

目 录

第一部分 用机经验

从文件恢复谈对磁盘数据的整理	(1)
将 BIOS 调用口的内容反汇编出并写入文件的方法	(2)
巧用 XTREE 成批后备文件	(2)
GWINT16.COM 的一处修改	(3)
谈谈 GW0520 CH- I 型机的复位	(3)
对双拼双音 LLDOS 的修改	(4)
巧用 WS 两例	(5)
在 FoxPLUS 下自动实现小键盘的内部切换	(6)
CCBIOS2.13E 汉字系统 1 字节制表符的运用	(7)
防止误操作删除文件和数据的方法	(7)
巧用批处理文件防止误操作格式化硬盘	(8)
BASIC 如何读取 PASCAL 数据文件	(10)
10MB 硬盘驱动器容量的扩充	(11)
101 键标准键盘在长城机上的使用	(12)
如何绕过口令进入 VAX/VMS 系统	(13)
MS C 5.0 与 TurboC 1.5 互调库函数	(14)
怎样查找忘掉的文件名	(15)
使用 XTALK 仿真 XENIX 终端	(16)
微机版 UNIX 安装过程中有关 DOS 共享的处理	(17)
管理 UNIX 软盘文件系统的 SHELL 程序	(18)
AutoCAD 目录路径设置的简易方法	(19)
单显 PC 机的应用技巧	(20)
利用假脱机输出提高微机使用效率	(21)
高密度软盘文件交换要诀	(22)
如何使汇编程序运行后返回操作系统	(23)
怎样检测计算机内存	(23)
软件版权的保护技巧	(24)
提高人机工作效率的途径	(25)
隐藏子目录一法	(25)
巧用 TREE 命令	(26)
限制软件运行次数一法	(26)
IBM 机键盘宏定义	(28)
怎样消除 WordStar 控制符	(29)
巧用 DOS 的外部命令 ATTRIB 实现对磁盘文件的保护	(29)

改变硬盘活动分区的有效方法	(29)
386 微机口令的使用	(31)
怎样将控制字符直接送入文本文件	(32)
DEBUG 可恢复备份文件	(32)
一个批处理程序的妙用	(33)
如何在开机后立刻知道现场记录	(33)
使 WS 在任意机型及汉字操作系统下运行	(33)
TYPE 命令新功能的使用	(34)
加速汉字的首尾一拼音输入	(34)
不同机型间二进制数据文件的共享	(35)
利用短语输入功能快速输入特殊字符	(36)
两个影响中西文 WORDSTAR 工作效率的因素	(37)
巧用绘图仪内部命令解决其幅面问题	(37)
如何读取键盘功能键的 ASCII 码	(37)
巧用 SUBST 命令	(38)
用程序进行硬盘备份文件的恢复	(38)
巧用“.”和“..”文件	(39)
使 PC/XT 机的系统时钟具有计时功能	(40)
用 SUM 命令快速实现 TOTAL 命令的功能	(40)
利用 RND()函数实现软加密	(41)
利用批处理文件建立菜单	(42)
未存盘文件简便恢复法	(42)
CHKDSK 文件应用几例	(43)
巧用 EDLIN“制表符”	(43)
INKEY()函数键值的获得	(44)
在 3+ 网上进行大批量文件备份	(44)
解释 BASIC 语言可用内存不足的解决	(45)
怎样处理因磁盘满而无法存盘的情况	(45)
PATH 命令应用一例	(46)
状态行动态提示帮助	(46)
被误操作覆盖文件的恢复	(47)
用机内时钟控制时间延迟	(48)
利用 seek 或 find 命令实现连续查找	(49)
提高中英文混合输入速度	(49)
怎样使 FORMAT 格式化完毕后响铃	(50)
购机时应注意的几点	(50)
WS 软件任意方式下的使用	(51)
Orcad 的汉化	(51)
WS 具有快速进入编辑状态功能	(52)
XENIX 系统引导部分破坏的解决法	(52)

文本文件名忘却后的模糊查找	(53)
怎样给 PC 机加装自动校时	(53)
怎样在工作站上配置字符终端	(55)
自动删除零字节磁盘文件	(55)
五笔字型系统一级汉字输入简码的自定义	(56)
文件的流密码加密	(57)
如何改正错误的硬盘类型号	(58)
利用 Norton Utilities 改进汉字 FoxBASE+Ver 2.00	(58)
提高 PC 兼容机的串行通讯速度	(59)
在微机启动时设置隐形口令字	(59)
DOS 输出改向“≥”的正确使用	(60)
CCBIOS 2.13 系统自动换页的实现	(61)
碳化钨绘图笔的改进	(62)
用 3F UNSHARE 命令删除 3+ 网无共享名目录	(62)
实用防改版权信息设置	(63)
磁盘逻辑坏簇的自动辨认与回收	(64)
10MB 硬盘驱动器容量的扩充	(65)
大于 64K 的数据存取	(66)
改善汉字软件用户界面的新途径	(67)
HL2861 电子办公机使用技巧点滴	(67)
在 LX-V 汉卡上实现 WPS 功能的调用	(68)
在 SCDOS 下扩充上下角标字符功能	(69)
模拟 DOS 内部命令法	(70)
消除打印机 2708 病毒的方法	(71)
VAX-FORTRAN 文件共享技术	(72)
改向输入时键盘控制权的获取	(73)
UCDOS 中五笔输入方式的改进	(74)
维护 XENIX 系统的一种方法	(75)
如何保存和恢复硬盘正常主引导记录	(76)
在 2.13E 中用 LOTUS1-2-3 制作表格	(77)
XENIX 操作系统下中文文件名的建立和删除	(77)
XENIX 操作系统安装盘的制作	(78)
用“热键”激发计算器进行字段输入	(79)
中华学习机使用经验点滴	(80)
怎样用 WS 进行列向排版	(80)
软盘非标准格式化磁道扇区识别场的读取方法	(81)
多个 READ 命令并存时实现全屏幕编辑	(81)
多条件任意组合查询统计功能的实现	(83)
DOS 硬盘上可执行文件防拷贝的一种方法	(84)
NOVELL 网络中应用软件从单用户到多用户的实现	(85)

四通打字机“磁盘格式不对”问题的解决	(86)
一种恢复 XENIX 引导程序的方法	(87)

第二部分 数据库应用

浅谈 Config. sys 在 dBASE III 中的功用	(89)
快速编制软件使用说明书的技巧	(90)
dBASE III 与 C 语言间的数据传递法	(90)
在 C-dBASE III 中用宏替换函数实现快速统计汇总	(91)
关键字的妙用	(92)
数据库记录检索方法比较	(92)
通用 dBASE III 数据块搬迁程序	(94)
汉化 FoxBASE+ 的 CDOW() 与 CMONTH() 函数	(95)
READKEY() 函数的妙用	(95)
dBASE III plus 永久菜单显示法	(97)
dBASE III 数据库通用录入程序的设计	(98)
灵活使用“建立库结构文件命令”	(99)
实现 dBASE III 数据库嵌套功能	(100)
消除数据库中“隐含字符”的实用程序	(102)
在数据库中巧插多条记录	(103)
实现多重模糊查询的方法	(103)
动态生成“条件字符串”完成随机查询	(105)
如何计算 FoxBASE 索引文件长度	(107)
子程序调用的一级跟踪	(108)
汉字 dBASE III SET COLOR 命令功能的扩展	(109)
FoxBASE 下仿全屏幕输入	(111)
用逐步形成条件的方法实现多重模糊查询	(112)
在 XENIX 系统下用 FoxBASE+ 实现多用户功能	(114)
FoxBASE 库文件结构中添加信息的方法	(115)
在程序运行过程中动态修改数据库结构的方法	(116)
在 dBASE 中实现类电子表功能	(117)
改变数据库记录物理顺序通用程序	(119)
用 C 语言直接从 FoxBASE+ 数据库中读取数据的方法	(121)
dBASE III 中的 DOS 仿真	(124)
如何在 dBASE III 中动态统计数据	(124)
利用数据库及拷贝技术生成过程文件	(125)
采取修改库结构法保护数据库	(126)
用 dBASE III 实现列向合并文件	(127)
FoxBASE 硬盘数据库备份函数的设计	(128)
在 dBASE III 下直接执行 BASIC 屏幕图形文件	(129)
用 dBASE III 打印标准的著录卡片	(130)

dBASE III 使用中容易忽视的几个问题	(132)
如何解决 dBASE III 编译工具 Clipper 对 Total 命令的缺陷	(132)
一种 dBASE III 数据库的录入技术	(133)
dBASE III 与 LOTUS1-2-3 数据库文件的互相转换	(134)
C-dBASE III 打印程序出错处理	(134)
优化 FoxBASE+ 操作	(135)
dBASE III plus 显示格式的改进	(136)
实现数据库中记录的任意移动	(137)
随意调整数据库记录顺序	(137)
dBASE III + 多功能菜单实用程序	(138)
在 FoxBASE 下实现联想式汉字输入	(139)
提高数据库文件安全性的简单方法	(140)
在 FoxBASE+ 下随意作图	(141)
PROLOG 数据库在 PE 中的编辑	(143)
FoxBASE 中关于声响的控制	(144)
从 dBASE III 到 MFoxBASE+	(144)
UNIFY 通用数据屏幕 ENTER 的扩充裁剪	(144)
在 FoxBASE 状态下测试打印机状态	(145)
把三角函数引入 FoxBASE 中	(145)
实现 dBASE 查询浏览内容的任意翻页	(146)
利用 FoxBASE 建立一库对多库的联系	(147)
巧用 DOS 重定向 I/O 功能实现大批量的数据库结构备份	(148)
FoxBASE+ 可选字段输入模块	(149)
一个可以得到键盘扫描码的程序	(150)
在 XENIX 系统中怎样同时存放两种 FoxBASE 版本	(151)
如何利用 dBASE III 的 F1 功能键	(152)
模拟 PCshell 界面操作相关数据库	(153)
FoxBASE+ 下屏幕数据的编辑输入法	(154)

第三部分 编程技巧

巧用 C 语言的联合	(157)
一个找回 BASIC 程序的外部命令	(158)
快速分解过程文件	(159)
C 语言中字符型变量的输入	(160)
如何在 XE 编辑软件中使用五笔字型	(161)
WS 内存文件恢复	(161)
简便通用的多层次菜单·命令文件调用程序	(163)
程序时钟延时技术	(165)
运用两个函数实现 FoxBASE 软件计时保护	(167)
Turbo Prolog 程序设计中实现尾递归删除的两种方法	(167)

一个简单实用的自动报错程序	(169)
磁盘文件压缩与恢复程序	(170)
用 BASIC 调试 FORTRAN 程序	(172)
制表程序快速编制法	(173)
解决取子串时出现半个汉字方法二则	(174)
纠正 PCTOOLS. EXE 的一字节代码错误	(175)
谈 SET COLOR TO 语句的实现	(175)
Turbo BASIC 的汉化处理	(176)
如何将隐式文件改为普通文件	(177)
利用操作系统函数 INKEY() 实现适时时钟功能	(177)
C-WS 的一个缺陷及其消除方法	(178)
实现 FoxBASE 中 PRG 文件自动编译的方法	(179)
用 BASIC 形成供批量输出标准文件名程序	(180)
如何自制 CCDOS 汉字编码表	(181)
用 BASIC 的绘图功能弥补 dBASE III 的不足	(182)
文稿字数统计实用程序	(182)
XENIX SYSTEM V 中 haltsys 命令的改进	(183)
OFFICE 制表专用尺的制作	(184)
C 语言直接读取 *. DBF 库结构的方法	(184)
2. 13D 汉字操作系统驱动程序修改一例	(185)
利用 FoxBASE+ 实现程序文件大小写字母自动转换	(186)
程序在不同地址时的换算	(186)
程序模拟键盘锁	(187)
用 C 语言读数据库中的数据	(189)
报表取舍过程的实现	(190)
错误信息记录程序	(191)
中文文本文件排版程序	(191)
音乐程序的编制	(192)
运用 error() 函数提高程序的容错率	(193)
一个辅助 dBASE III 数据录入的子程序	(194)
Quick BASIC 和 Turbo BASIC 的变通汉化方法	(195)
更改特定程序行的一种方法	(196)
2. 13H 系统 FILE16B. COM 文件的修正	(196)
为 Quick BASIC 程序自建图形汉字库	(196)
BASIC 两种数据文件的自动转换	(198)
建立姓氏字典实现数据库中按姓氏笔画排序	(198)
工资票面的自动统计	(199)
代码录入正确性的判断程序	(200)
词组文件的高效制作及其不同版本间的格式变换	(202)
增强多用户 FoxBASE 程序的安全性	(203)

巧用字符串模拟堆栈操作	(204)
实现 FoxBASE 程序自动演示运行	(205)
硬盘分区表的保存与恢复程序	(206)
用 C 语言实现光条菜单	(208)

第四部分 屏显与打印

AR-2463 打印机的兼容性及其在屏幕图形打印中的应用	(210)
一种组合的汉字 DOS 操作系统	(210)
C 语言中显示彩色字符的方法	(211)
清除“雪花”现象	(212)
Turbo C 语言的打印机输出	(212)
一种实用的打印程序 PRINTOOL	(214)
随意改变屏幕颜色的方法	(215)
CCDOS 下西文 Turbo C 显示彩色汉字的方法	(217)
XENIX C 语言中的清屏与单键响应	(217)
在屏幕任意位置显示代码及代码内容	(218)
屏幕显示内容的文件转贮	(220)
全自动打印 dBASE 数据库报表	(223)
用汇编语言实现光标键选菜单	(224)
Turbo C 单色显示器图形驱动文件的改进	(226)
怎样生成高级点阵空心字库	(227)
汉化 Turbo PASCAL 3.01A 编辑器的方法	(228)
在汉化 FoxBASE+V2.10 中建立中文下拉菜单	(229)
显示方式转换的软件实现	(230)
LOTUS 通用打印报表宏命令程序	(231)
一种通用制表方法	(232)
如何在 DOS 状态下直接显示 AutoCAD 的图形	(233)
FE 汉化	(236)
利用高点阵打印字库显示大汉字	(238)
文本文件的彩色显示	(240)
dBASE 与 WordStar 的屏色变换	(242)
用命令完成文本文件批打印	(242)
打印中途改变字体字号实用程序	(242)
游戏画面的拷贝	(243)
打印字号测试程序	(245)
SUPERCALC ³ 的实用打印	(245)
用 WS 实现分栏编辑和打印	(246)
CCDOS V4.0 四种字体的打印	(247)
特殊图形符号输出技巧	(248)
打印汉字及纯中文符号的区位码	(250)

报表和数据的快速假脱机打印	(250)
如何使用 dBASE 进行并列打印	(251)
巧用 SET MARGIN 命令实现多列打印	(252)
CCBIOS 2.13 打印命令的使用技巧	(253)
连续打印程序清单简法	(254)
FAT 表内容的直观显示	(254)
怎样使 CCDOS 下的光标闪烁	(255)
快速实现数据库信息显示	(256)
在 SCDOS 操作系统下打印高精度频率分布直方图	(257)
使 WORDSTAR 具有稿纸打印功能	(259)
DEBUG 调试工具清屏功能的扩充	(259)
电子表格处理软件中如何改变打印字号	(260)
运用 SUBSTB 函数进行逐字逐行显示	(261)
用软件解决新时代汉卡不打印问题	(261)
具有立体投影效果窗口的的设计方法	(262)
视频显示模式的简易转换法	(263)
在 WS 内实现字体控制和各种打印输出	(263)
测试打印机的状态	(264)
CCDOS2.1 平滑滚动显示的实现	(264)
打印字号对照表程序	(265)
WS 软件文章标题打印位置居中的计算	(266)
利用 WAIT 命令实现打印走纸控制	(267)
2.13 系统下如何打印彩色汉字	(267)
自带字库打印机打印方法	(268)
全屏幕设计应用软件题头	(269)
彩显色彩不纯的故障排除实例	(270)
用 GRP 图形设备实现屏幕表格	(270)
用 2.13D 的图形拷贝打印封面	(271)
FoxBASE+ 存屏功能的完善	(271)
将 CCDOS2.13F 的黑白菜单选择改成彩色	(272)
如何修改 CCDOS4.0 的屏幕彩色控制	(274)
多种屏幕显示方式的快速转换	(274)
多行文字的横向出入屏幕显示	(275)
屏幕信息延迟显示的几种设计	(276)
实现 NEW9P 打印封闭表格	(277)
CD-DOS2.13H 状态下的彩色屏幕设计	(277)
立体投影菜单的建立	(278)
FoxBASE 下的终端打印	(279)
在窗口内显示移动信息	(279)
在 WMIDOS5.0 系统上使用 2.13H 的打印功能	(281)

安装 ANSI.SYS 配量程序后屏幕不滚动的解决方法	(282)
屏幕显示方式的切换	(282)

第五部分 维修实例

M-1724 打印机常见故障及处理	(284)
3Plus 常见故障排除一例	(284)
负氧离子作祟导致 UPS 失常	(285)
浪潮 LC0530AT 微机维修实例	(285)
M-1570 打印机断针故障处理	(285)
M380/XP1 故障维修一例	(286)
微机硬盘驱动器故障维修一例	(286)
AR-3240 打印机常见故障维修	(286)
大容量硬盘零道损坏的处理方法	(287)
AST 286 故障排除一则	(287)
利用冷却法修复微机一例	(288)
286、386 微机设备配置信息错误常见故障及其排除法	(288)
长城 0520A 型微机电源故障的维修	(288)
紫金 3080 汉字打印机故障检修	(289)
M-1724 打印机故障维修一例	(289)
如何检测打印机断针	(289)
OKI-8360C 打印机打印错位的处理	(290)
雨季应注意磁盘驱动器故障	(291)
在 FoxBASE+ 状态下校核磁盘中的数据文件名	(291)
DMP-60 系列绘图仪接口的诊断及处理	(292)
NM9400 打印机不能打印汉字故障的维修	(293)

第一部分 用机经验

从文件恢复谈对磁盘数据的整理

很多文章都谈到过, DOS 删除文件时将其目录项的首字符标记为 ES。然后将 FAT 中该文件所占用部分置零, 因此, 当需要恢复文件时只需由用户指出待恢复文件的首字符, 然后根据目录项中指出的首簇号和文件大小, 即可将该文件恢复。然而, 文件在磁盘上的存放较复杂, 若要方便、准确地恢复原文件, 磁盘必须满足两个条件:

(1) 该文件被删后, 必须未对磁盘进行写操作。写操作包括磁盘内容的增加和减少。

(2) 如果待恢复文件被删后, 对磁盘继续进行了删除操作, 则该文件只能由人工逐簇来恢复了。

文件被删除后, 由于 DOS 对磁盘的使用是顺序搜索剩余空间, 如进行写盘操作增加内容, 被删除文件所占的部分磁盘很可能就被最先使用而不能被恢复。如对磁盘继续进行删除操作后, 被删除文件在磁盘上又不是连续存放的, 则夏修河同志所述方法已不适用。这时用 PCTOOLS 等软件工具也无法准确的恢复, 因为在恢复时它们是按连续未使用空间来处理, 因此恢复后的文件就可能含有原不属于它的内容, 而它本身的某些内容又不包含在该文件内。这样, 就只能人工搜索未分磁盘的空簇以确定该簇是否属于待恢复文件, 对于文本文件, 这种人工搜索尚不算很困难, 如待恢复文件是 .EXE 或 .COM 等二进制文件, 这种搜索就极不容易了。

为解决这个问题, 用户应尽可能地使数据文件在磁盘中顺序存放。使数据文件在磁盘中顺序存放的最简单方便的方法是对磁盘

文件备份 (BACKUP), 在删除原文件后再恢复, 即可将磁盘文件按目录顺序依次存放在磁盘上。但磁盘做备份很费时, 特别是对硬磁盘的备份, 需用大量的软盘, 如果用户手上有 PCTOOLS R4. 11 以上的盘, 就可用其中的 COMPRESS 对磁盘进行压缩和整理。用户也可自行设计一个磁盘数据整理程序, 该程序主要是使磁盘数据按用户指定的方式顺序存放。这样整理后的文件, 如被误删除后, 只要未在磁盘上增加内容, 就能被方便、准确的恢复。

在编制程序时, 应考虑首先按照在磁盘上的文件顺序对目录进行排序, 如按文件生成或最后修改日期、按文件属性、按文件现存放于磁盘中的先后顺序等。然后再对文件在磁盘上的存储进行整理。由于进行大量的读/写操作, 该程序运行的时间就较长, 为此建议在对目录进行排序时, 应注意将系统文件、只读文件及其它不需进行修改或修改后文件长度不发生变化的文件置于磁盘的前部, 而将常需修改而改变长度或正在编辑的文件置于磁盘较后的位置, 这样就能大大减少以后整理时的读/写操作, 从而加快处理的速度。按照这种方法排列的磁盘, 仅在首次运行整理程序时, 由于磁盘文件的存放较混乱而费时 (这时可使用 BACKUP, 软盘用 COPY 也可), 在以后的使用中, 如经常对磁盘进行整理则很快就能完成整理工作。对于子目录内的文件, 可将其 (子) 目录项作为一个文件, 每个目录项仍按根目录的方式进行整理。

经这样整理后的文件, 在磁盘中存放紧凑, 不仅在误删除后便于恢复, 而且能减少磁盘访问时间, 加快程序运行速度, 也便于用户对磁盘文件进行编程使用和监测, 使用户能

够及时发现文件的无故延长,即对计算机外壳类病毒具有检测作用。

对磁盘文件的整理工作,可在对磁盘文件做了较大的变动或用户认为必要的时候进行。如果用户自行编制了数据整理程序而该程序的运行又足够快,则可在每次关机前运行。

(滕 亮)

将 BIOS 调用口的内容反汇编出 并写入文件的方法

随着计算机的发展,目前,各种机型的 BIOS 功能不断扩充。但有关这方面的资料十分缺乏,这给用户开发软件带来困难。

我在开发 EGA,VGA,CEGA 等图形卡上的图形压缩存贮和还原显示软件时,为了弄清这几种图形卡的控制方法,用 DEBUG 对 BIOS 进行了分析,从中摸索出了如何将 BIOS 调用口的内容反汇编出并写入文件的方法。用这种方法用户就可得到一份较完整的反汇编程序清单。下面通过一个具体例子(把 int 10H 的 0 号功能口反汇编出)来掌握这种方法。

第一步:用 DEBUG 对需要反汇编的功能口进行一遍粗扫描,以便分清数据区与程序区。方法如下:

```
C>DEBUG      ;进入 DEBUG
A100          ;从 100H 开始汇编一段程序
mov AH,00     ;int 10H 的 0 号功能口
mov AL,06     ;汉字显示模式
int 10        ;int 10H 调用
              ;汇编结束
T=100         ;从 100H 开始执行
T             ;单步执行
T             ;单步进入 int 10H 调用口
```

此后,就可以用 DEBUG 的 U 和 D 命令来分清数据区与程序区,为下一步操作打下

基础。

第二步:用 edlin 将第一步的操作步骤写成一个文件(假设文件名为 ABC)。方法如下:

```
C>edlin ABC
A 100
mov AH,00
mov AL,06
int 10
      ↵
T=100
T
T
```

D 段址:偏址 偏址 ;反汇编出数据区的数据

U 段址:偏址 偏址 ;反汇编出程序区的程序

Q ;退出 DEBUG

第三步:利用 DOS 管道功能把反汇编出的内容写入文件(假设文件名为 XYZ)。方法如下:

```
C>type ABC|DEBUG>XYZ
```

以上操作的含义是把 ABC 文件的内容作为 DEBUG 的输入;把 DEBUG 的输出写入 XYZ 文件。完成以上操作后,XYZ 文件中的内容就是反汇编出的内容。此后,用户就可方便地用 WS 将其分页打印出。

(戴水贵)

巧用 XTREE 成批后备文件

许多计算机用户反映,对需从硬盘驱动器中将较多文件用 BACKUP 命令后备到 A 或 B 软盘上时,因已后备出的文件仅能用 RESTORE 命令恢复方能使用,而感到十分不便。这里介绍一种类似 PCTOOLS 的美国 XTREE 工具软件(1985 年产品,V2.00 版),只用 COPY 命令即可从后备盘中恢复拷贝盘中任意文件。

1. 在 C 状态下,调用执行 XTREE 文件。

```
C>XTREE
```

2. 将活动光标移至根目录或所需拷出的目录名上,并回车。

3. 同时按下 CTRL 键和 T 键,给拟拷出的全部文件打上标记。

4. 按下 CTRL 键和 C 键,并回车。

5. 指定目标盘符如 A:,并回车。

6. 此时屏幕提示是否对目标盘相同文件进行覆盖,如确认,则系统自动将已标记的文件顺序拷入 A 盘。屏幕上反向光棒随时移动提示正在拷贝执行的文件名和字节数。

7. 如 C 盘内需拷出文件数(容)量超出 A 盘所能容纳范围,屏幕提示:

Error:Disk Full. Press ANY KEY to Continue

此时,按回车键返回第 5 项,可放入新盘继续拷贝,此后从上一未拷完文件处开始拷贝,直至拷贝结束。

8. 按 F1,选 Y 后退出系统。

此法已在 IBM-PC/XT、长城 0520CH、286B 机上运行和验证。

(刘 斌)

GWINT16.COM 的一处修改

大家知道,GWINT16.COM 是长城系列机的键盘管理程序,我们使用时发现:用 Ctrl-F10 键选字型和行宽,当选了字型后按回车键或直接按回车键到置行宽时,如果按了退格键后就不能输入数字,只能输入 A~P 的字母。经分析,这是边界条件没有处理好造成的错误,修改的办法就是把比较 2 改为比较 3,步骤如下:

C>debug gwint16

-S CS:0FFFE 3D 00 67 搜索 CMP AX,

6700 指令(即搜索 Ctrl-F10 键)

-2F54:71D9 找到处理 Ctrl-F10 键程序入口地址

-U 71D9

-XXXX:71D9 CMP AX,6700

-XXXX:71DC JNZ 71FF

-XXXX:71DE JMP 7F59

-U CS:7FB0

-XXXX:7FB6 CMP BYTE PTR[6B0D],02 把
比较2改为比较3

-E 7FBA

-XXXX:02.03

-W

-Q

(魏 旭)

谈谈 GW0520 CH-Ⅱ 型机的复位

GW0520 CH-Ⅱ 型机以其主频 8 兆的高速度赢得了越来越多的用户,但是,它在机器复位处理上较 GW0520 CH 机显得有些繁琐。GW0520 CH 机在复位时不再去检查内存单元,而是直接启动磁盘,复位时间很短。而 GW0520 CH-Ⅱ 型机在复位时,若在此之前没有做过热启动,则与冷启动时一样,先要占用相当一段时间按每次 16K,直到 640K 内存全部检查完毕才去启动磁盘,这个过程相当费时,而一般情况下机器复位时没必要再去检查内存。在机器运行过程中,若遇到死锁不能热启动,为了节省时间,一般都是去按复位键复位机器,但对于 GW0520CH-Ⅱ 型机来说,这一过程往往与冷启动时所耗费的时间一样多,并不节省时间。

针对这一情况,笔者分析了 ROM BIOS 中关于这部分的内容,发现 ROM BIOS 程序在运行到内存地址 F000:E46FH(注意:《IBM PC/XT 硬件手册》上的 ROM BIOS 清单中把此地址误写为 F000:E46EH)时,它首先要检查一个名子叫 RESET-FLAG 的内存单元(占 2 字节)的内容,该单元的内存始地址为 40:72H,若其内容不为 1234H,则先检查内存,再做后续工作;若是 1234H 则跳过检查内存这一部分。1234H 实际上是热启动的标志,DOS 在接收到热启动的信息时,就把 RESET-FLAG 单元置为该值,以示标识,而每次复位,ROM BIOS 都要检查该单元,这就是前面提到的为什么 GW0520 CH-Ⅱ 机在复位时,若在此之前没热启动过,机器就要先检查内

存的原因所在。

为了省去这一费时的检查过程,像 GW0520 CH 一样很快地复位,笔者用 DEBUG 编制了一个向 RESET-FLAG 单元赋以 1234H 值的小程序 REST.COM,具体编制过程如下:

C>DEBUG

-A

```
189A:0100 1E      PUSH    DS
189A:0101 B84000   MOV     AX,0040
189A:0104 8ED8     MOV     DS,AX
189A:0106 B83412   MOV     AX,1234
189A:0109 A27200   MOV     [0072],AL
189A:010C 88267300 MOV     [0073],AH
189A:0110 1F      POP     DS
189A:0111 CD20     INT     20
189A:0113 ^ C
```

-RCX

:13

-NREST.COM

-W

-Q

REST.COM 可以放在 AUTOEXEC.BAT 中,每次开机后,它自动设置 RESET-FLAG 单元,以备机器死锁后,用复位键复位时不再费时。

(彭起顺)

对双拼双音 LLDOS 的修改

LLDOS 是目前国内比较流行的 CCDOS 的改进版,最主要的特点是采用了双音编码汉字输入方案,采用词汇输入方法,大大提高了汉字的输入速度。

但是,根据我们的体会,LLDOS 还有一些需要改进的地方,通过对原软件的分析,我们对个别地方做了一些修改,下面对此逐一进行介绍。

一、对用右翻页键“>”代替反斜杠键“\”转拼音提示行选择的修改

从其他方式转多字词汇和转拼音提示行选择,LLDOS 均采用反斜杠键“\”转换。我们认为转拼音提示行选择使用“\”键不如使用“>”键操作方便,因为拼音提示行右翻页键也是用“>”键操作,所以转拼音提示行选择和右翻页键用同一键操作,既符合操作习惯,也有利于提高输入速度。

修改方法:先将 HANZIKD.EXE(PC 机 LLDOS 文件)改名为 HANZIKD,再用 DEBUG 装入内存,用 S 命令查找修改地址。反斜杠键“\”的 ASCII 值为 5C(16 进制),在 PC-XT 机上运行:

C>REN HANZIKD.EXE HANZIKD✓

C>DEBUG HANZIKD✓

-S 100,E18E 3C 5C✓

5D97:0DE7

5D97:105B

5D97:CAD6

5D97:DBCF

5D97:DC07

-U CAD6✓

5D97:CAD6 3C2E CMP AL,5C

5D97:CAD8 7508 JNZ CAE2

5D97:CADA C6067F0301 MOV BYTE PTR
[037F],01

5D97:CADF EB36 JMP CB17

-U DC07 ✓

5D97:DC07 3C2E CMP AL,5C

5D97:DC09 7441 JZ DC4C

5D97:DC0B 803ECE0302 CMP BYTE PTR
[03CE],02

5D97:DC10 7F0A JG DC1C

“>”键的 ASCII 值为 2E,将 5C 修改为 2E。

-ECAD7✓

5C. 2E✓

-EDC08✓

5C. 2E✓

-W ✓

-Q ✓

C>REN HANZIKD HANZIKD.EXE✓

在长城机 GW-286 上修改 LLINT161.

EXE 文件,方法同上,参考地址为:CD02.E165。

经过上述修改后,操作人员就可以用“>”键代替“\”键转拼音提示行选择了。

二、对在拼音提示行选择字符的修改

在拼音提示行选择字符,LLDOS 有两点需要改进的地方。

1. 当键入“0~9”数字选择一次字符后,还可继续键入“0~9”选择若干次,这在大多数场合下是不适用的,而带来的后果是易引起误操作。例如,需要输入“比去年多产50万吨”字符串的“产”字,若采用拼音提示行选择的方法,则键入“if”后再键入右翻页键“>”转拼音提示行选择,这时选择“8”输入“产”,此时若紧接着输入“50”,屏幕上却显示的是“谗颤”。

2. 当选择一次后不能紧接着输入左翻页键面上的“,”号,例如,键入“ghhj”,在提示行出现重码选择提示:1规则,2鬼话,3桂花;在选择“1”后输入“规划”,若紧接着输入“,”号则出现响铃,不能输入。

遇到上述情况,操作人员必须键入回车清除提示行后才能输入“,”号,或正确输入“0~9”数字,因此,操作不方便,且易产生输入错误,不利于“盲敲”,影响输入速度。修改方法如下(PC-XT 机):

```
C>REN HANZIKD.EXE HANZIKD
C>DEBUG HANZIKD
-U DE4
5D97:0DE4 E9D5C3 JMP DIBC
5D97:0DE7 3C5C    CMP AL,5C
5D97:0DE9 7534    JNZ 0E1F
5D97: 0DEB C606800300 MOV BYTE PTR
[0380],00
5D97: 0DF0 803E050001 CMP BYTE PTR
[0005],01
-A DE4
5D97:0DE4 JMP 104F
5D97:0DE7 ^ C
-W
```

-Q

C>REN HANZIKD HANZIKD.EXE

在长城机 GW-286 上修改文件 LLINT16I.EXE,方法同上,参考地址为:0F3F,修改“JMP D714”为“JMP 1232”。

修改后,操作人员在第一次选择后不需键入回车键清除提示行就可以输入“0~9”数码和“,”号,若要再一次选择当前提示行中字符,可采用“ALT+”的方法,对区位码提示行选择也是用同样的方法。

(孙 鉴)

巧用 WS 两例

笔者认为,WS 系统有一些可以改进的地方,如 WS 文本文件与 DOS 文本文件的格式不太一致,影响了某些应用。WS 中的打印控制主要是针对原配 IBM-PC 的 FX-100 打印机设置的,与当前常见的打印机不一致。本文针对这两点,提出了两条解决途径。

1. 实现 WS 文件与 DOS 文本文件的互换

在 DOS 方式下直接用 TYPE 命令显示 WS 文件,显示的结果凌乱无章,这是因为 WS 的行末自动对齐换行功能对文本进行了处理。在 WS 文件中,回车、换行及空格均有软硬之分,硬的回车、换行和空格码为标准的 0AH、0DH 和 20H,是用户输入的字符;而软的回车、换行及空格为 8AH、8DH 和 A0H,是 WS 中根据需要自动插入之符号。同时,当一个单词后面插入软空格(A0H)以后,其最后一个字符的高位设置成“1”,例如 S(53H)变成(D3H),由此造成了显示结果的杂乱无章。而在一般的 DOS 系统中,所用字符均由七位 ASCII 码表示,高位为“0”。根据这一特点,可以编制程序滤掉其高位,而实现 WS 文件到普通的 DOS 文件之转换。转换结果除在 DOS 中可以方便显示外,还可以送入一些特殊的系统进行处理,例如转换后,可送入“科印”系

统进行排版。下面以 PASCAL 语言给出转换程序。

```
PROGRAM WS-TO-DOS(f1,f2);
  VAR  f1,f2:text;
        Ch:char;
BEGIN  reset(f1);rewrite(f2);
        WHILE not(eof(f1))DO
        BEGIN
            IF eof(f1)THEN
            BEGIN readln(f1);writeln(f2)END
            ELSE
            BEGIN read(f1,Ch);write(f2,chr
                (asc(ch)AND127))END
        END
    .END
```

反之,在有些场合,亦需要将 DOS 文件送入 WS 系统进行排版处理。根据上述思想,只需将文件中的硬回车换行改成软回车换行(只在段落处保留硬回车换行),然后送入 WS 进行重新排版。

上述处理主要是对 DOS 下的西文文件进行的,CCDOS 及汉化的 WS 对排版功能进行了修改,去掉了添加软空格的功能而重新设计换行处理,软回车与软换行仍予以保留。但由于 CCDOS 汉字都采用八位二进制码表示,因此,上述处理方法应稍加改变,转换程序将着眼于软回车换行(8AH,8DH)与硬回车换行(0AH,0DH)之间的互换。

2. 打印功能的重新设置

WS 本身具有强有力的打印控制功能,可以在文本中很方便地(打入 ↑P 及另一个字符)加入打印控制字符,但具体的实现则取决于打印机。而原始 WS 版本多是针对目前已很少使用的 PX-100 之类的打印机,因而这些打印功能在常见的打印机上难以发挥,在 CCDOS2.13 中可以另加入一些复杂的控制符加以弥补。WS 系统提供了一套专用于重新设置打印功能的安装程序 WSINSTALL,该程序可以根据打印机的控制码重新改变 ↑P 的输出参数,从而使得这行打印控制命令重新

生效,整个安装过程均有菜单导引,人机交互完成,十分方便。但是由于 WSINSTALL 软件中的缺陷,对于内存大于 640KB 的硬件配置,WSINSTALL 均运行不起来,而给出信息“Too many memory!”。针对这种情况,下边给出一个有效的解决方法:先调 CCDOS(任何版本均可),然后用控制键把屏幕设置成字符状态(通常用 ctrl-F7 命令),再调用 WSINSTALL 程序,这时就可以正常工作了。这样做的原因,是用汉字库占去部分内存,从而避开 WSINSTALL 中所存在的缺陷。(胡 庆)

在 FoxPLUS 下自动实现 小键盘的内部切换

FoxPLUS 程序运行过程中,以“人机对话”方式实现数据的输入或更新。对于数字输入,键盘上有两套键码:一个是同特殊符号连用的数字键;另一个是同光标移动连用的数字键(俗称“小键盘”)。

不少用户都有这样的教训:当从小键盘上输入数据时,并没有相应的数字出现,而是变成了光标的移动,导致输入失败(许多场合下,光标的左移、下移等均默认为是回车)。而正常的操作应当是,先使小键盘转换成数字状态,然后再输入。在程序要输入数据之前,采用程序的办法先自动将小键盘的状态切换成数字状态,避免了上述情况的发生。

基本原理:用 FoxPLUS 为用户提供的汇编接口,调用一个汇编子程序(程序清单附后,已在 IBM-PC 及各类 286 机上通过)。

使用方法:先将源程序 XJP.ASM 编译、链接,当出现“Warning:no stack segment”信息后,可以暂不处理,用 EXE2BIN 将程序转换成 XJP.BIN 文件,在用户程序中执行一次 LOAD XJP 操作,再在需要的地方执行 CALL XJP 即可。

汇编子程序清单:

D)TYPE'XJP. ASM

```
CSEG    SEGMENT BYTE
        ASSUME  CS:CSEG
KS      PROC    FAR
        MOV     AX,40H
        MOV     DS,AX
        MOV     SI,0017H
        MOV     AL,[SI]
        OR      AL,20H
        MOV     [SI],AL
        RET
KS      ENDP
CSEG    ENDS
        END
```

(李全忠)

CCBIOS2.13E 汉字系统 1字节制表符的运用

笔者根据实践经验谈谈 CCBios2.13E 版本以上的汉字系统1字节制表符的使用体会。

一般的 CCDOS 都具有专用制表符,他们位于09区内,都是2字节制表符,而 CCBios2.13E 以上的汉字系统提供了1字节制表符。

CCBIOS2.13E 汉字系统的24×24点阵打印字符使用专用字符库,位于 ZF24.COM 程序内,点阵为24×12,可用 ZZF24.EXE 程序修改,其中存有三种字体、字符,在操作系统下(C)ZF24 0₄可使用1字节制表符。

在程序设计时适当选用1字节制表符,可在打印行宽度不变的情况下,增加打印字符的个数。

例:在一个打印行有25个栏目的工资表中,共用了26个2字节的栏目分隔符“|”,当把它们分别用1字节制表符“!”取代时,可在原来一行内多打印26个字符,其打印长度不变,打印字符个数增多了,而打印表格的效果与2字节制表符一致。

值得注意的是,要在非纯中文状态下选

用1字节制表符,除了划横、直线用的2字节制表符要分别用两个对应的1字节制表符取代外,其余的2字节制表符只能用一个与其对应的1字节制表符取代。在同一个报表中,最好所有制表符都统一选用1字节制表符,但在同一报表纵向上,同一列栏与栏的分隔符一定要统一选用一字节制表符,否则将打印出非衔接的表格。此外,在 CCBios2.13E 汉字系统的批处理文件 AUTOEXEC.BAT 中的 FILE48 1之后增加命令 ZF24 0。

按照上述方法使用 CCBios2.13E 的1字节制表符,在打印行宽不变的情况下,能够增加打印字符的个数。

(刘昌斌)

防止误操作删除文件 和数据的方法

在计算机使用过程中,有时因为对 DOS 内部命令(DELET 或 ERASE)的使用及参数指定的错误而造成文件和数据丢失,原因就是使用了 DEL *.* 这个命令。使用该命令时,虽然系统提示:“ARE YOU SURE (Y/N)”,要求用户回答 Y(删除)或 N(不删除),但有的用户总是忘记了自己所指定的参数,而选择了 Y,把不该删除的文件全部删除。

为防止误操作将文件和数据删除,我们可以限制用户指定 *.* 这样的参数,但由于 DEL 命令是 DOS 的内部命令,不象外部命令那样容易在外层控制,只能通过修改 DOS 的 COMMAND.COM 文件来解决,因此我们可以采取改变系统对用户键入 Y 参数的处理,即将用户键入的 Y 当做 N 来处理,程序清单及说明如下。

C)MD COM₄;建立一个子目录,以防破坏根目录里的 COMMAND.COM 文件。

C)COPY COMMAND.COM\COM\COMMAND.COM₄;将 COMMAND.COM 文件拷到 COM 子目录里。

C:\DEBUG\COM\COMMAND.COM ↙; 调用 DEBUG
程序修改 COMMAND.COM 文件。

--T

--S CS:0 FFFF B4 13↙; 搜索 MOV AH,13 指令。

xxxx:3A45 ; 显示搜索到的地址。

--U3A45↙ ; 显示 xxxx:3A45 地址的程
序和代码。

xxxx:3A45 MOV AH,; 删除文件功能调用号。

13

xxxx:3A47 MOV DX,

005C

xxxx:3A4A INT 21

.

.

.

--S CS:0 FFFF B8 0A; 搜索 MOV AX,0C0A 指令。

↙

xxxx:3A24 ; 显示搜索到的地址。

--U3A24↙ ; 显示 xxxx:3A24 地址的程
序和代码。

xxxx:3A24 MOV AX,

0C0A

xxxx:3A27 INT 21

xxxx:3A29 LODSW

xxxx:3A2A OR AH,AH

xxxx:3A2E JZ 3A15

xxxx:3A2F CALL 2BE2

xxxx:3A31 OR AL,20

xxxx:3A33 CMP AL,

[3B9D]

xxxx:3A37 JNZ 3A3A

xxxx:3A39 RET

xxxx:3A3A CMP AL,

[3B9C]

xxxx:3A3E PUSHF

xxxx:3A3F CALL 2BD5

xxxx:3A42 POPF

xxxx:3A43 JNZ 3A15

xxxx:3A45 MOV AH,

13

xxxx:3A47 MOV DX,

005C

xxxx:3A4A INT 21

xxxx:3A4C INC AL

xxxx:3A4E JZ 3A54

xxxx:3A50 CALL 2B04

xxxx:3A53 RET

--A3A37↙

xxxx:3A37 NOP↙

xxxx:3A38 NOP↙

xxxx:3A39↙

--W↙

C)

用户可以在系统启动后,转到 COM 子目
录里执行一次 COMMAND 命令;也可以将原
COMMAND.COM 文件改名拷贝到 COM 子
目录里,再将修改后的 COMMAND.COM 文
件拷贝到根目录里。这样,使用 DEL *.* 这
类的命令,就不会删除任何文件了,若要删除
文件,必须键入文件的全名或部分字符。

(来文兰 张俊)

巧用批处理文件 防止误操作格式化硬盘

本人在指导学生上机实习时发现,学生
在使用 MS-DOS 2.11 中的 FORMAT 命令对
软盘进行格式化时,往往由于初次接触 DOS,
对 DOS 命令不熟,而错将硬盘格式化。

例如,学生如果键入下列命令,即会误将
硬盘格式化:

C:\FORMAT↙

注:机器所配西文 DOS 为 MS-DOS 2.
11,中文 DOS 不限,且 FORMAT.COM 文件
在硬盘的根目录下。

笔者编制了一个批处理文件,可尽量避
免由于误操作而造成的硬盘格式化;同时执
行此批处理文件,还可提示用户任选八种软
盘格式化命令之一对软盘进行格式化,具体
说明如下。

1. 为了防止误用 FORMAT 命令,可将
DOS 中的 FORMAT.COM 文件放入硬盘的子