

论坛

从一个程序的修改论优化设计..... 林月桢 (2)

电脑与法律

一封短信引起的对话——程序侵权的议论
..... 言 今 (3)

电脑应用

智能检测仪表中微型打印机的应用... 何伟等 (5)

“爬格子动物”的新生..... 刘一兵 (8)

软件介绍

新一代汉字操作系统——TTDOS ... 钟华宁 (9)

网络与通信

一种非磁盘间的文件拷贝方法 赖宏宇 (10)

高级 UNIX 连网技术讲座 第一讲 UUCP

(UNIX to UNIX COPY) (上) ... 冯家宁 (12)

IDEA

单独使用 213 打印驱动程序的方法
..... 汤 庸 (17)

DOS 高低版本的硬盘共享..... 葛增超 (17)

大学生之页

DOS 级通用光条式菜单程序设计 with 实现
..... 符云清 (18)

小辞典 (19)

中文信息处理

IBM PC / XT / AT 到 IBM AS / 400 的汉字变
换方法 李建明 (20)

DOS 下王码 5.0 与 2.13H 共存 陈晓乐 (22)

使用与维修

PC / AT 微机开关电源的故障定位法及维修实例
..... 郑钟峰 (23)

CTX-7 型显示器故障维修一例 ... 单昶贤 (24)

AR-3240 打印机使用经验..... 郑 良 (25)

激光打印机故障维修三例 杨居义 (26)

CR 3240 彩色打印机维修两例 张振隆 (26)

电脑辅助教学

通用教师业务档案管理系统的设
计 王坚城 (27)

动画演示摆线的形成 应海涛 (29)

中学天地

活用 ECHO 命令的功能..... 何管略 (30)

APPLE II 的磁盘比较程序 马 骥 (31)

扩展 MUSIC 功能 陈 剑 (32)

简讯

关于更改广东省部分城市计算机操作人员联合水
平考试时间的通知 (32)

通讯 何文生 (32)

游戏乐园

模拟城市设计 SIMCITY 玩法介绍
..... 林天臻 (33)

DIGGER 游戏的改进 彭铁忠等 (35)

《银河英雄》及其修改方法 赵战赬 (36)

游戏求解 闫益民 (39)

病毒防治

提高硬盘抗病毒能力一法 唐 红 (37)

MI.COM 在病毒检测中的应用..... 郭 江 (38)

V304 病毒的检测和清除..... 蔡宁浩 (39)

电脑用户

怎样从内存中直接得到 COMMAND.COM 的
清单 宋 捷 (41)

IBM PC BASIC 语言调用鼠标的程序设计方法
..... 陈惠生 (42)

一种给电路板设计软件包 TANGO 建立汉字库
的方法 李毅军 (44)

激光加密盘的解密方法及其备份 ... 韩文生 (45)

也谈用软盘启动 CCBios2.13H 汉字系统
..... 徐小平 (46)

真正、彻底地从软盘启动 2.13H 汉字系统
..... 金巍岗 (47)

DBASE 中的“特技”清屏演示程序
..... 郑明达 (48)

MS-DOS5.0 下使用金山 DOS5.1 中文系统
..... 朱丰池 (49)

在 2.13H 汉字系统下作饼状图 徐文兵 (51)

条件组合技术及其运用 殷秀霞 (52)

用“混沌方法”产生伪随机数 芮 勇 (55)

在 MS-DOS5.0 下运行 UCDOS 的简便方法
..... 喻顺洲 (55)

单片板与单板机

可编程控制器中的位处理器 徐巨善 (56)

名厂介绍

国营南京有线电厂 (61)

广告索引 (19)

从一个程序的修改论优化设计

广州市卫生防疫站 林月桢

一、前言

关系型数据库是国内微机应用最普遍的软件,开发于事务管理的软件已广泛应用于人事、物资和财会等领域。和 BASIC、COBOL、FORTRAN 及 PASCAL 等其它高级高语言相比,FOXBASE 和 DBASE 有更灵活、更直观的资料存取和操作简单易学等优点,但在数据计算中它的速度落后于前者 5~9 倍。一个处理大量数据的应用软件,统计部分最能体现程序员使用命令的灵活性和对应用专业的熟识程度,一个命令使用不当会增加成千上万次运算。如何减少运算,提高程序的运行速度,是关系型数据库程序员面对的首要问题。北京某大学电脑中心编写了一个“全国学生体调统计系统”,用于处理 1991 年全国学生体质健康调查的数据,系统是 FOXBASE+2.1,配置硬件:长城 0520 或兼容机,统计打印部分约 30KB。需要输出 13 类 608 个报表,每个报表纵向 5~16 栏,横向 5~17 栏。处理广州市 6384 名学生数据需 140 个机时。对原程序修改后,报表输出速度提高 4~19 倍,平均 7.7 倍。

二、方法与讨论

1、一个含有多个字段的大型数据库,每次操作可能仅在几个字段和部分记录中进行,产生一个中间数据库和“过筛”是减少读盘和运算的好方法。该数据库有 36 个字段,实际每次包括年龄只操作 2~4 个字段,将操作字段 COPY 到一个中间数据库,再按年龄过筛记录,即可统计分年龄组的各项指标。FOXBASE 以前的版本和 DBASE 可用 SET FILTER TO D5 = "&II" 实现,FOXBASE+2.1 用更快更强的筛选索引命令 INDEX ON D5 FOR D5 = "&II" TO PP,修改后报表输出速度快 2~4 倍。用 SET FIELDS TO “过滤”字段不能加快速度。

2、同一命令,表达式内容是特定变量时只操作一次,是字段名时反复几百次至几千次,应尽量减少对字段的操作。下例 D5、D28、D26 分别为数据库年龄、体重、身高字段,显然修改后的程序减少了大量的转换和计算。

原程序:

```
COUNT TO N FOR VAL (D5) = I
```

修改后:

```
II=STR (I, 1) &&将变量转为字符型
```

COUNT TO N FOR D5 = "&II" &&减少每个记录

D5 的转换

原程序:

```
SUM D28 / D26 * 1000 TO S
```

修改后:

```
SUM D28 / D26 TO S
```

S=S*1000 &&避免每个记录都作一次乘 1000 的计算

3、进行分类计算和求和时有两个方法:(1)用命令 COUNT 和 SUM,执行速度较快;(2)用叠加法分项叠加,速度慢,但每个记录读一次可完成。当要使用多个 COUNT 和 SUM 时,要反复读盘多次,此时不如用叠加法快。原程序统计视力用了 14 个 COUNT 和 SUM,统计出一个报表要 19 分 10 秒 (40MB3.5 英寸高速硬盘 ST351),改用叠加法用时 2 分 19 秒。

4、每个 SUM 或 REPLACE 都要读盘一次,条件相同时尽量在 254 个字符内写完一个命令。指针在字段间的移动比反复多次读盘要快。

5、在使用 IF...ELSE...ENDIF 时,利用嵌套并把条件为真机会较大者排在前面,减少无必要的比较。

6、执行 DO CASE 命令时,机器依次判断 CASE 的表达式来查找第一个出现逻辑值为真的表达式,并执行至下一个 CASE 才跳出 ENDCASE,应把机会最多的放在前面的 CASE 下。对应用专业不熟悉的程序员容易疏忽这一点。原程序统计肥胖时用了六个 CASE 累加 (1) 营养不良极, (2) 营养不良重, (3) 营养不良中, (4) 低体重, (5) 肥胖和 (6) 超重,营养正常人数由总人数减去六项求得。普通人群构成比例 (1) 0.04%, (2) 0.2%, (3) 7%, (4) 46%, (5) 0.4% (6) 3%, 正常 53%, 根据这些数据估算平均每个记录要作 5.1 次 CASE 判断,将顺序重排: 营养正常 → (4) → (3) → (6) → (5) → (2), 营养不良极由总人数减这六项求得。修改后平均每个记录只作 1.6 次比较,理论上提高速度 3.1 倍,实际测试提高 2.8 倍。原程序有多处类似缺陷,且一个 DO CASE..... ENDCASE 能完成的情况,却用了多个。

7、原程序为取一指标中文名称在一个 DO CASE...ENDCASE 中用了 20 多个 CASE,改用函数 SUBSTR(), 减少查找时间,精简了程序。

一封短信引起的对话

——程序侵权的议论

· 言今 ·

甲：你读了《电脑》杂志 93 年第 2 期吗？那上面刊登了读者沈凌的一封信。信中指出，登在该刊 92 年第 6 期上署名陈红新的《CEC-I 模拟音响》一文（以下简称“陈文”），是对他发表在 90 年 10 月 25 日《学生计算机世界》（周刊）上《CEC-I 音响几则》（以下简称沈文）的抄袭，并认为是侵犯了他的著作权。你对此有什么看法？

乙：我也读到了沈凌的信，并且找到了他发表在《学生计算机世界》上的文章。那是一篇短文，实际上是利用 CEC-I MUSIC 语句编成的七个非常简短的音响程序，即警车、救护车、流水、马蹄声、发动机、青蛙叫、蚰蚰声等七种模拟声音。而陈文也是利用 CEC-I 编成七个音响程序。其中，警车、发动机、水流声、蚰蚰声四个与沈文完全相同；“救护车”则是从“警车”程序经

修改参数而成，它比沈文原有的“救护车”更精练些。另外的两个是报时音乐、无线电报声，如果没有其它争议，则应承认这两个程序是陈某自己的创作。但就我看来，陈文不论是作为一篇文章，还是文章中提供的程序都违反了我国著作权法，是一种抄袭的行为。

甲：你能说得详细一些吗？可不可以认为陈某只是引用了沈文的某些部分而算不上是侵权呢？

乙：按我国著作权法规定，对他人已发表的作品，是可以合理使用的，但这种使用有严格规定，不但要注明引用内容的出处、作者，而且还要具备三个条件，其中包括：引用的目的仅限于介绍、评论某一作品或者说明某一问题；所引用

8、条件语句越长，判断花时间越多。原程序有多处这样的条件句子 $J=6.OR.J=7.OR.J=8.OR.J=9$
 $.OR.J=10.OR.J=11.OR.J=12.OR.J=17.OR.$
 $J=18.OR.J=19.OR.J=20$ ，显然不如 $J>5.AND$
 $J<13.OR.J>16.AND.J<21$ 。

9、原程序中还存在不少重复计算，如视力正常人数由总人数与视力不良人数相减不必 COUNT 一次，远视眼数 $COUNT\ FOR\ D12=2$ ，不必 $COUNT\ FOR\ D12=2.AND.D10<50$ 。若程序员了解一些学校卫生专业常识就不会出现这样的失误。

三、小结

以上缺陷均在原程序中出现一次或多次且对程序运行速度影响较大。关系型数据库的灵活性使程序员在实现某目标时可选择多种途径，同一命令也可用不同的表达方式，运用得当可大大地提高运行速度，程序员应根据调用数据库的特点和程序运行的目的，权衡各个设计

方案和使用命令，尽量避免冗余运算，充分利用现有设备资源，选择结构精简运行速度最快的最优化程序。电脑专业的程序员了解多一些应用专业的知识在编程中往往能事半功倍。

本文得到中山医科大学卫生统计学洪楠教授和儿少卫生学邓桂芬教授的指导，广州市卫生防疫站电脑室谢耀伟同志提出宝贵修改意见，特此致谢。

参考文献

- (1) 郑存陆等，汉字 DABSEⅢ编程技巧及应用，广东科技出版社
- (2) 启明等，最新颖关系数据库汉字 FOXBASE+ 中国仪器仪表学会情报科学会
- (3) 钟凯勇，电脑编程技巧 200 例，电子工业出版社

的部分不能构成引用作品的主要部分或实质部分。陈文显然不具备这些必要条件（这方面详细的规定可查阅中国著作权法第22条及其实施条例第27条），因为沈文的内容（四个子程序）构成了陈文的主要部分和实质部分。陈某这样引用他人作品，又不作任何说明，就我个人的判断，这便是抄袭行为，也就是侵权。

甲：计算机软件不是有时候允许相似的吗？

乙：不错，只要这作品是独立创作出来的，就允许相似甚至十分相似。也就是说作者是与被相似作品相隔离的，或者它的相似是由于受到客观条件制约。我国的计算机软件保护条例规定，一个软件，在下列几种情况下才不会因为与已有的软件相似而涉嫌侵权：由于必须执行国家有关政策、法律、法规和规章；由于必须执行国家技术标准；或由于可供选用的表现形式种类有限。陈文中的这几个程序看来都不属于这几种情况。即使受到利用 CEC-I 的 MUSIC 语句这一特定情况的制约，而且由于程序很短，这种制约性是很强的，因此，它允许两个程序有很大程度的相似表达，这也正好说明版权是不排他的。但陈某在选择音响时从内容到指令利用，都没有理由一定要和已有的（沈文）程序一模一样。从法律的角度来看，如果陈某以受制约为理由来为自己辩护，那么，在应用 CEC-I 的 MUSIC 语句前提下，只要原告或第三方举得出可以通过另外一些表达形式（指令的条数，种类，参数等）来实现这几种声音，陈某的辩护便会被驳倒。陈文中的“救护车”是经沈文的“警车”程序中改变参数而来，这正好说明这类程序表达形式不是唯一的。如果陈某真是自己创作，经过严格的技术分析，确认因受制约而出现与已有的程序雷同，法律上是允许的，同样享有版权。

甲：那么，“无接触”状况下是否可以允许两个作品（包括软件）相似呢？这两个作品，存不存在“巧合”的可能呢？

乙：这就要看《CEC-I 模拟音响》的作者陈某能不能证实自己从未“接触”过《CEC-I 音响几则》这篇文章了。但这对陈某来说，是相当困难的。因为沈文发表在一个计算机知识的通俗刊物

上，只要不是生活在隔离的状态，就很难证明自己没有与一个通俗的同行的刊物接触过。而在涉嫌侵犯著作权时，法律却要求你能出示“隔离状态”的真凭实据。更重要的是陈某在他的文章的前面一部分，留下了抄袭的痕迹。按沈文中提到，如果把这些子程序用在你的软件中“将会为你的软件增色不少”。而陈文也来一句“那一定会使自己的软件增色不少”。这只不过是简单的改头换面罢了，连我们这些具有普通常识的人都不难作出判断。

甲：在研究作品的原创性时，还要不要考虑别的因素？

乙：通常，在判断某作品是否具备原创性时，还要考虑到作者的创作条件（能力与环境）与该作品是否相副。《电脑》杂志过去有一篇文章曾打过一个比方：要人们相信“葬花词”的作者不是林妹妹而是薛蟠大哥，那是不可能的。我们把这种判断称为“力度”分析。但从现在这个具体条件来看，陈某倒是具备编出这类程序的能力的，他编出无线电报声和报时音乐这两个程序便是证明。他的过失就在于抄的成份大大超过了自创的成份。我们希望他能纠正错误，真正发挥自己的特长。作品是作者的智力劳动成果，不论是对一般作品还是软件（程序），抄袭行为都是不允许的。

甲：未发表的作品享不享有版权呢？

乙：一个作品完成了，即使未公开发表也同样享有版权。把别人未发表的作品以自己名义抢先发表，也会构成侵权。我记得《电脑》杂志社也处理过类似事件。

抄袭行为目前时有发生。防止这一恶习，要靠编辑和广大读者的共同努力。因为编辑们接触面毕竟有限，广大读者可就不同了。抄袭者也许能逃过编辑的审查，但难逃过千万读者的眼睛，这就是所谓“躲过了初一，躲不过十五”吧！

对广大软件工作者来说，不仅要主观上摒弃抄袭别人作品的不良思想，而且还要增加一些法律知识，防止因一时疏忽而被指控侵权。

甲：谢谢你的讨论。

智能检测仪表中微型打印机的应用

重庆渝州大学自动化及电子工程系 何 伟

重庆大学无线电系 张 玲

【摘要】 测试结果的输出, 是任何检测仪器所不能回避的问题之一。本文较详细地讨论了被广泛作为智能检测仪器输出设备的微型打印机, 及其实现对文字说明、图表、数据和曲线的打印方法, 汉化打印的原理。并提出了曲线打印中能扩大打印输出范围的特殊打印方法——浮动窗口。

一、微型打印机概述

微型打印机是一种超小型点阵式打印机。这种打印机通常采用了日本的 EPSON 公司的 Model150 (或 Model1160) 打印机机芯。它可在 45mm (或 58mm) 宽的普通打印纸上打印行宽为 96 点位 (或 144 点位) 的任何点阵图形 (点大小: $0.35 \times 0.35 \text{mm}^2$, 点距: 0.35mm), 或打印行宽为 16 (或 24) 的点阵字符 (字符大小: 5×7 点阵, 字符窗口: 6×8 点阵)。使用低压、直流微电机驱动打印头架和走纸机构是这类打印机的一个突出特点, 因而它可以直接接向 +5V 单板机或单片机的供电电源, 这为采用了单板机或单片机作控制器的检测装置的应用提供了极大的方便。同时, 还具有体积小、重量轻、价格低、功能强、通用性好和使用灵活等特点, 它的应用, 为检测装置的一体化设计提供了实现的可能。现在, 随着计算机应用的普及, 微型打印机同微型计算机一道, 在小型工业实时检测控制、自动化仪器仪表装置以及教学实验中得到了越来越广泛的应用。

微型打印机按机内有无微处理器可分为两种类型。其一为普通微型打印机, 这种打印机必须与相应的微机配套使用, 微机本身就兼作它的控制器, 因此, 打印时须占用微机一部分甚至全部的开销。例如, TP-801P 微型打印机就是利用 TP-801 单板机作它的控制器。其二为智能微型打印机, 这种打印机由于自带微处理器 (单片机), 打印时完全不占用微机的开销, 与微机接口也十分简单, 因此, 通用性强, 使用较为广泛。例如, TP μ P-16A 就是一种智能微型打印机。在智能检测系统的应用中, 微型打印机不仅可用来打印全部标准

ASCII 代码字符、自定义字符、图符, 而且还用来打印自定义组合汉字、图表和曲线等。

二、打印文字说明和图表

打印文字说明, 在测试结果的输出报告中是必不可少的。因为, 无论是何种应用领域, 检测的结果往往不会是孤立的数据, 它必须与测试条件比如初始预置、环境温度等联系在一起才具有实际的意义。例如, 在行为生物学中, 动物的行为不仅与温度、湿度有关, 而且还与季节、检测的时间、动物的体重和大小等有着密切的关系。同时, 人们常常还希望输出的报告能提供诸如检测的日期、测试时间长度等信息, 以便于直接存档和检索。因此, 在打印输出测试报告时, 有必要将这些参数、信息一并打印出来, 如图 1 所示。

```
*****
*CQDX-1 检测报告*
*****
日期: 1991-03-25
时间: 08:40
温度: T=26℃
:
:
:
```

图1 打印文字说明

打印文字说明必须要按一定的格式, 以使之清晰易读。如何才能得到一满意的格式呢? 这正是关键所在。根据笔者长期使用的经验, 建立字符打印格式数据库是行之有效的办法。即, 按任意自定义的输出格式, 将待打印字符的 ASCII 代码 (包括空格) 组成的数据集合写入 EPROM 中, 这些数据的集合就构成了字符打印格式数据库。打印时, 首先将整个数据库从 EPROM 中复制到指定的 RAM 区中, 对其中需要修改的初始预置、环境温度以及日期、时间等数字, 经 ASCII 码转换后写入相应的单元, 然后调用字符打印模块程序, 以打印整个数据库中的 ASCII 字符, 这样就可将复制在 RAM 中并经修改后的文字说明按原定的格式打印出来。

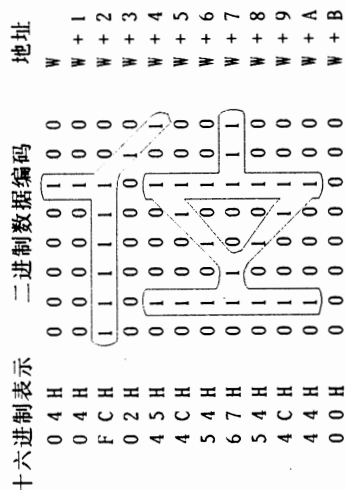


图 2 汉字编码举例

文字说明通常采用汉字以适应我国的国情和增强报告的可读性。然而, 打印机内部固有的字符中, 除了 96 种标准 ASCII 字符和一些特殊字符、图符处, 一般只有几个简单的常用汉字 (如年月日、甲乙丙等), 因此, 有必要利用打印机的自定义字符功能, 编制所需的汉字字库。打印机的自定义功能因机型不同而有一定的区别。普通微型打印机只能利用 EPROM 中留给用户的自定义字符区, 而智能微型打印机则还可利用“程序自定义”命令进行现场实时定义。尽管如此, 对汉字的编码却是一致的。图 2 是“位”字编码的例子。从这个编码中可以看出, 汉字字符窗口采用了 12×8 的点阵, 汉字编码被写入 12 个内存单元, 每个汉字占用了两个标准 ASCII 代码字符的打印位置和计算机内存空间。

它实际上是由两个自定义字符组合而成，因此被称为组合汉字。根据需要，汉字还可由更多的自定义字符组合成更大的汉字，但这种情形在编制打印程序时有特殊要求，因为，每个汉字将占用多个字符行。

打印文字说明时，在不同的应用领域常常需要打印一些不同的特殊字符或图符。例如，在药理学的实验输出报告中，♂和♀是经常用到的图符。不同打印机通常都是提供了一些特殊字符或图符，如果不敷使用的话，则必须利用打印机的自定义功能，对所需特殊字符或图符进行自定义编码。

打印二维图表,是为了使检测结果的输出报告更简洁、明了。打印图表与打印文字说明具有完全相同的过程。不过,这里建立的数据库是图表格式数据库,整个数据库除表头有少量文字外,其余部分均由制表字符构成,这样的数据库实际上是一个空白表格。打印前,必须将检测所得的数据结果转换成相应的 ASCII 代码,并填入复制到 RAM 里的图表格式数据库相应的单元中。

三、打印数据结果

数据是检测过程的直接目的，也是结果输出报告的主体。一般的检测过程，尤其是在实时监测中，所得的数据是大量的，为了便于数据分析和后续的数据处理工作，通常必须将这些数据直接罗列出来，即以采样点的时间先后顺序和一定的格式，将测试所得的全部数据直接打印在输出报告中。

打印数据结果时, 如果也采用打印文字说明的方式, 则存在如下问题: 首先, 由于输出的数据量较大(比如几百或几千个数据), 因此, 建立的打印格式数据库也必然较大, 一方面编制起来太繁琐, 另一方面数据库将占去较大的存储空间, 这对小型检测仪器是极不现实的。其次, 由于每次检测输出的数据量, 将随着采样密度和检测时间长度等初始预置的不同而不同, 因此, 也不可能通过建立一“死”的数据库来完成打印数据的要求。

为此,作者提出了通过建立一“活”的数据库的方法来完成打印数据的要求。显然,大量的数据输出,必然占据若干个打印行(例如几十或几百行),而每个(或每几个)打印行又具有相同的打印格式,如图3所示。如果,只建立一个包含一行(或几行)内容的打印格式数据库,这将节省大量的内存空间。打印时,首先将这个小程序从ROM中复制到RAM,然后将待打印的首行(或首几行)的数据经ASCII转换后依次填入RAM数据库里相应的单元中,并进行打印;打印完

后, 计算机再次将下一行 (或下几行) 数据的 ASCII 码填入库中, 并打印; 由于计算机的速度相对于打印机而言是无可比拟的, 这种通过刷新数据而“激活”数据库的实时处理方法, 并不影响打印机正常的打印, 该过程在程序的控制下不断重复, 直至全部数据打印完毕。

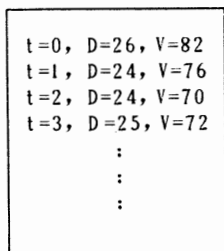


图 3 打印数据结果

必须指出, 如果测试结果数据量较少, 或测试结果所得的仅仅是几个参数值 (如压力、流量等), 则完全可以简单地用图表形式或文字说明的形式输出数据。

四、打印曲线图形

数据除了以图表形式和直接打印输出外, 为了使打印报告能更直观、形象地反映检测结果, 还常常以图形方式输出。例如, 振动曲线、心电图波形等。曲线打印功能在不同的微型打印机上有不同的实现方法。智能打印机中, 曲线打印可简单地直接反复使用“打印曲线”命令来实现, 而普通打印机则必须利用图形打印方式, 并通过设置打印点行数为 1, 和不断查询打印机的工作状态以控制打印机的工作, 来实现对曲线的打印。但不管哪种打印方法, 均可得到横坐标沿纸长方向的相同的曲线图形。

在打印曲线时, 由于微型打印机的打印行宽只有 96 点位 (或 144 点位), 这使得打印输出的曲线将受到极大地限制, 即纵向打印点迹的分辨率仅为 96 (或 114), 这在很多的应用场合是难以满足要求的。为此, 作者提出了能克服纵向输出范围低的打印方法——浮动窗口法。

浮动窗口法的基本思想是: 在打印曲线时, 将打印机看作一个浮动的输出窗口, 该窗口可根据具体的情况上下滑动以适应具体的要求, 如图 4 所示, 对窗口数据的打印是通过调用窗口打印软件实现的。窗口的高低由

进入检测时的初始预置确定, 并通过打印文字说明的形式反映在输出报告中。

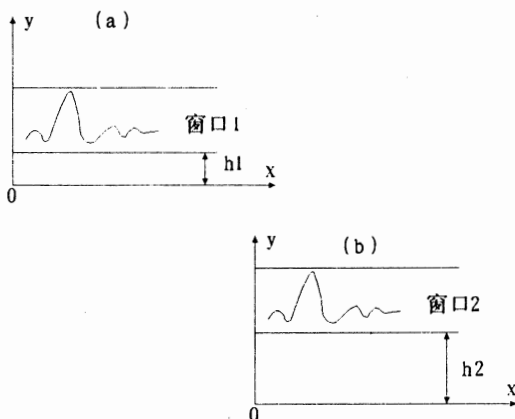


图 4 浮动窗口法示意图

浮动窗口法应用的关键是窗口高低的选择, 它必须因具体情况而定。例如, 当曲线幅值较小时, 可选择较低的窗口, 如图 4 (a); 反之, 当曲线幅值较大时, 可选择较高的窗口, 如图 4 (b)。因此, 在分析窗口曲线时, 真实曲线的幅值应是窗口曲线的幅值加上窗口的高度, 这在两次所得的曲线间的比较时十分有用。但是, 浮动窗口法的应用仍有其局限性, 其前提是: 所有打印曲线幅值的最大变化不超过 96 (或 144) 个单位。不过, 这已能应付大多数的应用场合了。如果在应用中, 曲线幅值的最大变化超过甚至远超过 96 (或 144) 个单位、或希望能提高输出曲线的分辨率, 还可采用压缩窗口法, 在此就不作介绍了。对该部分感兴趣的读者, 可来信与笔者联系。

五、结束语

打印机的应用中, 尤其值得注意的问题是代码转换。因为, 一方面, 所有待打印字符必须全部转换成相应的 ASCII 码, 另一方面, 文字说明和图表中的数字以及数据的打印必须采用 BCD 码, 因此对数字而言, 首先须进行 BIN-BCD 码转换, 然后再转换为 ASCII 码。这些大量的代码转换工作, 要求计算机在打印过程中进行实时处理, 即边转换边打印。

随着智能仪器的推广, 微型打印机的作用已越来越为人们所重视, 它的特殊应用还有待于进一步的探索。本文所述的打印方法, 均在实际的智能检测系统的应用中得到了验证, 并取得了良好的效果。

“爬格子动物”的新生

北京电影学院文学系 刘一兵

有人管我们这些搞写作的人叫“爬格子的动物”，于此可见这一行的艰辛。比如我，写一个电影剧本至少要填满四五本格子，就算手快，没有一星期也难以完成。如果再抄一遍，就又要三、五天。如果遇上个挑剔的编辑或出版厂家，让你改上几回，到最后你非被整得见了自己的作品就恶心不可。有时恨不得一把折断手中笔，改行去卖冰棍！正因为如此，当我得知有人在用电脑写作时，就毅然地掏尽了全家的积蓄，赶了一趟电脑“时髦”。

然而，电脑一架起来，苦恼也得随之而来。首先你得掌握一种中文输入的方式。市面上的中文输入软件五花八门，总共有上百种。我发现大多数的打字小姐用的是“五笔字型”，她们在说说笑笑之间就可以把中文飞快地输进去。真有点“羽扇纶巾谈笑间，檣櫓灰飞烟灭”的意思。于是我也就选定了“五笔字型”。可是，不久我就发现我错了。不是“五笔”不好，而是它不适合我用。首先，我的年纪已不比那些打字小姐们，“五笔字型”的口诀太多，规则太复杂，我记来十分吃力。有时一个字的某一部件的拆法记不准了，就使我中断写作半天。再者，打字小姐的工作是把现成的稿子录入进电脑，她们可以看着稿子进行拆字，甚至不用想稿子的内容是什么意思。我则不行，我的工作写作，脑子里出现的是形象的画面，是融汇着复杂情感的话语，是一旦激发起来创作冲动就会声泪俱下的激流。如果让我不断地中断感情和形象思维从创作中跳出来，去琢磨如何把中国字拆成几块，我的创作情绪就会立刻遭到破坏，可怜的灵感也随之烟消云散。看来，我必须找到一种既好学又高效而且不妨碍文学思维的中文的输入方法。可是，人们都告诉我，那是不可能。

我是在几乎绝望的时候偶然发现“自然码”的，却没想到她满足了我的所有愿望。首先这是以拼音为主进行中文输入的软件，而恰好我们在写作状态下大脑里首先出现的不是字型而是声音，这就保证了我在写作时的思维不会受到因拆字而产生的干

扰。你根本不用想那个字是如何写的，你只要脑子里“念念有词”，文章就会不断地从手下流出，她的拼音方法是双拼，每个字只需打两个键。由于是以词组的方式输入的，不仅效率高，而且几乎没有重码。最方便的还是她那独特的智能相关处理功能，它能帮你从重码词组中选取你所需的词。更值得一提的是“自然码”的自造词方式，你在写作中无需象使用其它软件那样退出编辑状态就可以把你所需的任何词造进去，下次再写到这个词的时候它就会应“邀”而出。我这个“电脑盲”加笨伯，在没有任何人教的情况下仅仅在写完一篇不长的文章之后就学会使用“自然码”了。我发现，仅仅追求录入速度是片面的，因为在写作的时候你更多的时间是花在构思上，花在词句推敲上，决定你写作速度的还不是录入的速度，而是思维的速度。例如我的录入速度是每分钟二十五字左右，听起来似乎太少，但是实际上足够用了，你依此速度至少一天可以写下一万多字哩！这样，一部构思完美的剧本通常只用四天即可完成。关键的好处是你既不用再誊写，也不用怕再改动，在电脑上改稿子是最方便不过的事了。

我如今已经用“自然码”写完了七部电影电视剧本和几十万字的文章，即使是写信也不再习惯用笔了。不知为什么，现在一天不摸摸键盘我的手就痒痒。只要在键盘前一坐，就会产生一种自信，知道自己没有被时代抛弃，这种异样的感觉使我觉得自己不是在写作而简直是在钢琴前弹奏心灵的乐章。我想，这都要归功于“自然码”呀！我的同事们看到了我的战绩以后也纷纷购置了电脑，如今我所在的文学系几乎成了电脑系，而且人人使用的都是“自然码”。我们还策划着要来一次“自然码”的输入比赛呢！

编者：大量的文字录入，可以讲是广大电脑使用者的一大烦恼，如果能用上一种好的输入方法，每个电脑的使用者都能高效地录入，烦恼就会变成乐趣。

需要自然码有关资料的读者，可以来函来电向本编辑部索取。

新一代汉字操作系统

——TTDOS

钟华宁

汉字操作系统 (CCDOS) 使电脑在中国的广泛运用起到了决定性的作用。所以自电脑进入中国以来, 广大的电脑研究人员便开始了对汉字操作系统的研制, 十几年来, 各类汉字操作系统层出不穷, 其中不乏佼佼者。如 2.13 系列, UCDOS, 联想 CCDOS 等, 其中以 2.13 系列最为出色, 2.13 系列以其广泛的适用性, 多种字体的打印深受广大计算机用户的喜爱, 堪称是中国汉字操作系统的一个里程碑。

但以上汉字操作系统都有一个致命缺陷, 即大部分西文软件都必须经过汉化才能在其操作系统上运行, 而汉化存在以下几点:

(1) 汉化周期长

汉化是一件难度高且极其烦琐的工作; 所以汉化一件大型软件需要较长时间, 这样一来, 无疑阻碍了中国计算机用户掌握最新国外先进软件的趋势。

(2) 软件质量降低

汉化后的软件大部分在运行速度, 软件可靠性方面都较原西文软件为低。

(3) 版权问题

一个汉化后的西文软件其版权属于谁同样是一个令人头痛的问题。

那么, 能否研制出一个可以直接运行未汉化的西文软件的汉字操作系统呢? 笔者经过一段时间努力, 并在许多同行和专家的指导下, 成功研制出这一套新一代的汉字操作系统——TTDOS。并献给广大的计算机用户。以下简述一下 TTDOS 的各项功能。

一、西文软件无须汉化直上中文

在 TTDOS 的试用过程中发现: 绝大部分 (尚未发现反例) 西文软件均能在 TTDOS 中安全可靠地运行, 并且能进行汉字的输入输出。如我们常用的 PCTOOLS 系列, Norton 系列, Borland 公司的语言

系列 (Turbo Pascal, Borland C++ 等), 数据库系列 (Dbase II, Foxbase, Foxpro 等), 文字编辑软件 (Wordstar, PE, Qedit 等), 以及其它一系列应用软件和系统软件。

二、网络功能

TTDOS 可以仅存在服务器硬盘中, 为各工作站共享, 打印字库不占用工作站任何资源, 而且打印速度决不比单机运行时为慢。

三、汉字输入法

TTDOS 的汉字输入法与目前流行的 CCDOS (如 2.13 系列) 基本相同; 有区位、拼音、首尾、快速、五笔和词组等多种汉字输入法, 增加了万能学习键和字典功能; 能方便地进行全角和半角的切换及制表符的生成。若某些西文软件的控制键与汉字输入键相同时, 用户可以屏蔽原有的汉字输入键而采用 TTDOS 提供的一套特有的输入键输入汉字, 解决两者在键盘控制上的矛盾。

四、适用性广

从低档的 PC 机至高档的 486 机及大型网络系统, 无论是 MDA 单色显示器或者是 VGA 高分辨率彩色显示器均能使用 TTDOS。当然, 越高档的机器越能充分发挥 TTDOS 的强大功能。

五、多种字库驻留方式

除可驻留在其本内存中外, 还提供了驻留在扩展内存, EMS, XMS, 硬盘, 软盘和驻留压缩字库等功能, 而且为 1M RAM PC 机用户专门设计了一个使用其高端内存的字库驻留程序, 这无疑对 PC 机用户来说是一个福音。因为他们终于有可能在进入汉字系统后运行大型软件了。

六、占用内存少

在不考虑字库的情况下, VGA 版和 MDA 版仅占

一种非磁盘间的文件拷贝方法

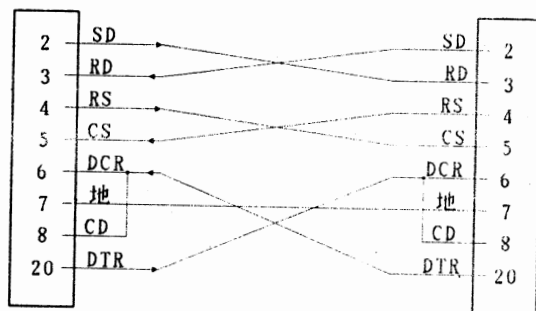
广州市计算机应用研究所 赖宏宇

通常，将文件从一台 PC 拷贝到另一台 PC 时，都采取将文件拷贝到磁盘里，然后把磁盘拿到另一台 PC，再把文件拷贝到 PC 里的方法。当两台 PC 靠得很近，人们可能会想到：能否用一条简单的电缆，把两台 PC 连起来，让文件从一台 PC 直接传送到另一台 PC，而不采用以磁盘作中间过渡的间接拷贝方法。况且，直接传输的误码率会较间接方法低。下面就介绍这样一种简便的做法。

一、硬件处理

我们都知道，PC 机上均配有一至两个 RS-232 接口。通过一条电缆，将此两台 PC 的各一 RS-232 接口

连接起来, 就能实现通信。电缆的具体接法如图 1。



[图 1]

48K 和 40K 内存(包括区位, 拼音, 首尾, 快速四种输入法)在一些内存资源管理软件的支持下, TTDOS 可以不占用任何基本内存(包括字库及 TTDOS 辅助软件)。

七、打印功能

TTDOS 无专门的汉字打印驱动程序, 但可直接利用 2.13 系列的打印驱动程序, 百分之百兼容, 这样, 我们就可以在 TTDOS 中利用 2.13 的强大打印功能了。另外, 提供经修改过的 Ncw9p 和 All9p 程序, 供九针打印机用户使用。

八、屏幕管理独具特色

所有西文软件在 TTDOS 上运行的输出效果与直接在西文 DOS 中运行的效果完全一样,其中包括字符和汉字的彩色输出、加亮、下划线,闪烁均能在 TTDOS 中完美实现。自动识别西文边框符,处理直接写屏速度极高,明显快于同类系统,包括以显示速度著称的倚天汉字系统。总之,在西文 DOS 中可以实现的输出效果在 TTDOS 中也能实现,这是许多同类系统无法实现的。同时 TTDOS 保留了西文 DOS 的所有功能和 BIOS 功能,用户可在其上开发任意应用软件(包括绘图软件),而且扩展了一些特有功能,方便用户对系统进行控制管理,汉字显示和绘图等工作。

九、辅助工能

TTDOS 为用户提供一套实用的辅助功能，能在任何情况下弹出菜单并选择相应的项目。如可以查阅

ASC 码表, 各区汉字及其区位码, 而且有一套专门的编辑键帮助用户查阅。另设有一个微型计算器, 用户可利用它进行数值的四则运算, 逻辑运算和进制转换等工作。总之, 这一套实用工具无疑是程序员的有效助手。

十、系统工具

TTDOS 提供了四套 ASC 码显示字符集供用户使用；用户可对词组文本文件进行修改并编译；16 点阵汉字编辑程序等，详细功能可查阅使用说明书。另外，绝大部分 2.13 的系统工具也能在 TTDOS 中运行。TTDOS 虽未提供专门的鼠标驱动程序，但因为拥有高速的字符显示功能和极高的兼容性，用户可直接利用其原有的西文鼠标驱动程序使用鼠标。效果是令人十分满意的。

TTDOS 无疑是新一代汉字操作系统的方向, 对 TTDOS 有兴趣的用户可直接与我们联系并欢迎上门参观指导。注: 最近在广州地区发现有某些单位未经笔者许可, 擅自出售该软件 (笔者研制过程中的试用版本), 敬请广大用户不要上当受骗; 最新版本的版头应为 Super TTDOS II。

地 址：广州市广源路云泉直街2号海垦大楼716
房超越电脑公司。

邮 编: 510500

电 话: 7700411-716 或 714

B B 机: 181-718848

联系人：钟华宁

95

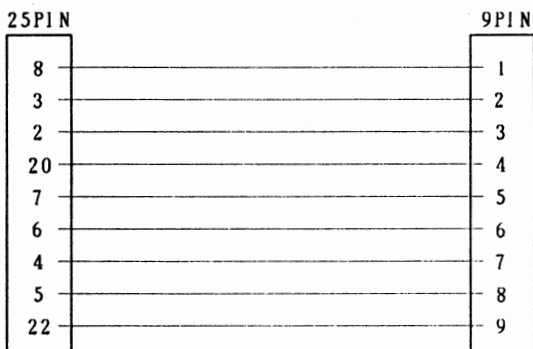


图 2

这里的 RS-232 接口是以 25 芯为标准。如果 PC 机上配的是 9 芯的 RS-232 接口, 可按图 2 作 25 芯与 9 芯的转换。将做好后的电缆接到两台 PC 的 RS-232 接口上, 就可开始通信。

二、通信软件

想在 PC 之间通信, 光有硬件还不行, 还要运行相应的软件。我们用 BASIC 语言编写了一个简单的通信程序, 该程序分发送程序和接收程序。文件发送程序为:

```
10 REM Basic Send File Demonstration Program
20 INPUT "SEND FILE:";SFILE$
30 OK$ = "SEND-READY":ANS$ = "RCV-READY":END$ = "END OF DATA"
40 ON ERROR GOTO 210
50 OPEN "COM1:9600,N,8,2" AS #1
60 OPEN SFILE$ AS #2
70 FIELD #2,128 AS C$
80 PRINT #1,OK$
90 INPUT #1,A$
100 IF A$ <> ANS$ THEN 80
110 ON COM(1) GOSUB 160
120 COM(1) ON:I=1
130 IF EOF(2) THEN 220
140 GET #2,I
150 PRINT #1,C$;I=I+1:GOTO 130
160 V$ = INPUT$(I,#1)
170 IF V$ <> CHR$(19) THEN 200
180 V$ = INPUT$(I,#1):IF V$ = CHR$(17) THEN 200
190 GOTO 180
200 RETURN
210 PRINT "ERROR OCCUR,EXIT FROM PROGRAM"
220 C$ = ENDD$+SPACE$(109):PRINT #1,C$
230 CLOSE #1:CLOSE #2:SYSTEM
240 END
```

文件接收程序为:

```
10 REM BASIC RECEIVE FILE DEMONSTRATION PROGRAM
20 INPUT "RECEIVE FILE:";RFILE$:END$ = "END OF DATA"
30 OK$ = "SEND-READY":ANS$ = "RCV-READY":XOFF=0
40 ON ERROR GOTO 180
50 OPEN "COM1:9600,N,8,2" AS #1
```

```
60 OPEN RFILE$ AS #2
70 FIELD #2,128 AS V$
80 LINE INPUT #1,A$
90 IF A$ <> OK$ THEN 80
100 PRINT #1,ANS$:I=1
110 C$ = INPUT$(128,#1):IF LEFT$(C$,19)=ENDD$ THEN 200
120 LSET V$ = LEFT$(C$,128):PUT #2,I:I=I+1
130 IF LOC(1) > 128 THEN 160
140 IF XOFF=0 THEN 110
150 PRINT #1,CHR$(17):XOFF=0:GOTO 110
160 IF XOFF=1 THEN 180
170 PRINT #1,CHR$(19):XOFF=1:GOTO 110
180 PRINT "ERROR OCCUR, FILE STORED INCORRECTLY"
190 GOTO 110
200 CLOSE #1:CLOSE #2:SYSTEM
210 END
```

程序说明:

(1) OK\$ 定义激活接收方标志; ANS\$ 定义回答激活信号标志, 也作为开始文件传送标志; ENDD\$ 定义文件传送完毕标志, 它是文件中没有的字符串, 可自行修改。

(2) 程序不但可以传送文本文件, 连普通的二进制文件也可传送。

(3) 程序执行 XON/XOFF 协议。当接收方输入缓冲区数据大于 128 字节时, 发 XOFF 通讯控制, 请求对方暂停发送, 然后用 XON 恢复之。

(4) ON COM (1) GOSUB 160 和 COM (1) ON 表示当允许 COM1: 通信时, BASIC 程序每执行完一句语句后就检查输入接收缓冲区有否输入字符, 有则不执行后继语句, 而进入 ON COM (1) 指定的陷阱子程序, 这样就为异步通信提供了手段。

(5) 这里有些通信语句平常较少介绍, 如 C\$ = INPUT\$(128, #1)、PRINT #1, CHR\$(19)、CLOSE #1 和 LOC (1) > 128 等, 有关的具体意思和功能, 请读者参考 IBM PC BASIC 的资料。

(6) 回答屏幕提问时, 请按照 drive:\path\filename 的格式整个输入文件名, 因此, 要事先知道原文件与目标文件的存放位置及文件名。

三、编后语

要在两台 PC 间传送文件, 这并不是唯一的办法, 程序也不是唯一的。市面上流行的一些通信软件也可在此硬件环境下传送文件, 不过操作稍为复杂。

本程序还有许多可改进之处, 如发送和接收可合二为一; 生成的目标文件长度为 128 字节的倍数, 可修改程序改变之; 接收后的目标文件时间变为新时间, 读者可想办法恢复为原文件时间……总之, 本文的目的, 是提请各位注意, 可以用如此简单的电缆和语句, 就能实现 PC 间的通信。但愿此文能起着抛砖引玉的作用。

高级UNIX连网技术讲座

广州昂立自动化工程公司 冯家宁

第一讲 UUCP(UNIX to UNIX COPY)(上)

1.1 引言

UUCP 是 UNIX 主机之间的 cp 程序,但它不只是拷贝程序。主机之间的文件拷贝只是这一通用连网程序的应用之一。这是一个批处理/缓冲池系统。UUCP 的字面含义似乎限制了它的使用。也许 II 是一个更好的称呼,因为那样能更明确地表示 UUCP 是一个包含文件转送、远程执行、维护和管理等许多程序的系统。所有这些程序历来都命名为 UU 某某。

在 UNIX 连网教科书中,UUCP 占有很大的比重。第一个 UNIX 网络应用程序设计在 UNIX 系统上用类似 UNIX 命令语法的方式运行,并可依靠其它 UNIX 程序的支持来运行。UUCP 需要 mail、sh 和 cron 的支持。这种依赖性既是 UUCP 的优点也是它的缺点。sh 能够并且习惯于实现任意复杂的远程命令执行,UUCP 则不太复杂。而且新的语义有可能加到 sh 上,新的语法有可能加强抑制 UUCP 被滥用的能力。程序模块通常被装配成可产生一定功能的软件包。但软件包可能被分成许多组成部份。

这里我们继续 UUCP 功过的争论。UUCP 的存在理由继续受到主张更完善的网络系统的人的质问。Dennis Ritchie 说:“UNIX 是一个简单的、连贯的系统,它把几个好的主意、模式推向极限”。他是 UNIX 设计的参与者、实现者。UUCP 与系统配合得很好。虽然曾有过一些奇怪、笨拙的版本在几年里出现,但成功的、寿命长的设计都离不开功能和硬件的一般化、系统部件模块化以及对现有和将来程序及设备的利用。

UUCP 总是那么受欢迎是因为它的成本低。就外围设备、处理周期和管理要求来说,它是很有效益的。UUCP 之方便可以抵消大多数的连网问题。所以它是一个通用系统。本讲讨论 UUCP 有两个目的。首先是让你熟悉历史上第一个 UNIX 连网方案。UUCP 运行有很长一段时间了。程序有了改进,增加了新的部件,但基本的策略仍旧是一样的。UUCP 的最大设计缺陷是它没能适应连网机器的数量以及大量的数据处理。在这方面虽然作了些补救,但其实现和对问题的处理也许只适用于某些教学目的。本讲的另一目的是提供一些必要的信息使读者能成为 UUCP 的专家。值得提醒的是,作

为一个专家并不只是了解一个系统如何工作。专门的知识往往通过探求为何一个系统不工作时获得。我们将讨论如何调试跟踪 UUCP;不过也许除非你确使用上它们,否则你是不会对这些内容感兴趣的。

熟悉 UUCP 的人应记住它与所支持的服务之间的区别。电子邮件和网络布告牌(Bulletin Boards)也许有赖于 UUCP,但它们不属于 UUCP。当出现电子邮件故障时,有可能是 UUCP 的问题也有可能是邮件系统本身的问题。这就是 UUCP 原理的使用。对 UUCP 的一些了解是可以用到许多有趣的工作上的。由于它的低成本和通用性,UUCP 使你既能快又省地建立起你的网络。

1.2 历史背景

1977 年 1 月前的一些时间是 UNIX 的复兴时期,贝尔实验室的 Mike Lesk 创造了 UUCP。这要追溯到在 1977 年 2 月举行的 UNIX/MERT 用户会议议程。它被印在 1977 年 2 月在贝尔实验室内发行的 Mini-System Newsletter 上。

有趣的是其中的第四项“UNIX 与 UNIX 通讯”并没提及 UUCP,它只提到了:

- * Ken Thompson 论 MPX 驱动程序
- * Dale DeJager 论 UCAT
- * Lee McJager 论 CU
- * PWB 机器间的通讯
- * Ken Thompson 论 KMC 微处理器对通讯的可能应用

但是在第七项的行间,我们能读到“软件的自动运送”

* Mike Lesk 关于更好地分发软件和对其在本地测试的方案。

* 讨论

该方案就是我们所认识的 UUCP。

在另一期出版物里写到:

有一个新的命令,叫“UUCP”,它调用系统以实现 UNIX 系统间的文件拷贝。该系统是 Mike Lesk 写的,它能使以 ACU 为宿主及中介的任意两台 UNIX 机器之间通过电话线以 300 波特以上的速度拷贝文件。USG 对该这一改进代码共享的技术感兴趣。任何人若

有 UNIX 和 ACU(自动呼叫单元)并对此有兴趣请联系 Aaron Cohen 地址 MHX6920.

在 1977 年 7 月一期里报道了软件仓库的建立以提供和接受软件。这里开始使用 UUCP 的命名。该书最后在 1978 年出版。Lesk 的目的是在建立一个系统,以他为中心可自行管理数目不断增长的 UNIX 机器。该系统的先决要求是提供一种机制使机器之间能自动转送文件。当时还没有这方面的任何东西,于是他写了 UUCP。uucp 命令在缓冲池目录中通过创建缓冲文件为请求建立缓冲池。待发送的数据先拷贝到 f 文件;目标系统和文件名、用户(目标系统会用 chown 查看)、存取权限(如是二进制形式则与 4096 作“或”运算)拷贝到 s 文件;一个类似的 r 文件被创建给请求用。然后 uucp 执行 tdemon,处理缓冲文件。

tdemon 执行以 e 打头的文件而对文件名以 c 打头的文件则有条件地执行。它们包含有文件名表并以空行、命令相随。当所有的相关文件都存在时,命令便开始执行。tdemon 使用一些文件与远程机器连接。这些文件表示连接的类型(直联或拨号)、电话号码(如用得上的话)注册名、口令、UUCP 应用程序的主目录。包含有敏感信息如电话号码、注册名的文件是以加密形式贮存的。存取这些文件的程序都带有解密匙。密匙本身又被加密以防轻易从程序的二进制形式看出,密匙的解密由另一包含不同密匙的程序执行。然后密匙作为 crypt 的参量去为敏感文件解密。这种对信息的谨慎处理也许使读者产生怀疑,但这一版本的 UUCP 是要在远程系统上注册并获得一个 shell 环境的。它所做的第一件事是用 chdir 转到 UUCP 子目录。随后 tdemon 运行 transmit。transmit 是一个通用工具程序,它把远程被等待的机器信号作为参量并对远程的键入作出响应。Lesk 提到有一个用户用 transmit 连接 ARPANET 并拷贝他的文件给 UNIX。拷贝文件的程序具有为 7 位接口转换二进制文件的能力。它们计算核对和,必要时可重发。经验说明用 300 波特线路是很少需要重发的;实际上核对和错误很多是程序本身的问题(有时候是驱动程序的问题)。

下一版的 UUCP 由 Lesk 和 Dave Nowitz 所写。这一版的 UUCP 开始广泛发行(PWB 和 V7)。它是全部重写的,但目的不变。最为显著的变化是当注册时 UUCP 不再接受交互式的 shell 而是执行一个用户不能直接利用的程序(uucico),它在通信线路的两端运行,一主一从地交替角色。其它的改变是增加了 uux 及改善了传输协议。

uux 为命令的远程执行提供了用户介面。当校验和错误的出现不再全是软件毛病时,也许就需要有传输协

议了。Greg Chesson 写了最初的但仍然是标准的电话线协议(g 协议),它能对数据打包、核对、作必要的重发。Lesk 和 Nowitz 有足够的远见提供多个协议。很明显,在两台机器直接连接时,比 g 协议高级的协议是没有必要的。以上两种连接便是当时所有可能的连接。但在以后的版本中,随着连接类型的增加,出现了对多种协议的选用。从表面上看,以后的版本都与第 2 版相象。该版是 1978 年到 1983 年间的标准。直至当时, UUCP 一直都因使用中查到的缺点而作改动。UUCP 并非设计很糟或编码不当以至需要 5 年的时间和一大批人去作调试。难办的是 UUCP 的广泛使用已经使它失去了控制。

当今的 UUCP 用户组织里,发送邮件给其它系统的用户是最重要的事情。UUCP 也许是造成 UNIX 急剧增长的原因。UNIX 变得如此普及是因为它是那样容易理解,因而也容易修改和增加东西。这包括了命令、应用程序以及内核。源代码既容易得到也乐意分发。UUCP 做到了更及时地分发,用户间快而频繁的通信。越来越多的软件应用在越来越多的地方就对 UNIX 提出越来越多的要求。UNIX 系统越多产生的软件也越多,要求也越多,如此不断循环。在当时 1200 波特异步 modem 已是很先进的情况下,UUCP 的设计也无法满足 UNIX 连网、适应不同的 UNIX 版本、传输更多字节、更远更快的需求。主要问题有安全、性能和设备相关。Mccady 通过他的亲身努力,编写了长长的 UUCP 出错报告,这是当时最好的 UUCP 信息资源,这里是一个条目的例子:

12) Sequence number generation is inefficient, slow, ineffective

References: genname.c

Solution: install Alan Watt's genname() mods - use base 36 or 64

Reported-by: ittvax!swatt (7 / 82)

Applicability: V7, BSD, S3

有 39 个条目并带有对不同版本通信细节所作的讨论。单是 81 年 11 月到 82 年 11 月就有 22 人编写了 39 页的资料。这些当然不是结束语而是一种启发。在当时 6 个版本(令人吃惊的是它都是兼容的)对出错报告读者来说都是熟悉的,而在其它地方还有别的版本。现在可以组织力量把这些修改、提高的建议容进一个 UUCP 里了。

UUCP 对那么多人来说是那么重要,以至不得不对它作更多的研究。大的项目主要集中在电子邮件和布告牌。系统管理员必须有能力作调试、增强和管理 UUCP 否则会出现数据丢失和通讯线路阻塞。对要求

高的人来说,这些工作是很耗时间的。使用的增长继续超越开发。在过去系统管理员(特别是大多数实验系统)不参予开发工作。也许他们的工作是硬件设计,但如果他们需要使 UNIX 发挥得更有效,他们就会成为软件专家。也许原来是想写一篇论文,但使用 UNIX 系统更为诱人。在任何情况下都是用户直接参予 UUCP 的技术支持和开发而不是供应商(对 UNIX 来说一直是那样)。

在 1983 年 4 月贝尔实验室的 Martin Levy 发送电子邮件给对 UUCP 有兴趣的同事。他的消息主要是呼吁统一每一个人都会使用到的 UUCP 的版本以省去大量的重复工作。如果所有人都使用一个版本,管理 UUCP 也将省事得多。很清楚,UUCP 管理的很大一部份是与他方协商,解决为何他们的 UUCP 不能工作的问题。有时你不得不证明原因不只是版本不对。于是在 1983 年 4 月中,十几个来自贝尔实验室的 UUCP 专家聚集在一起讨论。最有建设性的事是在使用哪一个 UUCP 版本上取得了一致意见。Peter Honeyman 记录到:

关于 UUCP 的活跃的讨论是会议的主题。UUCP 的版本就象使用它的地方一样多。主要的竞争者是 USG6.0,由 Morris 重新编码 Tom Truscott 修改。尽管 Cohen 对 6.0 的优点表示冷淡。但大世所趋,我们同意了以此为起点建立新的版本。

Honeyman(他所用机器的注册名: honey), Nowitz(dan)和 Redman(ber)立即着手写新的 UUCP(现在称为 Honey Dan Ber)。Steve Bellovin 为 Honey Dan Ber 写了很多代码使它能在 TCP/IP 上用 Berkeley 版 UNIX。最后就是一个版本的 UUCP,它包含在 Berkeley 和 Research 版 UNIX 上,由 AT&T 分销。这样便满足了所有人的需要。Honey Dan Ber 成了 AT&T 的标准又名基本连网工具程序并且被发明 UUCP 及 UNIX 的组织接受。虽然 Berbelor 又有它自己的版本,以下取自它的 README 文件:

@...Honey DanBer [sic] uucp (aka BNU1) which is as close to a standard uucp as there is...

意即不能排除其它版本接近标准的可能性。不管你运行的是什么版本,基本原理都和 Lesk/Nowitz 的版本大体一样。这就是 Honey DanBer 和 Berkeley 版的情况。除非特别说明,以下讨论的主要是 Honey DanBer 版。

1.3 一般描述

UUCP 是一个连网工具程序吗?网络是什么?如果说网络就是使我们能在计算机之间传送文件。在远处执行命令,那么 UUCP 是用于实现连网的。网络要求一些机器间的某种合作。但在 UUCP 运行现场并没有强制任

何管理控制,一个 UUCP 网往往趋向于无政府状态。也就是说,虽然一组机器都运行 UUCP,但无法确保任意两台机之间可以彼此通信。即使可以通信,远程执行命令的级别和文件传送的权限也许是很高的。作为一个 UUCP 网络的成员并不意味着你可与网上的所有成员通信。那是因为它并不是一个真正的网络。每一台被使用的计算机的通信机制(设备、注册等)都必须记录到本地机的数据库上。UUCP 是一组程序,它们为两地的文件传送和远程执行命令提供某种机制或功能。功能在某种程度上与机制独立并且通过相互约定实现。这种约定建立了通讯链路。链路上传输的细节是每一边单独建立的。两台机器有这样一种关系:其中一台机接收另一台机的文件并远程执行命令,而另一台机只能发送文件而不能执行远程命令。UUCP 的一端可以接受任何其它请求的连接。在一些公众熟悉的 UUCP 端允许一些未知的机器连机而收到许多电子邮件,但却无法给它们作出答复,因为它们是未知一方。这使大多数管理员们很伤脑筋。

不同的地方行驶不同的政策使网络成员的设想难以实现。对于一个 UUCP 邻居,你能考虑的是,你可以发文件到公共缓冲池,你可以远程地执行 rmail 命令,这样就能发文件给邻居用户。即使是这样表面上标准的相互作用也有可能是不允许的。一个 UUCP 关系是两台计算机上两个管理员间的关系。UUCP 被默认为一对一网络。当多个系统管理员同意所有机器都有通行权时, UUCP 就成为一个通用的网络。

uucp 是一条命令而 UUCP 是一个软件包。UUCP 由有多个命令构成。最基本的就是 uucp,它把文件从一台机器传到另一台机器而这台机器既可是本地的也可是远程的。uux 请求一条命令在另一台(远程)机器上执行。命令可以是单程序也可以是一程序序列(可以文件作参数),其中以执行程序机器名作程序的前缀。这些命令不是交互式的,即当 uux 结束时请求的动作并不一定被执行。它们会被缓冲,有可能立即行动也有可能等一段无法确定的时间才执行。这与管理系统负荷的策略、经济上对连接另一系统的考虑,以及有没有可连接的资源等有关。UUCP 互连的主要类型有专线直连、数据 PBX 线、Internet、拨号电话线。在远端建立起连接以前,缓冲请求是不会激活的。在这里,本地机是被动的。

当用户发出 UUCP 命令时,文件就会在缓冲子目录(通常是 /usr/spool/uucp)建立,它包含有控制、数据方面的信息。但如果 uucp 命令中不指定把文件拷贝到缓冲子目录,数据文件是不会被建立的。控制文件说明被拷贝的文件的实际路径名。在缓冲子目录中还有为每

一台远程机器开设的子目录。在较老的版本中所有文件都在缓冲子目录里本身建立,这在文件的数目很多时,机器就要花很多时间在该子目录里查找了。Berkeley 版本用 Truscott 的方法把控制文件存到各子目录里。这样就省去了查询与通信无关的机器的文件。其实在 Berkeley 系统里,情况并不太坏,因为文件在目录里不是线性存储的。目录族已经大大改善了存取的效果。

UUCP 命令调用 uucico(UNIX-to-UNIX copy in copy out)执行实际的传送。如果由于某种原因(时间不正确、没有资源等)uucico 失败,当 cron 周期地调用到 uucico 时还会重试。uucico 由文件 Systems(在某些版本是 L.sys,但都在子目录 /usr/lib/uucp 里)决定连接机制。然后由文件 Devices(L-设备)里选择适当的设备。它打开设备并用 Systems 文件里的信息连接远端。在拨号的情况下它打开自动呼叫装置并拨号。对于数据型 PBX,它打开一条线路并为 PBX 提供远程系统的名称和地址。对于 Internet 连接,它用 TCP/IP 连接远处,对于专用线,其打开就更简单了。很多版本的 open()子程序大都有设置速度、模式的参数。它们可以定义与设备的交互作用,如发送一换行符到数据 PBX 以引起注意。这些设备交互作用的定义存于文件 Dialers 中。一旦与远程机器的连接建立,两端通常就进入对话,并使远端的机器也执行 uucico。对话是典型的注册过程。此过程在 Systems 文件里规定,如同一张等待/发送命令表,如

```
login:uucp Password:sigplip
```

uucico 将等待远端发送 login:字符串,一旦收到它就发 uucp,然后远端发 Password:;它又发 sigplip。

当注册时,UUCP 调用的外壳不是交互命令的外壳而是 uucico。本地系统所发的对话命令总是以回车符结尾的。特殊字符能使过程终断或表示延时等。它们被列在表 1-1 上。

一旦两机器以 uucico 通信,工作就开始了。uucico 工作在主方式还是在从方式取决于调用它时所选的参数。特别当以一个 uucp 或 uux 命令调用 uucico 时或由 cron 调用时,-rl 标志表明是主方式。而当 uucico 由 uucp 注册过程调用就不需给任何标志,这是工作在从方式。本讲后有这些交互作用的详细例子。简而言之,连接远处的主机用-s<机器名>参数。一旦联络建立,开始注册,远端机的 uucico 被调用。该 uucico(从机上的)发送包括机器名的标识信息。主机认可连通正确的机器。然后它送机器名、序列号(远端机用于校验其标识用)。uucico 的调试标志。远端机对在 UUCP 缓冲子目录里的 Admin/audit 作校验。然后主机从从机送来的有效的协议表中选择一条,一个循环便开始。主机搜查在缓冲子目录里属于从机的控制文件。如要发送一个文

件主机通知远端目的地。远地机检查 Permissions 文件看这是否可行。Permissions 里有每台机里可以存放文件的子目录。存取许可由注册或机器名来决定,这要取决于连接属性(是呼叫还是被叫)。远端回答传送是否可以进行。主机是否发送文件取决于该回答。当所有控制文件都执行完后,主机就没有什么工作要做了,它会通过向从机以挂机信息而表明这点。这样从机可以确认挂机结束对话,也可能因为远端还有工作未完而不能挂机。如果挂机被拒绝两个 uucico 便调换角色,本地端变为从机远端变为主机。然后在新的主/从关系中又开始传送的循环。这样不断地变换角色,直至两端的工作都结束。

现在介绍 uuxqt。uucp 和 uux 为远端建立了控制及数据文件。但是当 uux 产生的数据文件送达远端时,它们在到达时就变成以 x 为前缀的执行文件。每次 uucico 与另一台机结束对话时,它都调用 uuxqt,它是专门处理执行文件的。有几种原因使 uuxqt 不能完成其工作。在某一时间也许所有的文件都不存在。或者它被配置成在某些条件,如系统装载时不执行。或者只允许有限数量的 uuxqts 同时运行(避免负载太重而使系统停止)。因而 uuxqt 可以被 cron 周期性地调用。uuxqt 是通过执行文件的调用而工作的。执文件中含有运行命令以及有可能是从远端传来的所需文件。它也要说明如何把执行状态报告给调用者。有很多可选择的方式报告命令执行的成败、输入命令的回送(以便可以看到它为什么失败)。

我们已经介绍了 UUCP 的基本骨架。它们是用户命令 uucp 和 uux 以及支撑程序 uucico 和 uuxqt。UUCP 还有好几个命令提供给用户和管理员。它们将在以下各节中与这四个程序一起讨论。但这四个程序是最关键的。它们是一个有效 UUCP 系统的最小集合。

1.4 硬件要求

硬件要求对于 UUCP 的安装来说是很少的。这是 UUCP 的最重要特征之一(再有就是独立于内核),这使得它广为应用。除了通常用户所有的 tty-端口或调制解调器,UUCP 并不需要更多的要求。被动端只需另一系统的主动接触。一台机器只要它能接受接触,它就是完备的一端。这可以最简单地通过专线接连接两台机而得到。也可以一个口一个方向地连接(一个注册用一个被注册用)。当然也有程序支持一个口两个方向用(直接或 modem)。但当 UUCP 呼叫另一台机器时,需有特别的 getty。UUCP 可以使用几乎所有支持全双工通信的任何硬件,这包括双绞线、拨号 modem、自动呼叫单元 modem,以太网板和数据 PBX。

我们已经提到 Systems、Devices 和 Dialers 文件,

让我们对它们作进一步的考察。首先可以通过好几种设备或同一个设备不同的路径(如几个电话号码)联系一个远程系统。所用的机制取决于好几种因素。一天当中的使用时间可以取决于不同时间电话或数据载体服务的收费率。也许应在不同的时间使用不同的方法。如通过由用户拨号的同一 modem 使用远端系统,那么远端系统的管理员就可以建议不要在高峰期使用,以免 modem 阻塞。这些限制都在 Systems 文件中有说明。以下是例子:

```
compunerd Any TCP,c Any compunerd
compunerd SaSu,WK 700-0900 Develcon 9600
compunerd in:uucp
compunerd Any ACU 1200 boston 555827 in:uucp
compunerd SaSu,wk 2300-1800 ACU 1200 boston
6666373 in:uucp
```

第一字段是远端系统名。第二字段是呼叫设备的时间,第三字段是设备的类型,第四字段是速率,第五字段是电话号码,最后是注册序列。呼叫时间一般采取如下形式<星期几><小时范围>,...[<重试时间>]

星期几用英文打头两字母的缩写。开头字母要大写。有两种特殊写法:

Wk 表示 MoTuWeThFr 而 Any 表示 SaSuWk。时间用 24 小时制,开始时间和结束时间用连字号分开。如不写出时间则假定为 0000-2359。可以有几个日期/时间的条目,它们之间用逗号(,)隔开。时间范围可以跨越次日,如 1700-0800。最后,分号(;)引入一个子段,规定在通信失败时下一次重试前的最小等待时间(以分钟计)。

本地系统管理员定义设备类型(第 3 字段)。这是一个类名用以在 Devices 文件中寻找条目。在那列有原生组的每一个成员。如我们看到有 Develcom、TCP 和 ACU 类。假定有 3 个 tty 线连接到一个 Develcon 数据交换机,有两条线连接到自动拨号 modem,设备条目就可表示为。

```
Develcom tty10-Any develcon\D
Develcon tty11-Any develcon\D
develcon tty12-Any develcon\D
ACUtty14-1200 penril\D
ACUtty13-1200vadic\D
TCP TCP64 uucp TCP\D
```

头一个字段是类名,第 2 个是联系类名的实际设备(在 /dev 目录内),第三是发往设备函数的任意字符串。在 TCP 条目中的 64 供 tcpcall() 函数内部使用,表示连接的服务器数。(对某些系统 64 是缺省值而 540 用于大多数 Berkeley 系统)。第四个字段也是由内部函数使用,

通常表示线路速率。TCP 设备表示服务器名、DN / TTY 的组合并且规定 DN 设备(如 cuao)。第五个字段是与拨号文件对应的字符串。余下的字段连同“电话号码”一起送到设备,或者象 Systems 文件的第五字段那样用 \D 或者经由 Dialcodes 文件那样用 \T 转换。不使用的字段要用连字号填入。Dialcodes 用于扩展或翻译电话码前缀的字母串。如:

boston 91617

将使 boston 代替 91617。管理员在 Dialcodes 文件中指明得到 PBX 外线的特别拨号(如 9)或最经济的通路拨号。下面是与本例有关的 Dialers 条目:

```
penril=W-P""\d > s\p9\c)-W\ p\ r\
ds\p9\c-)y\c:\E\TP>9\c OK
vadic=K-K ""\005\p -\005 \p -\005
\p-D\p BER?\E\T\c\r\c LINE
develcon""""\r\c est:\007\E\D\e\ 007
```

第一个字段是对应线路的关键字。第二字段是表示对设备操作的字符串。由于历史的原因,-DEC 801 呼叫设备的普及先于自动拨号 modem,等号和连字号通常和电话号码混在一起表示等待拨号音(Wait for dial-tone)和暂停(pause)。剩下的字段是类似于 Systems 文件里注册序列(即 expect / sent,等待 / 发送,但开始的次序有所不同)的对话底稿。这里以 send 开始。在 Systems 里以 expect 开始。表 1-1 所列的 Systems 文件的特殊字符在这里也有效。 \D 和 \T 在文件 Devices 里有描述,说明电话号码是否要用 Dialcodes 作翻译。penril 以外的序列应该也是明显的。正象 Systems 文件里的注册文本,每一个 send 命令用隐含的回车符结尾,除非以 \C 作补充。注意在 expect 序列中连字符表示如果所期待的信号没到,那么就发送连字符后面的字符串,并期待接收下一连字符后面的字符串。这对 Systems 文件的注册序列也同样适用。在表 1-1 中所列的合法字符并非全都对文件 Systems 和 Dialers 有意义,但含义都是相同的。penril 例子有点复杂但演示了 Dialers 的能力。

("")表示等 2 秒的延时,然后是隐含的回车符。等待 > 指示符 s 之后跟着是 0.25 到 0.5 秒的暂停,接着是 9,之后没有回车,等待)。数字寄存器 9 被设置。penril 被期待以“Sure?(Y / N)”作答复。如果在超时之内(可现场设置,通常约 45 秒)字符没收到,那么一个 W 接着暂停,回车,延时,一个 s,暂停,一个 9(不回车)。然后等待冒号:(penril 说“No:”)。然后送翻译后的电话号码,然后是 P 和回车。 \E 选择字符回送使在发送下一个字符前检查该字符是否发送成功。然后等待 >,然后送 9(按第九号寄存器的号码)。最后等待字符串“OK”,它说明号码已拨通,远端机作出回答。(待续)

单独使用 213

打印驱动程序的方法

武汉大学计算机科学系 汤 庸

使用过 CCBIOS2.13 系列汉字系统的用户都会体会到其打印功能的灵活、方便,打印字形字体美观多样,易于在程序中实现,尤其适用于数据库报表打印。但是 213 系统也有其不足之处。例如,在汉字 FOXBASE 中每个菜单彩条中间出现两条平行的黑线,使屏幕的美感受到影响;又如,用 C 语言编程不能直接使用库函数 initgraph() 进行图形初始化,这需要用户重新编写图形初始化函数。笔者进行了在非 213 汉字环境中独立使用 213 打印驱动程序的尝试,获得了成功,并已用于多个大型 MIS 中,产生了非常好的效果。

下面以 213H 的打印驱动程序为例,介绍 213 打印驱动程序的独立使用一个具体方法。

1、建立 213P 目录,装入下列 213H 系列文件

prtacom	打印机汉字驱动程序
file16b.com	读 16 点阵字库模块(打印用)
file24a.com	读 24×24 字库模块
file40a.com	读 40×40 字库模块
zf24.com	24 点阵字符字体选择程序
hzk16	16×16 点阵字库
hzk24s	24 点阵宋体字库
hzk24f	24 点阵仿宋体字库
hzk24h	24 点阵黑体字库
hzk24t	24 点阵图形字符库
hzk24k	24 点阵楷体字库

以上文件共占存储空间 2.9M 字节,只需两片 1.2M 软盘即可容纳。

2、建立 213P.BAT 文件

```
cd\213p
prta a
file16b
file24a lsfhk
file40a lsfhk
zf24 3
cd\
```

这样,在非 213 汉字环境,如西文 DOS、UCDOS、WPS 等环境下,运行 213P.BAT 即可完成 213 打印驱动。值得说明是在 UCDOS 或 WPS 汉字环境中必须取消打印设置,例如在 UCDOS2.0 中可以用 UCDOS/K 将打印制减开关置为 ON。

DOS 高低版本的硬盘共享

重庆后勤工程学院 葛增超

一、目的:

随着软件工程的不断发展,计算机的硬盘空间的需求越来越大,常常达到几十兆甚至上百兆,笔者最近碰到一个三维有限元程序,所需硬盘空间竟达七十兆字节。由于人们常用的 DOS3.3 版本所管理的硬盘空间有限,最高达到 33M 字节,难以满足要求,高版本的 DOS 则很好地解决了这一问题。如 MS-DOS5.0,最大可建立二千兆硬盘分区。但高版本的 DOS 某些功能尚不够完善,使用中存在问题,如 MS-DOS5.0 安装后,启动 WPS 桌面轻印刷排版系统即发生死机现象;大多数的汉字系统都难以正常运行。为此,笔者通过在硬盘上建立两种不同 DOS 版本分区的方法解决了这一问题。

二、建立方法:

1、了解硬盘 DOS 分区。硬盘上能建立两种 DOS 分区:基本 DOS 分区和扩展 DOS 分区。基本 DOS 分区含有运行 DOS 所必须的系统文件 IO.SYS、MSDOS.SYS 及 COMMAND.COM,如果要从硬盘上启动 DOS,磁盘上就必须有基本 DOS 分区,基本 DOS 分区一般设置在 C 盘。扩展 DOS 分区可以存储其它非系统文件,在扩展 DOS 分区上可建立最多 23 个逻辑驱动器,各逻辑驱动器可作为单独磁盘驱动器使用。

2、建立高版本 DOS 区。首先根据所需硬盘空间大小用 FDISK (高低版本均可) 删除原来扩展 DOS 分区中的一个或几个逻辑驱动器,保留基本 DOS 分区(一般为 C 盘)和其余逻辑驱动器不动。比如:原硬盘有 110M 字节,分为 C、D、E 和 F 四个驱动器,其空间分别为:33M、30M 和 17M,现需一磁盘空间为 47M,则可删除 E 和 F 驱动器, C、D 驱动器不动。然后用高版本的 FDISK 建立一足够尺寸硬盘空间的逻辑驱动器,(如 E, 47M)。最后退出 FDISK,将高版本 DOS 系统盘插入 A 驱,重新启动系统,用 FORMAT 命令格式化该逻辑驱动器,即为高版本的 DOS 区。该区只有用高版本系统软盘启动后方可进入,用低版 DOS 启动则不识别该驱动器。

3、两点说明。(1) 使用 FDISK 删除逻辑驱动器前必须确信该逻辑驱动器中的信息是无用的或已做了备份;(2) 以上使用 FDISK 不会影响基本 DOS 分区和其它逻辑驱动器中的内容。

DOS级通用光条式菜单程序设计与实现

重庆大学计算机系 符云清

菜单,特别是光条式菜单,作为一种良好的用户界面,深受广大计算机软件开发者和用户欢迎。因而广泛应用于各种应用系统中。但普通菜单都是针对某一具体应用,用特定的语言编写,一旦编制后,菜单显示及转移执行都是确定的,这使菜单程序的通用性受限。如硬盘上有以下子目录: C:\TC, C:\PCT, C:\CCDOS, 我们希望开机后,能通过菜单选择进入相应子目录,我们可预先按要求编一个菜单程序,并置于 AUTOEXEC.BAT 中,则可达到上述目的。但当硬盘上的目录改变后,则菜单程序又得重新编写,这样既麻烦又花时间,如果我们能建立一个 DOS 级通用菜单,它能接收用户输入的菜单参数并显示菜单,同时能将用户对菜单的选择传给 DOS,在 DOS 中根据选择去执行相应程序或应用软件,这样则达到通用菜单的目的,从而既省时又省力,提高效率。

程序设计的关键是如何将用户对菜单的选择传递给 DOS。众所周知, DOS 的功能调用 4CH 以终止当前程序的执行,并把控制权交给调用者。该功能允许被终止的程序传送一返回代码给调用它的进程。若被终止的程序是用 DOS 的 EXEC 功能调入的,调用程序可用请求功能 4DH 来取得该代码;若被终止的程序由 DOS 调用,则返回代码可用 DOS 的 IF 和 ERRORLEVEL 命令确定。所以我们可用 DOS 的 ERRORLEVEL 来传递用户对菜单的选择。在 TUBRO C2.0 中,可以很方便地使用函数 EXIT (cexitno) 来实现。并在批命令文件中用 IF ERRORLEVEL = <值> 来实现程序分支转移。

附录中给出程序 MENU.C, 编译连接成 MENU.EXE 后,在批命令文件中以如下方式使用: MENU <菜单项 1\$>.....<菜单项 n\$> <显示行位置> <显示列位置> <显示横向宽> 即可。选中菜单第 1 项,返回 ERRORLEVEL 值为 0,第 2 项为 1,依此类推。该程序可由用户确定菜单显示内容,并在屏上任意位置显示彩色光条式菜单(不能显示汉字,稍加修改即可)。由于 TUBRO C 中命令行参数以空格为限,而菜单中每一项中亦有空格,故采用在每一个菜单项后加“\$”来识别。

为进一步说明问题,现举一例详细说明:设硬盘原有如下子目录: C:\PCSHELL, C:\TANGO,

C:\WPSII, 则我们编写如下 AUTOEXEC.BAT(见附录部分),则可实现光条式菜单选择进入相应子目录,当硬盘上目录改变时,我们只需改变菜单命令行参数,而不必重新编制菜单程序,如硬盘上目录变为: C:\TC, C:\DM, C:\DOS, 编写的批命令文件如下(见附录)

```
A>type a:menu.c
#include <stdio.h>
#include <stdlib.h>
#include <conio.h>
#define UP 0x4800
#define DOWN 0x5000
#define RETURN 0x1c0d
char menu[10][80];
int items, len, row=1, col=5;
main(int argc, char **argv)
{
    int choice, i, j, l, key;
    /* to get all menu items */
    row=atoi(argv[argc-3]);
    col=atoi(argv[argc-2]);
    len=atoi(argv[argc-1]);
    for (i=1, items=0, l=0; i<argc-3; i++){
        j=0;
        while (argv[i][j] != '\0'){
            if(argv[i][j] != '$') menu[items][l]=argv[i][j];
            else {
                while(l<len-1) menu[items][l++]=' ';
                menu[items][l]='\0';
                items++, l=-1;
            }
            ++l, ++j;
        }
        menu[items][l++]=' ';
    }
    /* display menu */
    textbackground(BLUE);
    textcolor(YELLOW);
    for(i=0, i<items; i++){
        gotoxy(col, row+i);
        cprintf("%s", menu[i]);
    }
    i=(len-32)/2+col;
    textcolor(WHITE);
    textbackground(BLUE);
    gotoxy(i, items+row);
    cprintf("ARROW to move, RETURN to select\n");
    /* cursor move and return choice to DOS */
    choice=0;
    while(1){
        textcolor(RED);
```

广告索引

```

textbackground(WHILE);
gotoxy(col,choice+row);
cprintf("%s", menu[choice]);
do {
    key=bikey(0);
} while( key! =UP&& KEY! =DOWN&& key! =RETURN);
textcolor(YELLOW);
textbackground(BLUE);
gotoxy(col,choice+row);
cprintf("%s", menu[choice]);
switch(key) {
    case RETURN:
        exit(choice); /*return choice to DOS*/
    case UP:
        if(choice==0) choice=items-1;
        else --choice;
        break;
    case DOWN:
        if(choice==items-1) choice=0;
        else ++choice;
}
}

```

```

A>type a:autoexec.bat
cls
echo off
menu 1--Enter Pcsell$ 2--Enter Tango$ 3--Enter Wpsii$
3 20 31
if ERRORLEVEL=2 goto 12
if ERRORLEVEL=1 goto 11
if ERRORLEVEL=1 goto 10
:12
cd\wpsii
goto exit
:11
cd\tango
goto exit
:10
cd\pcshell
:exit
echo on

```

```

A>type b:autoexec.bat
;when directories changed, AUTOEXEC.BAT is rewrited as
follow;
cls
echo off
menu 1--Enter Tubro c$ 2--Enter Dos 3.3$ 3--Disk Manage
r$ 3 20 30
if ERRORLEVEL=2 goto 12
if ERRORLEVEL=1 goto 11
if ERRORLEVEL=0 goto 10
:12
cd\dm
goto exit
:11
cd\dos
gotoexit
:10
cd\tc
:exit
echo on

```

100

1. 广州白云山电源设备厂
2. 昌华计算机有限公司
3. 广州白云山电源设备厂
4. 广州市科教电脑设备厂
5. 广州电子设备公司
6. 广州市海谊电子仪器实业公司
7. 华力科技开发公司
8. 赛宝星河
9. 星辰电脑
10. 天河电子
11. 广州市新一代科技发展公司
12. 广州市声乐电脑商行
13. 三联电脑贸易部
14. 特强(广州)电子有限公司
15. 广州科星电子技术公司
16. 广州全通计算机公司
17. 广州袖珍计算机技术服务中心经营部
18. 艾西显示设备有限公司
19. 华粤电子系统公司
20. 清华大学科学馆

101

·小词典·小词典·小词典·小词典·小词典·小词典·小词典·

平时,我们在计算机使用过程中,经常会遇到一些缩写的计算机术语,如果结合其英文本义,则可能会更易理解一些。下面就摘录了几个术语的中、文英文解释,仅供参考。

- CAD**—— 计算机辅助设计(Computer-Aided Design)
- CAI**—— 计算机辅助教学(Computer-Assisted Instruction)
- CAM**—— 计算机辅助管理(Computer-Aided Management)
- 按 内容 编 址 存 储 器 (Content - Addoressable Memory)
- CPU**—— 中央处理机(Central Processing Unit)
- DMA**—— 直接存储器存取(Direct Memory Access)
- DOS**—— 磁盘操作系统(Disk Operating System)
- IC**—— 指令计数器(Instruction Counter)
- 集成电路(Integrated Circuit)
- MIS**—— 管理信息系统(Management Information System)

102

IBM PC/XT/AT到IBM AS/400 的汉字变换方法

广州市机关办公信息自动化中心 李建明

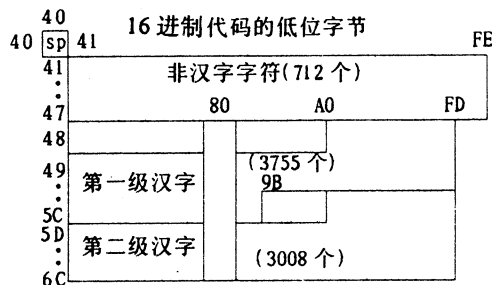
【摘要】本文阐述了 IBM PC/XT/AT 微机及兼容机与 IBM AS/400 小型机之间汉字变换程序设计方法的关键,文中介绍了两类机型汉字机内码的差异及转换方法,作者在 IBM AS/400 小型机上用 PASCAL 语言编写了转换程序,成功地实现了汉字的变换,本文介绍了转换程序的框图。

一、汉字转换的目的

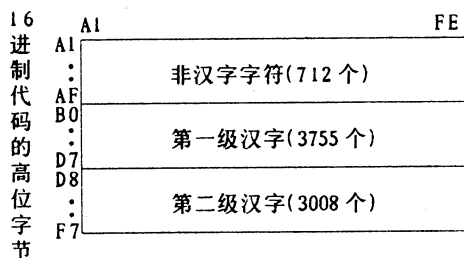
IBM PC/XT/AT 类微机在国内已得到广泛的应用,IBM AS/400 小型机在一九八八年推出以来,也显示出强大的生命力,在国内装机数也不断增加,得到越来越广泛的应用。这两类机型都配备有汉字系统,IBM AS/400 的 PC/SUPPORT 也提供了微机文件与 AS/400 文件的相互传送功能,但是由于两类机型对汉字机内码的设计有所不同,致使两类机型的汉字数据不能通用,用户现在微机输入的汉字数据不能为 AS/400 所用。为解决这个问题,本文介绍一种 IBM PC/XT/AT 类微机中文数据转换为 IBM AS/400 小型机中文数据的方法。

二、汉字内码的分析

IBM PC/XT/AT 与 IBM AS/400 都采用我国 1981 年公布的《通讯用汉字字符集(基本集)及其交换码标准》GB2312-80,此标准归纳了 6763 个基本字,其中使用频度较高的一级汉字 3755 个、二级汉字 3008 个。此外还包括西文、图形符号 700 多个,自定义汉字和符号约 800 多个,国标区位码把它分成 94 区,每 94 位。两类机型的内码分布区域存在较大差异,下面是两类机型的非汉字字符、第一级汉字、第二级汉字内码的分布区域。



AS/400 主机系统汉字字符集代码区



IBM PC/XT/AT 微机系统汉字字符集代码区

1、IBM PC/XT/AT 的汉字编码特点

在此类机器的 CCDOS 汉字系统内,把 GB 2312-80 规定的国标码的两个字节最高位都置“1”形成汉字的机内码。从而与高位为“0”的单字节西文码相区分。所以此类机器的汉字机内码的高、低位字节均从 A1H 至 FEH 共 94 个字节,与国标区位的区和位一一对应。如图所示, A1A1-AFFE 为非汉字字符区, B0A1-D7FE 为第一级汉字区, D8A1-F7FE 为第二级汉字区。

2、IBM AS/400 的汉字机内码特点

此类机器的汉字系统并没有把两个字节的最高位置“1”,汉字与西文字符仅用移出、移入控制字符 X'、0E'、X'、0F',将汉字字段括起来加以区分,如“A”表示西文字符 A。而“X'0E'、X'0F'”表示中文字符 A, 4141-47FE 为非汉字字符区, 48A0-5C9A 为第一级汉字区, 5CA0-6C9F 为第二级汉字区,注意 80H 被剔除在低位字节排列范围以外。其中每一个高位字节下可存放国标区位码的两个区共 $2 \times 94 = 188$ 个字(41H 至 FDH 可存放 189 个字,剔除 80H 后刚好可存

放 188 字)。

三、汉字内码的对应关系及变换方法

1. 从 IBM 5550 编码手册发现, 对于非汉字字符即国标区位码 1 区至 15 区的西文图形符号, 很难找出 IBM PC/XT/AT 与 IBM AS/400 的机器内码的对应关系, 比较复杂, 对这部分字符集, 我们可采用查表法实现内码变换, 具体方法是, 按区位码顺序把非汉字字符集变成一个如下的一维表:

1	100 101	200 201	800 901	1000 1001
一区字符	二区字符	……	九区字符	……

这样我们就可用指针 POINT 指向相应的字符。

$$\text{point} = \frac{(X1 - 160) * 100 + (X2 - 160) - 100}{\text{位码}} - 100$$

区码

其中 X1 是 IBM PC/XT/AT 汉字机内码的高位, X2 是低位。

2. 对于第一级汉字和第二级汉字即国标区位码 16 区至 87 区的字符集, 我们可找出 IBM PC/XT/AT 与 IBM AS/400 的机器内码对应关系, PC 的内码高字节从 A0H 至 F7H 对应 AS/400 内码高字节 48H 至 6CH, 中间呈线性变化。每个 AS/400 内码的高位字节对应两个 PC 内码的高字节 (48H 与 6CH 只对应一个)。由于 AS/400 每个高字节对应两个区共 188 个位码, 因此, PC 的内码低字节 A1H 至 FEH 可能和 AS/400 的内码低字节 41H-9FH 或 A0H-FDH 相对应, 其中偶数区对应 AS/400 内码低字节的 A0H-FDH 段, 奇数区对应 AS/400 内码低字节的 41H-9FH 段, 从以上分析可导出计算公式如下: X1、X2 为 PC 机的汉字机内码高字节和低字节, Y1、Y2 代表 AS/400 汉字机内码的高字节和低字节。

(1) 高字节变换

$$Y1 = ((X1 - 175) \text{ DIV } 2) + 72$$

(2) 低字节变换

当 X1 为奇数时

$$Y2 = (X2 - 161) + 65 = X2 - 96$$

若 $Y2 \geq 128$ (80H) 则 $Y2 = Y2 + 1$

当 X1 为偶数时

$$Y2 = (X2 - 161) + 160 = X2 - 1$$

3. 移出、移入控制符 X'0E、X'0F 的插入

AS/400 的汉字字段由一对移出控制符 (0EH) 和移入控制符 (0FH) 括起来, 要将 PC 中混合有单字节的西文字符和双字节的中文字符的一段文本转换为 AS/400 的文本, 除需转换内码外, 还需在双字节的

中文字符前加移出控制符 (0EH), 在其后加移入控制符 (0FH), 在混合字段中单字节字符与双字节字符的排列形式有下列四种:

- (1) 单字节 (0EH) 双字节 (0FH)
- (2) (0EH) 双字节 (0FH) 单字节
- (3) 单字节 (0EH) 双字节 (0FH) 单字节
- (4) (0EH) 双字节 (0FH) 单字节 (0EH) 双字节 (0FH)

从以上排列形式可看出:

- (1) 变化到双字节则插入 0EH;
- (2) 变化到单字节并且前面已插入 0EH 时则插入 0FH;

若用 $0E = 1$ 表示已插入 0EH, $0E = 0$ 表示未插入 0EH, $SZ = 1$ 表示当前字符是双字节字符, 非双字节字符时 SZ 置 0, $DZ = 1$ 表示当前字符是单字节字符, 非单字节字符时 DZ 置 0。

则在 SZ 、 DZ 置零前, $BH = SZ * DZ$ 为 1 时表示有变化, 为 0 时表示无变化。

这样:

当 $SZ = 1$ and $BH = 1$ (即变化到双字节) 时, 在当前字符前插入 0EH; 并将 $0E$ 置 1。

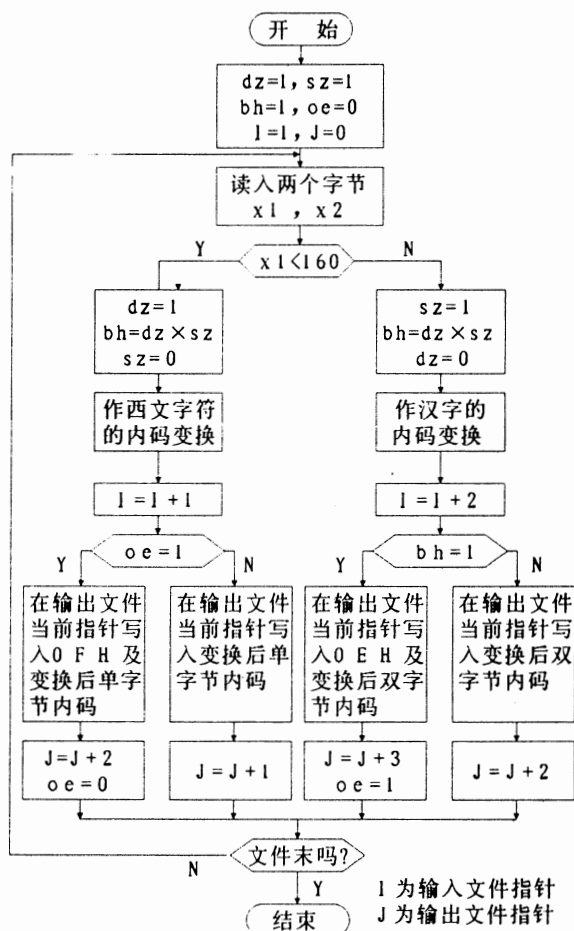
当 $DZ = 1$ and $BH = 1$ and $0E = 1$ (即变化到单字节并前面已插入 0EH) 时, 在当前字符前插入 0FH, 并将 $0E$ 置 0。

四、单字节字符集的转换

在 IBM PC 中, 西文字符用单字节 ASCII 码表示, 而 IBM AS/400 中, 西文字符用单字节 EBCDIC 码表示, 下面是 ASCII 码到 EBCDIC 码的转换表 (部分):

		16 进制代码的高 4 位											
16 进 制 代 码 的 低 4 位		0	1	2	3	4	5	6	7	8	9	A	B
	0			40	F0	7C	D7	79	97	4A			
	1			5A	F1	C1	D8	81	98				
	2			7F	F2	C2	D9	82	99				
	3			7B	F3	C3	E2	83	A2				
	4			E0	F4	C4	E3	84	A3				
	5			6C	F5	C5	E4	85	A4				
	6			50	F6	C6	E5	86	A5				
	7			7D	F7	C7	E6	87	A6				
	8			4D	F8	C8	E7	88	A7				
	9			5D	F9	C9	E8	89	A8				
	A			5C	7A	D1	E9	91	A9				
	B			4E	5E	D2	A0	92	C0				
	C			6B	4C	D3	5B	93	4F				
	D			60	7E	D4	B0	94	D0				
	E			4B	6E	D5	5F	95	A1				
	F			61	6F	D6	6D	96					

五、变换程序框图



六、结束语

本文提供的是一种转换算法，可根据不同的应用，用各种编程语言实现。对于象 DBASE III 的 DBF 文件的转换，应跳过 DBF 文件的表头信息，以免将表头信息误认为汉字内码。此外，要完成整个传送和转换过程，还涉及到 IBM AS/400 的 PC/SUPPORT、OS/400 和 DATABASE 等技术，但这不在本文讨论范围，请参考有关资料。作者已使用此方法将以前在微机输入的大量文件目录（DBASE III 的 DBF 文件）和文件全文（文本文件）传入 AS/400 的数据库中，节省了大量人力和时间，取得很好的应用效果。参照这种方法，读者也可得出 IBM AS/400 内码到 IBM PC/XT/AT 内码的转换算法，本文不再论述。

参考文献：

- IBM 5295-GP3 中文工作站用户指南
- IBM AS/400 PASCAL USER'S GUIDE
- IBM AS/400 PASCAL REFERENCE
- IBM Multistation 汉字编码手册

103

DOS 下王码 5.0 与

2.13H 共存

广东惠东职中三 (5) 电子班 陈晓乐

在目前的汉字系统中，CCBIOS2.13H 和王码 5.0 以其各自的特点及卓越的功能赢得了众多的用户。基于这两种汉字系统有各自不同的功能，在使用时都根据不同的需要选用不同的汉字系统。但是，由于 2.13H 启动时需要设备驱动程序 ANSI.SYS 的支持，如果引导 DOS 时，系统盘的 CONFIG.SYS 文件中，第一句配置命令不是 DEVICE=ANSI.SYS，那么启动 2.13H 时便会死机；而启动王码 5.0 时，若 CONFIG.SYS 有 DEVICE=ANSI.SYS 命令存在，进入王码后，当光标到达屏幕最底行时，屏幕便下再滚动，所有显示数据都停留在最底行上，严重影响使用（2.13H 在按下 CTRL-F5 退出后也出现这种情况）。这样，启动这两种汉字系统就需要两张不同配置的 DOS 系统盘，甚至有硬盘的用户也只能启动其中之一一种汉字系统，另一种要从软盘上启动，并且每次启动只能使用一种，这就造成了极大的不便。

造成上述情况的原因是对 ANSI.SYS 的使用，2.13H 在使用 ANSI.SYS 时，修改了它的光标定位范围，使它能在汉字显示中发挥作用；王码 5.0 却不需要 ANSI.SYS 的支持，所以王码没有修改 ANSI.SYS。因此，我们只需在 DOS 启动后，用程序修改驻留在内存的 ANSI.SYS 的显示行数，使它能在不同的环境中发挥作用。即进入王码前改成显示 24 行，退出汉字系统后又改回原来的 25 行。下面两个程序就能实现这个目的。

ANSI24.COM

```

PUSH DS
MOV AX, 70
MOV DS, AX
MOV AX, 1817
MOV [971C], AH
MOV [9723], AL
POP DS
INT 20

```

ANSI25.COM

```

PUSH DS
MOV AX, 70
MOV DS, AX
MOV AX, 1918
MOV [971C], AH
MOV [9723], AL
POP DS
INT 20

```

程序在 DEBUG 下键入，ANSI24.COM 在使用王码时行执行，可加在王码的启动批命令中，ANSI25.COM 在退出汉字系统后恢复屏幕用。有了这两个程序，就可以方便地更换使用两种汉字系统而不必重新启动 DOS 了。

程序均在 AT286，DOS3.3，高分单显上通过，使用效果良好。

104

PC/AT 微机开关电源的故障定位法 及维修实例

中山大学电子系 郑钟峰

在微机维修工作中,微机内置开关电源是一个易损件,特别是随着各种 UPS 电源的应用,对于方波输出的 UPS,若是使用不当(过于满载或空载,最好为标称满载功率的 60%~70%),极易损坏微机电源组件。微机电源虽然体积小,元器件不多,但由于其保护电路众多,一旦出了故障,往往不知如何对故障进行定位,而故障定位是维修的首要工作。现在在众多电脑刊物上见到的都是一些维修实例的介绍,对于某种特定的故障现象是可以依样画葫芦的,但这些刊物都没有系统地介绍如何根据各种故障现象进行故障定位的方法。下面根据作者的维修经验,结合一些实例重点介绍一下微机开关电源的故障定位法。

微机开关电源各部分的框图如图 1 所示:

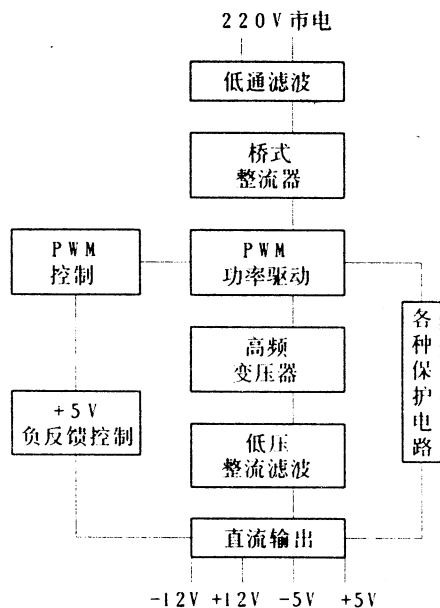


图 1

工作原理:

220V 市电输入经滤波及整流之后变成 300V 直流高压,该直流高压被送到脉宽调制器(PWM)功率转

换线路,在 PWM 控制线路控制下,变成幅值 300V 的矩形波,再经高频变压器降压及整流滤波即可输出 $\pm 12V$ 、 $\pm 5V$ 的直流稳压。调节 300V 矩形波的占宽比即可调节直流输出值,这也就是反馈稳压的主要原理。

在维修中,首先要确定是否为电源部分的故障。因为在维修中曾出现过由于软硬卡或显示卡的故障而造成电源部件进行过流保护的现象而误判为电源故障。判断的简单方法是当微机出现开机无电源指示时,在断电的情况下将微机扩展槽上的插板一块块的拔下,当拔下某一块插件而微机就能上电启动时,则故障应定位在这块插板上;而当所有的插板都拔下而故障依旧时,应判为电源(或主板,但主板一般不易坏)的故障。

当确认为电源部分的故障时,可根据其中的保险丝的情况来进行判断:

1、当保险丝爆裂时,这时可判断故障应在高频变压器的原边回路中,即图 1 中高频变压器的左边。在维修中较常见(占 80%以上)的是:①桥式整流器的整流二极管击穿。②高压滤波电容击穿。③功率驱动开关管击穿。对于这类故障,一般应该先进行静态测试,确认故障原因,不能匆忙进行通电,以免再次损坏元件。

2、若是保险丝完好无损,则故障一般在高频变压器的副边输出回路(或 PWM 控制电路,但此种情况较少),即图 1 中高频变压器的下边。在维修中,较常见的是 +5V 输出回路损坏。这是由于微机的主板、各种扩展板以及软硬盘主要是由 +5V 供电,负载最重,因而故障率也最高。最常见即为肖特基二极管损坏,引起保护电路动作造成无电压输出。

微机开关电源中 PWM 控制芯片损坏虽不常见,但是一旦怀疑它损坏之后,如何进行判别它的好坏呢?这里介绍一个方法:微机电源中 PWM 控制芯片一般用的是 TL494 (或 $\mu PC494$ 等不同厂家生产的同一类型的芯片)。在 TL494 的 12 脚上加上 +15V 直流电压,在 4 脚上加上 +0.25V 的死区电压,则在 5 脚可用示波器看到 0.4~4V 的锯齿波,在 8 脚和 11 脚可看到两串幅值均为 15V,但有 180 度相差的矩形波。若观察不到上

述波形，应判断为 TL494 损坏。

在微机电源的维修工作中，由于其特有的缺相保护电路（即 $\pm 12V$ 、 $\pm 5V$ 中任何一路或一路以上没有电压输出，则其它的几路电源也会立即进行保护）、 $+5V$ 空载保护电路（即如果微机电源不接上负载或负载电流太小，也会引起保护电路动作）、过流保护电路（即当输出回路短路时，将引起保护电路动作）等协同工作，因此在检修中若是微机开关电源不接负载的话，此时先是引起 $+5V$ 空载保护，接着又引起缺相保护电路动作，整个微机电源将无任何输出。因此，对于一台有故障的微机电源，在检修时，应在其 $+5V$ 输出端和地线之间接上阻值为 $3\sim 5$ 欧姆、功率为 $15W$ 以上的电阻作为假负载。作者是采用四只市面上常见的 5 欧姆 $5W$ 的水泥电阻，使用图 2 的连接作为一只 5 欧姆 $20W$ 的假负载，以避免空载保护电路误动作。接上假负载之后，若是故障如旧（无直流电压输出），这时可用另外一个可调稳压直流电源依次调节其输出电压为 $+5V$ 、 $-5V$ 、 $+12V$ 、 $-12V$ ，分别接在微机电源的 $+5V$ 、 $-5V$ 、 $+12V$ 、 $-12V$ 各个输出端，即标为 P8、P9 的两个插头上，插头上各条色线的分布及对应电压如图 3 所示。当接在其中某一路电压输出端而微机电源故障消失时，可判断故障应在此一路输出电压的保护电路上。

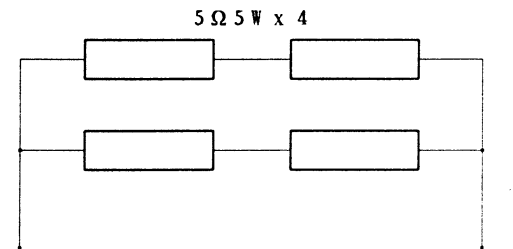


图 2

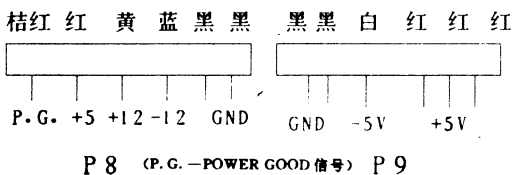


图 3

维修实例：

①一台新机使用不久，突然开关电源的滤波电容爆顶，保险丝爆裂。用万用表测量两个串连的高压滤波电容，其中爆顶的电容已变成一个电阻，桥式整流器完好。原因是此滤波电容的极性接反，开关电源使用一段时间后，极性接反的滤波电容过热爆裂，属生产厂家的质量问题。将另一极性接反但还未爆裂的电容一起拆下，换上另外两个参数相同、符合规格的电容即恢复正

常。

②微机使用一段时间以后，在开机时电源无电压输出，保险丝熔断。此时应判断故障在高频变压器的原边回路中。查桥式整流器有正常输出的 $300V$ 直流高压，检测出末级驱动功率管 2SC3039 均已损坏，并烧坏与 2SC3039 的 C、E 极并联的快速二极管 FR107。原因是用户曾经频繁的开关微机电源，而此 2SC3039 质量较差而击穿。更换损坏元件后恢复正常。

③微机电源无输出，保险丝完好。判断应为高频变压器副边回路的故障。查 TL494 无损坏。查 $+5V$ 输出端发现为肖特基二极管（作低压整流用）MBR2045 击穿损坏，更换后恢复正常。

以上所述主要是如何根据故障现象，结合开关电源的原理来定位故障部位。在定位之后，还应该根据不同的电路、不同的现象进行维修。有关这方面的内容由于篇幅所限从略。 (105)

~~~~~

## CTX-7 型显示器 故障维修一例

柳州铁路材料总厂 单昶贤

故障现象：

CTX-7 显示器开机光栅水平一条亮线，调场幅不起作用。

分析与维修：

根据故障现象初步判断为场扫描电路发生故障，显象管电路和行扫描电路工作是正常的。用万用表测行场扫描电路芯片 HA11414 的 2 脚电压  $0.07V$  低于正常值  $0.8V$ ，3 脚电压为  $2.6V$  低于正常值  $4V$ 。测场输出管 VT402 D2344 的 B 极电压为  $0.07V$  与 HA11414 的 2 脚电压相符，C 极电压为  $18V$  低于正常值  $28V$ 。用示波器查看 HA11414 的 2 脚场锯齿波输出波形和 3 脚场负反馈输入波形均有信号，但幅度不够大，查看 VT402 的 C 极无信号输出，由此判断 HA11414 是好的，VT402 没有工作，VT402 的 C 极工作电压  $28V$  是由电源  $113V$  经 R451 降压提供的，查 R451 损坏呈开路状态，标称值  $2W3.6K$  欧姆。由于 R451 损坏，VT402 工作电压不足，场输出管没有工作，造成光栅水平一条亮线的故障，换上相同规格的金属膜电阻，故障消失，恢复正常。 (106)



# AR-3240打印机使用经验

四川省泸州市经济信息中心 郑 良

Star AR-3240 高点阵多字库 24 针超高速汉字打印机以其卓越的性能深受用户青睐。我们在多年使用过程中，摸索到一些使用说明上没有或不详尽的经验技巧。特提供同行们参考，起到抛砖引玉之用。

## 一、关于右移打印

右移打印在打印文稿中经常使用，有以下三种方法：

1. (在 WS、高级语言中) 用  $\wedge[\text{n}]^*$  设置左边限，\* 用 Alt+n (先按住 Alt 键，再键入数字 n 后，放开 Alt 键) 产生 (在 BASIC 语言中可用 CHR\$(n))，可使 AR-3240 打印机右移 n 列打印。
2. (在 BASIC 语言下) 用 LPRINT TAB (n)；
3. (在 BASIC 语言下) 用 LPRINT SPACE\$(n)；

## 二、关于无级变倍、倍级放大

无级变倍、倍级放大可实现打印很多种点阵的字体，非常受欢迎。使用不当，也会出问题。正确使用方法：

1. 取消无级变倍不仅用 “ $\wedge\backslash\text{c00}$ ” 命令，还要加  $\wedge[\text{u1}]$  (或  $\wedge[\text{u4}]$ )。如： $\wedge\backslash\text{c4848}$  内…容  $\wedge\backslash\text{c00}\wedge[\text{u4}]$  (打印  $48\times 48$  点阵后，转为  $32\times 32$  点阵打印)。
2. 取消倍级放大不仅用 “ $\wedge[\text{c11}]$ ” 命令，还要加  $\wedge[\text{u1}]$ 、 $\wedge[\text{u4}]$ 。如： $\wedge[\text{u1}\wedge[\text{c33}]$  内…容  $\wedge[\text{c11}\wedge[\text{u1}\wedge[\text{u4}]$  (以  $24\times 24$  点阵为基础，纵横放大 3 倍，即  $72\times 72$  点阵，打印完后用  $\wedge[\text{c11}\wedge[\text{u1}]$  消除 ( $\wedge[\text{u1}]$  与 “内…容” 前面的  $\wedge[\text{u1}]$  一致)， $\wedge[\text{u4}]$  表示后面的用  $32\times 32$  点阵打印)。
3. 无级变倍右移方法要选上述右移方法 1、3，用 2 会出如下问题：即后面的无级变倍纵向点数不能变化 (如果实在要用 2，可用 WS 编辑状态下，无级变倍控制命令后加入  $\wedge[\text{@}]$  使打印机复位，并加上所需控制命令)。

## 三、标题居中

用大字作标题时，标题居中是经常遇到的问题。掌握不当，要靠来回加减空格实现，试若干次才行。笔者采用的如下方法，也许您会觉得非常方便：

先计算  $n:n$  (标题大字字符数) = (正文字体横向点数  $\times$  正文最大字符数) + 标题所用大字横向点数 + 1。比如正文每行用 32 (纵向)  $\times$  32 (横向) 点阵打印 56 个字符 (28 个汉字)，标题用  $48\times 48$  点阵打印，则  $n = (32\times 56) + 48 + 1 \approx 38$  (可记住这个数，以后就不必再计算了)。然后在 WS 中按 F4 再输入 n (如上面的 38)。最后将光标移至需控制行，按 Ctrl+O+C (WS 中将标题以标尺为准居中的命令) 三个键后，再将光标分别移至该行第一列和最末列，在插入状态下，写上控制码即可。这样不需试即可将所控标题居中 (无论您打印时是否右移均可实现)，避免了时间、设备和材料的浪费。

## 四、关于其它控制命令

$\langle\text{ESC}\rangle$  (CHR\$(27))、 $\langle\text{FS}\rangle$  (CHR\$(28)) 命令可分别用  $\wedge[\text{ ]}$ 、 $\wedge[\text{ \backslash}]$  代替，但必须运行联机程序 AR-3240.COM。注意控制命令的大小写。

1. ESC 3 n: 设定  $n/180^\circ$  行距
2. ESC A n: 设定  $n/60^\circ$  行距
3. ESC Q n: 设定右边限位置
4. ESC X  $n_1n_2$ : 设定左右边限位置
5. ESC l n: 设定左边限位置
6. FS 3 n: 设定  $n/360^\circ$  行距

上述命令中的  $n$ 、 $n_1$ 、 $n_2$  必须用 Alt+n、 $n_1$ 、 $n_2$  产生，不能直接写数字，参见上述第一点第 1 项。还有一些命令中的  $n$  也必须用 Alt+n 代替，这里不再举例。有必要补充一点，ESC c  $n_1n_2$  (纵向倍级放大) 中的  $n_1$ 、 $n_2$  要直接写数字；FS e  $n_1n_2$  (汉字无级变倍) 中的  $n_1$ 、 $n_2$  写法要视情况而定 (如果 FS 用 Alt+28 产生，就不能直接写数字；如果 FS 用  $\wedge[\text{ \backslash}]$  代替，就要直接写数字)。

## 激光打印机故障维修三例

四川长特计控处 杨居义

**一、故障现象：**LCS-15 激光打印机无法从纸盒内搓纸。

**分析与排除：**激光打印机的搓纸系统要完成工作是，将纸盒里的纸一张一张的送到进纸辊前，完成的时间为 0.5 秒~1



图1 搓纸辊

秒左右，换句话说也就是搓纸辊必须在 1 秒钟内将纸从纸盒内搓出，并送到进纸辊前，然后在准备搓下一张纸，周而复始，往返的工作，若在规定时间内没有完成上述搓纸工作，打印机就出现卡纸符号。因此只要搓纸系统某一部件有问题，就可能在在规定时间内完成不了这种工作，而搓纸辊是较关键的部件。我们发现由于搓纸次数太多（已达 6 千张），搓纸辊表面已磨得很光滑，想购买新的搓纸辊又觉得划不来，因而我们采用下面两种方法来解决，经试用后效果很好。

- 1、将搓纸辊表面用锯条按图 1 所示拉毛；
- 2、若方法一效果不很好，可采用在搓纸辊上绕橡皮筋。

**二、故障现象：**LDP-8 激光打印机打出来的纸是一张空白纸。

**分析与排除：**根据这种故障现象分析，造成这种故障原因的有充电、显影、转印这三方面，只要有一个方面出了问题，都可能使这类故障出现。将打印机加电，让打印纸进入打印机机体中央位置时，关掉打印机电源，打开机体检查纸上无字，而感光鼓上有字出现，说明充电、显影都正常，问题出在转印部件上，检查转印电极丝、高压都是好的，接插件处也良好，其它也看不出有什么问题，然后用万用表  $R \times 10K$  档测电极丝与电极座之间有电阻，而且只有  $200K\Omega$  左右，换一个电极座问题得到解决，后来将电极座拆下来发现，绝缘塑料已经老化，变成了导电材料，转印高压加上与机壳构成通路，转印电极丝上得不到高压，就造成了本故障现象的出现。

**三、故障现象：**LCS-15 激光打印机不能定影。

**分析与排除：**打印机不能定影，用手一摸纸上的字就掉了，这主要是定影系统出了故障。定影部分的工作原理是：在打印机加电后，主控板发给定影灯一个控制信号，使定影灯电源接通（灯管为 550W），定影辊表面温度逐步上升，当到达  $175^{\circ}\text{C} \sim 193^{\circ}\text{C}$  左右时（此温

度能使墨粉熔化），定影辊表面上的热敏电阻就检测到此温度并送到主控板，主控板收到此信号后就将定影灯电源断掉，低于此温度时定影灯电源接通，周而复始。因此只要定影系统有一个部件出了问题，此故障就可能出现。我们这台打印机是由于灯管两端的接头有一端接触不良造成打火，久而久之，上面就有很厚一层被氧化掉了，造成定影灯电源加不上，用沙子打磨接头表面直到氧化层被掉为止，然后将其固定紧，经过这样处理后，没有再出现打火现象。

108

\*\*\*\*\*

## CR3240 彩色打印机

### 维修两例

四川南充东观石油技校 张振隆

**一、故障现象：**

使用黑色带，打印机开机初始化、工作均正常，使用彩色带却不能初始化，不能自检。

**分析：**用黑色带时工作正常，而用彩色带时却不能初始化，亦不能自检，说明问题出在打印机的彩色打印控制部分。经检查发现字车上控制彩色选色的机械部分阻卡。

**处理：**清洗字车上的控制选色的机械部分：环形槽和控制杆，并将控制杆小心打磨光滑，加少量润滑脂，装配复原后开机彩色初始化正常，自检选色、打印均正常，故障排除。

**二、故障现象：**

打印机打印的字符或表格线无规律的不清晰或时现时无，但打印纸上有被打印的迹印。

**分析：**打印纸上有被打印的迹印，说明打印头出针正常，无断针现象，打印头与打印纸距离正常，故而重点检查了色带运行部分，发现色带无规律地时走时停，导致有时在色带的某一处连续打印，造成打印不清晰或无色。进一步检查发现色带驱动电机无规律地时转时停，用万用表检测发现部分信号线不通。这是由于 CR-3240 彩色打印机的字车色带驱动电机信号线设置在字车下右边，打印机工作时，字车往复运动致使部分信号线折断，随着字车运动，信号线时断时通，从而导致本故障。

**处理：**削开被折断的信号线，小心焊好，并用透明尼龙胶固定好，开机故障现象消失。

109

# 通用教师业务档案管理系统的设计

广州航务工程学校 王坚城

## 1. 问题的提出

教师业务档案是学校管理的重要组成部分,通常采用手工方式管理,资料分散,工作繁琐,劳动强度大,工作效率低,不利于查询、分类、汇总,很难及时掌握教师队伍的结构和业务情况。电子计算机的广泛应用,为了提高工作效率,减轻劳动强度,为此编制了《教师业务档案管理系统(JSYW)》,为教师业务档案提供一种现代化的管理手段。

## 2. 系统总体概况

### 2.1. 系统运行环境

#### 2.1.1 硬件环境

2.1.1.1. 机型: IBM PC/XT 及其兼容机,内存容量不小于 640KB

2.1.1.2. 磁盘机: 至少一个硬盘和一个软盘驱动器

#### 2.1.2 软件环境

2.1.2.1. 汉字系统: CCDOS2.13F (H)

2.1.2.2. 数据库管理系统: 汉字 FoxBASE2.1

### 2.2. 系统主要功能

2.2.1. 档案登记: 包括个人简历、业务档案(上课、备课、实践性教学、著书、进修)、奖惩和班主任工作。

2.2.2. 档案维护: 包括对档案登记的内容进行修改和删除、数据库初始化、建立索引和数据备份。

2.2.3. 档案查询: 具有较强的查询功能,个人档案可进行组合查询,12个查询项目,用户可以随意组合成查询条件,查找出满足条件的教师显示在屏幕上,并按部门打印出来;可进行分类查询,可查出同一门课、同一教研组、同一班的教师上课情况;班主任工作和奖惩可按某学期全体和某人历年查询。

2.2.4. 统计汇总: 包括教师年龄、教龄、工龄、学历、职称、政治面貌和性别的统计,年龄、教龄、工龄将统计出各年龄段的人数和百分比,并画出直方图,还可把统计结果打印出来;另对每学期的上课情况进行统计,可统计出每学期的有课、无课、兼职教师人数,总课时数、理论课时、实践课时、专职教师课时、兼职教师课时,高讲、讲师、助讲和教员的上课时数;还可以统计当前学期无课教师名单。

2.2.5. 报表输出: 具有丰富的报表,按各种要求输

出不同的报表二十多种。

2.2.6. 系统字典: 通过课程名登记和教研组登记,建立课程库和教研组库,方便进行业务登记、查询和修改工作,尽量减少汉字输入。

2.2.7. 课程表编制: 包括编制、修改和打印,这里只提供一种课程表格式,用户可利用系统字典从右表中选取所需的课程名,然后打印出来的一种手段。

## 3. 系统设计

3.1. 系统模块结构图: 见附图表。

3.2. 数据库设计: 本系统共有 11 个数据库。

3.2.1. 个人简历库: 包括编号、姓名、性别、民族、出生年月、职称、职务、何时工作、何时任教、所修专业、毕业院校、科室、教研组、学历、籍贯和家庭住址等内容。

3.2.2. 上课库: 包括编号、姓名、职称、课时数、周课时、考核方式、教材名称、任教班级、学生人数、平均成绩、优秀率、不及格率和教学质量等内容。

3.2.3. 备课库: 包括编号、姓名、职称、学年、学期、备课名称、课时数、教材名称、任教班级等内容。

3.2.4. 实践性教学库: 包括编号、姓名、学年、学期、实践名称、实践课时、任教班级、学生数、优秀率、良好率、及格率和不及格率等内容。

3.2.5. 著书(教研)库: 包括编号、姓名、书名、工作性质、出版时间和出版单位。

3.2.6. 进修(锻炼)库: 包括编号、姓名、职称、进修内容、成绩、进修方式和进修地点等内容。

3.2.7. 奖惩库: 包括编号、姓名、奖惩名称、学年和性质等内容。

3.2.8. 班主任文库: 包括编号、姓名、学年、学期、班级、学制、层次和工作表现。

3.2.9. 汇总库: 包括学年、学期、教师总数、有课教师人数、无课教师人数、兼职教师人数、课时总数、专职教师上课、兼职教师上课、高讲上课、讲师上课、助讲上课、教员上课、理论课和实践课等内容。

3.2.10. 课程库和教研组库: 课程库包括编号和课名;教研组库只有教研组字段。

### 3.3. 系统设计特点

3.3.1. 有良好的用户界面,采用全屏幕操作,全菜

单选择, 菜单多样, 具有动态感, 增强使用者兴趣; 系统还利用 2.13H 的特殊显示功能, 使屏幕显示更漂亮; 系统在屏幕下方随时提示用户系统所处的功能模块。

**3.3.2.** 系统采用了模块化树形结构, 整个系统由主控模块和档案登记、档案维护、档案查询、统计汇总、报表输出、系统字典和课程表编制七个子模块组成。各子模块之间既相互独立, 又具有一定的内在联系。

**3.3.3.** 为了方便用户操作, 尽量减少汉字输入, 进行业务资料输入、查询和修改时, 可利用系统字典从右表中选取课程名、教师姓名和教研组等。

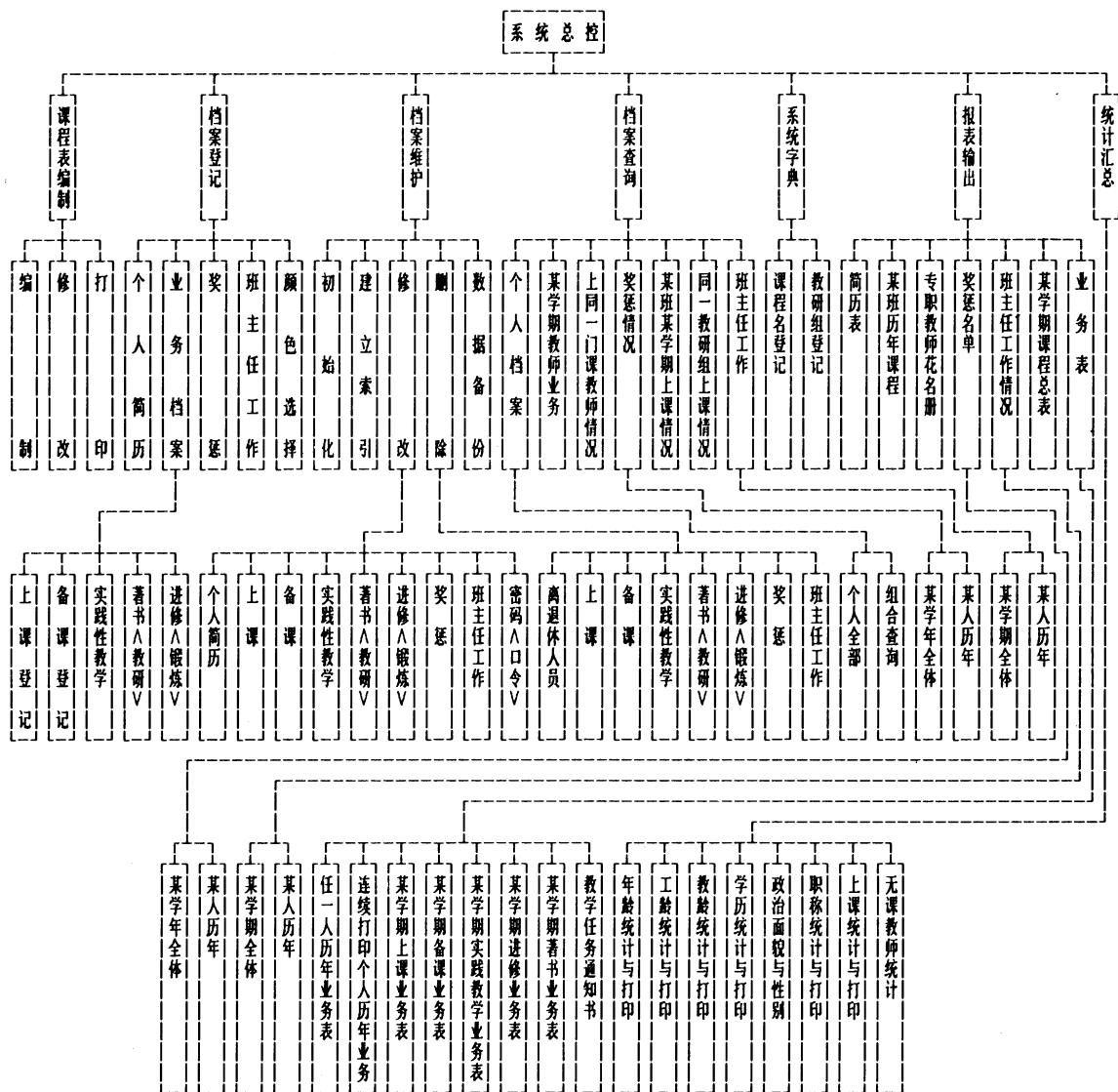
**3.3.4.** 系统还设置六种颜色供用户选择, 进入系统前, 用户可根据自己的爱好选择颜色。在档案登记子模

块下, 也设有颜色选择, 若用户对进入系统前所选颜色不满意, 可在此模块下重新选择。

**3.3.5.** 系统具有较强的通用性, 用户首次使用该系统, 首先要输入使用单位名称、一级密码和二级密码, 密码由用户决定; 用户还可在档案维护子模块下, 进行修改密码和使用单位。

**3.3.6.** 系统简单易学, 容易操作, 非计算机人员按屏幕提示操作即可使用。

**3.3.7.** 系统容错能力较强, 当查询不存在的记录时, 系统会提示“无此记录, 按任一键继续!”; 对判断“是否 (Y/N)?”都进行限制, 只能输入“Y”和“N”; 对菜单选择若超出范围, 屏幕右上方会提示“按空格键”。(附系统模块结构图)



## 动画演示摆线的形成

广州无线电厂 应海涛

摆线是圆上一点随着圆的滚动，而形成的轨迹。一般，根据圆滚动方式的不同，它可分为三种：直线摆线（圆在直线上滚动而形成），内圆摆线（圆在另一个圆上滚动而形成，两个圆心在切点的同侧），外圆摆线（圆在另一个圆上滚动而形成，两个圆心在切点的异侧）。

在教学实践中，很难画出摆线，更不用说演示摆线随着圆的滚动而逐渐形成的过程。

现在有了计算机，利用以下程序，可以很快画出三种摆线，并以动画形式演示其形成过程，从而使人加深了对摆线定义的理解。

调节程序开头的几个定义（#define），可以得到不同的摆线。

程序以 Turbo C2.0 编制，圆滚动的实现是利用 Turbo C 提供的两个图形函数：getimage 与 putimage。前者是将画面中特定的矩形块以位图像方式存入指定空间，后者则作相反的动作。调用 putimage 时，置它的最后一个参数为 XOR\_PUT，使每次将指定空间的位图像送到显示画面时，先与被覆盖的图像进行异或，这样以相同的参数每作两次 putimage，即可使画面恢复到第一次使用 putimage 前的样子。这是实现计算机动画的主要手段。以圆的一条半径随圆滚动而转动，指示圆的滚动。半径转动的实现，则是利用图形函数 setwrite mode 设置画线方式为 XOR\_PUT 方式，这样，以相同的颜色在相同的位置每画线两次，即可使画面恢复到第一次画线前的原样。（附程序清单如下）

```
#include <graphics.h>
#include <math.h>
#include <bios.h>
```

```
#define PI 3.14159265
```

```
#define Lr 40
#define Lx0 40
#define Ly0 210
#define Lrate PI/8
#define CirRx 300
#define CirRy 200
#define CirR 120
#define Cr 40
#define Crate PI/9
```

```
void main (void)
{ int i=VGA,j=VGAHI, x,y, x1,y1, rx,ry;
  int rx1,ry1, x2,y2;
  float thl, thc;
  char pl[(Lr+3)*(Lr+3)*2];
  char pc[(Cr+2)*(Cr+2)*2];

  initgraph(&i,&j,"c:\\bc\\bgi");

  setcolor(LIGHTRED);
  circle(Lx0+Lr,Ly0+Lr,Lr);
  getimage(Lx0,Ly0,Lx0+Lr+Lr,Ly0+Lr+Lr,pl);
  putimage(Lx0,Ly0,pl,XOR_PUT);

  rx=CirRx;
  ry=CirRy+CirR-Cr;
  circle(rx,ry,Cr);
  getimage(rx-Cr,ry-Cr,rx+Cr,ry+Cr,pc);
  putimage(rx-Cr,ry-Cr,pc,XOR_PUT);

  setcolor(11);
  line(Lx0,Ly0+Lr+Lr,getmaxx(),Ly0+Lr+Lr);
  circle(CirRx,CirRy,CirR);

  setcolor(LIGHTGREEN);
  setwrite mode(XOR_PUT);
  for (i=0; i<30000; i++)
  { thl=i/Lrate;
    thc=i/Crate;

    x=Lx0+i-Lr*sin(thl)+Lr;
    y=Ly0-Lr*(1-cos(thl))+Lr+Lr;
    line(Lx0+Lr+i,Ly0+Lr,x,y);
    putpixel(x,y,(getpixel(x,y) ^ YELLOW));
    putimage(Lx0+i,Ly0,pl,XOR_PUT);

    rx=CirRx+(CirR-Cr)*sin(thc);
    ry=CirRy+(CirR-Cr)*cos(thc);
    x1=rx+Cr*sin(thc*CirR/Cr);
    y1=ry-Cr*cos(thc*CirR/Cr);
    line(rx,ry,x1,y1);
    putpixel(x1,y1,(getpixel(x1,y1) ^ YELLOW));
    putimage(rx-Cr,ry-Cr,pc,XOR_PUT);

    rx1=CirRx+(CirR+Cr)*sin(thc);
    ry1=CirRy+(CirR+Cr)*cos(thc);
    x2=rx1+Cr*sin(thc*CirR/Cr);
    y2=ry1-Cr*cos(thc*CirR/Cr);
    line(rx1,ry1,x2,y2);
    putpixel(x2,y2,(getpixel(x2,y2) ^ YELLOW));
```

# 活用ECHO命令的功能

广东汕头金山中学 何管略

众所周知, ECHO 为 DOS 的内部命令, 它的主要功能, 是在 BAT 程序中显示信息。我们从操作实践中体会到, 这条命令, 看似简单, 要是活用起来, 功能倒也不少, 而且还有禁忌。现在简介于下:

ECHO 命令的格式: [ @ ] ECHO [ . ] [ ON / OFF ] [ TEXT ]

(1) 前后无任何参数的 ECHO 命令, 报告系统对 DOS 提示符, 和命令本身有无回应显示的状态。这种状态, 在 BAT 程序中, 可由参数 ON 或 OFF 来设定。而当 BAT 程序运行程序结束时, 系统会自动设置为 ECHO ON。

(2) 在 BAT 程序中运行 ECHO OFF 命令可以控制此后的 DOS 提示符, 和命令字符, 不要显示出来, 使屏幕保持整洁、醒目的状态; 但是, 对于当前行的信息, 系统还是会完整地显示在屏幕上的。为了避免这种“事与愿违”的现象, 可以键入 @ECHO OFF。其功能可解释为: 从当前行开始, 运行 ECHO OFF 命令; 但是此项功能, 也只在 BAT 程序中方才有效, 而在 DOS 命令行中, 系统是予以接受的。

(3) 要在屏幕上显示空行, 可用 ECHO 命令。值得注意的是: ECHO 和字符之间, 不能留有空格, 而且句号后面, 必须全部留空。否则, ECHO 命令会把它们当作信息 TEXT, 在屏幕上逐个显示出来, 反而破坏我们对空行的美观要求。

(4) ECHO 命令能够显示的 TEXT, 包括各类字符和汉字, 而且对字母的大、小写, 它都能够严格区

分。但是, 它对 I/O 重定向操作符“<”和“>”以及管道操作符“|”, 却不予照样显示; 此时 BAT 程序直接运行与该字符相应的操作。因此, 如在 ECHO 命令后面的 TEXT 中, 包含有这类操作符的话, 运行结果势将难以预料。

(5) 大家知道, ANSI 换码序列的功能, 是通过控制字无 Esc 来实现的; 而该键的 ASCII 码不能在 DOS 的命令行中保存, 所以无法直接获得。但是, 它却可以放在 ECHO 命令的 TEXT 中, 通过运行 BAT 程序, 来实现相应的功能。例如, 要将功能键 F10, 设定为 DIR / W 命令的话, 我们可在 BAT 程序中写入

```
ECHO <[0;68;"DIR / W";13p
```

只要运行一次, 这个换码功能, 就得到实现。

上述字符“<”, 是借用汉字图符打印出来的; 实际上, 它代表换码控制字元 Esc 的 ASCII 码, 在屏幕上显示的, 应是单字符。这个字符比较特殊, 许多文字编辑软件, 都跟 DOS 的命令一样, 键入时就立即执行它的取消功能, 而不保存它的 ASCII 码。我们编程测试时, 是用 PC SHELL 7.0 的文字编辑器, 通过 Alt+27 组合键, 从数字小键盘获得的。当然, 这项功能, 也可用下列 DOS 命令:

```
PROMPT $c[0;68;"DIR / W";13P
```

来实现, 各有优、缺点, 在此不赘述。

ECHO 命令也许还有其它功能, 本文愿起抛砖引玉的作用, 仅供参考。

(111)

```
putimage(rx1-Cr, ry1-Cr, pc, XOR-PUT);
```

```
if (bioskey(1))
    break;
```

```
line(Lx0+Lr+i, Ly0+Lr, x, y);
putimage(Lx0+i, Ly0, pl, XOR-PUT);
```

```
line(rx, ry, x1, y1);
putimage(rx-Cr, ry-Cr, pc, XOR-PUT);
```

```
line(rx1, ry1, x2, y2);
```

```
putimage(rx1-Cr, ry1-Cr, pc, XOR-PUT);
```

```
closegraph();
```

(137)

# APPLE II 的磁盘比较程序

黑龙江省伊春市气象局 马 骥

使用过 IBM-PC 系列机的朋友都知道,在其系统盘中有一个 DISKCOMP.COM 外部命令,它能在单驱动器或双驱动器下比较两张磁盘是否一样,对于保证磁盘备份的可靠性有很大作用,同时对于深入研究磁盘的存储格式也有很大的帮助。相反在 APPLE II 上就没有这个命令,在我使用过的几十种著名工具软件中也没有发现有这样的功能,就此,我编了一个 APPLE II 的 DISKCOMP 程序,经过几个月的使用还比较令人满意,现把它奉献给大家,如有不足之处请大家不吝赐教。现将其使用方法简介如下(源程序机器码附后):

程序启动后在屏幕的上方显示标题与提示,按下[ESC]键将退出 DISKCOMP 程序而返回,按下[RETURN]键将自动按其默认值进行双驱动器 23 磁道的磁盘比较,按[←]键与[→]键选择比较磁盘的终止磁道(最大为 43 道)。按下[SPACE]键将选择单驱或双驱。

比较时如果发生读错误将显示发生错误的磁道和扇区数值(十六进制),并继续执行磁盘比较。如两张磁盘完全一致,将不显示任何信息,直至比较完毕显示 DISKCOMP OK! 并等待按键返回。如果两张磁盘不一致将随时显示不一致的磁道和扇区,此时若按下[SPACE]键将暂停并等待按键以继续执行。若想终止比较只要按下[ESC]键即可返回主程序。

为了简化源程序,在程序当中使用了一些监控子程序,另外,其读写磁盘是借助于 DOS3.3 中的 RWTS 公用程序,所以,在输入该机器码时,一定要仔细,不得有误。在运行该机器码前,一定要先将其存盘,以免发生不测而前功尽弃!

该程序也适用于中华学习机及其兼容机,其磁盘操作系统为 DOS3.3,其他版本不保证正常使用。

源程序机器码如下:

```
0D00- A9 23 85 E8 20 84 FE 20 2F FB 20 E2 FB 20 58 FC
0D10- A2 00 20 F1 0E A5 E8 20 DA FD 20 62 FC 20 62 FC
0D20- A2 84 20 F1 0E 20 C6 0E C9 9B D0 06 20 58 FC 6C
0D30- F2 03 C9 A0 D0 13 AD 2C 0F C9 C1 D0 06 EE 2C 0F
```

```
0D40- 4C 0D 0D CE 2C 0F 4C 0D 0D A4 E8 C0 23 F0 0F C9
0D50- 88 D0 05 C6 E8 4C 0D 0D A4 E8 C0 2B F0 09 C9 95
0D60- D0 05 E6 E8 4C 0D 0D C9 8D D0 99 A9 04 85 25 20
0D70- 42 FC A2 CE 20 F1 0E A9 06 85 22 A9 17 85 25 20
0D80- 62 FC A9 00 8D EB B7 85 E4 85 E5 85 E0 85 E2 A9
0D90- 07 85 E6 A9 7F 85 E1 A5 E4 8D EC B7 A9 01 8D EA
0DA0- B7 20 CB 0E C9 9B F0 43 20 5A 0E E6 E4 A5 E4 C5
0DB0- E8 F0 0A C6 E1 C6 E6 A5 E6 C9 00 D0 DAA9 07 85
0DC0- E6 A9 7F 85 E3 A9 8F 85 E1 AD 2C 0F C9 C1 D0 26
0DD0- 20 E2 FB A0 00 20 9E FC A2 EE 20 F1 0E 20 C6 0E
0DE0- A0 00 84 24 20 9E FC C9 9B D0 03 4C 04 0D A9 01
0DF0- 8D EA B7 4C FB 0D A9 02 8D EA B7 A5 E5 8D EC B7
0E00- 20 CB 0E C9 9B F0 E4 20 5A 0E A9 8F 85 E1 A0 00
0E10- 20 CB 0E C9 9B F0 D4 C9 A0 D0 03 20 C6 0E B1 E0
0E20- D1 E2 F0 06 20 D2 0E 4C 2F 0E C8 C0 00 D0 E1 C6
0E30- E1 C6 E3 A5 E1 C9 7F D0 D7 A9 8F 85 E1 E6 E5 A5
0E40- E5 C5 E4 D0 B6 C5 EB D0 0E A2 68 20 F1 0E 20 E2
0E50- FB 20 C6 0E 4C 04 0D 4C 93 0D A9 0F 85 E7 A9 00
0E60- 8D F0 B7 A5 E7 8D ED 87 A5 E1 8D F1 B7 A9 01 8D
0E70- F4 B7 20 83 0E A5 E7 C9 00 F0 07 C6 E7 C6 E1 4C
0E80- 63 0E 60 ADC2 AA AC C1 AA 20 B5 B7 90 03 20 96
0E90- 0E A9 00 85 48 60 A2 37 20 F1 0E ADEC B7 20 DA
0EA0- FD A9 AC 20 F6 F6 A9 A4 20 F6 F6 ADED B7 20 DA
0EB0- FD 20 62 FC 20 E2 FB 20 CB 0E C9 9B F0 96 C9 A0
0EC0- D0 03 20 C6 0E 60 2C 00 C0 10 FB AD 00 C0 2C 10
0ED0- C0 60 A2 4F 20 F1 0E A5 E5 20 DA F6 A9 AC 20 F6
0EE0- FD A9 A4 20 F6 F6 A5 E1 29 0F 20 DA F6 20 62 FC
0EF0- 60 BD 00 0F C9 00 F0 07 20 F6 F6 ED 4C F1 0E 60
0F00- 2A 2A 20 04 09 13 0B 03 0F 0D 10 20 31 2E 30 20
0F10- 20 3C 7C 4D 41 4A 49 7E 3E 20 2A 2A 8D 8A A0 A0
0F20- C4 D2 C9 D6 C5 BA A0 C1 A0 D4 CF A0 C2 A0 A0 D4
0F30- D2 C1 C3 CB A0 A4 00 D2 C5 C1 C4 A0 C5 D2 D2 A0
0F40- D4 D2 C1 C3 CB AC D3 C5 C3 D4 CF D2 A0 A4 00 C4
0F50- C9 C6 C6 C5 D2 C5 CE D4 A0 D4 D2 C1 C3 CB AC D3
0F60- C5 C3 D4 CF D2 A0 A4 00 8D A0 C4 C9 D3 CB C3 CF
0F70- CDD0 A0 CF CB A0 A1 A0 A0 D4 C8 C1 CE CB D3 C5
0F80- A1 A1 A1 00 12 05 14 15 12 0E A0 D4 CF A0 C2 C5
0F90- C7 C9 CE A0 A0 A0 13 10 01 03 05 A0 D4 CF A0 C3
0FA0- C8 C1 CE C7 05 A0 C4 D2 C9 D6 C5 A0 3C 2D AC 2D
0FB0- 3E A0 D4 CF A0 CE C5 D7 A0 D4 D2 C1 C3 CB D3 A0
0FC0- A0 A0 05 13 03 A0 D4 CF A0 C5 CE C4 8D 00 05 13
0FD0- 03 A0 D4 CF A0 D2 C5 D4 D5 D2 CE A0 A0 A0 13 13
0FE0- 01 03 05 A0 D4 CF A0 D0 C1 D5 D3 C5 8D 00 C3 C8
0FF0- C1 CE C7 C5 A0 D4 C8 C5 A0 C4 C9 D3 CB A0 A1 00
```

# 扩展 MUSIC功能

· 陈 剑 ·

使用中华学习机 CEC-I 的用户都知道, BASIC 中的 MUSIC 语句只能带两个参数, 如果用户想让计算机连续演奏若干个音符, 那 MUSIC 语句就不能胜任了。为了解决这个问题, 我用 6502 汇编语言编写了下面这个程序, 你只要上机输入这个程序 (程序一), 然后回到 BASIC 状态使用 CALL 768 就为你的 CEC-I 建立扩展 MUSIC 功能, 以后只要使用 &A1, B1, A2, B2 ..... An, Bn, 0, 0 就可以调用本程序。其中 Ai, Bi 分别表示频率和音长 (与 MUSIC 语句中的定义相同), 最后的 0, 0 是结束标志 (不可省)。下面给出一个使用扩展 MUSIC 功能的 BASIC 程序实例 (程序二), 本程序将演奏一曲“世上只有妈妈好”。

## 程序一

```
0300- A9 4C      LDA    #$4C
0302- 8D F5 03   STA    $03F5
0305- A9 12      LDA    #$12
0307- 8D F6 03   STA    $03F6
030A- A9 03      LDA    #$03
030C- 8D F7 03   STA    $03F7
030F- 6C F2 03   JMP    ($03F2)
0312- 20 F8 E6   JSR    $E6F8
0315- 86 EC      STX    $EC
0317- 20 4C E7   JSR    $E74C
031A- 86 ED      STX    $ED
031C- A5 EC      LDA    $EC
031E- C9 00      CMP    #$00
0320- D0 07      BNE    $0329
0322- A5 ED      LDA    $ED
0324- C9 00      CMP    #$00
0326- D0 01      BNE    $0329
0328- 60        RTS
0329- 20 F9 D0   JSR    $D0F9
032C- 20 BE DE   JSR    $DEBE
032F- 4C 12 03   JMP    $0312
```

## 程序二:

```
10 CALL 768
20 & 114, 180, 128, 60, 152, 120, 128, 120, 95, 120, 114,
60, 128, 60, 114, 240, 152, 120, 128, 60, 0, 0
30 & 114, 60, 128, 120, 152, 60, 171, 60, 192, 90, 228, 30,
128, 60, 125, 60, 171, 240, 171, 180, 0, 0
40 & 152, 60, 128, 120, 128, 90, 114, 30, 152, 180, 171, 60,
192, 240, 128, 180, 152, 60, 171, 60, 0, 0
50 & 192, 60, 228, 60, 192, 60, 255, 240, 0, 0
```

## 关于更改广东省部分城市 计算机操作人员 联合水平考试时间的通知

鉴于今年我省成人高考的开考时间与我办原定的计算机操作员水平考试时间重复, 根据部分城市学校的要求, 为照顾应届毕业生参加计算机操作员水平考试, 经研究决定: 今年计算机操作员水平考试时间从 5 月 9 日改为 5 月 2 日 (星期天)。特此通知。

广东部分城市计算机操作人员  
联合水平考试实施办公室  
一九九三年二月二十日

114

## 通讯

1993 年广东省部分城市计算机操作人员联合水平考试工作会议 3 月 1 日于广州市职业技术教育中心 (广州市电子职业高级中学) 举行, 参加会议的有来自广州、番禺、东莞、佛山、顺德、江门、中山、清远、汕头、肇庆等十个城市的计算机水平考试实施办公室、劳动局、电子工业总公司以及各职业技术学校的有关人员代表。

会议首先通报了去年部分城市参加联合水平考试的基本情况, 与会代表一致认为, 我省部分城市参加的计算机操作人员联合水平考试, 采用统一考试大纲、统一试题、统一考试时间、统一评分标准, 更具客观、公正、代表性, 受到了各考生和用人单位的好评。与会代表还就目前我国各市电子信息技术应用与推广情况, 职业技术学校开展计算机教学情况以及对今年参加联考的学生、在职人员开展培训工作充分交换了意见, 并提出了不少积极的建议。在电子信息技术应用更为广泛的今天, 尤其是省、市有关部门对职业高中学生实行“双证制” (毕业证书和技术等级证书) 的要求, 使今年参加联考的考生变得尤为踊跃。联考实施办就今年联考若干问题作了具体步署, 并要求各市分考场做好联考的组织、培训与考务工作。分考场代表们一致表示: 要积极组织, 密切配合, 做好今年联考的各项工作。

何文生

115

113



# 模拟城市设计SIMCITY玩法介绍

华南师大附中 高二六班

林天臻

模拟城市设计(全称 SIMILAR CITY DESIGNER)是 MAXIS 公司在 1989 年推出的全新类型的模拟游戏。这个游戏在外国评价很高,问世当年就被外国杂志评选为 1989 GAME OF THE YEAR (88 年是 TETRIS 俄罗斯方块)。然而,遗憾的是,此游戏却没有在中国掀起热潮,也许是大家对这个游戏不大熟悉吧。好!笔者就诚意推介这个游戏给各位游戏爱好者,希望各位能从中领略到做市长的甜酸苦辣吧!

SIMCITY 的情节就是:你作为一个城市的市长,手头上拥有一笔资金,利用这笔资金对城市进行设计和建设,然后通过税收的手段取得资金。游戏的开始你只有一片荒无人烟的大片土地,一座城市的兴建从此开始.....

SIMCITY 中你可建设有:

1. ROAD ----- 公路 (\$ 10)
2. POWER LINE ----- 电线 (\$ 5)
3. RAILROAD ----- 铁路 (\$ 20)
4. PARK ----- 公园 (\$ 10)
5. RESIDENTIAL ----- 住宅 (\$ 100)
6. COMMERCIAL ----- 商业用房 (\$ 100)
7. INDUSTRIAL ----- 工业 (\$ 100)
8. POLICE STATION ----- 警察局 (\$ 500)
9. FIRE STATION ----- 消防局 (\$ 500)
10. STADIUM ----- 体育场 (\$ 3000)
11. COAL POWER PLANT ----- 火力发电站 (\$ 3000)
12. NUCLEAR REACTOR ----- 核电站 (\$ 5000)
13. SEAPORT ----- 轮船公司 (\$ 5000)
14. AIRPORT ----- 飞机 (\$ 10000)

SIMCITY 中你可以使用 BULLDOZE (\$ 1) (铲土机)将建筑铲去,只需按一下鼠标右键而无需选择推土机的肖像。

城市建设之初,设计是至关重要的——①你得选择一块适宜的兴家之地,当然濒海或建在河边是最合适不过了。②工业和住宅区不宜靠得太近,否则由于污染令至住宅区没有居民而成为废墟。③商业用房毗邻工业有利于工商业的兴旺,工商业的兴衰对于你的财政税收影响很大。④由于就业关系,住宅区最好与商业用房相邻。(从②~④可看出,商业用房介于工业与住宅区的布局是最为合理的。)⑤工业不能过于密集,否则区域内的犯罪率会直线上升,间接影响经济效益。⑥公路要

避免多个交叉,否则到后期经济发达时,交通问题会伴之突出起来。⑦你的城市只有一辆火车,所以铁路建设要避免分叉且最好是环形,以充分利用紧张的火车资源。

建设过程中,电脑会作为你的智囊团——它不时会给予你幽默而恳切的忠告。例如交通问题严重时,电脑会告诉你:你的城市交通令市民头疼并愤怒,如果再这样下去的话,劝你还是准备防弹衣;当城市人口达到某种程度时,电脑又会夸奖你的政纲,鼓励你继续努力。

另外你不要忘记“群众的眼睛总是雪亮的”——定期征询民意。从 WINDOWS 中的 EVALUATION 功能你便可以了解到建设中的问题。从 Is the mayor doing a good job? (市长是否干好他的工作)的赞成率和反对率中你可以知道市民对你领导工作的综合评价。从 What are the worst problems? 中你可以了解到市民对各种建设投诉。如果问题投诉率不高于 10%,你大可不必理会,只管尽快去解决一些投诉率高于 10%的问题。建议解决的方案如下:

**TRAFFIC (交通)**——先从 MAP WINDOW 中找到交通最拥挤的地方。解决的方法是尽可能将繁忙的路段的交叉路口删去,或者多建几条同向公路或用铁路舒缓,分担各公路的压力。

**CRIME (犯罪率)**——方法是建警察局和提高地价,不得以时才忍痛疏散工业。提高地价的办法很多,其中建设公园和减少污染是较为有效的方法。

**POLLUTION (污染)**——一般工业集中的地方污染是相对严重的,此外交通密度和辐射都会不同程度引至污染。具体解决办法相信玩者可以因势利导。

**HOUSING (住房紧张)**——多建住宅自然可以解决。

**HOUSING COST (楼价过高)**——解决途径有两种,一是大建住宅楼宇,二是择低地价的地方建楼。

**UNEMPLOYMENT (失业)**——在住宅区附近建商业和工业区以扩大就业即可解决。

**FIRES (火灾)**——防止火灾蔓延,就得将火灾周围地区铲平,而建组消防队则是最好的方法。

**TAXES (税率)**——也许你的市民穷了点,降低税率有利于刺激经济发展,但要注意财政入不敷出。有趣的是,当其它问题不突出时,市民们总喜欢投诉这一

点。

### 菜单说明:

**SYSTEM** ABOUT SIMCITY (SIMCITY 软件介绍)  
(系统操作) PRINT (将城市地图打印出来, 有大小两种尺寸供选择)  
LOAD SCENARIO (城市例范)  
Start New City (重新开始)  
Load City (调进一个城市)  
Save City as .....  
Save City  
Exit (退出游戏)

其中城市例范分别为:

Dullsville 1900 Beredom  
SanFrancisco 1906 Earthquake (1906 年地震后的旧金山)  
Hamburg 1944 Bombing (1944 年轰炸后的汉堡)  
Bern 1972 Traffic (1972 年交通异常拥挤的)  
Tokyo 1957 Monster Attack (1957 年受海怪干扰的东京)  
Detroit 1972 Crime (1972 年罪恶的底特律市)  
Boston 2010 Meltdown (2010 年沉没中的波士顿)  
Rio 2047 Flooding (2047 年受洪水袭击的巴西城市)

**Options** Auto - Bulldoze (自动铲土)  
Auto - Budget (自动预算; 不用此功能, 系统每年一月将把预算显示出来。)  
Auto - Goto (自动运行)  
Sound on / off (声音开关)  
Speed Fastest (以最快速度运行)  
Animate all (运行全部的机构)  
Frequent animation (经常的动画)

其中“以最快速度运行”项有:

Fast (快速度运行)  
Middle (中等速度运行)  
Slow (慢速度运行)  
Pause (冻结城市运行)

**Disasters** Fire (火灾)  
(模拟灾害) Flood (洪水)  
Air Crash (飞机空撞)  
Tornado (台风、龙卷风)  
Earthquake (地震)  
Monster (海怪侵扰)  
Disable (屏蔽灾害)

Windows Maps (城市整体布局)  
(调查窗口) Graphs (城市发展图)  
Edit (城市发展动画)  
Budget (财政收支预算)  
Evaluation (市民投诉)  
Close (关闭工作窗口)  
Hide (隐含工作窗口)

### 心得及建议:

1、建设过程时, 最好把运行速度减至 SLOW 甚至将发展 PAUSE, 以便集中力量建设, 不受城市发展的某些因素干扰。

2、初玩者建议将灾害屏蔽 (DISABLE) 起来, 方便建设。

3、玩者千万不要把预算中 TRANS (周转资金)、POLICE (警务支出)、FIRE (消防支出) 的 FUNDING LEVEL (资金水平) 调低, 这虽然能够减少财政支出, 但是同时由于差饷减少, 导致经济流通速度减慢并且道路维护及火灾预防等措施不能正常进行, 很快你就会发现你的城市会陷入一片混乱——公路被导毁、铁路被弄断以及电线被切断等等, 此时经济还会恶化, 这时你去花钱修补道路, 刺激经济已经是得不偿失了。

4、调整好税率。通常减低税率能刺激经济回升, 而较高的税率在几年内会有较多的收入。玩者不一定要追求高税率, 因为这会抑制经济发展, 特别在经济困难甚至入不敷出时, 不要立刻提高税率来弥补损失。最好的方法是大幅下降税率以刺激工商业的增长, 等到工商发展走上正轨时再逐步提高税率才最经济。

5、城市发展到一定程度时, 电脑会建议你建一个飞机场。这里建议大家尽量不要起, 因为一旦起了飞机场, 城市的交通问题会立即异常突出, 导致经济下降, 况且机场对城市的发展作用不大。(笔者曾经试过: 当城市发展规模很大、每年财政收入达到一千多元, 便依电脑之忠告建了个飞机场, 谁知经济立刻急剧下降, 建机场的第三年便扭盈为亏, 很快原先赚到的三万多建设资金全部用光, 整个城市随之成了废墟。)

6、预防交通过于繁忙, 可以在设计时就考虑用一条街一排楼的方式, 这样会大大减少道路交叉。

### 游戏作弊秘诀:

SIMCITY 存的以 CTY 为后缀的文件, 从偏移量 \$D24 开始连续四个字节为资金数目, 大家可以用 DEBUG 等软件进行修改, 千万不要贪心, 以免数目过多而爆机。

116

# DIGGER 游戏的改进

河北石家庄飞机制造厂计量处 彭铁忠 齐凤芸

DIGGER (挖金子) 是一个相当精巧的小游戏, 其代码不长, 但趣味性非常强, 早几年在 PC/XT 机上流行很广, 着迷者甚众。不过近来似乎再很少见到有人玩它, 一打听, DIGGER 在配置 EGA.VGA 卡, 以及以 80286、80386 为 CPU 的诸多高档机型上, 均无法运行。其表现有二, 一是在 EGA.VGA 卡上显不出正常图形。二是在高速 CPU 机型上, 各角色运动速度太快, 常常是上下左右键稍一动, DIGGER 便已在屏幕四周挖了一圈, 鬼也是象子弹一样追上来, 人反应跟不上。

经过对 DIGGER 较全面的剖析, 发现上述二种兼容性方面的缺陷, 主要集中在 DIGGER 内部二段代码上, 现将其摘录分析并作修改如下:

程序段 1 如下:

程序段 1

```
loc_a_042c:
    mov si, offset data_b_0255
    mov cx, 0eh
    mov bl, 0
    mov dx, 3d4h

loc_a_0437:
    mov al, bl
    out dx, al; 3d4 port, 6845 index reg.
    inc bl
    inc dx
    mov al, (si)
    out dx, al; 3d5 port, 6845 data reg.
    inc si
    dec dx
    loop loc_a_0437

loc_a_0444:
    ...
data_b_0255 ; 6845 register's data.
    db 38h, 28h, 2ch, 0ah, 7fh, 6, 64h
    db 70h, 2, 1, 20h, 0, 0, 0
```

该段程序实际上是通过 CGA 卡上 6845 寄存器编程, 来设置彩色中分辨率 (320×200) 图形模式。该模式事实上完全同 BIOS 的视频模式 5。由于 EGA.VGA 在寄存器编程这一级上, 并未达到与 CGA 完全兼容。因此这段程序在 EGA VGA 卡上也就实现不了预期模式设置目的。(实际运行时, 整个 DIGGER 画面被分割成几个部分, 呈歪斜状显示在上半屏幕)。此处修改时, 只需调用兼容性很好的 BIOS 视频

模式设置功能即可。

```
C>debug digger.com
-a 42c
xxxx: 042c mov ax, 5
xxxx: 042f int 10
xxxx: 0431 jmp 444
xxxx: 0433
-w
-q
```

程序段二如下:

程序段二

```
sub_a_023f proc near
    push bp
    sub sp, 6
    mov bp, sp
    mov word ptr (bp+2), 0

loc_a_024a:
    mov ax, (bp+2)
    cmp ax, (bp+0ah)
    jge loc_a_0267
    mov word ptr (bp+4), 0

loc_a_0257:
    cmp word ptr (bp+4), 64h
    jge loc_a_0262
    inc word ptr (bp+4)
    jmp short loc_a_0257

loc_a_0262:
    inc word ptr (bp+2)
    jmp short loc_a_024a

loc_a_0267:
    add sp, 6
    pop bp
    retn

sub_a_023f endp
```

如果将此段子程序取名 delayLoop(), 则可将其翻译成如下的等效语言程序:

```
void delayloop (int c)
{
    int i, j;
    for (i=0; i<c; i++)
        for (j=0; j<100; j++);
}
```

很明显, 这是一段以空循环延时的子程序。在 DIGGER 内部运行中, 该段程序之调用非常频繁。DIGGER 就是通过大量调用该段程序延时, 来协调各游戏角色之间的动作。我们知道, 当前各种微机机型, CPU 与主频不同, 指令执行速度差异很大, 故该段程序延时效果也随机种的不同而差异很大, 而 DIGGER 原先设计是以 6MHz 主频的 8088 为基础, 一旦遇到主

# 《银河英雄》及其修改方法

岳阳巴陵石化公司橡胶厂 SBS 车间 赵战鲲

现在 APPLE 机上的新游戏软件虽不多，但值得一玩的也有不少。如《银河英雄》(SPACE ROGUE)就很有值得一玩。这套软件是 ORIGIN SYSTEM 公司 1989 年出品，描写太空时代的一个 RPG。游戏中你不仅要通过与周围的人物对话以了解这个星系中各种势力间错综复杂的关系，还要为了生存和荣誉，参与星际贸易并与海盗进行战斗，最后完成一项艰巨的任务。

此游戏操作难度颇高，单是平安穿过 Malir Gates 就要苦练上数日，更不用说挣钱买装备和在数艘海盗船的炮火下逃生了，稍不留心便会 GAME OVER! 再加上大量英文对话，更使许多玩家进退两难。为了便于大家更快地投入到神奇的太空冒险中来，我在这向各位介绍几则改版方法供参考。

## 一、开机从上一次 SAVE GAME 处开始。

每次游戏开始，选定 OLD GAME 都需要输入 PASS WORD，因大家都没有“太空船驾驶手册”这一关很难得过。修改的方法是：将 A 面 \$0B 轨 \$5 扇区第 \$7D 字节的 \$38 改为 \$18，这样无论键入任何字符串都可获得通过。

## 二、赚钱

在漫漫星河中，赚钱是十分辛苦的，有时还会遭到海盗的劫掠，使我们一船的梦想成为泡影，又得从头再

来。为了迅速加强武器及防卫系统性能，我们可以对 SAVE GAME 中的有关数据进行修改，方法是：将 B 面第 \$20 轨 \$4 扇区第 \$25 字节开始的连续两个字节改为 \$FFFF 或其它值，你即拥有了大量现金，还有什么武器和防卫系统买不起呢？

## 三、加强船身防卫性能

在 A 面 \$0F 轨 \$C 扇区找到“2907”将其中的“07”改为“0F”或其它合适数值，这样可使我们的飞船能够承受更多的攻击及宇宙尘埃、气体的磨损和腐蚀，在战斗和穿越 Malir Gates 的时候，增加一份生存的希望。

## 四、顺利通过 Malir Gates

通过 Malir Gates 进行时空跃进是件需要速度、技巧、耐心的精细工作，如没有娴熟的驾驶技术是很不易成功的，而且在八大星系中穿梭，每次都要由 Malir Gates 进行超时空跃进，时间上太“浪费”。我们可以将 A 面第 \$0F 轨 \$D 扇区第 \$AD 字节开始的“90 0D”改成“B0 0D”。这样，只要飞船加速进入 Malir Gates 即可保证完成超时空跃进。

虽然，上述修改使游戏减少了几分乐趣，但相信对于正在屏幕前苦苦征战的英雄们还是很有帮助的。笔者和大家一样捧着字典，盼着终有一日能够获得 Golden Sunburst of Valor 的最高荣誉。

118

频更高的微机，延时时间便成倍下降，由此造成各游戏角色运行速度大大加快，因而难怪人的反应跟不上去。

一般来讲，要实现与机器速度无关的延时，利用系统时钟比较好。但在这段程序中不行，其一是系统时钟最小只精确到 1/18 秒，时间太长。其二是 DIGGER 还使用了系统时钟触发其后台音响，并为此经常变化系统时钟频率，使计时不再准确。为简便起见，我们仅在此修改其内部循环次数，延长其延时时间以适应高速机器。

此处修改，还有一个须注意的问题，就是标号 loc\_a\_0257 处指令“cmp word ptr (bp+4), 64h”为 4 字节机器指令 (83 7E 04 64)，若将 64h 增加超过 0FFh，例改为指令“cmp word ptr (bp+4), 2000h”，则机器码变为 5 字节 (81 7E 04 00 20)。所以还应

用其它较短指令代替其内部循环，修改如下：

```
C>debug digger.com
-a 257
xxxx: 0257 mov cx, nnnn
xxxx: 025a loop 25a
xxxx: 025c jmp 262
xxxx: 025e
-w
-q
```

其中 nnnn 为循环次数，一般 16M 机器，适用 2800H 较为合适，12M 机器，选用 2000H 较为合适。其余主频，读者自行试验确定，当然，如能编一段测速程序自动确定该值最好。

经上述修改，DIGGER 便可在各种高档机型上运行了，希望你也迷上它。

117

# 提高硬盘抗病毒能力一法

中国纺织大学机制 891 班 唐 红

硬盘由于容量大、速度快，稍有不慎就会成为病毒滋生的场所。为防止病毒进入硬盘，管理人员通常让硬盘启动后自动执行一条监测病毒的常驻 (TSR) 程序，如 CPAV 软件包中的 VSAFE.COM 等，但这种方法对从软盘启动的用户就无能为力了，这种情况下病毒就可能在不知不觉中侵入硬盘。因此，我们必须杜绝病毒的这一传播途径。其方案大致有两种：

一、改变软、硬盘启动顺序的优先级。

二、限制软盘启动用户对硬盘的使用权。

两种方案比较而言，第一种方案须从硬件上入手，相对比较困难；而通过对 PC-DOS 操作系统启动过程的分析，发现第二种方案还是现实可行的。

硬盘在启动时，先将位于硬盘 0 头 0 柱 1 扇区的主引导程序读到内存 0: 7C00H 处执行。正常的主引导程序所做的工作就是在与它处在同一扇区的分区信息表中找出一个活动分区，并把控制权交给该分区去引导系统。系统文件 IBMBIO.COM 得到控制权后根据硬盘的分区表给硬盘建立 BPB (基本输入输出参数块) 表，如果分区信息不正确，BPB 表将无法建立，系统也就不承认硬盘。

软盘启动时除了比硬盘少执行一段主引导程序外，其它过程完全相同。

通过上面的分析，我们要使从硬盘和软盘启动的用户对硬盘有不同的使用效果，就必须从两者唯一的区别处，即主引导程序入手。

首先要限制软盘启动用户对硬盘的使用权。从上面的分析可知，这只要把分区表移到别的扇区就可以了。另外，我们还必须保证硬盘启动的用户能正常使用硬盘，这就要在 IBMBIO.COM 检查分区表时告诉它到何处去取得正确的分区信息，这项工作就要由重编的主引导程序去完成。

将程序 NEWMBOOT.ASM 经编译、连接并由 EXE2BIN 转换成 COM 文件 (设文件名为 NEWMBOOT.COM)，然后用 DEBUG 对硬盘进行处理：

第一步，将原主引导扇区内容移到 0 头 0 柱 3 扇

区；

C>DEBUG

-A

2189: 0100 MOV AX, 0201

2189: 0103 MOV BX, 7C00

2189: 0106 MOV CX, 0001

2189: 0109 MOV DX, 0080

2189: 010C INT 13

2189: 010E MOV AX, 0301

2189: 0111 MOV CX, 0003

2189: 0114 INT 13

2189: 0116 INT 20

2189: 0118

-G

Program terminated normally

-Q

第二步，替换原主引导程序。

C>DEBUG NEWMBOOT.COM

-E7DFE 55 AA

-A

2199: 0100 MOV AX, 0301

2199: 0103 MOV BX, 7C00

2199: 0106 MOV CX, 0001

2199: 0109 MOV DX, 0080

2199: 010C INT 13

2199: 010E INT 20

2199: 0110

-G

Program terminated normally

-Q

新的主引导程序在引导时驻留内存，通过修改 13H 中断向量把对硬盘 0 头 0 柱 1 扇区的操作过滤到 0 头 0 柱 3 扇区，这样系统文件 IBMBIO.COM 就能得到移到该扇区的正确分区信息，使硬盘能正常使用；而软盘启动时由于没有执行这段主引导程序，也就得不到正确的分区信息，从而无法使用硬盘。

119

# MI.COM在病毒检测中的应用

广州中山大学计算机系 89 级应用班 郭 江

MI.COM 是 PCTOOLS 工具箱中的一个用于检测内存空间利用情况的小工具。在运行一些比较庞大的软件(如中文操作系统、汉化 FOXBASE+2.10 等)之前,我们常常使用该软件检测是否有足够内存空间。不仅如此,MI 在检测计算机病毒方面还有其独特之处。

大家都知道,绝大部分计算机病毒都需要在内存中划一块区域以存放其病毒代码。为了防止被其它软件覆盖,它必须采用比较安全的方式来存放其代码。最常用方法是修改系统数据区[0040H]:[0013H]处的内存总容量(一般为 640K,十六进制表示为 280H),使该位显示的内存总量不足 640K,同时病毒将其代码存放在内存高端地址,从而达到其隐蔽目的。绝大多数系统型病毒采用这种方法。

在任何时候,运行 MI.COM,一旦发现显示的内存总量不足 640K,则说明内存已有病毒在驻留,此时用 DEBUG 查看中断向量表,还能知道病毒驻留在何处,以及它接管了哪个中断。

对于文件型病毒,有的同样通过修改[0040H]:[0013H]处的内存总量,但大多数通过修改内存控制键,采用 INT27H 或 INT21H 的 31H 功能调用驻留内存。对于这种情况,MI.COM 又如何对付呢?其实 MI 显示的内存利用信息中,同样显示了内存控制键中的各个内存控制块(MCB)所指示的文件及其所占内存字节数。此时若发现有非 TSR 软件而现在又驻留了内存的程序,则该可执行文件很可能已染上病毒。

为了让读者明白 MI 的妙用,笔者附上两种病毒的检测显示:例 1、例 2

例 1 是现在比较流行的 AZUSA 病毒(又叫 2708 病毒),它占用了 1K 的高端内存。因此内存总量显示为 639K,该病毒使用新版的 SCAN85V91 可清除。不过笔者发现对不少病毒(特别是多重感染的情况),SCAN 清除不干净或清除不了。其实最安全的方法还是使用 NORTON 系列工具中的 NU.EXE 或 DISKEDIT.EXE,拿标准的硬盘(软盘)引导扇区覆盖带病毒的引导扇区。(建议在无病毒状态下给各种盘保存一个标准的引导扇区,以防不测!)

例 2 则是在本校比较流行的“爱国者”病毒(笔者为其命名,该病毒发作标志为显示字符串“WORKING HARD FOR OUR CHINA!!! \* WANG ZHEN

JIAN\*”),从该例中很明显地看出 DIRDEL.EXE 是病毒载体。由于该病毒编程上的失误(接管 INT21H 的同时,破坏了 INT12H),使得 MI 调用 INT12H 查看内存容量时,实际上调用了 INT21H,所以返回的值也就不同(在此为显示 25901K)。一般文件病毒是不这样做的。

至于一些不是上面所提及的两种驻留内存方式的病毒,则多是采用 48H 功能调用申请系统分配内存,也多是将病毒代码放在内存高端地址。对此病毒 MI 无法直接从显示信息中知道,但 MI 也不是毫无办法。笔者在日常上机操作过程中,对于常用操作系统所占内存数一般都有印象。某次运行 MI 时,发现可用内存由原来的 618K 变为 616K。此时内存总量仍为 640K,又无一可执行文件驻留内存。笔者用 debug 查看中断向量表,发现 INT21H 已被修改。且指向内存高端地址,这显然是病毒所为。经过对该病毒取样分析,才知这是国内较流行的 1575 病毒(又名毛毛虫病毒)。所以,建议用户记住自己惯用操作系统的可用内存数(一般只须记住刚启动完系统,无任何用户程序驻留时的可用内存数),此时使用 MI 将更是得心应手。

总之,MI 对于大多数计算机病毒的检测是有效的。如果用户同时又有一些新版的清病毒软件(如 SCAN85V91 或 86V94)。与 MI 相配合,则效果更佳。当然 MI 也不是万能的。比如现在较流行的新型病毒 DIR-2 病毒,MI 就毫无办法。该病毒用 SCAN 85V 91 可查出,并可使用配套 CLEAN 的清除。

MI.COM 的使用还须注意一个问题。某些较低版的 MI.COM 在显示内存信息时也显示了 MI.COM 所占内存。用户就难判断 MI.COM 本身是否已染上病毒。笔者使用的是 PCTOOLS 7.0 中的 MI.COM,则不会出现上面这种问题,与 MI.COM 功能相类似的工具还有 SMAP.COM,MSDOS 5.0 中的 MEMCHK,DRDOS 6.0 中的 MEM.EXE,以及 PCshell 中的 MEMORY MAP 功能等等。

## (例 1)

Memory Info V7

(c) 1991 Central Point Software, Inc.

Addr. Total bytes Program or device driver

# V304 病毒的检测和清除

广西信息技术公司 蔡宁浩

最近在我公司的微机上发现一种新病毒,这种病毒只感染可执行文件(.COM和.EXE),用当前流行的CPAV,SCAN,KILL均不能发现和清除,因这种病毒对可执行文件每次增加304字节,故在此暂命名为V304病毒。

## 一、病毒的特点

V304病毒是一种外壳型病毒,它驻留在后缀为.COM和.EXE文件中。当运行含有这种病毒的.COM文件时,病毒程序首先获得控制权,然后恢复原文件的前三个字节,对于未被感染的系统,它将其自身带入内存的某一特定地方驻留,同时修改系统的INT21H中断向量,指向病毒程序的相应服务部分,这样凡是用到INT21H中断调用均首先执行病毒程序,同时该病毒程序还修改了INT24H,使得病毒的感染过程能在没有任何信息提供给用户,增加了隐蔽性。这种病毒只感染后

缀为.COM和.EXE文件,被感染病毒后的文件日期维持原来不变。这种病毒感染可执行文件时,对于.COM文件只感染一次并使其增加304字节,对于.EXE文件则无论该文件是否感染了该病毒,每次均在其末尾增加304字节,直到程序无法运行或盘满为止。这种病毒属于良性,它不会破坏原文件的内容,只是在染有该病毒的系统运行程序时偶尔会出现死机,而且会使.EXE文件运行后每次都会增加304字节,使.EXE文件变得越来越长,直至无法运行。但是如果系统中不存在有该病毒,那么带有该病的.EXE文件的病毒程序部分永远也不会被运行,当然也不会驻留内存,也不会由.EXE文件进行传染。所以这种病毒只有附加在.COM文件后才会有传染,因此造成的危害也就相对小了很多。

## 二、病毒的检测

1、检查系统是否带有该病毒可采用DEBUG检

\*\*\*\*\*

```
05C8h 1, 024 Command C: \ / P / E: 512
0612h 629.472 <largest free area>
654, 336 bytes (639k) (1) total DOS 3.31 conventional memory
629, 472 bytes (615k) (2) largest executable program
0 bytes Extended (AT / 286 / 386) memory, reported by BIOS
320K bytes free, reported by XMS driver version 2.0 (2.50)
HMA exists
```

### (例2)

```
Memory Info V7
(c) 1991 Central Point Software, Inc
Addr Total bytes Program or device driver
```

```
OCC9h 3, 536 Command
ODB3h 1, 120 DIRDEL C: \BIN\WH\DIRDEL.EXE (3)
***** bytes (25091K) (1) total DOS 3.30 conventional
memory.
598, 096 bytes (584k) largest executable program.
384k bytes Extended (AT / 286 / 386) memory, reported by
BIOS.
```

### 例子说明:

- (1) 系统内存总量,一般为640K
- (2) 系统可用内存数,不同操作系统该值不同。
- (3) 染上病毒的可执行文件(dirdel.exe为非法

TSR程序)

120

## 游戏求解

编辑先生:

您好!

我手头有一个游戏软件,名叫《神州八剑》,是智冠科技有限公司1991年出版的。该软共158个文件,2,234,537个字节,1500个密码。这个游戏软件非常好玩,极富创意。

但我在玩的过程中,在第三关(转魂国)找不到魔法登山钩,第五关(减魂国)停时术不会使用,第七关(真刚国)找不到涉水术。致使该游戏玩不下去。

希望能有高手予以指点。(如有需要该软件者,请来信索取,地址:兰州铁路分局,邮政编码:730050)

致

礼!

闫益民

1993.1.30

121

查,方法如下:运行 DEBUG

```
C>DEBUG
-A
XXXX: 100 MOV AX, 3521
XXXX: 103 INT 21
XXXX: 105 INT 3
XXXX: 106
-G (注: XXXX 为段地址)
```

运行上述程序后,若发现 BX=005C,ES 可是任意值,则说明有该病毒。此外运行一个后缀为 .EXE 的文件退出后,若发现文件每次运行均增 304 字节,则说明该系统中有 V304 病毒。

2、检查文件是否有病毒可以用 PCTOOLS 和 DEBUG 作工具。用 PCTOOLS 检查所有的 .COM 和 .EXE 文件,先选择某一目录,并选择出该目录下的所有 .EXE 和 .COM 文件,然后用 F 命令查找十六进制数字 8B3601,01BF00,015703, F7B903, 00, 若发现有该字节数字,而且用 E 命令显示对应的文件内容发现这些字节的前三个字节内容和文件开始的前三个字节的内容完全相同,则说明对应文件有 V304 病毒。用 DEBUG 检测文件是否有 V304 病毒时,对于 .EXE 文件需先将文件变成非 .EXE 文件,然后用 S 命令查找上述的几个特征字节,若发现有这些特征字节,而且这些特征字节的前三个字节与文件头的前三个字节完全相同则说明该文件已带有 V304 病毒,但它不会被执行,因病毒不破坏 .EXE 文件的文件头,也不会修改程序的入口指针。对于 .COM 文件可直接检测。

方法如下:

```
C>DEBUG 文件名.COM
-T
-U
XXXX: XXXX MOV SI, [0101]
XXXX: XXXX MOV DI, 0100
XXXX: XXXX ADD SI, DI
XXXX: XXXX MOV CX, 0003
XXXX: XXXX PUSH CX
XXXX: XXXX CLD
XXXX: XXXX REPZ
XXXX: XXXX MOV SB
(段地址) (偏移地址)
```

若发现有上述汇编指令则说明有该病毒。

### 三、病毒的清除

因为 V304 病毒均附在原文件的尾部,而且对 .EXE 文件的原内容没有改变,对 .COM 文件也只修改其前三个字节,而且这三个字节也是可见的,因此利用 DEBUG 即可方便地清除它。具体按下面方法进行:

用干净的系统盘起动,然后运行 DEBUG 调入带有该病毒的文件,然后设法找到原文件的长度,重写文件,即可得到清除了该病毒后的文件。对于 .EXE 文

件,须先将其变成非可执行文件,然后找到与原文件头前三字节相同的位置,并确定其后部分是病毒程序(即含有检测时的字节特征),以此计算出原文件的长度,重写文件,最后将修改并清除了病毒程序后的非可执行文件拷贝回原文件即可。对于 .COM 文件只要找到原文件的前三个字节并写到偏移地址为 100H 处,修改文件长度。重写文件即可。现以一 .COM 文件带有该病毒后的清除方法作介绍。(假设在 C:\DOS\MORE.COM 文件中带有该病毒)

清除病毒步骤如下:

1、用干净无病毒的系统盘启动机器。

2、A>C:

C>CD DOS

3、C>A: DEBUG MORE.COM

4、-U100 107

```
XXXX: 0100 E91A01 JMP 021D
XXXX: 0103 21 3D AND [DI],DI
XXXX:0105031F ADD BX,[BX]
XXXX:01077409 JZ 0112
```

其中 101-102 的 2 个字节内容即为原文件长度(也可直接用 R 命令显示出 CX 寄存器的内容,然后减去 130H 得到原文件长度),修改 CX 内容为原文件长度。

-RCX

0254

:11A (改为原文件长)

5、找到原文件的前三字节内容,它放在跳转指令所指地址的前三个字节中,将它写到 100 处。

-D21A 229

```
B4 30 CD 8B 36 01 01 BF 00 01 57 03 F7 B
9 03 00
```

原文件前三字节      病毒特征字节

-e100

-1528: 0100 E9.B4 1A.30 01.CD

恢复原文件前三字节内容

6、存盘退出

-W(写入原文件)

-Q(退出 DEBUG)

至于 .EXE 文件,因病毒没有修改其文件头,故清除方法更加简单,大家可以按照以上的介绍进行清除该病毒。笔者曾用以上清除病毒的方法对本公司的微机上的所有染上该病毒的文件进行消毒,无一误操作,而且操作简便。当然对于文件较多的目录,需逐一清除该病毒亦是相当费时的,遇到这种情况可以找到原文件并将其覆盖带病毒文件,但一时找不到原文件的可以按上述方法清除病毒。大家不妨一试。



# 怎样从内存中直接得到 COMMAND.COM 的清单

四川省成都热电厂 (610051) 宋捷

PCDOS 是最基本的系统软件, 它由四部分组成: 即引导程序 BOOT、IBMBIO.COM、IBMDOS.COM 和 COMMAND.COM, 其中 COMMAND.COM 是 DOS 最后进驻内存的部分, 它承担着各种内、外部命令、批命令、管道及改向操作等的读取、识别、分解、执行等任务, 如最常用的 DIR、COPY 等命令主要是由 COMMAND.COM 借助 INT21H 来完成的, 因此, 如果没有 COMMAND.COM 命令, 我们就无法方便地运行各种各样的软件和程序, COMMAND.COM 是 PC DOS 一个重要组成部分。

COMMAND.COM 分为常驻、初始化和暂驻三部分, 在 A>提示符下键入 DEBUG COMMAND.COM 后, 常驻部分位于偏移 100-E2F, 初始化部分在 E30-175F, 暂驻部分则是 1760-63DB, 系统开工后, 常驻部分位于内存低端, 初始化部分位于后来的用户区, 将被应用程序覆盖, 暂驻部分在内存高端, 用 DEBUG COMMAND.COM 固然可以得到三个部分的清单, 但至少有以下两个缺陷:

1. 数据区很多重要数据全为 0, 不利于观察各种命令的动态执行过程;

2. 暂驻部分在初始化过程中曾经由内存低端全部移到高端, 这样动、静态条件下暂驻部分的程序及数据地址完全不同。

因此, 常驻和暂驻部分的清单可用 DEBUG 直接从内存中得到, 只有初始化部分的清单才需用 DEBUG 直接将文件 COMMAND.COM 装入来得到, 例如一台 640KRAM 的不带硬盘 IBMPC/XT 机就可用以下方法从内存中直接得到 COMMAND.COM 的清单:

建立一个批处理文件 PROG.BAT (附后), PROG.BAT 首先利用 DOS 的重定向功能生成一个文件 TEMP1 作为获取 COMMAND.COM 常驻和暂驻部分的清单的输入, 然后利用 DOS 的管道技术将 COMMAND.COM 在内存中的常驻和暂驻部分的清单输出到文件 CXQD1, 接着利用 DOS 的重定向功能生成一个文件 TEMP2 作为获取 COMMAND.COM 初始化部分的清单的输入, 最后直接用 DEBUG 装入 COMMAND.COM, 再利用 DOS 的管道技术将 COM-

MAND.COM 初始化部分的清单输出到文件 CXQD2, 注意 PROG.BAT 在生成清单时数据区用 DEBUG 的转储命令 D 来获取, 程序区用 DEBUG 的反汇编命令 U 来获取, 其中 D 命令和 U 命令后所跟的 COMMAND.COM 各部分动态与静态地址如下:

1. 常驻部分: 动态条件下的段址是 0B6F, 偏移 00-FF 是 PSP, 100-7BB 为程序区, 7BC-EIF 为数据区, 0B6F:0E20 为用户区第一内存控制块 (MCB);

2. 暂驻部分: 动态下的段址是 9B2A, 偏移 00-FF 为第二 PSP, 实际上仅供暂驻存放命令参数, 不包含真正 PSP 的其他信息, 100-3728 为程序区, 但 23C5-23CF, 2506-250F 两处全为 0 (未使用), 在反汇编时应注意这两个地方, 3729-465F 为数据区, 9B2A:4D90-A000:0000 为 640K 内存的最高端;

3. 初始化部分: 偏移 E30-145C 为程序区, 145D-175F 为数据区。

将文件 DEBUG.COM 拷贝到 A 盘后即可运行 PROG.BAT, 运行完后, 就可用各种字处理软件来查看 COMMAND.COM 的常驻、暂驻和初始化部分的清单了, 需要时也可将其打印出来。

以上程序在 IBMPC/XT 主机上实现, 操作系统为 DOS3.3.

```
A> TYPE PROG.BAT
@ECHO OFF
ECHO D B6F:0 FF> TEMP1
ECHO U B6F:100 7BB>> TEMP1
ECHO D 7BC EIF>> TEMP1
ECHO D 9B2A:0 FF>> TEMP1
ECHO U 9B2A:100 3728>> TEMP1
ECHO D 9B2A:3729 465F>> TEMP1
ECHO Q>> TEMP1
TYPE TEMP1; DEBUG>CXQD1
ECHO D E30 145C> TEMP2
ECHO U 145D 175F>> TEMP2
ECHO Q>> TEMP2
TYPE TEMP2 DEBUG COMMAND.COM>CXQD2
DEL TEMP1
DEL TEMP2
```

# IBM PC BASIC 语言

## 调用鼠标的程序设计方法

北京市总后医学专科学校电教中心 陈惠生

【摘要】本文介绍在 BASIC 程序中调用鼠标的方法，并给出一个利用鼠标作图的 BASIC 演示程序。

鼠标，也称鼠标器，作为一种微机的辅助输入设备在系统应用软件中被广泛地采用。对于今天的计算机操作人员来说，它的重要性不亚于键盘，特别是在图形处理系统中，它的应用使计算机操作人员的工作更加得心应手。如何在自己的应用软件中调用鼠标器，这是广大计算机软件设计人员所关心的问题。为了使计算机初学者便于理解，本文介绍 BASIC 语言调用鼠标的方法。

### 一、鼠标器的功能调用

在鼠标器驱动程序中，提供了 35 个功能调用，调用这些功能有三种方法：鼠标器菜单、鼠标器函数和 33H 中断。本文只介绍利用 33H 中断调用鼠标的方法。

在调用中断时，鼠标功能调用号放在 AX 寄存器中，其它参数分别放在 BX、CX 和 DX 中；调用后出口参数分别在 AX、BX、CX 和 DX 中。下表列出部

#### 鼠标功能调用表

##### 1、功能 0 鼠标初始化和取状态

入口参数：AX=0

出口参数：AX=鼠标器状态；BX=鼠标器按钮数目

功能描述：如果鼠标器安装正确，出口参数 AX=-1，否则为 0；BX 为鼠标器的按钮数目。

##### 2、功参 1 显示光标

入口参数：AX=1

出口参数：无

功能描述：调用此功能后鼠标器方可显示光标。

##### 3、功能 2 消除光标

入口参数：AX=2

出口参数：无

功能描述：调用此功能后消除鼠标器的光标。

##### 4、功能 3 取按钮状态和鼠标器位置坐标

入口参数：AX=3

出口参数：BX=按钮状态；CX=光标水平坐标；DX=光标垂直坐标。

功能描述：BX=1 表示按钮按下，否则按钮没按下；CX 和 DX 分别为光标的水平和垂直坐标。

##### 5、功能 4 设置光标位置

入口参数：AX=4；CX=水平坐标；DX=垂直坐标

出口参数：无

功能描述：此功能把光标设置在虚拟屏幕的指定位置上。

##### 6、功能 7 设置光标移动水平极限

入口参数：AX=7；CX=MINX；DX=MAXX

出口参数：无

功能描述：此功能设置了光标在屏幕上的水平坐标的最小值和最大值，可以限制光标的水平移动范围。MAXY 不大于屏幕 X 方向的最大分辨率。

##### 7、功能 8 设置光标移动垂直极限

入口参数：AX=8；CX=MINY；DX=MAXY

出口参数：无

功能描述：此功能设置了光标在屏幕上的垂直坐标的最小值和最大值，可以限制光标的垂直移动范围。MAXY 不大于屏幕 Y 方向的最大分辨率。

##### 8、功能 9 设置图形光标

入口参数：AX=9；BX=光标热点水平值；CX=光标热点垂直值；DX=指向屏

幕掩码的光标掩码的指针。

出口参数: 无

功能描述: 此功能定义了图形光标的形状、颜色和中心。此功能不能自动显示所定义的图形光标, 需调用功能 1 来显示。光标热点是相对于光标左上角的距离。

## 9. 功能 10 设置文本光标

入口参数: AX=10; BX=光标类型; CX=屏幕掩码或起始扫描线; DX=光标掩码或结束扫描线。

出口参数: 无

功能描述: BX=1 选择硬件正文光标, 为 0 选择软件正文光标。

## 二、BASIC 语言调用 INT33H

鼠标驱动程序是以 33H 中断服务程序的方式驻留内存, 只要能够调用 33H 中断, 即可调用鼠标功能。

在汇编语言或其他高级语言中可利用中断函数调用 INT33H 中断, 但是在 BASIC 语言中没有中断调用功能, 因而不能直接调用 INT33H 中断。

BASIC 语言具有调用机器语言的功能, 只要给出入口地址和段地址, 即可利用 CALL 语句调用 INT 33H 中断, 从而达到调用鼠标功能的目的。

在 MS-DOS 中 256 个中断向量地址是放在 0: 0 开始内存空间, 每个中断向量占 4 个字节, 两个字节的的中断服务程序偏移地址和两个字节的段地址。对于系统没有使用的中断向量有两种表示方法。一是中断向量的段地址和偏移地址为零, 一是中断向量指向的中断服务程序只是 IRET 指令 (16 进制代码是 OCFH)。通过利用 PEEK() 函数读取中断向量地址判断中断服务程序是否是一句 IRET 指令就可知道鼠标器驱动程序安装与否。PRO1 给出了检查是否安装了鼠标驱动程序。

```
1000 REM PRO1 检查鼠标驱动程序安装
1010 DEF SEG=0 '定义段地址为 0
1020 REM 取 INT33H 的段地址
1030 M33SEG%=256*PEEK(&H33*4+3)+PEEK(&H33*4+2)
1040 REM 取 INT33H 的偏移地址
1050 M33OFF%=256*PEEK(&H33*4+1)+PEEK(&H33*4)+2
1060 REM 检查鼠标驱动程序是否安装
1070 IF (M33SEG% OR M33OFF%) AND (PEEK(M33OFF%-2 <>
&HCF) THEN 1100
1080 PRINT "没有安装鼠标驱动程序"
1090 END
1100 PRINT "鼠标驱动程序已经安装!":GOTO 1090
```

在用 CALL 命令调用 33H 中断时需要传递 4 个整型参数 P1%、P2%、P3%和 P4%, 它们分别对应着鼠标功能调用中参数寄存器 AX、BX、CX 和 DX。调用格式为 CALL M33OFF% (P1%、P2%、P3%、

P4%)

由以上分析可知只要把鼠标功能号赋给整型参数变量 P1%, 其他参数赋给 P2%、P3%和 P4%, 然后用 CALL 命令调用即可。

PRO2 演示初始化鼠标器、显示光标并限定光标移动范围。

```
1500 REM PRO2
1510 DEF SEG=0
1520 REM 取 INT33H 的段地址
1530 M33SEG%=256*PEEK(&H33*4+3)+PEEK(&H33*4+2)
1540 REM 取 INT33H 的偏移地址
1550 M33OFF%=256*PEEK(&H33*4+1)+PEEK(&H33*4)+2
1560 P1%=0:CALL M33OFF%(P1%,P2%,P3%,P4%)
1570 DEF SEG '恢复 BASIC 数据段地址
1580 IF P1%=-1 1600
1590 PRINT "没有鼠标器!":END
1600 REM 定义水平移动范围为 10-630 象素
1605 P1%=7: P3%=10: P4%=630: CALL M33OFF%(P1%, P2%,
P3%, P4%)
1608 REM 定义垂直移动范围为 10-180 象素
1610 P1%=8: P3%=10: P4%=180: CALL M33OFF%(P1%, P2%,
P3%, P4%)
1615 REM 定义光标显示在 320,90 处
1620 P1%=4: P3%=320: P4%=90: CALL M33OFF%(P1%, P2%,
P3%, P4%)
```

注意在利用鼠标绘图时, 就先用功能 2 关闭光标完成一个绘图动作后再用功能 1 显示光标, 否则会破坏屏幕绘出的图形。PRO3 是一个鼠标画线演示程序, 略加修改就可以成为一个鼠标绘图程序。

```
1000 REM PRO3 鼠标画线演示程序
1002 REM LFBTN 画线;RTBTN 结束;MIBTN 清屏
1010 DEF SEG=0:KEY OFF
1020 DEF INT A-Y:SCREEN 2
1030 M33SEG=256*PEEK(207)+PEEK(206)
1040 M33OFF=256*PEEK(205)+PEEK(204)+2
1050 DEF SEG=M33SEG
1060 CLS:BJ=0
1070 IF (M33SEG OR (M33OFF-2)) AND (PEEK(M33OFF-2) <>
207) THEN 1100
1080 PRINT "没有安装鼠标驱动程序!"
1090 END
1100 P1=0:CALL M33OFF(P1,P2,P3,P4)
1110 IF P1=-1 THEN 1130
1120 PRINT "没有鼠标器!":GOTO 1090
1130 P1=1:CALL M33OFF(P1,P2,P3,P4)
1140 P1=7:P3=10:P4=630:CALL M33OFF(P1,P2,P3,P4)
1150 P1=8:P3=10:P4=180:CALL M33OFF(P1,P2,P3,P4)
1160 P1=4:P3=320:P4=90:CALL M33OFF(P1,P2,P3,P4)
1170 P1=3:CALL M33OFF(P1,P2,P3,P4)
1180 LOCATE 1,60:PRINT P2,P3,P4
1190 IF P2>1 THEN 1220
1195 P1=2:CALL M33OFF(P1,P2,P3,P4)
1197 LINE-(P3,P4),1:P1=1:CALL M33OFF(P1,P2,P3,P4)
1198 GOTO 1170
1200 IF P2=4 THEN CLS ELSE IF P2=2 THEN 1090
1210 GOTO 1170
```

# 一种给电路板设计软件包TANGO

## 建立汉字库的方法

李毅军

电路板设计软件包 TANGO 自面世以来, 以其方便、易学、实用、快速的特点深受广大电气设计人员的欢迎, 但美中不足的是该软件没有汉化, 因此无法在绘制的图上标注汉字。为了弥补这一不足, 我经过摸索, 发现可以用建原理图设计软件 TANGO Schematic 元件库的方法在 TANGO 软件包里建立一个汉字库, 具体方法如下:

1. 使用文本编辑软件 (如联想卡、金山卡、Wordstar 等) 或行编辑方法去编辑一个文本文件, 该文本文件的扩展名为 SRC, 这样这个文件的名字可取为 HZ.SRC。该文件的编制格式为 (以编制一个汉字为例):

(1) 文件中每一个汉字字符由若干行来描述, 各个汉字之间由一空行相区分。

(2) 对汉字字符的描述方式为:

A. 第一行: 将该汉字的标记符 (最好是该汉字的拼音符号) 放在单引号内。为使所编制的汉字在所建的汉字库内是唯一的, 对于同音的汉字最好在汉字标记符上有所区别。然后在汉字标记符的后面跟上特定标志字 "POWER".

B. 第二行: 使用三个数字来确定所编汉字在显示屏上所占尺寸的大小。该尺寸的单位为 0.1 英寸 (即 TANGO 原理图上一个显示栅格的长度)。第一个数字是汉字字符所占据的横向尺寸; 第二个数字是纵向尺寸; 第三个数字原是用来表示在建立元件库时一个封装里相同电路的套数, 在编制汉字时为了使所编制的源程序符合 TANGO 里建元件库的规范, 应将第三个数字定为 1。对于用作原理图上标注的汉字, 其横向尺寸和纵向尺寸可分别选取为 2 和 3 个显示栅格单位的长度, 这样打印出来的汉字为 4 号汉字。

(3) 根据所制汉字的形状编制该汉字的点阵图, 点阵图的规则为:

A. 以字符 "bitmap" (点阵图) 为引导行;

B. 在点阵图每行的开头用大括号 "{" 括住该行的行号;

C. 点阵图中一行里每 10 个作图字符等于一个显示栅格的长度, 纵向每 10 行的长度等于 TANGO 中一个显示栅格的长度。由于最小光标单位就是显示栅格单位, 因而汉字点阵图上每行的作图字符数及作图行数必为 10 的整数倍, 且这两个整数倍数字必与上面描述元件符号横向及纵向的尺寸相一致;

D. 在点阵图中用 "." 表空白, 用 "#" 表示汉字的画点。图 1 是编制一个 "制" 的例子 (用金山卡的文字文件编辑功能所编)。

```
'ZHI' POWER
2 3 1
BITMAP
{ 0}.....
{ 1}.....
{ 2}.....
{ 3}.....
{ 4}.....
{ 5}..#..#.....#..
{ 6}..#..#.....#..
{ 7}..#####.....#..
{ 8}..#..#.....#..
{ 9}..#..#.....#..#..
{10}.....#.....#..#..
{11}.....#.....#..#..
{12}..#####.....#..#..
{13}.....#.....#..#..
{14}.....#.....#..#..
{15}.....#.....#..#..
{16}.....#.....#..#..
{17}..#####.....#..#..
{18}..#..#.....#..#..
{19}..#..#.....#..#..
{20}..#..#.....#..#..
{21}..#..#.....#..#..
{22}..#..#.....#..#..
{23}..#..#.....#..#..
{24}..#..#.....#..#..
{25}.....#.....####..
{26}.....
{27}.....
{28}.....
{29}.....
{30}.....
```

图 1 编制一个 "制" 字的源程序

# 激光加密盘的解密方法及其备份

广州空司 韩文生

软盘加密的方法很多，可以分为软加密和硬加密。软加密就是在软盘上产生一个特殊的磁道或扇区的格式，如额外磁道技术、额外扇段技术、扇段对齐技术、宽磁道技术、弱位技术等，当标准的复制程序遇到这些“坏”的扇段或磁道时便停止复制，因为它不能将源盘中的数据读出；或者复制出一个没有“缺陷”的软盘。但当使用高级拷贝程序如 COPYWRIT 时，就可以复制出和源盘一样的目的盘，因此，这些方法目前已经难以继续使用了。

激光孔加密盘技术是近年来常用的效果很好的加密方法，它可以有效地防止被加密软盘的拷贝。这种加密方法是利用激光方向性好、能量高、均匀的特点，在软盘上打孔作为标记，造成一个必须由硬件复制的坏扇段。同时，在被保护的软件中配上加密程序，运行时，需要把加密的盘插入驱动器中，被加密的文件运行时寻找指定的坏扇区，若坏扇区存在，必然会产生 CRC 错误，但加密程序认为它是对的，程序继续运行。若在指定的扇区没有 CRC 错误，则认为此软盘是复制品，拒绝调出程序。

由于激光加密改变了软盘的物理特性，使用普通软盘机便无法拷贝，只能采取解密（破译）的方法，修改或者删除嵌入被保护软件中的保护程序，使其不读盘或不检查软盘上的坏扇区。其方法通常是利用 DEBUG，当跟踪到读盘并判断是否是源盘时，将源盘所读出的内容存入一个文件，修改源盘读盘程序，去掉判激光孔错误标志，改用打开文件读数据到相应缓冲区

的形式，就能较好地解决加密问题，同时，源盘就可以被拷贝了。如果在加密程序中增加了修改断点中断功能，利用时钟中断，使跟踪无法进行时，我们就要分段跟踪，找到时钟中断所设置的入口地址去运行，而不直接去跟踪时钟中断。例如：

加密源程序：

.....

```
0151 PUSH DS
0152 MOV ES,AX
0154 XOR BX,BX
0156 MOV DX,BX
0158 MOV CX,2701
015B MOV AX,0201
015E INT 13
0160 CMP AH,10
0163 JNZ 016D
0165 ES:
0166 CMP BYTE PTR [BX],90
0169 JNZ 016D
016B JMP 016E
016D IRET
016E POP DS
016F XOR BX,BX
0171 MOV DX,0000
0174 MOV CX,2702
0177 MOV AX,0201
017A INT 13
017C JB 016F
.....
```

修改后的程序：

.....

```
0151 JMP 015B
0153 DB 'HAN.EXE' 00
015B MOV ES,AX
015D MOV AX,3D00
0160 MOV DX,0153
0163 INT 21
0165 PUSH DS
0166 PUSH ES
0167 POP DS
0168 MOV BX,AX
016A MOV AH,3F
016C MOV DX,00
016F INT 21
0171 MOV AH,3E
0173 INT 21
0175 POP DS
0176 NOP
.
.
.
017D NOP
.....
```

注：HAN.EXE 为读盘读出的数据后存盘所取的文件名。此方法在 IBMPC/XT 机上调试通过。

126

\*\*\*\*\*

在汉字库源文件 HZ.SRC 编好后，即可进入 TANGO 软件包中的原理图设计软件 Schematic 执行 COMPILE 命令，显示屏上会提示你输入需转换的源文件名，输入源文件名后回车，稍候片刻即可在显示屏上看到你所编辑的源文件 HZ.SRC 已被转换成 TANGO 原理图编辑软件 SCHEM-EDIT 可接收的库文件 HZ.LIB，其扩展名 LIB 是在源文件的转换中由

COMPILE.COM 自动加上去的。至此，你已经成功地在 TANGO 原理图设计软件 Schematic 中建立了一个汉字库，若还需在以后的使用中扩展该字库，可以使用 TANGO-Schematic 中的 DECOMP 命令将 HZ.LIB 转换成其源文件 HZ.SRC 后使用前面所提到的文本编辑软件进行编辑，然后再使用 COMPILE 命令将 HZ.SRC 转换成 HZ.LIB。

125

# 也谈用软盘启动CCBIOS2.13H汉字系统

中国工商银行青海省西宁市支行科技部 徐小平

本刊 1992 年第 5 期刊登的《用软盘启动 CCBIOS2.13H 汉字系统》一文介绍了一种通过修改 2.13H 的读显示字库模块 FILE1A.COM 实现用软盘启动该系统的方法。这的确是一种行之有效的好方法。但尚有两点不足：

一、系统在工作中要随时从软盘读取二级字库，不仅速度缓慢，而且会增加软盘磨损。因软盘始终占据驱动器，使只有单一驱动器的计算机无法使用其他软盘。

二、需要修改原系统模块。

笔者向来非常喜爱 2.13 系统，并对其各种版本做过仔细研究：2.13H 系统提供了显示字库的 5 种安装方案，分别由 5 个模块来实现：

一、显示字库全部驻留硬盘

使用模块为：FILE0A.COM，为了提高显示速度，系统在内存开辟了一个可以改变大小的常用字库区，其取值范围为 50~800 个汉字，由启动批处理文件占的 FILE0A a2 的取值范围为 1~9、A~G（即 10~16），每一个单位代表 50 个汉字。

二、一级字库调入内存，二级字库驻留硬盘

使用模块为：FILE1A.COM，这时仅一级字库占用内存约 130K，既部分地节省了内存，又明显提高了显示速度，是最常用的一种方法。

三、显示字库全部驻留内存

使用模块为：FILE2.COM 这时两级字库共占内存约 256K，显示速度也最快。

四、显示字库装入虚拟盘

使用模块为：FILE3.COM 这时计算机应有 1M 以上的内存，一般建立大于 256K 的虚拟盘。

五、使用汉卡字库

使用用模块为：FILE4.COM

可见：只要采用显示字库全部驻留内存或装入虚拟盘的方法，就可方便的实现软盘启动。成功关键就在于批处理文件的正确编制。

我曾做了两种启动软盘，其特点是可直接启动计算机并进入 2.13H 系统，启动完毕后软盘即可抽出。且不用修改任何文件，并具有原系统除 24、40 点阵打印外的其他所有功能，如联想、预选、内部词组、五笔字型接口等。还可以实现 16 点阵各种字号的普通打印及特殊功能打印。经两年多的使用，效果良好。

第一种软盘采用了显示字库全部驻留内存的方法。适用于内存为 640K 的 PC/XT 机。全部文件放在两张 360KB 软盘上，启动时依次使用。一号盘上含有 DOS3.2 系统，二号盘上有 CC11、CV26 等多种显示模块，可适用于各种显示器类型。如只在一种机型上使用，可只拷入相应的模块。软盘剩余的空间可存放其他常用文件。两张盘上都要拷入自动批处理文件。（文件清单及批处理文件附后）

第二种采用了显示字库装入虚拟盘的方法，适用于内存存在 1M 以上的 286、386 机。先制作一张 1.2MB 的 DOS3.X 系统盘，并拷入 VDISK.SYS 文件。然后将 2.13H 系统除 24 点阵字库外的其他文件全部拷到这张盘上，（也可选主要文件）并编好自动处理文件，还要在 CONFIG.SYS 中加入一句：DEVICE = VDISK.SYS 384 512 64 / E 来定义虚拟盘。

制作时要注意以下几点：

软盘应含有 DOS 系统，以便直接启动计算机。注意 ANSI.SYS、CONFIG.SYS 两个文件必不可少，而且一定要拷自 2.13H 系统盘。也不要不用不含上述两个文件的一般系统盘启动后再调 2.13H 系统，否则会造成死机。

如果你想使用五笔字型，可将“外部输入方式盘”中的 KWB.COM 和 WBZX.COM 两个文件拷入你的软盘，并顺序执行：KWB 及 WBZX，然后按 Alt+F4 就可进入五笔字型输入状态。

第一种盘（由 2 张 360K 软盘组成）

一号盘文件清单：

|          |      |        |
|----------|------|--------|
| COMMAND  | COM  | 23791  |
| ANSI     | SYS  | 1664   |
| CONFIG   | SYS  | 27     |
| SYS      | .COM | 4620   |
| HZK16    |      | 261696 |
| FILE2    | COM  | 136    |
| AUTOEXEC | BAT  | 1024   |

二号盘文件清单：

|      |     |       |
|------|-----|-------|
| CCCC | COM | 43270 |
| CC11 | COM | 9879  |
| CC16 | COM | 9879  |
| CC25 | COM | 9879  |

# 真正、彻底地从软盘启动2.13H汉字系统

杭州钢铁厂计划处 金尧岗

本刊在 92 年第 5 期的“电脑用户”栏目中，曾刊登了一篇文章，介绍如何在软盘上启动 2.13H 汉字系统，主要做法就是修改 FILE1A.COM 文件。笔者根据该方法修改之后，确实能在软盘上启动 2.13 系统了，但笔者发现，实际上 CCCC.COM 程序仍然还要读硬盘，如果不对其进行修改，则在真正无硬盘环境下，还是无法从软盘启动 2.13H 系统。

分析了 CCCC.COM 程序后发现，读硬盘的子程序段是经求反加密后放在程序中的，先解密，然后将其对硬盘的操作改为对 A 盘的操作，并跳过读 C 盘 DOS 引导区的程序段再进行一次求反工作（实际上就又加密了），这时重新写回 CCCC.COM 就可以了。

总之，在 DOS 下用 DEBUG 工具对文件 CCCC.COM

作如下修改：

```
C>DEBUG CCCC.COM
-E 20A FF
-E 20E 14 D2
-E 260 FF
-E 274 FF
-A 135
0CB7: 0135 MOV WORD PTR [2795],213E
0CB7: 013B JMP 13F
0CB7: 013D ^C
-N CCCCA.COM ; 另取名为 CCCCA.COM 以示区别
-W
-Q
这样，CCCCA.COM 就是所要的文件了。
```

127

```
CE21-    COM      9879
CE25     COM      9879
CE26     COM      9879
CH11     COM      9879
CH21     COM      9652
CH25     COM      10001
CL25     COM      9879
CV26     COM      9879
INT10F   COM      2737
PRTA     COM      4227
YX1      COM      1152
LX1      COM      38567
FILE16B  COM      2417
CN       COM      576
CM       COM      275
AUTOEXEC BAT 1024
```

AUTOEXEC.BAT 文件清单：

```
ECHO OFF
CLS
ECHO CCBIOS 2.13H
ECHO PLEASE WAIT
FILE2 2
ECHO Please inser 2# diskette in drive A:
PAUSE
CCCC
CC11 (注：可根据显示类器型换用其他模块)
ECHO CCBIOS 2.13H 汉字系统启动完毕
```

```
INT10F
PRTA
YX1
LX1
FILE16B
第二种盘 (一张 1.2M 软盘)
AUTOEXEC.BAT 文件清单:
ECHO OFF
CLS
ECHO CCBIOS 2.13H
ECHO PLEASE WAIT
IF NOT EXIST D: HZK16 COPY HZK16 D:
FILE3 D2
CCCC
CV26 (注：可根据显示类器型换用其他模块)
ECHO CCBIOS2.13H 汉字系统启动完毕
INT10F
YX1
LX1
PRTA
FILE16B
KEY
CONFIG.SYS 文件清单:
FILES=20
DEVICE=ANSI.SYS
DEVICE=VDISK.SYS 384 512 64 / E
```

128

# DBASE中的“特技”清屏演示程序

江苏省海安县中学

郑明达

“屏幕的显示及清屏方法”是度量一个软件界面是否友好的一个很重要的方面。然而，在大量的 DBASE 系统管理软件中，由于受显示器分辨率的影响（中分每屏仅能显示 10 行），因而不少软件的屏幕处理显得过于呆板、简单。为此，笔者结合 DBASE 本身的特点，运用 2.13H 系统的特殊显示功能，终于找出了 DBASE 中的一些“特技”清屏方法，并完成了一演示这些清屏方法的程序，现整理写出来，供大家参考。

本程序演示了 DBASE 中的常用“特技”清屏十法，只要你选中清屏功能键码，机器便会在屏幕所开的一个窗口内演示本清屏过程。这些清屏方法大体上可分为三种：

## 一、“扫描式”清屏（计 4 种）

该清屏方法是利用其本身的显示命令完成的。

@CL,CR SAY " "

实现的基本方法是：不断改变命令中行与列的坐标值以达到在不同的位置显示空格的目的，从而清屏其原有位置上的内容。具体可参看演示程序。

## 二、“滚行式”清屏（计 2 种）

该清屏方法是利用 2.13H 特殊显示命令来完成的。

即：CHR(14)+“I 寄存器参数串”

实现的基本方法是：由于该命令执行 INT10 中断，可以完成屏幕行的上滚和下滚，这样，我们只要控制屏幕窗口中的行上滚或下滚相应的次数，即可实现清屏。具体可以参看演示程序。

## 三、“拉窗帘式”清屏（计 4 种）

该清屏方法是利用 2.13H 特殊显示命令：向矩形区填线的功能来实现的。即：

CHR(14)+“V 色号,左下角 X,Y,宽,高,类型,间隔,线宽”

实现的基本方法是：由于该命令可以用各种线填满整个矩形块，因而设色号参数为 0，线间隔为 0，让 X, Y 取得屏幕窗口的左下角坐标；矩形宽高定为屏幕窗口的大小，再执行该命令即可。具体可参看演示程序。

本程序运行环境：系统 (CCDOS 2.13H)；软件 (DBASE 或 DBASE plus)。

附：“特技”清屏演示程序

```
set talk off
? chr(14)+"q]"
a0="从左上角到右下角扫描式清屏"
a1="从右下角到左上角扫描式清屏"
a2="从右上角到左下角扫描式清屏"
a3="从左下角到右上角扫描式清屏"
a4="上滚行式清屏"
a5="下滚行式清屏"
a6="下拉窗帘式清屏"
a7="右拉窗帘式清屏"
a8="右斜拉窗帘式清屏"
a9="左斜拉窗帘式清屏"
do while .t.
clear
do while .t.
gl=" "
@ 2,20 say "请选择清屏功能键码(A-K):" get gl pict "x"
k=1
i=0
do while i<=9
j=str(i,1)
k=1-k
@ int (i/2)+4,5+k*35 say chr(65+i)+"-"+a&j
i=i+1
enddo
@ 9,25 say "k--退出演示程序"
read
gl=upper(gl)
if gl="K"
set talk on
clear
retu
endi
if asc(gl)>=65.and.asc(gl)<=74
exit
endi
enddo
clear
j=str(asc(gl)-65,1)
@ 1,5 say a&j+":"
? chr(14)+"C5]"
? chr(14)+"D158,148]"
? chr(14)+"B204,102]"
i=1
do while i<=5
@ i+2,21 say "第"+str(i,1)+"行欢迎您使用窗口清屏"
i=i+1
enddo
@ 9,30 say "按任一键,计算机将在该窗口内演示本清屏过程!"
IF ASC(GL)=65.OR.ASC(GL)=67
A=3
K1=8
K3=1
ENDI
IF asc(gl)=66.or.asc(gl)=68
a=7
k1=2
k3=-1
```



```

endi
if asc(gl)=65.or.asc(gl)=68
    b=21
    k2=44
    k4=1
endi
if asc(gl)=66.or.asc(gl)=67
    b=44
    k2=20
    k4=-1
endi
set cons off
wait
set cons on
if asc(gl)<=68
    do whiel a<>k1
        cl=b
        do while cl<>k2
            @ a,cl say " "
            cl=cl+k4
        endd
        a=a+k3
    endd
endi
@ 0,0 say " "
if asc(gl)=69
    i=1
    do while i<=5
        ? chr(14)+"I6,1,,,3,21,7,44]"
        i=i+1
        j=1
        do while j<=20
            j=j+1
        endd
    endd
endi
if asc(gl)=70
    i=1
    do while i<=5
        ? chr(14)+"I7,1,,,3,21,7,44]"
        i=i+1
        j=1
        do while j<=20
            j=j+1
        endd
    endd
    @ 0,0 say " "
endd
endi
if asc(gl)=71
    ? chr(14)+"v0,160,146,200,96,1,0,1]"
endi
if asc(gl)=72
    ? chr(14)+"v0,160,146,200,96,3,0,1]"
endi
if asc(gl)=73
    ? chr(14)+"v0,160,140,200,96,3,0,1]"
endi
if asc(gl)=74
    ? chr(14)+"v0,160,146,200,96,4,0,1]"
endi
@ 5,28 say"窗口内已清! "
j=1
do while j<=100
    j=j+1
endd
endd

```

## MS-DOS5.0 下使用

### 金山 DOS 5.1

#### 中文系统

吉林省四平市电视台 朱丰池

MS-DOS5.0 是 Microsoft (MS) 公司为 PC 机用户推出的最新操作系统版本。尽管 5.0 版本的内部结构与老版本差别不大,但它确实为一精巧的操作系统。它的 DOS 内核程序比 4.0 版本小的多,与 3.3 版本基本相同。而且在内存大于 1MB 的 286 以上机型中可通过 CONFIG 中设置 DOS=HIGH 语句,将 DOS 推入高内存运行;通过新增加的命令 HIMEM 可方便的管理扩展内存。如果用户的机器是 386 以上机型,还可通过 EMM386 设置适当的参数管理高内存和用扩展内存模拟扩充内存。经过合理优化的系统可使用户得到 620K 以上的基本内存空间。还有一些新版本特有的功能,如友好的用户界面 DOSSHELL, DOSSHELL 使用户可以同时运行多道程序;方便的 DOSKEY,允许用户随时定义键盘的宏;在安全上也增加了很多措施,如 MIROR、UNFORMAT 等。总之,新版本 DOS 给用户带来了极大的方便,深受 PC 用户的好评。金山 DOS5.1 软字库系统是在国内深受广大用户喜欢的中文处理系统,但它不能在新版本 MS-DOS 下运行,好多用户深感遗憾。为此,我对其进行了跟踪分析,通过改进,成功地在 MS-DOS5.0 版本下运行,现将改进及在 MS-DOS5.0 下的使用方法介绍给大家。

尽管在 DOS3.3 下运行的大多数程序可不经修改即可在 DOS5.0 下运行,但在运行一些加密程序或在程序中调用了 DOS 的未公开的功能调用的程序时,就可能出现问題。如运行金山 DOS5.1 软字库版本时,返回信息为: XSDOS.LHP 没找到或无 PUC 汉卡,不能进入该中文系统。经分析,发现只有 SPLIB.EXE 与 MS-DOS5.0 版本不兼容。原因是在 SPLIB 中使用了

MS-DOS 功能调用 AH=32H。这是 Microsoft 未公布的只供 DOS 内部使用的功能调用，它返回指定磁盘驱动器的 DPB 值。DPB 中含有该驱动器 DPB 的远指针，DPB 中包括盘类型、扇区与簇的有关信息。虽然好多资料介绍过无文档的 DOS 功能调用，举例很多商品软件都有从容采用的例子，并在 DOS2.0~3.3 版本中运行基本没有遇到不兼容的问题。但这次 MS-DOS5.0 版本至少将 AH=32H 的功能调用的返回信息的格式做了一些修改。下面是分别在 3.3 版本和 5.0 版本下调用 32H 功能调用后返回的 C: 盘 DPB 值的前 20 个参数:

MSDOS3.3 版本返回 DPB 远指针 DS:  
BX=7460H  
-DDS: 7460 747F  
027F: 7460 02 02 00 02 03 02 02 00-02 00 02 A1 00 D4 3F 40  
027F: 7470 81 00 B6 01 70 00 F8 00-80 74 7F 02 00 00 FF FF

MSDOS5.0 版本返回 DPB 远指针 DS:  
BX=13A4H  
-DDS: 13A4 13C3  
011C: 13A4 02 02 00 02-03 02 02 00 02 00 02 A1  
011C: 13B0 00 D4 3F 40 00 81 00 6B-00 70 00 F8 00 C5 13 1C  
011C: 13C0 01 00 00 FF

从上面返回的参数可以看出，3.3 版本的 DPB 指针在 DS: BX=7460H，5.0 版本在 DS: BX=13A4H。从返回的参数上看，两版本的 DPB 中的前 16 个数据内容是相同的，往下差别较大。但通过对 SPLIB 的跟踪发现程序对 DPB 参数的使用只是有限的几个，并且在使用数据中只有两处因在两版本中的偏移量不同，而使 SPLIB 在 5.0 版本中运行失败。其一是程序存取磁盘的介质参数，硬盘为 F8；其二是磁盘类型，硬盘是 81。在 5.0 版本中这两个参数的位置都比 3.3 版本的向后偏移 1。于是，在 5.0 下，对 SPLIB 做如下修改，修改后在 5.0 版本下运行通过。

REN SPLIB.EXE SPLIB  
DEBUG SPLIB

RDS+1000: 在 DS 原值上加上 1000  
原程序需改动的两条指令汇编显示如下:

UDS: 285B  
DS: 285B CMP BYTE PTR [BX+16],F8  
UDS: 28BB  
DS: 28BB MOV AX, [BX+10]  
用 A 或 E 命令分别修改语句为:  
DS: 285B CMP BYTE PTR[BX+17], F8

DS: 28BB MOV AX, [BX+11]  
W; 写回磁盘  
Q; 退出 DEBUG  
REN SPLIB SPLIB.EXE

修改后，我们可进一步通过优化内存，将 SPLIB 推入高内存运行。我们使用的是 386SX 机，VGA 卡，2MB 内存。在 CONFIG 中设置如下:

DEVICE=C:\DOS\HIMEM.SYS  
DEIVCE=C:\DOS\EMM386.EXE 256  
(接上一行后)FRAME=C800 X=A000-C7FF  
(接上一行后)X=CC00-CDFF I=C800-CBFF  
(接上一行后)I=CE00-EFFF RAM  
DOS=HIGH,UMB  
DEVICE=C:\DOS\SMARTDRV.SYS 1024  
DEVICEHIGH=C:\DOS\SETVER.EXE

在 AUTOEXEC.BAT 中设置如下:

ECHO OFF  
PATH C:\DOS; C:\PUC; C:\TO; D:\DOS5;  
C:\DOS\APPEND C:\PUC; C:\TO; D:\DOS5;  
LH C:\DOS\QMOUSE.COM  
LH C:\DOS\DOSKEY  
LH C:\PUC\SPLIB.EXE  
ECHO ON

通过上述设置，系统启动后，西文下可得 600K 基本内存，120K 高内存。有 4 个常驻内存的程序被推入高内存运行，DOS 进入扩展内存。通过 EMM386 分配 256K 扩展内存模拟扩充内存，供 PCTOOLS 等使用扩充内存的程序使用。还在基本内存运行了 DOS5.0 新提供的高速 CACHE——SMARTDRV，CACHE 使用扩展内存，可为软字库金山 DOS 系统提供高速缓存，大大提高了字库的读取速度。最可喜的是，通过 LH SPLIB.EXE 将这个需驻留内存的程序也推进了高内存 (XSDOS5.1 软字库系统运行后，SPLIB 将驻留内存，且系统退出时 SPLIB 占用的内存不能释放)。这样，在进入 XSDOS 中文系统后再退出，用 CHKDSK 检查内存，中文系统退出后能释放所有基本内存，与没进入前的内存数相同。至此，金山 DOS5.1 系统就能很好的在 MS-DOS5.0 下运行了。不但共享了 MS-DOS5.0 所有新功能特点，在中文下仍可得到 559KB 的自由空间。

## 在 2.13H 汉字系统下作饼状图

湖北省襄樊市湖北制药厂计算机室 徐文兵

众所周知, 2.13 序列汉字系统 H 以前的版本提供了画点, 线和填色功能, 可以作直方图, 但没有画圆弧命令, 从而无法直接作出饼状统计图。我们通过摸索, 在 2.13H 下运行 FOXBASE+ 应用软件, 成功地画出了理想的饼状统计图。

具体作法如下:

在过程文件中设置两个自定义函数 SIN (X) 和 COS (X), 采用台劳展开式计算角度的正弦和余弦, 供其它过程调用, 此处不作具体说明, 下面主要介绍画扇形子程序 PROC HSHX 及饼状统计图作法。所有角度均以度数为单位。

我们用一小角度的弦代替圆弧。当然, 角度越小越接近圆弧, 但花费的时间也越多。经试验取 0.5 度为间隔已足够美观, 速度也够快。因此, 在主文件中先计算好常数  $SN = \sin(0.5)$ ,  $CN = \cos(0.5)$ , 整理好一组要作图的数据。全部数据构成一个整饼状图, 每个数据对应一个扇形。在主程序中依次算出每个扇形的内部一点坐标 (XJ, YJ), 起始角和终止角, 然后调用过程 HSHX。

PROC HSHX

PARAMS YS, YXX, YXY, BJ, QJ, ZJ

• 调用参数: 颜色, 圆心 X, Y 坐标 (以像素为单位), 半径, 起始角, 终止角

SYXX=STR (YXX, 3) &&2.13 下作图参数变量必须转换成字符串; 然后用宏代换代入画图语句中

SYXY=STR (YXY, 3)

SBJ=STR (BJ, 3)

SQJ=360-QJ &&为了逆时针画出一个饼状图各扇形, 将起始角转换成; 其对圆周的补值

SA=YXX+BJ \* COS(0.017453 \* SQJ) &&计算圆弧起始点的坐标

SB=YXY+BJ \* SIN (0.017453 \* SQJ)

SD1=STR (SA, 3)

SD2=STR (SB, 3)

@1, 1 SAY CHR (14) + 'C7D&SD1., &SD2.L&SYXX., &SYXY.)' &&画圆弧起始半径

DO WHILE SQJ-1 > = 360-ZJ &&按 0.5 度间隔依次画出全部圆弧

SA=YXX+ (SA-YXX) \* CN+ (SB-YXY) \* SN

&&采用叠代方式计算后续弧端坐标, 避免每次算正余弦, 大大提高画; 弧速度

SB=YXY-(SA-YXX) \* SN+(SB-YXY) \* CN

SD3=STR (SA, 3)

SD4=STR (SB, 3)

@1, 1 SAY CHR (14) + 'C7D&SD1., &SD2., L&SD3., &SD4.)' SD1=SD3

SD2=SD4

SQJ=SQJ-0.5

ENDD

IF ZJ < 360 &&画圆弧终止半径

@1, 1 SAY CHR (14) + 'C7D&SD3., &SD4., L&SYXX., &SYXY.)'

ENDI

IF ZJ-QJ > 4 &&扇形角 4 度以上填色, 否则由于计算误差, 指定的填; 色点容易超出扇形外

SMM=STR (INT (XJ), 3)

SMM1=STR (INT (YJ), 3)

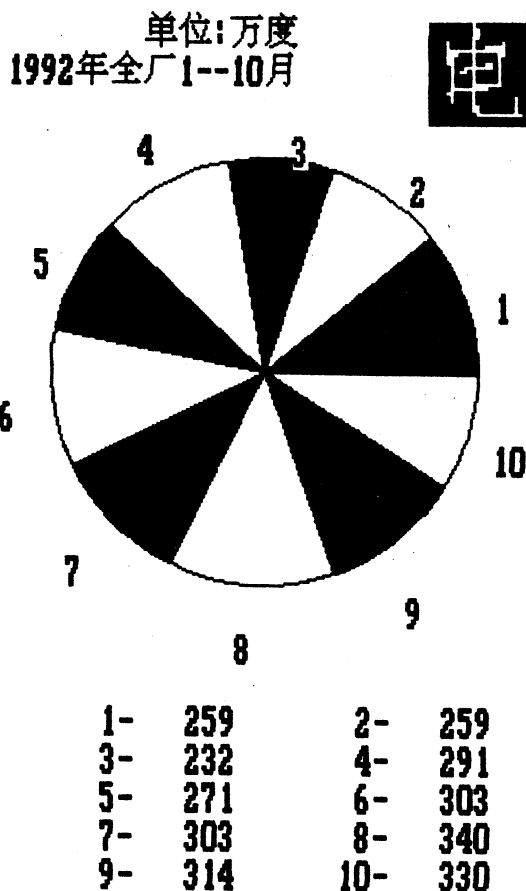
SYS=STR (YS, 2)

@1, 1 SAY CHR (14) + 'C4D&SMM., &SMM1.P&SYS., 7)'

ENDI

RETU

例如根据本厂 92 年 1-10 月用电量作统计图如下 (由于打印不便输出颜色, 故只有黑白两色扇形):



# 条件组合技术及其运用

华南师大计算机中心 殷秀霞

在程序设计中经常遇到条件的组合运用，例如，在已知基本条件及其组合的情况下，对数据库文件中符合条件的记录进行查询、修改、删除、统计、过滤、排序和打印等。如何才能有效地根据用户需求确定基本条件并实施其组合，是程序设计中值得探讨的一种技巧。本文结合应用软件开发实践，归纳出三种常见的条件组合模式和它们的模式结构，并通过例子说明它们的特点及使用场合。

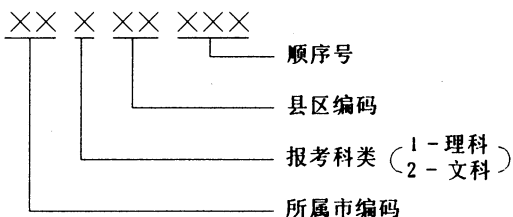
## 一、固定组合

固定组合是一种由若干个单一条件构成逻辑表达式后，固定到程序中加以运用的模式。

例如，已知考生登记表库文件 KSDJB.DBF 的结构如下：

| 字段英文名 | 字段中文名 | 类型 | 长度 | 小数位数 |
|-------|-------|----|----|------|
| KH    | 考号    | C  | 8  | —    |
| XM    | 姓名    | C  | 8  | —    |
| XB    | 性别    | C  | 2  | —    |
| XX    | 所在学校  | C  | 20 | —    |
| YW    | 语文成绩  | N  | 6  | 2    |
| SX    | 数学成绩  | N  | 6  | 2    |
| YY    | 英语成绩  | N  | 6  | 2    |
| ZF    | 总分    | N  | 6  | 2    |
| BZ    | 录取标志  | C  | 1  | —    |

其中考号的代码结构是



录取标志为“\*”时表示录取，不录取为空。

现在要分报考科类统计录取人数，方法是先确定有关的两个基本条件：

substr (KH, 3, 1) = “1”或“2”

BZ = “\*”

然后构成它们之间的“与”组合，并将组合的逻辑表达式固定到程序中（注意程序中划线的部分）

```
clear
use KSDJB
i=1
do while i<=2
is=str (i, 1)
go top
count to n&is for substr (KH, 3, 1) =is;
.and BZ = "*"
if i=1
@8+2*i, 30 say "理科录取人数:" +str (n&is,
4)
else
@8+2*i, 30 say "文科录取人数:" +str (n&is,
4)
endif
i=i+1
enddo
use
return
```

这种固定组合的特点是

- 1、组成组合条件的各个基本条件是确定不变的；
- 2、条件组合结构比较简单，通常是由基本条件的“与”、“或”、“非”运算连接而成；
- 3、这种组合之所以是固定的，是因为它被直接固定到程序之中，运用时如要改变其中的基本条件或它们之间的组合结构，就必须对程序进行修改。

因此，固定组合模式一般只适用于对库文件记录进行某种固定操作的场合。

## 二、选择组合

选择组合是一种在规定基本条件范围内由用户自行选择条件，然后按预先确定的组合结构在程序中予以实现的模式。

例如，已知物资调拨流水帐库文件 LSZ.DBF 的结构如下：

| 字段英文名 | 字段中文名  | 类型 | 长度 | 小数位数 |
|-------|--------|----|----|------|
| DH    | 调拨单号   | C  | 8  | —    |
| RQ    | 调拨日期   | D  | 8  | —    |
| BM    | 部门代码   | C  | 4  | —    |
| XM    | 支出项目编码 | C  | 3  | —    |
| WZ    | 物资名称   | C  | 20 | —    |
| XH    | 型号规格   | C  | 16 | —    |
| DW    | 单位     | C  | 4  | —    |

| SL  | 数量   | N | 10 | 2 |
|-----|------|---|----|---|
| DJ1 | 单价   | N | 10 | 2 |
| JE1 | 金额   | N | 10 | 2 |
| DJ2 | 调拨单价 | N | 10 | 2 |
| JE2 | 调拨金额 | N | 10 | 2 |

现在要根据用户的需求,对调拨单号、调拨日期、部门、支出项目和物资名称等条件进行任意选择的情况下,逐一查询符合条件的记录资料,其中调拨单号和调拨日期可以由用户指定某个范围,如果用户不选择其中的某个条件则作置空操作(回车即可)。

先确定有关的基本条件,设 d11、d12 为用户选择的调拨单号的上下界,初值为 SPACE (8); d21、d22 为用户选择的调拨日期的上、下界初值为 SPACE (8); d3 为用户选择的部门代码,初值为 SPACE (4); d4 为用户选择的支出项目编码,初值为 SPACE (3); d5 为用户选择的物资名称,初值为 SPACE (20)。基本条件是:

```
val (DH) >= val (d11)
val (DH) <= val (d12)
RQ >= ctod (d21)
RQ <= ctod (d22)
BM = d3
XM = d4
```

WZ = d5

实现条件选择组合的程序如下:

```
clear
use LSZ
@ 0,0 say "请输入查询条件"
d11 = space(8)
d12 = space(8)
d21 = space(8)
d22 = space(8)
d3 = space(4)
d4 = space(5)
d5 = space(20)
@ 2,0 say "调拨单号 至"
@ 3,0 say "调拨日期 至"
@ 4,0 say "部门代码"
@ 5,0 say "支出项目"
@ 6,0 say "物资名称"
@ 2,8 say get d11
@ 2,20 say get d12
@ 3,8 say get d21
@ 3,20 say get d22
@ 4,8 get d3
@ 5,8 get d4
@ 6,8 get d5
read
if d11 <> space(8)
tj = "val(DH) >= val(d11)"
else
tj = " "
endif
if d12 <> space(8)
```

```
if tj = " "
tj = "val(DH) <= val(d12)"
else
tj = tj + ".and.val(DH) <= val(d12)"
endif
endif
if d21 <> space(8)
if tj = " "
tj = "RQ >= ctod(d21)"
else
tj = tj + ".and.RQ >= ctod(d21)"
endif
endif
if d22 <> space(8)
if tj = " "
tj = "RQ <= ctod(d22)"
else
tj = tj + ".and.RQ <= ctod(d22)"
endif
endif
if d3 <> space(4)
if tj = " "
tj = "BM = d3"
else
tj = tj + ".and.BM = d3"
endif
endif
if d4 <> space(3)
if tj = " "
tj = "XM = d4"
else
tj = tj + ".and.XM = d4"
endif
endif
if d5 <> space(20)
if tj = " "
tj = "WZ = d5"
else
tj = tj + ".and.WZ = d5"
endif
endif
if tj <> " "
locate for &tj
if eof()
@ 8,0 say "没有符合条件的记录"
@ 9,0 say "请按任意键继续"
set cons off
wait
set cons on
else
do while .t.
@ 10,0 clear
if eof()
@ 20,28 say "查询结束, 请按任意键继续"
set cons off
wait
set cons on
exit
endif
@ 13,28 say "调拨单号:" + DH
@ 14,28 say "调拨日期:" + RQ
```

```
@ 15,28 say "部门代码:" + BM
@ 16,28 say "支出项目:" + XM
@ 17,28 say "物资名称:" + WZ
@ 18,28 say "型号规格:" + XH
@ 13,62 say "单位:" + DW
@ 14,62 say "数量:" + str(SL,10,2)
@ 15,62 say "单价:" + str(DJ1,10,2)
@ 16,62 say "金额:" + str(JE1,10,2)
@ 17,58 say "调拨单价:" + str(DJ2,10,2)
@ 18,58 say "调拨金额:" + str(JE2,10,2)
@ 20,28 say "还要继续查询符合条件的记录码(Y/N)?"
do while .t.
set cons off
wait to yn
set cons on
if upper(yn) = "Y".or.upper(yn) = "N"
exit
endif
enddo
if upper(yn) = "N"
exit
endif
continue
enddo
endif
use
return
```

选择组合的特点是:

1. 组成组合条件的基本条件是确定的, 有限的;
2. 在规定的基本条件范围内, 各基本条件的取舍及其取值是可选择的;
3. 被选择和确定的基本条件, 按照预先设计的组合结构 (在上面的例中, 条件的组合结构是所有基本条件的“与”连接) 进行组合。

因此, 选择组合模式比较适合于对库文件记录进行多条件, 多选择综合操作的场合, 这种模式在诸如人事管理、工资管理、学籍管理、文书资料管理等用得最多。

### 三、自由组合

自由组合是一种在程序运行过程中由用户自由构造

出条件组合的逻辑表达式, 并在程序中通过代换来实现的一种模式。例如, 对上述物资调拨流水帐库文件 LSZ.DBF 的查询中, 可以设计如下一段程序来实现按条件自由组合来进行有关的操作。

```
clear
@ 0,0 say "请键入要查询的条件组合式"
on error do cwcl
tj = space(80)
@ 2,0 get tj
read
on error
use LZS
locate for &tj
.
```

其中过程 CWCL 是一个错误处理程序:

```
@ 4,0 say "条件组合式语法错误, 按任意键重新键入"
set cons off
wait
set cons on
retry
```

自由组合的特点是:

1. 组成组合条件的基本条件可以是库文件中任何一个字段有关的条件, 固而基本条件是不确定的;
2. 基本条件的组合是“完全”自由的, 它们的取舍以及组合结构的逻辑表达式, 只要是符合语法规则的, 都予以承认;
3. 由于条件组合是自由的, 因而要求使用者掌握一定的逻辑代数知识, 懂得如何表达条件的组合方法和规则。

自由组合模式特别适合于组合条件不确定的情形, 例如可以用于综合查询和模糊查询等。

在实际应用中, 运用哪一种组合模式常常要根据需要来确定, 甚至要考虑几种模式结合起来使用, 固定组合的局限性比较大, 而自由组合又是相对的, 目前软件开发人员大多采用各种形态的选择模式, 例如单一条件的线性“与”组合和“或”组合等 (见图 1、图 2)

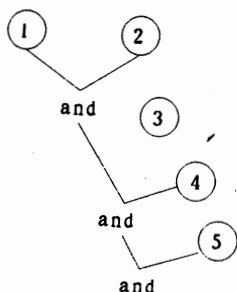


图1 单条件的线性“与”组合

①.and.②.and.④.and.⑤

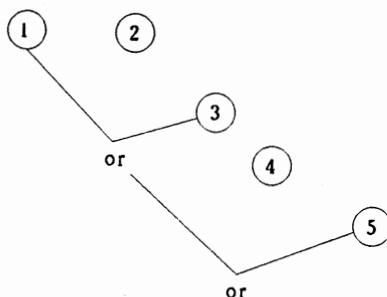


图2 单一条件的线性“或”组合

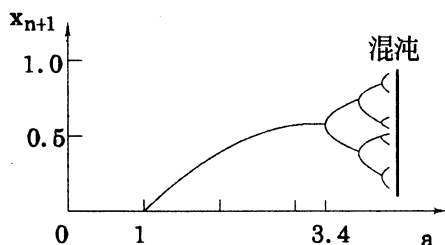
①.or.③.or.⑤

## 用“混沌方法” 产生伪随机数

清华大学自动化系 芮 勇

混沌科学是一门新兴的边缘科学,研究的是非线性动态系统的运动特性。正因为非线性,使系统具有了许多“美妙”的特性,其中著名的逻辑斯蒂方程  $x_{n+1} = a \cdot x_n \cdot (1 - x_n)$ ,  $a$  为参数,就具有如下特性:

当  $a < 3.4$  时,方程的迭代值将趋于某一终值,当  $a > 3.4$  时,迭代值将不趋于任何一个终值,而是在两个终值上“跳来跳去”,当  $a$  继续增大时,会出现“四分叉”、“八分叉”直至进入“混沌区”。混沌区内的迭代值完全是一个随机数,我们无法预知  $x_{n+1}$  会趋于何值,见下图。



本文的随机数产生方法就是基于以上原理,即利用逻辑斯蒂方程的“混沌区”来产生随机数,这是一种简便而实用的随机数产生法。

源程序如下,其中 `int chaos-random (int scale)` 即为“混沌”法随机函数,调用该函数将产生一个 0~scale 之间的随机整数,逻辑斯蒂方程中的参数  $a$  此处设定为 3.999999。

```
#include "stdio.h"
#include "stdlib.h"
main() /*主调函数*/
{
    int chaos-random(int scale);
    /*说明“混沌”随机函数*/
    int i;
    int random-value;
    printf("\n");
    for(i=1;i<50;i++)
    {
        random-value=chaos-random(20);
        printf("%d", random-value);
    }
    /*调用chaos-random()函数*/
    /*产生50个(0~20)的随机数*/
}
```

```
int chaos-random(int scale)
/*定义“混沌”随机函数*/
{
    static times=1000; /*迭代次数*/
    float parameter=3.999999;
    /*“混沌”区的参数*/
    float random-value=0.5;
    int return-value;
    int i;
    times++;
    for(i=1;i<times;i++)
        random-value=parameter*random-value*
        (1-random-value); /*“逻辑斯蒂”方程*/
    return-value=(int)(random-value*scale+0.5);
    return(return-value);
    /*返回(0~scale)范围内的随机整数*/
}
```

133

#####

## 在 MS-DOS5.0 下 运行 UCDOS 的简便方法

湖北省宜昌县人民银行 喻顺洲

在 MS-DOS5.0 下运行 UCDOS, 出现键盘死锁现象, 若把系统裁减的键盘置为 ON, 系统虽然能正常运行, 但无法从键盘输入汉字和进行屏幕切换, 这说明 UCDOS 和 MS-DOS5.0 的键盘程序不兼容。本人在 MS-DOS5.0 下运行 UCDOS 时, 找到了一个不需要附加任何服务程序, 只是利用 MS-DOS5.0 系统固有的功能特点, 达到了两者完美结合的途径。即在引导盘根目录下的 CONFIG.SYS 文件中增加一行 SWITCHES= /K, 但如果在 CONFIG.SYS 文件中已经使用了 DEVICE=ANSI.SYS 命令, 就在其后增加一个 /K 开关项, 即 DEVICE=ANSI.SYS /K, 重新引导后就可以在 MS-DOS5.0 下正常使用 UCDOS, 丝毫不损坏 UCDOS 系统的特点。需要注意的是按此方法配置了 MSDOS 系统后, 在配置 UCDOS 时, 如果系统配有 EGA (CEGA) 卡, UCDOS 的显示设置 ANSI 可以是 YES 或 NO, 但若是 VGA (CVGA) 卡, UCDOS 的显示设置 ANSI 只能是 NO 而不能是 YES, 否则不能正常运行。此方法已在 Compaq、AST、东海等系列机上通过。另外在配有长城 CVGA / 24 卡的机上可以直接在 MS-DOS5.0 下运行 UCDOS。

134

# 可编程控制器中的位处理器

华南师大微电子所 徐巨善

可编程控制器是专门为工业环境下应用而设计的微型计算机系统, 目前已经在国内广泛使用, 但对其硬件结构很少有详细资料, 尤其是其中的位处理器更是厂家的专用部件, 成为专利的核心。:《一个大型可编程控制器的电路分析》<sup>[1]</sup>介绍了 K3NPC 的详细逻辑框图。本文分析其位处理器的结构原理, 根据具体的逻辑电路, 说明用户指令的读取和执行过程。

## 一、位处理器的结构原理

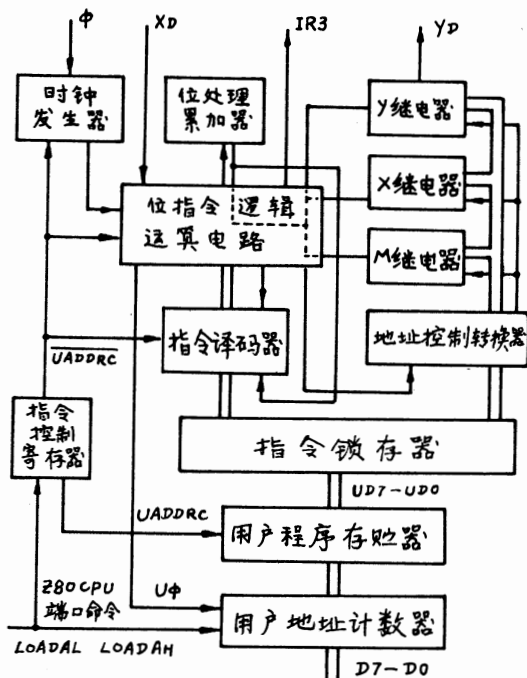


图1 指令译码和位处理器逻辑框图

位处理器是执行用户程序进行逻辑运算的核心部件, 其逻辑结构如图1所示。其中, 用户程序存储器、用户地址计数器、指令锁存器是读取用户程序指令的; 指令译码器和地址控制转换器是提供指令的逻辑控制信号和存储数据的导址电路; 位指令逻辑运算电路、位处理累加器和 X、Y、M 继电器是位指令逻辑运算电路; 时钟发生器、指令控制寄存器是用户指令执行时的启动和定时控制电路。

当用户指令执行时, Z80CPU 发送端口命令

LOADAL、LOADAH 将用户指令地址装入用户地址计数器, 并启动指令控制寄存器, 使时钟发生器、位指令逻辑运算电路、指令译码器和用户程序存储器进入有效工作状态。在时钟发生器定时控制下, 用户程序存储器将指令操作码和操作数 (UD7-UD0) 送入指令锁存器。指令操作码由指令译码器译出控制信号, 送位指令逻辑运算电路。指令锁存器还送出地址, 经地址控制转换器, 对 X、Y、M 继电器进行寻址, 译码输出控制信号也为继电器提供片选和读写信号等。在位指令逻辑运算电路控制下, 实现位指令的逻辑操作, 将结果存入位处理累加器或 M、Y 继电器。如果用户指令为非位指令, 则位指令逻辑运算电路输出中断请求信号 IR3, 由 Z80CPU 解释执行。X 继电器的数据由 Z80CPU 从输入电路读入, Y 继电器的数据 YD 由 Z80CPU 输送到输出电路。下面用详细逻辑图对各部分的结构原理进行具体分析。

## 1、时钟发生器和指令控制寄存器

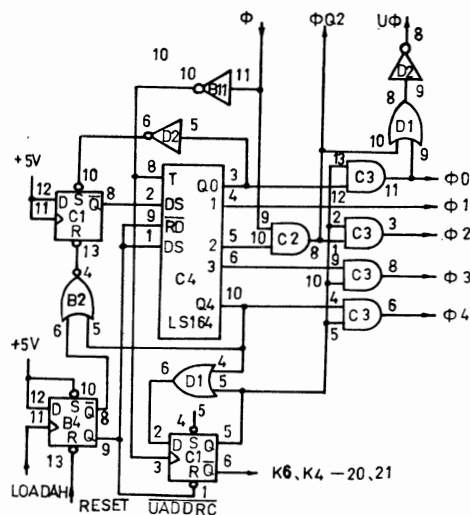


图2 时钟发生器和指令控制寄存器

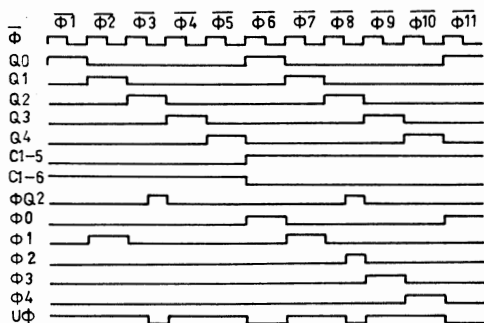
图2为时钟发生器和指令控制寄存器电路。时钟发生器主要由四位移位寄存器 C4 (74LS164)、D 触发



器 C1 及一些门电路组成, D 触发器 B4 (74LS74) 为指令控制寄存器电路。当 Z80CPU 端口命令 LOADAH 使 B4 成为 1 状态时, B4-9 UADDR C 1 电平使 C1-1、C4-9 直接复位无效, 并使 C4-1 输入为 1 电平, B4-8 UADDR C 使 C1-13 直接复位无效。这样, 系统时钟  $\Phi$  使时钟发生器产生五相时钟, 其时序如图 3 所示。由于系统复位时 C4、C1 均为 0 状态, C1-8 输出 1 电平到 C4-2 作为数据输入,  $\Phi$  作为 C4 的移位触发脉冲,  $\Phi 1$  的正跳变使 Q0 输出 1 电平, C1-8 输出 0 电平。此后,  $\Phi 2-\Phi 5$  移位输出 1 电平。Q4 的 1 电平, 使 C1-13 复位有效, C1-8 输出 1 电平。另外, Q2 和  $\Phi$  经 C2, 输出  $\Phi Q 2$ 。由于 C1-5 为 0 电平, 故这时  $\Phi 0-\Phi 4$  均为 0 电平。当 Q4 为 1 电平后,  $\Phi 6$  使 C1-5 输出 1 电平, 以后产生的 Q0-Q4 的正脉冲, 通过 C3 输出五相时钟  $\Phi 0-\Phi 4$ 。Q0 和  $\Phi Q 2$  经 D1、D2 输出用户地址计数器触发脉冲 U $\Phi$ 。由于指令控制寄存器 B4 为 1 状态, 所以  $\Phi 0-\Phi 4$  及  $\Phi Q 2$ 、U $\Phi$  能够不断循环输出。而当 B4 复位后, 输出脉冲的时序, 又如图 3 重头开始。五相时钟  $\Phi 0-\Phi 4$  和  $\Phi Q 2$ 、U $\Phi$  的功能如表 1 所示。

表 1. 互相时钟  $\Phi 0-\Phi 4$  和  $\Phi Q 2$ 、U $\Phi$  功能表

| 时钟名称       | 功 能                                           |
|------------|-----------------------------------------------|
| $\Phi 0$   | 使指令锁存器输出新指令                                   |
| U $\Phi$   | 用户地址计数器指向下一指令操作码                              |
| $\Phi 1$   | 非位指令时, 使指令控制寄存器复位                             |
| $\Phi 2$   | 位数据进入数据暂存器; 非位指令时输出中断请求信号 IR3                 |
| $\Phi Q 2$ | 使操作码存入指令锁存器                                   |
| U $\Phi$   | 用户地址计数器指向下一指令操作数                              |
| Q2         | OUT、SET、RST 指令时对 $\chi_2$ M 继电器写有效            |
| $\Phi 3$   | LD、LDI 指令时累加器右移; ANB、ORB 指令时累加器左移; 其他指令时累加器保持 |
| $\Phi 4$   | 位指令时数据接收到累加器; 非位指令时数据保持                       |

图 3 五相时钟  $\Phi 0-\Phi 4$  和  $\Phi 2$ 、U $\Phi$  时序图

## 2. 用户指令的读取和译码电路

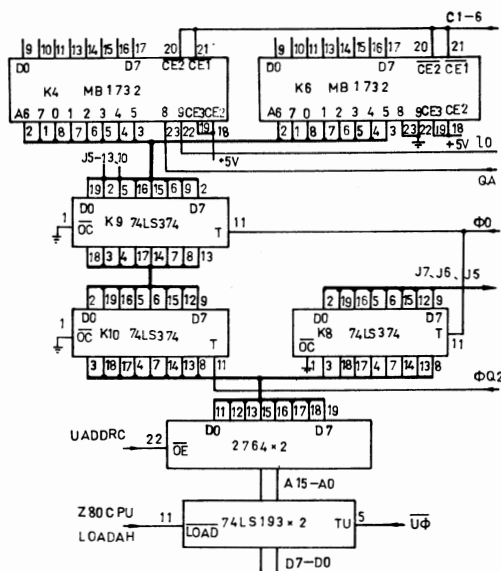


图 4 用户指令的读取和译码电路

图 4 为用户指令的读取和译码电路, 除用户地址计数器 and 用户程序存储器之外, 主要由锁存器 K8-K10 (74LS374)、指令译码器 K4、K6 (MB1732) 等组成。MB1732 实际是 1KX8 位 PROM, 存放用户指令的输出控制信号, 是可编程控制器的专用器件。使用两块 MB1732, 可译出 16 位控制信号, 实际利用 14 位。用户程序执行时, Z80CPU 使用端口命令将两字节地址写入用户地址计数器, 并启动指令控制寄存器, 使用用户程序存储器输出用户指令操作码。当  $\Phi Q 2$  正跳变时, 用户指令操作码存入锁存器 K10。随后, U $\Phi$  正跳变使用户地址计数器地址加 1, 指向用户指令操作数。当时钟  $\Phi 0$  正跳变时, 将操作数存入 K8, 同时将操作码存入 K9, 并送到指令译码器 K4、K6 的地址输入端。由于 K4-23 为位处理累加器的输出数据 QA, 而 K4-22 在可编程控制器运行时为 0 电平, 又由于  $\Phi 0$  为 1 电平时 C1-6 为 0 电平, 使译码器 K4、K6 输出有效。这样, K4、K6 输出 14 位控制信号, 使位处理器进入逻辑运算。同时, K8 和 K9-2、5 的输出作为 X、Y、M 继电器的地址, 送位指令寻址电路。接着, U $\Phi$  第二个正跳变使用户地址计数器加 1, 指向第二条用户指令的操作码。

## 3. X、Y、M 继电器及其寻址电路

X、Y、M 继电器及其寻址电路如图 5 所示, 主要由地址转换电路 J7、J6、J5 和控制转换电路 D5 (74LS157)、X 继电器 F6、Y 继电器 F4 (HM61116)、

M 继电器 F8 (TC5517) 及一些门电路组成。X、Y、M 继电器都是  $2K \times 8$  RAM, 作为用户数据存贮时, 使用 D0 一位。X 继电器的数据是用户程序执行前由输入电路从输入端读入的现场信号, 而 Y 继电器的数据是位处理器运算的结果, 在用户程序执行完一个周期后, 由输出电路发送到现场, 控制器件动作。M 继电器是位处理器逻辑运算时暂时存放数据的辅助继电器, 要使生产线连续工作, 就要保证这些数据在偶然停电时也不丢失。

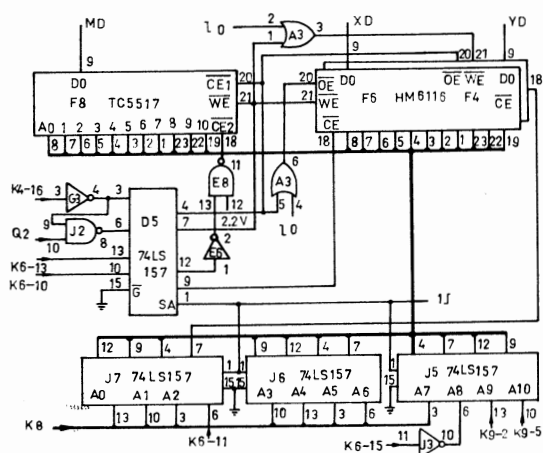


图 5 X、Y、M 继电器及其寻址电路

位指令执行时, 锁存器 K8 的指令操作数、累加器输出 QA 和 K9-2、5 指令操作码的输出, 为 X、Y、M 继电器提供地址。指令译码器 K6-10、11、13 和 K4-16 及定时信号 Q2 提供继电器的控制信号。当位处理器工作时, 地址控制转换电路 74LS157 的控制端为 1 电平。而 Z80CPU 直接寻址时为 0 电平, 其地址和控制信号由系统总线提供, 图中没有画出。

#### 4. 位指令逻辑运算电路

位指令逻辑运算电路是位处理器的核心部分, 其结构如图 6 所示, 主要由累加器 E1 (74LS198) 及其控制电路, 数据寄存器 K1 和逻辑运算控制电路, 非位指令中断信号产生电路 B1、B2 和指令控制寄存器复位电路等组成。其中 E1 为 8 位移位寄存器, SISO 的控制功能为: 00 保持, 01 右移, 10 左移, 11 接收。指令译码器根据用户指令译出控制信号, 使位数据从 X、M 继电器读出, 再与累加器的数据进行逻辑运算, 并将结果存入累加器 E1 或 Y、M 继电器。指令译码器输出的 14 位控制信号的功能如表 2 所示。

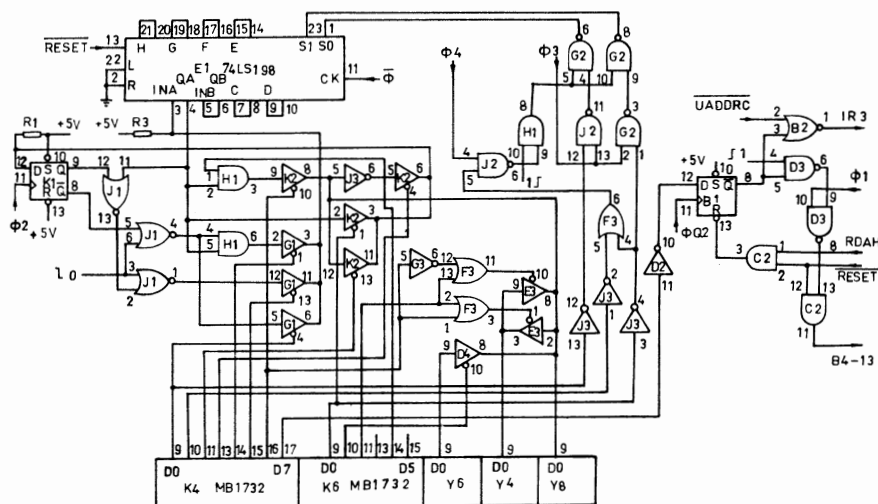


图 6 位指令逻辑运算电路

表2 指令译码输出控制信号功能表

| 控制信号  | 控制器件  | 输出 0 电平时的功能                                                           |
|-------|-------|-----------------------------------------------------------------------|
| K4-9  | G1-4  | LD、LDI 指令有效                                                           |
| K4-10 | J3-1  | LD、LDI、OR、ORI、AND、ANI 指令时, 控制累加器 SISO, 实现保持、右移、左移或接收                  |
| K4-11 | K2-13 | LD、OR、AND 逻辑指令有效                                                      |
| K4-13 | K2-4  | LDI、ORI、ANI 逻辑非指令有效                                                   |
| K4-14 | G1-1  | AND、ANI 与指令有效                                                         |
| K4-15 | G1-13 | OR、ORI 或指令有效                                                          |
| K4-16 | K2-10 | OUT、SET、RST 位指令有效(累加器 QA=1)                                           |
| K4-17 | IR3   | MC、MCR、SFT、PLS、SET C(T)、RST C(T)、OUT F(C)等非位指令时, 产生中断请求 IR3(累加器 QA=1) |
| K6-9  | K2-1  | ANB、ORB 指令有效                                                          |
| K6-10 | F6    | X 继电器有效                                                               |
| K6-11 | F4    | Y 继电器有效                                                               |
| K6-13 | F8    | M 继电器有效                                                               |
| K6-14 | H1-1  | RSTM(Y)指令有效(累加器 QA=1)                                                 |
| K6-15 | J5-6  | X、Y、M 继电器地址 A8                                                        |

由表 2 可见, 用户指令中的非位指令 K4-17 为 0 电平, 产生中断请求 IR3, 这时逻辑运算的结果必须为 1, 即累加器 QA=1, 这样才使 Z80CPU 执行中断服务子程序, 完成非位指令的解释执行。而其余控制端用来控制位指令的逻辑运算, 完成位指令的执行, 其中位指令 OUT、SET、RST 也要求逻辑运算结果累加器 AQ=1。用户指令译码器输出的控制信号, 应由指令操作码与对应的译码器输出控制信号对照表列出, 不过对于位指令, 由指令译码输出控制信号功能表大致可以判断。例如 LD X2 指令, 应是 LD、X 有效, 即 K4-9、10、11 及 K6-10 输出为有效 0 电平; ANI M5 应是 ANI、M 有效, 即 K4-10、13、14 及 K6-13 输出 0 电平; OUT Y105 应是 OUTY 有效, 即 K4-16、K6-11 输出为 0 电平; ORB 应是 OR、ORB 有效, 即 K4-15 及 K6-9 输出为 0 电平; RST M0 应是 RST、M 有效, 即 K4-16、K6-13、14 输出为 0 电平等。

## 二、位指令的执行过程

分析下述程序段位指令的执行过程:

```
LD X2      工程步进开关“自动”位置
OR M26     试灯按钮信号
OUT Y54    “自动”指示灯
```

工程步进开关“自动”位置接点接通时 X2 为 1, 断开时为 0。LD X2 指令是位处理器把接点状态对应的

X 继电器地址为 02H (16 进制) 的数据取出, 并存入位处理累加器 QA。M26 是辅助继电器 M 地址为 26 (十进制) 的状态, 对应于控制面板“试灯”按钮的状态, 按下时 M26 为 1, 否则为 0。OR M26 指令是位处理器将 M26 的状态取出与 X2 的状态进行逻辑“或”, 其结果存入累加器。Y 继电器地址 54H 为“自动”指示灯的状态, Y54 为 1 时该指示灯亮, 否则不亮。OUT Y54 是将 X2 与 M26 相“或”的结果送继电器 Y54, 最后决定“自动”指示灯是否亮。在生产线正常运行时, 工程步进开关“自动”位置接通, X2 为 1, 试灯按钮不会按下, M26 为 0, 工程步进“自动”指示灯应亮, 即 Y54 应为 1。

由图 2-图 4 可见, 当用户指令 LD X2 执行时, Z80CPU 使用户程序存储器输出指令操作码 10H (这要从用户指令机器码清单中查到), 时钟发生器输出时钟。

当时钟  $\Phi 1$  的正跳变后, C4 的 Q0 输出 1 电平, 因 C1-5 为 0 电平,  $\Phi 0$  为 0 电平不变。

当  $\Phi 2$  正跳变后, Q1 即  $\Phi 1$  输出 1 电平。由于系统复位时 B1-8、6 都为 1 电平, D3-8 输出 1 电平不变 (参看图 6)。

当  $\Phi 3$  负跳变, 即  $\Phi Q2$  正跳变时, 将操作码 10H 存入锁存器 K10,  $\Phi 2$  为 0 电平不变。

当  $\Phi 4$  即  $U\Phi$  正跳变时, 用户地址计数器加 1, 用户程序存储器输出指令操作数 02H,  $\Phi 4$  为 0 电平不变。

当  $\Phi 5$  正跳变后输出 Q4 为 1 电平,  $\Phi 4$  为 0 电平不变。

当  $\Phi 6$  正跳变后 C1-5 为 1 电平,  $\Phi 0$  输出正跳变, 指令操作码 10H 存入 K9, 操作数存入 K8。操作码送指令译码器 K4、K6, 操作数送到图 5 X、Y、M 继电器寻址电路的 J7、J6、J5。由于这时时钟发生器电路中的 C1-6 为 0 电平, 故指令译码器输出有效控制信号, K4-9、10、11 和 K6-10 为 0 电平, 其余输出为 1 电平。由于操作数 02H 送到 J5-J7 的 A7-A0, 又译码器 K6-15 输出反相后 0 电平送到 J5-6, 操作码 K9-2、5 为 00B (二进制数) 送到 J5-13、10。这样, X 继电器的地址为 002H。因 K6-10 为 0 电平, 经 D5-10、9 到 F6-18, 即 X 继电器的  $\overline{CE}$  有效。又因 K4-15 为 1 电平, 反相后 0 电平经 D5-3、4, A3-5、6 到 F6-20, 使  $\overline{OE}$  有效。这样, X 继电器处于读有效状态, F6-9 输出有效数据 D0。从图 6 可见, 由于 K6-10 为 0 电平, 三态门 D4-10 有效, 所以 X 继电器输出 D0, 即 X2 的 1 电平, 经 D4-9、8, K2-12、11 (因 K4-11 为 0 电平) 到达数据暂存器 K1-12。另

外,在这同时,  $U\Phi$  使用户地址计数器加 1, 指向第二条用户指令的操作码。

$\Phi 7$  正跳变后,  $Q1$  即  $\Phi 1$  输出 1 电平, 对电路无影响。

当  $\Phi 8$  即  $\Phi 2$  正跳变送到  $K1-11$  时, 数据  $D0$  存入暂存器  $K1$ , 使  $K1-9$  输出 1 电平,  $K1-8$  的 0 电平, 经  $J1$  反相, 1 电平经  $G1-5$ 、6 到达累加器输入端  $E1-3$ 。当  $\Phi 8$  负跳变, 即  $\Phi Q2$  正跳变时, 第二条指令的操作码存入指令锁存器  $K10$ 。

当  $\Phi 9$ , 即  $U\Phi$  的正跳变时, 又使用户地址计数器加 1, 指向第二条指令的操作数。又  $\Phi 3$  为 1 电平时, 由于  $K4-9$  为 0 电平, 经  $J3-13$ 、12 反相, 1 电平送到  $J2-12$ 。又  $K6-9$  为 1 电平, 反相后 0 电平送到  $G2-1$ 、 $J2-11$  为 0 电平,  $G2-3$  为 1 电平, 而这时  $\Phi 4$  为 0 电平,  $J2-6$  为 1 电平,  $H1-8$  为 1 电平, 所以  $G2-8$  为 0 电平,  $G2-6$  为 1 电平, 亦即累加器  $E1$  的控制端  $S1S0$  为 01 状态,  $\Phi$  的正跳变使  $E1$  进行右移, 原来  $AQ$  的数据存入  $QB$ , 输入端  $E1-3$  的数据存入  $QA$ 。

当  $\Phi 10$  即  $\Phi 4$  为 1 电平时, 由于  $K4-10$  为 0 电平, 经  $J3-1$ 、2 反相, 输出 1 电平, 经  $F3-5$ 、6 到  $J2-5$ 、 $J2-6$  输出 0 电平, 送到  $G2-10$ 、5, 使  $G2-8$ 、6 输出 1 电平, 即  $E1$  的  $S1S0$  为 11 状态。这时  $E1$  接收寄存器输入端的数据, 即  $E1-3$   $INA$  存入  $QA$ ,  $INB$  存入  $QB$  等等。到这里为止, 第一条用户指令  $LD X2$  将  $X2$  的数据取入累加器  $QA$  执行完毕。

当  $\Phi 11$  即第二个  $\Phi 0$  正跳变时, 第二条指令  $OR M26$  的操作码和操作数存入锁存器  $K9$  和  $K8$ , 并送指令译码器及继电器寻址电路。随后  $U\Phi$  的正跳变又使用户地址计数器加 1, 指向第三条指令的操作码。由于  $OR M26$  的操作码为  $60H$ , 操作数为  $1AH$ , 指令译码器输出控制信号  $K4-10$ 、11、15 及  $K6-13$  为 0 电平, 其余为 1 电平。辅助继电器  $F8-9$  的数据  $D0$ , 即  $M26$  的 0 电平, 经  $K2-12$ 、11 (因  $K4-11$  为 0 电平) 到  $K1-12$ ,  $\Phi 2$  正跳变时将数据存入暂存器, 使  $K1-9$  输出数据  $D0$ , 与累加器保存的数据  $QA$  的输出, 即  $X2$  的 1 电平进行“或”运算。在  $J1-12$ 、11、13“或非”后, 再经  $J1-2$ 、1 反相,  $J1-1$  输出“或”运算结果, 经  $G1-12$ 、11 (因  $K4-15$  为 0 电平) 到累加器输入端  $E1-3$ 。随后  $\Phi Q2$  正跳变将第三条指令操作码存入  $K10$ ,  $U\Phi$  正跳变使用户地址计数器加 1, 指向第三条指令操作数。又当  $\Phi 3$  为 1 电平时, 将累加器  $QA$  的 1 进行右移, 输入端“或”运算的结果移入  $QA$ , 原  $QA$  的值移入  $QB$ 。 $\Phi 4$  为 1 电平时, 又将数据接收到移位寄存器, 即“或”的结果由  $QA$  接收。至于对  $M$  继电器的

寻址过程与  $X$  继电器类似。这时,  $K6-13$  为 0 电平, 经  $D5-13$ 、12,  $E6-1$ 、2 及  $E8-13$ 、11 输出 0 电平到  $F8-18$  即  $M$  继电器  $CE2$  有效, 又  $K4-16$  为 1 电平, 经  $G3-3$ 、4 输出 0 电平, 经  $D5-3$ 、4 到  $F8-20$  使  $M$  继电器  $OE1$  读出有效。

当第三个  $\Phi 0$  正跳变时, 第三条指令  $OUT Y54$  的操作码和操作数存入锁存器  $K9$  和  $K8$ , 并送指令译码器及继电器寻址电路。随后  $U\Phi$  的正跳变又使用户地址计数器加 1, 指向第四条指令操作码。这时  $OUT Y54$  的操作码  $A0H$ 、操作数  $54H$ , 指令译码器输出控制信号  $K4-16$ 、 $K6-11$  为 0 电平, 其余为 1 电平。累加器  $E1-4$   $QA$  输出数据经  $H1-2$ 、3 (因  $K6-14$  为 1 电平),  $K2-9$ 、8 (因  $K4-16$  为 0 电平), 经  $E3-2$ 、3 (因  $K4-16$ 、 $K6-11$  为 0 电平), 到达  $Y$  继电器  $F4-9$  的数据  $D0$  端。根据  $X$ 、 $M$  继电器类似的寻址方式,  $Y$  继电器的地址为  $054H$ 。这时  $K6-11$  为 0 电平, 经  $J7-6$ 、7 到  $Y$  继电器  $F4-18$ , 使  $CE$  有效。又  $K4-11$  为 0 电平, 经  $G3-3$ 、4 反相 1 电平到  $J2-9$ 。当  $Q2$  为 1 电平时,  $J2-8$  输出 0 电平, 经  $D5-6$ 、7,  $A3-1$ 、3 到  $F4-21$  使  $WE$  有效, 从累加器  $E1-4$   $QA$  送来的数据 1 写入  $Y$  继电器, 使  $Y54$  为 1。

到此为止, 上述三条用户指令程序段才算执行完毕, 这也是指令译码器和位处理器执行位操作指令的具体过程。

另外, 如果执行的是非位指令, 则指令译码器输出控制信号  $K4-17$  为 0 电平, 则  $\Phi Q2$  正跳变时将  $D2-11$ 、10 反相后的 1 电平使  $B1$  成为 1 状态,  $B1-8$  输出 0 电平。由于这时  $\overline{UADDR C}$  为 0 电平, 所以  $B2-1$  输出 1 电平, 即中断请求  $IR3$  有效。 $Z80CPU$  响应中断请求, 执行非位指令。同时,  $B1-8$  的 0 电平使  $D3-6$  为 1 电平, 当下一个  $\Phi 1$  时  $D3-8$  输出 0 电平, 经  $C2-12$ 、11 使指令控制寄存器  $B4-13$  直接复位有效, 使位处理器各部分停止执行用户指令。此后, 只有  $Z80CPU$  重新启动下一条用户指令时, 指令译码器和位处理器才能继续执行用户程序指令。

### 三、结束语

本文是对  $K3NCPU$  的实际逻辑电路、用户指令机器码和用户指令译码器机器码, 进行大量实验测试和分析的基础上完成的。对于有关工程技术人员深入理解位处理器的结构原理, 促进可编程控制器的设计和应用具有积极的意义。对刘相安付研究员的密切合作, 对有关老师及工厂技术人员的支持和帮助表示深切的谢意。

注: [1]一个大型可编程控制器的电路分析 《电脑》1993 年第 2 期

# 国营南京有线电厂

国营南京有线电厂是高技术、多门类、综合型的电子产品整机生产企业。具有 50 多年建厂历史，现有职工近 6000 人，工程技术人员 1400 多人。研制生产：光纤数字、小同轴、电力载波通信产品系列、有线测量仪表，紫金微型计算机系列、紫金外设汉字打印机系列、紫金监视器系列产品以及多层高密度印制板加工等。是我国载波通信设备、微型计算机和外设产品的生产基地。

工厂 85 年以来被南京市连续评为“文明工厂”86 年获国家电子部“质量管理”奖、87 年获“国家一级计量企业”称号，88 年获“国家二级企业”号。

工厂产品曾先后荣获“国家银牌奖”三枚，部优奖六项，以及国家级、省、市级其它各类表彰多次。

工厂将不断致力于信息传输系统、数据处理系统以及外部设备三个领域的新技术开发和应用。我们的宗旨始终是：“质量第一，用户至上、服务及时、固守信誉”。欢迎合作。

南京有线电厂是机械电子部骨干企业，这个厂具有 50 多年建厂历史的老企业。是国家的载波通信设备、计算机和打印机显示器的生产基地，也是多门类的综合制造厂家。

半个多世纪来，这个厂在自身的不断发展壮大中，推出了一个又一个优质可靠的产品，培养了一批又一批的优秀人才，作出了应有的贡献，促进新产品的问世。他们继续把力量集中在信息传输系统数据处理系统及其外部设备三个具有高潜力的领域。他们研制生产的通讯设备、微机、打印机、显示器、仪表等产品为国防、水电、铁道、金融、教育统计等领域提供了大量的装备。

在新的历史时期，他们将继续发挥通讯与计算机结合、主机与外设配套、硬件与软件同时发展之优势，以向用户提供高质量的产品，让“紫金”之花遍地开放，为“紫金”产品做满意的服务为己任。

这个厂是国家一级计量企业、部质量管理奖、国家二级企业、设备管理优秀企业显示器产品曾二次获得部优质奖等荣誉、称号。他们仍将不遗余力地向管理求效益。九二年这个厂在南京市大型企业中获得利税前五十名的行列之中。

“紫金”显示器九二年批量出口，九三年又与外商签订合同出口销往国外。

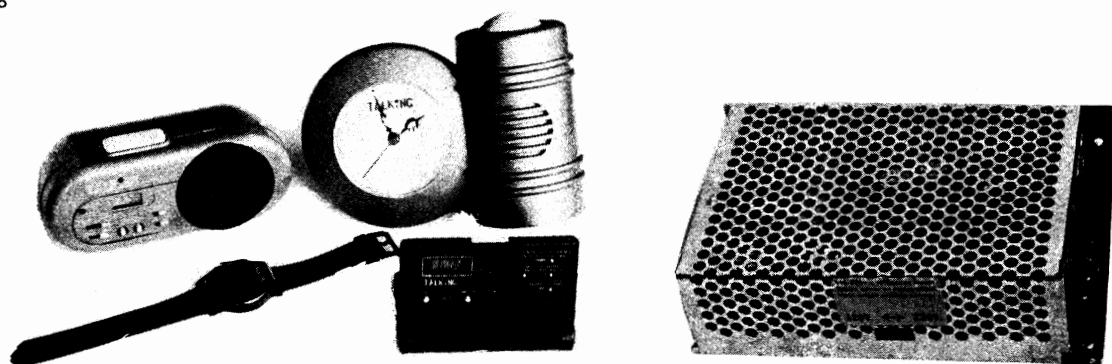
为使这个老企业焕发出新的历史使命，他们将奋力发展经济和外向型经济为“紫金”产品的开拓与发展。为通讯与计算机的腾飞干一番事业。

# 电子工业出版社广州科技公司

## 综合经营部邮购信息

本公司已于93年6月在穗正式开业。利用有关宣传媒介,通过邮购为读者服务,是公司的业务之一。

本公司自93年6月开始,将陆续在《电脑》杂志上介绍深圳、珠江三角洲及港台有关公司、厂家的一些电子产品,电脑配件,电脑软件以及电子版新书等,欢迎广大消费者惠顾垂询。



邮购商品一览表

| 品 名                | 规 格                                                                  | 单 价       | 邮 费     |
|--------------------|----------------------------------------------------------------------|-----------|---------|
| DAKE 打印机共享器        | 2×1 (配 2 根电缆)                                                        | 320 元 / 台 | 5 元 / 台 |
| DAKE 打印机共享器散件(含外壳) | 500 套以上起售                                                            | 195 元 / 套 | 按实际费用结算 |
| 单片机应用系统开关电源        | 5V / 10A, +12V / 1A, -12V / 1A;<br>外形尺寸: L158×W98×H48mm <sup>3</sup> | 175 元 / 台 | 5 元 / 台 |
| 单片机应用系统开关电源        | 5V / 15A (LED 大屏幕显示用)<br>尺寸同上                                        | 175 元 / 台 | 5 元 / 台 |
| 单片机应用系统开关电源        | 5V / 10A, +12V / 1A, -5V / 2A<br>尺寸同上                                | 175 元 / 台 | 5 元 / 台 |
| 软件狗 WATCH DOG      | 软件硬加密工具<br>体积: 60×50×15mm <sup>3</sup>                               | 200 元 / 台 | 5 元 / 台 |
| 袖珍固体留言录放机(不能放磁带)   | 存贮时间 20 秒                                                            | 82 元 / 台  | 5 元 / 台 |
| 桌式语言报时钟 (带石英钟)     | 可自动每小时报时一次, 带闹                                                       | 75 元 / 台  | 5 元 / 台 |
| 桌式语言报时钟            | 同上                                                                   | 60 元 / 台  | 5 元 / 台 |
| 盒式语言报时钟            | 同上                                                                   | 55 元 / 台  | 5 元 / 台 |
| 语言报时手表             | 同上                                                                   | 60 元 / 只  | 5 元 / 只 |

以上产品, 均为正式厂家产品, 如出现质量问题, 可在二个月内退换。

### 邮购注意事项:

- 1、来函及汇款要用挂号邮件寄来。
- 2、收货人的邮编、地址、姓名均要用正楷书写清楚。
- 3、以上商品均可批发, 批发价另议。

邮购地址: 广州市五山路华南师大科技服务楼 215 室

邮编: 510630      联系人: 陈 昊, 周青峰