

### 电脑与法律

重视 BSA 的忠告 ..... 本刊编辑部(2)

### 综述

千兵易得、一将难求 ..... 陈 炼(4)

### 电脑应用

电子数据交换(EDI)面面观(2) ..... 李凤仪(5)

用 PC-1500 制作程控电话自动计费系统 ..... 申屠文(9)

### 软件纵横

ORACLE RDBMS V5.1 系统分析及应用研究(下) .....  
..... 沈欣等(11)

### 网络与通信

高级 UNIX 连网技术讲座 第四讲 Stream .....  
..... 冯家宁(15)

XENIX 操作系统下终端打印技巧 ..... 洪系俊(18)

### 软件介绍

高效率的 E500 计算机测绘野外电子记簿软件 .....  
..... 卢耀权(19)

FOXBASE+/Mac 的独特风格 ..... 罗 辉(20)

APL 语音的特点和发展过程 ..... 王 退(22)

### 使用与维修

电子收款机基本结构 ..... 管 强(23)

ZJ3712 打印机一种非损坏性故障的维修 ..... 唐兆海(24)

LQ-1600K 打印机测试程序 ..... 刘云枫(26)

### IDEA

西文 Windows 3.1 和 SPDOS 5.1V、2.13H 汉字系统联合  
起 来使用的技巧 ..... 曹国钧(27)

解决 Compaq486 微机使用自然码输入汉字时死机的方法  
..... 吕 靖(28)

关于第三软驱驱动器的应用 ..... 彭 禾(29)

### 中文信息处理

两种流行汉字输入法之我见 ..... 郎加夫(30)

### ABC

使中华学习机具有简拼与词组功能 ..... 翁元祥(31)

LQ-1600K 打印机控制参数在 Fox 程序中的应用 .....  
..... 卜符成(32)

ARJ 压缩备份软件使用简例 ..... 陈哨海(33)

### 游戏乐园

聊斋志异指引 ..... 姚宇江(34)

“快打至尊”之“必杀技” ..... 王 智(35)

三国演义修改法 ..... 马 翔(36)

简单巧妙地修改《波斯王子》 ..... 西 华(36)

### 编读往来

几封读者的来信 ..... (37)

### 病毒防治

CMOS 和分区参数表的大敌——1107 病毒 .....  
..... 胡向东(38)

新 6.4 病毒特点及清除 ..... 秦德成(39)

Trop 病毒的自动清除 ..... 宋正英(41)

### 单片机与单板机

单片机与模糊控制讲座第七讲 MC6805 的指令系统和程序  
设计(2) ..... 余永权(43)

### 电脑用户

dBASE II、FoxBASE 命令函数标准化转换程序 .....  
..... 梁 波(45)

一个通用的音乐演奏程序 ..... 戴振喜(49)

给 SPDOSV5.1 增加六笔声形输入法 ..... 陈丁君(51)

BIOS 参数应用 ..... 晁明中等(52)

王码汉字系统(WMDOS 5.0)中几个未公开的功能调用  
..... 李志刚(54)

也谈 FoxBASE+反编译技巧 ..... 聂文广(55)

### 名厂介绍

兄弟工业株式会社 ..... (56)

### 小辞典

..... (57)

### 简讯

中国软件行业协会第二届全国会员代表大会召开 ..... (58)  
布尔电脑公司在广州设立办事处 ..... (59)  
93'中美软件合作洽谈在深圳举行 ..... (60)  
94' EPSON 电子元件展示会及应用发布会 ..... (60)  
全国计算机软件资格和水平考试函授辅导班第六期招生  
..... (60)

### 广告索引

1. 国营广州无线电厂
2. 广东省潮阳市陈店环东电子有限公司
3. 广州易通计算机应用工程开发公司
4. 广州电子设备公司
5. 特强(广州)电子有限公司
6. 赛宝星河
7. 广州科教电脑设备厂
8. 广州袖珍计算机技术服务中心
9. 京粤收款机系统
10. 香港现代电子出版社
11. 华楠建筑新材料公司
12. 广州海谊电子仪器实业公司
13. 中国计算机学会北京单片机应用技术协会
14. 广州市声乐电脑商行
15. 华力科技开发公司
16. 广州高新技术产业开发区顺达电子有限公司
17. 广州华信电子系统公司
18. 都帮机电有限公司
19. 华粤电脑工程公司
20. 清华大学科学馆
21. 广东天地广告有限公司
22. 南方软件有限公司

# 重视 BSA 的忠告

· 本刊编辑部



BSA 是商业软件联盟(Business Software Alliance)的英文简称。这一组织是在 1988 年为防止违法复制软件而成立的,总部设在美国华盛顿特区。它的任务是从事国际外计算机法律保护的宣传教育活动和调查工作。具体来说就是调查违法复制情况,围绕版权保护进行宣传教育,支援诉讼,参加知识产权国际会议等。这是一个维护软件开发者权益,教育用户提高知识产权意识的机构。近年来,在亚洲,特别是东南亚地区,展开了一系列活动,对软件盗版情况作了许多调查,发表过权威性研究结果。在 1993 年 12 月 6 日~7 日,93' 深圳中美电脑软件合作洽谈会上,BSA 向与会者散发了一份有关软件保护的宣传资料,这份资料对中国著作权法,计算机软件保护条例作了介绍,同时,还就软件侵权的一些问题进行了通俗概括的解释,这对我国广大软件工作者和用户来说,是一份很有教育意义的材料。本刊下面全文予以转载,其中方括号[]内的话,是本刊编者加进去的:

## 软件盗版及有关法律

### 有关法律

中国著作权法及其实施条例于 1991 年 6 月 1 日起施行。该法例保护的经营范围包括文学、艺术和科学作品。虽然计算机软件也在著作权法中列明为受保护作品之一,但其保护办法则在软件保护办法[指中华人民共和国国务院令第 84 号《计算机软件保护条例》]条款下另行规定。该软件保护办法自 1991 年 10 月 1 日起生效。

1992 年 10 月,中国加入了伯尔尼保护文学艺术作品公约和世界版权公约。中国与其他成员国之间已承认互相保护著作权的义务。

根据现存的中国法律体系,国内及国外软件作品均可受到保护(作者必须是国际公约或条约成员国的国民或居民,或其作品是成员国首次或同时发表)。外国的计算机程序应视为文学作品保护,而不必履行登记手续,保护期为

该作品首次发表后的 50 年。

### 侵权行为的处罚

如果发现某人或单位侵犯软件的著作权,侵权者将要承受法律责任,包括停止侵权行为、公开道歉、赔偿损失、由政府没收其非法所得、罚款等。根据著作权法实施条例规定,侵权行为的罚款可高达人民币一万至十万元,或者总定价的二至五倍。[可参看《中华人民共和国著作权法》第四十六条,《著作权法实施条例》第五十一条(二)]

侵权行为包括未经著作权拥有人许可,而擅自发表、复制、修改、翻译或发行其作品。[参看《著作权法》第四十五条]

著作权拥有人可以通过司法、仲裁及调解程序,解决侵权纠纷。[参看《著作权法》第四十八条、四十九条、五十

条,《计算机软件保护条例》第三十四条、三十五条、三十六条、三十七条]

### 身为使用者,你应该做些什么?

身为软件使用者,你的首要责任是购买原版软件,并且只供你个人使用。每一份软件只可在许可协议准许的情况下使用。[这对用户来说,是十分重要的法律与道德规范。不少用户目前很不了解这个规定。可参看《软件保护条例》第九条(三)、(四);第十八条,第十九条]如果是在商业机构,则每一部计算机应该有自己的原版软件及使用手册。在未事先取得软件公司的书面同意前,将一份原版软件同时供给多部计算机使用,或是将软件或使用手册复制而出租、分发,凡此种种行为都是不合法的。[参看《计算机保护条例》第二十一条,第二十二条]

在购买软件时,请小心确定你买的是正版软件。[软件的购买者必须查明该被购软件的来历,即使是不知道或者没有合理的依据知道该软件是侵权物品,但若该软件被查明为侵权物品时,很可能因被要求销毁而使用户遭受损失,虽然按条例可以向提供者追偿,但在时间、精力及多方面的损失是无法弥补的,参看《软件保护条例》第三十二条]许多盗版软件在外观上模仿得维妙维肖,但是品质却十分低劣。假如你使用盗版或者复制软件,除了可能面对以上涉及的法律风险外,亦可能面对其他商业风险,包括产生可能有缺陷的软件或损坏的磁片。

使用非法软件的一个最大害处是,它可以破坏重要的资料或数据。盗版软件可以将计算机软件病毒迅速蔓延至你的微机或网络系统上,以至危害你的整个商业操作结构。你的另一个责任是,只购买合法软件。因为当你购买或使用盗版软件时,你不但完全抹煞了软件创作及开发者的合法权益,同时也伤害了整个计算机软件工业。软件公司不分大小,都是实实在在花费数年时间开发各式各样的软件供大众使用。你购买原版软件所付出的金钱,有部份将成为软件公司的研究发展经费,而软件公司亦藉此提供更优良、更先进的软件供用户使用。但是,假如你购买盗版软件,那所有的钱将全放到了盗版者的腰包,开发软件的公司就连一毛钱无法受惠。盗版软件带来的不良影响包括以下几点:

- 减少了软件开发的经费。
- 降低了在当地提供售后服务的普及率。
- 阻碍了软件品质的提高。
- 破坏当地经济。由于合法零售商的销售量降低,间接减少了当地的收入和工作机会。

如果你使用非法软件,将不会得到任何质量、技术支

援或软件升级的保证。

### 盗版损失知多少

中国的软件盗版情况严重,据估计少于一成是合法取得的,其余的则是仿冒或非法复制的软件。[美国1990年认为在中国违法使用软件造成美国损失了2.90亿美元,1991年3月美国高级贸易官员在北京宣称由于中国软件保护条例未公布,到那时已给美国造成的损失达4亿美元。1993年,美国官员又宣布这个损失仍有1亿多美元]

### 政府当局决心严厉执行法律

如果你非法购买或使用软件,将可能因此而接受法律的制裁。中国于1991年颁布了著作权法及其实施细则和软件保护条例,接着的1992年参加了国际公约,可见中国当局极为重视著作权及软件保护。[中国外交部发言人12月2日发表评论时说,自中美两国政府1992年1月17日签订保护知识产权谅解备忘录以来,中国始终严格履行备忘录项下各项义务,加入了世界版权公约、保护文学和艺术作品的伯尔尼公约、保护录音制品的日内瓦公约以及志利合作公约;还根据有关国际公约的规定,重新修订了中华人民共和国商标法、专利法及其施行细则。为加强对知识产权的保护,中国法院成立了专门的知识产权法庭。中国司法机关及有关主管部门,根据中国参加的上述有关公约及中国制定的有关法律法规,严肃查处了一些侵犯商标、假冒专利、剽窃著作权的案件。]

### 你在反盗版的行动上可以获得的协助

如果你怀疑自己可能买到了盗版软件,请与商业软件联盟(BSA)联络以取得协助。他们会告诉你何处购买到合法的原版软件。如果你发现有人盗版软件,或经销非法盗版软件,也请通知商业软件联盟。地址:香港太古湾道12号太古城中心四座11楼 TEL: (852) 8653318 FAX: (852) 5606247

[我们希望我国软件工作者和广大计算机用户重视BSA的忠告:做一名守法护法的软件工作者,做守法护法的计算机用户。]

## 千兵易得

## 一将难求

## ——论计算机人员的管理

陈炼

开

发一个大型计算机大型应用软件,往往由许多计算机人员来合作完成,其中就必有“一将”来统领“众兵”。笔者长期从事软件开发,当过“兵”,也当过“将”,深深感到“一将”的举足轻重,也深深感到“千兵易得,一将难求”。现在我说说自己滥竽充数“当将”时的一些粗浅体会,恳请各位计算机同仁指正,也希望能够抛砖引玉。

## 一、“一将”应有的素质

软件开发中的“一将”,应该既是“众兵”的组织者,又是“众兵”的技术指导。两样能力都强,是最佳“将才”,偏于一方,可以培养,两样能力都差,只好“换将”了。

“一将”的精力,大致应分为三部分:

三分之一的精力用于计算机业务。软件开发,从一开始起,就有不可违背的客观规律。尊重这些客观规律,是“一将”应具备的最基本的素质。只有这样,才能正确地“对”“众兵”的工作进行统筹、分工和衔接,有效地进行软件进度的安排与考核。这些规律,多在《软件工程》中阐述了,多为系统分析员的工作,不多说。另外,提高自己的计算机业务,就能有效地对“众兵”进行技术指导,提高自己的威信。

另外三分之一的精力用于实际业务的熟悉。以财务软件的开发为例,必须既懂计算机又懂财务业务。“一将”对实际业务越熟悉,软件开发就越能有为的放矢。在许多单位,“一将”都由本熟悉实际业务,后又掌握了计算机这个工具的同志担任,因为他们组织开发的软件无论质量如何,最终总能“中的”。在此我奉劝计算机专业的同仁,无论在什么单位,都应努力熟悉本单位业务,这样才能找出软件开发项目,才能“当将”,“不想当元帅的士兵不是好士兵”。

还有三分之一的精力用于公共关系的协调,有时还不止,这有点“中国特色”。“众兵”的管理,与其他管理一样,讲究“德威并重”。对下级无“恩德”的领导难以具有感召力。还有要擅于灵活运用各种管理方法,如用谈话的方式解决思想问题等。另外要争取各界的支持,这一点很费脑筋。“各界人士”对计算机的认识不同,但他们是一个周

围环境,对软件开发工作总有不同程度的影响。耐心细致的工作做在这方面,会有意想不到的效果。

## 二、“一将”的管理特色

“众兵”的特点,是年轻化,高知识化。年轻,就好学,好强,也较易管理。“众兵”的学习气氛,对软件开发的进度,有着巨大的影响。竞争,是年轻人学习的真正动力。制造竞争,就是“一将”的方法了,另外,团结是一个值得注意的方面,知识分子有“不满”往往闷在心里,让“众兵”吐尽心声,可以维护好一个和睦的大家庭。

“一将”在工作中往往需要补充“兵源”,有时甚至一开始只有“将”,由“将”组“兵”。“兵”的选择,以高知识,逻辑思维强为准。要相信“千里马常有”,选“兵”过程中要敢于克服阻力,乃至力排众议。“兵源”中,计算机人员和业务人员最好各占一定比例。

“一将”作为“众兵”的技术指导,有不少辅导工作,这可以安排“强兵”来作,或者计算机人员和业务人员互为补充。另外,必须购置好齐全的资料。对“众兵”的劳动时间,“一将”应给予一定的自由度。程序员都有这种经历,设计时不愿打断思路,到下班了还想一气呵成,有时还可能加班到深夜。完成这件事后想休息,却又占用了第二天上班时间。“一将”应给周围环境作好解释工作,保护“众兵”的身心健康,同时和“众兵”一起注意影响。

在“众兵”的组织过程中,“一将”的“德威并重”应以“德”为主。因为,第一,“一将”往往行政职务不大,对于“众兵”的升调任免等无法说了算;第二“众兵”知识层次高,也就不会盲从,需要尊重。所以用“疏导”的办法较为有效。否则,他两个星期能完成的软件到两个月都不出来,也难以考核。逼急了就会跳槽,乃至“兵变”。作为“一将”,首先要让大家觉得跟你干有好处。如可以学到技术,可以解决一些生活上的困难等。“一将”的精力与智慧,应该用于充分发挥每一个“兵”的长处,而不是用于改正每一个“兵”的短处。

## 三、软件开发中“众兵”的群体心理过

程

“万事开头难”,软件开发刚开始时,“众兵”大都茫然不知所措,处于消极等待状态。这时的工作,应是尽快让“众兵”了解软件开发的目标以及自己该完成的任务。如果其他地方已投产了同类型软件,可组织参观学习。同时,做各种宣传鼓动工作。这一阶段是个难关,“一将”决不可有丝毫的拖拉,必须一鼓作气地渡过。

第二阶段“众兵”都已明白自己该干什么,都在干了,而且都沉醉于软件开发中,自动加班加点的情况屡见不鲜,极似一群计算机狂。这时“一将”不妨与大家“狂”在一起,一来是一种“身教”,二来对自己的计算机业务大有裨益。真正其乐无穷的是这一段时间,大家在软件设计中匠心独运,频频发现智慧的闪光。这时“一将”要及时向上级反映“众兵”的工作热情,请动“首长”们来慰问“众兵”,级别越高越好,并借机提出“众兵”生活中的一些困难,力争请“首长”们解决。另外,要防止“一粒老鼠屎,坏了一锅汤”,发现“老鼠屎”,及时清除掉,哪怕是“皇亲国戚”,也要用不同方式清除,千万不能姑息养奸。

最后一阶段是任务逐个完成,软件正式投入使用,“众兵”“干劲”消退。“一将”此时应制定一些具体的管理制度,否则连卫生都没人搞。另外还有一些零星的维护工作,需要督促“众兵”完成。然后就是“首长”们的表彰,奖励,拍板。“众兵”哪些生活困难可以解决,发展党员等等。这时“一将”多谦虚,把功劳归于“众兵”的努力,多宣传“众兵”劳苦功高,会得到许多真心的朋友。

## 四、“一将”的发现

虽然“一将难求”,但只要用心去挖掘和培养,还是“千里马常有”。“人无完人,将无完将”,十全十美的“将才”是没有的,都要在实践中培养和锻炼。一般来说,工作时间较长,逻辑思维强的优秀工作者,易成为“众兵”的技术指导。团支书等形象思维强,工作经验足的易成为“众兵”组织者,“一将”的发现,应在他们之中。

“千兵易得,一将难求”啊!

# 电子数据交换 (EDI) 面面观 (2)

中国海关管理干部学院 李凤仪

海关合作理事会 ADP 委员会推荐的 EDI 标准有:

## I. 数据元编码

数据元	推荐码
税则总目	协调系统(CCC)
海关手续	通用处理规则=1, 位码--(CCC)
货币	ISO3-ALPHA 货币码
原产地	ISOALPHA-2 码
交寄国	ISOALPHA-2 码
数量单位	ECE (欧洲经济委员会)Rec. No. 20 度量码单位
运输码	ECE Rec. No. 19 运输码模式
进口商/出口商等	通用处理规则(CCC)
货物型号、包装等	货物和包装码
集装箱标识	空运:IATA (国际航空运输协会)码 其他:ISO 6346
日期	ECE Rec. No. 7 YYMMDD

## I. 其它标准

电子信息语法规则 EDIFACT ISO 9735

数据元的描述和表示法 UNTDED (联合国贸易数据元指南) ISO 7372 国内的 EDI 标准, 牵涉到如何建立一个开放式的、符合国际标准规范的中文化应用平台问题, 要以中文信息处理技术国际化为支撑。

已制定的 EDI 国家标准有:

GB1988	信息交换用七位编码字符集
GB2311	信息交换用七位编码字符集的扩充方法
GB2312	信息交换用汉字编码字符集
GB7589	信息交换用汉字编码字符集 辅 2
GB7590	信息交换用汉字编码字符集 辅 4
GB8686	文本通讯用编码字符集
GB5271.9	数据处理词汇 09 部分数据通信
GB11383	信息处理、信息交换用八位代码结构和编码规则

GB7421	信息处理系统——数据通信——高级数据链路控制规程——规程类别汇编
GB7496	信息处理系统——数据通信——高级数据链路控制规程——帧结构
GB7575	数据通信——高级数据链路控制规程——规程要素汇编
GB12453	信息处理系统——开放系统互连——运输服务定义
GB12500	信息处理系统——开放系统互连——面向连接的运输协调规范
GB11589	分组交换数据网国际电路上的终端和转换呼叫控制规程与数据传输系统
GB11589	公用数据网和综合业务数据网的国际用户业务类别
GB2809	信息交换用目的时间表示法
GB2810	信息交换用顺序日期表示法
GB2808	全数字式日期表示法
GB9181	道路车辆分类与代码(机动车)
GB2260	中华人民共和国行政区划代码
GB2259	世界各国和各地区名称代码
GB4290	集装箱运输状态代码
GB4754	国民经济行业分类和代码
GB4880	世界语种代码
GB6582	运输方式代码
GB7408	星期编号
GB9648	国际单位制代码
GB10302	中华人民共和国铁路车站站名代码
GB6388	运输包装收发货标志
GB69446	危险货物分类和品名编号
GB12406	表示货币和资金的代码
GB10757	邮政通信网术语
GB1836	集装箱标记代号
GB1992	集装箱名词术语
GB193	包装储运指示标志
GB12410	国际航行船舶识别代码

## GB12462 世界海洋名称代码

正在制定的 EDI 国家标准有:

• EDIFACT 应用层语法规则 (等同采用国际标准 ISO 9735)

• EDI 贸易数据元目录 (等同采用国际标准 ISO 7372)

• 贸易单证样式(等同采用国际标准 ISO 6422)

• 贸易单证格式代码的位置 (等同采用国际标准 ISO 8440)

邮电部已组织技术人员,准备在青岛 EDI 设备安装完毕后,配合青岛的国家 EDI 试点工程,与用户有关单位一起,共同在终端上开发和实现中国 EDI 标准。

## 第二章 EDI 实施要点

### 一、可行性研究

#### 1、社会可行性

国内 EDI 的实施尚属通过宣传取得共识,逐步试点的准备阶段。应密切注意相关行业的实施进展,先积极创造条件逐步实施本行业 EDI (纵向 EDI)。在有条件时适时实现跨行业 EDI (横向 EDI)。

在有一定规模后可合二为一。

#### 2、技术可行性

首先必先建立本系统的计算机管理信息系统,这是应用 EDI 的首要条件。

其次 EDI 系统必须有网络环境的支持。

至于 EDI 标准,必须重视国际上 EDI 标准的发展动态,及时进行分析研究,并应着手确立国标的工作。

#### 3、经济可行性

应视本系统经济实力和外援实况,先试点,积累经验,再推而广之。

### 二、本单位准备

1、制作商业运作模型,准备坚实的贸易基础和企业内实施相应的管理、操作模式变革。

首先确定贸易目标和战略。然后从以下诸方面去研究。

- 贸易战略中成功的关键因素是什么?
- 哪些业务环节对成功起最关键作用?
- 这些关键环节需要什么信息,而且这些信息如何在经营、控制、评测及战略等水平上交流?
- 有什么可证明使用 EDI 能获得商业效益保证这种交流?
- 是否能够获得生产率效益?
- 在业务和投资方面需要冒什么样的风险?

• 需要什么科技、资源、资金、技术和时间投入? 这些投入是否比投在其他方面更值得?

• 你希望建立起贸易关系的贸易伙伴是谁? 他们能在五年之内建立起来吗?

• 实施 EDI 需要在组织及文化方面进行哪些变动?

• 为目前经营需要实行的短期权宜措施是否是以长期灵活性的计划投资为前提的?

• 对技术是否片面的强调,反而忽视了当前及未来的贸易因素?

• 商业运行及其管理需要哪些变化?

2、培养 EDI 人才,建立实施队伍。

3、建立 EDI 中心。

EDI 中心是信息交换服务中心,其主要任务为:

(1)标准格式转换

EDI 标准目前还未完全统一,各种标准都在使用,主要有美国的 ANSI X.12、TDCC、欧洲的 GTDI 和国际标准 EDIFACT 等,EDI 中心应具有各种标准的转换能力。

(2)文件处理

文件级 EDI 是将所收文件转送给对方。

文档级 EDI 是核对每一份 EDI 文件的类别、格式及长度是否符合标准,正确才传送。

(3)文件追踪

详细记录文件传输的时间及过程,提供适当的报表及查询功能。

(4)为各种不同型号的计算机提供用户前端 EDI 软件和通信介面支援。

(5)EDI 顾问服务。例如,为各种用户提出适当的 EDI 方案;帮助解决用户提出的各种技术问题和法律问题等。

(6)EDI 教育、培训和安装。

EDI 中心事实上是一个超级商情和信息中心,考虑到安全保密、信息资料的权威性和合法性、资料传递延误或错误责任以及信息资料合法使用等问题,EDI 中心由官办优于民办,或者是由政府监督下民办,不宜纯粹民办。

4、确定所采用国际标准和通讯模式,并作相应的业务改革。

标准化是实现 EDI 的关键环节。成功地在公司间交换数据的秘诀在于寻找一个双方都明了的格式,最佳的格式是数据交换后仍保存原始文件的格式,允许数据中保留一些应用时的标识。最重要的是必须选择一种稳定、使用广泛、具有通用性,有强大后盾支持并与未来 EDI 发展趋势相符合的标准。选择标准时既要考虑该标准将来的发展前途,也要考虑在本行业的“兼容度”。

5、制定 EDI 项目计划。

• 准备一份以商业运作模型中出现的重点为基础的

EDI 项目计划;

• 详列 EDI 基础设施、管理职能、支持职能、技术需要;

- 准备系统的具体规格;
- 准备谋求外部服务的标书;
- 评估标书;
- 开发系统;
- 准备销售计划;
- 实施;
- 管理及维护;
- 继续开发。

6、需求分析。

包括基础设施、管理职能、服务职能、用户现场、技术需要和人员配备。下面是设置与费用清单。

管理职能: EDI 筹划小组、开发、职员、项目计划、变化管理、资金、销售、收费及帐单、商业协议。

服务职能: 文件、培训、实施、帮助台、支持(后援)。

技术需要: 硬件、软件、保密、EDI 信箱、通讯、EDI 和其他标准。

人员: 项目管理、技术、后援、实施、销售、开发、咨询员、服务提供人员、工会代表、主要业务经理、领导。

### 三、实施

1、建立或租用 EDI 邮箱。或者订购提供 EDI 邮箱服务的 VAN(增值网)。

邮箱应具备以下功能:

- 邮件处理——收集、整理、发送与交易。
- 格式翻译。
- 自动检测——检测文件格式是否符合标准。
- 文件追踪与进展情况反馈。
- 有效和保密。
- 管理功能。使独立于网络服务商,自由地与贸易伙伴交换文件。

2、选择 EDI 传输方式。

传输方式由简到繁常用有以下几种:

• 文件采用 X.12 格式的磁带/盘/穿孔纸卡物理传递来实现 EDI 报文的传递——无风险、简单的 EDI。

• 租用线路实现点到点传输。

这种传输方式的缺点是

(1)费时,多贸易伙伴时要许多重复传送。

(2)租用线路的通信方式是同步的,即在发送 EDI 信报时要求接收方在线等待,有时由于接收方忙于与别的 EDI 用户通信,发送信报的用户只得一次一次地重发。

• 业务往来单位双方电脑直接连线。

这种方法实现起来需面对以下问题:

\* 连线对象众多,电脑通讯连线作业复杂,成本高。

\* 异种机联接困难。

\* 通信协议不同。

\* 报文标准复杂,资料格式转换复杂、耗时。

\* 运行时间不同,要考虑时差。

\* 作业效率低。

使用增值网(VAN)通信。

增值网是在公共电话网上建立的各种非话务通信网络,其中主要是进行各种应用的计算机网络。发送者与增值网建立连接,将 EDI 报文递交给增值服务中心,增值服务中心将报文经过转发投递到接收者的 EDI 邮箱中,用户也可请求在投递 EDI 报文时给收方用户发通知,收方在需要时进行读取。

下表列出目前主要的网络服务商及所提供的 EDI 服务。

例如,IBM 公司位于佛罗里达州的增值网络服务中

增值服务商 (ESP)	增值网	开始 日期	国家和 地区 (个)	服务项目
通用电子 信息服务 公司 (GEISCO)	EDI * 扩 展	1988. 11	70	零售、汽车、石 油、化学、航空、 电子、钢铁等。
Sterling 软 件公司	ORDERNET			
Mc Donnel Dompur	EDI * NET	1981.	70	政府、零售、运 输、制造、金融 等。
Comput 服 务公司	EDI 服 务网	1988.	200	银行、政府。
Shipnet 系 统公司	EDI 网络			
Telenet 公 司	EDI 服 务网	1987.	75	通信、制造、医 药、运输、食品 等。
IBM	信息服 务网		73	商业、家庭服务 PRODIGY 系 统
数据控制 公司	REDINET			
Martin Marrietta	EDI 交换			

心,拥有 15 台 IBM 3090 大型机,网络服务遍及 73 个国家和地区,IBM 提供的 expEDite 系列软件已通过该中心的用户邮箱服务(X.40)完成贸易伙伴间的 EDI 标准格式的数据传送。

这种方法只是一个部分的、权宜的解决办法,同一个

增值网公司的客户之间实行 EDI 易,而增值网络公司进行网络互连就难,经济上很难订立一致的、双方客户都满意的价目表,技术上 EDI 报文在两个增值网之间进行传递时很难实现状态指示服务,而且增值网只实现了计算机网络的下层,相当于 OSI 参考模型的下三层,而 EDI 的通信发生在不同的、往往是异种计算机的应用进程(EDI 系统之间,这就决定了 EDI 的应用进程与增值服务公司的联系相当松散,效率很低。增值网提供的 EDI 服务功能有一定的局限性而且缺少跟踪能力。

#### • 使用 MHS(文电作业系统)通信

MHS 是一个存贮转发的异步通信系统,它遵从 X.400 协议,X.400 协议规定到应用层,因此 EDI 应用进程与 MHS 系统接口相对容易和简单。MHS 是应用层最为丰富、全面的标准应用窗口,MHS 完整的开放系统互连标准、广泛的接受程度、灵活的通信方式、存贮转发方式和多项安全服务项目,为 EDI 业务提供了较理想的异步通信环境和转发代理的支持,适合 EDI 不需实时传送等特点,使 EDI 的通信只要在相邻的节点间建立会话,无需在收、发文电的端计算机之间建立端到端的实时会话。1991 年 3 月 CCITT(国际电报电话咨询委员会)批准了 X.435 协议,该协议支持 EDI 应用,定义了一种 EDI 文电系统(EDIMS),并定义了一个新协议——PEDI,规定了如何在 X.400 中收发 EDI 报文,且把 EDI 视为 MHS 的一个组成部分,支持 EDIFACT 标准,亦可自定义和引用其他标准,是新一代 EDI 系统出现的标志。

对 MHS 环境来说,EDI 只是一种新的业务,而非一种

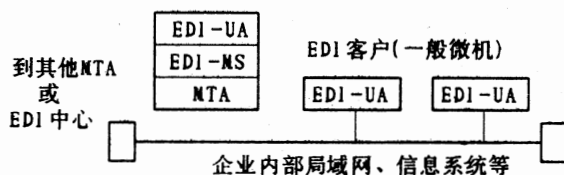
技术投入少而社会效益和经济效益大,欧美各国均在以官方形式引导用户利用 X.400/MHS 开展 EDI 业务,并开展把二者集成一体的研究,预计近年内 74% 的 EDI 将与 MHS 集成为一体,MHS 必将成为 EDI 通信的主流。

EDI 报文在 MHS 系统中的传递过程如上图所示,图中 EDIMS 为 EDI 在 MHS 中的传递系统或称为 EDI 报文系统,EDIMSUA 为 EDIMS 中的用户代理,MTA 是报文传输代理。

#### • 基于客户/服务器(Client/Server)的 EDI 系统

客户/服务器是一种基于计算机网络的数据处理模式,具有数据处理和应用分布的结构。客户发出处理请求,通过计算机网络传输给服务器,由服务器进行处理,处理完毕向客户返回处理结果。服务器主要实现集中的数据管理、共享服务和共享应用等功能,客户主要实现图形用户界面、预处理数据和各自的应用等功能。系统配置结构如下图:

#### EDI 服务器(工作站或高级微机 etc)



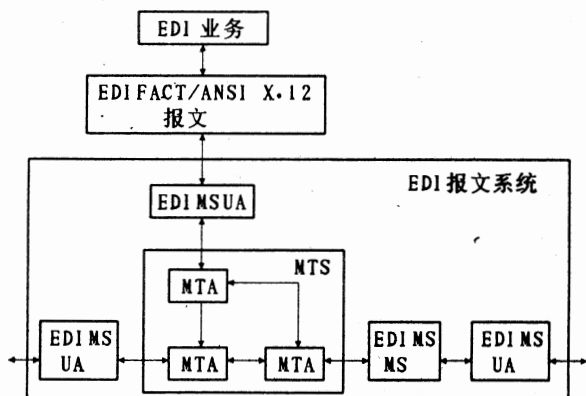
图中 EDI-UA 为 EDI 用户代理,它提供用户界面、帮助用户构成 EDI 报文;EDI-MS 为 EDI 信报存贮单元,它存贮和管理 EDI 报文;MTA 为报文传送代理,它可靠传送邮件。

EDI 服务器由 EDI-UA、EDI-MS、MTA 组成,它完成与 EDI 贸易伙伴或 EDI 中心的通信,实现 EDI 标准票据(如定单、发票、回执等)与 EDI 标准报文之间的转换,符合 CCITT X.435 建议。

EDI 客户机通过局域网上的 EDI 服务器完成信息转换和向外通信。

国外还在发展用光纤进行通信。

无论 EDI 传输类型如何选择,要应用 EDI 技术就要求有高水平的通信和各方面协作。(待续)



新技术,在已有的 OSI 基础结构中纳入或开发 EDI 应用



## 一、前言

随着经济改革的不断深入以及通信技术的不断发展,使用程控电话的单位和个人越来越多,而使用程控电话的费用非常可观,自然就涉及到一个计费问题。目前,在邮局使用的大多是可用于

多路控制的大、中型自动计费系统,通常由 IBM PC 或兼容机以及接口电路组成,整套系统价格昂贵。对于单位和个人以及收费电话亭,有人工计费和自动计费两种方法。人工计费既费事又不准,已越来越少;自动计费则大多购置市场上出售的专用程控电话自动计费器,有几种类型,分别由不同单位研制生产,但是,它们的性能价格比不高,只能显示,不能打印和贮存,更不能打印月报表。

本文将介绍使用几乎已淘汰的 PC-1500 机作为主控机器,外加少量的接口电路即可购成一套多功能的完整的自动计费系统,接口电路成本约为 30 元左右,下面对其硬、软件设计及基本操作过程作一般的介绍。

## 二、硬件构成

## (一)与计费有关的音频信号

在程控电话线路上跟计费有关的信号有以下二个:

1、被叫用户回答信号。信号频率为 2600HZ。在打程控长途电话时,当线路接通,对方(被叫用户)摘机时,由邮局交换机送来的信号,它是单脉冲信号,持续时间 150ms。这个信号也叫计费信号,在接收到这个信号时开始计费。注意,市内电话接通过无此信号。

2、回铃信号。信号频率为 450HZ。发出的是“嘟、嘟……”各长 1 秒、间断 4 秒、节奏较缓慢的断续音。

由于市内电话接通时无 2600HZ 的计费信号,因此判断对方是否摘机的方法与长途电话不同,相对而言要复杂一些,本人采用的方法是,当 450HZ 的回铃音消失的持续时间大于 4 秒(即大于二个回铃音之间的间隔时间),且叫对方用户还未挂机时,即开始计费。

## (二)接口电路

接口电路的作用是将电话线路上的各种音频信号、当前所处的状态等信息通过各种交换处理后送入 PC-1500 中,接口电路图见图(1)。主要由摘机、挂机检测电路、双音频拨号方式键码获取电路、2600HZ、450HZ 信号检测电路和 PC-1500 数据输入接口电路四部分电路组成。下面分别作一简要的说明。

## 1、摘机、挂机检测电路

程控电话机在挂机时,线路正、负端直流电压为 48~60V。摘机后下降为 5~7V。在免提方式下又拿起话筒时为 11~13V。见图(1),三极管工作在开关状态,此时  $V_A \leq 0.3V$ , 输出为低电平。摘机后,由于  $V_B \leq 0.3V$ , 三极管处于截止状态,此时  $V_A$  为高电平。

## 2、按键检测电路

目前程控电话有二种拨号方式,脉冲拨号方式和双音频拨号

# 用 PC-1500 制作 程控电话自动计费系统

申屠文

方式,这可以

一个开关来转换,但由于双音频拨号比脉冲拨号快得多,因此一般都使用双音频拨号,本文的电路是针对双

音频拨号方式而设计的,如果是脉冲拨号方式,键码获取电路将

简单得多,由于使用很少,这里不作介绍。

双音频拨号方式键码获取电路主要由双音频解码集成 MT8870 及其外围电路组成。双音频拨号是由拨号器将 16 个数字和符号以两个高低音频信号于复合编码发送,当 MT8870 接收到程控电话机送出的双音频信号时,内部电路对其进行放大滤波,将高频群信号和低频群信号分开并将其分别与标准的信号作比较,如果有效,则将 15 脚置为高电平,并将两有效信号解码为四位二进制码,输出的二进制码分别对就 11、12、13、14 脚。

## 3、2600HZ、450HZ 信号检测电路

这部分电路由 2 块 NE567 集成电路、一块 74LS74 集成电路及外围电路组成。

在图(1)中,上面一块用于检测 450HZ。下面一块用于检测 2600HZ。NE567 是一块单音 PLL 解码集成电路,平时 NE5678 脚输出高电平,当 3 脚的输入信号频率与其内部压控振荡器的频率相同时,8 脚输出低电平。内部压控振荡器的中心频率由连接 5、6、7 三脚间的电阻和电容调节。对于 450HZ 频率的调节较简单,可将电路连接到实际电话线路上,以便调节,在线路上出现 450HZ 信号时(忙音、回铃音都为 450HZ)。边调节可调电阻,边检测 8 脚输出状态,一直调整到 8 脚输出低电平为止。实测表明,当 6、7 脚电容为 0.047 时,5、6 脚间的电阻约为 49K。对于 2600HZ 的调整,通常要借助示波器或者音频发生器来完成,因为线路上的 2600HZ 是个持续时间很短的脉冲信号,它每次只有在接通电路后对方拿起电话的瞬间才出现。本人是借助示波器完成的。74LS74 是双上升 D 型触发器(有预置、清除端),它的作用是将 NE567 的输出信号保存起来,以便让 PC-1500 机在需要时取走。其中 4、10 脚为输入端,5、9 脚为输出端,输出到 PC-1500 中,9、13 脚用于复位清除。

## 4、PC-1500 数据输入接口

这部分电路主要由 74LS138 和 74LS244 组成,74LS138 用于地址译码,74LS244 是三态缓冲器,一次要读入 8 位数据。

为了简单方便,将 # 区的 01 单元作为输入数据的存贮单元,因而可将 ME1 接高电平,这样把 # 区的输入输出接口地址的范围缩小到 0~16K 之间,译码输入端分别接  $A_0$ 、 $A_1$ 、 $A_2$  三根地址线。三态的控制端接在地址译码电路的  $Y_0$  上。

## 三、软件设计

该系统主要由初始化,按键读取及计费信号检测计时、结果输出四部分组成。程序简要框图见图 2。

PC-1500 机可用 BASIC 编程和机器语言编程,用 BASIC 编程简单、容易调试和修护,适用于速度要求不高的场合。该系统的初始化、计时、结果输出部分都是用 BASIC 编写的。按键读取

部分是用机器语言编写的,且必须要用机器语言编写,否则,当使用自动拨号功能键重拨电话号码时,速度跟不上。

由于篇幅有限,程序清单略。

## 四、抗干扰措施

由于电话线路上各种信号干扰,电压波动以及 PC-1500 主机和电池失效等因素,容易破坏 PC-1500 机的程序,造成系统瘫痪。为此必须要对 PC-1500 机中的程序加以保护,最有效的办法是将扩充存储模块置为只读方式。CE-159、CE-161 是使用最多的 8K、16K 存储模块,内部有了锂电池,模块背面有一开关,当开关拨到只读方式时,只可读不可写,且将模块从机器拿出也不会使程序和数据丢失。但要正确地做到只读,除了设置开关外,还必须对某些单元作必要的处理才行,关于“只读”的设置方法,报刊、杂志上都未介绍过。具体设置方法如下:

1. 将存储模块设置为非“.”位置,即 RAN 方式。
2. 键入 NEW 256 (回车)
3. 键入 POKE 0.85.0.1.0.0.0.0.255
4. 输入程序,或从磁带读入程序。

5. 输入以下程序:

```
65273: "="; IF STATUS 2 > 16377 THEN 65278
```

```
65274: FOR A=STATUS 2 TO 16377
```

```
65275: POKE A.255:NEXT A
```

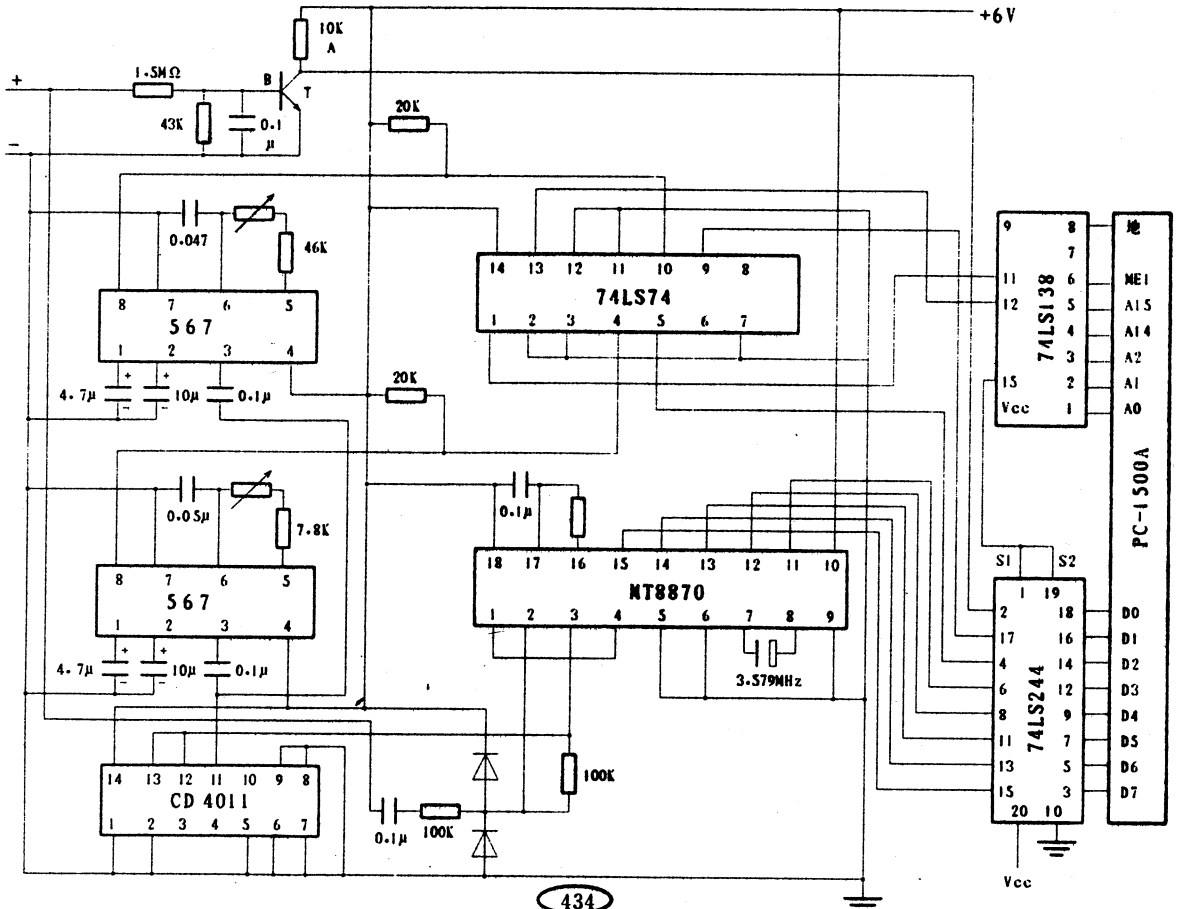
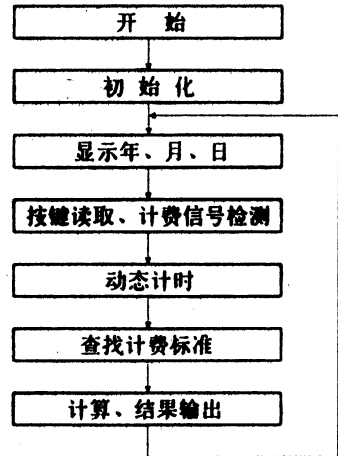
```
65276: POKE &3FF9.254.255.3.241.142.13.255
```

```
65277: END
```

```
65278: PRINT "MEMORY OVER!":END
```

6. 关机,将模块背面开关拨到“.”位置。

7. 开机,CL



# ORACLE RDBMS V5.1 系统分析及应用研究

(下)

上海邮电管理局计算中心 沈欣 杨峰

上海科技大学计算机技术系 陈霞萍

## 2.4 表的索引和聚簇技术

索引和聚簇技术是 ORACLE 改善 DB 性能的两种重要工具,尤其对提高连接查询的速度最为有效。利用这两种工具至少存在两种方法,只要重新组织编排 SQL 查询语句,可以获得同样的结果,但产生不同的性能效果。一方面,ORACLE 用户可以在自己的表上建立适当的索引或将几个表聚簇起来以改善 ORACLE 性能;另一方面,索引和聚簇使用不当却又会给今后的维护带来困难,增加维护负担。

### 2.4.1 索引

#### 2.4.1.1 ORACLE 的索引结构

ORACLE 内部大量使用了索引技术以加快查询处理的效率,它采用著名的 B+ 树索引方法。B+ 树索引是一种索引顺序结构。在 ORACLE 的 B+ 树索引中,上层的块包含索引数据和下层索引块的地址,最低层索引块中则包含每一个被索引的数据值及其对应的 ROWID。使用 ROWID 值就能定位实际的行。对于一个唯一索引,每个数据值就对应一个 ROWID;对于非唯一索引,一个数据值可以对应任意多个 ROWIDS。索引同时也是按索引值的 ROWID;对于非唯一索引,一个数据值可以对应任意多个 ROWIDS。索引同时也是按索引值的 ROWID 来排序。

#### 2.4.1.2 索引的作用和原则

索引最主要的作用就是能加快信息检索速度,减少磁盘 I/O 次数,特别是在做多表连接时,更应当使用索引改善性能;另外,通过创建唯一索引能保证表中每个记录里的一个字段或字段组合的值是唯一的。

对表拥有 INDEX 特权的用户都可以用 Create index 命令为该表创建索引,其中 PCTFREE 参数表示最初建立时 B+ 树索引每个叶结点的装填程度,一个表上可以建任意多个索引,但索引越多,则在表内数据作更新时为维护索引所需的开销也就越大。因此,用户应当在加速查询和降低更新速度之间作出权衡。对于一个供查询只读的表,较多的索引可能有好处;但对于一个频繁更新的表来说,索引少一些更为合算。通常是在表内装有数据之后再建立索引。

并置索引是建立在一个表的多个字段上的属性索引,它能使多字段的组合有唯一值标识表中记录,对于 where 子句含有多个条件的 SQL 语句,利用并置索引还可以加快查询速度。虽然在 create index 命令中并置字段的先后次序与他们在表定义中的先后次序无关,但次序的选定会影响到性能,应当将预计最常用的字段放在索引的最前面。如果 where 子句中指出的字段与并置索引中的字段完全相同,且次序一致,则并置索引能

有效提高查询速度;如果 where 子句中的字段只用到并置索引的开头几个字段,则该并置索引虽然仍被使用,但不如上面那么有效;如查 where 子句没有指明并置索引的第一字段,则该并置索引不能被使用。

#### 2.4.1.3 索引的压缩

索引的方式有非压缩型和压缩型两种,为了节省存储空间,一般把索引中索引字段的值压缩后加以存放(缺省值 compress)。索引值中实际被存储起来的只是它的一部分,这一部分足以把该值与前一个值和后一个值区分开来。压缩型索引值由于大幅度节省存储空间,从而可以增大 B+ 树中每个节点存放的索引项数,使 B+ 树有较少的层数,减少 I/O 次数,索引压缩的方式又有前部压缩和后部压缩两种。

为了执行压缩以及在检索时又要对压缩值进行展开,因此又增加了处理开销,压缩型索引中每个列约需要 10 个字节;非压缩型索引中每个列约需要 17 个字节,还要加上索引键值的开销。

#### 2.4.1.4 索引的使用和抑制

几乎在任何情况下,使用索引进行数据库访问要比使用全表扫描访问效率高,这是因为大多数 SQL 语句都只是抽取表中的某些特定行而不是全部行。如果预先知道查询要选出的行数大于全表记录的 25%,则采用全表的扫描反而比用索引快。

在下述情况下,ORACLE RDBMS 将使用索引:

(1) 索引字段直接出现在 where 子句中。例: where DNAME='DEVELOPMENT'

(2) 索引字段不被函数或表达式所修饰。该法则有一个例外,即当查询包括一个表达式,这个表达式含有 {MIN/MAX} (<col>{+|-}<常数>),并且没有别的列时。例: select 2 \* MAX(sal+1) from account.

在下述情况下,则不使用索引:

(1) 没有 where 条件子句。

(2) 索引字段被函数或表达式修饰。

(3) where 条件中的索引字段被 ISNULL 或 IS NOT NULL 修饰。

与索引有关的数据字典有 indexs 和 sysindexs。他们记录了用户表和整个系统全部表上所建的索引的信息。另外,ORACLE 还为用户提供了索引检验功能,即 Validate index 命令能检查出非法的索引格式或坏的索引结构,以及发现索引表目与行数据不匹配或存在坏的索引数据等异常情况,但不能查出没有对应的索引表目的实际数据行。一旦发现建立的索引不正

确,用户应重建这个索引。

## 2.4.2 聚簇

### 2.4.2.1 聚簇的作用

聚簇是 ORACLE 数据库存储数据的另一种特殊方法,它对于提高涉及几个表的连接查询性能作用显著。聚簇的作用就是把那些逻辑上经常在一起被查询的数据,使在物理上也临近存放,这样既可以减少数据的搜索时间和计算时间;又因为多个表共享公共的数据部分而减少了存储空间。聚簇的一组表都含有一个或几个公共的字段,即簇键或簇键的,他们的值就是表聚簇的依据。通过把这样的表聚成一簇,从而把面向索引的检索转化为面向物理的检索,加快了检索速度。

### 2.4.2.2 聚簇数据块

聚簇是将多个表的数据存储到同一个物理块上,使每个不同的聚簇码都存放在不同的块中。如果和一个聚簇码相关的数据较少,则占用整个 ORACLE 物理块就很浪费,为此 ORACLE 允许在其物理块中再划分成若干个逻辑块,每个逻辑块用来存放不同的聚簇码的数据。

一个簇块的开销是:每个物理块需 44 字节;每个逻辑块需 32 字节,每一个物理块可含若干个逻辑的簇块。若每个物理块只含一个逻辑块,那么开销是 76 字节;若每个物理块含二个逻辑格式,开销则是 108 字节,以此类推。

逻辑块的尺寸可以在 Create cluster 命令建簇时用 size 参数指明,缺省值为 ORACLE 块尺寸减去块头开销。显然,逻辑块尺寸直接和被聚簇的行数以及它们的平均长度有关,同时也要考虑为每个聚簇码的更新和插入留出一些空间。原则上尽可能使用较小的逻辑块尺寸。但如果逻辑块太小,当和聚簇码相关的数据太多以至一个逻辑块装不下就需要建立链接块拉链,由于链接块中的码值都是一样的,所以链接块中的记录不会在索引中出现,这样查询它们只能是沿着链逐块搜索,降低了速度;当然,逻辑块又不能太大,否则浪费空间太多,因为不同的聚簇码需要占用不同的逻辑块。总之,应当在平均逻辑块尺寸和最大逻辑块尺寸之间选定一个尺寸,从而在块内空间浪费和建立链接块之间作出权衡,尺寸逻辑适当将有助于空间利用率和系统性能的提高。通常,一个 ORACLE 块最多分成 6 到 7 个逻辑块,每个逻辑块的尺寸不应小于 300 字节。必须注意 size 参数所指出的仅仅是用户的逻辑块尺寸期望值,ORACLE 将根据 size 尺寸作适当调整调整尽可能用用户要求的分配,一般是微调大一些,使这个尺寸等于物理块尺寸的因子。用户可以通过数据字典 CATALOG 了解用户请求的逻辑块尺寸和 ORACLE 实际分配的逻辑块大小。

### 2.4.2.3 聚簇的创建和删除

用户在 create cluster 命令中指定聚簇使用的空间定义、逻辑块尺寸和为多个表使用的聚簇码。其中,聚簇码可以由一个或多个列组合而成,最多可聚簇 16 个列,若被聚簇的各个表有一个以上的公共列,则应把聚簇码形成并置的簇键,但簇键列的选择应使在每个聚簇码值下既不能包含太多的行也不能

太少,否则就会浪费空间或导致链接过多而降低性能。由于 ORACLE 在聚簇码上自动构造和维护一个索引,所以用户不必再用 create index 命令为构成簇键的列建立索引,除非想使簇键上的索引成为一个并置索引的一部分或成为簇中一个表的一个唯一索引。另外,当一个表中有一列或几列相同值出现频率很高,且又经常在查询中使用。则也可以为单表建聚簇,这时聚簇的作用类似于倒排表的作用。

一个聚簇被建立后,可进一步用 create table 命令在聚簇中创建表,即指明该聚簇中将它包含哪些表。注意表中的聚簇码必须与聚簇中的聚簇码在类型和尺寸上完全匹配,且在每个表中至少有一个簇键更在 create table 中被定义为 not null。表被聚簇时可能含或不含数据,最好在装入数据前就先建好聚簇并加入空表,这样可以避免以后将表中数据从非簇块搬到簇中去,因为这时的数据移动使用大量的 BI 文件开销,如果要将一个现存表装入到聚簇中去,且该表当前允许 NULL 值,那么在表装入之前必须先使用 alter table 命令修改表定义属性为 not null。表数据加载到聚簇中后,表中对应于簇键的列数据在今后维护中可以列新,但由于数据的存储位置依赖于簇键,所以改变一个记录的簇会导致该记录的物理重定位。若要删除一个尚已存在的聚簇,必须先删除或复制该簇中的所有表,然后才能删除这个簇并回收空间。

## 3. ORACLE 的查询优化策略

ORACLE V5.1 采用人工智能查询优化技术来改进查询性能,它使系统始终具有最佳存取路径的能力。并充分利用所有可用的索引或聚簇实现其优化。当然,这种优化技术随着新版本的推出也在不断研究和开发完善之中。

### 3.1 优化 select

ORACLE 对于单个小表的查询,通常采用快速的全表顺序扫描,不必要使用索引;而对于多表连接查询或搜索大表时,则会利用索引或聚簇来减少查询时间,尤其是在 where 子句中的列是索引列或聚簇列时,搜索速度会更快。当在 where 子句中出现许多列名时(包括索引列、聚簇列、普通列),ORACLE 则遵循下面一组规则决定哪个 where 条件应首先加以测试成为驱动条件来驱动该查询,这些规则是:

3.1.1 有索引的列要比没有索引的列快。

3.1.2 唯一索引要比非唯一索引快。

3.1.3 "ROWID=常数"要比任何其他条件搜索都快。

3.1.4 限定范围要比未限定范围快。

3.1.5 模式匹配 like 'X%' 要比 like '%X' 快。

3.1.6 非压缩索引要比压缩索引快。

ORACLE 查询优化器实际选用的查询路径,是采用评分方法对每种可能的存取路径进行权衡比较得出的最优路径。不同查询路径的得分表,得分最低的就是最快的路径。每个由 and 连接起来的子句和每个连接(包括内连接和外连接)都将分别被评分,连接的评分是根据不用索引而被连接的表的个数以及笛卡儿积的个数来进行的。如果得分相同,那么就象没有索引

时一样, ORACLE 用 from 子句的最后一个表作为驱动表, 因此应当把具有最少满足行的大表放在 from 子句的最后部。

### 3.2 优化 NOT

ORACLE 对 where 子句含有不等子句(!=或 NOT=)的谓词不使用索引, 但仍可以在该子句的其他谓词上使用索引。这是因为通常在带 NOT 修饰的谓词查询中, 满足条件的行数要比不满足的多, 从而做全表扫描反而比利用索引快。当含 NOT 的谓词与其他操作符连用时, ORACLE 将自动把该谓词尽可能转换成一个可用上索引的谓词。例如: NOT<转化为>=; NOT<=转化为<=; NOT>=转化为<; NOT<=转化为>。

### 3.3 优化 OR

在某些情况下, 使用 OR 子句的 SQL 语句转为 OR 的 IN 子句可用得上索引。一般只有在 OR 谓词中的列都已被索引的情况下, 才利用索引进行优化, 否则, 即使部分列被索引了仍可能不出现优化。例如: where deptno=10 or Job='CLECK' 当在 deptno 和 job 上都建有索引, ORACLE 才会使用两个索引并“合并”两具查询的结果, 此时, 最好把针对性最强的索引放在谓词表的最前部, 而把需要扫描大部分记录的谓词放在最后面, 使“不等于”测试的次数最少。

### 3.4 优化 GROUP BY

应当尽可能将选取条件放在 where 子句中, 而不是 group by 的 having 子句中, 即尽早地淘汰那些已不满足 where 条件的行, 而不必等到 group by 分组后再用 having 条件消除成组的行, 避免不必要的处理时间。例如: select job, arg(sal) from emp where job!='PRESIDENT' and job!='MANAGER' group by job 和 select job, arg(sal) from emp group by job having job!='PRESIDENT' and job!='MAWAGER' 两个查询中, 前者的执行明显比后者快。

### 3.5 优化连接

连接查询常常是影响速度的重要原因。为此一般只在连接列上建有索引的情况下才能做连接。研究表明: 在无索引的列上做连接, 宁可先在该列上建索引的连接快, 所以优化连接的一般规则是不仅应该在连接子句中用到的列上建立索引, 再做连接也比直接做无索引, 而且应尽量在编写 SQL 语时利用索引, 不要抑制索引的使用。对于连接到上无索引的查询, ORACLE 实施一种“分类/合并”例程, 先要把要排序的数据在内部分类或几组排序路线, 然后再把这些路线合并成最终的分类结果, 即先做排序, 后做连接, 如果被连接表中只有一个表有索引可用, 则通常使用另一个没有索引的表作为查询驱动表。

### 4. ORACLE 的内部封锁机制

为了在多用户系统中保证数据库的一致性和完整性, ORACLE 采用“封锁”方法来控制并发存取, 它把事务作为一个保持数据库数据完整性的单位, 每个事务总是为了某种“资源”(某种客体, 比如一个表)申请封锁, 事务结束后解锁。ORACLE 的封锁功能是自动完成的, 无需用户干预, 但同时也为有

经验用户提供封锁命令(locktable)以调整缺省的封锁。

### 4.1 ORACLE 的封锁方式

#### 4.1.1 共享方式 SHARED

当一个进程为了查询不变的数据, 希望阻止其他用户修改正为本进程所读取的数据客体时, 则以 SHARED 方式获得一个锁并封锁该客体。共享方式锁不能防止其他用户对共享更新锁的获得, 但能阻止持有共享更新锁的进程作更新。共享方式锁只能通过显式命令 locktable<表名>in shared mode 获得, 而释放该锁可以通过事务的 commit、rollback 或 logoff 等实现。

#### 4.1.2 其他方式 EXCLUSIVE

当一个进程为修改一个客体而不准其他进程以共享方式封锁或修改该客体时, 则以 EXCLUSIVE 方式获得一个行级封锁。排它方式锁可有效防止其他用户更新本进程正在读取的数据, 但他们仍能查询数据, 即使在他们查询时, 那些数据已以发生改变, 但已改变的数据并不会影响查询的结果(这是通过 BI 文件来实现)。排它方式锁的重要作用在于确保任何时候只能有唯一的进程正在更新表中的数据, 该方式锁既可以用命令 lock table<表名>in EXCLUSIVE mode 显式获得, 也能由 DML 语句(insert, update, delete)隐式得到。

#### 4.1.3 共享更新方式 SHARED UPDATE

如果一个进程想预先设定对一行或几行进行更新的权力, 以便将来某个时刻更新使用, 则以 SHARED UPDATE 方式获得一个锁。共享更新仍允许其他用户进程查询表, 或以共享更新方式封锁该表, 但它通过设定个权力实现行封锁, 可以阻止不使用行封锁的更新或企图以排它方式封锁表, 有效保证了将来能对封锁的数据行完成更新。共享更新锁既可以用 lock table<表名>in shared update 显式获得, 也可执行 SQL 语名 select for update 隐式获得。例如: SQLFORMS 中就采用了共享更新方式锁。

当一个进程准备申请一个封锁时, 若已存在的封锁不准新申请的封锁共存, 或在显式申请一个封锁时未选用 nowait 参数, 则该进程等待封锁, 直至某个进程释放了它所需要的资源后才能从等待封锁转化为实际封锁。等待着同一个资源的多个锁将按 FIFO 原则排队获得该资源。

### 4.2 ORACLE 的锁类型

ORACLE 中封锁按照封锁的资源或对象不同分为三大类。

#### 4.2.1 内部锁

内部锁用于保护 ORACLE RDBMS 的内部结构, 其封锁对象包括: 用于保护 ARH 通讯所要求的数据结构, 保护 BI 文件中的磁盘块、保护 SGA 中常驻高速缓冲区的一些数据字典数据和保护记录活动的数据库操纵/定义事务的数据结构等。由于这些结构用户是不能访问的, 所以一般兴趣不大。

#### 4.2.2 字典/语法分析锁 DDL

DDL 锁是由 SQL 的 DDL 语名引起的, 其封锁对象是数据

字典。它把改变数据库对象的操作(表定义、列定义、存取权限等)和改变数据的操作分隔开,DDL锁是ORACLE在调用语法分析时隐含获得的,所以用户不能以明显方式获得它。DDL锁可分成三类。

#### 4.2.2.1 字典操作锁

当执行对数据字典有影响的操作时(create/droptable、index、view、cluster或在表中增加数据区间/索引区间),以排它方式获得该锁数据字典、从而保证在任何时刻只出现一个字典操作和存在一个字典操作封锁。

#### 4.2.2.2 字典定义锁

在任何时刻都可能存在几个字典定义锁,它们可以是共享方式也可以是排它方式。当某进程在执行DDL或DCL命令时,以排它方式获得该锁,可以防止进行在进行面向用户的字典操作又进行语法分析而不致造成在查询字典同时有某个表的结构被改动;当用一个用户作查询或DML(insert、update、delete)语法分析时,以共享方式获得该锁,当一个用户获得一个字典定义锁后,一般总是要获得一个字典操作锁,然而,获得一个字典操作锁的用户,却未必要一个字典定义锁。

#### 4.2.2.3 表定义锁

表定义锁可以是共享方式,也可以是排它方式。当用户修改表定义时,以排它方式获得该锁;而当用户在一个SQL语句中引用表时,则以共享方式获得该锁。实际上,每个活动着的SQL语句都获得共享方式锁,这种锁的作用在于防止当正有一个活动着的SQL语句作用于某表时,字典中与该表有关的项目又被修改。例如:当某个SQL语句在语法分析中涉及到某个表时,则不允许进行有关该表的字典操作(alter、create、drop),必须稍后等封锁解除了再作尝试。

#### 4.2.3 表/行锁DML

DML封锁作用是为了保证数据的一致性,其封锁的程度可以是表或行。表级封锁确保了一致性,但降低并行性;行级封锁则既确保了一致性,又改善了并行性。DML锁可以显式地由一个进程获得,也可以通过某些SQL语句以隐含方式获得。注意行级封锁不能通过事务rollback释放。

##### 4.2.3.1 共享方式表级封锁

只能用lock table<表名>in share mode显式获得,用防止其它用户对封锁的表进行更新,但仍可以读取,避免在两次查询之间对表作更改。

##### 4.2.3.2 排他方式表级封锁

利用DML语句能以隐含方式获得该锁,也能用命令lock table<表名>in exclusive mode显式获得。其作用是禁止其它用户获得一个共享方式锁或执行任何DML事务(因为DML事务要求获得排它方式锁),但此时仍允许其他进程读表,只是其中包含的数据未必是当前最新的。如果某个更新事务要涉及多个表,则使用排他方式锁可减少死锁的发生。

##### 4.2.3.3 共享更新方式行级封锁

行级封锁通过执行命令select...from 表名 for update of

列名显式获得。它确保打算被更新的行,在取到这行到该行进行更新之间这段时间内不被修改,因此行级锁永远是排他方式锁。行级封锁发生在共享更新方式之中,实际上,行级封锁与共享更新方式是一样的,确切地说若用select...for update来申请行级锁,那么就处于共享更新方式下。因此,用户可按两种方式来请求共享更新方式,即select...from或locktable<表名>in share updatemode命令显式申请它。如果某个用户以共享更新方式封锁了一个表,那么其他用户若要对表进行插入、更新、删除也必须对该表使用共享更新锁才行,否则要等待到该共享更新方式锁被释放为止。尽管都使用了共享更新方式,但实际上是不允许有两个用户在同一时刻更新同一个表。在真正进行更新时,仍以排他方式封锁该表直至该事务被提交或回退。

#### 4.3 死锁的检测

通常,用户无需以明显方式封锁任何表,这是因为任何形式资源独占都会伴随死锁的危险,即使封锁也只能对属于自己的表或具有select权限的表进行封锁,但不能封锁数据字典。ORACLE能自动检测死锁,并根据已修改的数据块个数把完成工作量最少的最年青的事务撤消掉,从而释放该事务所占用的全部资源,使其它进程继续工作。通常,为避免死锁,可以让访问同一一些表的各进程都按同一顺序来封锁这些表。

ORACLE封锁的总目标应该是从提高系统性能角度出发,尽量减少资源竞争以增大吞吐量,即使从开始更新动作到提交或复原之间的时间极小。为此,提出关于封锁的一些建议:

4.3.1 对于update和delete操作,因为在提交之前可能有一段时间,此时就应封锁欲改动的记录行,在完成修改后立即提交。

4.3.2 如果当前有许多进程正利用共享更新方式进行更新(如SQL\*FORMS),则不要使用共享方式锁或隐式锁,而应该采用共享更新方式锁,这样其它用户就能使用行级封锁,以增强并行性。

4.3.3 尽可能把应用设计成对一个表进行访问的各进程都使用共享更新锁,从而提高并行性。

4.3.4 在应用负载较高时,不要对基础数据结构(表、索引、簇、视图等)进行修改。

#### 参考文献

1. 《计算技术》编辑部:《ORACLE关系数据库管理系统》
2. 中科院希望高级电脑公司:《ORACLE关系数据库系统教程》
3. 国家统计局计算中心:《ORACLE关系数据库系统说明》
4. 《小型微型计算机系统》编辑部:《ORACLERDBMS实用手册》

# 高级 UNIX 连网技术讲座

## 第四讲 Streams

昂立自动化工程公司 冯家宁

### 4.1 引言

让 UNIX 适应新的硬件环境或让新的硬件适应 UNIX 的基本做法主要是提供适当的设备驱动程序。如果新的硬件与旧的相似,那么现有的驱动程序是很容易修改的。驱动程序的开发只需考虑数据与硬件设备的通信方式和异步信号,如中断的通信。由于设备驱动程序实现系统调用且常常要响应非常低级的中断,它们必须与 UNIX 内核有密切的联系。一般它们都编译进内核代码里或在自举时与内核相连接。随着窗口系统的流行,一种新型的驱动程序,通常叫软件驱动程序已经兴起。它不与硬件直接接触而是与另外的驱动程序接触。这样一种伪驱动程序为几个进程提供一个接口就象一个通常的电传机设备可以实际上既是真实电传机的接口也可是网络的接口,这样多个用户进程能输出到一个物理的连接上,反过来把一条连接分给多个用户进程也行。

类似地,在网络方面,新型的软件驱动程序也在兴起。它位于用户系统调用或甚至内核调用与各种网络设备之间。软件驱动程序是一种伪驱动程序,因为它并不与硬件直接打交道。它通常实现一族网络协议里的一层或几层,或是作为不同机器或设备的数据表示的转换的过滤程序。由于网络协议分为各层协议,我们可能要把几个软件驱动程序造起来,其中每一程序处理协议的一层。这样的软件驱动程序不能写成单一的硬件驱动程序。

对窗口和连网的另一个新要求是在使用时并非要指定硬件设备不可(例如,发出系统调用 read 时)。我们可以想象有好几个设备为写操作打开,可以为窗口或网络选择适当的设备并执行通常的 write 调用。唯一的不便是要用户来决定“路由”,因为有几个设备在文件描述表里打开。但在基本 UNIX 里没有类似的 read 系统调用,我们必须求助于分时驱动的轮寻,或用户一级的报文传递,或其它设备。传统的 UNIX 设备接口是不适合于这种读操作的,对这种写操作也是不理想的。

传统的 UNIX 设备接口,一般称为系统调用接口(用户与内核交接处),对于携带窗口和网络的信息也是不适合的。有些信息是起数据表示过滤器作用的,有些是为控制目的的,描写路由或设置窗口参数。读/写接口并不能很好地处理这些,因为它没有区别控制包和数据包的界线也不能处理在线的控制信号。处理这类事情的传统接口是 ioctl 调用,它能在用户与内核间传递任意的但大小固定的控制用的结构。这会导致用户每一设备的 ioctl 调用和每一厂商的 ioctl 调用的混淆。没有接口两端对

话的整体的描述。嵌入数据中的控制信号不能用这种安排来处理,因为 read/write/ioctl 是分离的系统调用。最后,要处理协议各层要求在各层之间来回传递报文。这要求每一层至少能识别另一层发送或接收的信息。

STREAMS I/O 系统为开发和安装这种驱动程序提供一个基础,使它们有简单结构和明了的模块,能在系统运行时动态地修改。特别是统一了内核与用户间的接口,具有相当确定的对话形式。而且延续了传统用户/内核接口的 open/read/write/ioctl/close 系统调用,并另外增加了两个为处理窗口和网络,能处理控制/数据组合消息的调用 putmsg 和 getmsg。它为 ioctl 对话提供简单的规范,保持了传统,使驱动程序的开发具有灵活性和模块化。

### 4.2 介绍

STREAMS 由 Dennis Ritchie 创造,他是 UNIX 的创造者之一。在 1984 年 10 月 AT&T 贝尔实验室技术杂志上他发表了题为“一个 STREAM 输入-输出系统”的论文。在 1986 年 6 月,它成为 System V UNIX Release 3.0 的完整部分。它将继续是 System V UNIX 的部分并且是所有内核驱动程序开发的基础。它出现在 System V Release 4.0——1989 年推出的“统一的”UNIX 版。

Ritchie 把 STREAMS 接口描述为“基于灵活地结合子程序的设计,它代替传统的进程与终端或网络的死板的连接”,并运行在 AT&T Bell 实验室信息科学研究部的大约 20 台机器上。目前发行的 STREAMS 版本提供了我们描述的基本接口并有几个模块多工作的接口。

一个 stream 是一组模块,它在称为头 head 的一端提供遵守某一协议的用户/内核接口,而在另一端提供与传统设备驱动程序类似的驱动器接口。头与驱动器间的作用就如同过滤器一样转换数据。或象路由选择器那样为数据在设备的进出而在低端选择 stream: 为数据在进程的来往而在高端选择 stream。任何模块可以收发它自身的消息,包括控制请求,响应后出错指示。一个 stream 打开后,它的两端是固定的,但是模块却可以按需要动态地被推入或弹出 stream。通过一个 stream 与另一个 stream 的连接实现多工转换。

在 System V 的标准版里,现在包含 STREAMS 所需的全部代码,许多标准驱动程序代码都在使用它。特别是 poll 系统调用及有关信号处理的调用都是用 STREAMS 的“多道读出”

功能实现的。所有网络接口都以 STREAMS 为基础,特别是 Transport Level/Transport Provider Interface。它是 System V 网络传输层协议的编程接口。网络产品的厂商已经用 STREAMS 接口写了或正在写他们的驱动程序。这使他们的开发工作随着连网和窗口领域的迅速扩大,在维护一大组协议,适应标准的改变或接受新的标准方面显得非常简单。

撇开一些有关 stream 头和驱动程序的问题(所有这些问题在后面有详细的讨论),从概念上说,消息流通过一个 stream 是非常简单的。一个 stream 是一叠模块。它们分为写与读两端,写是正向(头→驱动程序)的而读是反向(驱动程序→头)的。整个结构形成一个双重的链表,使每一端可以找到另一端(图 4-1)。

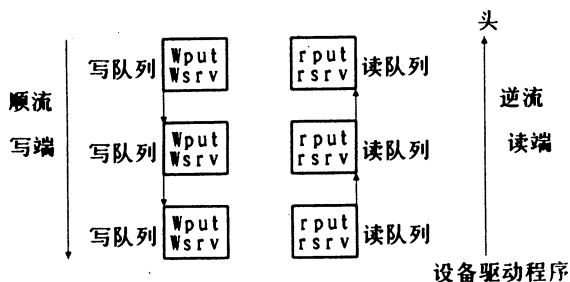


图 4-1 一个 Stream 里的模块连接

每一端可以有一个 put 过程,它执行简单的消息传递,然后或是执行另一端的 put 过程传递消息,或是把消息连接到供以后处理的队列。端与端之间可以通信。每一端可以有一个服务或 srv 过程,它从队列中取存储消息(通常是由 put 过程)并传递它们或再存储它们。srv 过程可以回答消息。一端的 put 过程被另一端执行直至完成而无需睡眠。一端放到调度表中被它本身的或另一端的 put 或 srv 过程打开(enable)。STREAMS 的调度程序是很简单的。它一打开就是准备返回给用户的系统调用,它依照调度表来运行,依次执行每一 srv 过程。这能保证每一打开的 srv 过程在系统调用返回前被执行。

#### 4.3 构造一个 stream

在详细考虑 stream 之前,我们看看普通的 UNIX 字符设备的驱动程序。这些驱动程序是单独的目标码模块(mydriv.o)。C 函数帶著变量执行 open, read, write, ioctl, 和 close 等系统调用。系统调用与这些函数的联系是通过标准字符设备开关表 cdevsw[]。它是由主设备号来索引的。系统调用通过 cdevsw[] 找到相应设备的 C 函数。这实际上正是 STREAMS 在 UNIX System V 的做法。

一个域 dstr 已被加到 cdevsw[]。此域指向一个描写

STREAMS 构造的结构。该指针的 NULL 值表明一个设备是普通的非 stream 设备,它用一般的方法来处理。这样当内核 open 遇到非零指针时,它要调用特别 STREAMS 版本的 open 及 close 等。dstr 的值指向 STREAMTAB 结构:

#### STREAMTAB

```
{ QINIT *st—rdinit: /* definitions for the read side */
  QINIT *st—wrinit: /* definitions for the write side */
  QINIT *st—muxrinit: /* definitions for the mux head read */
  QINIT *st—muxwinit: /* definitions for the mux head write */
}
```

QINIT 结构所描写的子程序是内核连接设备码的一部分:

#### QINIT

```
int (*qi—putq)(): /* Put procedure location */
int (*qi—srvp)(): /* Service procedure location */
int (*qi—qopen)(): /* Open procedure location */
int (*qi—qclose)(): /* Close procedure location */
int (*qi—qadmin)(): /* Admin procedure location */
MODINFO (*qi—minfo): /* Basic information struct */
MODSTAT (*qi—mstat): /* Basic statistics struct */
}
```

#### MODINFO

```
ushort mi—idnum:
char *mi—idname:
short mi—minpsz: /* Min packet size accepted */
short mi—maxpsz: /* Max packet size accepted */
ushort mi—hiwat: /* Hi water count */
ushort mi—lowat: /* Lo water count */
```

#### MODSTAT

```
{ long ms—pcent: /* Count of calls to put for this queue */
  long ms—scent: /* Count of calls to srv for this queue */
  long ms—ocent: /* Count of calls to open for this queue */
  long ms—ccent: /* Count of calls to close for this queue */
  long ms—acent: /* Count of calls to admin for this queue */
  long *ms—xptr: /* Location of private statistics struct */
  long ms—xsize: /* Size of private statistics struct */
}
```

STREAMTAB 结构由模块以外的变量来存取。换句话说,其它的变量可以宣布为静态的或对某些子程序来说是本地的。STREAMTAB 结构的名必须是 STRinfo。这里 STR 是存于/etc/master.d(System V, Release 3.2 或相当)stream 配置文件的名字。特别的 STREAMS open() 过程执行如下动作:

\* 为 Stream 头分配一对 QUEUE, 并填上标准的 stream 头 QINIT 结构。

\* 为设备驱动程序分配一对 QUEUE, 并填上 STREAMTAB 描述的 QINIT 结构(现在忽略多工结构)。

它连接两个 QUEUE 对如图 4-2(以后称为 rque 和 wque)。



它从适当 MODINFO 结构拷贝的水位标记到驱动程序队列。

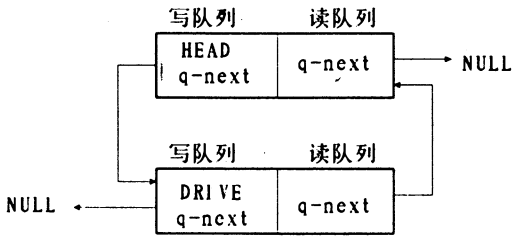


图 1-2 连接 QUEUE 对

现在已执行了一系列设置子程序。特别是，如果 rque 的 QINFO 结构指定了一个 open 子程序，它现在应被执行。它的调用方式与通常设备驱动程序的 open 子程序的调用非常似：

ropen (q-p, dev, flag, sflag)

这里 q-p 指向读队列而读队列又指向 open 子程序，dev 是一设备号，flag 是用户的 open 系统调用提供的，sflag 是类 open 标记，它将在以后讨论。

最后的结果是一些消息流向的集合，如图 4-3 所示。它把你需要知道的有关 stream 的知识符号化。

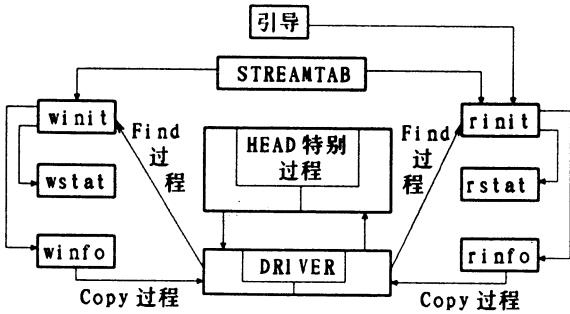


图 1-3 STREAMS 初始化流向

#### 4.4 队 列

从基本上说，stream 是一对队列的集合，具有特殊的双链结构。每一队列是如下数据结构的事件：

##### QUEUE

QINIT \*q-qinfo:  
MSGB \*q-first:  
MSGB \*q-last:  
QUEUE \*q-next:  
QUEUE \*q-link:  
caddr\_t q-ptr:  
ushort q-count:  
ushort q-flag:  
short q-minpsz:  
short q-maxpsz:  
ushort q-hiwat:  
ushort q-lowat:

QINIT 在上一节中有定义，MSGB 在下一节定义。

队列对由一读端和一写端组成。它们在内存中分配在成双的数据块里，读端在低地址。q-flag 的一个位说明一具体队列是在读端还是在写端。

read 队列在由指针 q-next 连成的单链表里，NULL 值为结尾。write 队列以同样的方式连接，但方向相反。顺着读指针向上移动和顺着写指针向下移动或在任何点改变方向都是简单的。包含第一个读队列（和最后一个写队列）的队列对叫 head 对，包含最后一个读队列（和第一个写队列）的是 driver 对（图 4-4）。

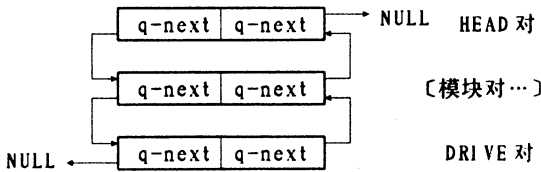


图 4-4 队列的向上流与向下流连接

现在我们考虑有关上面结构的两个重要的术语。在每一队列中（读或写），我们用 q-first 和 q-last 指针把消息连成单链表，消息可以增加至任一端。消息队列或者属于读队列 rque 或者属于写队列 wque（图 4-5）。

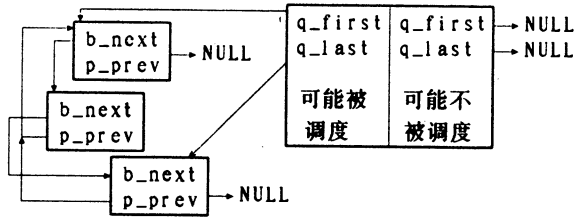


图 1-5 消息队列

最后有一个 STREAMS 调度队列，用 q-link 指针形成一单链表（图 4-6）。STREAMS 本身跟踪表头和表尾以便增加队列给 STREAMS 调度。这一队列形成先进先出队列。q-link 指针的目的是使 STREAMS 调度程序快速地移到下一调度项目。

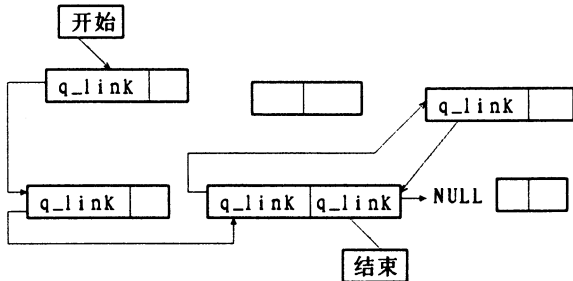


图 1-6 队列调度

UNIX/XENIX 操作系统是目前国际上最为流行的一种操作系统,有着 PC-DOS 操作系统不可比拟的优点,那就是它提供了多任务、多用户、分时的功能,而广泛应用于银行、商业各界,受到了人们的喜爱。

386、486 系统主机都配有一个标准的并行接口,可接一个并口打印机,供系统中各终端用户使用,但随着终端用户的逐渐增多,打印量将越来越大,这样使用一台系统打印机将不再使用户感到满意,从而终端打印就提到了一个重要的议事日程上来了,通常情况下,利用终端打印都是先将终端设置成透明打印方式,然后将要打印的内容定向到终端屏幕,从而终端接收到的字符直接送到打印机而不显示于屏幕,最后再将终端设置成退出透明打印方式状态,那么能不能有一个通用的终端打印命令(设为 LPRT),使得它就象系统本身带有的主机打印命令 LP 一样,可以随时随地调用呢?再进一步设想,能不能使终端打印机成为可共享的系统打印机呢?亦即系统中的每一用户只要知道该终端打印机的名字,就可以使用该打印机呢?答案是肯定的。

笔者通过长期实践摸索,对于前者用 C 语言编了一段小程序,设程序名为 LPRT. C。经过编译后,即 #CC-0 LPRT LPRT. C 便生成了一个可执行的目标文件 LPRT。将它放在系统的 /BIN 目录下,且将它的存取权限改为可执行权,所有者和组名都改为 BIN,即:

```
# chmod 777 lppt
# chown bin lppt
# chgrp bin lppt
```

这样我们就可以在任何目录下,调用命令 LPRT 在终端上打印文件了。程序如下:

```
# define ESC 27

# include <stdio.h>
main (argc, argv)
int argc;
char *argv[]:
{
    int c;
    FILE *fp; *fopen (":
    if (argc==1)
```

```
{
    printf ("没有找到文件! \n");
    exit (0);
}
else
{
    while (--argc>0)
    if ((fp=fopen (*++argv, "r"))==null)
    {
        printf ("lppt 打不开文件 %s\n", *argv);
        break;
    }
    else
    {
        printf (" %c[5i", ESC); /* 设置成透明打印方式 */
        print (fp);
        fclose (fp);
        printf (" %c[4i", ESC); /* 取消透明打印方式 */
    }
}

print (fp)
FILE *fp;
{
    int c;
    while ((c=getc (fp))!= EOF)
        putchar (c);
}
```

## 操作系统下的

## 终端打印技巧

◆ 洪系俊

那么,怎样使终端打印机成为可共享的系统打印机呢?显然后者比前者就更进一步了。第一步首先需要设置终端打印机。

# mkdev lp

屏幕上显示:

Do you wish to continue (Y/N)?

选"Y"继续。

接着屏幕上显示

1. Add a new printer
2. Remove a printer
3. Reconfigure an existing printer
4. Assign a default printer
5. Print lp status information

选"1"增加一个打印机

屏幕上显示:

1. Parallel printer
2. Serial printer
3. Remote printer

广州袖珍计算机技术服务中心与上海测绘院联合研制的 E500 机测绘电子记簿系统已应用于生产。该软件适用于地形测量、地籍测量和工程测量。E500 计算机测绘野外电子记簿软件(简称 GSY 软件)具有以下八大特点:

#### 一、仪器组配方式灵活

野外测绘仪器的组配方式可选用下面任一种

- 1、全站仪(全自动记录)
- 2、测距仪+电子经纬仪(全自动记录)
- 3、测距仪+光学经纬仪(半自动记录)
- 4、电子经纬仪+人工丈量(或无电子测量输出/口的测量设备)(半自动记录)
- 5、光学经纬仪+人工丈量(或无电子测量输出/口的测量设备)(按键记录)

#### 二、适用机型多

采用的电子测绘仪器可选用的国外机型有:索佳(SOKKIA)、宾得(PENTAX)、拓普康(TOPCON)、尼康(NIKON)等系列。国内有北京及南京生产的电子经纬仪等等。也可按客户要求增加适用机型。

#### 三、省劳力、效率高

采用全自动记录时,对准目标后,只需在 E500 机按两个键(无需在仪器按任何键),即可完成全部记录计算,日测效率可达 2000 点,观测、记录仅需一人。

#### 四、测量数据一屏显示

所记录的斜距、水平角、垂直角、视标高及算出的平距、X、Y 坐标、高程和编码全部显示在 E500 机一个屏幕上,便于查阅。

#### 五、测量方法多种多样

采用的测量方法有十多种供选择,主要方法有:

- 1、极坐标法;2、边长前、后方交会;3、方向交会;4、截距法;5、偏移测量;6、加减距测量等。

#### 六、编码方法简单易学

测点的编码只用 10 种符号及数字键,简单易学好记,90%的测点编码只用一个键。

#### 七、自动打印成果表、自动绘草图

可选用四大系列的绘图仪打印输出成果表和自动绘制各种比例尺草图,便于现场校核。适用的绘图仪有 DXY 系列、DP 系列、515P 系列、PP40 系列共十多种型号。

#### 八、既可机助成图,又可手工成图

已有机助成图设备的单位,可把所采集的数据即传送到微机,用于数字化成图,暂时没有机助成图设备的单位,也可采用小平板仪现场手工成图。而数据可保留在磁带或微机磁盘上,留待将来作为数字化地图的数据。

## 高效率的 E500 计算机 野外电子记簿软件

广州袖珍计算机技术服务中心

◎ 卢耀权

438

选"2"串口打印机

接着屏幕上显示

Texnas instruments 855 printer

询问打印机的类型,可选"2",接着屏幕显示:

Is this the default printer (y/n)?

选"N"非缺省打印机,最后再选 Q 退出,至此为止,打印机的配置工作就做完了,下面要做的,也就是最重要的工作,那就是修改/usr/spool/lp/interface 目录下的接口打印文件 zddy. 将它改为:

title=Y3

copies=Y4

options=Y5

shift:shift:shift:shift:shift

# If it is necessary to change the baud rate or other stty settings for

# your serial printer add the appropriate options here:

stty ixon ixoff 0<&.1

echo "\033[2h"

echo "\033[5i"

# send the file(s) to the standard out Ycopies times

while ["Ycopies" -gt 0]

do

for file

do

cat "Yfile" 2<1

echo "\n\c"

done

copies='expr Ycopies -1'

done

echo "\033[4i"

echo "\033[2l"

exit 0

至此,所有工作都已做完,系统中的所有用户均可使用该打印机了,使用方法如下:lp -dzddy 文件名

以上所有,均在 NDEC38 微机、长城 GW220 中西文终端、LQ-100K 打印机、XENIX system V 2.3.2 操作系统下通过实现。

437

FOXBASE+ / Mac 是 Fox 软件公司为 Apple 公司的 Macintosh 高级微机提供的数据库管理软件。在适应于 Mac 微机上运行的众多数据库管理软件中, 由于 FOXBASE+ / Mac 的速度最

快、性能最好、操作简便,

自 88 年推出以来

一直受到 Mac

用户的交口

称誉。

FOXBASE+ /

Mac 囊括了

标准 DBASE

的所有特征, 并

为适应 Mac 操作环境

作了许多的改进的扩充, 对许多

命令和函数提供访问 Mac 工具箱

(TOOLBOX) 的支持, 因而具有许多独特的风格。

### 1、具有优秀的支撑环境。

FOXBASE+ / Mac 运行于 Macintosh 高级微机操作平台。Mac 微机基于摩托罗拉的 M680X0 CPU。其 OS 结构与 PC DOS OS 迥然不同。Mac 由于其漂亮的图形用户界面、标准的程序接口、先进的多媒质技术、丰富的应用软件群等一直在微型机领域独领风骚。运行在 Mac 操作环境中的 FOXBASE+ 数据库软件同样沿用和继承了 Mac OS 的特点。它也包含了当前最先进的图形用户界面, 支持工程和过程语言, 为 Mac 机上的高性能数据库管理建立了标准。

### 2、全新的图形用户界面。

数据库用户可以不需编写任何程序或输入程序命令而直接利用高性能的图形“控制板”和易于使用的滚动菜单等用户界面建立、打开、检验、修改、增删和管理数据库, 以图形交互式建立索引、数据关系、查询方式。滚动式菜单可以通过连按(Click)和拖曳(Drag)动作鼠标器操作。

### 3、在线帮助(Online—help) 机制。

FOXBASE+ / Mac 为用户提供了一个通用的在线帮助(Online—help) 机制。FOXBASE+ 的在线帮助机制包括两个方面, 一是它为用户提供了一个联机帮助手册;

二是系统开发者可以非常容易地为应用系统装备类似的在线帮助机制。

Online—help 机制之所以易于使用和最受欢迎, 原因在于系统将所有

的 HELP 正文均存储

在某一个数据库

文件中。如

系统本身

的联机帮

助手册

册, 将各

种命令和

函数的帮助

信函都存储在

HELP. DBF 数据库

中, 系统提供在线帮助时都是

从该库中提取帮助正文用特定方式显示。

在应用系统向用户提供本应用系统的在线帮助, 也是将应用系统的所有帮助正文存储在某一数据库中。该库只需满足一个可以接受的条件: 该库至少包含两个字段: 第一个字段一定是字符(CHAR)字段(内容是将 Topics 面板上显示的帮助主题名); 第二个字段一定是备注(MEMO)字段(存储实际的帮助正文)。然后在系统中指定欲使用的帮助数据库, 并设定帮助热键激活时相应在线帮助的激活方式, 就可以在应用程序的任何地方提供实时方便的在线帮助了。该帮助库可以动态修改, 也可以增加任意多个字段, 帮助主题可以用任何形式的表达式来选择, 所以在应用系统中几乎不会受到什么限制。在帮助数据库中可以提供不同数量的帮助信息; 也可以为使用不同语言的地方提供不同的帮助正文, 这些正文都出自同一个数据库。

### 4、性能强大的命令(Command) 窗口。

FOXBASE+ / Mac 系统提供了一个 Command 窗口, 该 Command 窗口内涵远比 DOS 的 FOXBASE+ 版本要丰富得多。

①通过该窗口能编辑并直接执行任何合法命令。其与 DOS 版本一样亦提供一个历史命令表, 本次启动 FOXBASE+ 后键入的命令在 Command 窗口中保留下

# FoxBase+ / Mac

## 的独特风格

罗 辉

来,通过鼠标或光标键可以选择历史命令重新执行。

②通过图形用户界面提供一条命令时,实际上即在 Command 窗口产生了一条命令,该命令也将保留在历史命令表内,并可以通过命令窗口被编辑和执行。

③命令窗口保留的命令可以从窗口中删去,也可以直接编辑后传递给用户程序,从而允许你充分利用 FOXBASE+ / Mac 功能的全部优点使你的文件及数据处理任务自动化。

④Command 窗口的历史命令的全屏幕操作方式,较之 DOS 的 FOXBASE+ 版本的行操作方式,使用更简单,操作更直观方便。

### 5、“见即所得”技术。

FOXBASE+ / Mac 支持用户屏幕输出的多种字型 (FONTS)、多种字体 (SIZES)、多种打印风格 (STYLES), 用户可以随意进行三者的组合选择,而且用户选择了任一显示模式后,该用户屏幕输出立即体现出你所选择的字型、字体和风格,真正实现了“所见即所得”(What you see is what you obtain)技术。

### 6、多窗口、多色彩。

FOXBASE+ / Mac 是一个多用户系统,它支持 9 个用户输出屏幕,这些屏幕就象报表中的一页一样。通过 Soreen 命令可显示并激活 9 个屏幕之一,程序所产生的所有输出在當前所选择的输出屏幕上显示。同时激活并同时在多个屏幕进行程序或文本的编辑 (EDIT)、帮助 (HELP)、跟踪 (Trace)、调试 (Debug)、状态 (STATUS) 和便函以及图形编辑。各种窗口可以被移动、改变大小、放大至全屏幕和缩小为一个肖像 (ICON)。

系统的颜色支持也非 DOS 版本所能比拟。9 个用户屏幕都可以具有各自省缺的前景和背景颜色,而且在任意屏幕上进行正文或图象输出时都可以重新设定色彩,色彩可以按一定比例的灰度进行变换,各种颜色可以在屏幕、打印机等输出设备上直接输出,只要这些设备支持多种色彩输出功能。

### 7、功能强化的 BROWSE 功能。

FOXBASE+ / Mac 的 BROWSE 功能包含标准 BROWSE 的功能的同时,还有它许多独特之处。BROWSE 可以编辑记录和图像字段,被拖入 BROWSE 的数据库中的若干字段构成的子集通过 PUSHER 指针可

以任何顺序被选择显示和操作,字段的长度或顺序可以随时进行动态调整。BROWSE 窗口也能被水平地分为两个独立部分处理某些包含许多字段的大型数据库记录。

### 8、支持图形数据字型。

存储在数据库中的图形可以是单色或彩色的,利用通常的剪 (CUT) 贴 (PASTE) 操作将图形放置到数据库中。图形显示时,可以被剪辑或按比例变换成任意尺寸的矩形帧,在输入格式中甚至可以将数据库图形当做按钮使用。

### 9、屏幕 I/O 和调试工具丰富。

FOXBASE+ / Mac 有十分丰富的工具供用户的自定义的输入格式,如发射按钮 (Radio buttons)、检验盒 (Check boxes)、弹出式菜单 (Pop-up menu)、垂直和水平按钮数组、正文编辑、菜单拉杆 (Menu bar) 等输入/出工具,使应用程序的编制简单而功能和十分强大。它还是提供了跟踪 (Trace) 和调试 (Debug) 窗口,在源程序级进行程序的调试,使我们的程序调试过程变得更加简便和舒适。通过跟踪 (Trace) 窗口,我们可以一行行地监视 (monitor) 程序的执行,相关的若干程序行语句显示在该窗口中,便于用户能结合上下文检查程序;用 Trace 窗口的各种标准的 Slider 滑动机制,使用户能随意观察程序各模块的执行情况;用户可以建立或删除任意多个断点,以便于监视,……。通过 Debug 窗口可以监视程序执行时任意表达式的取值。在 Debug 窗口中当指定的表达式取值真时,可通知 FOXBASE+ 挂起执行。

### 10、兼容性好。

FOXBASE+ / Mac 完全兼容工业标准 DBASE 语言, DOS 系统下编制的 DBASE 程序,可以不做任何修改便直接在 Mac 下运行,只需做的是将 DOS 下相应文件拷贝到 Mac 机上即可,而 DOS→Mac 的文件和数据转换可通过多种方式方便地实现。这种兼容性包括:程序、数据文件、报告定义标号定义文件、内存变量文件及其它方面。

FOXBASE+ / Mac 的优点和强大功能,远远不止这些。它提供了多方面有价值的语言扩展。它良好的开发环境,优秀的设计风格,对我们进行其它数据系统设计,亦或应用软件设计,都具有很好的借鉴和参考意义。

# APL 符号语言的发展过程

江苏省科技情报所编译室 王 遐

在《基础 APL》中译本即将出版之际,该书作者,日本西川利男博士特意写信致中国读者,简明扼要地介绍了 APL 语言的特点和发展过程。现将原文译出,以便更多的读者了解 APL 语言。

## 致中国读者

APL(A Programming Language)是加拿大数学家 K. E. Iverson 创立的一种极为独特的程序设计语言,其特点大体可以分为术语和功能两个方面。

首先讲一下术语。BASIC 和 FORTRAN 等程序设计语言,使用了 read, write, for--next, if--then--else 等英语关键字,而 APL 与之不同,使用独特的 APL 文字进行处理。Iverson 是数学家,因而发明了由 1, 2, 3……这些数字或 +, -, = 之类的记号延伸而来的用于计算机处理的独特文字,并使用这些文字进行程序设计。可以说,这些文字是以功能为基础想象出来的计算机世界的象形文字。

一谈起文字,共同使用相同汉字的日本人和中国人格外概,不言而喻,汉字是从古至今一直有使用着的象形文字。对于使用字母 A, B, C……的欧美国家的人来说,有时似乎觉得汉字很神秘,但有时他们自己也不知不觉地搬用了这种强有力的工具。

APL 文字确实是由汉字发展而来的一种计算机文字。例如,表示把数据按顺序排列起来这个意思,APL 的记号是 ↑ 和 ↓,这和汉字的“上”和“下”的想法如出一辙。这不正是人类共同的,谁都能明白的表记方法吗?在 Iverson 的诞生地——加拿大,不仅讲英语的人多,讲法语的人也很多,人们常说:与其利用英语 read, write 等,不如不受英语的束缚为好。这是迈出国门,向人类共同的交际媒介迈进的第一步。我也有同感,想来中国的各位读者也有相同看法吧。

其次讲一下功能。APL 比至今流行的程序设计语言, FORTRAN, COBOL, BASIC, 甚至现在的 C, PASCAL 等语言都方便,而且功能更强。首先是方便,APL 对于用户来说是一种简单易用的语言,这是因为 APL 不是编译型语言,而是翻译型语言。若是用 FORTRAN 和 C 语言,写出源程序后,必须经过编译、连接才能运行。而 APL 和翻译型语言 BASIC 同样,一旦输入进行处理的程序就能立即执行,并且 APL 不必对使用的变量、类型等进行说明。尤其是,与 BASIC 比较,它处理数组的能力特别强,并且还备有许多功能极强的函数。APL 十分适合于科学技术计算、统计计算、数据库等方面。

另一方面,APL 虽然曾经是只能在大型计算机上运行的语言,但由于最近计算机硬件的迅速发展,今天,它在小型计算机、个人计算机上都能随心所欲地运行了。APL 现在已成为一种可以不管计算机硬件的大小,在科学技术、商业等所有领域里都非常活跃的通用程序设计语言。

国际 APL 联合会聚集了全世界的 APL 用户,每年召开一次学术交流会。今年(1993 年),会议将在 IVERSON 的故乡加拿大的多伦多召开,世界各国的研究成果将在会上发表。现在,很多国家都有 APL 的学会和研究会。在日本,日本 APL 协会也很活跃,经常举办研讨会、发行学术刊物等。

众所周知,计算机开始是在科学技术领域里发展起来的,现在已成为各个领域现代化不可缺少的重要工具。程序设计是必要的基础知识,就象谁都必须掌握数学一样重要。希望中国的各位读者务必了解并使用 APL 这一方便的、功能极强的、独特的程序设计语言。

5 年前,叶素与我在日本筑波工业技术院研究所中相识,得以学习 APL。此后,将拙著译成中文,克服了许多困难,现在终于进入出版阶段了。拙著若能有助于日本和中国的科学技术合作,我则不胜欣喜。

正好在一年前,我访问了南京中山陵、明孝陵、灵谷寺等名胜,紫金山郁郁葱葱的自然景色至今难以忘怀。

西川利男

1993 年 5 月

440

# 电子收款机基本结构

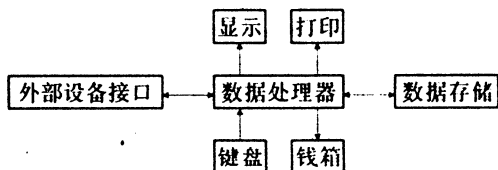
京粤电脑中心总工程师 管强

## 前言

电子收款机 (ECR) 与销售点终端 (POS) 的主要作用是处理商业销售数据, 而对数据处理能力的不同, 决定了 ECR 或 POS 的分类和档次。

由于 ECR 或 POS 是处理商业销售数据的专用设备, 从这个意义上讲, 无论什么档次的 ECR 或 POS, 都有大致相同的基本结构, 本文将从硬件 和软件两方面介绍电子收款机的基本结构。

收款机硬件大致可分为显示、打印、键盘、数据处理、数据存储、外部设备接口、钱箱 7 个基本部分, 结构如下:



这 7 个基本部分的作用与特点是:

### 1. 键盘

收款机键盘通常由数字键和功能键组成。收款机功能通过键盘的功能键 数反映, 所以, 不同档次, 不同厂家的收款机, 键盘也不相同。

收款机键盘使用频繁, 击键次数多, 可靠性要求很高, 目前大致采用两 种类型的键盘:

- a. 琴键式键盘, 每个键独立, 键连通与断开通过机械接触的方法。
- b. 薄膜式键盘, 键盘表面由一张薄膜覆盖, 各个键藏在薄膜下, 防水、防尘。键的连通与断开通过机械接触的方法。

有的收款机 (通常为 POS 系统) 可以同时接两个键盘, 一个是本身的专用键盘, 一个是通用微机键盘。

### 2. 显示

收款机的显示由两部分组成, 一部分为主显示器, 一部分为付显示器。显示器通常采用 LED 或是荧光数码管, 有些 POS 采用 CRT 显示屏。

主显示器用于指示收款员操作的显示操作结果。付显示器用于为顾客显示操作结果, 通常显示商品的数量和金额。

### 3. 打印

收款机配有一个微型打印机, 为顾客打印销售小条或作为销售记录的存根。

微型打印机有两种型式, 一种是滚轮打印机, 打印字符铸在钢轮上; 一种是点阵打印机, 打印字符由打印针头组合而成。

微型打印机还分为双头和单头两种。双头打印机同时打印两条纸条, 一条给顾客, 一条留存存根。单头打印机也可以打印存根, 通常用无碳复写纸的第二联。

收款机用微型打印机要求使用寿命长, 打印速度快 (大于每秒 2.2 行字符)。目前国产收款机除了可以打印英文、数字, 也可以打印中文。

### 4. 数据处理

收款机的数据处理部分作用是处理、计算各种销售数据, 控制收款机各种部件与设备, 通常称为 CPU。收款机数据处理能力的大小, 与 CPU 性能和相应的存储单元大小关系密切。

现代收款机的数据处理部分分为专用和通用两部分。

专用是指根据某一类收款机特定的功能而专门设计的数据处理部分, 具有结构紧凑、可靠、生产成本较低的特点。但由于是专门的使用环境而设计, 适应面窄, 不易扩充和升级。

通用是采用标准数据处理部分 (常指 PC 类微机) 设计而成的收款机。具有系统扩充容易, 便于连接各种外部设备如网络系统, 适应面宽, 但成本较高。

从现在发展情况看, 采用通用设计的收款机 (特别是 POS) 将成为趋势。这是因为用户对收款机的功能要求越来越高, 希望收款机不仅能完成计算、记录 数据的功能, 还希望在商业信息管理中作为数据计算、查询、统计、分析的 销售工作站。这样对收款机在数据处理速度, 外部设备连接能力, 联网能力及系 统扩充能力上都有着较高的要求。而专用设计的收款机想要很好的适应这不断 变化的用户需求, 显然是很困难的。

### 5. 数据存储

数据存储部分用于存储收款机所产生的各类数据, 如销售、汇总、审计、税 收数据等。收款机根据类型不同, 所采用的数据存储介质类型也不尽相同。大 致有如下种类:

- \* 只读存储器 (ROM), 通常用于存储程序。
- \* 电可擦除只读存储器 (EEPROM), 用于存储销售、汇总、审计、税收等长期存放、不能随意更改的数据, 有些系统控制参数也放在里面。
- \* 静态随机存储器 (SRAM), 用于存储销售明细数据或是暂时存放的系统控制参数。

\* 磁盘 (DISK) 可以同时存放上述三种存储器存放的数据。比较起来, 前三种存储器属于电子器件存储器, 具有数据存取速度快、可靠的特点, 缺点是容量太小, 成本高。

### 6. 外部设备接口

外部设备接口包括接口本身及接口驱动部件, 用于收款机连接各种外部设备, 如条形码、打印机、读卡机等。有如下类型:

- \* RS232 串行异步通讯口, 常用于连接条形码、信用卡阅读器、串行票据打印机等设备。亦可用于收款机之间的数据通信或是与主计算机相连, 将收款机作为主计算机的仿真终端。
- \* Centrinice 并行口, 常用于连接外接打印设备。
- \* 网络通信接口, 用于收款机本身联网。目前较为流行的网络接口有 RS485、RS422、以太网 (Ethernet)。最近又发展到光纤通信网络

## ZJ 3712 打印机一种非损坏性故障的维修

江苏省洪泽县工商银行 唐兆海

现象 1: 打印机在联机状态下, 开机插纸后进纸不停。

现象 2: 打印机联机时, 即使还没有装打印纸, 进纸轴就滚动不停。

现象 3: 打印机在开机后联机状态下, 插入存折进纸轴不动或者微动后就停止。

故障分析: ZJ 3712 是一种票据、存折打印机, 在银行系统内广泛使用。它的进纸退纸控制由三个按钮和两个光电三极管组成, 从打印机正面起, 第一个光电管称之为 A 管, 第二为 B 管, 其工作逻辑原理描述如图:

(图见附件图)

从上图并结合故障现象可见, 进纸电机的控制电路出现故障; 然而联机状态下, 进纸到位是由两个光电管的状态控制的, 所以, 应首先检查之, 再查别的。如下是故障排除的过程。

故障排除:

对现象 1, 打开面板用逻辑笔或万用表检查发现 B 管一直处

于导通态, 进纸时此管没有被关断。细查发现该管的固定螺丝松动偏离原位, 进纸无法关闭它, 调整后故障消除。

对现象 2, 也发现现象 1 的情况, 但是调整 B 管后, 重新开机进纸轴仍转动不止, 送纸后可以到位。再查发现 A 管被灰尘遮盖, 光电管不能工作, 用酒精擦净装好, 开机正常。

对现象 3, 均发现 B 管不能工作, A 管有的可以、有的不能工作, 且都呈关闭状态, 查为灰尘或异物所覆盖, 擦净按好后, 故障排除。

建议: 由于银行的机器多数在公共场合下使用, 机器常被尘埃异物等污染, 因此在这种环境中的机具应经常清理, 以免经常发生故障。

附件图:

脱机态					
进纸键	退纸键	A 管	B 管	动作	结果
1	0	0	0	强制进纸	有效
0	1	0	0	强制退纸	有效
联机态					
进纸键	退纸键	A 管	B 管	动作	结果
0	0	1	1	无纸开机	等待进纸
0	0	0	1	插入打印纸	进纸有效
0	0	0	0	滚轴走纸	进纸到位
0	0	1	0	发光报警	轴停纸偏

442

(FDDI)。

\* 外接键盘接口, 常用于外接标准键盘或是具有键盘接口的条形码阅读器; 通常只有通用设计的 POS 系统有此类接口。

### 7. 钱箱

钱箱用于收款员放钱, 分金属钱箱和塑料钱箱, 与收款机之间有电路连接, 正常情况的开启由收款机操作控制。

电子收款机软件基本结构:

收款机软件是根据硬件基本结构设计的, 主要作用是数据处理和设备调度与管理。基本结构可分为 4 个部分:

#### 1. 数据采集

主要功能是接收、控制数据采集设备, 如键盘、条形码、信用卡阅读器等。将这些设备输入的数据接收、控制、转换为数据处理部分可辨别的格式数据。

#### 2. 数据处理

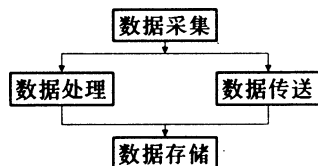
对各部分送来的数据作计算、变换、汇总、分析等数据处理, 将结果再送回至各有关部分。

#### 3. 数据存储

控制数据存储设备, 将数据处理部分处理过的数据按一定数据格式存储, 或将已存储的数据取出供数据处理部分进一步处理。

### 4. 数据传送

将数据处理部分处理后的数据传送到显示打印设备, 或是将这些数据通过收款机网络传送到其它地方, 也可以接收设备传来的数据, 由数据处理部分处理。



以上简要的介绍了一下电子收款机的基本结构, 可以认为, 现代的收款机就是一部专用电脑。本文限于篇幅, 不可能详细论述, 尤其是软件部分, 实际上要复杂得多。

随着我国商业领域的计算机信息管理和零售业在税收管理上的需要, 电子收款机在今后几年将大量使用, 也将开始被人们所接收和熟悉。本文仅作为电子收款机的入门介绍, 供读者参考。

441



# 谈 谈 VGA 彩 色 显 示 器

都帮机电有限公司 淡玉婷

在当今的电脑市场上,不论是原装进口微型计算机,还是为广大国内用户所接受的兼容机,在其硬件系统配置中,没有一个能象显示器那样能具有电脑与人间的亲密关系。从黑白的 MGA 到现下广为流行的 Super VGA,所有厂商的无不在技术上寻求突破,使显示器尽到其逼真及高分辨率的义务。为适合不同客户所需,从占市场销售量的近八成的 14 寸始,到 21 寸,传达信息的重任几乎全部落在了 VGA 的“肩上”。因此,不论是换购彩色显示器还是第一次添购电脑,或是其它欲使用 VGA 彩色显示器的用户,完全有必要对 VGA 彩色显示器进行一定的了解,以便较好地选购和使用。

## 一、显示器的工作原理

所有显示的画面的显示都是由连续的光点所产生,由显像管辐射出的电子以极快的速度,打在屏幕上,利用人的视觉暂留现象加上射出电子的速度极快,使显示器显示出清楚的画面。单色显示器仅一个电子枪,而彩色则有三个控制三原色(红、蓝、绿)的电子枪,利用此三原色的重叠,产生人眼所见的彩色画面。

电子束射出后,须通过一道充满着细圆孔的壁面,只有少数电子束通过孔径,再打到发光体上。

对解析度(通常称为分辨率)的衡量,通常都是以水平点数乘上垂直点数。如标准 VGA 的  $640 \times 480$ 。而分辨率与显示器运作颇重要的水平及垂直扫描频率间的关系是:在标准 VGA 模式下,水平频率 31.5KHz,垂直频率则为 60Hz,电子枪以 31500 次/秒的速度对水平 640 个点进行着色,直到 480 行完成,再重新回到起始位置。而垂直频率即指每秒扫描完成的画面次数。

事实上,扫描又有交错式与非交错式(即隔行扫描与逐行扫描)之分:所谓交错式,是指电子枪先对所有单数行进行扫描,再对双数行进行扫描,如此须扫描两次,方可完成一个画面,此乃隔行扫描。而非交错式扫描则指从头到尾一次扫描完毕亦即逐行扫描。从显示原理来讲,要做到逐行扫描,水平工作频率就必须较高,在  $1024 \times 768$  的分辨率下,至少须在 40KHz 以上,但也须看搭配的显示卡而定。而就对眼睛的保护而言,非交错扫描是较好的选择。

另一对眼睛有较大伤害的是显示器的幅射线,这也是彩色显示器在广为流行之后,较为人们所关心的一个话题。一台显示器中,其前后左右都有或多或少的幅射线。而

世界各国也均制定了幅射线的安全标准。使用者在货比三家时,不妨了解和观察一下。现在的国内流行一些 VGA 彩显中就有在低辐射高品质,电器性能方面表现优异的彩色显示器,如由世界第八大彩显生产厂家生产的 RED 即 AOC 彩显就为其中之一。

## 二、显示器与显示卡

有了显示器,就必须配备一个显示卡,否则显示器就不能进行正常的彩色显示。在显示器与显示卡的配合中,画面若有不良情况,也不一定全是显示器的问题,以颜色为例。彩色显示器的三原色电子枪按理可搭配显示出无数种颜色,但受限于显示卡,而就有了 256 色与 16 色之差别。

## 三、彩色显示器的发展与市场

作为电脑的灵魂之窗,彩色显示器进入 1994 年,14 寸彩显仍将是市场的主流,也是用户的最基本的要求,显示器制造则朝可接受多种频率及通行扫描发展,且已蔚为潮流,同时分辨率向四位数迈进。另外,由于一般用户均可接受的价格水平,使 14 寸彩色显示器在用户眼中已非高利润产品。

在产品方面,如上所述逐行扫描及多种频率显示,已属市场平常,但在平面直角的 14 寸产品方面,则仍不多见。显示器的组件器,以显像管(CRT)所占生产成本最高,这仍将是日货的天下。与显像管相同,对画面品质有决定影响的是肉眼看不到的点距。所谓点距是描显像管上两个荧光点间的距离,点距是显像管执行扫描任务时,必定会产生结果。点距愈小,显示画面就愈锐丽,清楚,一般来说,愈大的显像管才能接受较大的点距,而 14 寸屏幕的点距应在 0.39mm(或 0.31mm)以内,画面品质以 0.28mm 的点距为目前 14 寸产品的市场标准。

另一个与点距有关的是收敛度,而这也是厂商在介绍中较少提及的,其指的是组合成一个白点的三原色的点之密合程度。收敛度不佳,会造成文字边缘的汇聚不佳现象,通常在传统圆弧造型的产品中,这种现象在屏幕边缘较严重。

因此,在产品选购中,收敛度亦应是一个值得关注的指标。总之,只有对 VGA 彩色显示有了一个较全面的了解和认识,才能真正地,较好地选购和使用 VGA 彩色显示器,真正做到物尽其用。

本文提供的 LQ—1600K 打印针测试程序可以随时检测 LQ—1600K 打印针的状态。运行此程序,24 根针就一一对应地在打印纸上印出一根细线。若出现磨损或断针

现象,则与打印针相对应的细线就会出现漏打或空白,从而可以判定是哪一根针出现了故障,引导维修人员迅速予以修复。

### 一、打印机打印原理

LQ—1600K 打印机的打印头上分布有 24 个打点针。当打印头移过纸面时,一部分打点针被电流脉冲激发而撞击色带。当一个打点针撞在色带上时,在该撞击点处色带与打印纸相碰,一个小点就会被印在打印纸上。这样,随着打印头掠过纸面,打点针以不同的组合方式无数次撞击色带,不同的文字、数字和符号即被打印出来。

为设定打印机在每列中哪个会击发,首先必须将垂直列划成三个各有 8 只针的部分,并分别考虑各部分。由于对各部分来说 8 只针都有 256 种可能的组合,所以这就需要有一个编码方法,可以用来在 256 种可能图案中确定所希望的组合。这种编码方法如下所示:

128 ○  
64 ○  
32 ○  
16 ○  
8 ○  
4 ○  
2 ○  
1 ○

要击发某针,可将其数字送入。为了在同一时刻击发一只以上的针,只需将各针的数码相加并把和送入打印机。用这套编码,如送入 128 则击发顶针,而送入 1 则击发底针。如果您想只击发顶针及底针,只需简单地将 128 加 1,然后送入 129。

128 ●	128 ○	128 ●
64 ○	64 ●	64 ○
32 ○	32 ○	32 ○
16 ○	16 ●	16 ○
8 ●	8 ●	8 ○
4 ○	4 ○	4 ●
2 ●	2 ●	2 ●
1 ○	1 ○	1 ○
170	74	134

通过把相应的编码数字相加,就可以击发打印针的任

# LQ—1600K 打印针测试程序

刘云枫

何组合。上图列出了三个例子来说明如何进行计算以击发特定的打印机图案。

使用这种编码方式,8 只针的任何组合加起来都是一个位于 0 至 255 之间的十进制数,并且没有重复。

### 二、程序实现

要实现 LQ—1600K 打印针的测试,首先须将打印机设置成图形方式。LQ—1600K 提供了一条指令允许用户使用 11 种图形选择中的任意一种。在打印机接收了这条指令后,它将下一个数字翻译成打印针模式,并将其打印下来。需要注意的是,在打印机设置成图形方式之后,应给打印机提供足够字节的图形数据,否则打印机将停止运行并等待更多的数据,看上去象是被锁住了。另一方面,如果用户提供太多的数据,多余的部分将被作为普通文件翻译和打印。

源程序清单和 LQ—1600K 打印机的 24 根针的测试结果如下。本程序在 AST—286 机和 GWBASIC 的软件支持下通过。

```
10 'LQ—1600K 打印针测试程序
100 WIDTH "lpt1:", 255
105 '测试第一~第八针
110 FOR I=1 TO 8
120 LPRINT CHR (27) " *" CHR (32) CHR (240) CHR (0)
130 FOR X=1 TO 240
140 LPRINT CHR (256 / (2 ^ I)) : CHR (0) : CHR (0) :
150 NEXT X
160 LPRINT "第":I:"针":
170 LPRINT
180 NEXT I
185 '测试第九~第十六针
190 FOR I=9 TO 16
200 LPRINT CHR (27) " *" CHR (32) CHR (240) CHR (0)
210 FOR X=1 TO 239
220 LPRINT CHR (0) : CHR (128 / (2 ^ (I-9))) : CHR (0) :
230 NEXT X
240 LPRINT "第":I:"针":
250 LPRINT
260 NEXT I
265 '测试第十七~第二十四针
270 FOR I=17 TO 24
280 LPRINT CHR (27) " *" CHR (32) CHR (240) CHR (0)
290 FOR X=1 TO 239
300 LPRINT CHR (0) : CHR (0) : CHR (128 / (2 ^ (I-17))) :
310 NEXT X
320 LPRINT "第":I:"针":
330 LPRINT
340 NEXT I
350 END
```

西文 WINDOWS3.1V 是 Microsoft 软件公司于 1992 年 3 月推出的新一代的操作系统, 它可加载多任务程序, 并能充分利用微机所有的内存空间, 并可管理 16MB 以内的扩展内存空间。SPDOS5.1V 和 2.13H 是国内比较流行的汉字操作系统, SPDOS5.1V 提供了双拼双音的输入法, 用户可以较快地输入汉字, 而 2.13H 具有较方便的打印控制码, 因此, 如何将西文 WINDOWS3.1V 与 SPDOS 5.1V、2.13H 汉字系统联合起来使用, 是电脑用户十分关心的问题。笔者经过实践, 实现了西文 WINDOWS3.1V 可加载应用软件, 同时, 应用软件在 SPDOS5.1V 汉字系统上运行, 但又能用 2.13H 的打印控制码进行打印应用程序的运行结果。下面以笔者用 FOXBASE+ 开发的应用软件 GYS (主文件为 GYS.EXE) 为例具体介绍其实现步骤。

1、在 WINDOWS 3.1V 的 MAIN 组中, 用 PIF EDITOR 图符启动 PIF 文件编辑器, 建立一个启动应用软件 GYS 的 WINDOWS 运行环境 GSH.PIF, 下面就是 PIF 的基本配置项:

```
Program filename d:\gys\gsh.BAT
windows title:
option parameter :
satrtup directory: d:\gys
video memory:High Graphics
Memoey Requirement:KB Required:-1
KB Desired:-1
EMS Memory:KB Required:512
KB Limit: 1024
XMS Memory:KB Required:0
KB Limit:1024
Display usage:Full screen
Exection:Background
Close Windows On EXit:ON
Advanced 选择项可采取系统默认真
```

2、编制应用软件 GYS 的启动批处理文件 GSH.BAT, 其内容如下:

```
ECHO OFF
CLS
```

```
PATH %PATH%: D:\GSY\C:\213\C:\WPS21:
(3)APPEND C:\WPS21:
SPDOS CIZU/VGA>NUL
(5)CALL PRI213
GYS
```

其中 (3) 为 SPDOS 查找 CIZU 词组文件所设置的路径;

(5) 为调用 2.13H 的专用打印驱动批处理文件。

### 3、调入 SPDOS 5.1V 的方法

调入 Super CCDOS 5.1V 汉字系统, 应两个文件, 它们是 SPLIB.EXE (建立 XSDOS.LPH 字库的索引表) 和 SPDOS.COM 基本汉字模块。经笔者试验, SPLIB.EXE 模块不能在 WINDOWS 3.1V 中调用, 即在 WINDOWS 中重复调用 SPLIB, 将使 SPDOS.COM 模块不能正常调入内存, 严重时系统将崩溃。为了解决这个问题, 笔者采取一个

折中的方法, 即在 MS DOS 系统的 AUTOEXEC.BAT 中或在进入 WINDOWS 之前先调入 SPLIB 模块。

### 4、2.13H 的

专用打印驱动批处理文件 PRT213.BAT

## 联合起来使用的技巧

重庆医药设计院工程经济室 曹国钧

的编制方法 2.13H 汉字系统占用了 MS DOS 系统的中断号 78-7FH。其中 7A-7CH 分别用来读取 16×16, 24×24, 40×40 点阵字库, 对应于 2.13H 的文件模块为 FILE16B.COM, FILE24A.COM 和 FILE40A.COM; 另外, 2.13H 也替换了打印管理中断 INT 17H。其文件模块为 PRT.COM。2.13H 系统在打印汉字时, 先调用 INT 17H。然后, 根据汉字的字型分别调用 INT 7A, INT 7B 和 INT 7C。最后, 在打印机上打印出 2.13H 的汉字。

因此, 根据上面的分析, 2.13H 的打印系统完全可以独立于 2.13H 汉字系统, 不过, 其它一些系统 (包括目前的汉字系统, 如 SPDOS, UCOS 等) 或应用软件不能占用 7A-7CH 中断, 且 INT 17H 是由 2.13H 的 PRT.COM 模块设置的。在金山汉字系统 SPDOS 中, 当调入 SPDOS 后, 就可调入 2.13H 的专用打印系统, 但 SPDOS 的打印程序 PRT16.COM, PRT24.COM 和 PRT40.COM 就不能再调入, 否则, 它们将替换了 2.13H 打中断的 INT

17H: 另外, 在 Super WPS 系统中完全可用 PRT16.COM, PRT24.COM 及 PRT40.COM 打印系统, 在退出 WPS 系统后, 又恢复了 INT 17H 中断, 这样, 就不会影响 2.13H 的打印系统。2.13H 的专用打印系统 PRT213 的批处理如下

```
echo SPDOS 系统下 2.13H 专用打印系统
ECHO+
ECHO 重庆医药设计院 曹国钧 1993.2 月
ECHO+
C:
cd \ 213
file16b
prt
file24a 1SFHK
FILE40A 1SFHK
zf24 3
```

从 PRT213.BAT 中可以看到, 2.13H 专用打印系统只需下列文件:

```
FILE16B.COM PRT.COM
FILE24A.COM FILE40A.COM
ZF24.COM HZK16
HZK24S(H. K. F. T) HZK40S(H. F. H. K)
```

5、在 WINDOWS 3.1V 中建立应用软件 GYS 的图符在 windows 的 file 菜单中选择 NEW 子菜单, 出现提示后, 在 Program Item 中配置如下:

```
Description: GYS 1.0V
Command Line: e:\cgj\wins\gsh.pif
Working Directory:
Shortcut Key: None
```

然后用 Chang Icon 选择自己认为较好的图符, 用 OK 就将刚才的选择存到 WINDOWS 的 MAIN 组, 用户只需用关标或鼠标器选择该图符, 用 FILE 菜单的 OPEN 子菜单的就可将应用软件 GYS 调入内存中。

经笔者试验, 在 WINDOWS 3.1V 运行 GYS 软件要比在 MS DOS 环境中运行速度快, 出现的错误少, 而且能使 GYS 在 EMS 扩充内存空间中运行; 用户也可重新配 GSH.PIF 使 GYS 也能在 XMS 扩展空间中运行。由于 GYS 是用 FOXBASE+2.10V 编制的, 而 FOXBASE+ 的特点就是微机的可用内存空间越大, 则运行于它上的速度将是越快, 因此, 笔者为 GYS 配置了 512KB 的 EMS 内存。

以上方法已在 AST 486/33M. COMPAQ 386/25E 上的运行通过。

445

## 解决 Compaq 486 微机

### 使用自然码输入汉字时死机的方法

大庆石油学院 吕 靖

自然码输入汉字以其易学, 输入汉字快而深受广大用户的喜爱; Compaq 486 微机以其质量优良和性能可靠而在我国拥有大量的用户。我们单位购买了一台 Compaq 486 微机, 在使用自然码输入汉字时发现, 一进入自然码状态就死机, 需重新启动(冷启动)计算机。后经反复研究、试验, 终于找到了解决的方法。

Compaq 486 微机随机所配的操作系统是 Compaq DOS 3.31。虽然该版本的操作系统具有很强的功能, 但在内存管理方面与 MS DOS 不完全兼容。Compaq DOS 3.31 在扩展内存(Expanded Memory)管理上有三种方法可供选择: ①当所执行的软件需要使用扩展内存管理程序; ②当一启动 DOS 操作系统时, 自动激活扩展内存管理程序; ③不使用扩展内存。造成进入自然状

态既死机的原因是扩展内存管理方法选择了②, 解决死机的方法是选择①或③, 选择③虽然解决了死机的问题, 但造成计算机资源的极大浪费, 不可取; 选择①比较好, 既可解决用自然输入汉字的问题, 又可执行一些大型的软件, 但不能先运行大型软件然后再进入自然码, 否则还是死机。能否进入自然码状态, 主要取决于是否激活了扩展内存管理程序。查看的方法是: 键入 CEMM, 然后回车, 若屏幕上显示 CEMM Inactive 时, 就可进入自然码状态; 若屏幕上显示 CEMM Active 时, 重新启动(热启动)计算机, 然后再进入自然码状态, 否则死机。我们在实际应用过程中, 选择①管理扩展内存的方法, 使用效果良好。

446

## 关于第三软驱动器 的应用

四川省培陵地区卫生局 彭禾

国内 286/386/486 微机大都为双软盘驱动器配置, 随软盘趋向小型化、大容量、高速度, 近年的规格以 1.2MB+1.44MB 软驱组合较为常见。由于 1.2MB 驱动器的磁轨密度为每英寸 96 道 (96TPI), 而 360KB 驱动器相应为 48TPI, 前者磁轨宽度仅及后者二分之一, 因此, 用高容量软驱写的 360KB 磁盘, 往往在低密驱动器上不能读或读出信息可靠性差, 尤其是驱动器陈旧或/及低质量磁盘。正如 DOS V5.0 手册提示: “因硬件差异, 有些 360KB 驱动器不能可靠地读出在 1.2MB 驱动器中用开关/F: 360 格式化的软盘 (数据)”, 此类软盘难能在 PC/XT 上正常使用, 实例众多, 误事不浅。

PC DOS V3.30 有含 EXDSKBIO.DRV, 支持在 IBM PC/XT Model 286 等机型上安装第三软盘驱动器, 需添插适配卡等, 国内用户甚少。

AST P-P II-PP 型微机主板上均有第三软盘驱动器接口——针式连接件, 添加 360KB 或其它驱动器十分方便, 随机提供的‘前置性’第三软驱动程序 ASTDSK.DRV 及 DOS DRIVER.SYS 共占用基本内存 4KB (3824B+224B), 设置后上述问题即迎刃而解。实例如下:

AST PP3/33 213V

第一软驱 1.44MB 第二软驱 1.2MB 第三软驱 360KB (添置)

C: 33MB D: 100MB E: 80MB

MS DOS V5.0-CONFIG.SYS:

SHELL=C:\STA\COMMAND.COM/P [置子目录中]

COUNTRY=002..C:\STA\CONTRY.SYS [取中国式日期]

DEVICE=C:\STA\ASTDSK.DRV 0 [第三软驱前置驱动程序]

DEVICE=C:\STA\DRIVER.SYS /D: 2 /F: 0 [第三软驱 DOS 驱动程序]

DEVICE=C:\STA\HIMEM.SYS

DEVICE=C:\STA\RAMDRIVE.SYS 2048 /E

DEVICE=C:\STA\SMARTDRV.SYS 1216

DOS=HIGH

DEVICE=C:\STA\DRIVER.SYS /D: 0 /F: 7 [添第一软驱逻辑名]

DEVICE=C:\STA\DRIVER.SYS /D: 1 /F: 1 /C [添第二软驱逻辑名]

DEVICE=C:\STA\ANSI.SYS [供 CCBIO 2.13H]

FILES=35

BUFFERS=2 [置 2 少占基本内存]

启动微机, 屏幕显示如下, 告知安装成功

```
:
Loaded External Disk Driver for Drive F [装载第三软驱 F:]
```

```
:
(.....virtual disk G:)
(.....Disk Cache)
```

### 认识

一. AST P 系列微机添第三驱动器所需部件属通用件。若缺指定的信号线 (用户手册: 5. 25" drive-P/N 220508-002. 3. 5" drive-P/N 220505-002) 可购 IDE 制式硬盘适配卡用软驱信号线替代, 效应良好, 未安装第二硬盘的微机上供软驱用电源连接件现成, 无需叉接电源连接件。

二. 第三软驱安装的关键在信号线的正确连接。在 PP 型微机上, 信号线近端插入主板上第 1-2 软驱连接件 J24 前面的第 3 软驱 34 针连接件 J25, 它位于软驱与开关电源之间, 空隙小, 插接时视野更窄, 需倍加细心, 谨防压歪连接针, 导致装载出错 (提示检查第三软驱所有的连接用品)。驱动器端必须接相当用于 A 驱的连接件, 闲置位置居中原供 B 驱的连接件。P II 型第三软驱 34 针连接标识为 P4。P 型机上取名 J20, 位于开关电源左侧, 安装操作有所不同, 详见用户手册。

三. 若第三软驱为 3.5" 规格, 其信号线驱动器端应有两组 3.5"+5.25" 软驱连接件, 购置时试插前组连接件于 3.5" 驱动器无误。

四. 第三软驱动盘符常规列硬盘及其逻辑盘符之后, 本例命名为 F:。物理驱动器不超过五台, 在 DOS V3.0 及高版本的 CONFIG.SYS 中无需置 LASTDRIVE 命令行, 任取默认值。若使用 SUBST 命令, 以上述为例, 当添 LASTDRIVE=J。

五. 第三软驱功能犹同 A: 及 B:, 执行 360KB 软盘全盘复制命令:

d>DISKCOPY F: F: 或 F>d: [\path \]

### DISKCOPY

其结果一如既往。但性能优良的软盘整盘复制软件 DUPLICATE.EXE V3.0 不认第三软驱, 对原为 1.2MB+360KB 软驱组合的微机, 宜将 360KB 软驱调整为第三软驱。

六. 高容量软驱添逻辑名, 主要为同规格高容量软盘间单文件或文件组复制提供方便, 省去动用‘中间盘’—硬盘或虚拟盘之烦。360KB 软驱无需添逻辑盘名。

七. 3.5" 软驱采用扇区密位技术, 置 720MB 磁盘每磁道 9 扇区, 1.44MB 磁盘增至 18 扇区/磁道, DOS V5.0 支持的 2.88MB 软驱取 36Sec/T, 皆无上述弊端——‘不兼容’。

# 两种流行汉字输入法之我见

吉林珲春计划经济委员会 郎加夫

在我国,有多种多样的汉字输入方法。如五笔划、五笔桥、五笔字型、音形码、首尾码、大众码、声韵部形码、人大码、区位码、电报码、双音双拼、自然码、层次四角码等。每一个使用汉字输入的人,都会选用其中的一种或数种。而各种汉字输入方法之间,有些有联系,有些则完全没有联系,它们有各自的特点和应用范围,对不同的使用者有着不同的效果。

## 五笔字型汉字输入法

五笔字型汉字输入法,是中文输入速度第一次超过西文输入速度的一种汉字输入法,它符合人们的写字习惯,即使对某个字不认识,仍然可以利用本法进行输入。大多数专业计算机录入员和一些作家都使用这一方法。但初学者在使用这一输入法时感到困难的是要把130多个字根背熟。记下130多个字根的形象之后,每个字根还要对照键盘上的位置,分别放到各区中,在使用时还要有识别码。可见要学好五笔字型是相当困难的。有人主张先把键盘记住并记住25个键名,再将字根分成五组,对应每一个键。记住键盘后,再熟悉汉字拆分原则,之后再上机实践。也有人主张边学习边上机实践来增加记忆。无论怎样,学习这个方法都需要大量的记忆。

## 自然码汉字输入方法

自然码汉字输入方法是在拼音输入方法基础上加以改进而形成的一种双音双拼汉字输入方法。自然码系统是北京超想技术开发公司软件专家周志农完成的。是一个以音为主,以形为辅的,集各种汉字输入方案的功能为一体的汉字输入软件。音码部分采用与CCDOS拼音相近的双拼编码,其记忆量仅比CCDOS拼音多一倍。输入单字时可以选择输入,也可以加上部首的声母(形部)进行盲打输入。盲打输入时其单字的重码率与五笔字型相当,字词混合重码率比五笔字型低得多。

自然码系统是一个先进的以词为主导、字词混合编码、智能处理的第二代汉字输入软件,还有极强的可扩展能力,在输入过程中就可以随时增加词汇而不影响输入,可随造随用,有音码容易掌握、反应快等特点;又有形码效率高、可输入不认识字的长处。

这种方法,简单易学,几乎不需要记忆。双拼双音,每个汉字只要打两个键即可以出一个汉字。但自然码中有大量的词组,尤其是双字词很多,而双字词的重码率很低,这

样利用自然码输入方法进行词组输入是十分快速的。双字词、多字词均是四个键完成输入,这与五笔字型相比是相同的按键数,故自然码的输入速度也是较快的。由于这是一种拼音输入方法,对早已熟记拼音的人来说,是不需要花太多时间就能掌握的。

它的强大的造词功能是十分有用的。如“墙体材料”在某一篇文章中几十次出现,那么就可以在文章输入的同时,将“墙体材料”造成一个词,取四个字的声母作为代码,这样在以后的文章输入时只要打入这四个字的声母,即可以将“墙体材料”一词输入。而对于这个词组也不需要记忆它的代码,一看到这个词,即可以联想出它的代码来了。这对于非专业计算机录入人员来说是十分方便的。因此在我国,自然码汉字输入方法是一种良好的方法。

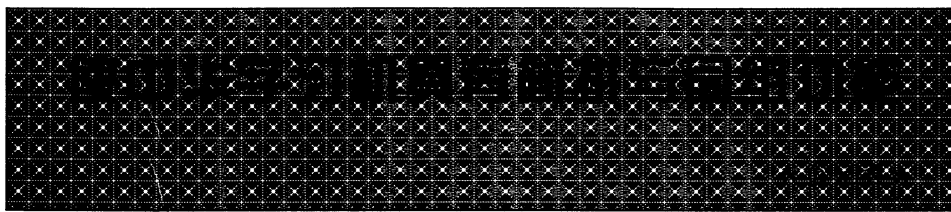
不论用哪种汉字输入方法输入汉字,都有一个熟练程度问题,若不熟练,再好的汉字输入方法也是不能提高输入速度的。但对于不同的汉字输入方法,有着熟练时间长短之分的。如五笔字型汉字输入方法,要想熟练,最快要个把月的时间,而自然码汉字输入方法最多一周时间即可熟练操作。

用什么方法可以取得高速输入的效果则因人而异,据“在1992年9月,海峡两岸中文电脑输入表演赛”上的统计结果,在第三名中才有使用五笔字型输入法的。作为非专业汉字输入人员,在使用五笔字型输入时,只有熟练才能快,而自然码汉字输入系统只需少量的记忆,有些在键盘上已标出相应记忆的字符,这样不论何时使用均可以在几分钟内恢复记忆,加快输入速度。

总之,在汉字输入的各种方法中,不论那种方法,都有其独特的特点,各有自己利的一面,也有弊的一面。例如对小学一二年级学生来说,用拼音方法是好的。一方面使学生练习了拼音,另一方面也使学生加快了识字的速度,增加了识字量;对小学高年级学生用音形汉字输入方法较好,可以帮助学生识别音和形;对于专业计算机汉字录入人员来说,五笔字型好些,可以进行快速盲打,提高输入速度;对于非专业计算机汉字录入人员来说,自然码好些,可以减少记忆量,随文章而随时增加词组。

有了各种不同的汉字输入方法,人们可以根据自己的情况,来选择不同的输入方法,以利于自己的工作需要。

中华学习机内存中固化了国家标准二级汉字库，为



编成的词组库生成源程序附后，供参考。

我们使用提供了方便，但其汉字输入法只有区位和拼音两种。与目前长城机和 IBM 机以及四通上广泛使用的多种汉字输入法相比，不能不使人感到有些不足，但有幸的是，设计者为我们留下了 F4 和 F5 两键，供扩充新的输入法，为了充分利用这两个空闲的端口，笔者参考了有关杂志中介绍的方法，集各种特点于一体，利用汇编语言编制了一小段程序，使中华学习机上具有了简拼和词组的输入功能，大大方便了用户，同时也将这两种方法移入了重庆版的 CWS 中，现将主要功能和使用方法介绍如下：

首先将词组库调入内存，再输入下面的这段汇编语言程序，用 JPCZ 存盘，然后运行，即可生成一个带简拼和词组的系统，程序定义 F4 键为简拼，F5 键为词组，简拼定义的拼音如下：

```
V.....ZH    I.....CH    U.....SH    J.....AN
K.....AO    L.....AI    F.....EN    H.....ANG
G.....ENG    S.....ONG    Y.....ING
```

此简拼输入法最大的特点是：一个汉字不管有多长，最多三键就可输入，解决了输入速度慢的问题，如果再配合使用词组输入，则将如虎添翼，输入速度大大提高，甚至可与 IBM 机输入速度媲美，另外，本简拼采用的拼音缩写方法与 IBM 机输入方法完全相同，因此不管在 IBM 机上还是在中华学习机上，不需要记忆两套键码，具有实用意义。

使用词组时，必须先建立词组库，为了方便读者的使用，特将用 BASIC 语言编写的源程序附后，定义词组时，可用一个字符或一串字符代表一个词组，如“中华学习机”可用“ZHXIJ”，也可用一个“Z”代替，读者甚至可以只敲两键，就可输入一个长达七十余个汉字的词组，此词组所采用的方法最大的特点是，可根据不同专业来建立专门词组库，特别适合中华机内存较小的特点。{可根据需要调入专用词组库}并可随时向词组库中添加新的词组。

读者还可根据有关文章介绍的方法将 CWS 中的五笔输入法拆解开，使五笔、词组和简拼分别独立成模块，使用时，在 HELLO 程序中可根据需要的选择的灵活的装入所需的模块，使 CEC 的输入编辑功能大大提高一步。

最后将用汇编语言编成的主程序 JPCZ 和用 BASIC

程序 JPCZ:

#! 注:从 9000G 开始运行

```
8F1E-7F BC
8F20-F2 7F C6 B4 3A 20 C9 88 D0 16 CE 99 03 30 06 20
8F30-77 C3 4C ED 8F A9 00 8D 99 03 20 F3 8F 4C ED 8F
8F40-AD 10 C0 AD 00 C0 C9 20 F0 1D C9 41 10 03 4C EF
8F50-8F C9 5B 30 03 4C EF 8F AE 99 03 9D 9A 03 EE 99
8F60-03 20 77 C3 4C ED 8F AE 99 03 D0 03 4C ED 8F A9
8F70-00 85 0A A9 10 85 0B A0 00 B1 0A F0 63 C9 FF F0
8F80-14 D9 9A 03 D0 0F C8 CC 99 03 30 ED B1 0A C9 FF
8F90-D0 03 4C AD 8F C8 B1 0A D0 FB C8 84 0C 18 A5 0A
8FA0-65 0C 85 0A A5 0B 69 00 85 0B 4C 77 8F C8 AE FE
8FB0-02 B1 0A F0 20 48 A9 7F 9D 00 02 E8 20 2B C3 68
8FC0-9D 00 02 E8 20 2B C3 C8 B1 0A 9D 00 02 E8 20 2B
8FD0-C3 C8 4C B1 8F 8E FE 02 A9 00 8D 99 03 4C ED 8F
8FE0-A9 07 20 2B C3 A9 00 8D 99 03 20 77 C3 A9 00 20
8FF0-AB C3 60 20 9B C3 CE FE 02 A9 08 20 2B C3 60 00
9000-A9 0B 8D 91 03 A9 90 8D 92 03 60 AD 10 C0 AD 00
9010-C0 C9 06 F0 2D C9 14 D0 26 09 80 8D AE 03 A9 0B
9020-AD 91 03 A9 90 8D 92 03 A9 80 8D 8F 03 A9 90 8D
9030-90 03 A9 8F 85 FC A9 1E 85 FB A9 00 20 6E C3 4C
9040-AB C3 09 80 8D AE 03 A9 0B 8D 91 03 A9 90 8D 92
9050-03 A9 26 8D 8F 03 A9 8F 8D 90 03 A9 90 85 FC A9
9060-71 85 FB A9 00 8D 99 03 20 6E C3 4C ABC3 4C 0B
9070-90 7F B4 CA 7F D7 E9 3A 20 AD 00 C0 C9 14 F0 EE
9080-C9 88 D0 06 20 ABC3 4C 30 F9 ADC9 03 C9 A0 D0
9090-01 60 C9 E0 D0 05 A9 A0 8D C9 03 85 D0 C9 FB 10
90A0-11 C9 E1 30 05 38 E9 20 85 D0 C9 DB 10 04 C9 C0
90B0-10 06 20 ABC3 4C DD FA AD 99 03 D0 37 20 18 91
90C0-A5 D0 A0 07 D9 21 91 F0 0A 88 D0 F8 18 69 20 85
90D0-D1 D0 0B C8 B9 21 91 85 D1 B9 22 91 85 D2 A0 00
90E0-20 ABC3 98 48 B9 D1 00 F0 08 20 E3 FA 68 A8 C8
90F0-D0 F1 68 60 20 18 91 A0 FF C8 B9 2B 91 D0 04 A5
9100-D0 D0 C9 C5 D0 D0 F2 A2 FF E8 C8 B9 2B 91 F0 CE
9110-C9 E0 30 CA 95 D1 D0 F1 A2 04 A9 00 95 D1 CA D0
9120-FB 60 D6 FA E8 C9 E3 E8 D5 F3 E8 CA E1 EE CC E1
9130-E9 CB E1 EF C6 E5 EE C8 E1 EE E7 C7 E5 EE E7 D9
9140-E9 EE E7 D3 EF EE E7 00 7F BC F2 7F C6 B4 3A 20
```

]LIST

10 D\$ =CHR\$(4):S=4096

```

15 PRINT D$: "PR #3":PRINT:HGR2:VTAB 5
20 N=120:DIM C$(N),Z$(N)
40 PRINT CHR$(12)
50 PRINT:PRINT"请选择:":PRINT
60 HTAB 9:PRINT"①生成词组表"
70 HTAB 9:PRINT"②添加新词组"
80 HTAB 9:PRINT"③生成完退出"
90 PRINT
100 GET G$:IF G$ < "1" OR G$ > "3" THEN 100
110 ON VAL(G$) GOTO 120,150,200
120 INPUT "请输入新词组库名:":F$
130 R=S:GOTO 250
150 INPUT "请输入原词组库名:":F$
160 PRINT:PRINT D$: "BLOAD":F$: ".A":S
170 R=S+PEEK(43616)+256*PEEK(43617)-1:GOTO 250
200 PRINT CHR$(12):VTAB 4:HTAB 10:PRINT "再见":
END
250 PRINT CHR$(12):PRINT"请输入代码和词组:(输入完用
"! "退出)"
260 T=0
270 T=T+1:PRINT:INPUT"词组代码:":C$(T):IF C
$(T) = "!" THEN T=T-1:GOTO 290
280 INPUT "词组:":Z$(T):GOTO 270
290 PRINT:PRINT:INPUT"要修改吗(Y/N)? ":N$:IF N
$ = "N" THEN 400
300 IF N$ <> "Y" THEN 290

```

```

310 PRINT:FOR I=1 TO T
320 PRINT I"—":C$(I):":Z$(I):":INPUT T$,T1
$:
325 IF T1$ <> " " THEN Z$(I)=T1$
330 NEXT I
340 ZH=FRE(0):GOTO 290
400 PRINT CHR$(12):VTAB 3:PRINT"正生成词组库,请稍
等..."
510 FOR I=1 TO T
520 HC=LEN(CHR$(I),J,1):R=R+1
530 FOR J=1 TO HC
540 POKER.ASC(MID$(C$(I),J,1)):R=R+1
550 NEXT J
560 POKE R,255:R=R+1
570 HZ=LEN(Z$(I))
580 FOR J=1 TO HZ STEP 3
590 POKE R.ASC(MID$(Z$(I),J+1,1))
600 POKE R+1.ASC(MID$(Z$(I),J+2,1)):610 R=R+2
620 NEXT J
630 POKE R,0:R=R+1
640 NEXT I
650 POKE R,0:R=R+1
660 PRINT:PRINT D$: "BSAVE":F$: ".A":S: ".L":R-S
670 GOTO 40

```

449

## LQ-1600K 打印机控制参数 在 FOX 程序中的应用

江苏杨中县统计局 卜符成

1600K 打印机以其较好的性能价格比,受到广大用户的喜爱,但是 1600K 打印机使用手册中介绍的打印机控制参数存在着一些不足:一是所举的例题都是在 BASIC 环境下运行,二是一些控制参数介绍得不够准确。本人把 1600K 打印机的一些常用控制参数(如字体、修饰字、字间距、行间距、压缩、划底线),以 FOX 编程方式介绍给大家。程序和结果附后,说明如下:

1、打印机控制参数可以单个使用,也可以组合使用。

2、程序中所举的字间距和行间距控制参数仅是一个例子,用户更改个别参数,可以随意定义字间距和行间距。

3、启用控制参数,可以用? 命令,也可以用@命令。

4、用户定义的控制参数,都可以用打印机初始化命令清除。

### 程序清单

```

SET PRIN ON
SET DEVI TO PRIN
? CHR(27)+CHR(64)+'打印机初始化'
? CHR(27)+'IA'+ '打印随机字体'
? CHR(27)+'IB'+ '打印倍宽字体'

```

```

? CHR(27)+'IC'+ '打印倍高字体'
? CHR(27)+'ID'+ '倍宽倍高字体'
? CHR(27)+CHR(69)+'实现强调打印'
? CHR(27)+CHR(70)+'取消强调打印'
? CHR(27)+'IA'+ '一行中'+CHR(27)+'ID'+:
'不同字体'
? CHR(27)+CHR(71)+'实现双重打印'
? CHR(27)+CHR(72)+'取消双重打印'
? CHR(27)+CHR(113)+CHR(2)+'阴影字体打印'
? CHR(27)+CHR(113)+CHR(3)+'空心阴影字体'
? CHR(27)+CHR(113)+CHR(1)+'空心字体打印'
? CHR(28)+CHR(45)+CHR(1)+'画一个点底线'
? CHR(28)+CHR(45)+CHR(2)+'画二个点底线'
? CHR(28)+CHR(114)+CHR(1)+'当前字压缩 4/1'
? CHR(28)+CHR(15)+'标准字符宽度'
? CHR(27)+CHR(64)+'打印机初始化'
? CHR(28)+CHR(83)+CHR(128)+CHR(128)+:
'字间距为 0 点'
? CHR(28)+CHR(83)+CHR(0)+CHR(10)+'字间距扩大'
? CHR(27)+CHR(51)+CHR(25)+'压缩行间距'
? CHR(27)+CHR(51)+CHR(40)+'扩大行间距'
? CHR(27)+CHR(51)+CHR(30)+'正常行间距'
? CHR(27)+CHR(52)+'打印斜体字符'
? CHR(27)+CHR(53)+'取消斜体字符'
? CHR(28)+CHR(74)+'纵向打印'
? CHR(28)+CHR(75)+'横向打印'+CHR(13)+CHR(13)
SET PRIN OFF
SET DEVI TO SCRE

```

450



## 一、说明:

1. ARJ 是一个集压缩、释放于一体的备份程序。该程序只有 ARJ.EXE 一个文件,但有许多的应用功能。这些功能的发挥,主要在于其内部命令及参数的运用。

2. 欲了解有关命令及参数的使用,可在 DOS 的功能项,键入:ARJ-? ✓

3. 每次只能选择一个命令功能关键字,但允许选择多个参数功能项。

4. 在输入命令表达式时,命令关键字要跟在 ARJ 后面,参数选择项可放在命令关键字后,也可放在表达式最后,但参数前要加一横。

以下是 ARJV2.30 的使用简例。

## 二、压缩

1. 将当前目录下的文件压缩成文件 ABCD. ARJ  
ARJ A ABCD

如果 ARJ.EXE 文件在 C: 盘的 DOS 子目录下,键入  
C:DOSARJ A ABCD

2. 将当前 AAAA 目录下的所有子目录及所有目录下的文件压缩成文件 ABCD. ARJ

ARJ A -R ABCD AAAA

3. 将当前目录下所有扩展名为 .COM 的文件压缩成文件 ABCD. ARJ。同时删去当前目录下的所有扩展名 .COM 文件:

ARJ M ABCD \*.COM

4. 在 A: 驱动器用多个软盘备份硬盘当前目录下的文件:

ARJ A A:ABCD -VVWA

压缩备份后,在第一张盘上的文件名是 ABCD. ARJ。第二张盘上的文件名是 ABCD. A01。余类推。ARJ 不会覆盖备份盘上的原有文件。

5. 将当前目录下的文件压缩成自我释放文件 ABCD. EXE

ARJ A ABCD -JE1

压缩备份后的文件 ABCD. EXE。可在 DOS 状态下,键入 ABCD ✓ 就可直接释放原被压缩的文件,而无需借助其他程序释放。将文件压缩为自我释放文件时,不能设置密码参数。

6. 将压缩文件 ABCD. ARJ 转换成自我释放文件 ABCD. EXE

ARJ Y -JE1 ABCD

7. 将当前目录下的文件压缩并设置密码 BBBB

ARJ A ABCD -GBBBB

-G 是要设置密码的

设置的密码,中间无空格。

若释放该文件,键入

ARJ E ABCD -

GBBBB

8. 将当前目录下扩展名为 .WPS 的文件压缩为 ABCD. ARJ。并设密码 BBBB

ARJ A ABCD \*.WPS -GBBBB

## 三、释放

1. 释放压缩文件 ABCD. ARJ 于当前目录下

ARJ E ABCD (ABCD. ARJ 文件也应在当前目录下)

2. 释放原多层目录压缩的文件 ABCD. ARJ 于当前目录下,在释放时忽略询问提示:

ARJ X ABCD -JYCNY

若命令关键字 X 改为 E (即 ARJEABCD-JYCNY),则将压缩文件中的所有原文件都释放在当前目录下,而不恢复原子目录。

3. 释放软盘备份的 ABCD. ARJ 文件于硬盘的当前目录下:

ARJ E A:ABCD -VVWA

4. 将 C: 盘 AAAA 目录下的压缩文件 ABCD. ARJ。释放于当前目录下:

ARJ E C:AAAAABCD -P1

## 四、其他

1. 删除压缩文件 ABCD. ARJ 中的所有 .BAK 文件:

ARJ D ABCD \*.BAK

2. 合并压缩文件 AAAA. ARJ 于 BBBB. ARJ 中:

注意:后面的文件 AAAA 一定要加扩展名 .ARJ。若没有选择参数,两个压缩文件中的同名文件将同时保留并重复压缩。在释放时会提示是否覆盖同名文件。

3. 检验压缩文件 ABCD. ARJ 的完整性:

ARJ I ABCD

4. 修改压缩文件 ABCD. ARJ 中文件名 READ. ME 为 READ:

ARJ N ABCD READ. ME ✓

回车后程序运行暂停,要求操作者输入文件名:

READ ✓

5. 查看压缩文件 ABCD. ARJ 中被压缩的文件内容:

ARJ L ABCD

如要显示文件的更详细的内容:

ARJ V ABCD

ARJ 尚有许多功能,由于篇幅的所限,仅介绍以上常用的功能。

# ARJ 压缩备份软件使用简例

陈哨海

《聊斋志异》为台湾精迅公司 93 年推出的一套大型全中文游戏。该游戏以鬼怪为题材，情节曲折，极为新奇有趣，令人爱不释手。且难度是随着游戏进程而增大的，较易上手。但要过关并不是

## 聊斋志异指引

那么容易。因为这套软件设计比较严密，又制造一些假象令玩家走入歧路，还有不少陷阱，玩起来也特别繁琐，所以极费时间。玩家还必须有一些八卦方面的知识，才可排除某些障碍。为使您这位大英雄早日战胜赤符鬼王统治下的人民，特向各位介绍游戏的玩法及笔者的心得。

### 一、玩法

玩家在 与赤符鬼王决战前必须先过五关，下面这些是攻关的步骤：

#### 第一关

先在古盘城中右面的一间房间里找到金形石（开后此房的机关在那个站者不走的人脚下）。找到金形石后再从居民口中问出金形咒的下落，去找连启环和张萱儿并问得金形咒，才可进入木尊塔挑战木轮明王。

#### 第二关

先在寒口村问得金口术，在邻街的一间房子里找到铁甲人，施金口术，问得木形咒及木形石的下落。上毒雾山找木形石。此时必须拥有金蟾鞋（当然兔儿岛的鸿毛鞋更好，但价钱较贵，一千两银子才能买到，视玩家经济情况而定）才可，不然找到木形石后逃不出土坑。最后来到墨园土印塔前的旋转八卦上，使用木形石及木形咒进入塔内挑战土印冥王。

#### 第三关

在渡江口问出水形咒及水形石下落。水形石在千年寒冰洞，必须要涉水过河。弄不好会淹死。生命力低的（级别低）可先去映月村附近的破房子后的空地上找到浮光石，浮光石可载人渡河。来到火焰塔前的旋转八卦上，使用水形石与水形咒（必然在晚上）进入塔内挑战火焰明王。

#### 第四关

从映月村的村民口中问得瞌睡药下落（映月村的一些空房不要进去，内有陷阱），在寒口村的水池中找到瞌睡药。然后去映月村找丹士，从丹士手上得到九转灵丹（千万不要服用）及火形咒及火形石的下落。再找柳叶真人求得破体术（没有破体术打不死金轮法王），购得黑猪血，并找到火形石。接着就去金龙塔挑战金轮法王。进塔前必须已购买了通天石，通天石可看见本楼层的俯视图，方便玩家上塔擒魔。同时必须在塔内取走仕女图。

#### 第五关

在兔儿岛问出土形石下落及土形咒，再从仕女图中得到胭脂咒，买得阿鼻剑，从兔儿岛坐渡船去映月村，在映月村附近的深水潭中找到土形石。上去水月楼第三层用胭脂咒引出水月居士。

消灭水月居士后，在映月寺问出龟笛的下落及进入不夜城的方法。再去龙王角找朱丹鬼。朱丹鬼反锁自己在房内，若想进去必先破去门前的八卦阵（不懂得八卦的人往往不知从何入手）。其实

只要把八卦阵变为卦象，然后踩动门前的机关就打开房门，求见朱丹鬼，问得解尸毒的方式后。去关月桥旁的土坡上取虎头蕉，放在关月桥的旋转八卦上，施降毒术，解去村民的尸毒。然后回映月寺取得龟笛。直接去不夜城遗址（用千年龟甲查阅具体位置）。在一个旋涡（较难寻找）上使用龟笛，唤出沉于水底的不夜城，此时需破除八卦阵才可打开大门。破阵的方法是把左边的八卦阵变成卦，右边为卦，便可去后启动机关打开大门。在其中的一个正转八卦上（即逆时针旋转的八卦）使用九转灵丹进入冥府。（只要使用通天石就可在迷宫中找到出口）。进入塔顶挑战赤符鬼王。

### 二、玩法心得

玩家在开始时千万不要乱花钱，但必须施舍给“叫化子”，可弄到三粒解尸丹以救危急。尽量积累经验，多在城外打鬼怪。过关后可去长孙化宇处升级，并取得金钱。晚上不要在留在城中，因晚上城中的鬼会使你中毒（在古盘城与映月寺都有人会解毒）。

下面是必买之物：

- 1、地图竹筒（指出金、木、水、火、土五塔等位置的地图）
- 2、千年龟甲（指出您在地图中的所在座标）
- 3、通天石（指出您所在地的俯视图）
- 4、阿鼻剑（杀伤力最大的武器，最后几关必用）
- 5、万仙文（可拥有最大的法力）
- 6、天王袍（防御能力最强的护甲）
- 7、鸿毛鞋（可飞跃较高的物体，且不会在树林中困死）

其中第四、五、六、七项只有兔儿岛才可买到（可较迟买）其余的可买可不买（视经济情况而定）如：千年灵芝、爆雷弹等。

特别要注意的是——过关次序不可超越。因为每过一关后才可询问出下一个过关用的宝物及所在地。只有问出宝物所在地才能找到宝物。找到宝物所在地后，必须离开此地，转一圈回来后宝物才出现。

在与各魔头战斗时，魔头有许多幻化物攻击你。这时你不只是要避开攻击，还要狠打幻化物，因为幻化物才是魔头的致命弱点。

下面是一些有用的咒语（可省去询问）

- 1、金形咒：利他夜兰利利弥迦
- 2、木形咒：娑婆娑婆娑珂迦他
- 3、水形咒：弥利娑婆娑多他夜
- 4、火形咒：利娑婆利他夜他夜
- 5、土形咒：夜夜娑婆利利他夜
- 6、破体术：利利他夜
- 7、胭脂咒：娑婆娑婆娑婆
- 8、降毒术：利多兰珂
- 9、金口术：娑珂娑夜
- 10、解毒术：利弥弥夜
- 11、恢复体力书：弥夜多利

玩家还要多用存储功能，以免前功尽弃。

# “快打至尊”之“必杀技”

大连理工大学 王智

快打至尊是近来比较流行的一个计算机游戏，它类似于大型游艺机的 Street Fighter。所以深受广大青少年计算机爱好者的喜爱，该游戏分低、中、高三等，共有八人供选择，每人各有一身必杀技，掌握了这些必杀技，你就能顺利过关，就能由低级向高级进军。下面就介绍功能键和必杀技的使用方法：

功能键设置：

Q	W	E
A		D
Z	X	C

Tab——拳

Shift——脚

Q——向后跳

W——向上跳

E——向前跳

A——向后

D——向前

Z——向后蹲

X——向下蹲

C——向前蹲

各人必杀技使用方法：

斩：

天雷斩——顺序按 D、C、X、Tab

旋风腿——顺序按 X、Z、A、Shift

过肩摔——靠近对手，再按 Tab

铁头功——先按 Z 蹲下，然后同时按 E 和 Shift

萨尔可夫：

回旋抛——靠近对手，同时按 Tab 和 Shift

碎头捶——先按 X 蹲下，然后同时按 Tab 和 Shift

折腰摔——靠近对手，再按 Tab

子弹：

蜻蜓点水——顺序按 X、C、D、Shift

滚踢——后退两步，同时按 D 和 Shift

连环踢——顺序按 X、Z、A、Shift

碎头摔——靠近对手，再按 Tab

凤凰：

火凤凰——后退三步，同时按 Tab、Shift、D

回旋脚刀——顺序按 X、Z、A、Shift

蝎子尾——顺序按 A、Z、X、Shift

倒挂金钩——先按 X 蹲下，再同时按 TAB 和 Shift

法兰克：

翔龙拳——前进二步，顺序按 X、C、D+Tab（注：同时

按 D、Tab）

气动拳——顺序按 X、C、D、Tab

空旋摔——靠近对手，同时按 Tab 和 Shift

鬼丸：

鬼啃头——靠近对手，再按 Tab

旋风忍术——后退二步，同时按 Tab 和 D

龟忍术——同时按 Tab 和 Shift

岚：

劈挂掌——后退二步，同时按 Tab 和 D

大佛脚——顺序按 X、C、D、Shift

双截踢——同时按 Tab 和 Shift

大车轮——靠近对手，再按 Tab

龙卷风——顺序按 X、Z、A、Shift

金太极：

八卦掌——顺序按 X、C、D、Tab

太极波——顺序按 X、Z、A、Tab

疾刺——后退二步，同时按 Shift 和 D

空抛——靠近对手，再按 Tab

太极刀光——后退二步，同时按 Tab 和 D

相信在掌握了以上这些技七之后，你们必将在此游戏中得到更大的乐趣。

## 三国演义修改法

马翔

## 简单巧妙修改《波斯王子》

西华

许多玩家在游戏过程中会发现：难度较低时，可以轻而易举地统一天下，不能看出三国演义的趣味所在。当难度较高时，游戏却变得十分困难。原因是即使将领的属性（体能、谋略、战力、魅力）高于敌手时，在战争中都会显得较为逊色，游戏的跨度太大。这样，便很需要提高将领的属性，而且要大大高于敌手。但是主公的物品又不可以无限地增加将领属性，且将领的属性在增加到 90 时，便无法提高。使主公难于统一天下，难于与别国主公争雄。因此，玩家们都希望能在瞬间提高将领属性，甚至体能高于张飞（100）；谋略胜于诸葛亮（100）；战力赛过吕布（100）；魅力超过刘备（100）。其实这并非不可能，只要用 PCTOOLS 对文件略加修改，就可轻而易举地实现这一梦想。具体操作如下：

假设游戏记忆中储存了新郡主在远东一项，且新郡主的体能为 95，谋略为 90，战力为 65，魅力为 65。进入 PCTOOLS，用 FIND 对文件 DATA2.GRP 进行查找，查到“5F5A5A41”（注意：PCTOOLS 中 FIND 只能查找十六进制数据，需把十进制数据化为十六进制数据方可，5F5A.5A.41 就是 95,90,90,65 的十六进制数）所在的地址，并对其进行修改，上个数据都改为 78。当你再进行游戏时，便会出现一位无论体能，谋略，战力，魅力都是 120 的新郡主，真是“天地英雄气，千秋尚凛然。”这样，无论在何难度内，新郡主可以在几十回合内活捉吕布，略施小计便可制服诸葛亮。

只要重复以上的操作，就可以把各将领的属性都大大提高。使你充满信心，再一次完成千秋霸业。

注意事项：

1. 各位在修改将领属性时，要注意是否有不同的将领具有相同的属性，因此，在修改时，要密切注意，否则，就会帮助别国将领提属性，反而于己无益。

2. 各位在修改将领属性时，千万不要过于贪心。否则就会使将领属性畸变，影响程度正常运行，修改时，最大不能超过(127)10。

3. 如果多位将领属性被修改，反而会大大减少游戏的乐趣，使游戏过于容易，失去三国演义独特的吸引力，每个主公修改五人以下为佳。

4. 用 PCTOOLS 还可修改金、粮数量。

454

笔者非常喜欢玩电脑游戏，精彩的《波斯王子》更是爱不释手，然而总觉得此游戏限制的时间和生命点数太少。拜读了 92 年第五期的《修改〈波斯王子〉》一文，喜出望外，但按照该文所说去修改时，却行不通——在所示地址处的数据非文中所示。

近日，笔者找到一条捷径。《波斯王子》总共有 29 个文件，其中，25 个数据文件（扩展名是 .dat），1 个执行文件（prince.exe），另外还有 3 个文件：prince.sav, prince.hof, prince.hlp。这三个文件各有妙用。prince.hlp 里有密码表及键盘控制命令（可用 type prince.hlp>prn 打印出来）。prince.sav 长度为 4 个字，第一个字存放时间，可以从 0000H 到 0FFFFH；第三个字存放关数，可以从 00H 到 0EH；第四个字存放生命点数，可以从 0000H 到 0FFFFH。当关数为 0EH 时，王子径直往左走 3 个版面，就可与美丽的公主相聚。此时背景响起美妙的音乐，接着出现一个说明，大意是：Tyrant Jaffer 死了，他的统治崩溃了，波斯人民为他们的勇敢的王子欢呼。不管是谁，只要拯救了美丽的公主，他将成为“波斯王子”。接着，出现一幅画面，题头是“波斯王子”，你可以输入你的名字，成为“波斯王子”。而名字就存放在 prince.hof 里。利用 pctools 可以很方便地修改 prince.sav 和 princ.hof。预先设定好 prince.sav 中的时间、关数（要大于 3）和生命点数，然后进入游戏，当屏幕出现“Loading...”时，同时按“Ctrl”和“L”，即可进入预先设定的那一关，并带有所设定的时间和生命点数。因此，你可以随心所欲地玩任何一关，而不必担心时间和“血”不够。

笔者在此还想给广大的“玩家”们一个提示，拿到一个不熟悉的新游戏，看看它的文件组成，在诸如。HLP、.DAT、.TXT 等文件里，往往可以找到“攻关秘诀”和游戏技巧。特别是现在许多游戏都有了储存功能，更要重点分析分析。SAV 文件（如果有的话），你定会找到你想要的东西。祝大家玩得开心！

455

尊敬的编辑老师：

您好。

拙作《能使软盘翻倍的程序-800》在贵刊 93 年 2 月号上刊出，本人深感荣幸。不过排版有两处失误：一是题目“能使软盘翻新的程序-800”应为“能使软盘翻倍的程序-800”；二是其大小应为“不足 10K”。

自该文刊出后，本人陆续收到各地读者许多来信，因本人当时忙于毕业设计和工作分配问题，难以一一复信。因此借贵刊稍作声明：该软件不足 10K，但源程序 800II.ASM 长达 26K 之多，全部打印需几十页打印纸，恕难以寄送源程序；本人乐意与各位交流软件，请来函联系。本人通信地址：北京二二〇信箱，邮编：101149。

另外，本人将把软件“800II”寄电脑杂志社，欲索该软件请与编辑部联系。

祝编安！

杨海军

93.11.1

**编者：**800II 文件已保存在本期盘中，有意要该程序的读者可用索取程序盘的办法向本编辑部索取。（具体办法见本刊 93 年第 3 期程序第 39 页）

编辑同志：

你们好！

我是贵刊的忠实读者，见贵刊经常刊游戏求解启示，也想请你们替我刊登一启示。我有一个中文游戏《侠客英雄传》，玩了数月之久，仍毫无进展，其中我找到武当、峨嵋、四川唐门，学会许多功夫，但最后的丐帮大会和少林寺却无法进入，另外还有一密洞，进入后询问密码，但不知密码如何寻找。因此我希望借助贵刊读者中的游戏高手指点一二，将不胜感激。

## 几封读者来信

另外，我还有许许多多好玩的游戏愿意和此中爱好者分享，有需者可来函索取。（联系地址：浙江省舟山市电力公司变电工区，邮编：316000）

你们的忠实读者：傅剑

93.11.7

**编者：**《侠客英雄传》的攻关秘诀已有高手投稿，该文将在本刊 94 年第 1 期刊出，敬请留意。

电脑编辑部的全体人员：

你们好！

关于贵刊第 7 期第 28 页登载的那个《勇者斗恶龙》有关的提问，我略知一二。

其中傅剑先生所提的“金匙”的寻找方法是在一个叫“萨汗镇”的地方开始的。通过与那里的居民对话，可以知道原持有“金钥匙”的主人十分宠爱狗。所以可在此镇唯一的一只狗的附近调查，便可发现“金钥匙”。不过“萨汗镇”是在茫茫大海的中央，十分难找，且在上海上行船时不断会有很多实力较强的怪物来袭击你。

另外值得一提的是，本游戏中的三个主人公，其中一个的等级升至一定程度时，会有一种能够打开所有门的法术（包括金门），那时有无金钥匙也无关紧要了。

在此祝傅剑君玩得开心！

此致

敬礼

李詮

93.8.17

编辑同志：

您好！

本人非常欣赏贵刊的“游戏乐园”栏目。因为我是一个电脑游戏爱好者，但目前我有一个游戏——《三界论》，于第三块版图中，因寻找不出密文而玩不下去，盼能通过贵刊一角向全国广大游戏爱好者求

解。

若有需要此游戏者亦可来函索取（联系地址：广西柳州市飞鹅路菜园屯 87 号，邮编：545007）

此致

秦庆忠

93.10.18

编辑老师：

您好！

我是《电脑》的忠实读者，现在有个游戏上的疑问，希望您能在中拨冗指点。游戏《黑暗世界(DARKSEED)》是一个美国恐怖游戏，由台湾的一家公司汉化，讲述的是异形故事，但我现在已经拿到了黑暗世界中的望远镜、圆锹、逗狗棍子、棺材中的日记和时钟钥匙等物，但现在却找不到麦肯基带回的物品，同时也不知该怎么对付警察。另外，开始时借书曾提到“找一个相关的朋友要一个有用的字”，这个朋友是谁？是什么字？若您把这个游戏玩通关，望指点一二。（通信地址：河南郑州上街长城铝业公司计控室自动化室，邮编：450041，电话：8922922）

李元峰

93.10.5

**编者：**以上是几位玩家的“求救信”，望这一方面的高手不吝赐教，赶快投稿。稿先到先刊出。

# CMOS 和分区参数表的大敌——1107 病毒

胡向东

最近笔者连续修复数个硬盘，发现了目前在广州流行的一种对 CMOS 和硬盘主引导区极具攻击性的新病毒——1107 病毒，目前下发的 KILL 软件及 SCAN108 检测均不能测出，应该提醒人们高度重视。

1107 病毒是一种文件型病毒，专门感染 COM 文件和 EXE 文件，使文件约加长 1107 字节，故笔者取其名为 1107 病毒。

1107 病毒更改了功能调用中断 int21H。其病毒 int21H 的入口在  $\times\times\times\times:01B0H$ 。在带有 1107 病毒的内存中断向量表 0000:0084 处可看到病毒的入口地址。

1107 病毒的 int21H 类似于其它文件型病毒，病毒驻留内存后，就会通过文件的读写，感染硬盘或软盘的 EXE 文件和 COM 文件。

该病毒最大的特点在于通过更改键盘硬中断 int9 和时间硬中断 int8。精心设计了一个恶作剧。

int8 是一条时间硬中断，（在 PC 机上是每秒发生 18.2 次），而该病毒设置了一个初值为 FFFAH（即 65530）的计数器。

每执行一次病毒 int8，计数器减一，然后判断计数器是否为零，若不为零，则转到原来的 int8 去执行。

而病毒 int9，起着监测按键情况的作用，每当用户开机后按一次键，就会执行一次键盘，硬中断 INT9。而病毒设计者通过每执行一次病毒 INT9，就恢复一次计数器初值，使计数器的值还原为 FFFAH。

而当病毒 INT8 检测到计数器的值为零，即感染 1107 病毒的电脑，开机后不按键若干分钟（比如说去接个电脑或中途接洽客人等），相隔时间长短与机器时钟的主频有关，病毒 INT8 就会象魔鬼一样跳出来作恶。

1107 病毒发作时第一个攻击对象就是 CMOSRAM 中的内容，它通过 CMOSRAM 地址的  $\square$  地址 70H 和数据的  $\square$  地址 71H 对 CMOSRAM 中的系统设置等内容大肆破坏。

其次就是用一大堆垃圾来复盖硬盘的主引导扇区，毫不留情地破坏硬盘的分区参数表。从而重新启动后无法正常进入硬盘。

在带 1107 病毒的中间向量表 0000:0020 和 0000:0024 中，分别可以看到病毒 INT8 的入口地址为  $\times\times\times\times:016B$  和病毒 INT9 的入口地址  $\times\times\times\times:0113$ 。

1107 病毒的另一个特点是加密技术达到了较高的水平，值得我们借鉴。

该病毒采取的是一边走一边译码的方法，即后边的机器码，是与前边某一段机器码运算后还原的，若用 DEBUG 等调试工具将该病毒从头到尾打印出来，打印出来的程序语句就象天书一样难懂。

你若要进行动态跟踪，病毒就途插入封锁键盘的语句

```
ORAL. 1A
```

```
OUT21. AL
```

你若更改 IP 指针跳过这二句，但病毒设置的循环计数器 CX 的值很大，很难有耐心用手工跟踪方法跳出病毒的循环。

若你要更改病毒的键盘语句，病毒的后半段译码的准确性就出误差，真是一环紧扣一扣。

1107 病毒感染 COM 文件时，把原 COM 文件的前 11 个字节搬到带毒文件旁文件末尾偏移量为 0324H 的地方。而带毒的 COM 文件的文件开头的语句是：

```
100 0E          PUSH  CS
101 8C  C8      MOV   AX, CS
103 05  D5  01  ADD   AX, 01D5
106 50          PUSH  AX
107 68          DB    68
108 0001        ADD   [BX+DI], AC
10A C8          RETF
```

这可以作为感染上 1107 病毒的 COM 文件的判别依据。

1107 病毒感染 EXE 文件时，堆栈指针 SP 被更改为 063DH。指令指针 IP 被列改为 0100H。而原来 EXE 文件的 SS、SP、IP、CS 值，被依次改到带毒文件离文件末尾 0324H 的地方。而带毒的 EXE 文件开头的语句是：

```
100 2E          CS:
101 8C  06  2A  02 MOV   [022A], ES
105 0E          PUSH  CS
106 1F          POP   DS
107 B9  D0      MOV   CX, 00DA
10A BE  09  03  MOV   SI, 0309
```

根据  $SP=063D$  和  $IP=0100$  以及上述的语句特征，可作为感染上 1107 病毒的 EXE 文件的判别依据。

清除 COM 文件的 1107 病毒，可从文件末尾倒数 324H 字节处找出原文件的头 11 个字节，然后搬到原来的位置，然后文件长度减去 1107 字节，再写盘即可。

清除 EXE 文件的 1107 病毒，可把该 EXE 文件先变换成一般的文件，然后用 DEBUG 把该文件调入内存，然后从文件末尾倒数 324H 字节处依次找到原 EXE 文件的 SS、SP、IP、CS 值，然后把找到的 SS 值填到文件头偏移量为 0EH 的位置，把原来的 SP 值填到文件头偏移量为 10H 的位置，把原来的 IP 值填到文件头偏移量为 14H 的位置，把原来的 CS 值填到文件头偏移量为 16H 的位置，文件长度减去 1107 字节，然后根据这个长度换算成实际所占扇区数，填入文件头偏移量为 04H 的位置，然后把文件长度除以 512 的余数，记入文件头偏移量为 02H 的位置，接着存盘，再把该文件名改名为 EXE 文件即可。

由于该病毒破坏性极大，笔者已在发现该病毒的当晚，写出了清除 1107 病毒的解毒软件，并成功地为人们排忧解难。

需要数据修复或其它咨询的朋友，欢迎来函来电与本人联系。地址：广州五山路华师附中围墙北 22 号广州正大电脑应用与数据修复有限公司，邮编 510630，电话：7570626，7570627，5515961—5931，BB 机：3322322—4364。

新

“6.4”

病毒的特点及清除

浙江石油勘探处

秦成德

新“6.4”病毒曾有所闻,近来在我单位的微机上也发现了此病毒,但使用 CPAV (1.2 版,可诊断 1009 种病毒)及 SCAN 等软件却不能检查出来。为此,笔者对此病毒做了一些分析,现将其特点和防治方法介绍如下。

### 一、新“6.4”病毒的表现

该病毒寄生于可执行文件 (.COM 文件和 .EXE 文件)中,带毒文件一般仍可正常运行。但一旦带毒文件被执行,该病毒即驻留内存,并感染其后运行的可执行文件 (COMMAND.COM 文件不受传染)。带毒文件虽可执行,但运行速度变慢,并经常引起死机。每当 1991 年以后的 6 月 4 日,该病毒将会被激活。病毒发作时,中断正在运行的程序,在屏幕上出现一屏用英文写的反动标语 (原文略) 持续数分钟,十几分钟后继续运行原程序。

### 二、“6.4”病毒的特点

该病毒的特点及传染机制与“犹太人”(“黑色星期五”)病毒相似,估计是“犹太人”病毒的中国版变种。但没有“犹太人”病毒那样的破坏性。

病毒程序本身长 1824(720H) 字节。偏移 00H~17H 是一段引导程序,18H~1FFH 是加密后的病毒宣言,200H 开始是病毒程序代码;其中 24EH~28DH 是参数区,保存了原来的 INT21 中断地址及 EXE 文件文件头部原来的一些信息(经加密处理)等。偏移 13H 开始的七个字节“B8 00 02 50 CB F6 C7”是该病毒的标志,最后的七个字节(719H~71FH)也是这个标志。由于该标志是病毒引导程序的最后五个字节和加密后病毒宣言的前两个字符组成的,没有对应的 ASCII 码显示,因此,不易识别。

该病毒感染 .COM 文件时,是加在原程序的前面,并在原程序的后面再加上七个字节的病毒标志作为结尾。因此,感染后的 .COM 文件比原文件增加了 1831(727H) 个字节。其中,包含三个病毒标志(偏移 013H、719H 及最后七个字节处)。

感染 .EXE 文件时,病毒程序加在原程序后面,同时修改文件头部的有关参数,使 CS:IP 指针指向病毒程序的偏移 200H 处,从病毒程序代码开始执行。由于该病毒程序必须从某一节 (Paragraph) 的偏移 00H 处开始,因此,如果原 .EXE 文件的长度不是 16 的整数倍,增加的病毒程序与原程序尾之间将插入若干字节随机数据。于是,对于 .EXE 文件,病毒感染后长度将增加 1824(720H) 至 1839(72FH) 字节不等。

### 三、“6.4”病毒的诊断

如前所述,现有的病毒诊断软件大都不能诊断出“6.4”病毒的存在。因此我们应根据上述的病毒特点来对其进行诊断。

#### 1、对可执行文件的诊断

如果可执行文件保留有备份,可比较两者的文件长度。若硬盘上的 .COM 文件比备份文件加大了 1831 字节或 .EXE 文件加大了 1824 至 1839 字节,则基本可以认为是被感染了。再用 DEBUG 命令来分析该文件,如在其中发现上述病毒标志,则可认定其中包含了病毒。为方便起见,可编制一个批处理文件来诊断可疑文件。这个批处理文件可命名为 CVF.BAT (Check Virus—File),其内容只有一行:

```
C:\DS\DEBUG %1 < CVF.DAT
```

再编制一个 ASCII 码文件 CVF.DAT,内容如下:

```
S CS:00 820 B8 00 02 50 CB F6 C7
```

Q

将这两个文件拷贝到待诊断文件同一目录下,即可对可疑文件进行检测。用法是

```
C:>CVF Filename
```

这里,Filename 是可疑文件的文件名(包括扩展名)。如果该文件带有“6.4”病毒,则 DEBUG 的 S 命令将搜索到两个病毒标志的所在地址,显示出

```
XXXX:0013
```

```
XXXX:0719 (对于 .EXE 文件)
```

```
或 XXXX:0113
```

```
XXXX:0819 (对于 .COM 文件)
```

如无上述显示,则可认为该文件未被感染或不是“6.4”病毒。

#### 2、对内存进行诊断

为了诊断内存中是否有“6.4”病毒,可以检查中断向量 INT21 是否被修改,但这样做比较麻烦。一个简单的方法是运行一个已知的无毒程序,然后对比其运行前后的文件长度。也可编制了一个批处理文件来做此工作,将其命名为 CVM.BAT,内容如下:

```
@ECHO OFF
```

```
CD DOS
```

```
COPY TREE.COM TREE-V.COM > NUL
```

```
TREE-V > NUL
```

```
DIR TREE*.COM
```

CD \

这里,笔者选用了DOS的外部命令TREE.COM来作为测试程序(应确定该程序本身未被感染,当然也可用别的可执行文件来代替)。运行后,自动列出TREE.COM和TREE-V.COM两个文件的目录。内存中无病毒时,两者的文件长度应一致;如已有病毒驻留,则TREE-V.COM将被感染,其长度会比TREE.COM大1831字节。

#### 四、消毒及免疫

对于内存中的驻留病毒可以简单地用重新启动机器的方法清除;但必须确保AUTOEXEC.BAT文件中所涉及的文件都是干净的,或者用干净的软盘来启动。

对于被感染的文件,如果保留有备份,可用干净的备份文件将染毒的文件加以覆盖。如果没有备份,就只能应用DEBUG等工具软件来进行人工消毒了。下面介绍人工消毒的方法。

##### 1、对.COM型文件的消毒

“6.4”病毒寄存于.COM型文件的前部,消毒时,只需将前面1824(720H)个字节删除即可。具体做法如下

```
C: >DEBUG File.COM
```

```
-R CX
```

```
CX aaaa
```

```
:bbbb
```

```
-W 820
```

```
Writing bbbb bytes
```

```
-Q
```

这里,File是待消毒文件的文件名(不包括扩展名.COM);aaaa是寄存器CX的原值(十六进制值原文件长度);bbbb是消毒后的文件长度值,应有 $bbbbH = aaaaH - 720H$ (可用DEBUG的H命令来求取)。存盘后的文件已被消毒,但仍保留了末尾的七个字节的病毒标志,可以达到免疫的目的。如果让 $bbbbH = aaaaH - 727H$ ,则可完全恢复原文件。

##### 2、对.EXE型文件的消毒

.EXE型文件消毒手续比较麻烦,因为除了截去寄存的病毒程序外,还需恢复其头部的有关参数(主要是SS、SP、CS和IP寄存器的初值)。对病毒程序的分析可知,这些参数是被加密后存放在病毒程序的偏移285H~28CH处。因此,第一步是对有关数据进行解密,获取原始数据:

```
C: >DEBUG File.EXE
```

```
-G=200 216
```

```
-D CS:285 28C
```

```
XXXX:0285 aa aa bb-bb cc cc dd dd
```

```
-Q
```

显示的数据aa~dd就是解密后的寄存器初值。其中,aa aa是IP的初值、bb bb是CS的初值、cc cc是SP的初

值、dd dd是SS的初值。把这些数据记下来,以便将它们恢复到.EXE文件头部的控制信息表中。

第二步,计算控制信息表中的文件长度字的原值。为了用DEBUG程序来获取及修改.EXE文件头部的控制信息,必须先要将文件扩展名改为非.EXE:

```
C: >RENAME File.EXE File.0
```

```
C: >DEBUG File.0
```

```
-D 102 105
```

```
XXXX:0102 ee ee ff ff
```

这里,ee ee ff ff是文件长度字的值;其中ff ff是文件所占的分区数(每分区512字节),ee ee是文件在最后一个分区中所占的字节数(注意,左边字节是低位,右边是高位)。为了从文件中截去病毒,需将文件长度数减去720H:可从ee ee中减去120H,从ff ff中减去03H(若ee ee的值小于120H,则应在ee ee中加上E0H,从ff ff中减去04H),就可得到文件长度字的原值gg gg和hh hh。

第三步,用所求得的参数修改文件头部的有关参数。

```
E 102
```

```
XXXX:0102 ee.gg ee.gg ff.hh ff.hh (修改长度字)
```

```
-E 10E
```

```
XXXX:010E xx.dd xx.dd
```

```
xx.cc xx.cc (修改SS,SP值)
```

```
-E 114
```

```
XXXX:0114 xx.aa xx.aa xx.bb xx.bb
```

```
(修改IP,CS值)
```

```
-R CX
```

```
CX pppp
```

```
:nnnn
```

这里, $nnnnH = ppppH - 720H$ ;若待修改的文件长度大于64K,则有可能CX的值ppppH小

于720H,这时,应向BX的值中进行借位相减(文件长度值的高位存放在BX中,因此还应相应修改BX的值)。

```
--W
```

```
Writing nnnn bytes
```

```
-Q
```

至此,将病毒程序从文件中截去了。最后,只需将存盘文件File.0的扩展名改回.EXE即完成了全部消毒工作。(消毒后的.EXE文件可能比原始文件在尾部多若干个字节随机数据,但这不影响其功能)C: >RENAME File.0 File.EXE

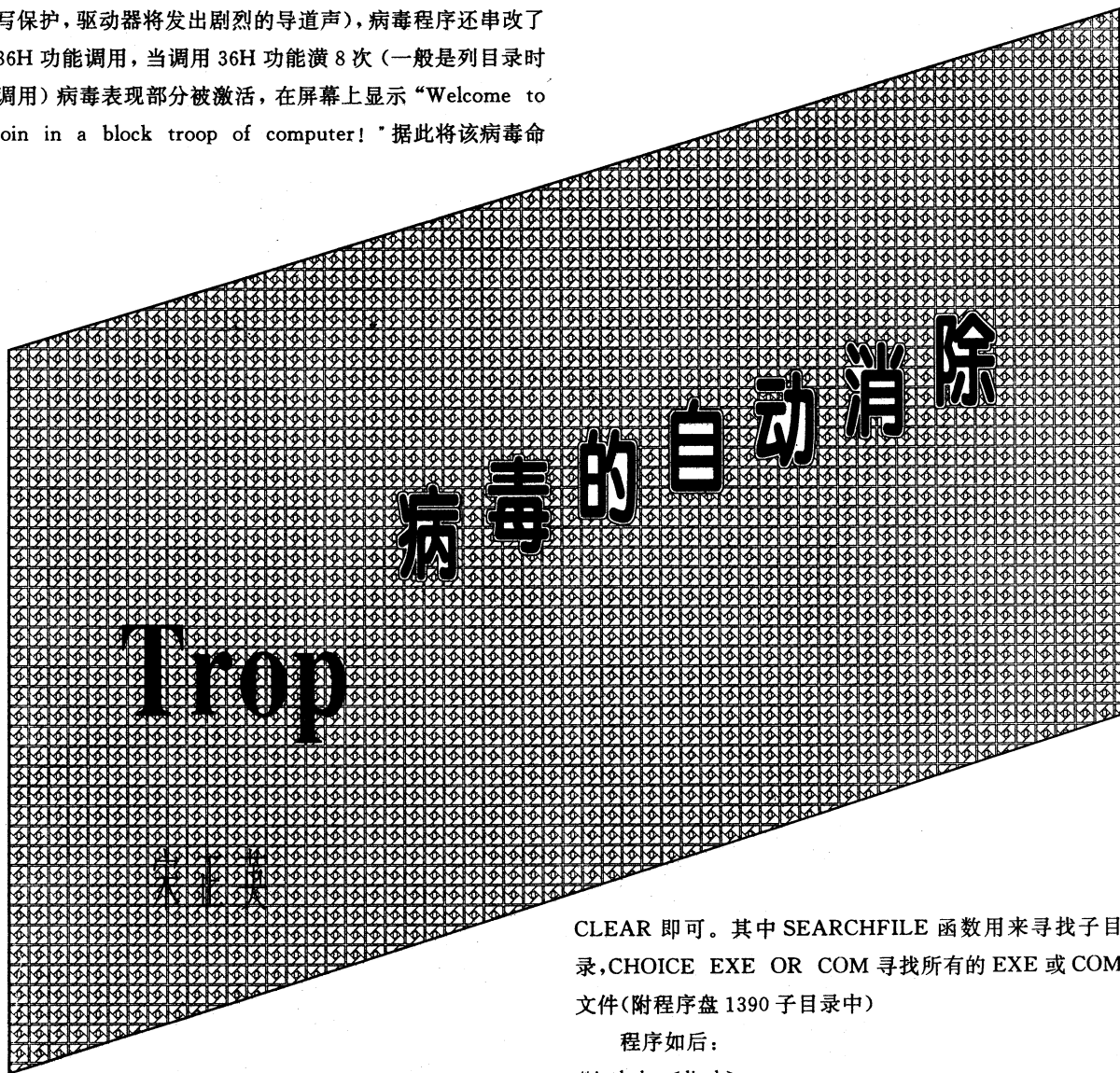
为了使可执行文件对“6.4”病毒具有免疫功能,可在文件尾人为地加上七个字节的该病毒标志“B8 00 02 50 CB F6 C7”,就可达到免疫的目的,具体做法不再赘述。(新“6.4”病毒的清除程序:458.COM在本期程序盘中)



最近笔者在计算机上发现一种病毒，用 KILL62、SCAN102、CPAV 1.2 均查不出是什么病毒。笔者就对病毒进行了分析，得知是一种攻击 COM 与 EXE 文件的病毒，该病毒通过修改内存链驻留内存，病毒有效长度 1522 字节。盗用 DOS 的 21H 号中断，每当通 4BH 功能运行程序时，病毒将试图感染被执行程序，(如程序所在磁盘贴了写保护，驱动器将发出剧烈的导道声)，病毒程序还串改了 36H 功能调用，当调用 36H 功能满 8 次(一般是列目录时调用)病毒表现部分被激活，在屏幕上显示 "Welcome to join in a block troop of computer!" 据此将该病毒命

多此法就不可取了，笔者为此特编制一病毒清除程序，编译成可执行文件后，只需键入 CLEAR 盘符回车后即可清除该盘中所有文件中的 Troop 病毒。程序用 Turbo C++1.0 编制并调试成功。

为了日后扩充方便，特留了接口函数 KILL 与 CHK-MEM。读者巨发了新病毒，可以编一解毒函数添进



名为 Troop 病毒。由于病毒还修改了 11H 和 12H 与功能调用。因此在带毒环境下，染毒文件长度不变。该病毒使用的染毒标志非常巧妙，它通过 DOS 的 57H 号功能调用改文件的设置时间秒值为 60 秒，正常情况下秒值为 0~59 秒。

用 DEBUG 可方便地清除该病毒，但如果染毒文件很

CLEAR 即可。其中 SEARCHFILE 函数用来寻找子目录，CHOICE EXE OR COM 寻找所有的 EXE 或 COM 文件(附程序盘 1390 子目录中)

程序如后：

```
#include <dir.h>
#include <dos.h>
#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include <string.h>
#include <conio.h>
void checkmem (void);
```

```

void searchfile (void);
void choiceexeorcom (void);
void outspace (int slen);
int main (int argc, char *argv[ ]).
{
    char curdriver=getdisk (). curpath[MAXDIR], drive;
    getcwd (curpath, MAXDIR);
    if (argc! =2 || argv[1][1]! =' ':
    {
        puts ("Use clear like...[BS]clear [Drivername:]
        [BS])");
        exit (0);
    }
    puts ("CLEAR v1.0 By Song Zheng Rong 1993.9
    [BS])");
    checkmem ();
    drive=toupper (argv[1][0])-'A';
    setdisk (drive);
    chdir (" \\ ");
    searchfile ();
    setdisk (curdriver);
    chdir (curpath);
}
void searchfile (void)
{
    int p;
    struct fblk dirment;
    choiceexeorcom ();
    p=findfirst ("*. *", &dirment, 0x3f); //Search sub
    ject-dir
    if (! p&& dirment. ff-name[0]='.', ')
    {
        p=findnext (&dirment);
        p=findnext (&dirment);
    } // Jump '.', ' and '..'
    while (! p)
    {
        if ((dirment. ff-attr& 0x10) ==FA-DIREC)
        {
            chdir (dirment. ff-name);
            searchfile ();
            chdir ("..");
        }
        p=findnext (&dirment);
    }
}

```

```

}
}
int kill (int my, char *fn);
void choiceexeorcom (void)
{
    struct fblk fname;
    int hav, x, y, flags, slen;
    char path[MAXPATH];
    getcwd (path, MAXPATH);
    hav=findfirst ("*. *", &fname, FA-RDONLY | |
    FA-HIDDEN | | FA-SYSTEM);
    while (! hav)
    {
        x=wherex (); y=wherey ();
        slen=strlen (path);
        if (slen! =3)
            printf ("Scanning %s(2)s", path, fname. ff-name);
        else
            printf ("Scanning %s%s", path, fname. ff-name);
        outspace (slen);
        flags=0;
        if (strstr (fname. ff-name, ". EXE") !=NULL) flags=
        1;
        if (strstr (fname. ff-name, ". COM") !=NULL)
            flags=2;
        if (kill (flags, fname. ff-name) ==0)
        {
            gotoxy (x, y);
        }
        else
            printf (" [BS]");
        hav=findnext (&fname);
    }
}void outspace (int slen)
{
    register int i;
    for (i=0; i<slens+4; i++)
        printf (" ");
}

```

# 单片机与模糊控制讲座

## 第七讲 MC6805 的指令系统和程序设计(下)

广东工学院 余永权

### 三、程序设计

MC6805 单片机的程序设计可采用汇编语言, Motorola 公司为 MC6805 单片机配套有袖珍交叉汇编程序 PASM (Portable Cross Assembler)。利用该汇编程序, 用户可以轻而易举地把汇编语言源程序处理成 MC6805 的机器代码。另外, Motorola 公司还提供袖珍交叉连接编辑程序 PLD (Portable Cross Linkage Editor), 使用户十分方便地建立和编辑源程序。

在下面给出在程序设计中常用的子程序。

#### 1. 双字节补码相加子程序 DADD

功能:  $(\$40, 41) + (\$42, 43) \rightarrow \$44, 45$

程序清单:

```
DADD:  LDA  $41  ;低字节相加
        ADD  $43
        STA  $45
        LDA  $40  ;高字节相加
        ADC  $42
        STA  $44
        RTS
```

#### 2. 双字节无符号乘法子程序 UMUL

功能:  $(\$50, 51) \times (\$54, 55) \rightarrow \$52, 53, 54, 55$

工作原理说明: 这个子程序采用常规的移位加法方式完成乘法过程。在程序中先对部分积高位清零, 把乘数放入部分积分低位。再执行下面过程:

(1) 把 32 位部分积右移一位。

(2) 如果移到进位位中的是“1”, 则把被乘数加到部分积高位; 否则不加。

(3) 测试是否循环了 16 次, 否则转回 (1), 是则结束乘法。乘法的执行流程如图 1 所示。

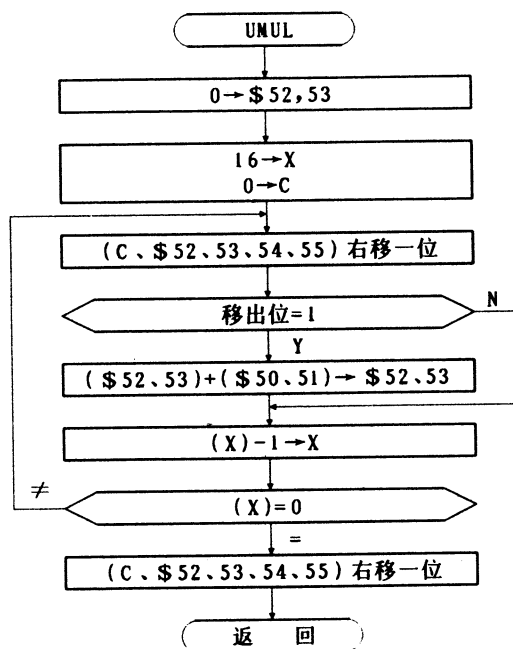


图 1 双字节无符号乘法框图

程序清单如下:

```
UMUL:  CLR  $52  0 -> $52, 53
        CLR  $53
        LDX  *16  ;循环计数器初值
        CLC
        UML1: ROR  $52  ;C: $52, 53, 54, 55 右移一位
               ROR  $53
               ROR  $54
               ROR  $55
               BCC  UML2
               LDA  $53  ;移出位为 1, 则执行加法
               ADD  $51
```

STA \$53  
LDA \$52  
ADC \$50  
STA \$52

UML2: DECX :循环次数减1  
BNE UML1  
ROR \$52 :最后再右移一次  
ROR \$53  
ROR \$54  
ROR \$55  
RTS

### 3. 双字节整数二进制到十进制转换

对于一个已知的二进制数 A, 如果求相对应的十进制数  $q_i (i=0, 1, \dots, n-1)$ , 则首先把 A 除以 10, 得出余数  $a_0$ ; 再把商除以 10, 得出余  $a_1$ ; ... 一直下去, 直到求出  $q_{n-1}$  为止。对于一个 16 位二进制, 它的最大十进制数为  $2^{16}-1$ , 即是 65535; 故用 5 位十进制数是可以表示任何 16 位二进制数。

功能: 把 (\$52, 53) 中的双字节二进制数转换成 5 位 BCD 码。说明: 结果被在 X 所指明的 5 个字节中, 低位结果放在低地址单元中。程序用 \$50, 51, 54, 55, 56, 57 作工作单元。在程序中还调用了双字节无符号子程序 UDIV。UDIV 是  $32 \div 16$  位除法, 商和余数都为 16 位; 除法过程采用左移一位相减、上商来实现。UDIV 的功能为  $(\$50 \sim \$53) \div (\$54, 55) \rightarrow \$52, 53$ , 余数存于 \$50, 51; UDIV 执行完毕时,  $C=0$  说明除法正常,  $C=1$  说明有溢出。这个程序读者可以自行编制。

双字节整数二进制到十进制转换子程序的清单如下所示:

IBTD: LDA \*5 :五位十进制数  
STA \$57  
LSLA :10→\$54, 55  
STA \$55  
CLR \$54  
IBD1: CLR \$50 :扩充为四字节  
CLR \$51  
STX \$56 :保护指针  
JSR UDIV :调用无符号除法子程序  
LDX \$56 :恢复指针  
LDX \$51  
STA X :存放一个十进制数  
INX :指针加1  
DEC \$57 :循环次数减1  
BNE IBD1  
RTS

### 4. 低通滤波器子程序

数字滤波器可实现低通滤波。一般数字低通滤波器的算法为:

$$y(k) = \alpha x(k) + (1-\alpha)y(k-1), 0 < \alpha < 1$$

其中,  $x(k)$  为本次采样值,

$y(k-1)$  为上次计算结果,

$y(k)$  为本次计算结果。

对于采样周期为 T 时, 则数字滤波器的截止频率为:

$$f = \alpha 2\pi T$$

也即是说, 截止频率和  $\alpha$  的值, 与采样周期 T 有关。

一般为了编程方便, 对  $\alpha$  的值取为  $2^{-i}$ 。这样, 在计算  $\alpha x(k)$  时, 只要把  $x(k)$  右移  $i$  位即可。在实际中, 为了更便于处理, 可把算法作一定的处理, 则有

$$y(k) = \alpha x(k) + (1-\alpha)y(k-1) \\ = \alpha [x(k) - y(k-1)] + y(k-1)$$

下面取  $\alpha = 2^{-3}$ , 并按此编制低通数字滤波器程序。 $x(k)$  的值存放在 \$52, 53 中,  $y(k-1)$  的值存放在 \$50, 51 中, 工作中用 \$54 作暂存器。结果  $y(k)$  存放在 \$50, 51。在程序中, 数据的高字节放在低地址中, 低字节放在高地址中。

程序清单如下:

DGFL: LDA \$53 :计算(X-Y)  
SUB \$51  
TAX  
LDA \$52  
SBC \$50  
LSRA :右移三位  
RORX  
LSRA  
RORX  
LSRA  
RORX  
STA \$54 :高位字节  
TXA :低位字节  
ADD \$51 :再加上→Y \$50, 51  
STA \$51  
LDA \$54  
ADC \$50  
STA \$50  
RTS

# DBASE Ⅲ、Foxbase 命令及函数 标准化转换程序

洛阳轴承研究所 梁波

**【摘要】**用户在编写程序时,对一些命令或函数(字符长度>4)常采用简写形式,看上去缺乏可读性。本文介绍的转换程序,可以将用户程序中所使用的命令或函数的简写形式转化为完整的、标准的形式,增强了程序的读性,便于维护。

一条命令或一个函数,不仅具有其自身的功能,而且还具有完整的意思,在程序中就好像一条注释一样,有利于程序的阅读。dBASE Ⅲ、Foxbase+命令及函数,若其字符长度>4,可以采用3个字符以上、全部字符以下的简写形式,以代替完整的、标准的名称,这样加快了程序的输入与编写。对于编写者来说,短期内,阅读此程序可能不会感到有什么困难。但对于他人来说,则可能会因命令或函数写得完整(如 alte, dime, recc(), deci),看上去让人感到含糊,象缺少注释一样而不便于阅读,如果编写者在完成程序的调试后,能够利用一转换程序,将自己程序中所使用的不完整的命令及函数转化成完整的、标准的命令及函数形式(如 alternate, dimension, recount(), decimals),这无疑将增强程序的可读性,也便于今后的维护。本文正是出于此考虑而编写了这一命令及函数的转化程序。

利用本转化程序,可以将用户程序中不完整的命令和函数转化成完整的、标准的形式,而不会使用户程序产生任何错误。

本文介绍这一转换程序的编写方法,并在文后附原程序清单。

## 一、数据库的建立

命令及函数字典库 COMM—FUN.DBF(存4个字符以上所有命令及函数) dBASE Ⅲ、Foxbase+的命令及函数有如下特点:

命令:同一条命令(4个字符以上)只能在同一条语句中出现一次,命令后总有

一空格,命令前是头头或者空格。

函数:函数后总是紧跟“(”,同一函数可在同一语句中重复出现。

根据其特点建立命令及函数库,库中共有149条记录。库结构如下:

字 段	字段名称	类 型	长 度	小 数	&& 注释
1	COMM— FUNC	C	013	000	标准的命令或函数
2	ADDI- TION	C	001	000	命令为空格,函数 为(

COMM—FUN.DBF 记录如下

RECOND	COMM—FUNC	ADDITION
1	accept	
2	deleted	(
3	alternate	
4	dimesion	(
5	reccount	(
6	input	
.....	.....	.....

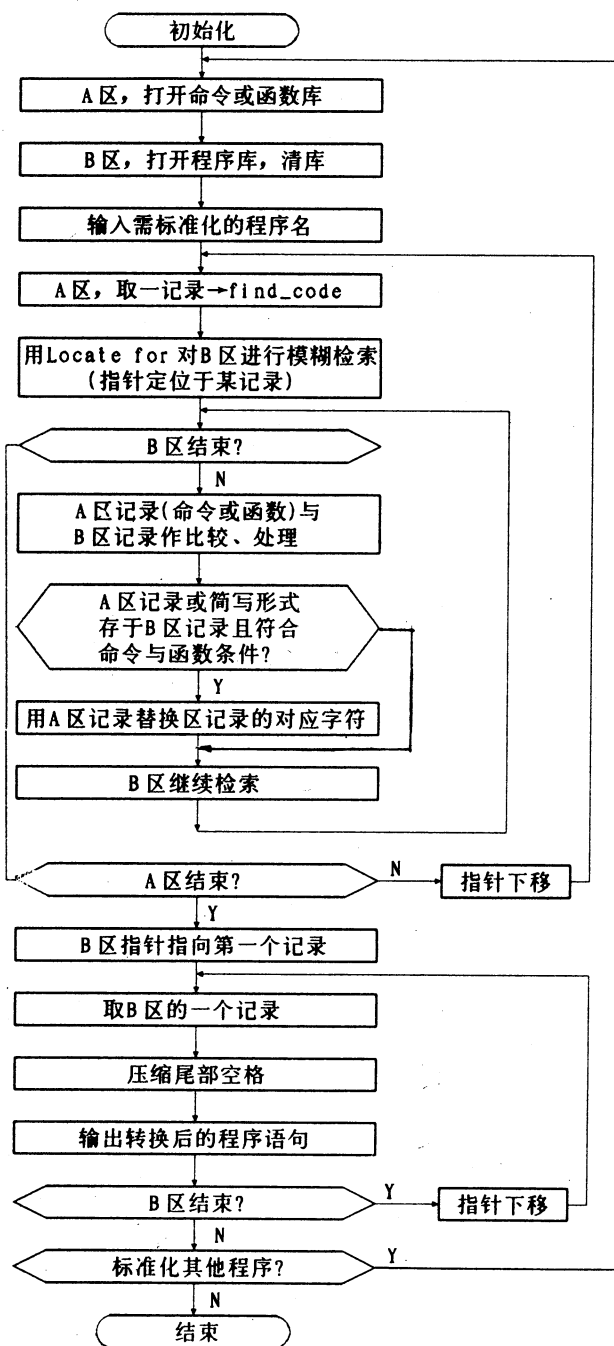
2、临时程序库 WHOLE.DBF(存需要标准化的程序语句)

## 二、程序的检索替换方法

一条字符长度>4的命令或函数的简写形式至少是前四个字符,因此从命令或函数库取一完整命令或函数的前四个字符,利用 LOCATEFOR 对程序库进行模糊检索。若存在,指针指向该记录,然后再利用命令或函数的特点,对此记录进行分析、判断,若检索命令或函数的简写形式存在,则用标准的命令或函数形式取代,逐步进行循环至记录结束;然后,再从命令或函数库取一命令或函数重复上述操作至记录结束,从而完成程序的转换。对用户程序进行转换所需要的时间与用户程序的大小有关,用户程序

越大,所用时间越长。

### 三、程序框图



### 四、原程序与转换后程序的比较

两程序的可读性及给人的感觉显然不同,转换后的程

序使人感觉程序完整,某些命令或函数就象注释一样,使得程序可读性较强。

原程序

```

set deci to 2
set alter to temp.txt
dime a(1)
usetemp.dbf
input to a—input
    
```

转换后程序

```

set decimal to 2
set alternate to temp.txt
dimension a(1)
use temp.dbf
input to a—input
    
```

附:standard.prg 原程序清单(本程序经过转换)

程序用 Dbase III 编写,若需要编译,仅对个别宏代换略作修改即可。

```

* -----
* : 程序名称: STANDARD.PRG 命令及函数标准化转换程序
* : 功 能: 对程序中所使用的命令及函数进行标准 操作
* :      (即将不完整的命令及函数写成完整的形式)
* : 目 的: 可以增强程序的可读性,便于程序的维护。
* : 使用文件: COMM—FUN.DBF 存放标准的命令及函数名。
* :      (这些命令及函数的长度>4)。
* :      : WHOLE.DBF 存放需要标准化的程序。
* :      APPEND FROM '& FILENAME' SDF
* -----
    
```

\* 初始化数据库

```

set talk off
set escape off
do while .t.
    select 1
    use comm—fun
    go bottom
    length=recno()
    go top
    select 2
    use whole
    delete all
    pack
do while .t.
    set color to gb/gb
    clear
    set color to gb/b
    filename=space(13)
    charater=space(13)
    line—len='-----' & 14*(—)
    @. 10 say 'w'+line—len+line—len+'-'
    @ow()+1.10 say 'm'+space(56)+'m'
    @ow()+1.10 say 'm'+space(56)+'m'
    @ow()+1.10 say 'm'+space(56)+'m'
    @ow()+1.10 say 'm'+space(56)+'m'
    @ow()+1.10 say 'm'+space(56)+'m'
    @ow()+1.10 say 'm'+line—len+line—len+'m'
    @ow()+1.10 say 'm'+space(56)+'m'
    @ow()+1.10 say 'm'+space(56)+'m'
    
```

```
@ow 0 +1.10 say 'm'+space(56)+'m'
@ow 0 +1.10 say 'm'+space(56)+'m'
@ow 0 +1.10 say 'm'+space(56)+'m'
@ow 0 +1.10 say 'm'+line-len+line-len+'m'
@ow 0 +1.10 say 'm'+space(56)+'m'
@ow 0 +1.10 say 'D'+line-len+line-len+'H'
@3.15 say '系统：      命令及函数标准化转换系统'
@ow 0 +1.15 say '作者：      梁 波'
@ow 0 +1.15 say '电子工业部洛阳轴承研究所军品部'
@ow 0 +1.15 say 'copyright (C) 1992.12'
@ow 0. col 0 +4 say 'Version 2.00 1992.1993'
@2.20 say "输入欲完整的文件名" get filename
@ow 0 +1.20 say "你的程序是大写还是小写字符"
@ow 0 +1.20 say "编写的大写 [1] / 小写 [2]" get
charater
read
if len(trim(filename))=0
select 1
use
select 2
use
return
endif
if .not. file('&filename')
* 如果文件不存在,重新输入.
loop
endif
if trim(charater)='1'.or. trim(charater)='2'
if trim(charater)='1'
up-down='upper'
else
up-down='lower'
endif
exit
else
? chr(7)      && 选择大小字符以查找替换.
loop
endif
enddo
start-time=time()
start1=substr(start-time,1,2)
start2=substr(start-time,4,2)
start3=substr(start-time,7,2)
filename=trim(filename)
point=at('.',filename)
bak-file=substr('&filename',1,(point-1))+'.bak'
select 2
append from &filename sdf
* 将 .PRG 文件向 .DBF 文件传递.
again1=.t.
again2=.t.
```

```
@17.12 say '转换系统正在进行完整化,完整了'
col1=col()
@17.col 0 say '00.00'+ '%. 经过了'
col2=col()
@17.col 0 say '00:00:00'
do while again1
find-code=substr(a->comm-func,1,4)
* 取命令或函数的前四位
if up-down='upper'
find-code=upper(find-code)
else
find-code=lower(find-code)
endif
whole-num=len(trim(a->comm-func))
* 读取一命令或函数.
select 2
* 进行模糊检索
locate for find-code $ whole-cod
set color to r/gb
select 1
end-time=time() && 设定当前操作时间
end-time=time()
end1=substr(end-time,1,2)
end2=substr(end-time,4,2)
end3=substr(end-time,7,2)
hour=val(end1)-val(start1)
minute=val(end2)-val(start2)
second=val(end3)-val(start3)
pass-time=str(hour,2)+':'+str(minute,2):
+':'+str(second,3)
percent=recno()/length*100
* 进行标准化操作所完成的%
@7.col1 say str(percent,5,1)
@7.col2 say pass-time
select 2
* 程序核心部分,进行命令及函数的标准化操作
do while .not. eof()
num=whole-num
aid-code=whole-cod
code=space(0)
do while again2
* 判断一命令或函数是否存在于一语句中,存在则进行标准化.
* 命令形式:' command+space(1)'
* 函数形式:' function+( '
hole-code=trim(substr(a->comm-func,1,num)):
+a->addition
if up-down='upper'
start-num=at(upper(whole-code),aid-code)
else
start-num=at(lower(whole-code),aid-code)
endif
```

- \* 取命令或函数的全称,进行检索替换,若无结果,则取命令或
- \* 函数(整长-1),再次进行循环至命令或函数的长度=4 为止。

```
do case
case start—num=0 . and. num<4
    code=code+aid—code
    replace whole——cod with code
exit
case start—num=0
    num=num-1
    loop
case start—num<>0
* 判断“命令”的前一位置是否为空格或顶头
* * 若为真,说明是命令,可进行替换。
if start—num<>1 . and. a—>addition=space(1)
    pro—fix=substr(aid—code, start—num-1, 1)
    if pro—fix<>space(1)
        find—space=substr(aid—code, (start—num+
            num), (len(aid—code)—start—num—num+1))
        space—num=at(space(1), find—space)
        length—c=start—num+num+space—num-2
        code=code+substr(aid—code, 1, length—c)
        aid—code=substr(aid—code, length—c+1, :
            (len(aid—code)—length—c))
        start—num=at(find—code, aid—code)
    if start—num<>0
        num=whole—num
        loop
    else
        code=code+aid—code
        replace whole——cod with code
        exit
    endif
endif
endif
code1=substr(aid—code, 1, (start—num-1))
if up—down='upper'
    code2=upper(trim(a—>comm—func))
else
    code2=lower(trim(a—>comm—func))
endif
code=code+code1+code2
aid—code=substr(aid—code, (start—num+num), :
    (len(aid—code)—start—num—num+1))
if a—>addition=space(1)
    code=code+aid—code
else
    start—num=at(find—code, aid—code)
    if start—num= 0
        * 判断函数是否在于一语句中重复出现。
        * * 若为真,则进行重复检索替换。
        code=code+aid—code
```

```
else
    num=whole—num
    loop
endif
endif
replace whole——cod with code
exit
endcase
enddo again2
select 2
continue
enddo
select 1
if .not. eof()
    * 判断是否对所有的命令或函数均完成了标准化
    skip
    loop
else
    exit
endif
enddo again1
if file('& bak—file')
    delete file & bak—file
endif
* 将原 .PRG 文件转化为 .BAK 文件。
copy file '& filename' to '& bak—file'
delete file & filename
select 2
set alternate to temp.txt
set color to n/gb
go top
do while .not. eof()
    set alternate on
    ? trim(whole——cod) && 消除尾部空格,逐行输出。
    set alternate off
    skip
enddo
set alternate to
set color to r/gb
delete all
pack
use
* 生成新的 .PRG 文件。
copy file temp.txt to & filename
select 1
use
select 2
use
enddo .y. && 返回程序的输入状态。
cancel && 程序结
```



# 一个通用的 音乐演奏程序

戴振喜

笔者曾亲身体会到，作为计算机软件人员，在调试程序时，由于精神高度集中，最容易感到疲劳。在适当的时候能听听音乐，轻松一下脑子，常常能使调试效果事半功倍。但要带一个录放机进机房，往往有多种不便。于是，便自行设计了一个音乐演奏程序，随时可以在计算机上演奏。

由计算机唱歌，这已不是新的技术。类似的程序，人们已曾见过。但多是一个程序一首歌，歌谱嵌入在程序当中，需要换一首歌，则需要改动程序。而且同一首曲子，在不同档次的机器上演奏效果相差很远，其失真程度甚至达到不能忍受的地步。本文要介绍的是一个用 C 语言和汇编语言设计的通用演奏程序。其通用性在于：用适用于各种歌曲。它允许歌谱数据独立于程序而存在，歌谱数据文件名作一个参数，演唱不同的歌曲时，只需改变这个参数，而不需改变程序。并且可以同时输入多个参数，即连续演唱多首歌曲。其通用性还在于：它适应多种档次的微机。由于设置了速度开关，只要给出不同的速度参数，本软件在 8088、80286、80386 微机上都可以演奏自如。也由于有速度开关，本软件不能在同一台机器上演奏出不同快慢节奏的曲子。本程序不仅是软件人员调试程序时可以使用，就是在其它场合其他人员欣赏计算机的演奏也是别有情趣。下面介绍具体的实现原理和使用方法，最后给出全部源程序。

## 一、音符的固有频率

歌谱由音符组成，每个音符都对应一个频率，以不同的频率去振动喇叭，就可以产生不同的音调。奏一首曲子或唱一支歌应先定调子。定调就是确定一种调中所有音符的固有频率。表 1 给出了 12 种调子的所有音符与频率的关系。

在表 1 中，每种调的音符“1”的低、中、高音的频率分别为调符所在行的三个数（如 C 调 1—262/秒，1—524/秒，1—1048/秒）。别的音符除“3”与“4”和“7”与“1”之间是半音关系，顺序往下移一格外，其它相邻两音符之间都是全音关系，顺序往下移二格。这样，就可以确定 12 种调的所有音符的频率。

表 1 音符固有频率表

音符 频率 调符	!	1	i
C	262	524	1048
C'	277	554	1108
D	294	588	1176
bE	311	622	1244
E	330	660	1320
F	349	698	1396
F'	370	740	1480
G	392	784	1568
bA	415	830	1660
A	440	880	1760
bB	446	932	1864
B	494	988	1976

## 二、曲谱的机内表示

由于曲谱中的许多符号，在 ASCII 码中是没有的，计算机则无法表示，所以需要曲谱转换为机内表示，即机内谱。这里，采用较接近于简谱的数字和符号表示曲谱中的音符和节拍。由音符的机内码和节拍的机内码合起来就构成了机内谱。

### 1、音符的机内码

在机内码中，分别用“<”、“=”和“>”表示音符的低、中、高音，具体见表 2。

表 2 音符机内码表

低		中		高	
1	1<	1	1=	i	1>
2	2<	2	2=	2	2>
3	3<	3	3=	3	3>
4	4<	4	4=	4	4>
5	5<	5	5=	5	5>
6	6<	6	6=	6	6>
7	7<	7	7=	7	7>

### 2、节拍的机内码

简谱的基本音符常用的有：全音符，二分音符，四分音

符,八分符,十六分音符和三十二分音符等6种。从概念上来说,基本音符还应有六十四分音符和一百二十八分音符,如5是六十四分音符,5是一百二十八分音符。只不过在简谱歌曲中,这两种基本音符几乎不用。

以上8种基本音符都对应着各自的节拍,为使其机内码更为简明,节拍的机内表示全部采用整数。表3表明了音符、节拍和机内码的关系。由表3可推算出其它节拍的机内码。例如“5”为3/4拍,机内码为24;又如“5”为3/2拍,机内码为48。

表3 节拍机内码表

名称	节拍数	机内码	例子
全音符	4	128	5 - - -
二分音符	2	64	5 -
四分音符	1	32	5
八分音符	1/2	16	<u>5</u>
.....	.....	.....	.....

### 3、曲谱机内表示的例子

以歌曲《信天游》的第一句为例:

曲谱: 1 6 5 0 | 1 6 1 2 0 |

机曲谱: 1=32, 6<16, 5<16, 5<32, 0=32, 1=32, 6<16, 1=16, 2=32, 0=32

由例可知,曲谱中每个音符(或叫唱名)对应一项机内谱,如“1”对应“1=32”,“2”对应“2=32”等等。

### 4、休止符的处理

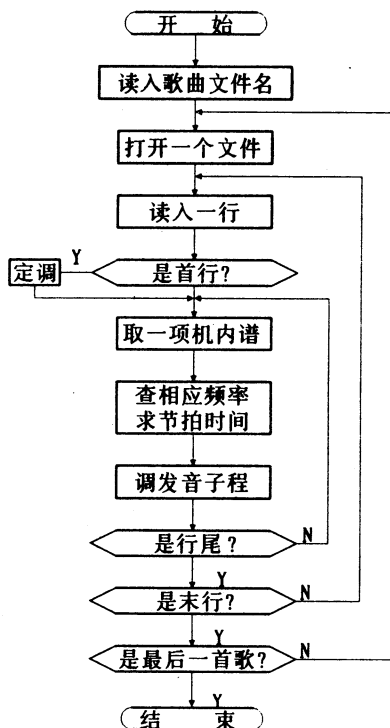
在以上曲谱的例子中,有两个休止符“0”。本程序约定:休止符“0”的机内码为“0=”,其节拍的机内码取决于其所占的节拍数,这与其它音符相同,所以例中休止符的机内谱为:0=32。但表1中不包含“0”的频率,这在程序中的如何处理呢,因为休止符是不发音的,只要振动频率小于20次/秒,就会听不到,所以在程序设计时,将“0”所对应的频率设为19。这就实现了休止的时间由其节拍数决定。

### 三、程序设计

有音符的频率表和机内谱,程序设计就比较简单了。本程序由两部分组成,一部分是由C语言(优化C86)写成,具体完成对机内谱的读入和识谱功能;另一部分由汇编语言写成,它以子程序的形式被识谱程序调用。该子程序只负责完成一个音符的发音。实现流程见图1。

本程序在长城0520CH、长城286B、浪潮0530、Tdw386等微机上都运行过。

### 四、程序的使用



### 1、输入机内谱

按照机内谱的表示格式,可以很容易地把曲谱译成机内谱,然后输入机内(用任一常用编辑软件都可),并给定文件名,存入盘上。

输入格式也很简单,只要遵循两点。其一:机内谱按顺序输入,每两项机内谱用逗号隔开;其二:调符(如1=C)放在第一行的行首。可单独占一行,也可在这一行后面紧跟机内谱,但要用逗号分开。

### 2、启动格式

执行程序的名字为:sing.exe

启动格式为:sing {filename}

其中:filename是歌名,即机内谱的数据文件名,filename可同时有多个。表示空格。

n是速度参数,控制节奏的快慢,速度可有19个档次,n取1~19中的整数,即0<n<20。n越小,速度越快,反之越慢。n可缺省,缺省时为1。

### 五、源程序和几首歌曲的机内谱

这里附上本软件的全部源程序和三首歌曲的机内谱。源程序的编译连接是在大模式下进行。三首歌曲是:《两只老虎》、《信天游》和《九九艳阳天》。

编者:由于篇幅所限,本文有关的程序将放在本期程序盘Z9子目录中。

前些时候,发现北京大学新技术公司与香港金山公司联合研制的 Super-CCDOS V6.0F 汉字系统中提供的六笔声形汉字输入法非常不错。它重码率低,词组丰富,可以与五笔字形媲美,而且它简单易学,不用死背字根,一般人学半小时就能掌握。

SPDOS V6.0F (汉

卡) 汉字系统虽然

用的单位不

多,但

它的前身

SPDOS V5.1 纯软

件汉字系统及 WPS 文字处

理系统,以其优越的性能、良好的

界面和较强的适应性,深受广大计算机用

户的欢迎。

笔者发现,SPDOS V6.0F 中提供的六笔声形汉字输入法程序,只要稍作改动就能很可靠地移植到 SPDOS V5.1 (或 SPDOS V5.0) 纯软件汉字系统中去。笔者用从 SPDOS V6.0F 中移植得到的六笔声形输入法程序在 SPDOS V5.1 版中使用已有三个月,从未发现过任何不正常现象。

移植步骤如下:

1、初始准备。

把 SPDOS V6.0F 中的六笔声形输入法程序 (LBSX.COM, LBSX.OVL) 复制一份,用 DEBUG 程序把 LBSX.COM 读入内存。

2、修改版本号。

因 SPDOS 汉字系统中的汉字输入法程序在装入前都要检查当前的 SPDOS 版本是否与之一致,否则不能装入。LBSX.COM 把版本号比较代码存放在 CS:88D0 处:

—D 88D0

20A5:88D0 30 36 53 75..... 06Su.....

06 即为版本号代码。

.....

用 A 命令把前两字节即 30H、36H 改成 31H、35H (如果是 SPDOS V5.0 版则改成 30H、35H)。

3、修改输入法号。

因 SPDOS V6.0F 汉字系统中,输入法程序在装入前均要用 INT 16H. AH=2AH 中断服务功能 (其作用为返回输入法入口地址及输入法号。)测试当前环境,分析本输

入法是否已装入,如果已装入则退出安装;否则接着检测输入法号是否已经被分配完,如果没有程序就自动就前分

配输入法号,如果已分配完,就提示“输入法已装满!”程序自动退出安装。因 SPDOS V5.1 环

境与 SPDOS V6.0F 环境有所不同,无

法通过“分析输入法号分配情况

及自动赋予输入法号”这

一程序段。所以,笔

者人为地赋值

输入

法号为

07H (当然也可以

赋为别的号,只要不影响

其它程序运行就行),同时跳开

“分析输入法号分配情况及自动赋予输入

法号”这一程序段。这一程序段较长,在此只讲与

移植有关的部份。改动从 CS:8868 开始:

—U 8868

20A5:8868 7505 JNZ 886F

.....

886F 81EE0007 SUB SI,0700

8873 1E PUSH DS

8874 07 POP ES

8875 8BC6 MOV AX,SI

.....

—A 8868

20A5:8868 JMP 886F ;跳过测试比较。

—A 886F

20A5:886F MOV AX,2307 ;赋输入法号。

8872 NOP ;空指令是为保证程序字节数的一致性。

—A 8875

20A5:8875 NOP ;同上。

8876 NOP ;同上。

到此,程序修改完毕,用 W 命令写盘退出 DEBUG 程序,移植成功。

以后,可以在 SPDOS V5.1 (或 SPDOS V5.0) 纯软件汉字系统上使用了。通过按 ALT 键加功能键 F8 即能选择到所移植得到的六笔声形输入法。用上述方法只要稍作改动参数也可以把六笔声形汉字输入法移植到 SPDOS V5.0 以前所有版本和四通公司研制的 CWS 文字系统中去。

## 给 SPDOS V5.1 增加六笔声形

### 输入法

· 陈丁君 ·

IBMPC 系列微机在加电启动时, 存放于 ROM 中的基本输入/输出系统 (BIOS) 首先进行内部的诊断测试、硬件的配置分析以及基本输入/输出设备的初始化等工作。同时, BIOS 把这些操作的结果数据存放在 RAM 内存区 40:00—40:FF 中, 这些就是 BIOS 参数。这些参数作为识别系统工作状态及环境的重要信息将在以后的系统工作中被系统引导程序、系统诊断程序以及用户应用程序所利用。一般情况下这些参数是由 BIOS 管理的, 但在实际的应用中, 根据一些特殊的需要, 我们直接对这些数据进行控制, 可收到一些非常好的效果。下面就这方面谈几则应用。

1. 在 40:08—40:0F 处的 4 个字分别存放的是 1—4 号并行口的口地址。它们一般是作为主机与打印机相联的基地址。将这些打印机基地址相互交换, 可达到打印机操作之间的切换, 这样我们在不同打印机上操作时可避免对不熟悉的打印机控制语句的修改。下面就是将第一和第二个打印机操作交换的汇编程序 (本文的汇编程序都按照在 FoxBASE 下使用标准书写的, 若使用于 DOS 下只需将 RETF 换成 MOV AH, 4C 和 INT 21H 即可。):

```
POP DS
MOV AX, 40: 本文所有数据均为 16 进制
MOV DS, AX
MOV DX, [08]
MOV AX, [08]+2
MOV [08], AX
MOV [08]+2, DX
PUSH DS
RETF
```

另外, 打印机病毒对这些基地址的破坏, 会致使正常的打印机输出操作出错。例如: 打印病毒发作时会将第一个打印机口基地址置 00, 在必要时查看这些口地址可帮助我们诊断打印机病毒。(DEBUG 中的 D 命令可很方便的查看)

2. 在 40:10 处的一个字存放的是系统设备配置信息。具体表示方式如下:

B0	B1	B2	B3	B4	B5	B6	B7
有无驱动器	有无协处理器	主机板上存储容量	显示器配置	驱动器台数			
B8	B9	B10	B11	B12	B13	B14	B15
未用	RS-232 接口数	有无游戏接口	未用	打印机台数			

查看这些数据, 不仅可了解系统设备的数目及类型, 若对它相关位进行修改, 还可实现一些特殊的应用。例如:

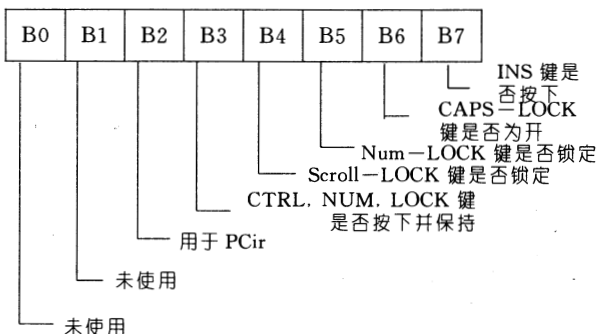
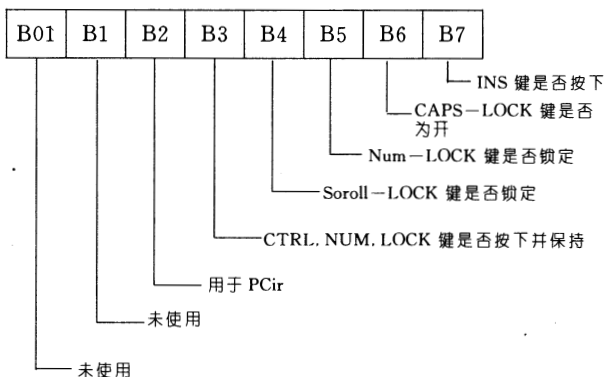
## 8300 参数应用

长特四厂 晁明中  
长特三厂 周俊辉

要在 Hercules 卡上实现 CGA 的图形仿真, 就需将表示视频适配器类型的 4,5 位置成彩色类型值。

3. 在 40:13 处的一个字存放的是系统主机板内存容量。对不同机型, 此处只能有三种值: 0100H 代表 256K, 0200H 代表 512K, 0280H 代表 640K。查看此处的值不仅可知到系统板 DIP 的配置, 还可帮助我们诊断系统病毒, 因为系统引导型病毒为了达到驻留内存的目的, 要修改此处的值, 以窃取一片内存区域存放病毒体。例如: 大麻病毒减少 2K, 打印病毒减少 1K, 新世纪病毒减少 4K 等等。

4. 在 40:17—40:18 处的两个字的每一位表示的是系统当前换挡键和双态键的状态。40:17 和 40:18 上各位表示的意义分别是:



上面各位表示的意义表肯定时值为 1, 否则为 0。

以上的键盘数据一般都是系统根据用户在键盘上的按键操作而设定的, 若我们对这些数据相应位置 1 或 0, 则可直接在程序内部实现特殊键态的转换。例如: 在汉字 FoxBASE 程

序运行中,把小键盘从光标控制功能转换到数据输入功能可使用以下汇编程序来实现:

```
PUSH DS
MOV AX, 40
MOV DS, AX
MOV AH, 20
OR [17], AH
POP DS
RETF
```

这一方法在一些老式(无 Num-LOCK 键态指示灯)键盘上使用更显方便。另外汉字与英文字母输入(这与 CapsLOCK 键态有关)之间的转换应用此法也可收到极佳的效果。

5. 在 40:1A—40:1D 处以字为单元分别是键盘缓冲区头字符指针和尾字符指针,在 40:1E—40:32 处为键盘缓冲区。键盘字符在缓冲区内要占用两个字节:第一个是 ASC II 码(若扩充码则为 ASC II 0),第二个是扫描码。若我们事先在键盘缓冲区内置放一些字符值(当然还应调整缓冲区的头、尾指针),则可获得与在键盘上操作一样的效果。下例就是可实现英汉状态之间自动转换的汇编程序(以 CTRL+F7 为例):

```
PUSH DS
MOV AX, 40
MOV DS, AX
MOV AL, [1A]
INC AL
INC AL
MOV [1C], AL
DEC AL
DEC AL
XOR AH, AH
MOV DI, AX
MOV WORD PTR [DI], 6400:6400 为 CTRL+F7 的扫描码
POP DS
RETF
```

一些大型的应用软件、游戏程序的使用演示就采用了直接控制键盘缓冲区的方法。另外,把头字符指针值置成尾字符指针值即可清除键盘缓冲区内字符。

6. 在 40:60 处的 2 个字节存放的是光标起始线及终止线值,它决定了当前显示光标的形状。当我们在一些图形及色彩较凌乱的屏幕上操作时(例如 Fox-BASE 下变量的增强显示输入时),光标的辨别较费眼,我们就可修改此处的值以改变光标形状,同时视觉上给人一种新颖感。下面的汇编程序就将光标设置成方块形状:

```
PUSH DS
MOV AX, 40
```

```
MOV DS, AX
MOV [60], 6:6 可根据需要设置
POP DS
RETF
```

7. 在 40:6C 处 1 个字节存放的是 DOS 启动时要求用户输入时间的计数值,并由定时器中断每秒修改 18.2 次。故在一些实时应用程序中用它来作实时控制,简单的如条件激发的判断程序和定时延时程序等。还由于它特有的变化率,常常利用它来生成随机数。

8. 大家知道,要热启动系统需在键盘上按 Ctrl+Alt+Del 三键,但我们在程序内部就可实现这三键热启动的同样功能,这就要利用到 40:72 处的系统重置状态字:

```
MOV AX, 40
MOV DS, AX
MOV WORD PTR [72], 1234:1234 为系统热启动标志
JMP FFFF:0000
```

虽然 DOS 管理下的 INT19H 可较快系统重启动,但其仅仅是读出软盘或硬盘中的系统引导程序到 0:7C00H 处,后将控制权交给引导程序,并未进行内存或外设的检测及初始化等系统自检工作。显然,倘若内存染有病毒或驻留有其它程序,采用 INT19H 快速重启动,并不能清除它们。

9. 在 50:00 处的一个字节为屏幕拷贝状态值,其实它属于 DOS 的内部参数。它可有三种值分别表示屏幕拷贝的不同状态:00 表完成,01 表正在进行,FF 表出错。于是,我们可控制它在一些重要的信息管理过程中拒绝屏幕拷贝:

```
PUSH DS
MOV AX, 50
MOV DS, AX
MOV [00], 1:1 若 0 则允许拷贝
POP DS
RETF
```

这样,我们在不修改键盘管理程序情况下,就轻松地屏蔽了 Print 或 Shift+Print 的屏幕拷贝功能,而并不需去修改 INT5 屏幕拷贝中断。

上面谈及的 BIOS 参数只是整个 BIOS 参数中很小的一部份,实际上 BIOS 参数还有许多其它方面的实际应用。由于 dBASE III 有直接在内存单元上控制数据的 POKE 命令(虽然 POKE 命令用不到汇编程序,但实现的思想一样)以及 FoxBASE 中有汇编程序的接口(LOAD 和 CALL 的组合使用),故 BIOS 在这方面的应用效果特别明显,这也是本文的汇编程序都按此接口标准书写的原因。直接对 BIOS 参数操作的主要特点是方便、灵活,汇编程序完全可在 DEBUG 调试工具中立即实现,并且生成的程序短小、精悍。

# 王码汉字系统(WM 5.0)中几个

## 未公开的功能调用

潍坊市质检所微机室 李志刚

王码汉字系统的显示模块中,除了通常汉字系统所具有的显示中断(INT 10H)基本功能调用(AH=0~0FH)外,还增加了几个扩展调用(AH=18H~0F6H),为WMDOS增加了不少特色,其中有三个比较实用。下面分别介绍一下:

### 1. AH=18H 建立工作方式

入口参数:

AL=0 识别汉字内码方式

AL=1 不识别汉字内码方式

系统通常工作在“识别汉字内码方式”既一般汉字系统的作状态,当将工作状态转换为“不识别汉字内码方式”时,可以在屏幕上正确的显示扩展ASCII字符,我们便可以在汉字状态下用单字节制表符来制作菜单,窗口边框,表格等,并且不会因遇到半个汉字而造成显示混乱。

### 2. AH=0F4H 保存屏幕

入口参数:

CH=左上角行值

CL=左上角列值

DH=右下角行值

DL=右下角列值

CS:BX=屏幕存储区首址

BP=CS

注:该功能调用前,一定要将BP=CS,否则将造成死锁。

### 3. AH=0F5H 恢复屏幕

入口参数同上

屏幕保存功能可以将屏幕的当前显示内容保存起来,保存区域由CX、DX指定,数据保存在由CS:BX指定的内存中。在执行其它屏幕显示程序后,可由恢复屏幕功能将原内容重显,这样便省去了编制重画屏幕程序的麻烦。

将以上三个功能接合起来使用,可以方便的作出高质量的弹出式、下拉式菜单和窗口程序,设计出的程序界面可与现行西文软件媲美。如用在西文软件的汉化中,西文制表符仍可使用,无须将其转换,可保持西文软

件的原有特色。

为让大家进一步了解其用法,特编制SAMP.ASM演示程序,大家只要将其用MASM汇编、连接,并转换成COM形式的程序便可在WMDOS下运行,运行前可将屏幕上显示一些内容,以观察屏幕的保存恢复效果。源程序如下:

;SAMP.ASM(COM)

;WMDOS V5.0 屏幕保存,恢复与ASCII扩展制表符的显示  
CODE SEGMENT

ORG 100H

ASSUME CS:CODE,DS:CODE,ES:CODE

BEGN: JMP START

WIND DB 0C9H,43 DUP(0CDH),0BBH,0DH,0AH  
DB 0BAH,43 DUP(20H),0BAH,0DH,0AH  
DB 0BAH,43 DUP(20H),0BAH,0DH,0AH  
DB 0BAH,43 DUP(20H),0BAH,0DH,0AH  
DB 0BAH,43 DUP(20H),0BAH,0DH,0AH  
DB 0BAH,43 DUP(20H),0BAH,0DH,0AH  
DB 0BAH,43 DUP(20H),0BAH,0DH,0AH  
DB 0BAH,43 DUP(20H),0BAH,0DH,0AH  
DB 0C8H,43 DUP(0CDH),0BCH,0DH,0AH

COUNT DW \$ - WIND ;所画框的数据长度

BT1 DB 'WMDOS V5.0 屏幕保存、恢复 \$'

BT2 DB '与扩展制表符的显示 \$'

BT3 DB '按任意键退出..... \$'

BUF DB 4000 DUP(?) ;屏幕保存缓冲区

START: MOV AH,02

MOV BH,0

MOV DX,0 ;设置光标位置

INT 10H

MOV AX,1801H ;设置为"不识别汉字

int 10h ;内码方式"

MOV AH,0F4H ;保存屏幕

MOV CX,0 ;保存范围为(0,0)(7,46)

MOV DH,07

MOV DL,46

MOV BX,OFFSET BUF ;取缓冲区首址

PUSH CS ;BP=CS

POP BP

INT 10H

MOV SI,OFFSET WIND ;画框

MOV CX,COUNT

LOP: MOV BH,0

MOV AH,0EH

MOV AL,[SI]

INT 10H

INC SI

LOOP LOP

MOV AX,1800H;设置为"识别汉字内码方式"

INT 10H

MOV AH,02 ;写汉字

MOV BH,0

MOV DH,2

```
MOV DL, 10
INT 10H
MOV DX, OFFSET BT1
MOV AH, 9
INT 21H
MOV AH, 02
MOV BH, 0
MOV DH, 3
MOV DL, 10
INT 10H
MOV DX, OFFSET BT2
MOV AH, 9
INT 21H
MOV AH, 02
MOV BH, 0
MOV DH, 5
MOV DL, 10
```

```
INT 10H
MOV DX, OFFSET BT3
MOV AH, 9
INT 21H
MOV AH, 8
INT 21H
MOV AH, 0F5H
MOV CX, 0
MOV DH, 07
MOV DL, 46
MOV BX, OFFSET BUF
INT 10H
INT 20H
CODE    END
        END BEGN
```

465

## 也谈 *FOXBASE+* 反编译技巧

江西省景德镇陶瓷学校 聂文广

《电脑》杂志 93 年第一期刊出林荣庆先生的《谈 *FOXBASE+* 反编译技巧》一文，提出了一个简便的 *FOXBASE+* 的反编译方法，即打开进搜集命令行显示，然而几乎所有的用 *FOXBASE* 编制的软件，在运行过程中都要关闭命令行显示，即使用 `SET ECHO OFF`，林先生文中提出的解决办法是使用 `SET STEP ON` 命令，虽然可行，但存在几点不足，一：由于每执行一条命令都要暂停一下，使得反编译的速度太慢。二：由于每执行一条程序命令都附加一条提示命令，使得反编译出的文本文件太长，不便整理。三：如果源程序中设置了 `SET STEP OFF` 命令，此方法也就失效了。

本人在使用 *FOXBASE* 编程和反编译过程中，有一点小经验，在此供大家参考。

*FOXPCOMP* 有两种编译形式，一种是加密编译，一种是不加密编，我们通常所说的反编译都是在不加密的程序进行的，对于加密的程序必须先解密再进行反编译，对于如何解密，《新浪潮》和《计算机世界月刊》杂志都刊过文章作了具体的介绍，在此不再重复。

反编译的关键就是如何使程序中的命令 `SET ECHO OFF` 变成 `SET ECHO ON`，下面提出两种简单的办法：

一、通过编译得出命令 `SET ECHO OFF` 的 *FOXPCOMP* 编译代码为 04473351，而 `SET ECHO ON` 的编译代码为 04473352，用 *PCTOOLS* 或 *DEBUG* 程序对源程序进行查找替换，把所有的 04473351 都替换成 04473352，然后在 *FOXBASE* 的圆点状态下键入：

```
SET ALTE TO <文本文件名>
```

```
SET ALTE ON
SET TALK OFF
SET ECHO ON
DO <被反编译的文件名>
```

这样反编译便通知无阻了。

二、*FOXBASE* 本身提供了一条 `ON ESCA` 命令，其功能为，当按下 `ESC` 键时，便中断当前运行的程序，转入执行 `ON ESCA` 后的命令或程序，使用 `ON ESCA` 进行反编译的方法如下：

在 *FOXBASE* 圆点状态下键入：

```
SET ESCA ON
ON ESCA SET ECHO ON
SET ALTE TO <文本文件名>
SET ALTE ON
SET TALK OFF
SET ECHO ON
DO <被反编译的文件名>
```

在反编译时一旦碰到程序关闭命令行显示时，按下 `ESC` 键即可重新打开命令行显示。

以上两种方法比较而言前一种方法更为可靠，对于第二种方法如果源程序中设置了 `SET ESCA OFF`，此方法也就失效了。

466

# 小辞典

**adapting unit interface (AUI)** 连接部件接口  
**adaptive differential pulse code modulation (ADPCM)**  
 自适应差分脉冲编码调制  
**adaptive equalization** 自适应均衡技术  
**adaptive equalizer** 自适应均衡器  
**adaptive packet assembly** 自适应信息包软件  
**adaptive routing strategy** 自适应路径选择策略  
**ADB (Auto-dBASE)** 自动 dBASE 数据库  
**ADC (Analog to Digital Converter)** 模拟数字转换  
**ADCCP (Advanced Data Communication Control Procedure)** 先进数据通信控制过程  
**AD/DA converter** 模数数模转换器  
**additional license** 增加特许  
**additional memory** 附加存储器  
**addition transition network (ATN)** 增强变迁网络  
**additive process** 加成形成工艺  
**add-on card** 附加插件板,附加卡,添加卡  
**addressing capability** 寻址能力  
**addressing capacity** 编址能力  
**addressing level** 定址级  
**addressing mode** 编址方式  
**address maintenance code (AM)** 地址修改码  
**address mask** 地址屏  
**address-move to** 地址移动  
**address resolution protocol (ARP)** 地址转换协议, 地址分解协议  
**address stack (ASTK)** 地址堆栈  
**address strobe (AS)** 地址选通脉冲  
**ADDs (Advanced Data Dictionary System)** 先进数据词典系统  
**ADELA (Architecture Design and Evaluation Language)**  
 体系结构设计和评价语言  
**ADF (Adapter Description File)** 适配器描述文件  
**adhesive** 粘合剂  
**ad hoc** 特设,特定的;尤其,关于这  
**ADI (Autodesk Device Interface)** 自动桌设备接口  
**adjective phrase** 形容词性短语  
**ADLC (Advanced Data Link Controller)** 先进数据链路控制器  
**administrative tool** 管理工具  
**ADP (Architectural Design Program)** 建筑设计软件包

**ADP (Automatic Data Processing)** 自动数据处理  
**ADPCM (Adaptive Differential Pulse Code Modulation)**  
 自适应差分脉冲编码调制  
**ADS (AutoCAD Development System)** AutoCAD 开发系统  
**ADT (Abstract Data Types)** 抽象数据类型  
**advanced automated system (AAS)** 高级自动化系统  
**advanced basic input output system (ABIOS)** 高级基本输入输出系统  
**advanced broadcast television system (ABVS)** 先进广播电视服务  
**advanced computing element (ACE)** 先进计算单元  
**advanced computing environment (ACE)** 先进计算环境  
**advanced data communication control procedure (ADCCP)** 先进数据通信控制过程  
**advanced data dictionary system (ADDs)** 先进数据词典系统  
**advanced data link controller (ADLC)** 先进数据链路控制器  
**advanced factory manufacturing system (AFMS)** 先进工厂制造系统  
**advanced image management system (AIMS)** 高级图象管理系统  
**advanced intelligent network (AIN)** 先进智能网络  
**advanced interactive executive (AIX)** 先进交互执行  
**advanced interactive mapping display** 高级交互图象显示  
**advanced mobile phone system (AMPS)** 先进移动电话系统  
**advanced modeling extend (AME)** 先进建模扩展  
**advanced modeling extension (AME)** 先进建模扩充, 高级建模[实体模型]扩展  
**advanced networked Chinese language environment (ANCLE)** 基于网络的高级中文语言环境  
**advanced network system architecture (ANSA)** 先进网络系统体系结构  
**advanced network test center (ANTC)** 先进网络测试中心  
**advanced program to program communication (APPC)**  
 先进的程序到程序通信  
**advanced program to program interface (APPI)** 先进程序至程序界面  
**advanced radio data information service (ARDIS)** 高级无线电数据信息服务



大

约 60 年前,位于日本正中部的名古屋市热田街,一位名叫安井正义的青年从事缝绖机修理工作。在当时,一说到缝绖机,全都是美国胜家公司的产品,日本还不具备生产同类产品技术能力,仅是对出故障的缝绖机进行修理而已。安井正义的父亲也是位缝绖机修理工,安井继承了父亲的职业,从父亲那里学到了许多有关缝绖机知识的同时,逐渐对缝绖机产生了浓厚的兴趣,并诱发了他要试制缝绖机的想法。安井知道缝绖机国产化的难度,但他想即使少量也罢,也要用自己的能力生产出廉价的缝绖机,供给更多的人使用。在这种意念鼓舞下,他开始了专心致志的学习和研究。

此后,这一愿望终于得到了实现。1928 年,他研制出了用于缝制帽檐的缝绖机。是兄弟公司工业缝绖机首号产品的问世。接着他又开始着手缝绖机销售的准备工作,考虑刚研制成功的缝绖机使用哪种商标的问题。商标将意味着缝绖机的容貌,应使用最为合适的东西加以装饰。经过深思熟虑之后他将商标定为“兄弟”二字。因为缝绖机的开发并不是安井一个人的成果,其中也包含了他的弟弟妹妹们,为了他能如愿以偿所付出的辛劳。这就是大家所知一直沿用至今的兄弟之名的由来。

在此之后,诚实的安井生产出的缝绖机受到很高的评价,很快,家用缝绖机的开发于 1932 年成功,安井国产化的理想逐步地得到了实现。

1934 年,安井正义又肩负众人之托,成立了专门生产缝绖机的股份有限公司。为了发扬当时的创业精神,他留下以下三句话语:第一,“将进口产业变成出口产业”;第二,“要给想工作的人提供工作”。第三,“创造愉快的工作环境”。

公司职工也能充分理解这些方针。在牢固的信赖关系中,全体职工体味着人生的价值,努力生产出高质量的缝绖机。随着这些产品的问世,兄弟牌缝绖机的声誉不断提高,受到了用户的信赖。在这种情况下,公司又分别于 1947 年和 1948 年开始向外出口家用及工业用缝绖机,并逐步在外赢得了信誉。就这样,兄弟公司由从前很小的作坊式工厂发展成今天跻身于世界的大企业。

此后,兄弟公司将缝绖机作为稳定公司的支柱产品,不断向其它领域的产品进行挑战。1954 年,开始了编织机的生产。同年,利用制作缝绖机所掌握的电机技术开始了洗衣机的生产。1961 年,打字机的开发又获得成功。1966 年,开始生产计算机(首号电子产品)。1971 年,开始生产世界首创的高速点式打印机;缝绖机的生产累计达 1000 万台。1980 年开始生产办公用电子打字机;打字机的生产累计达 1000 万台。1984 年,公司成立 50 周年;开始生产日文文字处理机;作为正式指定厂家向洛山矶奥运会提供打字机。1985 年,开始在 BIUK (英国)生产电子打字机;开始生产程控攻丝设备。1987 年,开始在 BIUS (美国)生产电子打字机,在 BIUK (英国)生产微波炉、点式打印机;开始传真机的生产,涉足通信设备领域。1989 年,开始生产彩色复印机,涉足图像设备领域;成立 BITM (马来西亚)、BIIR (爱尔兰)海外公司。1991 年打字机产量累计达 2000 万台(除去海外生产)。现在,这些产品与缝绖机并驾齐驱成为兄弟公司的拳头产品。

# 兄弟株式会社

467

brotherbrotherbrotherbrotherbrotherbrotherbrotherbrotherbrotherbrotherbrotherbrotherbrotherbrother

# 热烈祝贺

## 中国软件行业协会全国第二届会员代表大会召开

〔本刊讯〕 中国软件行业协会全国第二届会员代表大会于 11 月 28 日至 30 日在济南召开，参加会议的有来自全国各地、各行各业的百多名代表。电子部计算机司司长、中国软件行业协会第一届会长杨天行同志主持大会并致开幕词。胡貽志同志代表第一届理事会作工作报告、许孔时同志作了修改章程的报告。在分组会上，代表们对工作报告及章程展开了热烈的讨论。代表们认为在我国软件产业发展过程中，软件行业协会起了相当重要的作用，参与制定了我国软件发展规划、产业发展战略和《计算机软件保护条例》，在推动我国软件市场建设、培训软件人材等方面做了许多工作。

大会选举产生了协会第二届理事、常务理事、正副理事长及秘书长。大会还筹建了组织工作；市场与交流；政策与法规；规划与宣传；教育、培训、考试指导及信息与发展六个专业工作委员会。大会还举行了专题报告会。中软总公司黄晓明总经理作《开发软件产品，开拓软件市场》专题报告。他介绍了发达国家软件产业的规模及规范生产，指出我国的差距及努力的方向。本刊吴军主编在会上作的《软件保护对广东省经济、技术发展的影响及对策》专题报告给与会者留下了深刻印象。

（本刊记者 詹前）

### 中国软件行业协会工作报告(摘录)

1984 年 9 月 6 日至 8 日在北京召开了中国软件行业协会成立大会，当时的电子部部长江泽民到会作重要讲话并被大会推选为协会的名誉会长。成立大会以无记名投票的方式选出 55 人组成协会的理事会，并选出杨天行为协会会长。9 年来，协会作了大量工作，对我国软件发展得到很大的促进作用。

我国软件行业已拥有一支素质比较高的软件产业队伍。据粗略统计，从事软件开发的专业技术人员约 8 万人，加上应用部门的软件技术人员，总人数约有四十万人。其中 70% 以上是青年技术人员。专门或主要从事软件开发的公司已有近千家，其中约有百家中外合资的软件企业。我国的软件产品年市场销售额约 20 亿元。软件出口值约为每年三千万美元的规模，其中软件产品的出口仍偏少，

以软件劳务输出和来件加工制作为主，出口对象以日本、新加坡和美国为主。

我国软件产业发展中的一个极重要的标志就是软件保护条例的制订、颁布和实施。我国的软件产品实行登记制度以来已正式受理软件登记申请五百多件，以软件登记



中国软件行业协会第二届理事长杨天行

为基础，根据软件保护条例，我国软件和知识产权的侵权行为已开始受到法律上的制约。今年 2 月 23 日是个值得记入我国软件发展史册的日子，这一天，北京海淀人民法

院公开审理了我国首例计算机软件侵权纠纷案,被告中科远望公司侵犯了原告北京微宏电脑软件研究所开发的 UNFOX 2.1 反编译博士 V2.1 软件的著作权,得到了立即停止侵权的处理,法院判被告公开道歉并赔偿原告 4.6 万元,并处以一万元罚款,此判决已发生了实际法律效力。同时,各法院还正受理多起软件侵权案,涉案企业知名度更高,赔偿也将更大,这些都说明我国软件产业已初步走了法制化的轨道。

总之,我国软件产业已初具规模,正在逐步走上开发工程化,成果商品化,应用社会化,经营企业化,管理法制化的轨道,并且创造条件与国际的大环境接轨。在形成我完整的软件研究、开发、生产、流通、维护、管理和服务体系。协助和组织学校、研究所进行软件技术研究、模型研究和原型开发,由软件企业实施产品开发、商品化、销售服务,分工合作。

5. 保护软件知识产权,宣传和维护我国的“著作权法”、“专利法”和“软件保护条例”。要使大家认识到,如果知识产权得不到保护,软件产品就失去了市场,权利人的投资不能回收,也就不会有软件产业等。

6. 加强与国外软件行业的联系,组织经验和技术交流,进行技术合作。增加与在华外商的交流,尊重他们合法经商的权利,帮助他们解决困难,同时维护我国用户的正当权益和保证我国有关政策的贯彻执行。

7. 加强协会组织建设,如筹备成立组织委员会、编辑出版委员会、软件保护委员会、对外联络委员会等。做好发展会员工作。

愿意履行会员义务,遵守行规行约,按期交纳会费,均可申请入会。经协会理事会审查同意,交纳入会会费后,即可成为正式会员。会员证由本协会统一印制。

### 第二十一条 会员的权利

1. 享有选举权和被选举权;
2. 享有建议权、批评权、监督权;
3. 优先参加协会组织的各项活动;
4. 优惠享受协会提供的各种信息资料;
5. 单位会员有权委派会员代表人,参加会员代表大会行使相关的权力;
6. 有退会的自由。

### 第二十二条 会员的义务

1. 遵守协会章程和行规行约,执行协会决议,维护协会声誉和利益;
2. 积极参加协会组织的活动,完成协会委托的各项工作,按规定向协会提供信息资料;
3. 按期交纳会费。

**第二十三条** 协会的最高权力机构是会员代表大会,每四年召开一次,特殊情况可适当提前或延期。

中国软件行业协会秘书处地址:北京市海淀区复兴路乙 20 号 通信:北京 162 信箱  
邮编:100036  
电话:6841429-409

469

## 中国软件行业协会章程(摘要)

**第一条** 中国软件行业协会 (China Software Industry Association) 缩写成 (CSIA) (以下简称协会) 是从从事计算机软件产品研究、开发、生产、出版、销售、服务、教育和管理工作的企事业和个人自愿结合组织起来的全国性的行业组织。

**第二条** 本协会是经国家民政部注册、登记,具有全国性社团法人资格的社会团体。

**第三条** 本协宗旨是加强全国软件行业企事业或个人之间的合作、联系和交流;加速计算机应用社会化、软件开发工程化、软件产品商品化和软件经营企业化;开拓国内外软件市场,发展我国软件产业;在政府和行业组织、企业集团之间发挥桥梁、纽带作用,促进软件产业的发展。按照国家的法律、法令和政策开展本行业的各项活动,为会员服务,维护会员和本行业的合法权益。

**第十九条** 凡软件行业的企业和个人,承认协会章程,

## 布尔电脑公司在广州设立办事处

**【本刊讯】** 布尔电脑(香港)有限公司广州办事处于一九九三年十一月十六日,在广州世界贸易中心大厦八〇五室投入运作。这是布尔公司继在北京和上海设立办事处外,在中国境内设立的第三家办事处!

布尔集团是全球十大电脑集团之一,具有七十年的悠久历史,年营业额逾六十四亿美元。布尔集团在全世界拥有四万多名员工,并在全球一百多个国家设有办事处以及维修中心。

布尔电脑(香港)有限公司在广州市场主要推广大、中、小型电脑系统产品,以及自动柜员机、智能卡和高速打印系统。较多应用于金融业、政府部门、制造业、零售业以及科技通讯业,对提高上述行业的运作效率和服务质量将起促进的作用。(本刊:波仔)

470

## 93' 中美软件合作洽谈在深圳举

[本刊讯] 93' 深圳中美电脑软件合作洽谈会于今年 12 月 6 日至 7 日在深圳市举行。出席洽谈会的美方代表团由 Avalon Software、Glenco Engineering、GUPTA、Intormation、Inter-National Research Institute、PC EXpress、Systematics Information Services、Quertardeck office system、Netsoft 等几家美国公司组成，这次活动由美国商务部、深圳市科技局、深圳中美商务中心主办、广东省电脑商会协办。会议由美国驻广州总领事馆商务处主持，广州、深圳、珠海等软件开发经销单位派出代表参加。美方的目的是在中国寻找市场，代理商，合作者和与计算机用户广泛接触。



在洽谈会开幕式上，美国驻广州总领事馆商务领事班明峰先生，深圳市朱悦宁副市长、深圳市科师包恒先生等，先后讲了话。讲话中着重强调开展软件合作的意义和实行软件法律保护的重要性，软件产业才能正常发展。著名的国际性软件保护组织商业软件联盟(BSA)，在会议上散发了宣传

开幕式后，参加洽谈会的美国公司，分别作了技术讲座和演示；并各自与中方的有合作意向的公司。这次洽谈会对我省软件技术、市场和法律保护，起到促进作用。(本刊记者 吴海)

## 94' EPSON 电子元件展示会及应用

[本刊讯] 由爱普生香港有限公司主办，中国电子进出口公司华南公司和广东省计算机公司协办的 94' EPSON 电子元件展示及应用发布会于 12 月 15、16 日在广州花园酒店举行。国内用户对 EPSON 这个名字并不陌生，但熟悉的是他生产的打印机及电脑产品，其实爱普生还生产一系列高水平的电子元件。早在 80 年代初，香港大部份的电子厂商已经开始采用爱普生所生产的电子元件。为使国产电器升级换代，并打入国际市场，香港爱普生将提供世界一流的基础电子元件及技术给国内电子厂商。



这次展示会主要介绍爱普生三种最新电子元件产品：1、TM-300K 销售终端微型中英文票据打印机，该机全兼容 TM-300 系列，使用 ESC/POS 标准命令集，内存 16×15 点阵国标二级汉字库。2、SD-880 3.5/5.25 英寸标准 5.25 及 3.5 英寸软盘功能，但只有一般 5.25 英寸软盘驱动器的高度。3、TCM-A0696—中文传呼机，低功耗、能显示 122×32 点 2 行每行 7 个中文字及一些特定的图形。(詹前)

## 全国计算机软件资格和水平考试函授辅导班(第 5 期)

为配合国家人事部、电子部计算机司每年进行的全国计算机软件专业技术资格和水平考试，经北京市软件行业协会考试指导委员会主办，清华大学计算机系、北京大学计算机系等校专家、教授主讲。

招生对象：各行业从事或有志从事计算机软件工作的人员。

辅导内容及方式：初级程序员(相当技术员)、程序员(相当助工)、高级程序员(相当工程师)、系统分析员。考试大纲的要求(详细内容见简章)，发放统编教材及有关资料，通过《函授通讯》进行教学指导、布置作业、解市设立面授站。备有全套辅导音像教材，供各地集中培训使用。

学习时间：1994 年 3 月初开学，至 9 月份结业，进行模拟考试。成绩合格发钢印结业证明。

报名办法：即日起开始报名，填写报名表 1 份(简章及报名表函索即寄)。

学习费用：报名费 10 元，学费(含教材、资料、《函授通讯》、邮寄费用等)：初级程序员级 120 元，程序员级 160 元，系统分析员级 180 元。联系电话：2561144—2373，电挂：1331。联系人：张岱，邮汇：北京清华大学计算机系训学校。邮码：100084。银行汇款：北京清华园城市信用社，帐号：07252—13。收款：中国软件行业协会软件