用软件用于自我保护的识别标记、BIOS 功能扩展程序代码的栖息之地。

由于 INT13H 中断能调用 BIOS 磁盘服务程序，直接对硬盘物理扇区进行操作，并且这种操作与分区无关。所以要想读写硬盘 0 磁道的某个扇区，在程序中使用 INT13H 中断调用功能即可达到目的。

使用调试程序 DEBUG，通过 INT13H 中断调用可以读出这些扇区内容，DEBUG 程序应在 DOS 实模式下执行。

所有对 INT13H 中断的调用都应在 DOS 实模式下运行，因为 WINDOWS 95 以上的操作系统都是保护模式，对 INT13H 中断调用的支持不尽相同。

举例来说，WINDOWS 95 不支持 INT13H 中断调用；而 WINDOWS 98 在 DOS 窗口下支持 INT13H 中断调用；WINDOWS 2000 不支持 INT13H 中断调用，但可以使用 API 函数 CreateFile 对物理扇区进行读写。

为了将程序的编译和运行环境统一起来，本书中所有程序统一使用 DOS 实模式，可以是 DOS 6.22、DOS 7.0 和 DOS 7.1。

本书介绍的工具程序和应用范例，是指硬盘处于不正常状态时，如何利用硬盘物理扇区读写技术修复数据结构和恢复文件。在这种情况下，是不可能启动被修复硬盘上的 WINDOWS 系统的。

在硬盘不能启动的情况下，可以有两种操作模式。第一种模式是用软盘启动，在软盘上运行有关的工具程序，对硬盘扇区进行读写操作。这种操作模式简便易行，但程序的运行速度比较慢。第二种模式是用一块正常的硬盘启动，运行有关的工具程序，这块硬盘设置为第一硬盘。将不能启动的硬盘设置为第二硬盘，在 INT13H 中断调用的入口参数中，把有关寄存器的参数设定为 81H，就能对有问题的硬盘扇区进行读写操作。

以下是用 DEBUG 读取笔者硬盘主引导记录的过程，关于 INT13H 中断的使用方法，在后面有关章节和附录中再详细说明。