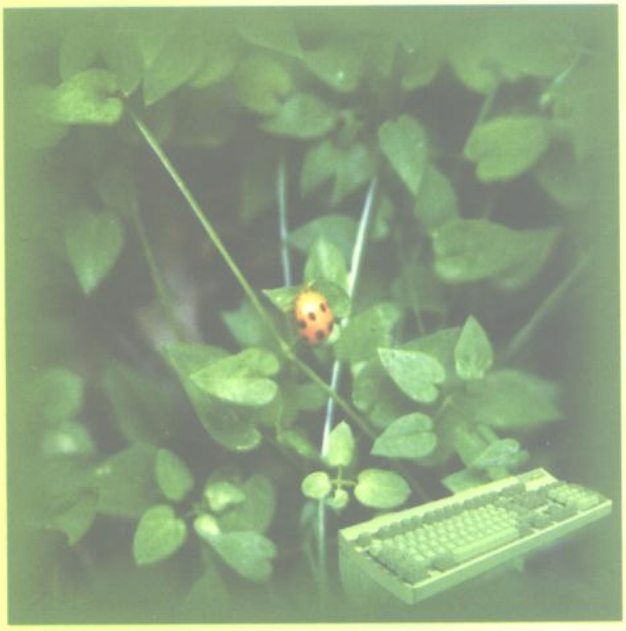
Operating System

操作系统及其使用技巧(上)



计算机应用技术精华丛书

程吉宽·史新元 主编
邓露林 副主编

本丛书是由首届全国计算机操作、编程、应用、维修有奖征文大赛的精华集粹而成。它突出实用性、启迪性，能起到举一反三，触类旁通的作用，是奉献给读者的“融理论与实践于一炉”的高质量的计算机应用技术丛书。

電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
URL: http://www.phei.com.cn

计算机应用技术精华丛书

操作系统及其使用技巧

（上）

程吉宽　史新元　主　编
邓露林　副主编

電子工業出版社
Publishing House of Electronics Industry

内 容 提 要

本书是由电子工业出版社、中国计算机报社、中国计算机用户杂志社、中国电脑教育报社、软件世界杂志社共同举办的“全国首届计算机操作、编程、应用、维修征文大奖赛”操作系统类优秀论文汇集而成。书中汇集了DOS、UNIX等操作平台下的各类精选文章，每一篇文章均是作者实际工作的经验总结。该书内容丰富，技术实用，对广大计算机爱好者及计算机使用人员均有极大帮助。

计算机应用技术精华丛书

书　　名：操作系统及其使用技巧(上)

主　　编：程吉宽　史新元

副 主 编：邓露林

责任编辑：李新社

印 刷 者：北京李史山印刷厂

装 订 者：三河市万和装订厂

出版发行：电子工业出版社出版、发行　URL:http://www.phei.com.cn

北京市海淀区万寿路173信箱　邮编100036　发行部电话:68214070

经　　销：各地新华书店经销

开　　本：787×1092 1/16　印张:13.25　字数:339.2千字

版　　次：1998年3月第1版　1998年3月第1次印刷

书　　号：ISBN 7-5053-4114-6
TP·1814

定　　价：17.00元

凡购买电子工业出版社的图书，如有缺页、倒页、脱页者，本社发行部负责调换

普及计算机技术

推广计算机应用

孙俊人

一九九七年七月三日

中国电子学会理事长孙俊人工程院士的题词

编委会名单

主　任：梁祥丰

副主任：吴金生　胡毓坚　李新社　史新元

委　员：王　仪　王仲文　文宏武　邓露林　史新元　李　颖
李明树　李新社　杜国梁　宋瀚涛　许　远　赵　平
赵丽松　武　航　张务谦　张春晖　张祖荫　张　欣
吕再峰　杨丽娟　施玉新　胡毓坚　郭　立　高　平
梁祥丰　秦　梅　徐三南　龚兰方　程吉宽

评委会名单

主　任：李超云

副主任：李　颖　王明君　孙毓林　徐三南

委　员：王　仪　王仲文　王明君　文宏武　李明树　李超云
许　远　孙毓林　武　航　苏子栋　张务谦　张春晖
张祖荫　吕再峰　宋瀚涛　吴金生　杜国梁　高　平
秦　梅　徐三南　龚兰方　程吉宽

获奖名单

二等奖

MS－DOS6.22 内存管理技术	王长军
UNIX 系统 Boot 区的保存与恢复	于东　辛爱国

三等奖

进程间管道通讯技术在 UNIX 系统中的应用与分析	庄文祥
UNIX/XENIX 系统通用文件打印工具	刘忠会

出 版 前 言

为了普及计算机技术、推广计算机应用、迎接蓬勃兴起的全民学习计算机的热潮、总结广大计算机技术工作者、应用者、爱好者的技术成果以及在工作中的经验、体会、技巧，使之变为社会的财富。由电子工业出版社、中国计算机报社、中国计算机用户杂志社、中国电脑教育报社、软件世界杂志社共同发起组织的首届全国计算机操作、编程、应用、维修征文大奖赛，现在圆满结束。这次征文活动在读者中引起了强烈的反响，得到了广泛的响应，它有以下几个突出特点：

参与的普遍性，征文作者有计算机专业工作者，也有普通电脑使用者、爱好者、退休的老年人，还有在校的中学生。

内容的广泛性，来稿内容涵盖了计算机的各个技术领域。有开发项目完整的方案及程序设计，也有某个应用软件的使用技巧，还有硬件的防范及维护、维修经验。

文章的实用性，每篇文章阐述一个问题，讲深讲透，“拿来”就能用。

技术的先进性，来稿中包含了当前计算机的“热门”技术，如 Windows 95、Internet、多媒体及图形图象处理技术、Visual Basic、Visual C + + 等等。

从大量征文中经过主编认真遴选，编成《计算机应用技术精华丛书》——《数据库编程与应用(上、下)》、《网络软件与操作》、《多媒体与图形动画技术》、《办公软件实用技巧》、《操作系统及其使用技巧(上、下)》、《计算机语言及其应用实例》、《微机硬件故障防范与修复》共九种，现在正式出版发行。

应该说明的是：入选《计算机应用技术精华丛书》的文章就其技术性、实用性、先进性、可读性而言都是征文中比较优秀的。从入选《计算机应用技术精华丛书》的文章中经过由计算机专家组成的评委会评选出的获奖征文，其质量更胜一筹，相信《计算机应用技术精华丛书》的读者会有所鉴别。

征文活动的成功，《计算机应用技术精华丛书》的顺利出版，要感谢广大读者的热情参与、各册主编和各位评委的辛勤劳动。

发起组织征文大赛的初衷是重在参与，不以入选、获奖论英雄。入选、获奖的毕竟是少数，参与就是胜利！同时，由于篇幅所限，有的比较好的文章也未能入选，对此，除深感遗憾外，更要特别感谢那些积极、踊跃的参与者。

对于征文的组织、《计算机应用技术精华丛书》的出版，组织者们尽管尽心尽力了，但由于时间比较仓促，文章的挑选、编校的质量难免仍有疏漏，恳请读者指正。

目　录

第一部分　DOS 操作平台

第二部分　UNIX 操作平台及其它

第一部分　DOS 操作平台

浅析 CMOS 信息

安徽省贵池市秋浦西路 20 号（邮编 247100）　刘东明

在 IBM 及其兼容机上，CMOS RAM 存储器是用来存储受保护的机器系统信息，即使关机后，这些信息也不会丢失。若这些信息受到意外的损坏或改变，则下次开机后将出现错误信息，甚至不能引导机器。因此若我们能对这些信息的放置位置有所了解并知道如何得到它们，则这些信息将可以很好地为我们服务。关于 CMOS RAM 和如何修改 CMOS 信息中的口令字等内容的文章时常出现在各种报刊杂志上，为数较多的是修改口令字，有介绍硬件方法的，时而也有介绍软件方法的，但基本上都是针对某台具体机器而言的。众所周知，对 CMOS 存储器的访问，是通过两个 I/O 端口来实现的，第一端口（70h 端口）是用来设置 CMOS 中的地址。70h 端口是只写的（out 命令），因此你不能用它来读 CMOS 中的当前地址。第二个端口（71h 端口）是数据端口，用于读写编址的数据字节，它可用来将 CMOS 中的数据信息读出（in 命令），通过修改内存中的数据后，可利用 out 命令再将新值写回 CMOS 中。目前的高档微机上一般有 128 字节的 CMOS 信息，而早期的 286、386 微机只有 64 字节 CMOS 信息，因而对一台具体的机器，首先要判断它的 CMOS 信息字节数是 64 字节还是 128 字节，这就必须进行测试。测试的具体方法是：

第一、测试检查所有的 CMOS 存储地址线是否可用。在只有 64 字节的 CMOS 系统中，内存地址线 6 一般是不连的。例：要访问 CMOS 存储器中第 1010010b（52h）字节，而存储器地址线 6 并没有连上，则将丢失传递过来的地址线中最有效的位，得到的只是 CMOS 存储器中 0010010b（12h）字节的数据。所以可通过比较 CMOS 存储器低 64 位字节（要跳过 RTC 值，即实时时钟和日期信息，字节 0-dh，因它们不断变化）和高 64 位字节的内容，如全相同，则为 64 位字节 CMOS。

第二、要测试弄清 CMOS 存储器中确实安装了高 64 位字节地址，即地址线 6 确实连上了。从高 64 位字节中读一个值并将一个新值再写入该字节，然后再去读这个值，当读出的值与写进去的值不相同或读出的值与低 64 位字节对应的值（即当前偏移值减去 40h）相同时，则说明 CMOS 并设提供高 64 位字节，反之 CMOS 是 128 字节信息。

下面列出 128 字节 CMOS 信息的主要内容（列出的是已为一些厂商共同遵守的标准）：

字节 0-dh：　实时时钟及日期、时间信息（即 RTC 数据）。

字节 0ah：　状态字节，该字节控制 RTC 的速率。

字节 0bh：　状态字节，该字节控制 RTC 的特征。

字节 0ch：　状态字节，该字节包含一个标志位，用来标识目前激活的 RTC 中断。

字节 0dh：　状态字节，该字节用于 NMI（不可屏蔽中断）的实现和禁止。

字节 0e-0fh：诊断信息，用于诊断引导结果和标志关机状态。

系统配置信息：

字节 10h：　软驱类型。位 7～4 用于驱动器 0，位 3～0 用于驱动器 1；具体为：

0000b——没有驱动器

0001b——360KB 驱动器
0010b——1.2MB 驱动器
0011b——720KB 驱动器
0100b——1.44MB 驱动器
0101b——2.88MB 驱动器

字节 11h：　保留。

字节 12h：　硬盘类型。位 7～4 用于物理 C 盘，位 3～0 用于物理 D 盘；具体为：
0000b——没有硬盘
0001b～1110b——类型 1～14
1111b——使用扩展硬盘类型

字节 13h：　保留。

字节 14h：　外部设备信息。位 7～6 软盘驱动器个数，位 5～4 显示器类型，位 3～2 保留，位 1 协处理器是否安装，位 0 软驱是否安装。

字节 15-16h：　常规内存。

字节 17-18h：　扩展内存。

字节 19h：　扩展硬盘 C（当硬盘类型号＞15 时）。

字节 1ah：　扩展硬盘 D（当硬盘类型号＞15 时）。

字节 1b-2dh：　保留。对 AMI BIOS，1b～23h 为硬盘类型 47 物理 C 盘的具体参数（柱面、磁头、预补偿、控制字节、启停区、扇区数/每道），24～2ch 为硬盘类型 47 物理 D 盘的具体参数（柱面、磁头、预补偿、控制字节、启停区、扇区数/每道）。

字节 34-3fh：　保留。对 AMI BIOS，38～3dh 为口令字字节信息，3e～3fh 为 34～3dh 累加和检查字节；对 Super/486 机，38～3dh 为口令字字节信息．．．。

字节 40-7fh：　保留。（在不同的 BIOS 中是不尽相同的）

其中：

字节 4ah：　AST laptop BIOS——backlight 暂停时间 0～19 分钟。

字节 4bh：　AST laptop BIOS——硬盘暂停时间 0～19 分钟。

字节 4c-52h：　AST BIOS 的系统口令字设置字节信息。

字节 60-67h：　phoenix BIOS 硬盘类型 48 的参数（柱面、磁头、预补偿、启停区、扇区数/每道）

字节 75-7ch：　phoenix BIOS 硬盘类型 49 的参数（柱面、磁头、预补偿、启停区、扇区数/每道）

从以上可看出，对不同厂家的 BIOS 甚至是同一厂家的修改版本，CMOS 内容的具体部分也是存在差异的，因此我们必须先判断出口令字在 CMOS 中的具体位置，才能准确地清除口令字。现给出 CMOS 信息的读出和写回的汇编程序：

程序段 1：

```
U 100
CS：0100 MOV BX，0000
CS：0103 MOV AL，BL
CS：0105 OUT 70，AL
```

```
CS：0107 IN AL，71
CS：0109 MOV [BX+0180]，AL
CS：010D INC BX
CS：010E CMP BX，0080   ；若是 64 字节 CMOS，将 80 改为 40 即可。若不修改，则
CS：0112 JNZ 0103       ；读出的 128 字节中，除 180～18dh 和 1c0～1cdh（RTC 值）可
CS：0114 INT 3          ；能不同外，高 64 位字节与低 64 位字节对应位置的值全相
                        ；同，事实上对 CMOS 信息读出了两次（因地址线 6 没有连
                        ；上）。
```

该程序段是将 128 字节 CMOS 信息读到 180h 开始的内存单元中。

程序段 2：

```
U  200
CS：0200 MOV BX.0000
CS：0203 MOV AL，BL
CS：0205 OUT 70M，AL
CS：0207 MOV AL，[BX+0180]
CS：020B OUT 71，AL
CS：020D INC BX
CS：020E CMP BX，0080   ；若是 64 字节 CMOS，只需将 80 改为 40 即可。若不修改，
CS：0212 JNZ 0203       ；则对 CMOS 进行了两次写入，CMOS 中存储的信息是 1c0-
CS：0214 INT 3          ；1ffh 的内容（第 2 次写入的高 64 位字节值）。要引起重视，
                        ；若 1CO-1ffh 的值不是真正的 CMOS 信息，写入后将会出
                        ；现配置混乱。
```

该程序段是将 180h 单元起始的 128 字节 CMOS 信息写回 CMOS RAM 中。

利用上述的二个小程序段，可方便地修改 CMOS 中的内容。例：一台 AST ASN 800N 486/33S 笔记本机，因口令遗忘，无法进入 CMOS 设置菜单，笔者受人委托，要求清除口令字，先用程序段 1 读出 CMOS 信息，然后将 1CC-1D2h 置 0（180+4C=1CCh），再用程序段 2 写回 CMOS，重新开机，口令字已清除。（对 AMI BIOS，当清除口令字后，因累加检查和 3e-3fh 字节的值难以得知，所以机器重新启动后，发现和不对，硬、软驱等信息变为没有安装，要重新配置这些信息）

AMI BIOS CMOS 口令破解方法详解

兰州市七里河区工林路 345 号（邮码 730050）　温鲜红

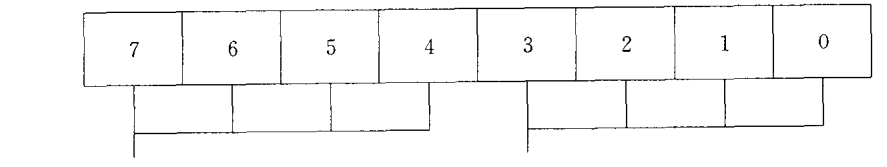
萌发破解 AMI BIOS CMOS 口令的念头，是从 CMOS SETUP 屏蔽掉 5.25 英寸软盘驱动器，以致虽安装有 5.25 英寸软盘适配器却不能使用的前提下产生的。而此问题的解决，最后归结到如何破解 AMI BIOS CMOS 口令这个问题。

对于 CMOS 硬加密系统的破解，通常有三种方法可直接或间接地达到目的。

法一，硬件消除法。此法操作简单，只须打开机箱，找到位于主板右下方的暗绿色的 CMOS 供电电池，然后短接电池正负极使其数据遗失，从而避开 CMOS 口令直接进入 SETUP 实用程序。此法直接对硬件进行操作，对于高档微机不易采用，怕误操作引起元件损坏或发生意外故障。因此不到万不得已的情况，一般不用此法。

法二，软故障法。通过 70H 和 71H 端口直接修改 CMOS RAM 数据区，重新启动时当发现与原 CMOS RAM 数据不符时，系统会自动跳过口令并使所有值变为默认值，这种方法简单易用，是避开 CMOS 口令的一种行之有效的办法。但必须搞清楚 CMOS RAM 各数据含义，才能有效地加以使用。而一般对其 10H 和 14H 偏移处的数据进行操作，10H 和 14H 处数据含义如下：

①软盘驱动器类型字节（位移 10H）：



安装第一台软驱的类型

0000＝无软盘驱动器

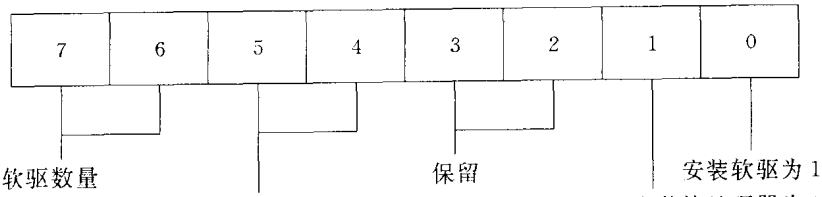
0010＝大容量软驱（道密度 96TPI）

安装第二台软驱的类型

0001＝双面软驱（道密度 48TPI）

0011～1111＝保留

②设备标志字节（位移 14H）



安装软驱数量

00＝1 台

01＝2 台

1×＝保留

显示器配置

00＝保留

11＝单显

01＝彩显 40×25

10＝彩显 80×25

保留

安装软驱为 1

安装协处理器为 1

CMOS RAM 是 ROM BIOS 的一部分，它由 64 个字节组成。具体可由程序 I 读得。

程序 I：

```
//This program may get CMOS RAM data.
# include <dos.h>
# include <stdio.h>

int main ( )
{int i;
 for (i=0; i<64; i++)
 {if (1%16==0) printf ("\n");
  outportb (0x70, i);
  printf ("%02x", inportb (0x71));
 }
return 0;
}
```

程序运行后输出 CMOS RAM 结果如下：

```
04  49  27  55  22  13  00  09  01  95  26  02  50  80  00  00
24  eb  f0  30  4f  80  02  00  0d  2f  00  e9  03  0f  ff  ff
08  e9  03  11  00  00  00  00  00  00  00  00  00  19  07  53
00  0d  19  a9  80  0f  84  aa  8b  c3  ac  32  6c  26  04  7b
```

程序 I 已显示了 AMI CMOS RAM 的标准数据，只需通过 70H（地址口）和 71H（数据口）端口往某地址写一与原来不同的数即可绕过 CMOS 口令，重新启动后可直接进行 CMOS SETUP 实用程序。若对 10H 位移处数据操作，可有：

```
outportb (0x70, 0x10);
outportb (0x71, 0x04);
```

通过对 AMI CMOS RAM 数据区的分析，可通过修改 10H 和 14H 两处解决本文开头所提出的软驱被屏蔽掉的问题。首先用下面语句设置软驱参数：

```
outportb (0x70, 0x10);
outportb (0x71, 0x24);
outportb (0x70, 0x14);
outportb (0x71, 0x4f);
```

设置完后可直接对软盘进行任何操作，待操作完然后再恢复 10H 和 14H 原来的值，这样就可绕过 CMOS 口令进行软盘操作。具体恢复操作如下：

```
outportb (0x70, 0x10);
outportb (0x71, 0x20);
outportb (0x70, 0x14);
outportb (0x71, 0x0f);
```

法三，分析法（程序法）。通过分析 CMOS 硬加密系统，找出其加密过程的源代码，以达到破解口令的目的。

就当前比较常见的AST BIOS和AMI BIOS来说，前者CMOS口令位于偏移70H～77H处（最多允许8个字符，以80H结束），而且以字符的内码形式存放，只不过有的字节第8位变为1，因此破解时只须每个字节与7FH相与，所得即为其内码，查内码表即可得其加密口令。而AMI BIOS口令位于偏移量B7H～BDH处，共7个字节（和CMOS RAM保留部分37H～3DH相同），其口令最多允许6个字节长度。由于CMOS RAM是ROM BIOS的一部分，其驻留内存高端，具体段址为F000H。通过对其驻留段的分析，找出了其验证和更新口令的核心部分，其偏移量为F315H，见程序Ⅱ。

程序Ⅱ：

```
C:\>DEBUG
-U F000:F315
F000:F315  F6C3C3     TEST BL,C3
F000:F318  7A01       JPE  F31B
F000:F31A  F9         STC
F000:F31B  D0DB       RCR BL,1
F000:F31D FEC8        DEC AL
F000:F31F 75F4        JNZ F315
F000:F321 C3          RET
```

该法是对CMOS核心子过程F315H的活用，由于验证和更新部分都调用F315H子过程对口令进行破密或加密。因此用C语言编制了一个CMOS口令破解程序，其核心是F315H子过程，通过C语言的嵌入式汇编功能实现其奇偶校验位检测、带进位循环右移等功能。源程序见程序Ⅲ，并在COMPAQ 386兼容机上已调试通过。

程序Ⅲ：

```
//This grogram may get CMOS password.
# include <dos.h>
# include <io.h>
# include <stdio.h>
# include <fcntl.h>

unsigned char password [7];
unsigned char encode [7];
FILE *fp;

int creat_file ( );

int main ( )
{int i, n=-1;
 unsigned char ch;
 creat_file ( );
 fp=fopen ("cmos.dat","rb+");
```