

# 計算機應用研究

1991

2

APPLICATION RESEARCH OF COMPUTERS 《計算機應用研究》雜誌





## 《計算機應用研究》雜誌辦刊單位

四川省電子計算機應用研究中心 新疆電子計算中心  
貴州省科委計算中心 甘肅省計算中心  
安徽省計算中心 廣西計算中心  
吉林省計算中心 青海省測試計算中心  
四川省電子學會

## 《計算機應用研究》雜誌編輯委員會

主任委員：張執謙

副主任委員：李澤民

委員：曾光初 賈洪鈞 張國棟  
王小華 羅海鵬 劉鐵軍 余凱

1991年第2期（總第40期）

出版日期：1991年3月 本期責任編輯：李澤民

計算機應用研究（雙月刊）  
（公開發行）

國內統一刊號：CN51—1196

主編：張執謙

副主編：李澤民

編輯出版：《計算機應用研究》雜誌社

通訊地址：成都市人民南路4段11號附1號

郵政編碼：610015

印刷：新都一中印刷廠

訂閱處：全國各地郵局

總發行：成都市郵政局

郵發代號：62—68

廣告經營許可證：川蓉工商廣字005號

每冊定價：1.80元



# 计算机应用研究 第8卷 第2期(总第40期)

## 目 录

### 软件篇

专家评估算法及软件支持系统.....	刘定君 瓮正科 龚宇清 杨晋峰(1)
微机市场预测猜测变量模型的探讨.....	张鸿鸣(5)
数据处理组合软件的原理与技术.....	高卫民(7)
八皇后问题定解条件的简化.....	廖大勇(11)
MOTHER——管理信息系统生成系统.....	张小强 刘定君(12)
试论多种程序语言的综合应用.....	钟廷娇(14)
通用文本文件显示程序.....	王晓武(16)
一个小型显示程序.....	柯昌献(16)
内存驻留程序的通用设计环境.....	肖俊良(17)
通用报表自动打印程序的设计与实现.....	赖小强(20)
快速修改汉字库.....	郭继展(23)
可在 PC-1500 机上运行的流程图绘制软件.....	杨大顺 陶明华(26)
介绍一例打印机绘图程序.....	别社安 汪克让(29)
实用小程序.....	黄焕如(30)
菜单工具软件 MENUTOOLS 的设计与实现.....	王道顺 孙国良(31)
dBASE II 实现下拉式菜单的方法.....	姜振斌(35)
反弹加密.....	何其才(36)
让计算机来演奏你喜爱的乐曲.....	潘作秋(37)
一种新颖方便的“P”解密方法.....	陈庆章(49)
微机辅助生成信息系统数据流程图.....	方 明(56)
友好的多窗口磁盘菜单操作系统.....	李志伟(59)
dBASE II PLUS 中 MEMO 字段的格式化打印方法.....	许建潮(60)

### 硬件篇

单片机控制系统硬件设计研究.....	刘熙 龙卫红 程智强(40)
--------------------	----------------

### 系统篇

一个实用化的语音分析/合成系统的设计.....	刘榕和(45)
-------------------------	---------

### 维修篇

算法语言查错技术及查错词典.....	牛海发(42)
香烟锡箔的妙用.....	张伟中(51)
再谈硬盘维护.....	李建华(52)
硬盘系统故障的维修方法.....	任 基(54)
PC-XT 系统板维修一例.....	胡 红(58)
2024 打印机怪癖故障二例.....	齐 毅(55)
磁带机机械调整.....	陈兰英(10)
PC 机软盘故障分析与维修.....	牛海发(61)
长城 GW-0520DH 微机硬盘驱动器控制卡维修一例.....	黄 岗(62)

### 信息篇

袖珍机面临的问题及其对策.....	曹来发(13)
简讯 8 则.....	(4, 13, 19, 28, 34, 44, 50, 63)

## 《计算机应用研究》杂志办刊单位

四川省电子计算机应用研究中心  
安徽省计算中心  
吉林省计算中心  
甘肃省计算中心  
河南省计算中心

贵州省科委计算中心  
新疆电子计算中心  
广西计算中心  
青海省测试计算中心  
四川省电子学会

## 《计算机应用研究》杂志编辑委员会

主任委员：张执谦

副主任委员：李泽民

委员：贾洪钧 曾光初 龚宇清  
张国栋 罗海鹏 刘铁军  
余凯 杨剑波

## 《计算机应用研究》杂志社董事会

董事长：周赛渝

董事：唐珍 孙传江 陆慰椿  
吴地兴 郑国基 黎蓉  
黎瑰常

---

### 计算机应用研究（双月刊）

（公开发刊）

一九九一年 第二期  
第八卷（总第40期）

主 编：张执谦  
副 主 编：李泽民  
本期责任编辑：李泽民

编 辑 出 版：《计算机应用研究》杂志社

内 文 印 刷：新都一中印刷厂

封 面 印 刷：成都市玉泉美术印刷厂

出 版 日 期：1991年3月31日

本刊通讯地址：成都市人民南路4段11号附1号

本刊邮政编码：610015

订 阅 处：全国各地邮局

总 发 行：成都市邮政局

---

每册定价：1.80元

本刊邮发代号：62-68

国内统一刊号：CN51-1196

广告经营许可证：川蓉工商广字005号

# 专家评估算法及软件支持系统

新疆电子计算中心 刘定君 瓮正科 龚宇清 杨晋峰

**摘要:** 本文将特尔斐法和交叉影响法揉合在一起,对专家可评审的问题,归纳出六种数据类型,并给出每种数据类型的分析处理算法。针对各类数据研制了从评审表的制定开始到预测数据管理、数据处理,最后进行结果输出的通用“专家评估法软件支持系统”,最后给出该软件在“新疆 2000 年计算机发展战略与对策”预测过程中的应用情况。

**一、引言:** 预测技术可分为定量预测、定性预测和主观概率预测法。定量预测是根据历史数据和数学模型得到预测结果的;定性预测通常是根据专家们的思维过程得到“最好的”预测结果的;主观概率预测法是根据决策者的一些主观判断估计得到一种概率的,而不是根据对历史信息的客观分析。

专家评估法是定性预测中常用的一种方法,在缺乏足够的统计数据和原始资料的情况下,利用专家评估可以对某些技术问题或社会问题作出定量分析估计,得到这些文献上还未反映的信息。下面先介绍专家评估方法数据类型及相应的分析处理算法,然后介绍专家评估法的计算机软件实现,最后给出该软件系统的一个使用实例。

**二、专家评审问题类型及分析处理算法。** 专家评审问题调查表是专家评估法的重要工具,是信息的主要来源,专家可评审问题尽管内容、形式繁多,但从其评价数据类型看,可以归纳成以下几种。如果根据问题类型制定调查表,则可方便地对问题进行评价分析处理,获得较好的预测结果。

**(一)专家评审问题的类型** 1. “定量法”评价,要求专家对某问题(或某方案)给出一定量的评价,如给出事件完成时间,事件发生概率,某设备达到的数目,某问题的重要性、正确性等

2. “比重法”评价,要求专家在某问题的各个方案中分配 100 分,以确定各个方案在该问题中所占的比重。

3. “选择法”评价,一是要求专家在某问题的各个方案中选一最优者;二是要求专家在某问题的各个方案中选出满足要求的方案。

4. “排队法”评价,要求专家对某问题的各个方案之间的相对重要性进行评价,一般对各方方案给出评价

分数(0—100)和建议等级(1—P, P 为方案个数)。

5. “交叉影响法”评价,如果各预测事件间存在着相互关系,则一种事物发生的概率将受到相关事物的发生与否的影响,交叉影响法就是对相关事件组成的系统进行计算仿真实验,以提高预测的可性,交叉影响法是对特尔斐方法的一个补充,它要求专家对事件发生概率及各事件间的交叉影响程度做出主观判断。

6. “专家权威性”评价,采用自我评价和审查评价互为补充的方法。首先由专家自己填写发给的一张有关科学技术的自我评价表,然后经组织者作适当计算调整,得到不同专家对不同问题的权威系数,以供数据处理时参考。

**(二)专家评审问题的分析处理算法** 专家们对上述前四类问题的评价均可排列成下面的矩阵:

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{k1} & a_{k2} & \cdots & a_{kn} \end{bmatrix} \quad K \times n$$

其中, n 为参加评审的专家数, k 为该类问题的方案总数。

$k = k_1 + k_2 + \cdots + k_m$ 。  $K_i$  为第 i 个问题的方案数,  $i = 1, 2, \cdots, m$ , 为该类问题的问题数。显然,所有专家对某一方案的评价组成矩阵 A 的一行,那么对前四类问题的处理可划为矩阵 A 的数据操作,现分述如下。

1. “定量法”评价数据处理方法一般有二种,一种是对年代采用四分法,一种是对事件的发生概率,问题的技术参数等采用定量数值平均值法。

**(1)事件完成年代的四分算法** 中位值为其最大概率的完成年代,下四分点为乐观完成年代,上四分点为悲观完成年代。

While  $i \leq k$  do

begin 对  $a_{ij}(j=1, 2, \cdots, n)$  排序;



if  $n = \text{奇数}$ , 中位值:  $= a_i, \frac{n+1}{2}$ ;

下四分点:  $= a_i, \frac{n+1}{2} - \frac{n}{4}$ ; 上四分点:  $= a_i, \frac{n+1}{2} + \frac{n}{4}$ ;

if  $n = \text{偶数}$ , 中位值:  $= (a_i, \frac{n}{2} + a_i, \frac{n}{2} + 1) / 2$ ;

下四分点:  $= a_i, \frac{n}{2} - \frac{n}{4}$

上四分点:  $= a_i, \frac{n}{2} + \frac{n}{4} + 1$ ;

$i := i + 1$ ;

end

end

(2) 对事件发生概率、问题技术参数等的平均值算法  
平均值即为  $n$  个专家对某问题某一方案的定量评价,  $Q_{wj}$  为第  $j$  个专家对该问题的权威系数。

while  $i \leq k$  do

begin 对每一行求总和  $\text{sum}$ ; 求加权总和  $Q_{\text{sum}}$ ;

均值:  $= \text{Sum}/n$ ; 加权均值:  $= Q_{\text{sum}} / \sum_{j=1}^n Q_{wj}$ ;

end

end

2. “比重法”评价数据处理算法 把分数区间(0, 100)五等分, 然后统计专家对某方案评价的分数落入这五个区间的评价数占总评价人数的比重。则最大比重的区间数值即为该方案在问题中占的比重。

while  $i \leq k$  do

begin 统计评价分数落入上述五个区间的数目;

统计加权评价分数落入上述五个区间的数目;

比重:  $= \text{评价数}/n$ ; 加权比重:  $= \text{加权评价数} / \sum_{j=1}^n Q_{wj}$

end

end

3. “选择法”评价数据只有“0”和“1”。规定专家选中的方案其值为“1”, 落选的为“0”, 则对矩阵  $A_{kn}$  的每一行统计“1”的个数, 求出“1”占的比重即为该方案被选的比重, 比重最大者为最优, 次大者为次优。具体根据需要来选择方案。

“选择”算法

while  $i \leq k$  do

begin 统计“1”的数目  $\text{Sum}$  (相当对于  $A_{kn}$  的每一行求和),

计算“1”加权后的总和  $Q_{\text{sum}}$ ;

方案被选择比重:  $= \text{Sum}/n$ ;

方案加权被选比重:  $= Q_{\text{sum}} / \sum_{j=1}^n Q_{wj}$

end

end

4. “排队法”评价数据相对重要性处理算法

各方案均有二个值: 分数 0—100, 建议等级 1— $p$ ,  $p$  为该问题的方案数, 通过专家评价的集中程度, 协调程度以及协调程度的统计显著性对各方案的重要性进行排队, 矩阵  $A_{kn}$  的奇数行为评价分数, 偶数行为建议等级。

集中程度用方案的满分率和均值来衡量。

协调程度用变异系数  $v$  和协调系数  $w$  体现, 变异系数代表对方案评价相对波动的大小, 是方案的标准差  $S$  和均值  $M$  的比值, 协调系数代表全部  $n$  个专家对全部方案评价的相对波动

$$w = 12 \sum_{j=1}^n d_j^2 / n^2 (p^3 - p)$$

其中:  $\sum_{j=1}^n d_j^2$  是各个方案等级和与各方案等级和的算术平均值的离差平方和。

协调程度统计显著性采用泊松准则进行

$$X_k^2 = \sum_{j=1}^n d_j^2 / pn(n+1)$$

自由度  $f = n - 1$ , 根据指定的置信水平  $\alpha$  和自由度  $f$ , 查  $t$  分布表得到  $\alpha$  下  $f$  自由度时的  $X_k^2$  值如  $X_k^2 > X_k^2$ , 则说明专家评价的协调程度是显著的, 亦即专家意见协调。

相对重要性处理法

对奇数行: 求分数均值  $M_i$ , 加权分数均值  $QM_i$ ;

求标准差  $S_i$ ; 加权标准差  $QS_i$ ;

变异系数:  $= S_i / M_i$ ; 加权变异系数:  $= QS_i / QM_i$

对偶数行: 求等级总和  $M_i$ , 加权等级总和  $QM_i$

求各方案等级和均值  $R$ ; 各方案加权等级和均值  $QR$ ;

$$\text{求各方案离差平方和 } \sum_{j=1}^n d_j^2 = (R - M_i)^2;$$

$$\text{加权 } \sum_{j=1}^n d_j^2 = (QR - QM_i)^2;$$

$$\text{协调系数 } w: 12 \sum_{j=1}^n d_j^2 / n^2 (p^3 - p);$$

$$\text{加权 } Q_w: 12 \sum_{j=1}^n Qd_j^2 / (\sum_{j=1}^n Qw_i)^2 (p^3 - p);$$

$$X_k^2 = \sum_{j=1}^n d_j^2 / pn(n+1); QX^2 R = \sum_{j=1}^n Qd_j^2 / p (\sum_{j=1}^n Qw_j)$$

$$(\sum_{j=1}^n Q_{wj} + 1);$$

“排队法”评价的数据处理方法比较复杂,当计算的协调系数  $W(0 \leq w \leq 1)$  很小时,说明专家们的评价不协调,此时可能是由于专家组中存在一些持异端意见的高度协调组,这时需要找出各协调组,分析它们的意见,以取得较好的预测信息根据两两专家评价的“接近量度”可以找出高度协调组;根据两两专家评价等级的相关系数可以找出持异端意见的专家组,具体方法略。

5. “交叉影响法”是根据专家的主观判断和估计确定事件发生的概率及其各事件之间的相互作用,这种作用由专家评价的强弱程度,转换为定量的数值,全部组成一交叉影响矩阵,规定一个公式(如  $p_a = p^n + k \cdot p_a(1 - p_a)$ )——公式(11)用来计算已知某一事件发生后,其它事件的概率变化,亦即对各事件交叉影响下的发生概率进行计算机仿真实验。

(i) 求出全部专家回答的交叉影响矩阵的均值矩阵,对均值矩阵进行下述计算;

(ii) 随机选一事件  $D_n$ ;

(iii) 产生一个 0 到 0.99 之间的随机数  $R$ , 如  $R > R_n$ , 则  $D_n$  未发生, 如  $R < R_n$ , 则  $D_n$  发生, 此时要根据某公式(如公式(1))修正其它事件的概率;

(iv) 再随机选一未被选出的事件, 重复(ii)和(iii), 直至所有事件都被选出一;

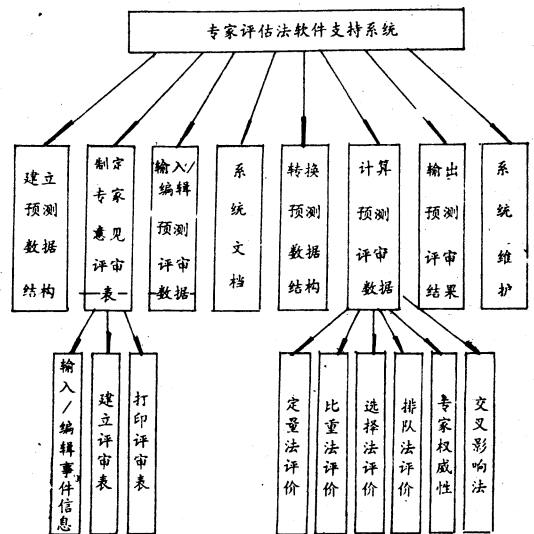
(v) 一次“试验”结束, 各事件恢复初始概率;

(vi) 大量重复(ii)~(v)步骤(千, 万次)以前, 每个事件选出后, 确实发生的累计次数除以总的试验次数, 即反映了事件交叉影响下的修正概率。

**三、专家评估法软件支持系统** 该系统由 dBASE-3 和 PACAL 实现, 它试图为各类使用专家评估法的预测、决策者们提供一个方便的预测辅助工具。它有两个特点, 一是采用动态数据结构, 在每次预测时首先根据预测需要建立数据结构, 这样就大大节省了存储空间, 方便了具体计算; 二是适应性强。由前所述, 不同预测的专家可评审问题, 其评价数据均可归纳成上述几种数据类型。本软件系统不但实现了上述的算法, 还可以追加新的数据类型和算法。预测者们可使用此软件系统从制订专家意见评审表开始, 到评审数据录入、转换、计算、输出(表、图)通过人——机对话来实现。

1. 系统总体结构 本系统采用模块结构, 由八个功能独立的模块组成, 使用时, 只需要根据一定的顺序运行, 即可得到专家评估法的评审结果, 本系统的控制结构如图(1)所示。

2. 系统功能 下面按一般预测工作流程分别介绍本系统的八个功能模块。



图(1)系统控制结构

(1)“制定专家意见评审表”, 用 dBASE-3 实现, 首先输入预测问题的有关信息, 然后根据具体的需要建表, 建好的表全存入数据库中, 需要时可随时打印多份, 同一张表可以打印二种形式。

(2)“建立预测数据结构”, 用 dBASE-3 实现, 根据输入的参数建立相应的数据库结构, 用来存贮各种预测数据, 这种数据结构是为方便用户输入数据设计的。

(3)“输入/编辑预测评审数据”, 用 dBASE-3 实现, 可同时输入、修改、删除, 由于预先建立了预测问题辅助信息, 所以保证了在输入/编辑过程中能实现基本的自动检测。

(4)“转换预测数据结构”, 用 dBASE-3 实现, 把录入数据结构表(1)转换成便于计算的数据结构表(2), 亦即矩阵  $A_{kn}$ , 对预测数据的计算处理, 只需对表(2)结构中的每一行施行同样的处理。

专家号	问题号	方案 1	方案 2	...	方案 p
-----	-----	------	------	-----	------

表(1)录入数据结构

问题号	方案号	专家 1	...	专家 n
-----	-----	------	-----	------

表(2)计算数据结构

(5)“计算预测数据结构”, 用 Pascal 实现, 现可进

行六种数据的计算,(一)定量法;(二)比重法;(三)选择法;(四)排队法;(五)交叉影响法;(六)专家权威性。将来吸收的方法可以通过“系统维护”功能连接到总模块下,计算的结果为表或图,它们均分别存放于结果数据库中。

(6)“输出预测评审结果”,用 dBASE-3 实现,对计算结果数据文件按指定的范围、方式输出。

(7)“系统文档”,是本软件系统的文档管理模块,可以打印或访问系统/使用说明书。

(8)“系统维护”,这是供熟悉 dBASE-3 的较高级用户使用的,当需要吸收方法或数据类型时,当支撑本系统的最基本结构元素需扩充时,可使用此模块对本系统进行维护。

**四、应用举例** “新疆 2000 年计算机发展战略与对策”预测研究采用二轮专家评估法。下面介绍第二轮咨询情况及处理,它一共包括 37 个事件,预测内容有:有关计算机事件的完成年代,计算机开发应用的方向及手段,计算机发展的有关政策措施等。

第一步,预测小组决定预测内容,列出具体的预测事件,并按评审数据类型将各事件分类,组织预测专家小组。

第二步,“制定专家意见评审表”,使用此功能模块可以制订专家评估法的各种咨询表,表的形式由用户自己决定,第二轮咨询中,共有七张咨询表,现只列举“选择法”评审意见调查表如表(3)。当然,制订咨询表必须首先输入预测事件的某些信息,如问题名称,方案数目等。使用此模块还可打印咨询表封面。

制定好各种咨询表后,分发给各个选定的专家,进行咨询。

第二轮咨询表附加第一轮咨询的处理结果,以供专家参考。

第三步,“建立预测数据结构”,当咨询表收完后,确定咨询的一些参考数(如参评的专家数等),根据这些数即可使用此模块建立预测数据相应的结构,建好结构即可输入预测数据。

第四步,“输入/编辑预测评价数据”,按不同的数据类型把收集到的咨询表送入相应的数据库中。

第五步,“转换预测数据结构”,当全部输入完成后,使用此模块即可把输入数据结构转换成计算数据结构。

第六步,“计算预测评审数据”,按不同的数据类型计算结果自动存入结果数据文件。

第七步,“输出预测评审结果”,按不同的问题类型输出结果,下面仅列出选择法第 3 个问题的计算结果,如图 2,柱高为方案被选的比重。“%”柱为考虑权威性的计算结果,“#”柱为不考虑权威性的计算结果,最高柱为最优方案,显然方案 5 为最佳方案,即 80% 的专家认为“改善计算培训组织”的最佳方案为“上述五个方案的综合”。

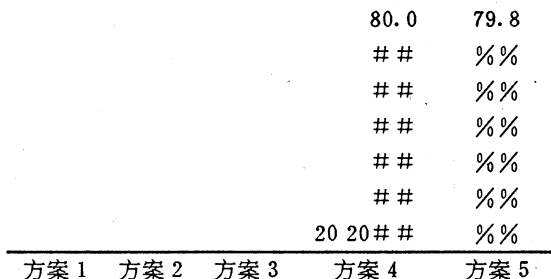


图 2 选择法第 3 问题处理结果

### “汉字高点阵激光打印程序”简介

为了适应出版业务和办公室公文处理等需要高质量的汉字打印文本的要求,我们使用普 II 型激光打印机开发出 48 点阵的汉字激光打印程序。该程序能够打印出 5 种字体,即宋体、黑体、仿宋体、楷体 and 繁体;17 种字型,两种打印方式(横向打印和纵向打印)。并且为了实现在同一页面上能够打印出不同字体,不同字型和不同打印方式的文本,程序设置控制符识别功能。打印幅面有 A4 和 B5 两种。与针式打印机、激光打印机相比,除具有高质量的打印文本外,还能一次性打印多达 100 份相同的文本。整个程序采用菜单操作,全汉字显示,不仅对纯中文文本,对中西文混合文本亦能打印出高质量的文本。(成都 陈勋)

需此软件者,可同本刊联系。

编号	问题	方案	评价值
	⋮		
3	改善计算机培训组织	1. 依靠大学培养的大学生	0
		2. 依靠社会进行计算机普及	0
		3. 依靠内地的计算机培训组织	0
		4. 建立区级培训中心,培训和提高在职人员	0
		5. 上述四个方案的综合	1



# 微机市场预测猜测变量模型的探讨

吉林省计算机技术研究所 张鸿鸣

**摘要:**本文在介绍了四种变量模型的基础上提出了微机市场预测的步骤。

预测的目的就是为了解决这一未来复杂而又不肯定的因素,并研究应付它的方法,向决策者建议各种可供选择的方案和可行性范围,它告诉决策者一个事件在各种不同条件下的可能结果和对策,这显然就成了当前人们向往美好未来的一种热门学科的原因所在。

市场预测是商业、企业、计划部门决策过程中不可缺少的部分。随着管理工作日益科学化,科学的预测方法也就越来越需要了。市场需求预测方法一般分两类:一类是时间序列的预测方法,一类是因果模型方法。因此,研究猜测变量模型对进行市场预测十分必要。

我们用 $(\partial q_i)/(\partial q_j)$ 表示第 $j$ 个厂家为它增加一单位的产量,将会引起第 $i$ 个厂家也会随之增加的产量。若第 $j$ 个厂家自认为其增加产量时,第 $i$ 个厂家的产量仍会保持不变,则 $\partial q_i/\partial q_j = 0$ ,同理,用 $\partial q_i/\partial q_j$ 表示第 $j$ 个厂家认为它单价变动一个单位,将会引起第 $i$ 个厂家的价格变动的数量,若第 $j$ 个厂家自认为它的价格变动不会引起第 $i$ 个厂家的价格变动,则 $(\partial q_i)/(\partial q_j) = 0$ 。

在猜测变量模型中,属于猜测产量变量的有古尔诺模型和斯达克贝尔模型等。属于猜测价格变量的有贝兰德模型和市场份额模型等。

**一、古尔诺模型。**早在1838年,法国数理经济学家古尔诺曾研究过双头垄断厂商的行为而建立此模型。在此模型中有三个基本假定:(1)双头垄断厂商的产品同质;(2)猜测产量的变量为零,即

$$\frac{\partial q_A}{\partial q_B} = \frac{\partial q_B}{\partial q_A} = 0$$

(3)两厂商采取相同的价格。

此模型要解决的问题是:

- ①每个厂商将生产多少产品?②商场价格为多少?
- ③每个厂商的利润为多少?

设A、B为模型中的两个纯质双头垄断厂商,他们各自所面临的需求曲线是共同的市场需求曲线 $p=F(q_A=q_B)$ ,则A、B厂商的成本函数分别为:

$$\begin{aligned} \Gamma TC_A &= TC_A(q_A) \\ \Gamma TC_B &= TC_B(q_B) \end{aligned} \quad (1)$$

利润函数分别为:

$$\Gamma V_A = TR_A - TC_A = F(q_A + q_B)q_A - TC_A \quad (2)$$

$$\Gamma V_B = TR_B - TC_B = F(q_A + q_B)q_B - TC_B$$

从(2)式可以看出,每个垄断厂商的利润,不仅是自家产量的函数,而且也是竞争对手厂商的函数。就A厂而言,除非他知道B厂的产量,或者以任何方式猜测B厂的产量,否则他将无法决定对自己最有利的产量,一旦他猜测(认定)B厂的产量为 $q_B$ 时,A厂将在条件

$$\Gamma \frac{\partial V_A}{\partial q_A} = MR_A(q_A) - MC_A(q_A) = 0 \quad (3)$$

$$\Gamma \frac{\partial^2 V_A}{\partial q_A^2} = MR'_A(q_A) - MC'_A(q_A) < 0 \quad (4)$$

之下生产。同理当B厂猜测3的产量为 $q_A$ 时,他将在条件

$$\Gamma \frac{\partial V_B}{\partial q_B} = MR_B(q_B) - MC_B(q_B) = 0 \quad (5)$$

$$\Gamma \frac{\partial^2 V_B}{\partial q_B^2} = MR'_B(q_B) - MC'_B(q_B) < 0 \quad (6)$$

之下生产。显然,从(3)和(5)式可分别解得,

$$q_A = TP_A(q_B) \quad (7)$$

$$q_B = TP_B(q_A) \quad (8)$$

对每一厂家为得到最大利润,必须各自依据关系式(7)和(8)来决定自己产量,称(7)和(8)式为两个厂家的古尔诺反应函数。反应函数(7)和(8)可以作出图形,表示一种形态。

A厂依据B厂的每一种猜测定出自己的产量 $q_A$ 。于是,满足(7)式的每一产量组给 $q_A, q_B$ ,在A厂的反应函数 $q_A = TP_A(q_B)$ 上。同样,B厂也会因自己对A厂的猜测而定出自己的产量 $q_B$ ,因此这产量组合的轨迹构成了B厂的反应函数 $q_B = TP_B(q_A)$ 。然而,除两函数的相交点外反应函数上的任何一点都表示两厂家对对方厂的产量所作出的错误猜测,他们必须经过多次的调整方可准确地判断出对方的产量,以作为决定自己厂产量的依据,而这个产量的组合恰是两个反应函数的交点。依据古尔诺的观点,交点便是双头垄断厂商的均衡点。因此,有的学者称此交点为“古尔诺均衡点”或“古尔诺解”。

**二、斯达克贝尔模型。**在古尔诺模型中,隐含着两个厂商皆为跟随者的假设。所谓跟随者,是指其产量的多少须视其竞争对手的产量多少而定。因此,跟随者必有反应函数存在。但实际的双头垄断厂商很可能不像

古尔诺模型所描述的那样“单纯”，即一方很可能利用对方的单纯性，以事先的机智来决定自己的产量，使自己处于某种领导地位而成为领导者。所谓领导者，是指厂家自认为是领导者，并猜测他的对手为跟随者，他视其对手的反应函数来决定自己的最大利润产量。1934年，德国经济学家斯达克贝尔把纯质双头垄断厂商分为四种情况，从而建立了双头垄断的另一种模型——斯达克贝尔模型。

情况 1: A 厂为领导者, B 厂为跟随者; 情况 2: A 厂为跟随者, B 厂为领导者; 情况 3: A、B 厂皆为跟随者; 情况 4: A、B 厂皆为领导者。

对上述四种情况进行具体分析, 并代入数字得下表:

斯达克贝尔模型四种情况的结果 表 A

情 况	p	q	q	q+q	V	V
A 为领导者, B 为跟随者	72.5	25	27.5	52.5	158.3	152.08
A 为跟随者, B 为领导者	71.67	17.5	37.5	55.0	103.18	212.63
A、B 皆为跟随者	79.33	20	30	50	150	200
A、B 皆为领导者	69.17	25	37.5	62.5	75.08	118.88

(表中:  $q_A$ 、 $q_B$  分别为 A、B 厂的反应函数;  $V_A$ 、 $V_B$  分别为 A、B 厂的利润;  $p$  为市场价格。)

由表 A 可知, 对 A 厂而言, 他为领导者, 而 B 厂为跟随者时, A 厂的利润最大。同样对 B 厂而言, 他为领导者而 A 厂为跟随者时, B 厂的利润最大。而 A、B 皆为领导者时, 两厂的利润都较低, 但市场的价格低, 行业的产量大。

**三、贝德兰模型。**1883 年, 法国数学家贝德兰曾对古尔诺的猜测产量变量提出批评, 认为假设其对手的价格不变是不妥当的, 他认为, 寡头之间应有激烈的价格竞争, 且不论市场价格如何, 只要任一厂家降低价格而其他对手的价格不变, 这一厂家便能在降低价格的过程中得到好处。故而提出了以价格为猜测变量, 并假定猜测价格变量为零, 即  $\partial p_B / \partial p_A = \partial p_A / \partial p_B = 0$ , 而建立此模型。

依据贝德兰的观点, 寡头垄断厂商将会在价格设定方面动脑筋, 他们相互猜测对方的价格, 于是价格便成为独立的自变量, 而产量(需求量)便成为因变量。在猜测价格的基础上, 便产生了他们各自的需求曲线。

设 A、B 厂家的需求曲线分别为

$$q_A = D_A(p_A, p_B) \quad (9)$$

$$q_B = D_B(p_A, p_B) \quad (10)$$

而厂家的成本函数分别为

$$TC_A = TC_A(q_A) \quad (11)$$

$$TC_B = TC_B(q_B) \quad (12)$$

则他们的利润函数分别为

$$V_A = p_A \cdot q_A - TC_A(q_A) \\ = p_A \cdot D_A(p_A, p_B) - TC_A[D_A(p_A, p_B)] \quad (13)$$

$$V_B = p_B \cdot q_B - TC_B(q_B) \\ = p_B \cdot D_B(p_A, p_B) - TC_B[D_B(p_A, p_B)] \quad (14)$$

在他们各自认定对方的价格为一常数时, 他们各自的最大利润必须满足

$$\frac{\partial V_A}{\partial p_A} = D_A(p_A, p_B) + p_A \frac{\partial D_A}{\partial p_A} - \frac{dTC_A}{dq_A} \cdot \frac{\partial D_A}{\partial p_A} \\ = q_A + (p_A - MC_A) \frac{\partial D_A}{\partial p_A} = 0 \quad (15)$$

$$\frac{\partial V_B}{\partial p_B} = q_B + (p_B - MC_B) \frac{\partial D_B}{\partial p_B} = 0 \quad (16)$$

由(15)和(16)式联立组成方程组, 可求得两厂家的价格  $p_A$  和  $p_B$ , 再将其代入(9)和(10)式, 便可得两厂家的产量  $q_A$  和  $q_B$ 。

寡头之间的这种价格猜测, 会产生一些连锁反应。当任一厂家, 比如说是 A 厂察觉了 B 厂的价格较低时, 他为夺回丢掉的市场, 他会重新设定他的产量和价格。如此继续下降的结果, 是市场价格的连续降低。当两厂的成本相同时, 则成本的高度可阻止价格的连续, 而使厂家利润减到零。从诸多的寡头竞争的实际情况来看, 如果说古尔诺模型把寡头厂家描述的具有某种“单纯”性的话, 那么贝德兰模型的成功性也很小。因为他们描述的寡头之间的这种价格战最终使他们事与愿违, 甚至使一些成本较高的寡头厂家在竞争的压力下遭到破产。因而, 在寡头之间竞争的同时, 为了他们各自的利益往往存在着某种形式的勾结。

**四、市场份额模型。**前面几个模型的分析都是建立在厂家追求利润最大的原则上进行的。然而有些厂家(多是生产异质产品的厂家)为了维持他某一水准的长期利润, 可能不去追求眼前的最大利润, 而只是希望自己的产品在市场上维持一定的市场占有率。今以一实例说明如下:

设在某一异质双头垄断市场中, A 厂希望自己的产品市场占有率为 1/3。B 厂的需求函数和成本函数分别为

$$P_B = 155 - 3q_B - q_A$$

$$TC_B = 2q_B^2 + q_B + 78$$

求: ①A 厂的产量应为多少?

②B 厂的产量、价格和利润各为多少?

解: 因为

$$\frac{q_A}{q_A + q_B} = \frac{1}{3}$$

# 数据处理组合软件的原理与技术

常州市统计局计算站 高卫民

**摘要:**本文提出一种有别于通用报表程序的,而用于数据处理系统设计的工具软件——数据处理组合软件的设计思想与方法。根据数据处理的特点,就软件的原理、可行性、优越性及设计技术进行了详细叙述。

一、数据处理组合软件的原理及特点。“组合”是指将几个独立部分进行组装。数据处理组合软件的原理就是设计出数据处理过程中完全不同的独立程序模块,既可以独立地完成某一项任务,又可以通过“组合”方式将这几个独立模块组装成一个特定的应用系统,完成指定的系统任务。因此,“独立性”与“可组合性”是该软件的主要特点。我们知道,数据处理从功能上化分可有

数据录入、审核、检索等,对应的程序模块的操作对象与操作结果都是数据文件,程序模块之间很少有状态信息与控制信息传递,这就为数据处理组合软件的设计提供了极好的条件与可能。

数据处理组合软件的设计,从设计性能上讲,比一般通用数据处理要优越得多。主要表现在以下几个方面:

(接上页) 由此可知  $q_A = 0.5q_B$ , 此即为 A 厂的反应函数,代入 B 厂的利润函数中,得

$$\begin{aligned} V_B &= p_B \cdot q_B - TC_B \\ &= (155 - 3q_B - q_A)q_B - (2q_B^2 + q_B + 78) \\ &= 154q_B - 5.5q_B^2 - 78 \end{aligned}$$

$$\frac{\partial V_B}{\partial q_B} = 154 - 11q_B = 0$$

解得

$$q_A = 0.5 \times 14 = 7 (\text{单位})$$

$$p_B = 155 - 3 \times 14 - 7 = 106 (\text{元})$$

$$\begin{aligned} V_B &= 106 \times 14 - (2 \times 14^2 + 14 + 78) \\ &= 1484 - 484 = 1000 (\text{元}) \end{aligned}$$

由上面具体例子可知,若某一厂家希望自己的产品能占有一定市场分额,则他们必有反应函数存在。而他的竞争对手就可利用该反应函数找到自己利润最大的产量。此种情况与斯达克尔贝尔模型中,一个为领导厂家,而另一个为跟随厂家的情况相类似。所以不同的是,本模型的反应函数是基于维持某一市场份额,而斯达克尔贝尔模型的反应函数是建立在利润最大的基础上,故两者的含意不同。

总之,在我们的国家中,研究市场预测猜测变量模型也有极其重要意义,也完全适合我们国家中一些骨干厂家进行市场预测,合理地制定生产规划,指导骨干厂家的生产,保证市场供应,调解市场平衡。

## 五、微机市场预测。

①预测对象分析;②建立预测模型;③确定预测方法;④模型数学化;⑤编制微机程序;⑥微机计算、处理;⑦预测结果精度分析;⑧预测价值分析;⑨实际应

用。

在这些步骤中,建立预测模型最关键。

在前面比较系统的讨论了古尔诺模型、斯达克尔贝尔模型、贝德兰模型和市场份额模型等猜测变量模型。目的在于形象地描绘市场预测对象,在整体上表现出预测对象及其相关因素的联系、依存、变化和运动的关系。建立模型与所采取的理论有关,理论决定模型,不同的指导理论就有不同的模型。如果预测者不能清晰而透彻地说明预测对象的性质,他就不可能建立有效的预测模型,而没有模型也就不能进行任何的预测。因此,有了预测模型,就比较容易进行预测了。依据预测对象性质和预测模型,预测者就可以采用各种不同的方法进行预测。

在我们社会主义国家中,应用微机进行市场预测,也必须研究建立预测模型。只有这样才能提高预测精度,应用价值高的结果指导市场的需求,把握经济发展规律。自从计算机技术问世以来,微型计算机地应用得到广泛地发展,它立刻就成为市场预测的有效工具。应用微机进行市场预测也就是根据所建立的市场预测模型,编制出应用微机进行市场预测的数学算法,再根据这数学算法用计算机程序语言编出计算机程序。将市场预测中收集来的大量数据输入计算机,经过计算机计算和处理之后,得出精确度较高的结果,运用得到的结果去进行市场预测。

应用微机进行市场预测后,大大提高了市场预测的精度,可以比较准确地研究市场经济发展规律,保证市场需求平衡,造福于人类。



1. 程序设计模块化好, 独立性强 一般通用数据处理软件设计虽然也考虑了程序模块化, 但总需进行整体设计。而组合软件只强调功能模块的独立设计, 模块之间的唯一联系是数据文件本身。

2. 适用于多语种设计 一语种设计的程序调用另一语种设计程序的方法已广为程序设计者所采用。但这里所指的多语种设计的含义是: 每个独立模块由于其独立性, 它的设计不依赖于整个系统, 因而可以充分考虑选择满足独立功能、设计方便、灵活的程序语言。

3. 便于多用途、多功能的程序设计 由于将数据处理的功能模块化, 并且独立设计, 因而便于在局部功能模块上考虑多功能的选择与适用面的拓宽。

4. 组合系统的功能扩展性好 由于组合系统本身是由完全独立的功能模块所组成, 因而当组合系统有新的要求时, 一般可以通过增加新的功能模块的办法来解决。另外, 在单一功能模块上的功能扩展, 丝毫不会影响其它功能模块性能。

另外, 数据处理组合软件在使用上也有鲜明特点:

1. 满足单一功能处理要求, 这对于数据处理任务进行分项或分片完成时, 可使系统开销少, 操作简单。2. 系统可大可小, 用户可以根据数据处理要求选用部分独立功能模块组成系统。

## 二、数据处理组合软件的设计考虑

1. 增强独立功能模块的操作功能, 扩大适用范围

由于独立功能模块的设计目标单一, 因此程序设计可以充分在扩大适用范围与增强操作功能上下功夫, 而不至于使设计过分复杂和程序过分庞大。只有这样, 才能使得数据处理组合软件较一般通用数据处理软件优越。

2. 提供良好的用户界面与简单的程序接口 独立功能模块的操作对象与内容是由用户来描述的, 因而, 设计一个良好的用户描述环境, 对于软件的推广使用与减少误操作差错是十分重要的。然而, 用户描述还必须转换成程序接口信息, 才能控制程序的运行。程序接口要求尽量简单, 并能方便程序设计。

3. 各独立模块的设计规范性 虽然独立功能模块之间是相互独立的, 但在设计和操作上应考虑各模块之间的规范性, 这对于提高程序设计性能和保证用户界面友好具有重要意义。规范性可以从下面几个方面考虑:

(1) 数据文件格式 因为数据文件是各独立功能模块之间唯一用来传递操作结果信息的工具, 它应该在文件格式上、字段排列上考虑规范化, 这对程序设计 with 用户定义都有明显好处。

(2) 屏幕设计 作为用户与系统之间的窗口, 各独立功能模块从屏幕的整体设计上应尽可能考虑规范化, 使用户在使用由独立功能模块组成的组合系统时, 有一个整体感。

(3) 用户界面定义 用户界面定义应考虑屏幕对话形式(即屏幕设计)的规范性, 定义参数的存储格式(数据文件格式、文本文件格式、内存文件格式和变量形式)的规范化, 这将给系统的扩展、维护带来方便。

4. 屏幕的自动编排 屏幕的自动编排要求系统根据数据特点自动设计数据在屏幕上的位置, 这种设计主要在数据录入、检索模块中使用, 以使用户摆脱不必要的屏幕描述。当然, 也应考虑用户的特殊要求, 使用户可自行定义简单的屏幕编排方式, 如屏幕编排按指定分组设计, 选择数据项目的编排设计等。

5. 功能扩展性 组合系统的功能扩展, 可通过增加功能模块来解决, 而不会影响原系统性能, 所以不必在程序设计上特别考虑。然而, 独立功能模块的功能扩展需在程序设计上专门考虑, 具体可采用“软功能开关”设计方法。“软功能开关”是程序设计时, 为程序的内部功能与可选的操作功能之间设立开关标志, 而每个开关标志之后就是该功能处理程序。只要控制“开关状态”, 就可以控制模块中的功能实施。有些功能需要在程序的多处设置“开关”, 因而可以增设一个“总开关”控制。所以, 独立模块的功能扩展往往可以通过“带开关功能子程序”的设计, 插入原程序模块来实现。

6. 活菜单 对于数据处理的专用程序或在一定范围内的数据处理通用程序, 菜单目录往往是预先确定的。但对于数据处理的组合软件的每个独立功能模块, 它们的操作对象在设计时都是未知的, 因而必须采用“活菜单”设计技术。这里提供两种参考方法:

(1) 菜单自生成设计 系统根据用户界面定义操作对象后, 自动生成屏幕菜单编排。这种设计方式操作简单、直观, 但当屏幕容纳不了菜单内容时, 就必须采用分屏设计, 这在设计上增加了困难。

(2) 菜单目录检索方式 系统预设计一个用户操作屏幕, 同时提供用户检索菜单项目的操作命令, 用于用户不清楚输入选择时检索使用。这种方式相对前一种方式不太直观, 但操作比较方便, 程序设计也简单。

7. 数据文件的管理 系统对数据文件管理的好坏, 直接影响到数据处理的结果, 是系统性能评价的重要因素。在系统设计时, 主要考虑下面几个方面:

(1) 文件分类管理 根据数据的操作特点, 我们可以将数据文件分为: A. 原始数据文件; 即录入或批处理读入的未经加工处理过的数据文件; B. 目标数据文

件;经数据处理加工得到的数据文件;C.临时数据文件;在数据处理过程中建立的用于存放中间数据的文件。

(2)文件的建立与格式 临时数据文件是在数据处理过程中,需要时建立,用完后删除的。因此,不占用存储空间,它的文件格式可以从方便、灵活上考虑。对于原始数据文件和目标数据文件,可以考虑在用户定义结束时创建文件,也可以在数据处理过程中缺省时自动建立。这两个文件由于长期占用存储空间,因而应从文件格式上考虑尽可能少占空间。另外,原始数据文件的格式还应考虑到检索方便,目标数据文件的格式应考虑输出打印处理简单。

对于原始数据文件的格式,一般有:

#### A. 索引文件格式

对象	索引词 1	...	表 1 指针	表 2 指针	...
对象 1	值 11		地址 11	地址 12	
对象 2	值 21		地址 21	地址 22	
对象 3	值 31		地址 31	地址 32	
⋮	⋮		⋮	⋮	

这种存储方式的特点是公用一个检索区,占用存储空间少。但由于数据检索过程中文件之间的交换频繁,影响检索速度。

#### B. 在每个数据文件中建立索引区

对象	索引词 1	...	数据域
对象 1	值 11		对象 1 数据
对象 2	值 21		对象 2 数据
对象 3	值 31		对象 3 数据
⋮	⋮		⋮

显然,这种存储方式占用空间多,随数据文件数的增加和对象的增多,索引区占用的重得存储量是很可观的。然而,在指定数据文件的检索处理时,方便和速度快是显而易见的。

C. 作为以上两种存储方式的综合考虑,这种方式将少量的、使用频繁的和个别数据文件专用的索引词合并和数据文件中,而将其它不常用的索引词存放在索引文件中。这种方式虽然兼顾了前两种方式的长处,但程序处理要复杂的多。

对于目标文件格式,也可以有以下两种:A. 主栏

索引方式:这种目标数据文件无主栏名称栏目,但有主栏索引代码,数据文件输出打印时通过主栏索引代码,到指定主栏目录库中区检索得到主栏名称输出。

B. 带主栏名称的目标数据文件格式:这种方式因为自主栏名称栏目,因而输出打印处理简单,但占用存储量要比前一种方式多。

(3)文件的存储方式 如果将所有程序文件、目录文件、各种数据文件无规则地集中在一起,那么将给系统管理、维护、管理带来困难。一般地,程序文件与系统目录文件集中存储在一起,对于数据文件,如果也同程序集中存储在一起,那么最好对原始数据文件、目标数据文件、临时数据文件进行分类,有规则的命名。然而,更好的方法是,对原始数据文件、目标数据文件和临时数据文件名建立一个目录区,而数据处理对在三个目录建立搜索区,既保证了程序运行,又改善了系统管理环境。

(4)数据文件的转移 系统设计最好考虑在数据文件的转移之前,判断文件的存储量,以保证转移成功,这对硬盘与软盘之间的数据文件转移尤为重要。

### 三、数据处理组合软件的独立功能模块划分

1. 数据录入与修改模块 完成数据录入与修改操作,设计可考虑:(1)适应表长、表宽变化的数据表操作;(2)提供批数据处理;(3)提供录入查询手段;(4)可选按行、按列数据录入(修改)操作;(5)可选数据复录操作;(6)可选数据录入中审核;(7)可选数据校验码处理。

2. 数据检索模块 完成对数据的检索,设计可考虑:

(1)检索索引方式(按对象索引或按项目索引)选择;(2)检索索引词的选择;(3)检索输出方式(屏幕显示与打印输出)的选择;(4)检索项目的选择;(5)检索范围的确定;(6)批数据修改。

3. 数据审核模块 完成对数据文件的集中审核,设计应考虑:

(1)审核的数据范围选择;(2)数据的审核项目的选择;(3)被审数据的条件描述;(4)数据错误输出的定位方式;(5)审核关系的类型:行间、列间、表间关系及缺表情况。

4. 数据加工模块 完成对数据的各种加工处理,设计可考虑:

(1)数据叠加;(2)数据合并;(3)数据截取;(4)数据派生(计算);(5)数据汇编;(6)数据分组汇总(超级

## 磁带机机械调整

地矿部北京信息中心 陈兰英

**摘要:** 本文介绍 Wang CO-11 磁带机机械调整方法。

**一、PE/NRZI 读站校正** 该校之目的使读站(磁头)与磁带行走路径直,方法如下:(1)将示波器探头 I 接读数据电路板测点 TP603;将示波器探头接读数据电路板测点 TP703;(2)装一盘扭斜校准磁带,注意去掉写环,在读方式中观测读出信号峰值间的时间位移。如果时间位移小于 25micro-inches (75IPS 时应 < 0.33sec),则不需填隙片。

若探头信号滞后,且假设时间位移大于 25microinches,这时,需要在磁头总成的供带盘一边填隙片,填隙片数目可根据时间距离计算。(3)若探头信号超前,则需在磁头总成的主动轮一边填隙片。(4)填过隙片之后,需要重新检查时间位移,达小于 25microinches 为止。

**二、磁带轨迹调整** 当更换了主动轮、主动轮马达,磁头总成或读站方位被调节后,均需要进行磁带轨迹的检查与校准。进行这些机械调节前,必须先检查走带情况,所有导轮部件工作协调一致,以保证磁带边不损伤的前提下进行。

**1、调整方法:** (1)示波器探头 I 接读电路板 TP603,探头 II 接测点 TP703。

(2)装一盘扭斜校准带,正行走带,观察这两轨的模拟信号,峰值时间位移,记录 T1( $\mu$ s)和两者之间的相位关系,(3)去掉引导轮 B 的垫圈。(4)反行走带,观察外部轨迹间的位移,记录 T2( $\mu$ s),如 T1 与 T2 总是固定的相位关系,即  $T2 = T1 \pm 50/V$  (V 是传输速度)。说明轨迹是合适的,不需主动轮马达填隙片,

(续上页)

汇总);(7)数据追加;(8)数据排序;(9)数据表体旋转。

**5. 数据打印模块** 完成带主栏与不带主栏的数据文件的制表打印,设计应考虑:

(1)对不同型号打印机的适应;(2)字体、表长、表宽、分页等控制;(3)打印范围(主栏与宾栏)的选择;(4)主栏排序选择;(5)打印页号、日期、辅助表头选择。

**6. 文件转换模块** 完成数据文件的格式转换,设计时应考虑:(1)程序特定的数据文件格式与数据库文件或标准格式文件的互相转换;(2)转换的数据范围选择。

**7. 代码文件管理模块** 完成代码文件的录入、修改、检索、审核、打印处理,设计时应考虑:

(1)多级代码文件管理(即多个代码文件之间存在上、下级关系);(2)索引代码的唯一性;(3)一般代码的合法性;(4)备注(非代码)信息的输入;(5)代码的校验方式;(6)代码的可选分组打印。

**8. 系统初始化模块** 完成数据处理“组合”系统的操作准备,设计应考虑:

(1)各独立功能模块共用参数(由用户定义)的统一处理;(2)数据文件环境的清理与建立;(3)可能的“组合”系统操作要求确定。

**四、用户描述(定义)环境的设计**

**1. 系统预设缺省定义状态**,并提供即时状态的查询手段,可以减少用户的描述,简化操作,提供正确性。

**2. 对用户定义参数的检查**,避免非法描述。**3. 提供可选项目描述与顺序描述两种方式**。**4. 简化数据文件格式的描述**,方法之一是在一个数据文件中所有数据长度采用同一格式。**5. 提供“说明”与“举例”信息**,帮助用户掌握。**6. 提供系统操作目录文件以及系统代码目录文件之间的信息传送与接收方式**,提高系统描述的效率和加快系统的生成。**7. 屏幕的多窗口技术使用**,合理安排“选择窗口”、“状态信息窗口”、“操作提示窗口”等。

**五、程序的运行方式** 作为数据处理组合软件,可选独立功能模块的使用方式完成单项功能处理;也可选择有关独立功能模块组合使用,以完成系统数据处理任务。组合的方法可采用 DOS 系统下的批处理方式或 UNIX 系统下的 SHELL 程序控制,但更为简单的方式是编制一个“组合系统”主菜单生成程序,该程序根据用户选择的功能模块自动生成一个表明操作功能的主菜单,由该菜单控制程序产生的控制信息控制 DOS 系统的批处理或 UNIX 系统的 SHELL 控制程序。

**六、结束语** 数据处理组合软件作为数据处理系统设计人员的新的工具软件,可以大大加快各数据处理应用系统的研制,也有利于系统数据的规范化和操作规范化,该软件的使用对于计算机在数据处理领域里的推广应用无疑是十分有益的。



# 八皇后问题定解条件的简化

南充市高级中学 廖大勇

著名的高斯“八皇后问题”的定解条件一般表述为(原题附后):

(1)每行、每列最多只有一个皇后,即对任意  $I, J$  来说,  $A(I) \neq A(J)$ 。(数组  $A$  表示某行皇后所在列)

(2)每一条对角线上最多也只有一个皇后,即:

$A(I) - A(J) \neq I - J$  (45°对角线,  $I > J$ )

或  $-[A(I) - A(J)] \neq I - J$  (-45°对角线,  $I > J$ )

对于第二个条件可以合写为:

$ABS(A(I) - A(J)) \neq I - J$

因此,“八皇后问题”的定解条件通常都表示为:

$A(I) \neq A(J)$

或  $ABS(A(I) - A(J)) \neq I - J$

这个定解条件要用两个关系表达式来实现,仍嫌不够简洁。科学追求形式的完美与内容的高度统一。能否将上述定解条件进一步归纳为一个统一的表达式呢?

为了便于推导,不妨把上述定解条件改写成如下形式:

(1)、 $A(I) - A(J) \neq 0 * (I - J)$

(2)、 $A(I) - A(J) \neq 1 * (I - J)$ , 当  $A(I) > A(J)$

(3)、 $A(I) - A(J) \neq -1 * (I - J)$ , 当  $A(I) < A(J)$

容易看出,上面的三个式子中  $(I - J)$  前面的乘数 0、1 和 -1 都可由  $SGN(A(I) - A(J))$  分别提取出来,也就是说,“八皇后问题”的定解条件可以等效地表示为:

$A(I) - A(J) \neq SGN(A(I) - A(J)) * (I - J)$

稍作分析可知,本文的简化定解条件非常巧妙地概括了原定解条件的三种不同情况,却并未增加更多更深的内容,仅把  $ABS(x)$  函数换成  $SGN(x)$  函数,内容简洁、形式优美,真正做到了形式和内容的高度的统一与完美!

上机验证表明,本文简化定解条件对于“八皇后问题”的各种解法都能适用,对于推广的“八皇后问题”也是完全正确的!显而易见,这一简化的定解条件必定能使解古典“八皇后问题”的各种程序更加简洁易读、少占内存、执行效率更高!

\* 八皇后问题:在一国际象棋盘上,如何摆放八个皇后,使它们彼此不相吃? 皇后的吃法是可吃同行、同列、同对角线上的子。

(续上页“磁带机机械调整”)

这种情况下,可重新上好引导轮 B 的垫圈。(5)若反行走带扭斜超出了上面关系式的范围,则需给主动轮马达填隙片。拿掉主动轮马达固定角“X”和“Y”下面的所有垫片。(6)测量正行扭斜( $T_1$ ),然后,拿掉导轮 B 的垫圈,观察反行扭斜;通过按紧一点,靠近导轮 B 的磁带边缘,使其接近和远离仓板,测定所产生的反向扭斜(使  $T_2$  接近  $T_1$ )。(7)假若让磁带靠近仓板使扭斜有所改善,则主动轮马达需要在固定角“X”下面填隙片。假若让磁带远离仓板使扭斜得到改善,则需在固定角“Y”下面填隙片。(8)填隙片的数目根据扭斜的大小决定,所使用的填隙片有点 2mil(密尔 1%英寸=25.4)和 1mil 两种。(9)放入足够的填隙片,使  $T_2 = T_1$ , 相位对称。(10)重新放好导轮 B 垫圈。

**二、检查方法** 这项工作是做电子检查,以校验磁带正反方向走带轨迹相对关系,是否符合要求,是通过

正反方向走带比较两个外部测点模拟信号峰值的时间位移来确定的。

假如传输已全部统调,则两个方向的时间位移将是相同的,但两峰值间的相位关系相反的,由于磁带轨迹的精确直接取决于机械调整,以及允许的误差,完全的精确通常是做不到的,所以常确定一个范围,来保证两个方向可靠工作。

(1)示波器探头 I 接读电路板 TP603,探头 II 接 TP703。(2)装调扭斜带。正行走带(脱机方式),观察两轨波形峰值的时间位移(图(2))记录  $T_1$  和两轨之间的相位关系,反方向走带记录  $T_2$ 。(3) $T_2$  的值与  $T_1$  的对应关系可用下式表达。

$T_2 = -T_1 \pm 50\mu s/V$  (对应 75IPS 的  $50/V = 0.66\mu s$ )

式中:  $V$  代表传输速度

如果不符合上式,则必须校准传输

(参阅磁带轨迹调整步骤)。

# MOTHER——管理信息系统生成系统

新疆电子计算中心 张小强 刘定君

**摘要:**本文着重讨论了 MOTHER——管理信息系统生成系统的设计思想及系统结构,简单介绍了该系统的一些重要模块的功能,并描述了使用 MOTHER 系统生成用户应用系统的基本过程。

**一、设计思想** 管理信息系统是收集、存储及分析数据,以供企事业单位的各级管理人员使用的数据处理系统。这种系统的特点是面向管理工作提供管理所需要的各种信息,一个通常的管理信息系统的工作一般包括收集数据,输入到计算机存储起来,经过一定的运算分析处理形成结果并以某种形式输出表现出来。管理信息系统只是根据它所面向的管理业务的不同,业务多少的不同以及业务与业务间关系的不同而有所变化。一般管理信息系统都遵循这样一个模式。正是由于这样一个基本模式的存在才使得构造一个管理信息系统生成系统成为可能。

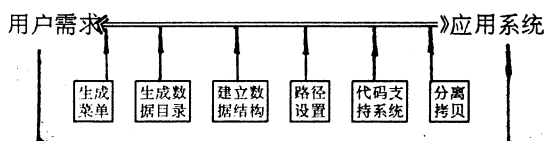
MOTHER 系统的设计目标是建立一个生成环境,在此生成环境中,提供一般管理信息系统所需模块的生成模块,并允许新模块的增加和对旧模块的维护;提供把模块进行组合链接,灵活设置的功能;另外,允许生成的结果与生成系统环境的最终分离,从而生成一个好象按用户要求定做的实用系统。

MOTHER 系统并不是从无到有地生成应用系统的程序,使用 MOTHER 系统只是在它提供的用户界面上把用户的要求加以具体的描述,MOTHER 系统提供应用系统菜单生成功能,数据目录生成功能,系统数据模型建立功能,模块库管理功能以及若干通用功能生成模块。提供用户应用系统所需要的功能模块。系统的所有数据均以多级目录的方式进行管理,系统中的所有功能模块均以数据目录中的数据为源数据进行处理。

这样,开发人员在掌握了 MOTHER 系统之后,可以在该系统提供的环境中应用原型法来开发用户应用系统。开发过程大致可以分成以下四步:

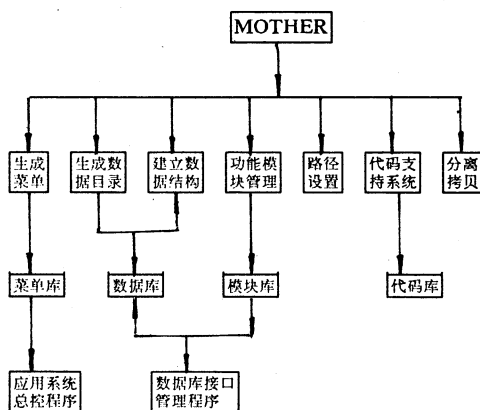
1. 确定用户需求;
2. 开发初始原型,即以 MOTHER 为交互应用系统,满足用户的基本需求;
3. 原型评价,即用户评价系统,提出修改意见;
4. 修改和完善原型,最终与 MOTHER 系统分离;利用 MOTHER 系统生成应用系统的过程可用图

一表示。



图一

**二、系统结构** MOTHER 系统是在汉字 DBASE3-PLUS 上开发成功的,系统结构如图二所示。



图二

MOTHER 系统的操作处理对象有四种,即菜单库,数据库,模块库和代码库,它通过自身的管理模块控制着这四类“数据”。MOTHER 系统的功能可以分成二种,即生成功能和管理功能。下面分别对各功能加以简单介绍。

生成功能有:

1. “生成菜单”,允许用户在对应用系统进行了系统分析的基础上根据应用系统的要求建立树型结构的多级菜单、菜单树的叶子取值于模块库。
2. “生成数据目录”,帮助用户建立应用系统的数据目录,目录结构为树型结构。
3. “建立数据模型”,建立应用系统的数据库结构,用户输入格式,数据库横向运算关系以及数据库索引。

所建立的数据库均按用户的要求登记到指定的数据目录。

4. “路径设置”,可以对应用系统中的每一个功能模块设置一条通向数据库的访问路径,否则,访问都将从根目录开始。

5. “分离拷贝”,该程序用于将应用系统从 MOTHER 系统中分离出来,生成用户应用系统软盘。

管理功能有:

1. “应用系统总控程序”,此程序通过对菜单库的管理,实现用户对自己的应用系统的控制使用。

2. “数据库接口管理程序”,此程序使得应用系统功能模块使用数据目录中的数据库成为可能。

3. “功能模块管理程序”,它是开发者维护本系统的一个接口,它为不断完善增强 MOTHER 系统的功能提供了可能。只要在此程序进行登记,新的功能模块便能加入到 MOTHER 系统的模块库中。模块库中现有数据编辑,数据显示,数据查询(可实现多库查询),统计计算,数据输出(可输出卡片,简表,报表,报告),数据合格性检验,数据库备份,数据库运算(库的分割与合并)。这些功能模块是应用系统进行事务处理的基本保证。它们以数据目录中的数据库为处理对象,可以通过

路径设置专为某子目录服务。

代码支持系统是一个代码辅助输入系统。用户可根据应用系统的具体要求选择是否使用此代码系统。它包括五个功能:

1). “建立代码库”。对输入重复量大的数据项及要求规范的数据项建立代码库。

2). “编辑代码库”。全屏幕编辑代码信息。

3). “链接代码库”。将任一代码库链接到任一数据库的任一数据项,这样在输入数据库时,遇到链接了代码库的数据项,代码信息就会自动显示在屏幕下端,用户键入相应的数码,对应的信息就会自动装入数据库。

4). “撤销代码库”。将指定代码库与指定数据项的链中断。

5). “删除代码库”。将指定代码库从代码库目录中删除并撤消此代码库与所有数据项之间的链。

三、结束语 MOTHER 系统是在对 MIS 探讨的基础上的一个初步实践,研制过程中曾对几个应用系统进行过生成,得到了较满意的结果,但是我们的工作还存在着不少问题,比如,模块库中用户可使用的功能模块的进一步通用,灵活,丰富等。恳请各位提出宝贵的意见。

## 袖珍机面临的问题及其对策

SHARP PC-1500 系列袖珍机涌入我国已历经 8 个春秋,其数量约 40 万台左右,加上 CE-158 接口和 515P 绘图机,耗资约十亿元,比国内 8 千台大中型机和 30 万台微型机的总和还多,在我国计算机推广应用中扮演着重要角色。

然而,袖珍机当前也面临着以下问题:1. 最令人头痛的是我们自已不会仿造其零部件,给维修工作带来难题,往往是拆此补彼,因此,必须大力开展袖珍机的保养维护工作的宣传;2. 有的人追求“洋、大、全”,喜新厌旧,有了高档机,不用低档机,使袖珍机遭受冷遇,利

用率较低,因此,为了发挥其久远深长的经济效益,应积极进行袖珍机的多层次和深层次开发。

鉴于以上两点,笔者特提两点对策:①应主张机器第一,软件第二,开发软件的前提是机器完好无损,因硬件是“体”,软件是“魂”,“体”之不存,“魂”将焉附?机器坏了,软件再好,也势必“魂不附体”,无济于事;②积极开展袖珍机的零部件仿制工作。国内应有科研单位,大专院校或组装工厂勇敢地承担此项任务,使袖珍机继续发挥“余热”,焕发青春,再使用 8 个春秋。

(北京 曹来发)

### 会议消息

为庆祝中国中文信息学会成立十周年和“中文信息通联世界”征文活动的胜利,《中文信息》杂志社等单位将于今年 6 月 26 日至 30 日在成都市人民南路 4 段 11 号四川省科协科技活动中心举办《中文信息通联世界学术会》和第二届中文电脑全国交流会,欢迎各界各业踊跃参加此次“硬件软件大汇展,用户厂家大交流”的天府盛会!

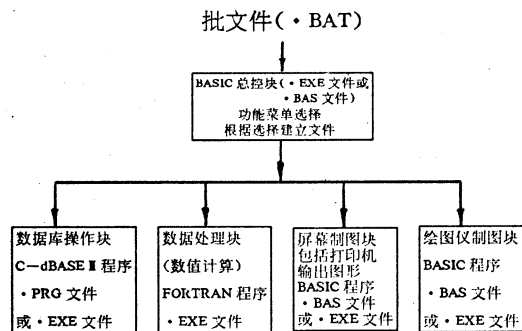


# 试论多种程序语言的综合应用

兰州大学 钟廷姣

**摘要:**本文阐述了微机系统多种程序设计语言综合应用的必要性及可能性,同时还提供了一种多语言系统的接口技术。

**一、多种程序语言综合应用的必要性** 使用多种程序设计语言在微型计算机上开发一个大而复杂的软件系统是一件十分有意义的工作。因为人们已经发现单用某一种语言或某一种系统软件来开发应用程序时往往不方便,其功能受到许多限制。例如,广泛流行的 C—dBASE II 虽有较强的数据管理能力,但分析、计算能力较弱,数组运算、图形、音响等功能也没有;而且,统计运算的速度也很慢。因此,只用 DBASE 语言编制程序,只能满足局部的功能,缺乏更高层次的应用。又如,任一种高级程序语言,无论是 FORTRAN、PASCAL、BASIC,……,也无论是解释型或是编译型,虽然各自都有较强的科学计算能力,有的还有较强的会话能力、图形及音响能力,但却没有很好的数据管理能力。当然,任一种程序设计语言不可能面面俱到,尽善尽美。也就是说,程序语言的多(功能全)、快(运算速度快,解释或编译速度快)、好(程序优化性好、代码质量高)、省(占用空间小)仅仅是相对标准。实践证明,多语言系统可充分利用微机的资源(硬件资源和软件资源),与单语言系统相比,多语言系统不但功能强,完善,而且易于修改、扩充;易组装和调试;稳定性好,覆盖技术得到充分利用;容错性强等优点,下图就是一个多语言系统的简要框图。通过它,可初步领略到多种程序语言综合应用的优越性。



图一 简要框图

上面框图所揭示的多语言系统中,利用了 DBASE

II 管理数据,组织数据的能力;FORTRAN 的快速计算能力;BASIC 的会话能力,图形及音响能力等。各功能块的程序可以是编译型的可执行程序,也可以是解释型的源程序,或是二者兼有,根据具体问题而选择。编译型虽然运行速度快,但占用内存较多,解释型则正好相反,所以,在设计多语言系统时,用户应根据问题的大小、难易程度及资源的配置情况等灵活地选择、组织自己的系统。

大家知道,一个系统通常由数据输入、数据值计算、分析比较,结果输出及系统的调度控制等组成,随着计算机在数值计算、非数值计算(如各类名目繁多的事务管理等)的日趋普及,用户对软件功能的要求也越来越高,而在目前很难找到一种程序设计语言同时兼有下述功能:

1. 具有较强的数据管理能力:数据输入、修改、追加、删除、存贮、检索等。2. 能够进行点对点地通讯;3. 易于各类数值计算:数组运算、复杂的算术运算、大量的函数运算,高精度运算及分析比较判断等;4. 图形功能:屏幕制图、绘图仪制图、打印机制图及表格输出等。5. 音响功能。6. 人一机会话功能等。

多语言系统的设计是建立在结构化设计的基础上,结构化设计是把一个大系统分解成具有层次结构的若干模块,然后自顶向下逐步细化各模块,使之完成某一特定功能。计算机程序语言的选用通常由细化后各模块的功能决定。

下表是常用程序语言的主要功能比较。

功 能 语 言	数据 管理	数据 通讯	数据 处理	屏幕 制图	屏幕数 据显示	绘图仪 制图	音响
C—dBASE II	强	无	弱	无	强	无	无
FORTRAN	弱	无	强	无	弱	无	无
BASIC	弱	强	较强	强	强	强	强
PASCAL	弱	强	强	强	强		较强

当然,程序语言的选择是多种多样的,但考虑各种语言的特点,比较各自功能,并根据模块设计要求而选择语言所用的方法应是一致的。实践证明,只要总体结构设计合理,各功能模块划分恰当,用户仍可感觉到整

个多语言系统是由一种语言编写。这是因为采用了灵巧的接口技术,使得系统在调度控制,各种状态的转换(如 DOS 状态、DBASE 状态或其它的高级语言状态等)完全是自动的,不需要人工干预。

**二、接口设计** 多语言系统的接口设计除了借助于盘文件外,本文还提供一种在批处理程序中采用“虚拟文件”式的接口技巧,这是作者在多年的研究及工作实践中总结出来的简便接口,它是一种行之有效的实用方法,该方法可将任意语言的程序文件串连起来。前面已经特别强调,这些程序既可以是编译型的可执行文件;也可以是解释型的源文件;还可以是批处理文件。根据总控制块的菜单选择,用户可随意进行分支转移。也许大家已经注意到前面介绍的框图内,在 BASIC 总控块中有“根据菜单选择建立文件”一句话,这是一句非常重要的话,它实际上就是指出接口设计,比如:选择功能(一),则程序立即建立“ONE”号文件,随后立即返回到 DOS 系统,进入批处理程序断口的下一条执行,实现选择转移。这种软接口技术,不同于一般的批处理文件。大家知道,常用的批处理程序是将各个文件名排列起来,并依次执行,一旦这排列次序确定后,就不能根据用户需要而任意挑选程序执行。而本文提到的“虚拟文件”技巧,就大大地增强了批文件的能力,弥补了原来的不足,它可根据用户的选择而随意进行与分支转移。

下面,我们来看一看两种类型的批处理程序。

#### 一、一般常用形式:

```
1. AUTOEXEC. TXT
ECHO OFF
FILEL
CCCC
ECHO ON
2. ZZ. BAT
ECHO OFF
CLS
BASIACA A;PAM
DBASE A;JISO
ECHO ON
```

#### 二、采用“虚拟文件”式的批处理

```
LDXT. BAT
ECHO OFF
CLS
;LOOK
BASICA ZK
IF EXIST ONE GOTO LI
IF EXIST TWO GOTO L2
IF EXIST THREE GOTO L3
```

```
IF EXIST FOUR GOTO L4
IF EXIST FIVE GOTO L5
;LI
ERASE ONE
DBASE KCZ
GOTO LOOK
;L2
ERASE TWO
FXXT
GOTO LOOK
;L3
ERASE THREE
HXXT
GOTO LOOK
;L4
ERASE FOUR
GOTO LOOK
;L5
ERASE FIVE
C;
```

在 LDXT. BAT 文件中,ZK 是菜单式的总控制块文件名,是解释型的 BASIC 源程序文件,ONE、TWO、THREE、FOUR、FIVE 均是文件名,是在 ZK 中根据功能菜单的选择而建立的,是无任何内容的“空文件”,且文件名的存在也只是在一个瞬间,从上面的程序中可以看出,虽然在 ZK 中根据选择建立了相应的文件名,尔后,控制马上就进入了相应的分支,并立即消去了对应的文件名。所以,当系统运行结束时磁盘上根本就不存在这些文件名。笔者将这种按口技术取名为“虚拟文件”式的接口技术,甚至可以说是一种巧妙地接口技术。

在 LDXT. BAT 中,KCZ 是 C—dBASE 解释型的源文件,即 KCZ. PRG,是数据库操作子系统。TUXT 是 BASIC 源文件,即 TUXT. BAS,是绘图子系统,FXXT、HXXT 均是批文件,由它们可进入别纵向分析计算子系统及横向分析子系统。

从上面的批处理程序可看出,若干串连起来的文件名,可以是解释型的源文件,也可以是第二层的批文件。当然,还可以是编译型的可执行文件。分支转移完全由总控块的菜单选择决定。

肯定无疑的是,采用“虚拟文件”接口技术的多语言系统中,不同语言编写的程序模块之间的调用速度快于操作系统提示符下启动各语言程序。毫无疑问,多促程序语言的综合应用将把应用软件的开发提高到一个新的水平,达到更高层次的应用,更有效地利用微机系统的软硬件资源。读者不妨一试。

## 通用文本文件显示程序

武汉 59190 部队自动化站 王晓武

熟悉 DOS 的 TYPE 命令的用户,可能都有同样的体会,即 TYPE 命令无法正常显示用 WordStar 编辑软件建立的文书文件。这是因为 TYPE 命令是按标准的 DOS 文本文件格式设计的,而用 WordStar 建立的文本文件,具有一些 TYPE 命令不能处理的控制码(如:换行、分页、打印字型等)。另外,DOS 的 TYPE 命令不能分页显示,而且,在比较高档的微机上(如,286,386 机)显示文本文件、速度太快,用户很难看清。综上所述,DOS 的 TYPE 命令是不能满足用户要求的。

参见 TYPEPLUS. ASM 程序清单,以下为程序说明:

(1)本程序通过系统功能调用 21H 来访问磁盘文件。它首先在数据段定义一个区域作为文件控制块(FCB),并定义一个 DTA 字节用以接收从磁盘文件中读出的信息。程序首先将程序前缀区(PSP)的文件名和扩展名送到 FCB 中相应位置,然后,打开指定的文件并设置缓冲区(使用 INT 21H, AH=1AH),并将 FCB 中当前块置为 0,记录长度置为 1,当前记录号置为 0。

(2)由于记录长度为 1,程序每次仅接收一个字符(或半个汉字)。字符 0AH 表示一行的结束,字符为 CTRL-2(1AH)表示整个文件结束。

(3)程序每读一记录,系统会自动对 FCB 中当前块号,当前记录号等信息进行修改以达到顺序读出文件中各个记录的目的。字符读出后,程序要进行比较判

断读出的字符是否为 1AH,09H 以及是否 Wordstar 的(换行、分页、打印字型等)控制字符,根据这些判断转相应的子程序进行处理。

(4)正常(或已屏蔽了 WS 控制字符的字符)字符读出后即显示在屏幕上,显示行数计数达到 9(或 24)时,此时已满一屏,等待用户打一键,若键入 ESC 时中途退出,若键入 PgUP 返回第一页重新显示,按其它键继续下页显示,当碰到 1AH 关闭文件(INT 21H AH=10H),返回到 DOS 状态。

(5)程序在显示字符时,子程序中寄存器 CX 的值为循环次数,控制字符显示速度,一般按阅读文件的速度显示,286 的机器以 0C000H 为宜,XT 机则以 09000H 更合适。

(6)程序在系统功能调用进行文件操作后,将检查出口参数(AL),以确定是否有输入输出错,如果有错则显示出错信息,然后返回到 DOS 状态。

(6)TYPEPLUS 程序的用途不仅体现在微机的文本文件显示方面,尤其在微机接上大屏幕显示设备时,利用该程序显示文字信息能起到净化屏幕的作用,达到影视字幕的显示效果。(需程序清单者,可同作者或本刊联系)。

作者工作单位:总后基地指挥部自动化站

通讯地址:武汉市汉口永清路 20 号

邮 码:430010

### 一个小型显示程序

在 dBASE-III 中,显示可用命令 KISP、LIST 和其他方法实现,但缺少灵活、随意性,有时总感到不尽人意。此处向读者提供一例小型显示程序。此程序可根据数据库字段随意修改,屏幕格式可随意设计、显示速度随意控制、显示前可加说明语句、显示可暂停,修改扩充余地大。程序清单如下:(注意编号是为说明附加)

```
1 use aaa
2 go bott
3 zz=recno()
4 go top
5 clear
6 ? "请输入显示控制数:"
7 input "N(10-80)" to pt
8 pt=pt+50
9 clear
10 x=1
11 ? "县计委文件登记情况"
12 y=1
13 do while x<=zz
14 loca for 编号=x
15 ? "
```

```
16 ?? 编号
17 ?? "——"
18 ?? 分类
19 ?? " "
20 ?? 文名
21 ?? " "
22 ?? 密级
23 z=pt
24 do while z>u
25 z=z-1
26 enddo
27 y=y+2
28 x=x+1
29 ? " "
30 if y=3
31 pt=pt-50
32 endif
33 enddo
34 use
```

(贵州三都县计委信息中心 柯昌献)

# 内存驻留程序的通用设计环境

广州中山大学计算中心 肖俊良

**摘要:**本文所介绍的程序为用户提供了一个设计任何适合自己用途的后台任务和热情激活型的弹出(POP-UP)任务模块的环境,它们均可以无害地任意使用 DOS 功能调用,当不再需要时还可以从内存解除已经驻留内存的代码。

TSB(Terminate-and-Stay-Resident)是在 IBM PC 上非常流行的软件开发方法之一。使用这种方法开发的软件有:Sidekick, Superkey, Pctools 以及 Lightning 等。它们均使用了类似的方法来提高产品的竞争力。但用户在无法得到这些软件源程序的情况下,难以借用这些方法,少数资料虽有零星介绍,但均结合自己的实际例子进行,并没有为用户提供一个内存驻留程序的通用设计环境。本文的内容填补了这个空缺。

本程序可以做到:

1. 一个热键激活 POP-UP 任务;一个开关停止或继续后台任务。
2. POP-UP 任务只能在文本方式下才能激活,因为它仅保护了用户寄存器。结束前将自动恢复所保存的内容。
3. 可以确定是否已经安装,并可以防止重复安装。
4. 可以防止内存驻留代码的重入。
5. 可以使用任何 DOS 功能调用,消除了 DOS 不可重入问题所引起的任何副作用。
6. 当用户不再需要时,可以从内存中解除驻留代码并释放所占用的内存。

以下为本程序的流程:

Flow map of General Design environment of TSR program

Start;

Display heading;

Call INITIALIZATION;

IF INITIALIZAION=TRUE then

tell the user of suer of successful install;

terminate and stay resident

{use DOS function call 31h}

Else

tell the user of failure;

exit.

INITIALIZATION;

If already installed then

return FALSE.

Set display buffer based on Video mode;

Reroutine int 08h;

Reroutine int 09h;

Reroutine int 13h;

Reroutine int 21h;

Reroutine int 28h;

Locate the INDOS flag;

Store interrupt vector value;

Set BUSY to FALSE;

Return TRUE.

INT08;

{int 08h handler}

Disable interrupts;

Call original DOS INT 08H handler;

If BUSY=FALSE and IN-BIOS=FALSE  
and INDOS=0 then

BEGIN

set BUSY=TRUE;

save caller's stack;

set stack to local stack;

call PROCESS;

restore caller's stack;

set BUSY=FALSE;

END.

Restore interrupts;

Exit.

INT09;

{int 09h handler}

Disable interrupts;

Call original DOS INT 09H handler;

If BUSY=FALSE and IN-BIOS=FALES  
and INDOS=0 and HOTKEY

pressed and current video mode is test mode then

BEGIN

set BUTY=TRUE;

save caller's stack;



```

set stack to local stack;
save current video display;
call PROCESS;
restore previous video display;
restore caller's stack;
set BUSY=FALSE;
END.

Restore interrupts;
Exit.

INT13;
{int 13h handler}
Set IN—BIOS to TRUE;
Call original DOS INT 13H handler;
Set IN—BIOS to FALSE;
Exit.

INT21;
{int 21h handler}
If AH=1,7,8 then
repeat
call int 16h function 01 to see character is ready;
until ketperssed;
jump to original DOS int 21h handler.
Else
if AH=0Ah then
call int 16h function 01 to see character is ready;
if chaeacter is ready then
call DOS function call to read character and process
character;
until (ending condition).
Else
if AH=0Ch then
clear DOS buffer and call other function.
Else
jump to original INT 21H handler.
Exit.

INT28.
{int 28h handler}
If BUSY = FALSE and IN — BIOS = FALSE and
HOTKEY pressed then
BEGIN
restore the video;
restore original interrupt vector;
deallocate this program memory;
exit.

```

Else

jump to original DOS INT 28H handler.

PROCESS;

Turn the cursor off; {only POP—UP interface}

call INT 65H; {POP—UP interface} or

call INT 66H; {BACKGROUND interface}

Turn the cursor on; {only POP—UP interface}

Exit.

一个 TSR 程序在返回 DOS 之前,必须做以下几件事:

1. 检查内存是否已经安装了一个拷贝,如果是,则立即退出安装过程;

2. 确定显示适配器的类型;

3. 拦截所需要的中断向量,并且重新设置为新的中断处理程序。例如,拦截键盘中断处理程序用于测试是否输入激活热键;

4. 确定 indos 标记。这是一个未写入文档的 DOS 功能调用,用来确定是否可以中断 MS—DOS 的工作。

5. 使用 31H 号功能调用使程序驻留在内存并返回 MS—DOS。

我们碰到的第一个问题在于确定内存中是否已经安装了一个拷贝,而 MS—DOS 并未提供一个程序是否已经驻留在内存的功能调用。不能简单地查看用于激活 TSR 之中断向量所指向的程序代码来确定,因为其他 TSR 也可能在后面又取代了这些向量,而另一些 TSR 程序则检查规定的中断向量中存放的特殊代码。

本程序通过 DOS 已分配的内存块来跟踪、搜索特别的识别标记。DOS 在已分配的内存块前有一个长度为 16 字节的内存控制块 MCB,其中只有前 5 个字是有用的:

第 0 字节 若是内存控制块链的最后一块,则为 ASCII 90,否则为 ASCII 77;

第 1—2 字节 若该储块已被撤消分配则为 0,否则指向程序的 PSP;

第 3—4 字节 该存储块的大小(16 字节的整数倍)。

从 MCB 的信息可以求出下一个存储块的段地址。

未写入文档的 52h 功能调用会取回 DOS 内部变量的地址,其中 es:[bx-2]存放的是 DOS 的第一个 MCB 所在的地址。

当活动驻留代码时,本程序会保存一部分显示内存区(当前活动页),以使用户可以在显示器上显示信息。本程序拦截了多少个中断,用户可以使用功能调用 25h 获得规定的中断向量,35h 设置规定的中断向量。

本程序文件拦截了五个中断: int 08h, 运行后台任务; int 09h, 运行 POP-UP 任务; int 13h, 避免前、后台及 POP-UP 任务同时访问不可共享的磁盘设备, 因为一个时刻只能有一个进程使用磁盘, 并且要求同步工作, 因此当 ROM BIOS 进行磁盘服务时绝不要去激活 TSR; int 21h, 用于 DOS 提示符下执行后台任务和 POP-UP 任务模块。DOS 提示符下等待命令时, 使用 OAH 功能调用, 此时 indos 标记不为零。为了在 DOS 提示符下能够执行后台任务和 POP-UP 任务, 作者重新设计了 DOS 功能调用 1, 6, 7, 8, OAH, OCH, 使得应用软件使用这些功能调用时 indos 标记等于零。不使用 DOS 空闲中断 int 28h 执行 POP-UP 及后台任务的原因是, 当用户使用 1-OCH 功能调用是会破坏 DOS 堆栈, 即使 1-OCH 功能调用句柄式功能调用加以模拟, 亦难以保证与 PCTOOLS、SK 软件联用时不发生意外; int 28h, 用于在 DOS 提示符下解除内存驻留代码。

一个未写入文档的 DOS 功能调用 34h 会取回称为 indos 的标记的地址, 这个计数器表明当前 DOS 代码的活动次数。若 indos 大于零, 表明已经进入 DOS 代码一次以上。MS-DOS 代码是不可重入的, 这意味着不可在任意点处中断后使 MS-DOS 代码为另一个进程重新使用。

最后, 将 BUSY 标记设置为 FALSE, 表示 TSR 未活动(int 08h、int 09h 在准备活动驻留代码时将检查这个标记, 以防止驻留代码的重入)。然后使用 MS-DOS 功能调用使程序驻留内存。

int 09h 将检查用户是否按下 Alt shift 激活热键, 不同的 TSR 应该使用不同的激活热键。然后检查 BUSY 状态, 只有为 FALSE 才能激活 TSR。作为最后的检查, 必须确定当前不是图形显示模式, 因为图形模式需要大量的显示内存, 而本程序却只能保存一页的文本显示内容, 一旦确定可以激活驻留代码, 将保护堆栈段寄存器和指针之值, 并设立内部堆栈区, 这可以避免测试是否有足够的堆栈区可供使用, 在跳出驻留代码前将恢复原堆栈区, 最后, 将保护显示内存的内容和

当前光标位置, 在执行完规定的中断服务程序后将予以恢复。

int 08h 用于运行后台任务, 执行规定的另一个中断服务程序, 并且不再保护显示区的内容和当前光标位置。

int 13h 进入前设立磁盘活动标记, 退出前清除磁盘活动标记。请注意, int 08h、int 09h、int 28h 都检查此标记, 只有磁盘无活动时才能激活驻留代码。MS-DOS 的键盘输入调用 01h、06h、07h、08h、0Ah、0Ch 由 int21 模块模拟, 当应用程序使用这些功能调用时, 通过适当的编程把 indos 标记转化为零。

你也许想解除内存驻留代码, 释放驻留代码占用的存储空间, 以便提供给其他程序使用。int 28h 查看所有的键击, 当敲 L shift-R shift, 即解除热键、无磁盘操作、BUSY 为 FALSE 时执行 DEINSTALL 函数。

DEINSTALL 首先恢复所有的中断向量, 然后释放驻留程序本身、环境块副本, 这要使用 MS-DOS 的 49h 功能调用即可。

如果以后又运行了其他 TSR 程序, 那么将同时被解除。

EXTEND. ASM 是使用本程序提供的任务接口显示系统当前日期和时间的例子。其中系统当前日期和时间的获得使用了 DOS 功能调用。

目前, 使用本软件应注意:

1. 要求 MS-DOS 2.0 或更高, 最少 10K 自由内存。
2. IBM PC 系列及其兼容机。
3. 必须是最后一个运行的 TSR 程序, 虽然常用的 TSR 软件, 如 SK、SUPERKEY、DOUBLEDOS 4.0 均可在本程序后正确运行, 但 PCTOOLS 的驻留功能除外, 因此建议用户遵守本规则。
4. 只有使用软开关 Alt shift 关闭后台任务, 才能使用软开关 L shift-R shift 解除内存驻留代码。
5. PCTOOLS 的驻留功能、SK 在本程序之后运行时, 将显示 SK 已经驻留的信息, 这并非错误, 请参考第 3, 4 点。需此软件者, 可同作者联系。

### 好消息

《1990 年全国计算机应用研究学术交流会议论文集》现已出版, 共录入 158 篇具有新颖性、实用性、通用性、学术性的学术论文, 全书采用电脑激光照排, 读者对象为从事计算机应用开发的广大科技人员, 并具有良好的参考及收藏价值, 每册 29.5 元, 需要者, 请同本刊读者服务部联系。印数有限, 欲购从速。

开户行: 成都工商银行跳伞塔分理处, 帐号: 89501299。

户名: 四川省电子计算机应用研究中心; 邮编: 610015。

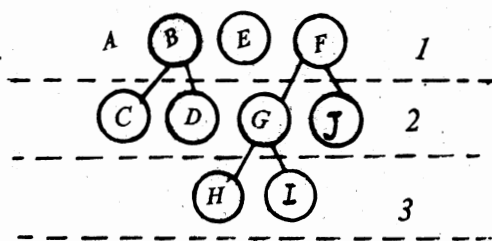
# 通用报表自动打印程序的设计与实现

四川省电子计算机应用研究中心 赖小强

**摘要:**目前 MIS 在向更加高层的方向发展,对 MIS 也就越来越要求方便、灵活,本文针对 MIS 有关表格通用自动打印作一些分析、介绍,在《成都市环境监测信息管理系统》中得到实现,收到了理想的效果。

**一、引言** 在信息管理中,大量的数据大都采用报表的形式表达,因此开发 MIS,其中报表形式的输出占居了相当的比重,一般的方法是采用编写固定的一一对应的报表打印程序与数据统计计算连接的方法实现打印输出的。这种方法,系统开发周期长,工作量大,占用较多的磁盘空间且不利于系统的扩充和维护等。随着报表的不同及报表量的增加,形式又经常变化等,以上方法就显得有些力不从心。针对这一普遍存在的问题,本人设计了一种通用报表自动打印程序,它的最大优点是占用较少磁盘空间,节省编程工作量,方便、灵活,可完成各种报表的自动打印。

**二、设计思想** 表格的打印一般包括:表头、表体的产生,数据库和数据项的确定,最后输出打印这样一个过程。其中表体的产生是表格打印过程中的重要环节。表体实现了通用自动生成,其余的表头、数据库和数据项则是从变量(字段)和数据文件中读取数据顺序打印即可,是不难实现的。因此表体的自动打印则是本文要谈的主要内容。通过分析发现表体数据项名称之间是一种树结构。全部表体数据项的结构实际上是由多个树组成的森林构成。显然表体每个数据项名称与森林中各个树结点有一一对应的关系。由于打印顺序是按层数由小到大,从左到右打印,因此我们在考虑打印顺序时必须在树结构的基础上产生一些中间数据,以符合按打印顺序打印出的表格是我们事先设计的形式。用数据文件形式描述树结点是不困难的,只要实现了对这些树结点的访问,就可以产生出中间数据。通过分析发现,可采用树的前序遍历法实现表体的各数据项的访问并产生出中间数据。下面用图示的方法加以说明。



这是由 4 棵树组成的表体。其中:最大层数是 3;最小层数是 1;叶结点是 A、C、D、E、H、I、J。在描述这些结点作输入时,是按从上到下先左后右的顺序原则。即按 A、B、C、D、E、F、G、H、I、J 顺序描述。如上图先从左边树第一层根结点开始,如果没有第二层结点则输第二颗树的第一层根结点 B(如果有第二层则输左结点)。现 B 第二层有结点按从左到右的顺序输入 C、D(如有第三层,则先 C 后输第三层的左结点),然后 E 照此下去……,完成全部结点输入。

对输入结点的描述包括:①结点文字内容;②文字个数;③该结点有几个分枝;④该结点在第几层;⑤结点文字横写或竖写;⑥竖写时每次横写的文字个数。

完成了以上参数的输入后,通过程序产生中间数据,计算出行、列宽度,基本算法框图从略(可详见本刊 90 年论文集)。中间数据产生后,接着就是产生表体。为了提高程序的灵活性,产生出的表头表体以变量的形式存放在 \*.EME 文件中,这样不仅节省空间,而且大大方便制表用户。表格生成好后,只需每次使用各自表格所对应的 \*.EME 文件,打印输出即可。

**三、通用制表打印程序的实现** 前面已经提到,表体产生后是以变量的形式放在 \*.EME 文件中的,\*.EME 文件中包括三部分内容:一、表头、附表头,二、表符串(即 +, -, ……等),三、表体中的文字内容(如数据项名称等),这三部分却是一一对应的,就是说某层表符串对应某层的文字串。程序通过产生的

A	B		E	F		
	C	D		G		J
				H	I	

中间数据从上到下来判定生成那一种表符。(是“┐”或“└”或“┌”)。首先确定表符,然后第一层的文字串也就随之而产生。困难的是产生中间层的表符,它不仅与同层中间数据有关而且与上层的数据有关。通过分析发现上层文字是通过上层数据已经生成了的。那么上层文字的形式已经反映出上层数据对这层数据项的描述。因此我们只要将上层文字的形式进行一次扫描即可确定当前层应产生什么样的表符与上层文字相匹配,(实现方法详见产生中间层源程序)。

**四、结束语** 以上介绍的通用报表自动打印程序是将事先设计好的表格形式用数据描述,按照一定的方法输入到数据文件中,程序自动生成并与统计数据文件连接直接打印输出。这种方法不需用户画表、连接、打印,也不需每次打印时重新生成。它速度快,使用方便灵活、易学、易维护。全部程序在长城 286-EX 机上采用 DBASE III 实现,但是要想形成一个完整系统,还必须进行进一步的完善,以适应不断发展的 MIS 的需要。

#### 产生中间层的源程序

```
Para man1, man2, man3, man4, man5, man6, bk2,
bk3, bk4, bk5, bk6, bk7, nx, c
set talk on
bk&c=" "
man&c=" "
hc=2
select 1
use trr
do while not .eot()
if nx4>val(c)
skip 1
LOOP
endif
if x6="*"
if hc=2
bk&c=bk&c+"┐"
else
hc=hc-2
fh=subs(bk&c, hc, 2)
hc=hc+2
if fn=" "
bk&c=bk&c+"┐"
else
bk&c=bk&s+"┐"
endif
```

```
endif
hhx=bx8
h=x7+1
hc=hc+2
nnx1="┐"+subs(nx1, h, hhx*2)
man&c=man&c+nnx1
repl x7 with x7+hhx*2
do while hhx>0
bk&c=dk&c+" "
hc=hc+2
hhx=hhx-1
enddo
if x7>=hx2*2
repl x6 with "≠"
else
skip 1
LOOP
endif
endif
if x6="≠"
skip 1
LOOP
endif
hhx=hx3
do case
case upper(x5)="S"
if hc=2
bk&c=bk&c+"┐"
hc=hc+2
h=1
else
c1=val(c)-1
c1=str(c1, 1)
hhx=bx8
fh=subs(man&c1, hc, 2)
if fh="┐"
hc=hc-2
fh1=subs(bk&c, hc, 2)
hc=hc+2
if fh1=" "
bk&c=bk&c+"┐"
hc=hc+2
else
bk&c=bk&c+"┐"
hc=hc+2
```



```

endif
else
bk&c=bk&c+" "
hc=hc+2
endif
h=x7+1
endif
hhx=bx8
nnx1="|"+subs(nx1,h,hhx*2)
repl x6 with " *"
man&c=man&c+nnx1
repl x7 with x7+hhx*2
do while hhx>0
bk&c=bk&c+"—"
hc=hc+2
hhx=hhx-1
enddo
if x7>=hx2*2
repl x6 with "≠"
else
skip 1
LOOP
endif
case upper(x5)="H"
if nx3=0
hhx=hx2
nnx1="|"+subs(nx1,1,hhx*2)
man&c=man&c+nnx1
else
if hx2>=hx3
? "errow"
else
kg=hx3-hx2
nnx1="|"+space(kg)+trim(nx1)+space(kg)
man&c=man&c+nnx1
endif
endif
if hc=2
bk&c=bk&c+"└"
hc=hc+2
else
c1=val(c)-1
c1=str(c1,1)
fh=srbs(man&c1,hc,2)
if fh="|"
hc=hc-2
fh1=subs(bk&c,hc,2)
hc=hc+2
if fh1=" "
bk&c=bk&c+"└"
hc=hc+2
endif
else
bk&c=bk&c+"┐"
hc=hc+2
endif
do while hhx>0
bk&c=bk&c+"—"
hc=hc+2
hhx=hhx-1
enddo
endcase
repl x6 with "≠"
skip 1
enddo
if len(bk&c)>3
go top
man&c=man&c+"|"
nx=nx+1
hc=hc-2
fh=subs(bk&c,hc,2)
hc=hc+2
if fh=space(2)
bk&c=bk&c+"|"
else
bk&c=bk&c+"└"
endif
hc=hc+2
hc=2
c=val(c)+1
c=str(c,1)
bk&c=" "
man&c=" "
else
close databases
nx=9
return
endif
enddo
return

```

## 快速修改汉字库

河北廊坊炮兵导弹学校 郭继展

中文 DOS 在设计汉字库时,为了减少存储空间,常删掉区位码(全称“通讯用汉字字符集(基本集)及其交换码国家标准 GB2312-80”,87 区 94 位)中的 10-15 区(全为空白);有的进一步先把 9 区(制表符)移至 6 区(舍弃 6 区的希腊字母),再删掉 8~9 区;有的更狠一点,删掉 56~87 区全部二级汉字。但也有的又在 10~12 区制造了上下标及斜体字符等。

这就出现一个问题:当用到一些符号时,本 DOS 字库删掉了;试换其他字库,出来的汉字竟“牛头不对马嘴”。如果用造字程序,一个一个地造一区字,实非易事,还可能打乱原字库。本文以 16×16 点阵字库为例(其原理完全适用于其它点阵),给出借助另一字库快速修改字库的方法和程序(见后)。

要修改一个字库,必须了解这个字库,可先——

### 一 显示 DOS 所含区位码字符

首先,我们可以计算出字库中删掉了多少个区。以修改 CCDOS3.0 为例,库 CCLIB(下称 B:L2)长 237632 字节,每个区 94 个字,每个字占 32 字节:

$$237632 / (94 * 32) = 79 \quad 87 - 79 = 8$$

可知删掉了 8 个区。显然,其中 6 个为 10~15 区。

程序中 80-110 语句,按输入区号给出启动所用 DOS 的该区位码字符。运行这段程序,可以知道被删掉的还有 8 区、9 区。实际上 9 区字符没有真正被删掉,它占了 6 区,CCCC.EXE 文件使 9 区与字库 6 区对应起来。

运行这段程序还会发现一个问题,输入区号 Q=6 或 9,显示的是相同的制表符,输入 Q=8 或 16,显示的全是 16 区汉字“啊……剥”,似乎有两区的点阵是重复的。实际上并非如此,这是 CCCC.EXE 文件中,为了提高找字速度未与理睬造成的,并不影响使用。

为了解字库的真实情况。可以——

### 二 按区或位连续显示字库点阵

先看一张全字库的盘紫金 DOS3.0,库 CCLIB(下称 A:L1)长 243648 字节,删掉了 10~15 空白区。我们可以把“A:L1”当作一个随机数据文件,记录长 32 字节。程序中的 120-200 语句,可以按区或位连续显示字库点阵。程序根据字库长度、区号和位号,计算出字的记录号读取记录,把记录分作 32 个字节,每个

字节再根据 8 个位上是 1 还是 0 分别显示亮块或空格(为了把字加宽成方块,皆为重复显示),形成一个字,同时在字的左面显示行号,上面显示列号。

如果显示“B:L2”,会看到外区(区位码表)与字库内区对应关系为:

外区	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17
	18 19 .....87
内区	1 2 3 4 5 7 8 9
	10 11 .....79

现在假定我们想恢复“B:L2”中第 6 区被删掉的希腊字母,同时又舍不得丢掉 9 区的制表符,这可以借助全字库“A:L1”——

### 三 替换某区全区字库点阵

这有两种方法:

#### 1. 仍保留字库“B:L1”大小

这样做的好处是:在一张 DOS 盘上还可以拷上 FORMAT、DISKCOPY 两个常用的外部命令,且可少占近 6K 内存。

方法是运行程序 210~290 语句,先读取“A:L1”和“B:L2”第 6 区中的第一个字,双双显示出来,用户认为无误之后,程序再将“A:L1”中第 6 区的全部记录写入“B:L2”第 6 区。再为 9 区选一个新的“殖民区”,譬如第 5 区(全是从未用过的字符),重复上述过程。

至此工作只是做了一半,还应修改 CCCC.EXE 文件中 10H 中断程序的 19 号功能块,使区码为 9(国标码为 29H,机内码为 A9H)时转换为 5,原程序为:

276C AND DX,7F7F 机内码在 DX 中,“与”掉高位得国标码

2770 PUSH AX

2771 CMP DH,29 是否 9 区区位码

2774 JNZ 2778

2776 MOV DH,26 是,改为 6 区

2778 CMP DH,30

277B JB 2780

277D SUB DH,08 8-15 区 8 个区无点阵  
将 2776 的“MOV DH,26”改为“MOV DH,25”即可。

#### 2. 不保留字库“B:L2”大小

用“A;L1”整个地替换“B;L2”(COPY 即可)。这样做的好处是:字库变成了全库,什么也不缺了。

这工作也只是做了一半,仍需修改上述模块。这时各区已无“侵占”,且增加了 8、9 两区,上述程序修改下述两处即可:

```
2771    JMP      2778
277D    SUB      DH,06
```

但有时我们并不需要整区、整库地显示或替换字库点阵,而只是想将库中的某一个或几个字点阵(如自造符号)显示一下或迁移过来,这就是——

#### 四 显示和替换单字点阵

运行 300—360 语句,程序根据用户输入的两组区位码,显示两个字的点阵以及区号、位号、字库内部对应的区号、位号和记录号,用户认可替换后,即将一库字点阵存入另一字库。如果用户只是想显示字点阵,当

10 REM 显示和修改字库点阵

20 KEY OFF;SCREEN 2,0;DEFINT A—Q,W;DEFSTR R—V;S=CHR\$(219);CLS

30 LOCATE 1,30;PRINT “程序运行说明”:

LOCATE 3,23;PRINT “1. 显示区位码字符(当前所用 DOS)”:

LOCATE 4,23;PRINT “2. 按区或位连续显示字库点阵”

40 LOCATE 5,23;PRINT “3. 替换某区全区字库点阵”:

LOCATE 6,23;PRINT “4. 显示和替换单字点阵”:

LOCATE 7,23;PRINT “5. 用点阵造字”

50 LOCATE 9,25;PRINT “请选择(1—5)”

60 INPUT I;IF I<>INT(I) OR I<1 OR I>5 THEN 50 ELSE PRINT

70 ON I GOTO 80,120,210,300,370

80 INPUT “输入区号 Q=”;Q;QK=2;REM CCDOS2.13 启动

90 GOSUB 500;IF QN=-1 THEN PRINT “该 DOS 无”;Q;“区”;GOTO 80;

ELSE PRINT “对应字库”;QN;“区”;REM QK=87—库长/(32\*94)

100 FOR I=1 TO 94;PRINT CHR\$(160+Q);CHR\$(160+I);;NEXT

110 PRINT;PRINT;GOTO 80

120 GOSUB 700;QK=QK1

130 INPUT “按区还是位显示字点阵(1、2)”;Q;IF Q=2 THEN 170

140 INPUT “输入区号 Q=”;Q;GOSUB 500;IF QN=-1 THEN PRINT Q;“区无字库!”;GOTO 140

150 G=94\*(QN-1)

160 FOR W=1 TO 94;G=G+1;GET 1,G;T=R1;GOSUB 590;NEXT;STOP

170 INPUT “输入位号 W=”;W;Q=0

180 FOR G=W TO 8178 STEP 94;GET 1,G;T=R1

190 Q=Q+1;GOSUB 500;IF QN=-1 THEN PRINT Q;“NO!”;GOTO 190

200 GOSUB 590;NEXT G;STOP

210 GOSUB 690

220 INPUT “输入字库 L1、L2 要修改的区号”;Q1, Q2

230 IF Q1=0 OR Q2=0 THEN CLOSE #1, #2;END

240 Q=Q1;QK=QK1;GOSUB 500;QN1=QN

250 Q=Q2;QK=QK2;GOSUB 500;QN2=QN

260 IF QN1=-1 OR QN2=-1 THEN PRINT “L1 或 L2 无此区点阵!”;GOTO 220

然回答“N”即可。

库中缺少的符号,如其他字库也找不到,用户可以自己——

#### 五 用点阵造字

在方格纸上,框出 16\*16 的方块来,将小方格涂黑色使呈字形,黑色部分记作 1,白色部分记作 0,每一行一分为二成两个字符串,这样共是 32 个字符串,将它们按由上至下、每行先左后右的顺序写入 490DATA 语句中(现在造的是一个空格),全 0 的字符串可用一个 0 表示。运行 370—490 语句,程序先显示指定区位处的字(一般应为空白),用户确认位置后,程序显示所造的点阵,用户满意后方存盘。

程序采用菜单式,一个主模块,五个子模块,六个子程序,如下:

```

270 G1=(QN1-1)*94+1;G2=(QN2-1)*94+1;  
W1=1;W2=1;PRINT "第一位上对应"; GOSUB 530  
280 IF B=0 THEN 220  
290 FOR I=2 TO 94;GET #1;T=R1;LSET R2=T;PUT #2;NEXT I;GOTO 220  
300 GOSUB 690  
310 INPUT "用 L1 的点阵替换 L2 的点阵,输入 Q1, W1,Q2,W2,=";Q1,W1,Q2,W2  
320 IF Q1=0 OR Q2=0 OR W1=0 OR W2=0 THEN CLOSE #1 ,#2;END  
330 Q=Q1;QK=QK1;GOSUB 500;QN1=QN  
340 Q2;QK==,QK2;GOSUB 500;QN2=QN  
350 IF QN1=-1 OR QN2=-1 THEN PRINT "L1 或 L2 无此区点阵!";GOTO 310  
360 G1=(QN1-1)*94+W1;G2=(QN2-1)*94+W2;GOSUB530;GOTO310  
370 GOSUB 700;QK=QK1  
380 INPUT "输入造字处的区位号 Q,W=";Q,W; GOSUB 500  
390 IF QN=-1 OR W<I OR W>94 THEN PRINT Q; "区无字库!";GOTO 380  
400 G=94*(QN-1)+W;GET #1,G;T=R1;GOSUB 720;GOSUB 590  
410 INPUT "Is it this word ? (Y/N)";V;PRINT;  
IF V<>"y"AND V<>"Y" THEN SCREEN 2,0;GOTO 380  
420 VV="";FOR I=1 TO 32;READ V0  
430 D=0;FOR J=1 TO 8;D=D+VAL(MID$(V0,J,1))*2^(8-J);NEXT J  
440 V0=CHR$(D);VV=VV+V0;NEXT I  
450 T=VV;GOSUB 600  
460 INPRT "This Picture is good,isn't? (y/n)";V  
470 IF V="y"OR V="Y" THEN LSET R1=T;PUT #1,G;CLOSE  
480 STOP  
490 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,  
00100000,00000100,  
00100000,00000100,00100000,00000100,00111111,11111100,0,0,0,0  
500 IF Q<=15-QK THEN QN=Q;COTO 520  
510 IF Q>=16 THEN QN=Q-QK;ELSE QN=-1  
520 RETURN  
530 PRINT TAB(13);"两字点阵如图:";GOSUB 720  
540 G=G1;GET #1,G;T=R1;GOSUB 720;Q=Q1; QN=QN1;W=W1;GOSUB 590  
550 G=G2;GET #2,G;T=R2;Q=Q2;QN=QN2;W=W2;GOSUB 590  
560 COSUB 720;SCREEN 2,0;INPUT "这样替换对吗(Y/N)";V  
570 IF V<>"y"AND V<>"Y" THEN B=0;RETURN  
580 R=R1;LSET R2=R;PUT #2,G2;B=1; RETURN  
590 SCREEN 0,0;PRINT "Q=";Q;"QN=";QN;"W="W;"G=";G  
600 PRINT TAB(25);"1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6"  
610 FOR L=1 TO 31 STEP 2  
620 PRINT TAB(23);USING"# #";L;;K=L; COSUB 650  
630 K=L+1;;GOSUB 650;PRINT  
640 NEXT L;PRINT;PRINT;RETURN  
650 N=ASC(MID$(T,K,1));M=128  
660 FOR J=1 TO 8;C=INT(N/M);N=N-M*C;M =M/2  
670 IF C=1 THEN PRINT S,S;;ELSE PRINT " ";  
680 NEXT J;RETURN  
690 OPEN "B:L2"AS #2 LEN=32;FIELD #2 ,32 AS R2;QK2=8;REM 237632 字节,79 区  
OPEN "A:L1"AS #1 LEN=32;FIELD #1,32 AS R1;QK1=6;REM 243648 字节,81 区  
710 RETURN  
720 FOR I=1 TO 1000 :I=I;NEXT I;RETURN

```



## 可在 PC-1500 机上运行的流程图绘制软件

镇江师范专科学校计算机室 杨大顺 陶明华

不论是在计算方法和程序设计语言的教学,还是在软件开发和计算机科学研究工作中,都经常需要绘制大量的流程图。为了减轻有关人员的手工劳动,提高流程图绘制的速度和质量。我们在国内拥有大量用户的 SHARPPC-1500 计算机和 CE-150 绘图仪上开发了流程图绘制软件。

### 一、流程图绘制软件运行环境和主要功能

本软件用 BASIC 语言编写,约占 3K 字节,可在配有 CE-150 绘图仪接口的 PC-1500 机上运行。

本软件可用于按用户的具体要求绘制流程图,其主要功能有:

1、根据用户绘制的流程图草图和具体要求打印出一定形状(按通常规定),适当大小和一定颜色(黑、兰、绿、红四种任选)的流程图框。

2、如果流程图中的信息全部是由英文字母和数码等构成,则可按用户的要求,在框中适当位置打印出所需信息。

### 二、使用说明

1、流程图绘制软件可以帮助用户通过人机对话,在计算机的绘图仪上绘制流程图。

2、在绘制流程图过程中所使用到的基本图框有 6 种。

3、为了使用程序方便,每一个基本框用 0~9, A~G 的一个代码表示。例如 O 和 A 分别表示起始框和结束框;2 和 C 分别表示带尾线处理框和不带尾线处理框;G 表示条件转向框等。

4、软件使用的方法如下:

(1)程序起动后,可根据流程图草图的实际图框的分布情况,输入该流程图的结构代码(FIG STR 图框代码命名。例如当整个流程图依次由起始框、输入框、条件选择框、处理框、输出框和结束框构成时其结构代码为 01321A)。

(2)根据程序运行过程中出现的提示信息,让用户通过手工分别输入(A)相应框的颜色(COLOR 黑—O、蓝—1、绿—2、红—3);(B)图框的行数(NUMBER OF LINE);(C)图框中需打印的英文信息(如果是中文则打入空格以便以后填写)。

(3)在计算机打印出的流程图上用手工加上适当

的程序流向线等(必要时还要填写适当中文信息),这就构成了一幅完整的流程图。

(4)由于 PC-1500 机绘图仪 CE-150 功能的限制,只能直接打出较简单的一行流程图。当流程图分支较多,由几行并列而成时,可以按上述步骤一行一行地打印出来,再拼接。为了保证各行之间相应图框能准确对位,在结构代码中特地设计了起空行的代码 6、7、8、9 分别对应于走 2、3、4、5 行空行的指令,用户可根据具体情况合理选择。

(5)本软件可绘制常见的解一元二次方程的流程图,它由两行拼接而成,一行的结构代码为 0123221A,另一行的结构代码为 676721,作者可提供流程图供用户参考。

### 三、程序清单

本文给出了整个流程图绘制软件的清单,读者只需根据清单,仔细地键入相应信息,校对无误后就可以启动该程序运行并用来绘制自己所需的流程图。为了以后使用的方便,最好通过 CE-150 接口,把该程序存贮在磁带上。由于程序并不很复杂,故说明从略。

```
50 Y=10;YY=20;X0=0;X1=80;X2=85;X3=90;
  X4=95;X=50;XX=56
100 S=-4;DIM X$(1)*30
110 INPUT"FIG STR=";X$(1)
120 NN=LEN X$(1)
130 FOR I=1 TO NN
135 Y$=MID$(X$(1),I,1);M=ASC(Y$)-47;
  Y1=20
136 IF M=24 THEN GOSUB 2000
140 IF M>10 LET M=M-17;Y1=0
150 ON M GOSUB 900,1100,1000,1200,1300,1400,
  1500,1500,1500,1500
160 NEXT I
170 END
900 INPUT"*BEGIN END* COLOR=";C
910 COLOR C
920 GRAPH;GLCURSOR(110,S);SORGN
930 LINE(-X,0)-(X,0)-(XX,-6)-(XX,
```

```

- 14)-(X,-YY),O,C
935 LINE (X,-YY)-(-X,-YY)-(-XX,-14)
    -(-XX,-6)-(-X,0),O,C
937 IF Y1=O THEN 945
940 LINE (O,-Y1)-(O,-2*Y1),O,C
943 LINE (Y1/8,-1.7*Y1)-(0,-2*Y1)-(-
Y1/ 8,-1.7*Y1),O,C
945 GLCURSOR (-110,-16);SORGN
950 TEXT
960 INPUT "$ <=8B";A$
970 N=LEN A$;T=9-INT(N/2)
980 LPRINT TAB T;A$
990 RETURN
1000 INPUT "* PROCESS* COLOR=";C;COLOR C
1010 INPUT "NUMBER OF LINE";M
1025 GRAPH;GLCURSOR (110,S);SORGN
1030 LINE (-X3,O)-(X3,O)-(X3,-YY*M)-
X3,-YY*M)-
    (-X3,O),O,C
1037 IF Y1=O THEN 1045
1040 LINE (O,-M*Y1)-(O,-(M+1)*Y1),O,C
1043 LINE (Y1/8,-M*Y1-.7*Y1)-(0,-(M+1)*
Y1)-(-Y1/8,-M*Y1-.7*Y1),O,C
1045 GLCURSOR (-110,-16);SORGN
1050 TEXT
1055 FOR J=1 TO M
1060 INPUT "$ <=14B";A$
1070 N=LEN A$;T=9-INT(N/2)
1080 LPRINT TAB T;A$
1085 NEXT J
1090 RETURN
1100 INPUT "* INP OUT* COLOR=";C;COLOR C
1105 INPUT "NUMBER OF LINE";M
1110 GRAPH;GLCURSOR (110,S);SORGN
1120 LINE (-X1,0)-(X3,0)-(X1,-YY*M)-(-X3,-
YY*M)-(-X1,0),O,C
1137 IF Y1=O THEN 1145
1140 LINE (O,-M*Y1)-(O,-(M+1)*Y1),O,C
1143 LINE (Y1/8,-M*Y1-.7*Y1)-(0,-(M+1)*
Y1)-(-Y1/8,-M*Y1-.7*Y1),O,C
1145 GLCURSOR (-110,-16);SORGN
1150 TEXT
1155 FOR J=1 TO M
1160 INPUT "$ <=14B";A$
1170 N=LEN A$;T=9-INT(N/2)
1180 LPRINT TAB T;A$
1185 NEXT J
1190 RETURN
1200 INPUT "* CHOICE* COLOR=";C;COLOR C
1205 M=3
1210 GRAPH;GLCURSOR (110,S);SORGN
1220 LINE (O,O)-(X3,-M*Y)-(0,-M*YY)-(-X3,-
M*Y)-(0,0),O,C
1237 IF Y1=O THEN 1245
1240 LINE (O,-M*Y1)-(0,-(M+1)*Y1),O,C
1243 LINE(Y1/8,-M*Y1-.7*Y1)-(0,-(M+1)*
Y1)-(-Y1/8,-M*Y1-.7*Y1),O,C
1245 GLCURSOR (-110,-16);SORGN
1250 TEXT
1255 FOR J=1 TO M
1260 IF J=2 INPUT "$ <=12B";A$;GOTO 1270
1265 INPUT " $ <=3B";A$
1270 N=LEN A$;T=9-INT(N/2)
1280 LPRINT TAB T;A$
1285 NEXT J
1290 RETURN
1300 INPUT "* LOOP* COLOR=";C;COLOR C
1305 INPUT "NUMBER OF LINE=";M
1310 GRAPH;GLCURSOR (110,S);SORGN
1320 LINE (-X2,0)-(X2,0)-(X4,-Y*M)-
(X2,-YY*M)-(-X2,-YY*M)-(-X4,-Y*
M)-(-X2,0),O,C
1337 IF Y1=O THEN 1345
1340 LINE (O,-M*Y1)-(O-(M+1)*Y1),O,C
1343 LINE (Y1/8,-M*Y1-.7*Y1)-(O,-(M+1)*
Y1)-(-Y1/8,-M*Y1-.7*Y1),O,C
1345 GLCURSOR (-110,-16);SORGN
1350 TEXT
1355 FOR J=1 TO M
1360 INPUT "$ <=14B";A$
1370 N=LEN A$;T=9-INT(N/2)
1380 LPRINT TAB T;A$
1385 NEXT J
1390 RETURN
1400 INPUT "* CONTINUE* COLOR=";C;COLOR
C
1410 GRAPH;GLCURSOR (110,S);SORGN
1430 LINE (-6,0)-(6,0)-(12,-6)-(12,-14)-(-6,-20),

```

```

0,C
1435 LINE (6,-20)-(-6,-20)-(-12,-14)-(-12,-6)-(-
6,0),0,C
1437 IF Y1=0 THEN 1445
1440 LINE (0,-Y1)-(0,-2 * Y1),0,C
1443 LINE (Y1/8,-1.7 * Y1)-(0,-2 * Y1)-(-Y1/8,-
1.7 * Y1),0,C
1445 GLCURSOR (-110,-16);SORGN
1450 TEXT
1460 INPUT "$=1B";A$
1480 LPRINT TAB 9 ;A$
1490 RETURN
1500 LPRINT
1510 M=M-6;WAIT 100
1515 PRINT "LF";-M
1520 FOR L=1 TO M
1530 LPRINT
1540 NEXT L
1550 RETURN
1590 RETURN
2000 INPUT "*ON GOTO(GOSUB) * COLOR=";C;
      COLOR C
2010 INPUT "NUMBER OF LINE";M
2020 INPUT "NUMBER OF BRANCH)2? (Y/N)";B$
2025 GRAPH ;GLCURSOR (108,S);SORGN
2030 LINE (-108, 0)- (108, 0)- (108,-YY * M)-(-
108,-YY * M)-(-108,0),0,C
2038 FOR XW=-80 TO 76 STEP 156
2042 LINE (XW,-M * Y1)-(XW,-(M+1) * Y1),0,C
2043 LINE (XW+Y1/8,-M * Y1-.7 * Y1)-(XW,-(M
+1) * Y1)-(XW-Y1/8,-M * Y1-.7 * Y1),0,C
2044 NEXT XW
2045 GLCURSOR (-108,-16);SORGN
2050 TEXT
2055 FOR J=1 TO M
2060 INPUT "$<=14B";A$
2070 N=LEN A$;T=9-INT (N/2)
2080 LPRINT TAB T;A$
2085 NEXT J
2087 GOSUB 2100
2090 RETURN
2100 GRAPH ;GLCURSOR (108,S);SORGN
2110 FOR XW=-80 TO 76 STEP 156
2130 LINE (XW-6, 0)-(XW + 6, 0)-(XW + 12,-6)-
(XW+12,-14)-(XW+6,-20),0,C
2135 LINE (XW + 6,-20)-(XW-6,-20)-(XW-12,-
14)-(XW-12,-6)-(XW-6,0),0,C
2140 NEXT XW
2145 GLCURSOR (-108,-16);SORGN
2150 TEXT
2160 INPUT "S$=1B";A$;LPRINT TAB 2;A$;
2165 IF B$(<)"Y" THEN 2180
2170 LPRINT TAB 4;".....";
2180 INPUT "E$=1B";A$;LPRINT TAB 15;A$
2190 RETURN

```

需此软件者,请同作者联系

## 介绍一种 CASL 调试软件——SCOMET

SCOMET 是一个集编辑、汇编、运行功能为一体的调试器。汇编时遇到错误,能自动转入编辑状态,并提供错误类型和出错点信息。运行中可以实行单步、多步及断点跟踪,可以随时显示模拟寄存器和内存单元的内容。另外,它也能把已汇编的程序反汇编为源程序。其内装编辑程序可编辑的最大文件为 64K。编辑操作方法与 WORDSTAR 基本上相同。上述所有操作都借助菜单和窗口进行。整个屏幕划分成编辑、运行和菜单提示三个窗口。编辑窗口可扩展至全屏。运行窗口分

为前景和背景两个相互覆盖的子窗口。前景显示寄存器与内存状态及反汇编结果,背景显示 CASL 程序中 I/O 指令的输入输出内容。所显示的及用户输入的数字可以是十进制或十六进制,由用户随时切换。运行窗口出现的地址总是同时显示为十进制和十六进制数。

SCOMET 的操作采用与 DEBUG 相似的单字符命令,但命令中的参数采用人机交互方式输入。

(山东大学 刘桓中)

需此软件者,请同作者联系。

# 介绍一例打印机绘图程序

天津大学水利系 别社安 汪克让

**摘要:**本文向读者介绍了一个打印机绘图程序,并介绍了程序的功能及其特点。

根据计算机绘图原理,笔者采用 FORTRAN 语言编制了一个较为完善的绘图程序库。利用该库,只需要了解库中各子程序所具有的功能,直接对其调用即可编出所需要的各种绘图程序,当然,也可根据需要对该程序库补充完善。

下面,列表简述程序库中部分子程序的功能。

## 一、打印机绘图程序库

编号	子程序名	功能	编号	子程序名	功能
1	SETPLT	设置绘图区域	22	PCURXY	画函数 $x=f_x(t), y=f_y(t)$ 的曲线。
2	SETBLK	设置绘图子区域及绘图比例等。	23	TERMXY	画多项式 $x=f_x(t)y=f_y(t)$ 的曲线。
3	SETSCR	设置显示器的绘图状态。	24	CIRCLE	过三点作圆。
4	SETPRN	设置打印机绘图状态。	25	PARAB	过三点作抛物线。
5	PRNDOT	打印机绘图画点,具有绘实线点、虚线点、点画线点以及抹去点的功能。	26	ELLIPS	画圆、椭圆、弧线
6	PRNPLT	将绘制的图形存盘和在打印机上打印。	27	RECT	画矩形。
7	PAUSEP	绘图暂停。	28	GRID	绘矩形格子。
8	PDOT	在显示器上画点。	29	PLOY	绘正多边形。
9	PLINE	在显示器上画直线。	30	BAR	绘条形图。
10	MDOT	在绘图坐标系中画点。	31	AROWHD	绘箭头。
11	MLINE	在绘图坐标系中画直线。	32	AXIS	绘坐标轴。
12	MRECT	在绘图坐标系中画矩形。	33	DOTPL	将一系列的点用各种符号画出。
13	MNUM	在绘图坐标系中写整数。	34	CURPL	将一系列的点用光滑曲线连接。
14	CHANGE	绘图坐标系和实际坐标系之间的坐标转换。	35	SATFPL	将一系列的点用各种函数进行曲线拟合。
15	DOT	画点。	36	PLANE	绘制各种平面、空间网格图。
16	LINE	画直线。	37	ROTTATE	坐标旋转变换。
17	NUMBER	写实数。	38	DOTR	在旋转坐标系内画点。
18	PCURX	画函数 $f(x)$ 的曲线。	39	LINER	在旋转坐标系内画直线。
19	PCURY	画函数 $f(y)$ 的曲线。	40	NUMBR	在旋转坐标系内写数。
20	PTERMX	画多项式 $f(x)$ 的曲线。	41	TRANS	空间坐标和平面坐标之间转换。
21	PTERMY	画多项式 $f(y)$ 的曲线。	42	DOTS	在空间坐标系中画点。
			43	LINES	在空间坐标系画直线。
			44	NUMBS	在空间坐标系中写数。

图 1 是调用绘图库绘图的一个例题。

**二、应用实例** 该程序是调用上述绘图库,用 FORTRAN 语言编写的一个数据处理实用图形程序。该程序适用于各种微机,可进行广泛的数据处理及图

形绘制。绘制的图形可存盘,可在打印机上输出,也可在显示器上显示。

### 1. 程序功能

(1)将所给的离散数据用各种符号标出,或用各种



光滑曲线连接,或进行各种曲线拟合。

绘点的符号有:‘.’,‘+’,‘×’,‘□’,‘△’

曲线连接方式有:直线连接和光滑曲线连接。

拟合的曲线有:指数函数  $y=ae^{bx}$ , 幂函数  $y=ax^b$ ,

多项式函数  $y=\sum_{i=0}^k a_i x^i$  等。

(2)可进行一些简单的绘图、制表等。该程序调用库中的功能子程序(15—44),可绘制各种图形。

## 2. 程序特点

(1)绘制的图形可在 24 针点阵打印机上打出,可存盘多次打印,对于 IBM 系列微机及其兼容机,图形还可在显示器上显示。(2)图形的大小、位置、数量、性质可随意选择,并进行叠加。(3)图形处理时,可自动绘出坐标轴并标注刻度值。(4)绘图数据输入简单、方便。

3. 示例 (1)将实验数据用多项式曲线拟合,如图 2 所示。(2)绘制有限元单元网格图,如图 3 所示。

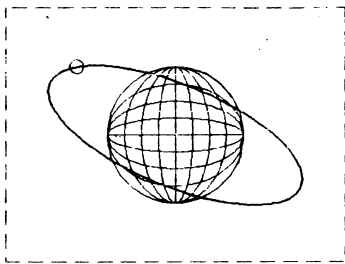


图 1

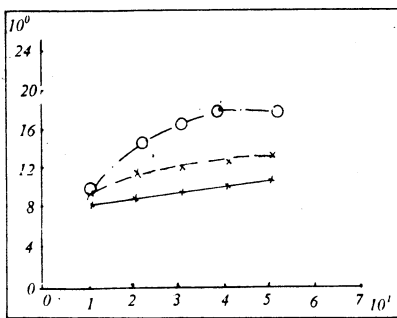


图 2

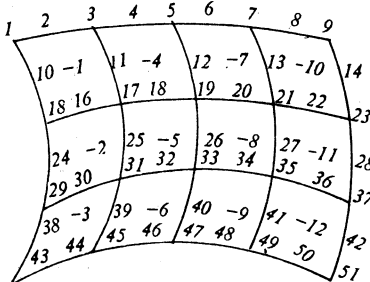


图 3

C>DEBUG

-A100

```

0100 MOV     DX,005B
0103 CS;
0104 MOV     [0057],DX
0108 MOV     AX,CS
010A CS;
010B MOV     [0059],AX
010E PUSH    BX
010F MOV     AH,OF (读当前显示状态)
0111 INT     10
0113 POP     BX
0114 CS;
0115 MOV     WORD PIR [0055],B800
        (彩色显示缓冲区段地址)
011B MOV     DX,03DA (CRT 状态地址)
011E MOV 有 AL,[BX]取参数 1 存图 2 取图
0120 CMP     AL,32
0122 JZ      0135      实用小程序
0124 CMP     AL,31      程序功能:直接存
0126 JNZ     0152      取屏幕图形。
0128 CS;              软件环境: IBMPC
0129 LDS     SI,[0053]  及其兼容机。
012D CS;              具体调用方法:(见
012E LES     DI,[0057]  上页文末)程序清
0132 JMP     031F      单
0134 NOP
0135 CS;
0136 LDS     SI,[0057]
013B LES     DI,[0053]
013F MOV     CX,2000
0142 CLD
0143 IN      AL,DX
0144 TEST    AL,01
0146 JNZ     0134
0148 CLI
0149 IN      AL,DX
014A TEST    AL,01
014C JZ      0149
014E MOVSW  (字传送)
0150 LOOP    0143
0152 REIF
-RXCX
;53
-NSR. BIN
-W

```

# 菜单工具软件 MENU TOOLS 的设计与实现

中国科学院光电技术研究所 王道顺 孙国良

**摘要:**本文介绍了 IBM PC 系列微机上使用的菜单工具软件 MENU TOOLS 的设计思想,实现软件功能的一系列方法。

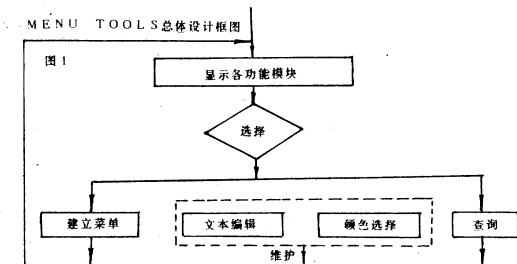
**一、前言** 菜单,作为人机交互技术之一,现已成为应用软件乃至系统软件中必不可少的一部分;好的菜单应当是设计简单,使用方便,给用户带来生动感,有人称它为软件的“外包装”。

目前,国内编制的软件中,菜单大多采用普通菜单,主菜单与子菜单的区别不够明显,采用数字或英文字母编号进行选项,用户使用不便,速度不够理想。没有充分开发彩色功能,菜单画面不够醒目,更为麻烦的是:对开发软件人员来说,每编一套应用软件时,就需要重新设计并编写一次菜单,浪费精力浪费时间。为了尽量简化应用软件在进行这种“外包装”时的烦琐手续,我们开发了本菜单工具软件 MENU TOOLS。它适用于各种微机及与之配套的彩色适配器。

**二、MENU TOOLS 的设计思想** 菜单,作为用户友好接口的必备工具,在程序设计时可随时增添、删除和修改菜单显示项,菜单画面的颜色也要随用户的喜欢作不同的匹配。

由于系统软件和应用软件通常具有复杂性及计算机内存限制,采用模块化的程序设计方法来进行设计 MENU TOOLS。

菜单软件作为人机交互技术及其管理工具,我们主要从菜单建立、维护、查询和生成的菜单与用户软件的接口这几个方面来进行设计,其总体设计框图见图一:



图一: MENU TOOLS 总体设计框图

下面对这些功能模块一一讨论:

**1、菜单建立** 菜单是用户进行人机交互过程中生成的,由用户输入文件名, MENU TOOLS 自动建立用户所需要的二级下拉式菜单执行文件(·EXE),它由两部分内容决定完整的屏幕菜单显示:菜单的文本文件和菜单画面的颜色数据文件。

## 2、维护

• 菜单项的增、删 随着应用软件不同,需要对菜单上显示的选项作相应改变, MENU TOOLS 可直接利用全屏编辑软件(wordstar)建立显示的菜单文本文件。具体实现,用户利用(↑)键或(↓)键移动醒目行,在菜单文件编辑处(MENU file Editor)处按下[Enter]键,即可对菜单文件进行增、删。

• 菜单框架的缩、放 菜单框架及显示的菜单选项与菜单屏幕成一定比例,可使画面美观,在编辑方式下,增加、抹除空行,可改变菜单框架纵向大小,同样增加、抹除空列数,菜单框架的横向大小作相应改变。

• 菜单画面的颜色选取 菜单作为系统软件或应用软件的“外包装”,配有一定的色彩,可使菜单画面产生用户所希望的那种生动感,颜色选取功能模块为用户提供了选取菜单屏幕的边缘色,菜单画面的背景色,前景色,而且在颜色选取时,菜单选项部分的画面随光标所指的颜色成动态同步变化,操作方便,并且有八种颜色供选取。

其中,菜单画面的背景色包括:边框、中心、主菜单、子菜单和显示研制单位;前景色由边框、主菜单、子菜单、研制单位和光标行(醒目行)组成。

**3、查询** 用户常常需要对建立的菜单进行查看,以便对菜单进行维护, MENU TOOLS 提供了这一功能。实现它,用户只要在查询处按下[Enter]键即可,看到建立的主菜单,再利用(↑)或(↓)键和[Enter]键看到各个子菜单项。

## 4、建立的菜单与应用软件的链接

建立菜单的目的是提供用户以菜单方式去驱动系

统。MENU TOOLS 建立的菜单提供了菜单执行功能,用户按一下[Enter]键,菜单利用提供的批处理文件即可执行外部模块,执行完以后再返回主菜单画面,供用户再选择其它项。

除了上述几个主要方面外,我们利用翻页方法,在建立的菜单中增加了求助信息显示,用户在系统执行过程中,只要按一下<F1>键,可显示出软件的一些求助信息。并且求助信息也是通过全屏幕编辑软件进行编辑。

**三、软件功能实现的一些方法** 完成上述设计思想,除了使用 Microsoft C5.0 提供的函数外,还需要对屏幕直接控制,由于实际的菜单程序可移植性强,但访问屏幕的程序都与硬件及操作系统存在内在联系,因而必须避免调用 C 函数中普通控制台 I/O 函数,通过对视频适配器访问编写一系列函数,并采用 IBM 兼容的 ROM-BIOS 作为它的操作系统,之所以选用 DOS-BIOS 是因为它使用广泛,而且源码可移植性强。

下面,我们具体介绍一些功能实现的方法

**1、访问视频适配器** MENU TOOLS 对屏幕进行直接控制,因此有必要对适配器的构成进行了解。

最常见的有下列几种适配器:单色适配器、彩色/图形适配器(CGA)以及增强型图形适配器(EGA)。其中,CGA 和 EGA 具有几种操作方式,包括 40 列或 80 列文本操作,以及图形操作。在本文中我们采用 80 列文本方式。

一般来说,访问视频适配器有三种方式。一是通过 DOS 调用,但它速度太慢;其次通过 ROM-BIOS 调用,速度较快;三是直接读写视频存储器,很快但较复杂,而且它降低源码的可移植性。

在软件编写中,采用第二种方式访问视频。

**2、MENU TOOLS 中颜色选取与传递** 由于 IBM PC 系列微机上采用各种显示卡,显示彩色的方法就因此有不同的编码,为了避免因显示卡的原因,降低程序的通用性,采用 PC 机中固有的图形字符显示色作 MENU TOOLS 中颜色值。

- 选取:将选项行数与光标行数结合起来,将光标所指的颜色值写进一个数据文件中。

- 传递:建立的菜单通过指针定位函数对选取的颜色数据文件进行读取。

**3、实现 MENU TOOLS 画面的一些主要功能函数**

- text()写彩色字符串

利用 ROM-BIOS 中断 10,功能 9,再加上 C 提供的 int86()函数可以很方便实现。

```
/*      Display a string      */
#include <dos.h>
union REGS regs;
text (attrib, row_y, column_x, string)
int attrib, row_y, column_x;
char * string;
{
    int i;
    regs.x.cx = 1;
    regs.h.dh = row_y;
    regs.h.bh = 0;
    for (i = 0; string[i] != '\0' && string[i]
        != '\n'; i++) {
        regs.h.ah = 0x02; /* Set cursor position */
        regs.h.dl = column_x + i;
        int86(0x10, &regs, &regs);
        regs.h.ah = 0x09; /* Write a character */
        regs.h.al = string[i];
        regs.h.bl = attrib;
        int86(0x10, &regs, &regs);
    }
    regs.h.ah = 0x02; /* Turn cursor off */
    regs.h.dh = 25;
    regs.h.dl = 0;
    int86(0x10, &regs, &regs);
}

• Colorbar()在指定区域着色
利用中断 10,第 6 号功能实现。

/*      Display a color block */
#include <dos.h>
union REGS regs;
colorbar (attrib, row_y1, column_x1, row_y2,
column_x2)
int attrib, row_y1, column_x1, row_y2, column_x2;
{
    regs.h.ah = 0x06;
    regs.h.al = 0x00;
    regs.h.bh = attrib;
    regs.h.ch = row_y1;
    regs.h.cl = column_x1;
    regs.h.dh = row_y2;
    regs.h.dl = column_x2;
    int86(0x10, &regs, &regs);
}

• frame()画彩色边框。
```

这个子程序用到 256 个 ASC II 字符串几个特殊的图形字符,利用它可以画一个双线矩形框。

下面我们给出这些图形字符的十六进制码。

十六进制码	字符
C9	左上拐角符号(┐)
BB	右上拐角符号(┌)
C8	左下拐角符号(└)
BC	右下拐角符号(┘)
CD	水平线(=)
BA	垂直线(∥)

程序附后。

·屏幕边缘色

适配器为中分辨

适配器为高分辨

```
#include <dos.h>
union REGS regs;
plat(attrib)
int attrib;
{
    regs.h.ah=0x10;
    regs.h.al=1;
    regs.h.bh=attrib;
    int86(0x10, &regs, &regs);}

#include <dos.h>
union REGS regs;
plat1(attrib)
int attrib;
{
    regs.h.ah=0x0b;
    regs.h.bh=0x00;
    regs.h.bl=attrib;
    int86(0x10, &regs, &regs);}
```

4、菜单显示 将欲显示的菜单文本文件从磁盘中读出,存放在一个二维数组中,这个数组很方便地访问菜单文件中各行,利用 text(),可以显示出很好的正文菜单,采用这种方法,正是设计思想中菜单维护功能模块。

5、保存屏幕与恢复屏幕 在内存中申请了二块各 4000 字节的存贮区用来存放显示的主菜单、子菜单,系统作出选择后,将显示的菜单保存,在适当时候恢复。在文本方式下,C 提供的 Movedate()可完成这一功能,具体实现:当激活标志为 0 时,一直保存显示的菜单,否则就恢复。

6、建立的菜单与应用软件链接 在应用软件中,菜单起着主控程序的作用,它和各功能模块,采用“菜单分枝技术”,对每个分枝直接利用 C 语言中 System()函数去驱动各个功能模块。

另外,输入用户响应问题。采用光标输入法,即利用(↑)或(↓)键移动醒目行,直到所需选项。然后按下<Enter>键即可。其中,醒目行是以反色出现。

四、结束语 MENU TOOLS 达到了设计思想,为用户的应用软件的“外包装”提供了一种工具,很有实用性。希望本文能起到抛砖引玉的作用。最后对施卫美本人及他提供的菜单程序表示感谢。

#### 参考文献

- 1、李智渊:《C 语言》,成都电讯工程学院出版社,1988
- 2、鸿健编译:C 语言高级程序员编程指南,中科院希望高级电脑技术公司
- 3、Microsoft C5.0 库程序参考手册(6,7),中科院科海培训中心,1989

#### 程序清单

```
/* Display a frame */
#include <dos.h>
union REGS regs;
textframe (attrib, row _y1, column _x1, row _y2,
column _x2)
int attrib, row _y1, column _x1, row _y2, column _x2;
{
    int length=column _x2-column _x1-1, width=
row _y2-row _y1-1,i;
    regs.h.ch=0;
    regs.h.cl=1;
    regs.h.ah=0x02;
    regs.h.cl=length;
    regs.h.dh=row _y1;
    regs.h.dl=column _x1;
    regs.h.bh=0;
    int86(0x10, &regs, &regs);
    regs.h.ah=0x09;
    regs.h.al=0xc9;
    regs.h.bh=0;
    regs.h.bl=attrib;
    int86(0x10, &regs, &regs);
    regs.h.ah=0x02;
    regs.h.dl=column _x2;
    regs.h.bh=0;
    regs.h.ah=0x90;
    regs.h.dl=0xcd;
    regs.h.bh=0;
    regs.h.bl=attrib;
    int86(0x10, &regs, &regs);
    regs.h.ah=0x02;
    regs.h.dh=row _y2;
    regs.h.bh=0;
    int86(0x10, &regs, &regs);
    regs.h.ah=0x09;
    regs.h.al=0xcd;
    regs.h.bh=0;
```

```

int86(0x10, &regs,      regs. h. bl=attrib;      &regs);
&regs);                  int86(0x10,&regs,      }
                           &regs);                  }
regs. h. ah=0x09;
regs. h. al=0xbb;        regs. h. ch=0;
regs. h. bh=0;           regs. h. cl=1;
regs. h. bl=attrib;
int86(0x10, &regs,      for(i=1;i<width;
&regs);                  i++){
regs. h. ah=0x02;        regs. h. ah=0x02;
regs. h. dh=row_y2;      regs. h. dh=row_y1+i;
regs. h. bh=0;           regs. h. dl=column_x2;
int86(0x10, &regs, &regs)regs. h. bh=0;
                           int86(0x10, &regs,
                           &regs);
regs. h. ah=0x09;        regs. h. ah=0x09;
regs. h. al=0xbc;        regs. h. al=0xba;
regs. h. bh=0;           regs. h. bh=0;
regs. h. bl=attrib;      regs. h. bl=attrib;
int86(0x10, &regs, &regs)int86(0x10, &regs,
                           &regs);
regs. h. ah=0x02;        regs. h. ah=0x02;
regs. h. dl=column_x1;   regs. h. dl=column_x1;
regs. h. bh=0;           regs. h. bh=0;
int86(0x10, &regs, &regs)int86(0x10, &regs,
                           &regs);
regs. h. ah=0x09;        regs. h. ah=0x09;
regs. h. al=0xc8;        regs. h. al=0xba;
regs. h. bh=0;           regs. h. bh=0;
regs. h. bl=attrib;      regs. h. bl=attrib;
int86(0x10, &regs, &regs)int86(0x10, &regs,

```

### 长江 CHC 系列河道测绘系统研制成功

国家“七五”重点科技攻关项目——长江 CHC 系列河道测绘系统由水利部长江水利委员会水文局研制成功,不久前在南京通过水利部的鉴定,使江河湖泊水道测绘实现自动化和现代化。

该系统共有 2、3、4 三种型号,全系统由测量仪器,计算机(LT3400 便携机)、绘图仪、回声测深仪、对讲机,以及接口电路和数传机等部分组成,分岸台和船台两部分,测绘时岸台和船台两部分配合进行,用计算机控制,能实时采集、传输、存储测绘数据、自动绘图,在屏幕上展绘航迹显示水深,具有精度高、功能多、速度快、自动化程度高、减轻劳强度、节约物力等优点,效率比原老设备提高数十倍,能对江、河、湖泊、近海沿岸、水库和潮汐河段的水下地形,岸上地形进行测绘,还可测绘测量人员不可能到达的淤泥、沼泽、悬崖、陡壁、崩岸测点地区。经对武汉、沙市、南京、马鞍山、崇明大片江段数百个测点试测数万个数据表明,该系统的各次指标达到设计要求,成本仅是进口货每套近 30 万美元的 1/15。该系统的研制成功,进一步提高了我国江河湖泊水道测量的现代化水平,对有效开发利用长江具有重要意义。(李相彬)

```

(上接 39 页) 760 IF R(1)="b" OR R(1)="B"
THEN BIANY=-1 ;GOTO 710
770 IF R(1)="#" THEN BIANY=1 ;GOTO 710
780 IF R(1)=")" AND R(1+1)="-" THEN RST=
TYP ;PRINT #2,"ML";
790 IF K=1 OR FST=1 THEN K=0 ;FST=0 ;PRINT
#2,"L";USING "###";M(NUM);
800 PRINT #2,TYP;
810 IF BIANY=-1 THEN PRINT #2,"-"; ELSE IF
BIANY=1 THEN PRINT #2,"+";
820 BIANY=0
830 IF GPT=1 THEN GPT=0 ;PRINT #2,".";
840 RETURN
900 I=I+1
910 IF R(1)="-" THEN H=H+1 ;GOTO 900

```

```

920 IF R(1)="/" THEN PRINT #2,"ML"; ;GOTO
900
930 IF R(1)="\" THEN PRINT #2,"MN"; ;GOTO
900
940 IF R(1)="b" OR R(1)="B" THEN BIANY=-1 ;
GOTO 900
950 IF R(1)="#" THEN BIANY=1 ;GOTO 900
960 IF 4 MOD H=0 THEN L=4/H ELSE L=4\H + 1
;GPT=1
970 PRINT #2,"L";USING"###";L; ;H=2
980 RETURN
1000 PRINT #2,CHR$(34)
1010 PRINT #2,HANG; "PLAY" ;CHR$(34); ;
HANG=HANG+10
1030 RETURN

```

## dBASE III 实现下拉式菜单的方法

安徽省安庆石油化工总厂炼油厂 姜振斌

**一、问题的提出** 随着我国微型计算机的日益增多, dBASE III 新型关系数据库管理系统以它在社会各个领域的广泛应用而受到广大用户的青睐。目前大多数用户都是通过菜单来驱动系统的, 由于受到功能的限制, 使用的菜单只能是传统的方法, 即由用户按菜单提示的序号进行选择。显得非常麻烦, 且难免给人以呆板的感觉。自从 FoxBASE+ 出现后, 下拉式菜单受到了欢迎。用各种语言编制的下拉式菜单相继出现, 那么在 dBASE III 上有没有实现下拉式菜单的可能呢? 笔者经过一段时间的研究, 使这个答案得到了肯定, 并且在已开发的系统中加以应用获得了成功。

**二、实现方式** dBASE III 本身没有编制下拉式菜单的命令, 因此要实现这个过程必须具备以下条件:

1> 按要求对菜单中的某一条进行光栅显示且仅仅是唯一的。

2> 用上<↑>、下<↓>等键选择菜单时光栅要随之上下移动。按回车键后程序接受的命令应该是被选定的某条菜单所执行的功能。

根据以上两点, 笔者选用了 dBASE III 设置屏幕颜色“SET Colo To”命令, 和在 Compiler<编译 dBASE III>中有有效的函数 INKEY() 来完成任务。

大家知道, “SET Colo To”命令可设置字符颜色还可设置底色。例本文后附的程序实例中的“SET Colo To 4,,2+”和“SET Colo To 4/1,,2-”两条命令混合使用就能达到对某一菜单进行光栅显示。

至于 INKEY() 函数, 表示击键, <按任一键>返回的是一ASC II 码。例如按“↓”键, 返回的是“24”, 按回车键返回的是“13”。这样根据返回值的不同选择菜单的某一条进行光栅显示, 达到上下移动的目的。

**三、程序实例** 该程序是笔者编制的一系统主程序的一部份, 由于该系统目录比较简单故采用内存变量 CDML 存放, 变量“MLS”代表菜单目录个数、“MS”代表被选定要置光栅的目录。程序首先设置的“4”。

程序的 9—22 行是显示菜单, 并对选定的某条进行光栅显示。23—26 行按任意键, 实际上程序执行到

这里处于等待状态。27—41 行是对选择做出反应并且规定: 当光栅在第一条目录再按“↑”键时光栅移到最后一条菜单, 同样, 当光栅在最后一条菜单再按“↓”键时, 光栅移到第一条菜单, 按回车键后程序跳出 7—42 行循环体执行别的操作。为了不至于出现光标干扰, 程序设置第 8 行消除光标, 并在结束时恢复<本系统在 2、13D 中文系统上编制>。

由于笔者水平有限, 难免有很多缺点, 欢迎广大同行批评指正。

```
1: * * dBASE III 下拉式菜单 *
2: SET TALK OFF
3: CLEA
4: CDML="系统主菜单
(1)录入数据
(2)数据处理
(3)查询打印
(4)退出操作"
5: DO WHIL. T.
6: STOR 4 TO MLS, MS
7: DO WHIL. T.
8: ? CHR(14)+"119,01"
9: XI=1
10: GB=4
11: SET COLO TO 4,,2+
12: @2,28 SAY SUBS(CDML,1,10)
13: DO WHIL. NOT. XI>MLS
14: IF XI=MS
15: SET COLO TO 4/1,,2+
16: ELSE
17: SET COLO TO 4,,2+
18: ENDI
19: @GB, 25 SAY SUBS(CDML,11+
(XI-1)*15,15)
20: GB=GB+1
21: XI=XI+1
22: ENDD (下转 37 页)
```



# 反弹加密

新疆第12陆军医院 何其才

**摘要:**本文利用文件在磁盘中的存放形式,通过对文件目录项的处理,实现文件的加密,经加密的文件他人无法印、阅、使用和复制。

**一、设计思想** 我们先了解一下文件在磁盘上的存放形式,DOS在格式化磁盘时,逻辑上把整个DOS区域划分成了以512字节为单位的连续扇区,无论软盘或硬盘,它们的逻辑0扇区为引导扇区。映象如附图所示。

DOS下的文件是以目录项和相关数据的形式存放在磁盘上,一个目录项为32字节,连续地存放在目录区(子目录在数据区),而文件的内容存放在数据区。这样只要我们对文件的目录项作部分处理就可达到预期的目的。

目录项的第26—27字节存放该文件的开始簇号,第12—21字节为保留,如果我们利用部分保留字节存放开始簇号,然后把26—27字节的内容改为FF7(坏簇标识)。这样就达到了加密的目的。在解密时只要把保存在保留字节中的内容弹出给26—27字节,然后送0到相应的保留字节,解密完成。如果在加密时把文件的开始簇号进行逻辑判断和相应的算术处理,然后保存,判断的结果再占用保留中的一个字节。在解密时根据判断结果作相应的逆处理,然后弹回。我们只要做到对局部用户,用不同的处理方法,就可以使每个用户得到不同的锁。这样就是有些用户有解密程序或知道开始簇号保存在保留字节中,也不敢轻易地解密或把其作为开始簇号处理,否则可能会造成损失。

**二、实现方法** 在做这项工作之前主要是在根目录或子目录搜索要加密文件的目录项且将其读到指定的内存,然后对其进行加解密处理就不难实现了,前者可根据附图中的BPB和FAT的内容来实现。本算法的编码由汇编语言分六步完成:(1)取命令行参数或

从键盘读入;(2)将参数中的小写字母转为大写字母;(3)对参数进行处理;(4)取磁盘参数;(5)在磁盘中搜索加解密文件的目录项;(6)加解密处理。具体的步骤不再赘述。另外在加密时一定要对所用保留字节的内容进行判断是否为0,解密时对开始簇号判断是否为FF7。否则就不能进行相应的工作。

**三、使用说明** 当我们的加密实现后,再利用批处理写三条命令:(1)对某文件解密;(2)执行某文件;(3)对其加密。这样每次运行时,只要打入批处理名即可。本方法已在IBM-PC机,长城0520C-H机和长城28EX机上实现。此方法理论上对DOS2.0以上版本管理下的所有文件均能加锁。如果我们要对批处理和解密文件本身保密,我们用修改后的DOS外部命令CHMOD对加密和解密文件的属性进行设置,在运行前置为档案,运行后置为隐藏,这样我们的批处理就变为7条命令。批处理本身的保密我想可借助于命令处理程序COMMAND.COM来实现。如果把加解密和批处理文件作为钥匙保存的话就不存在以上的处理。

00H	E9XXXX 或 EBXX90	BPB
03H	OEM名和版本(8字节)	
0BH	每一扇区的字节数(2字节)	
0DH	每一分配单位的字节数(2字节)	
0EH	保留扇区,始于0(2字节)	
10H	FAT的个数(1字节)	
11H	根目录中的目录项数(2字节)	
13H	逻辑卷中总扇数(2字节)	
15H	介质描述字节	
16H	每一FAT的扇区数(2字节)	
18H	第一道的扇区数(2字节)	
1AH	磁头数	
1CH	隐含字节数	
1EH	引导程序	

(附图)

# 让计算机来演奏你喜爱的乐曲

重庆大学计算机系 潘作秋

微型计算机能演奏音乐,然而计算机只能识别“机器曲谱”,即必须用一种特定的语言记谱。对于简谱,计算机是无法识别的,这让我们想随心所欲地演奏自己喜爱的乐曲是很不方便的。如果人工地把简谱翻译成机器曲谱,那又太费力了。为此,笔者设计了一个能完成这个“翻译”功能的程序。

由于简谱是由多行组成的,而目前的绝大多数微型计算机只能一行一行地识别,因此,要对用简谱写的曲谱略作修改一下,把多行的“移”到一行上来,产生“类简谱”作为中间媒质。对简谱改写成类简谱的原则是:

1、为了增强可读性,应尽量与简谱保持相似。与简谱中相同的有:音符 0、1、2、……、7,等同短线“—”,节拍间隔线“|”,以及用以表示延长音的点“.”。

2、速度快慢的约定:快用 K,慢用 M。

3、升半音号“#”、降半音号“b”原写不变,但须写在所升降字符的左侧,如: #5。

4、连拍组合用的下横线“—”,用圆括号“(”、“)”来代替,如:  $\underline{35}$  写成 (35),  $\underline{665}$  写成 (6(65))。注意圆括号必须配对,即“(”的数目必须等于“)”的数目。

5、升 8 度号“.”用“^”,降 8 度号下圆点“.”用“v”,这两个符号都写在所升降字符的左侧,例:“2”写

成“^2”,“7”写成“v7”。

6、连音用的弧连线“^”用斜杠号及反斜杠号“/”、“\”来代替。例:  $\widehat{31}$  写成 /31\;  $\widehat{5|51}$  写成 /5|51\。注意:

①斜杠号与反斜杠号也必须配对。

②若弧连线后是等同短线,则要把等同短线都括进去。如:  $\widehat{5|5}---$  写成 /5|5---\。

③若弧连线从等同短线开始,则要把等同短线前面的音符也括进去。如海玻 --  $\widehat{|2}---$  写成 /2---|2---\。

7、选择段落连线用花括号“{”、“}”。例  $\underline{3152-|5-|3151-|1-|}$  写成 {3(15)2-|5……}|3(15)1-|1……。

8、重复段落号“||:”、“: ||”分别用方括号“[”、“]”表示。注意:①“[”与“]”可不配对。②“[”与“{”相遇时,“}”写在“]”的前面。

9、结尾必须附加上一个特殊字符,如“@”以示结束。其它符号都省略不写,如:调门。

按这些原则编写出来的类简谱,与真正简谱很相似,且又简单明了,容易修改。初次使用时,可能觉得别扭,但用过几次之后,就会得心应手了。

(上接 35 页)

23: JIAN=0

24: DO WHIL JIAN=0

25: JIAN=INKEY()

26: ENDD

27: DO CASE

28: CASE JIAN=13

29: SET COLO TO 4,,2+

30: EXIT

31: CASE JIAN=5

32: MS=MS-1

33: IF MS<=0

34: MS=MLS

35: ENDI

36: CASE JIAN=24

37: MS=MS+1

38: IF MS> MLS

39: MS=1

40: ENDI

41: ENDC

42: ENDD

43: IF MS=4

44: EXIT

45: ELSE

46: II=STR(MS,1)

47: DO JB&II

48: ENDI

49: ENDD

50: ? CHR(14)+“I19,I1”

51: QUIT

现在,若要欣赏一首乐曲时,就可用 Edlin 或 Wordstar 等编辑软件编写一个对应的类简谱,作为源数据文件。然后,执行“翻译”程序把类简谱文件翻译成用 BASIC 语言写的目标文件(即机器曲谱)。这目标

文件一执行就可听到优美的音乐了。

附:笔者在 PC 机上用 BASIC 语言编的翻译程序的清单。

### 程序清单

```

10 DEFINT B-N ; DEFSTR O-T ; DIM R(4000)
12 PRINT ; INPUT "INPUT original data file _name:" , ORIG
13 OPEN ORIG FOR INPUT AS #1
15 INPUT "INPUT objective JP file _name ( * * *.BAS) :" , OBJT
16 OPEN OBJT FOR OUTPUT AS #2
20 PRINT "PRINT " PLEASE WAIT!"; PRINT
25 I=0 ; TYP=" " ; M(0)=4 ; M(1)=8 ; M(2)=16 ; M(3)=32 ; M(4)=64
27 H=2 ; GPT=0 ; ISO=0 ; GAIN=-1 ; MARK=0 ; NUM=0 ; HANG=10
28 K=0 ; FST=1 ; KBGN=0 ; KEND=0 ; LOC1=-2 ; LOC2=0 ; BIANY=0
30 T=INPUT$(1, #1)
35 IF T="k" OR T="M" OR T="Z" THEN SU=T ELSE SU="Z" ; R(0)=T ; I=1
40 IF SU="K" THEN JZ=120 ELSE IF SU="M" THEN JZ=80 ELSE JZ=100
46 GOSUB 1010 ; PRINT #2, "O3 MN T"; JZ ; CHR$(34)
60 R(I)=INPUT$(1, #1)
70 IF R(I)<>"@" THEN I=I+1 ; GOTO 60
80 LARGE=I ; CLOSE #1
90 GOSUB 1010
100 FOR I=0 TO LARGE-1
105 IF I=LOC1 THEN I=LOC2 ; MARK=0 ; LOC1=-2 ; LOC2=0
110 IF R(I)>="1" AND R(I)<="7" THEN GOSUB 700 ; GOTO 105
120 IF R(I)="0" THEN GOSUB 600 ; GOTO 105
130 IF R(I)="(" THEN GOSUB 400 ; GOTO 240
140 IF R(I)="|" THEN GOSUB 1000 ; FST=1 ; GOTO 240
150 IF R(I)="^" OR R(I)="v" OR R(I)="V" THEN GOSUB 300 ; GOTO 105
170 IF R(I)="{ THEN LOC1=1 ; GOTO 240
180 IF R(I)="}" THEN LOC2=1+2 ; GOTO 240
190 IF R(I)="/" THEN PRINT #2, "ML" ; GOTO 240
200 IF R(I)="\" THEN PRINT #2, "MN" ; GOTO 240
210 IF R(I)="b" OR R(I)="B" THEN BIANY=-1 ; GOTO 240
215 IF R(I)="#" THEN BIANY=1 ; GOTO 240
220 IF R(I)="[" THEN GAIN=1 ; GOSUB 1000 ; GOTO 240
230 IF R(I)="]" THEN GOSUB 1000 ; FST=1 ; IF MARK=0 THEN MARK=1 ; I=GAIN ELSE MARK=0
240 NEXT I
260 PRINT #2, CHR$(34) ; PRINT #2, HANG+10 ; " END" ; CLOSE #2
280 END
300 Q=R(I)
305 I=I+1 ; TYP = CHR$( (VAL(R(I))+1) MOD 7 +65)
308 IF TYP < "A" OR TYP > "G" THEN 305 ELSE I=I+1

```

```
310 IF R(I)=Q THEN KEND=1 :GOTO 340
320 IF R(I)="." THEN GPT=1 :I=I+1 :GOTO 340
330 IF R(I)="-" THEN GOSUB 900 :GOTO 350
340 IF K=1 OR FST=1 THEN PRINT #2,"L";USING" # #";M(NUM); :K=0 :FST=0
350 IF KBGN=0 THEN IF Q="^" THEN PRINT #2,">"; ELSE PRINT #2,"<"; ELSE KEND=0
360 PRINT #2,TYP;
365 IF BIANY=-1 THEN PRINT #2,"-"; ELSE IF BIANY=1 THEN PRINT #2,"+";
370 BIANY=0 :IF GPT=1 THEN PRINT #2,"." :GPT=0
380 IF KEND=0 THEN IF Q="^" THEN PRINT #2,"<"; :KBGN=0
ELSE PRINT #2,">"; :KBGN=0 ELSE KBGN=1 :GOTO 300
390 RETURN
400 NUM=NUM+1 :FST=0
410 IF R(I+1)="(" THEN I=I+1 :GOTO 400
420 IF R(I+1)="0" THEN I=I+1 :GOSUB 600 ELSE 440
430 IF R(I)>="1" AND R(I)<="7" AND R(I=1)="-" OR R(I)="(" OR R(I)=")"
THEN 460 ELSE I=I-1
440 PRINT #2,"L";M(NUM);
450 I=I+1
460 IF R(I)="(" THEN 400
470 IF R(I)>="1" AND R(I)<="7" THEN GOSUB 700 :GOTO 460
480 IF R(I)="0" THEN GOSUB 600 :GOTO 430
490 IF R(I)="." THEN PRINT #2,"." :GOTO 450
500 IF R(I)="^" OR R(I)="v" OR R(I)="V" THEN GOSUB 300 :GOTO 460
510 IF R(I)="/" THEN PRINT #2,"ML"; :GOTO 450
515 IF R(I)="\" THEN PRINT #2,"MN"; :GOTO 450
520 IF R(I)="b" OR R(I)="B" THEN BIANY=-1 :GOTO 450
525 IF R(I)="#" THEN BIANY=1 :GOTO 450
528 IF R(I)<>")" THEN 450 ELSE NUM=NUM-1
530 IF RST<>"" THEN PRINT #2,"L";M(NUM);RST;" MN"; :RST=""
533 IF NUM<>0 THEN K=1 :GOTO 450
540 K=0 :FST=1
550 RETURN
600 I=I+1 :ISO=ISO+1
610 IF R(I)="0" THEN 600
620 IF M(NUM) MOD ISO=0 THEN L=M(NUM) /ISO ELSE L=M(NUM)\ISO+1 :GPT=1
630 PRINT #2,"P";USING" # #";L; :ISO=0
640 IF GPT=1 THEN PRINT #2,"." :GPT=0
650 RETURN
700 TYP =CHR$( (VAL(R(I))+1) MOD 7 +65)
710 I=I+1
720 IF R(I)="-" THEN GOSUB 900 :GOTO 800
730 IF R(I)="." THEN I=I+1 :GPT=1
740 IF R(I)="/" THEN PRINT #2," ML"; :GOTO 710
750 IF R(I)="\" THEN PRINT #2," MN"; :GOTO 710(下转 34 页)
```

# 单片机控制系统硬件设计研究

西北工业大学计算机科学与工程系 刘熙 龙卫红 程智强

**摘要:** 本文针对实时控制应用的特点,研究了提高单片机控制系统(实时)性能的两种硬件技术,即伪 DMA 技术和共享存储器双机互连技术。实践证明[1],文中给出的电路是现实有效的。

**一、引言** 单片计算机具有集成度高、功能强、价格低、体积小、可靠性高等优点,因而在智能仪器仪表、家用电器和实时工业控制等控制应用领域得到广泛应用。本文针对实时控制应用的特点,研究了提高单片机控制系统(实时)性能的两种硬件技术,即伪 DMA 技术和共享存储器双机互连技术。实践证明[1],文中给出的电路是现实有效的。

**二、单片机的伪 DMA 技术** 实时控制系统最基本的功能之一是要实现对被控制对象的特定参数采样以及对被控制对象的控制参数的初始化和动态调整。例如,对某被控制对象的温度参数进行采样,若采样值超过额定阈值,则修改相应的温度控制参数进行降温

处理。单片机控制系统中,对被控制对象的特定参数采样实际上是将参数从采样参数寄存器(SPR)读入到单片机内部寄存器(如 8031 单片机的累加器 A[2])再送入数据存储器 M 中;相应地,对被控制对象的控制参数的初始化或动态调整则是将相应的控制参数值从 M 读出经单片机内部寄存器送入到控制参数寄存器(CPR)。显然,上述数据传送(流动)都经过单片机 P,即,  $M \leftrightarrow P \leftrightarrow SPR/CPR$ ,这在实时性要求较高的场合往往不能满足应用的要求。对此,有效的途径是(不增加太多成本和电路复杂性)建立 M 与 SPR/CPR 之间的直接数据通道(伪 DMA 通道)。

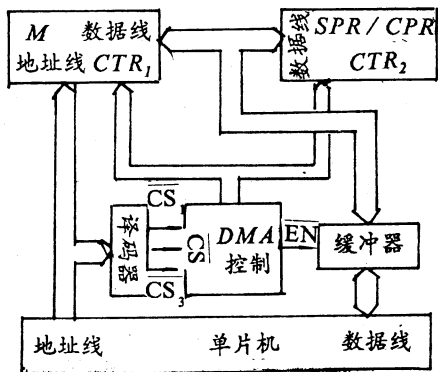
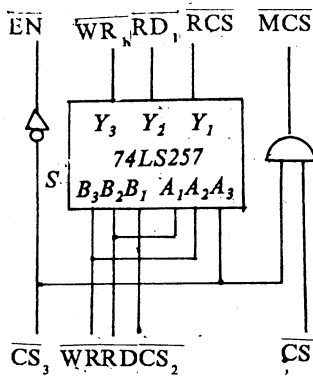


图1 单片机伪DMA控制逻辑

图1是我们设计的一个伪DMA控制电路[1]。电路的基本思想是:将读SPR/CPR的RD信号同时作为M的WR信号,实现伪DMA方式读SPR;同理,伪DMA方式写CPR时,则将CPR的WR信号同时作为M的RD信号。图中,CTR1为M的控制信号集合,即  $CTR1 = \{\overline{WR}_2, \overline{RD}_2, \overline{MCS}\}$ ,其中  $\overline{WR}_2 = \overline{WR}$ ,  $\overline{RD}_2 = \overline{RD}$ ;CTR2为SPR/CPR的控制信号集合,  $CTR2 = \{\overline{WR}_1, \overline{RD}_1, \overline{RCS}\}$ 。必须指出的是,当SPR/CPR与M进行伪DMA



操作时,必须关闭它们与单片机之间的数据通路。否则,当从M读数到CPR时,将导致将M中的数也读到单片机中,从而破坏单片机内部寄存器的值(如8031单片机的累加器A);而当从

SPR读数到M时,又将单片机内部寄存器的内容送上数据通路,与从SPR读出的数据发生“线与”,从而出错。图1中的缓冲器即完成隔离单片机数据通路与伪DMA通路的功能,它可由74LS244或74LS245芯片构成。

**三、共享存储器互连技术** 单片机受处理能力或开发工具的限制,一般不单独作为系统控制机器。通常

是将其与另一种功能较强的机器系统互连以实现较复杂的系统控制。这里,很重要的一个问题是双机间的互连通信。一般地讲,单片机都提供有串行通信功能(如8031单片机提供一个全双工串行通信口[2]),但串行通信往往不能满足控制系统的高实时性要求。对此,最有效和最直接的解决办法是采用双端口存储器来实现机间互连。目前,市场上出售的双口存储器价格较昂贵,这无疑会增加研制费用和生产成本。这里,我们提出一种采用一般RAM构成双口存储器CM(Common Memory)的方法[1]。该方法能获得近似于标准双口存储器的性能,而且结构简单,易于实现。

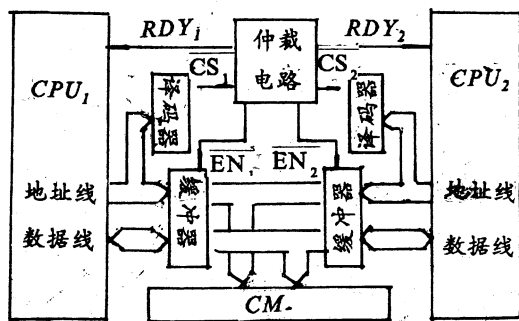


图2 共享RAM

**1、基本原理** 标准双口存储器有两套地址线、数据线和读写控制线引脚,分别由两个CPU使用,双方互不干扰。而一般的RAM只有一套线路,双CPU合用时必然会出现冲突。解决的办法是让两个CPU交替使用这同一套地址线 and 数据线(用缓冲器将两个CPU的地址线和数据线隔开)。如图2所示,当CPU<sub>1</sub>或CPU<sub>2</sub>要访问CM时,地址译码器产生CM片选信号( $\overline{CS}_1, \overline{CS}_2$ )向仲裁电路发出请求,仲裁电路根据是否访存冲突,通过发RDY<sub>1</sub>或RDY<sub>2</sub>信号延迟相应CPU的访存操作,同时通过发使能信号 $\overline{EN}_1$ 或 $\overline{EN}_2$ 启闭相应的缓冲器,从而实现CM的正确读写。

**2、8031与8086的共享存储器互连** 必须注意的一个事实是,8031单片机没有RDY引脚[2],即8031单片机对共享RAM的读写操作不能被延时。因此,在设计双口存储器互连电路时应做特殊处理。假设8031为图2中的CPU<sub>1</sub>端,8086为CPU<sub>2</sub>,图3为我们设计的一个仲裁电路[1]。

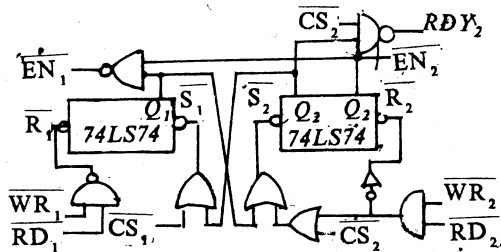


图3 共享RAM仲裁逻辑

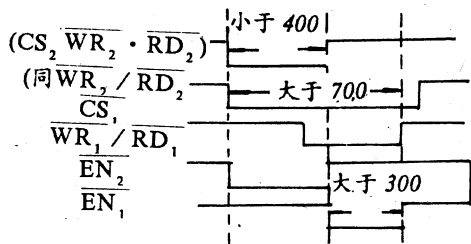


图4 共享RAM读写时序分析

读者一定注意到了,8031的访存申请信号为 $\overline{CS}_1$ ,而8086为 $(\overline{CS}_2 + \overline{WR}_2 \cdot \overline{RD}_2)$ ,即推迟了8086的访存申请。这样,尽管8031的访存操作不能加入等待周期,但时序上已足以保证在发生访存冲突时,8031和8086都能正确读写。即,

- 当8031获准访问CM时,RDY<sub>2</sub>为高。这时,若8086发出访存请求,RDY<sub>2</sub>将使8086访存操作延迟。

- 当8086获准访问CM时,若8031请求访问CM,时序上仍能保双方正确读写。如图4所示,对于主频大于或等于5MHZ的8086CPU来说, $\overline{WR}_2/\overline{RD}_2$ 的宽度小于400ns,而12MHZ的8031从 $\overline{CS}_1$ 有效到 $\overline{WR}_1/\overline{RD}_1$ 的上升沿之间有大于700ns的时间。显然,8031可以有多于300ns的存储器读写时间,这保证了CM的无冲突访问(一般静态RAM的存取时间 $\leq 150$ ns)。

**四、结语** 本文研究了提高单片机控制系统(实时)性能的两种硬件技术:伪DMA技术和共享存储器双机互连技术。文中给出的电路经作者的实践证明是现实有效的。

#### 参考文献

- [1]刘熙,“MIL-STD-1553B 串行总线分析器的设计与实现”,西北工业大学硕士论文,1988,1。
- [2]《MCS-51 微计算机用户手册》,复旦大学微机开发应用研究室编,1986,5。



# 算法语言查错技术及查错词典

西北工业大学 牛海发

**摘要:** 本文以 FORTRAN77 语言为例,较全面地阐述了算法语言查错技术及查错词典的概念和具体应用。

**一、问题的提出** 计算机程序设计语言,已经成为大部分大专院校各类学生教学计划的必修课程。计算机算法语言种类繁多,就应用比较广泛的 BASIC、COBOL、PASCAL、FORTRAN 和汇编等语言而言,它们的学习对象不尽相同,用途也有所差别。用各种语言设计编写出的程序,在上机操作运行时,都会出现各种各样想不到的问题或错误信息,要解决这一问题,提高使用计算机的效率,就必须掌握算法语言查错技术。本文根据作者多年从事计算机系统维护管理工作和指导各类学生,使用算法语言在计算机上所进行的许多课程设计、毕业设计等工作,以其中具有分时操作系统、多用户终端的(Alpha)小型机的 FORTRAN77 语言为例,较全面地介绍在运行阶段出现错误故障的分析、查找和处理技术,以及查错词典的概念和应用。

**二、运行阶段的错误分析处理** 通常所说的运行阶段的错误故障是指:程序在经过编译、连接编辑以后,执行时所出现的错误。这类错误一般都不大容易查找,特殊的也就更难排除了。因为,它不仅需要用户掌握一些有关编译系统和操作系统方面的知识,而且还要求用户对程序在运行阶段的内存、数据动态变化能够作一定的分析,并有所估计等。为了解决这方面的问题,根据实践经验,在这里介绍一些查错的方法和技术。

## 1、人工逐条分析查找方法

**方法一:**根据一般的原则和经验,采取对错误进行跟踪、定位、分析并处理等。如用 PA USE 语句分段调试。

**方法二:**查找与错误有关联的程序段就能定出准确的位置。比如,有一程序是“函数自变量有错”,分析程序后,查找了与主程序有关联的子程序,发现错误出在对负零(或负数)求开方。

**方法三:**当把错误限制在局部小范围后,对与错误有关联的语句进行考察,就可准确的找到出错语句。如“被零除”的错误出现后,首先把范围缩小,对仅余的程序段,首先找出有除法的算式,对分母分别加以修正之后进行试探考察,就能找出出错的除法算式,对有些动态数据错误就可以加以控制。

**2、利用计算机的单步执行命令 DEBUG,可以对较大的程序进行错误追踪查找。下面通过几个不完整的**

**程序实例,介绍分析、查找、处理方法:**

**举例一:**运行后结果不对

简写程序如下:

```
DIMENSION N(10),XK(40),YK(40)
M=40
...
CALL SB(XK,YK,X,Y,M)
...
SUBROUTINE SB(XK,YK,X,Y,M)
DIMENSION X(5),Y(5),XK(M),YK(M)
...
```

分析:该程序在运行阶段并没有给出错误信息,但在用户建立的数据文件中,出现了一些乱七八糟的字符,从而使得结果文件数据遭到破坏而不正确。这样的错误比较麻烦,因为它表面上并没有错误信息出现,会使人觉得莫名其妙。甚至有人怀疑是计算机系统出了问题。但从计算机的可靠性来说,一般是不会有问题的,除非计算机病毒产生,才会对系统造成危害。经过对程序检查分析,发现错误出在子程序 SB 中的 X 和 Y 两数组上,原因是它们是两个数组,根据程序上下文关系,应在主程序中加以定义处理。即在主程序中,把 X、Y 定义为数组:

```
DIMENSION X(5),Y(5)
```

重新编译、运行,结果完全正确。

**举例二:**运行阶段分别三次出错分析处理实例:

程序如下: DIMENSION X(24),Y(24),XK(5),YK(4),A(3,4)

```
OPEN(2,FILE='ZH.DAT',STATUS='OLD')
OPEN(1,FILE='ZH1.DAT',STATUS='NEW')
...
DO 3 I=1,24
XK(1)=Xk(1)+1
...
WRITE(2,7)(A(I,4),I=1,3)
FORMAT(3X,3F10.4)
STOP
END
SUBROUTINE SUB(N,A)
DIMENSION A(3,4)
```

...

DO 9 J=K,M

A(K,J)=A(K,J)/P

...

END

第一次运行程序:

a) 由于语句:  $XK(1) = Xk(1) + 1$ , 而产生了 75 号错误, 错误信息如下:

? subprogram not found in ZH

在 ZH 文件中没有找到子程序。

b) 查错处理:

检查后, 由于赋值语句中的右部表达式  $XK(1)$  数组元素中的 K 写成了 k。这是由于 Alpha 机把大小写字母作为不同的字符而产生的。只要把上语句中的小写 k 改成大写 K 即可通过。

第二次运行程序:

a) 由于语句:  $WRITE(2,7)(A(1,4), I=1,3)$  而产生了 16 号错误, 错误信息如下:

? File not open in ZH

在 ZH 中文件打不开。

b) 检查处理错误: 经查错后, 找出产生的原因在于 WRITE 语句中的通道号使用不对。因为在 OPEN 语句中, 定义的输出文件(一般状态  $STA\ TUS = 'NEW'$  为输出文件)通道号为 1。所以在 WRITE 语句中, 通道号必须用 1 才对。而在本程序中用的是 2, 必然在程序执行中产生 16 号错误。

处理: 只要将 WRITE 语句中的通道 2 改为 1 即可。

第三次运行目标程序: a) 由于语句:  $9\ A(K,J) = A(K,J)/P$  而产生了 66 号错误, 错误信息如下:

? Divide by zero in suB

在 SUB 中被零除。

其中, SUB 是子程序名。这类错误也可能在主程序中出现。

b) 检查处理错误: 这类错误产生的一般原因是: 除数为零或接近于零。接近于零就是该数比 Alpha 机的最小实数  $10^{-38}$  还来得小, 则计算机也作为零处理。

经过检查此程序的子程序 SUB 后, 发现该子程序中只有一个语句中有除法, 则可以肯定是该语句的分母为零或接近零, 也就是 P 为零。改错时只须把 P 的值调整后, 如加以 0.00001, 再运行后, 错误消除。一般对于具体问题, 用户可根据自己实际问题要求: 加以修正。

在修改了上述三个错误故障之后, 再经编译、运行目标程序, 得到了正确结果。

### 三、运行阶段死锁、死机故障的分析处理

一、死锁: 这个概念听起来并不陌生, 然而, 这里所讲的死锁并不同于操作系统中的死锁概念, 这种死锁实质上是一种死循环。这种用户程序造成的故障, 对计算机系统有一定的危害。因此, 这类错误要及时处理, 不能让其发展下去。特别是对磁盘文件不断产生读写错误, 程序运行中断不了, 一直在不断地造成磁盘位图的读写错误, 以致使磁盘位图发生混乱, 造成不能读写, 影响其他用户使用磁盘设备。一般会在各用户终端上出现 Device full(设备已满)和 File not open(文件打不开)等错误信息。使得各用户无法将程序存盘, 不能使用计算机系统。

1) 错误发生的原因: 这种错误, 在编译阶段和运行阶段都有可能发生。原因是: 对 FUNCTION 函数子程序定义后引用上的违法; 使用 WRITE 语句和对应的 FORMAT 语句有问题; 使用 READ 语句和 FORMAT 语句有错误而造成。

2) 死锁后在屏幕上的信息: 一般是整个终端屏幕闪烁“\*”号, 但是, 有时也有其它字符闪烁在整个终端屏幕。如: “A(1), A(2), A(3), ..., A(n)”这种故障只是发生在某个用户终端上。

#### 3) 发生死锁后的故障处理方法

a) 一般可在旁边的终端上调出某一个占用内存比较大的程序, 将出现死锁用户的程序冲掉, 以免故障继续延伸下去, 这样死锁可消除, 必要时, 也可在主机控制面板上热启动系统。

b) 把发生死锁的用户程序调出, 主要检查 WRITE 语句和 FORMAT 语句。一般故障就可以排除。下面举一实例, 程序简写如下:

...

WRITE(100,200)M,K,BIG,SMALL

WRITE(9,200)M,K,BIG,SMALL

...

200 FORMAT(1X,F10.0,F10.0,4HBIG=,6HSMALL=)

...

程序在运行时出现死锁, 其信息为: \* \* \* \* \*  
\*, 在分析这类错误时, 着手应先检查 WRITE 语句和对应的 FORMAT 语句。检查后错误在标号为 200 的格式语句, 应改为: 200 FORMAT(1X, 2I5, 4HBIG=, F10.0, 6HSMALL=, F10.0)即可。

#### (二) 死机故障

所谓死机, 就是 Alpha 机的 AMOS 分时操作系统已经完全瘫痪, 失去对各用户终端的控制。死机常常还

会造成主机出现 24 号或 27 号假象的硬件故障。

死机一般也是由程序错误引起的故障,而且是错误故障中最为严重,对用户和计算机系统危害最大的一种故障,它可以加速计算机系统某些元器件或电路上的损坏,特别是存储器模块的损坏,从我校其它 Alpha 机系统的损坏上,已经证明了这一点。所以,对此类故障决不可掉以轻心,必须加强防范,才能保证系统的正常运行。该故障发生后,使得各用户程序已无法运行,只有重新启动系统,才能再使用计算机系统。

引起死机故障的原因很多,一般常见的是数组下标失控或数组没有定义等问题所致。

数组下标失控,是指定义的数组小,而使用的数组大。数组下标失控造成死机的原因是引用数组时用表达式计算机下标而造成的。

数组没有定义,将会导致计算机把数据错误地当成指令。这样有可能引起不存在指令的执行,进而发生死机故障。所谓不存在指令,是指产生了 Alpha 机指令系统以外的指令。发生不存在指令有两种可能:一种是在本终端上出现;另一种是在紧接着的下一个用户终端上出现。它将造成内存地址的破坏。

分析处理死机故障,要使用户了解发生死机的危害,尽量避免死机故障的发生,不断提高编程质量,提高调试程序的效率,充分利用计算机系统资源,减少不必要的人力、物力浪费。

**四、查错词典** 所谓查错词典,就是把出错信息和出错语句一一对应起来,供用户查阅。使上机用户能方便地在调试程序中使用,不需要过多地思考分析。特别对初上机的学生用户尤为适用,对他们上机实习可以起到事半功倍的效果。

查错词典是笔者多年来经过对大量的程序错误分析、查找、处理,并根据每个错误出现的频率情况,把它们归类后所形成的。现仅列举以下最常出现的几个词加以说明:

1) 错误信息为: file no open, 应检查 READ 或

WRITE 语句中的文件通道号是否和 open 语句中的通道号相同,并注意把它改为和 OPEN 语句中的 STATUS 状态一致即可。

2) 错误信息为: 全屏 \* ; 首先检查 WRITE 和对应的 FORMAT 语句或 READ 和对应的 FORMAT 语句,一般问题都出在这里,如果这两条语句都正确。再检查 FUNCTION 及其调用 CALL 语句中的形、实参关系。错误检查出后,按正确修改即可。

3) 错误信息为: Adress 地址号 Error; 应检查数组名的定义和使用关系,检查出后,按定义后才能使用的原则,修改数组名即可。

4) 错误信息为: 死机, 检查时首先应热启动系统,再调出程序检查数组下标的使用,特别要检查数组下标使用表达式的语句,检查出后,根据题目要求,可把 DIMENSION 语句中定义的该数组加大即可。

5) 错误信息为: in 文件名 subprogram 子程序名 not found; 检查: ①内部函数名是否正确。②调用语句和子程序中的虚实元结合是否正确。③还有一种情况,可能是设有定义的数组,被误认为是内部函数等。分别按上述次序查找,查出后,根据题目要求进行修改即可。

6) 错误信息为: Device full; 这种错误信息一出现,应由系统维护人员做磁盘检查维护,之后用户才能使用计算机系统。

7) 错误信息为: Out of memory

? Illegal instruction at 地址号

此类错误只要用 VIRTUAL 语句把某大数组,如 A 数组定义在磁盘上,即 VIRTUAL A(200,200)即可。

8) 错误信息为: ? Divide by zero in SUB; 这是由于子程序 SUB 中某除法算式的分母为零或接近于零,根据题意进行修改。

9) 错误信息为: 编译阶段出现 \* ; 原因是 FUNCTION 函数变量有问题。检查: 形、实参结合是否正确,若不正确,修改即可。

# 一个实用化的语音分析/合成系统的设计

厦门宏泰发展有限公司工程设计部 刘榕和

**摘要:**本文将讨论一个建立于 IBM PC/XT、AT 机上的实用化、低成本的语音分析与合成系统的设计原理、硬件及软件结构。

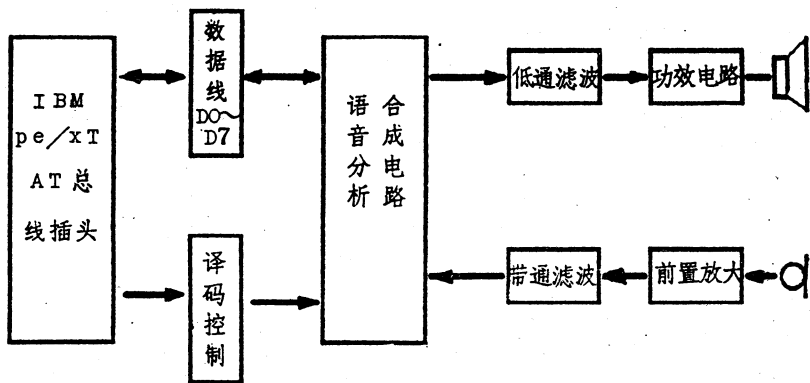
**一、引言** 随着半导体技术和微计算机技术的发展,近年来,语音分析与合成技术也有了很大的进步,被广泛地应用于各种领域,目前,基于语音分析合成系统主要从两方面来进行的:第一方面是为解决人机交互等问题而发展的,像语音的理解,语音的特征分析等;另一方面是为了解决语音的数字传输与还原而进行的。多年来,人们为实现这两方面的发展,进行了大量的研究工作,取得了巨大的成就,产生了各种各样的语音数字化及还原的方法和算法,同时,VLSI 技术产生了各种语音分析与合成的集成电路,使人们可以更方便、更灵活地利用这些电路建立起适用的系统。

语音的分析与合成主要在于语音数字化及还原方法,从应用的角度来看,目前还是以两个途径来展开的:一个是传统的利用 A/D、D/A 转换器对语音进行一般性和数字量化,然后,对这些量化后所得到的数字信息进行时域或频域上的处理工作,以求取各种语音的特性,以便利用这些特性来达到识别与合成的目的,这类方法也派生出其它相类似的语音处理方法,象 LPC(线性预测法)、谱带式声码器等;另一个途径就是

利用增量调制技术。增量调制技术由于其简单、易于实现,比较受人们的重视,同时,以增量调制技术为基础,也发展出各种各样的增量调制的语音分解与合成方法,像线性增量调制, CVSD(可变斜率增量调制), ADM(自适应增量调制)等技术,目前,大部分语音处理电路都是利用这些原理来制造的。

微型计算机特别是高性能价格比的 PC 机出现后,人们普遍地将注意力集中到了利用高性能的 PC 机来完成语音的分析与合成方面的工作,但是,迄今为止,在 PC 机上提供的语音分析及合成系统大多是利用传统的 A/D、D/A 技术设计的,价格较高,不利于推广应用。为了研究 ADPCM 技术实现的语音在时域和频域特性,以便为新型数字通讯设备的研究与设计服务,我们研制出了一个在 PC XT/AT 机上使用的语音分析/合成器系统即 VDT—I 系统,该系统使用高性能的语音分析/合成集成电路,配置了为系统而开发出的较强的软件,系统的使用表明:系统工作十分有效,价格较低,功能较强,适合于研究开发各种语音处理产品。

## 二、硬件系统结构



图(一)VDT—I 语音分析合成系统框图

VDT—I 语音分析/合成系统结构框图参见图 1, 由图可见, 其硬件结构十分简单, 核心部件是一高性能的 VLSI 语音分析/合成器件 MSM6258, 加上部分外围器件。与微计算机的接口采用 IBM PC、XT、AT 总线。可插入各种 PC 机上使用, 扬声器与麦克风通过电缆引出, 配合适当的软件, 可方便地进行合成/分析工作。

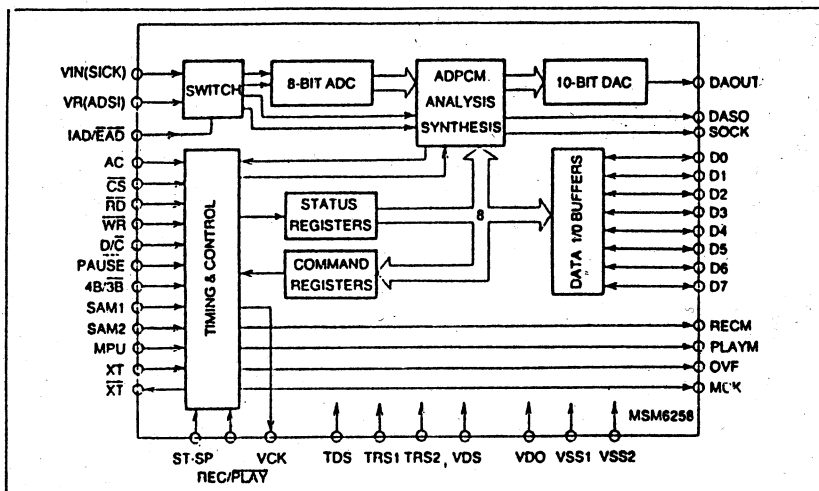
目前, 国内可见的语音电路, 大体可以分成二类: 一类是基本是以 CVSD 技术原理而设计出来的, 这类芯片主要用于语言的简单记录与重放, 仅具备写存贮器 (DRAM, SRAM) 电路的简单接口, 内部缺乏与微计算机接口使用的命令控制方式。同时, 这类芯片所记录的语音信号不经过任何压缩, 记录数据量大, 常见的有像 UM5100, UM5101, TA6668, TMS3477 等。另一类是利用了 LPC 原理而设计出来的, 像 TSP50C4X, 这类电路有微机接口, 但对语音的分解需借助于复杂且又昂贵的 LPC 分析设备, 不易使用。同时, 这二类电路主要适用于语言的合成, 难于进行识别分析工作。经过较细

致的考虑, 我们决定使用 MSM6258 语言分析/合成电路作为核心器件。

### 1. MSM6258 基本结构

MSM6258 是一个高性能的、ADPCM 方式工作的语音处理机。使用低功耗的 CMOS 技术制作。芯片上包含有完整的 AD/DA 转换电路, 可大大地减少外围电路。这种器件以二种形式封装: 一种具备了各种存贮器 (DRAM, SRAM) 的接口电路, 用存贮器来记录与重放语音信号; 另一种具有较好的微处理机接口, 可与 INTEL 公司的各种八位微计算机直接接口, 经过简单的变形, 也可以与各种微计算机连接, MSM6258 器件上具有一个语音电平检测电路及短语 (词组) 选择电路, 从而更进一步地增加了系统应用的灵活性。该器件使用 4~8MHz 频率, 通过改变不同内部时标的方法, 可以在不同时钟频率下选择不同的采样频率方式。

MSM6258 的与 CPU 接口的基本形式见图二 (a), 其具有 MPU 结构的电路框图见图二 (b)



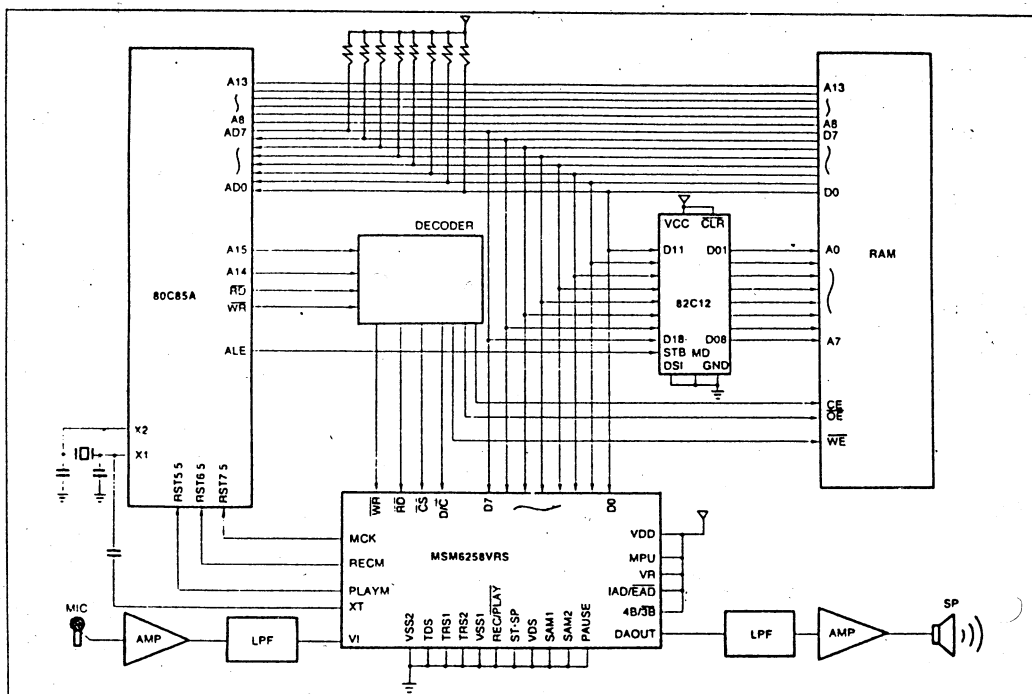
图二(a)MSM6258 内部框图

MSM6258 应用于 CPU 接口形式时, 以下管脚是应使用的:

MPU: 该管脚选择器件进入与 CPU 接口状态。当给定使该管脚为高电平时, 内部电路可以与外部 CPU 通讯。由于输入信号在电路内部被下拉了, 因而必须进

入高电平状态来使用该管脚。

CS: 片选输入端。当该管脚接收到低电平信号, 器件将开放数据总线与外部 CPU 之间进行数据传递。若设置该脚为高电平信号, 则器件将呈现高阻状态, 从而禁止了读/写操作。



图二(b)与 80C85—A 微处理器接口形式图

MCK;该管脚可以输出一个用来与 CPU 同步的信号。在记录语音或重放语音时,在该管脚上输出了一个等于二分之一采样频率的周期信号。

$\overline{\text{RD}}$ :当使该端为低电平时,CPU 可以从 MSM6258 读取语言编码数据或状态数据。

**WR**:当该端为低电平时,CPU 可以向器件写入合成数据或控制命令。

$\overline{D/C}$ : 这是命令/数据选择管脚。当选择为高电平时, 数据总线提供的是语音编码数据。若使该管脚为低电平, 则 MSM6258 将存取命令或状态信息。

D。 $\sim$ D<sub>7</sub>:数据总线端,可以与 CPU 数据总线直接接口。

SAM<sub>1</sub>, SAM<sub>2</sub>: 利用施加于这两处管脚以不同电平, 可以选择不同的采样频率。见下表:

SAM <sub>1</sub>	SAM <sub>2</sub>	采样频率	设主频率为 4. 096MHz
L	L	$f_{s1}=f_{oso}/1024$	4. 0K
H	L	$f_{s2}=f_{oso}/768$	5. 3K
L	H	$f_{s3}=f_{oso}/512$	8. 0K
H	H	非法状态	

利用该表,可以求算出在选定的采样频率和采样时间时,所需要的存贮容量:

$$K(\text{位}) = t(\text{秒}) \times \text{采样频率} \times \text{ADPCM 位数}(K \text{ 位/秒})$$

DASO:数字/模拟串行输出端。在该管脚可以得到在记录与重放(合成)工作方式中的复原的语音 PCM 数据。

VI(SICK):该管脚也是一个模拟/数字管脚。取决

于选取什么样的 AD 转换器。当使用内部的 A/D 转换, 可以输入一个经滤波了的语音信号。但使用外部 AD 转换器, 则可以输入一串 PCM 数据, 输入用同步时钟。

DAOVT:这是用来输出合成的语音信号的。输出的信号可以经低通滤波器滤除其高频谐波。

利用以上管脚的功能,我们可以得出数据总线在不同选择管脚电平下的状态表:



CS	D/C	RD	WR	状态	CS	D/C	RD	WR	状态
L	H	L	H	ADPCM 数据输出	L	L	H	L	命令输入
L	H	H	L	ADPCM 数据输出	H	X	X	X	高阻状态
L	L	L	H	状态数据输出					

MSM6258 所提供的数据格式可以是 3 位/4 位形式。

总线	D	D	D	D	D	D	D	D
4 位数据	B <sub>0n</sub>	B <sub>1n</sub>	B <sub>2n</sub>	B <sub>3n</sub>	B <sub>0n+1</sub>	B <sub>1n+1</sub>	B <sub>2n+1</sub>	B <sub>3n+1</sub>
3 位数据	XX	B <sub>0n</sub>	B <sub>1n</sub>	B <sub>2n</sub>	XX	B <sub>0n+1</sub>	B <sub>1n+1</sub>	B <sub>2n+1</sub>

在 4 位数据格式中, B<sub>3</sub> 为符号位, 当其为“1”时, 表示语音波形是递增的, 随后的 B<sub>2</sub>~B<sub>0</sub> 分别表示语音的 3 位情况。

在 3 位数据格式中, B<sub>2</sub> 为符号位。其意义与 4 位形式类似。B<sub>1</sub> 与 B<sub>0</sub> 则分别表示高位与低位。

MSM6258 所提供的与 CPU 接口的最方便之处, 就是具有软件控制用的命令与状态数据。当 CPU 发送命令到语音芯片时, D<sub>0</sub> 为“1”, 可以强行停止记录或合成工作。D<sub>1</sub> 为“1”时, 可以启动合成输出工作。D<sub>2</sub> 为“1”, 可以启动芯片开始记录。D<sub>3</sub>~D<sub>7</sub> 位(D<sub>3</sub> 为最高位) 可以用来设置初始指针(通常可以设为 0); 当 CPU 从语音芯片读取状态数据时, D<sub>0</sub>~D<sub>4</sub> 可以设置初始指针(D<sub>4</sub> 为最高位), D<sub>5</sub> 位不使用, D<sub>6</sub> 为“0”表示处于静音状态, 为“1”表示有语音输入。D<sub>7</sub> 位为“0”表示处于合成输出状态, 为“1”表示处于记录状态。利用状态位, 经过一些办法进行处理, 要以实现语音记录与合成的间歇控制。

2. 外围电路设计的考虑 应用 MSM6258 芯片建立语音处理系统时, 外围电路的设计特别是滤波器的设计是十分重要的。一般的语音带通常从 300Hz~3400Hz 左右。为了加宽语音谱带宽, 以获得较好的质量, 目前, 在有些系统上使用宽带语音, 带通从 100Hz~7000Hz 左右。但是, 在宽带情况下, 为保证采样定理, 必须提高采样速度, 这样不仅增加了采样数据量, 还必须考虑在与 CPU 的接口设计上, 采取措施, 以提高响应速度。在 VDT-I 系统中, 由于考虑到该系统主要是为通信系统中的语音分析和合成设计作为开发式工具, 因此, 选用了窄带语音频谱, 因为这个带宽已经能够较好地复盖“电话质量”的训练样本。

从系统结构框图已经看到, 正常的语音信号, 经过前置放大后, 必须经过一个带通滤波器进行预处理, 然

后送到语音识别器。对于这个带通滤波器的主要要求是具有较高的 Q 值、良好的稳定性、调整容易。为此, 选用了双二次带通滤波器。电路形式可见图 3(a)。其传递函数:

$$V_2/V_1 = GB_0 / (S^2 + B_0S + W_0^2)$$

式中 G 为增益,  $S = j\omega$ , B 为通频带。

电路的设计参数为  $G = R_2/R_1$ ,  $B = 1/(R_2C)$ ,

$$W_0^2 = 1/(R_3R_4C^2)$$

由于电路的设计参数可知为调整带通滤波器, 十分简单, 改变 R<sub>1</sub>, 可以改变增益, 由  $Q = f_0/B$ , 可知改变 R<sub>2</sub> 可以改变 Q 值。而中心频率则可以利用 R<sub>3</sub> 来调整。同时, 可以利用多个二阶双二次带通滤波器来级联以获得高阶的带通滤波器。

在语音输入过程中, 麦克风的选择也是十分重要的。使用驻极体型麦克风, 灵敏度高, 高频特性好, 但同时也产生噪声大, 语谱强弱分配不均匀等缺点, 从而影响语音量化数据, 尖峰过多, 影响识别的质量。因此, 应选用电动式麦克风。这种麦克风低频语谱与高频语谱分布较平坦、噪声小, 语音量化的特性明显改善。

MSM6258 的输出是脉宽调制型信号。因此, 必须采用低通滤波器消除输出信号的高次谐波的噪声。良好的低通滤波器对获得高质量的合成语音是十分关键的。为此, 我们选用双二次低通滤波器, 其截频点为 3600Hz, 以利于合成较高的女声。电路形式见图 3(b)。其传递函数为  $V_2/V_1 = Gb_0 / (S^2 + b_1S + b_0)$

设计参数为  $G = R_3/R_1$ ,  $b_0 = 1/R_3$ ,  $b_1 = 1/R_2$

双二次低通滤波有非常好的调整特性和稳定性, 改变部分元件值, 即可改变其设计参数。R<sub>1</sub> 影响着增益, R<sub>2</sub> 影响着通带响应, R<sub>3</sub> 则可以改变截止频率。为了获得较好的合成语音频率特性。我们在设计中使用二个二阶双二次低通滤波器进行级联, 使输出的语音的

包络十分好,还原时的逼真度较高。

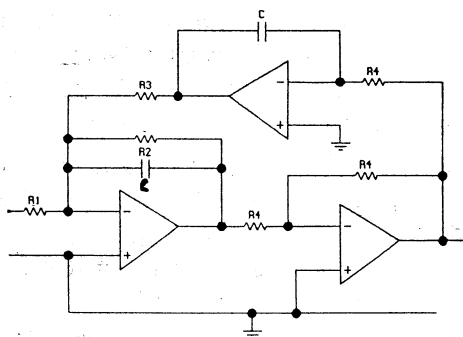


图 3(a)二阶双二次带通滤波器

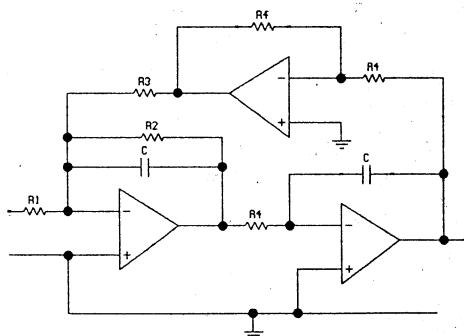


图 3(b)二阶双二次低通滤波器

### 两种新颖方便的“P”解密方法

关于 IBM BASIC 程序的“P”加密解除方法,已有很多。笔者这里提供二种方法,既新颖,又甚为简单方便。

方法一:

第 1 步:用命令 BSAVE “UP”,0,1 建立一个文件。执行该指令后,会在磁盘上产生一个名为 UP.BAS 的程序。

第 2 步:用 LOAD 命令将已进行“P”加密的 BASIC 程序装入内存。此时若用 LIST 命令列出程序清单,会出现“Illegal function call”提示。

第 3 步:键入命令 BLOAD “UP”,1124。那么,加密的程序清单就被列出来了。

方法二:

第一步:将下面程序输入并执行。

```
10 CLS
```

```
20 INPUT “输入文件名:”;P$
```

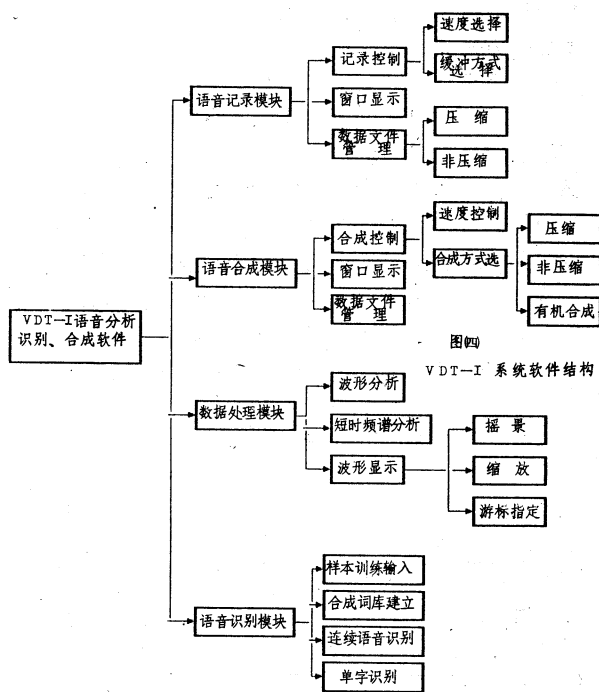
```
30 FOR A = 1 TO 6 : READ B : POKE A + 17, B : NEXT
```

```
40 PRINT “S = 18 : Call s” : LOCATE 1, 1 : LOAD P$
```

```
50 DATA 48, 192, 162, 100, 4, 203
```

第 2 步:当程序要求输入文件名时,将已加密的 BASIC 程序名(含扩展名)键入。程序执行后会显示“S = 18 : Calls”,此时光标在该行之首,再按<Enter>或<Return>,便完成解密工作。

第 3 步:用 LIST 指令就可将已进行“P”加密的程序清单列出来。(安徽 陈庆章)



图四:语音分析合成系统程序结构图

### 三、软件结构

为了使 VDT-I 语音分析/合成系统有效地工作,为其研制了较强的软件系统。其基本程序组成见图四。程序使用 Microsoft C 语言 5.1 版及宏汇编语言混合编制。总量为 200K 左右,使用窗口显示技术。系统工作的全部动态过程都可以较好地反映出来。

1. 语音记录模块 该程序模块主要完成语音的量化工作。在这个工作中,通常应该能够正确地反映出语音分析/合成器件的各种状态信息,同时能够控制系统完成数据记录,以及进行有关的数据转换工作。因此,在系统上设计了三个窗口:两个状态窗口,一个控制窗口。第一状态窗口显示出语音芯片的工作状态,像记录、暂停、静音等;第二状态窗口则可以看出在记录过程中,每秒的数据记录量(字节),以及开始记录工作以来总的时问(S),静音时间(S),以及记录数据的总字节数。利用状态窗口可以观察在不同环境下,同一训练样本受环境的影响情况,也可以观察同一环境下,不同年龄组的训练样本的差异性。控制窗口,可以利用鼠标器或键盘移动光标来控制记录的起动、暂停和停止,也可控制数据格式(3位、4位、压缩)和选择记录速度,或设立缓冲方式。系统还提供定时记录方式。

在记录中,数据缓冲区的设置是十分重要的。因为相对于磁盘的存取周期来说,语音系统采样速度较高,不可能随时将数据写入盘上。为此,设计了两种缓冲方式:第一种为固定区分配算法,通过搜索机内存贮器,以确定最大可用的内存空间,然后以1K字节为一块形式分配给语音系统。同时,在状态窗口显示出来,当数据存满后,系统停止工作,这里,数据一次性复制到指定的磁盘上;第二种方式为随机分配法,使用环形缓冲区,利用语音间歇时的静音过程,将记录数据写入指

定磁盘。这种方法适用于内存容量较小的系统上。

在语音记录过程中,由于人的语音具有间歇时间。为此,除去间歇时间的数据是有意义的。这种消除了间歇数据的格式称为压缩形式。压缩形式下有二个数据文件同时存在,一个是BIN形式的语音数据文件,另一个是BIN形式记录间歇时间,其分辨率可达到1ms,提供压缩形式,可以为将记录的数据逼真还原创造有用利条件,同时,也减少了语言间歇时间所占用的空余数据容量,提高内存的利用率。当然,也可以选用非压缩的语音记录方式,即将采取的ADM语音数据直接存放。

2. 语音合成模块 语音合成是分析的逆过程,其显示窗口与语音记录模块基本是类似的,但对语音合成来说,其数据格式是不大一样的,有三种格式:第一种为非压缩形式,直接将记录的语音数据重放。第二种形式为压缩形式,即将语音记录压缩形式的文件重新进行还原恢复。第三种格式是有机合成法,即可以把按各单字发音的数据通过链表的形式有机的组织起来,形成较长的语言,同时,利用尾音数据来模拟人类自然发音的间歇性,从而能够更为逼真地合成出语音。这种合成方式是一般的语言处理芯片所难以做到的。另外,若进行混合处理,还可以创造出自然环境的混合声,使合成出的语音更为柔和、自然。

使用有机合成方式的链表格如下:

起始标志	计数长度
区 号	起始位置
起始位置	计数长度
计数长度	结束标志

区号——说明某一字发音位于某一个局部语音数据文件中。目前按国标中文分区方法来确定分区;起始数据位置——某一字发音在分区文件中的具体位置;

计数长度——相应于该字发音数据的总长度。

链表长度仅受到磁盘驱动器剩余存贮容量的限制。因为在语音合成开始以前,软件将根据需要合成的中文文字自动算出有机合成方式中的链表,然后根据该链表在各局部发音数据文件中抽取数据,送往内存缓冲区进行语音合成输出,当完成了语音合成时,退出该功能选项,自动地删除该链表。否则,可以重放。

致作者

- 1、本刊不受理复印稿、复写稿及铅笔、圆珠笔誊抄稿,请工笔书写(或打印)于方格稿纸上;
- 2、图纸要求正规描绘图(图中文字宜用激光照排字),并寄一份复印件附于稿中;
- 3、来稿请注明通讯地址和真实姓名,以便联系;中文标题下须附英文标题;
- 4、因人手所限,一律不退稿,请作者自留底稿;
- 5、本刊办理“优先快发”稿件手续,有意者请在来稿中注明;
- 6、为利于审稿,请作者能提供必要材料,如鉴定意见,详尽图纸,软盘,程序清单等。

3. 数据分析处理模块 数据处理是语音分析过程中的一个重要工作,通过数据处理可以对记录下的语音数据时行各种特征分析。由于该语音分析系统是以自动增量形式来对语言进行记录,因此,其特征表现形式与传统的频域分析的方法有较大的差别。在增量形式主要体现出的波形的时域变化特性,若要进行频域分析,则需要进行十分复杂的转换工作。因此,对增量形式的分析工作,我们目前只对其波形的拐点、峰值点、各峰之间的语音数据个数等进行分析统计,从而得出各种为识别工作所依赖的特征数据。同时,为了更直观地观察不同训练样本之间对同一训练所产生的各种差异(在数据窗口显示出来)。系统可以快速左右移动地观察数据曲线,也可以利用缩放功能来细致地观察局部数据曲线。

为了可以利用在以传统 A/D 转换器形式记录下来的数据时行分析工作,在数据分析处理模块中,还提供了对语音波形能量的短时富氏变换程序,该程序是一个可变系数的数字滤波器,通过对 PCM 数据时行频域分析,可以求取语音在时变频谱的各种特征,以强化分析结果。

4. 语音识别模块 对于计算机语音系统来说,主要目的有二:一个是合成——通过各种方法以合成出逼真的自然语言;另一个是识别——它是语音理解的基础。由于自然语言以及语音信号的多变性、复杂性,因此,语音的识别要比语音的合成困难得多。

为了更好地研究 ADM 方式下语音识别的特点,

在 VDT-I 语音系统软件中设计了一个实验性语音识别模块,其主要设计思想是基于动态规划时间伸缩匹配方法(DTW)。同时增加了模糊判决方法,该模块中对待识别字进行各种形态分析,然后,与样本库中的每一个字的特征序列进行比较,然后按最大相似原理来确定待识别字,详细的识别方法,此文从略。

在使用语别功能之前,应首先选择识别模块中的样本训练功能,即按一定顺序逐字地读样本词汇,每读完一个字,需将该字的实际意义输入系统,以便使系统记忆下该字的各种特征序列,存入文件中,利用该功能也可以建立起标准的合成用数据库。但建立时,需采用发音准确、清晰的语音样本,同时,系统将不对合成用数据进行特征序列的抽取与分析了。

目前,VDT-I 语音识别工作还是比较简单,识别率有待于进一步地提高。但是作为一个实验性的语音分析/合成系统,在使用之后,我们觉得基本上还是满足了设计要求的。

以上,较详细地介绍了基于 PC 机上的实用性语音分析/合成系统的基本情况。笔者认为,由于目前尚缺乏高速数字信号处理系统的应用单位,使用性能价格比较高的 ADM 的语音处理芯片设计语音分析/合成系统是十分有益的。因为可以利用它来进行中低档语音合成产品的开发和设计工作,同时语音产品目前在我国的工业与民用产品使用的较少,实用商品化的语音产品在我国的四化建设中有着十分广阔的应用前景。

### 香烟锡箔的妙用

你在使用计算机时,可能遇到这样一种故障:按某个键或某几个键时,无任何反应,但其他键均可正常工作。产生此种故障的原因有多种:①有杂物卡在键盘内,使键盘不能到位;②对应该键的印刷电路上复盖灰尘或氧化物,使该键接触不良;③按键的导电橡胶上有灰尘或导电橡胶的阻值变大,线路无法正常接通。

对于上述最后一种故障,可先用酒精棉球清除灰尘,然后用万用表测量该导电橡胶的阻值,正常时应小于 200Ω,若测出的阻值大于 200Ω,则说明导电橡胶导电性能变差。其简易修复方法是找来一张香烟锡箔,按导电橡胶的大小、形状剪一小片香烟锡箔,将其胶粘到导电橡胶上,待胶水干后,将键盘安装复位,即可恢复该键的功能。(浙江 张伟中)

## 再谈硬盘维护

齐齐哈尔市八一八九〇部队自动化站 李建华

**摘要:**本文在列举硬盘故障现象并分析其故障原因基础上,介绍了排除故障的方法和避免故障的措施。

IBM-PC/XT、AT机及其各种兼容微机的硬盘出现故障,是经常发生的事。硬盘一旦出现故障,或者无法从硬盘直接启动系统;或者不能往硬盘上写数据;或者不能从硬盘上读取数据,从而严重影响微机的正常使用。

产生硬盘故障的原因很多,解决的方法也很多,近年来,在各种杂志上发表了一些这方面的文章,这对微机用户来说都是重要的参考。

本文着重介绍我们在微机使用中,硬盘遇到的故障及其解决方法,以供微机用户参考。

**一、我们遇到的硬盘故障现象及其原因** 在我们运行的数十台微机中,除软件故障外,主要故障就是硬盘故障。

1. IBM-PC/XT、AT及其兼容机有时不能自举,转ROM BASIC。

在冷、热启动时,操作系统不能正常运行,使之屏幕不出现正常的提示符C>。而是转ROM BASIC,从而出现BASIC的提示符OK。

产生这种现象的主要原因,是由于硬盘不能正常复位和读硬盘0磁道一扇区失败所引起。前者可用“ADVANCED DIAGNOSTICS”诊断盘来诊断,从而得知具体原因;后者可能是0磁道损坏或读电路损坏而造成。

2. 开机或热启动时,C盘不能自启动。用A盘启动后,不能转入“C:”状态,并显示“INVALID DRIVE SPECIFICATION”;用FDISK操作硬盘时,显示“ERROR: READING/Writing FIXED DISK”;用格式化命令格式化C盘时“FORMAT C:”不能执行。

这种现象产生的主要原因是:(1). 运行过程中突然断电,从而破坏了硬盘上零磁道上的引导文件;(2). 在玩游戏时或刚开机时,系统在启动过程中,乱敲键、乱送命令从而破坏了引导文件;(3)零磁道损坏,从而引导文件被破坏。

3. 为了提高DOS性能,往往采用DOS3.0以上的版本格式化硬盘并装入系统,但很多软件要在DOS2.1等低版本操作系统下运行。这样,再用低版本(如DOS2.1)重新格式化硬盘时,硬盘不接受,并在屏幕上

显示“INVALID DRIVE SPECIFICATION”等。

这种现象的主要原因是:高版本的操作系统对零磁道要求不严格,而DOS2.1以下版本的操作系统必须放在零磁道。高版本的操作系统“占领”了硬盘后,对低版本的操作系统再进入硬盘有“排斥”现象。

4. 硬盘磁头仅围绕一个柱面转,硬盘工作指示灯一直亮着,时时发生“嗡”、“嗡”声,无法从C盘上启动。

这种现象就是磁头卡在一个磁道位置上了。导致这种现象的主要原因是:(1). 硬盘读写过程中突然断电,使磁头卡位,不能正常地返回到“起停区”; (2). 运输或搬运过程中,没有将磁头“抬起来”(即离开数据区,返回起停区),从而使磁头在“随意”运动中,卡在某一位置上。

实质上,硬盘的自举引导是由硬盘ROM BIOS的INT19模块、位于硬盘0道1扇区的第一引导模块和位于0道2扇区的第二引导模块所负责完成的。INT19模块负责把硬盘磁头复位至“00”道,把0道1扇区上的内容读入内存0:7C00单元去,否则INT19引导失败,转入ROM BASIC。硬盘第一引导模块主要是检查从0道1扇区读入的引志指示器是否有效,并读入0道2扇区的第2引导模块(BOOT),并同时检查BOOT块的最后两个字节是否有效签名“55AA”。

除磁头卡住等机械方面的故障外,都应根据磁盘自举过程的基本原理去寻根排除故障。

**二、排除故障的方法** 在排除故障前,若能从硬盘上读取数据,则应在A驱动器系统启动后,把硬盘上有用文件和数据拷贝出来,以防止维修硬盘过程中,破坏掉有用数据,而无法恢复。

1. 上述1、2、3种硬盘故障,我们采用的排除方法我们从实践中,摸索出了一套排除这三种故障的方法:

(1). 先用PC-DOS3.0(或以上)版本的软盘,从A驱动器启动系统;(2). 在A>状态下打入FDISK(回车)屏幕显示:

CHOOSE ONE OF THE TO FOLLOWING

1. CREATE DOS PARTITION
2. SHANGE ACTIVE PARTITION
3. DELETE DOS PARTITION
4. DISPLY PARTITION DATA

## 5. SELET NEXT FIXD DISK DRIVE ENTER CHOICE

(1)

PYESS ESC TO YETUYN TO DOS

(A). 消除 DOS 分区: 选择(3), 按提示进行; (B) 重新建立 10~100 柱面的 DOS 分区。选择[1], 按提示建立 10—100 柱面的 DOS 分区, (主要目的是先避开 0 磁道, 也可选其他柱面范围建立 DOS 分区)。

(3) 将 PC-DOS3.0 软盘取出, 换上 PC-DOS2.1 盘, 用该版本的格式化文件, 来格式化硬盘:

A:FORMAT C:(回车)

这样就将硬盘的 10~100 柱面格式化了。

(4). 用 PC-DOS2.1 在 A 驱动器上重新启动系统, 然后运行该版本的 FDISK:

A&gt;FDISK(回车)

(A). 选择提示(3), 消除硬盘已建立的 DOS 分区。

(B). 重新建立 DOS 分区: 选择(1), 将硬盘从 0 柱面开始, 全盘建立 DOS 分区。

(5). 用 PC-DOS2.1 重新格式化硬盘, 并带上系统:

A&gt;FORMAT C:/S

(6). 打开 A 驱动器门, 关机重新启动(或热启动)系统, 即开始从 C 盘上正常启动系统。

(7). 将各种系统文件、字库、应用文件等分别拷入 C 盘。到此为止, 硬盘修复。系统文件和数据文件恢复。

我们的各种型号微机, 凡出现上述三种情况的故障, 均采用这套“分区、格式化方法”进行了有效地排除。

若硬盘仍需采用 DOS3.0 以上版本, 应将上述过程中的 PC-DOS2.1 改为 PC-DOS3.0(或更高)版本操作即可。

## 2. 排除上述 1、2、3 种故障的其他方法

(1) 若是 0 磁道上的介质没有真正损坏, 可以重新对硬盘初始化、分区、格式化。

首先调用初始化程序 DTCFMT 或 LOWFORM(若没有这两个文件, 可参考有关资料, 彩 DEBUG 编制), 然后调用 FDISK 程序, 创建 DOS 分区, 建立第一引导模块, 再调用格式化命令 FORMAT 带系统引入参数 S, 即完成了第二引导模块的建立及三个系统文件的安装。

(2) 若是 0 磁道上的某一区域确实损坏或整个 0 磁道损坏, 可以从结构上调整避开 0 磁道。在系统断电的状态下, 将硬盘步进电机 0 磁道检测器挡块上的两个螺钉拧下, 然后步进电机转轴反时针转动一个微小

距离。这就相当于步进电机的磁头小车相应地在磁盘中心方向位移一个微小距离。由于“00”磁道信号的产生与步进电机的相序有关, 转动步进电机转轴, 紧固挡块螺钉, 通电测试“00”磁道位置。

3. 排除磁头卡住的方法 我们在实践中, 采用如下方法, 有效地排除了硬盘磁头卡住的故障。

(1). 在关机的状态下, 在水平方向上, 轻轻敲打硬盘, 使磁头离开卡住的位置, 再加电启动时, 自行恢复正常。

(2). 采用 SHIPDISK 命令文件, 移动磁头

在 A 驱动器上启动系统后, 将具有 SHIPDISK 文件的软盘, 插入 A 驱动器内, 关门后打入:

A&gt;SHIPDISK(回车)

屏幕上出现:

SHUTDOWN COMPLETED-- OK TO TURN POWER OFF

这样, 可以关机。重新启动, 磁头恢复正常位置, 系统可以从 C 盘启动起来。

三、避免故障应采取的措施。硬盘, 也叫温盘(WINCHESTER DISK), 是一种高精密的磁表面存储设备。它采用全封闭结构, 其特点是磁头采用接触式起停。温盘处于工作状态时, 额定转速为 3600 转/分钟。这时磁头与盘片之间的间隙仅为 0.4~0.5μm。

对于这样精密的设备为了少出故障, 延长寿命, 长期可靠地工作, 建议采取以下措施: 1. 机器运输或搬运时, 切记将磁头从数据区移到“搬运区”, 以避免磁头撞击盘面造成数据破坏, 或磁头卡在某一位置上。可以用驱动器上的“锁”, 打开或锁定来完成; 也可由控制器发出指令自动完成。2. 硬盘虽然是密封结构, 但仍应注意通风, 以利于散热。仍需在比较清洁的环境中工作, 以免过多的尘埃阻塞硬盘的呼吸过滤器。3. 不紧靠强磁场, 以免破坏磁盘, 或破坏记录的数据。4. 每次关机前, 应运行 SHIPDISK 或 PARK 锁头程序, 使硬盘磁头离开数据区, 再切断电源, 这是保护 0 磁道的有效方法。5. 在系统启动过程中, 不能乱敲键, 乱往内存送命令, 以免破坏硬盘 0 磁道的引导文件。

## 参考资料

1. 《微型计算机汉字作系统 CC. DOS》, 钱培德编著
2. 《IBM-PC/XT 微机系统电路原理及维修技术》, 曲春皓、陈在勤等编
3. 《IBM-PC 及其兼容机常见故障维修 500 例(一)》, 《新浪潮》杂志编辑部
4. “硬盘不接受格式化命令怎么办?”, 汪锋, 《计算机世界》, 1989 年 8 月 10 日
5. “PC/XT 硬盘的低级格式化”, 《软件报》, 1987 年 9 月 16 日
6. “PC/XT 机硬 0 磁道保护的意义及方法”, 蓝涛、王方年, 《中国计算机用户》, 1988 年第 15 期



## 硬盘系统故障的维护方法

成都气象学院 任基

硬盘系统故障主要指硬盘不能自举,读写操作失败及系统不承认硬盘等现象。传统的解决这类故障的方法是重新格式化硬盘。这样固然可行,但费时费力,并且储存在硬盘的大量文件也会因无法备份而丢失。从DOS结构及其操作过程可知,DOS系统的加载主要是通过几个模块来完成的。其中任意环节出现故障,都将导致系统引导失败。如果不是错误信息“1701”,那么系统故障就可能不是硬件问题,而是由于两个引导模块或两个系统文件的某部分有错误或损坏造成的,即为软件问题,根据故障现象或信息提示找出原因是完全有可能进行修复的,不必通过格式化的方法。

系统加载,两个引导模块起着极为关键的作用,它们不仅承上启下,而且本身包含着极为重要的分区信息和BPB表。硬盘故障多数都与这两个模块的好坏有联系。因此,对于正常使用的微机,为了有备无患,应有这两个模块的备份文件。这也是做好系统维护的必要条件之一。需要指出的是,第二引导模块可在DEBUG下直接用L命令读出,但第一引导模块不属于任何分区,对它的读写只能借助于BIOS中的INT13功能调用。下面利用DEBUG程序给出备份第一引导模块的步骤和小汇编程序。

C>DEBUG

—A ;第一步:小汇编;将模块读入  
ES,BX为1000H开始的内存之中

MOV AH,02 ;02→AH为读,03为写功  
能

MOV AL,01 ;读入扇区数→AL  
MOV BX,1000 ;缓冲区首址→BX  
MOV CH,0 ;磁道号→CH  
MOV CL,01 ;扇区号→CL  
MOV DH,0 ;磁头号→DH  
MOV DL,80 ;磁盘号→DL(81H为D盘)  
INT 13 ;调用中断INT13  
INT 20 ;返回系统  
-G ;第二步:执行上述小汇编

程序,将备份文件存盘

—N BOOT1.DAT;置备份文件名  
—R CX 200 ;置文件长度  
—W 1000 ;将BOOT1.DAT存盘  
—Q

第二引导模块的备份工作与上面讨论的步骤一致,但在汇编程序中CL值应取分区表中DOS分区的第③项开始扇区数值,DH值应与第②项开始磁头号一致,不同的DOS版本,这两个值也可能不同。或者在进入DEBUG后直接键入:L1000201,即可代替汇编程序

将第二引导模块读入内存ES:BX为1000H开始处。下面分别介绍各类错误原因的修复方法。

1、第一引导模块损坏,其现象主要有开机后直接进入ROM BASIC或者显示第一引导模块的某种出错信息,甚至无任何出错信息就陷入死循环,用软盘启动后仍无法进入硬盘,显示信息“Invalid drive or filename”。原因主要的由于模块被破坏,分区信息无法查到。修复方法就是用好的第一引导模块备份文件重新写入0道1扇区。如果没有备份文件,可找一台正常工作的硬盘容量、分区情况都一样的机器,将其第一引导扇区的内容读出,再写入待修复机器的相应扇区之中(各种硬盘的第一引导模块执行代码基本一致,区别就在分区表上。如对自己机器分区情况熟悉,也可采用任何一种第一引导模块,将其分区表改为待恢复盘的分区内容,再写入该盘中)。具体步骤是:

第一步:将备份文件读入ES:BX为1000H开始处的内存之中

第二步:利用前面的小汇编程序,将AH,02改为AH,03,再执行即可恢复

2、第二引导模块损坏,其现象主要是当执行硬盘的读写操作命令时会出现死机现象。其原因是由于模块损坏,DOS找不到硬盘BPB表中的I/O参数或使用错误的参数信息。恢复方法仍然是利用正常的备份文件将原扇区重写。如没有也可以用其他机器的相同DOS版本的第二引导模块来代替,但BPB表要与待修复盘内容相同,否则就要修改。做为一个例子,下面给出DOS3.0以上版本的写入第二引导模块的小汇编程序。DOS3.0以下要作相应修改后使用。此程序也可用于读出第二引导模块,使用时只需修改AH值即可。

MOV DX,0180 ;DOS3.0为下DH为00  
MOV CX,0001 ;DOS3.0为下CL为02  
MOV BX,1000 ;缓冲区首址→BX  
MOV AX,0301 ;读出时用02→AH

3、DOS系统文件损坏:当系统显示第二引导模块的出错信息时,很可能是两个隐含的系统文件损坏。此时只需从软盘用SYS.COM程序将两个系统文件重新写入即可。如果执行SYS C:时系统显示“No room for system on destination desk”,则要进入C盘将两个隐含文件的只读属性去掉后再执行SYS.COM程序。

利用以上方法可对一些在引导扇区内加密的软件进行解密。如CCDOS2.13E或2.13F等是用其专门的激光加密盘在第二引导扇区设置工作参数,对照其引导扇区内容,并将自己的引导扇区做相应修改,就相当于在自己盘上设置了它的工作参数,使之为我所用。

## 2024 打印机怪癖故障二例

重庆 59193 部队科技科 齐毅

## 例 1

• **故障现象:**打印机加电后,字头小车始终不停地作往复运动,且不能联机打印。

• **故障分析与排除:**2024 打印机回车电机的动作是由辅 CPU6801(U<sub>5</sub>)直接控制的,其 P<sub>16</sub>、P<sub>17</sub>口在 P<sub>20</sub>、P<sub>47</sub>口的输入信号作用下,只控制其运动方向,而回车电机的启闭则是由 P<sub>23</sub>口上来自主 CPU6803(D2)的 ENQ 信号控制的;分析主 CPU 的命令时序,产生 ENQ 命令的因素很多,但与故障现象联系得最紧的是数据中断请求信号(DATINT)、与软件复位信号(IPSGN);这两个信号分别由主机送来的数据选通信号(DATA STROBE)及软件复位信号(IP PRIME)的负跳沿触发带复位端的双 D 触发器 #B<sub>3</sub>(74LS74)在其 9 脚及 5 脚产生(高电位有效)。见原理图。

用示波器测 DATA STROBE 及 IP PRIME 始终为高电位(无效状态)但测 #B<sub>3</sub> 的 9、5 端恒为高电位(有效

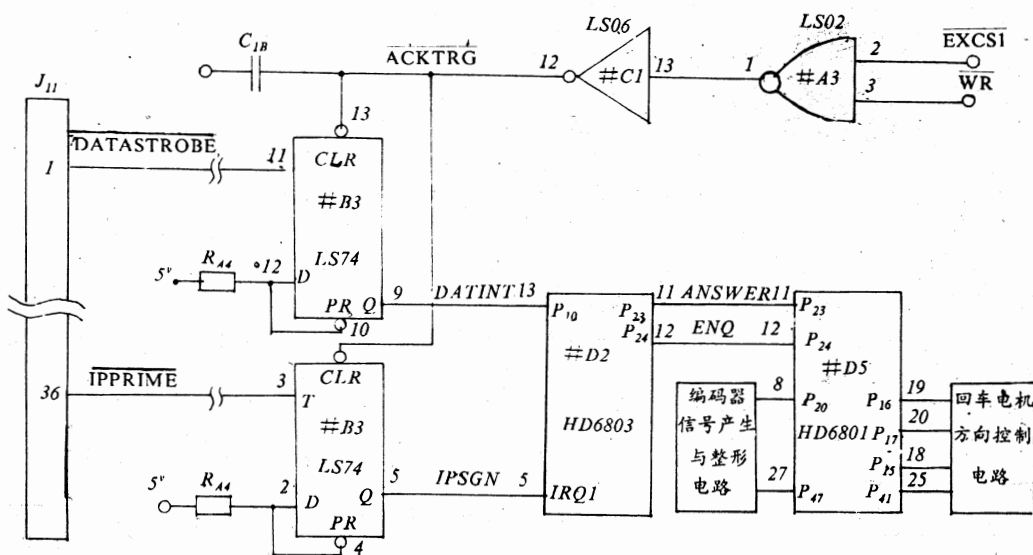
状态),这与工作原理不符,换下 #B<sub>3</sub>,故障消失。

## 例 2

• **故障现象:**打印机加电复位正常,主机发来复位命令后,字头小车不停地往复运动,且不能联机打印。

• **故障分析与排除:**如前分析,软件复位信号  $\overline{\text{IP PRIME}}$  有效的负跳沿触发 #B<sub>3</sub>(74LS74)的 T 端(3 脚),使其 Q 端(5 脚)产生有效高电位信号 IPSGN,通知主 CPU 作复位操作,并控制辅 CPU 使回车电机工作;若  $\overline{\text{IPSGN}}$  信号不撤除,回车电机将不停地工作,字车也就不断地来回运动; $\overline{\text{IPSGN}}$  信号是靠 #B<sub>3</sub> 的 1 脚上的 ACKTRG 信号撤除的,而 ACKTRG 信号则是由 EXCS1 与 WR4 LS 02)和 #C1(74 LS 06)在 #C1 的 12 脚上产生的,如原理图。

用示波器测 ACKTRG 信号,无有效的负脉冲,测 #C1 的 13 脚,信号正常,故判定 #C1 坏,更换后故障消失。



电原理图

# 微机辅助生成信息系统数据流图

西安石油学院 方明

**摘要** 本文介绍了一个微机辅助软件开发工具“数据流图生成软件”，它基于目前信息系统开发系统分析阶段使用较为广泛的结构化系统分析思想，使用一种用来描述系统功能的形式语言对目标系统的功能进行描述，由微机自动绘制出数据流图和产生数据字典中的基本内容。

**一、前言** 管理信息系统的系统分析阶段的技术方法有许多，使用较多的是结构化系统分析技术，该阶段产生的文档有：1)一套分层的数据流图；2)一本数据字典；3)一组小说明。在这之中，关键是一套分层的数据流图，数据字典和小说明则是对数据流图的补充说明。但是，目前数据流图的绘制多以手工为主，因此，为了提高系统分析的质量，确保系统分析的成功、降低系统分析的成本、增强系统分析的可维护性、避免手工方式下文档难于验证、检查、不规范、不一致和难管理，有必要使用计算机来绘制数据流图。

**二、数据流图的绘制过程** 数据流图有四种基本符号：正方形表示数据的源点或终点，也称外部实体；圆形代表变换数据的处理；开口矩形代表数据存贮；箭头表示数据流，作为新系统逻辑模型的数据流图主要是利用上述四种符号来准确地、在逻辑上描述新系统的功能、输入、输出和数据存贮等。

结构化系统分析中数据流图的绘制是将系统的各功能逐渐分解分层绘制数据流图。它的步骤是首先从问题描述中提取数据流图的四种成分；其次是画出系统的顶层数据流图；第三步就是画分层的数据流图；分层数据流图的画法是：1)确定父子关系，2)平衡性检查，3)给予图和加工圆圈编号，4)确定分解程度。为了实现上述绘制过程，在用计算机所设计的绘制数据流图软件系统中包含有这样一些功能：1)系统功能描述程序 2)功能描述处理程序，3)一致性检验程序，4)图形生成程序，5)交互式绘制数据流图程序。这五个功能为系统分析员提供了两种绘制数据流图的方式。一种是通过使用本系统所设计的一些形式语句即功能描述语句 FDL(Function Description Language)对系统功能进行描述而由计算机自动绘制出数据流图；另一种是通过使用键盘移动光标交互式绘制数据流图。

**1、使用 FDL 语句绘制数据流图** FDL 是本软件中设计的一些简单、易学的若干形式语句，主要包含以下几条语句：

- 1) PROCESS  $P_n$  <名称>
- 2) USES FLOW  $F_{n1}$  <数据流名称> FROM  $P_{n2}$
- 3) READ FLOW  $F_{n1}$  <数据流名称> FROM  $D_{n2}$  <数据存贮名称>
- 4) TO DERIVE FLOW  $F_{n1}$  <数据流名称> TO  $E_{n2}$  <外部实体名称>  
或 TO DERIVE FLOW  $F_{n1}$  <数据流名称> TO  $P_{n2}$
- 5) WRITE FLOW  $F_{n1}$  <数据流名称> TO  $D_{n2}$  <数据存贮名称>

上述各语句中的  $F_{n1}$ 、 $P_{n2}$ 、 $D_{n2}$ 、 $E_{n2}$  分别是数据流、过程、数据存贮和外部实体的编号，语句中的各元素的名称是可选项。使用这几条语句可以有效地描述一个数据流图。例如图 1 所示，数据流图用 FDL 描述为图 1。

```
PROCESS P1
USES FLOW F1 FROM E1
READ FLOW F2 FROM D1
TO DERIVE FLOW F3 TO P2
TO DERIVE FLOW F4 TO P3
PROCESS P2
USES FLOW F3 FROM P1
WRITE FLOW F6 TO D2
PROCESS P3
USES FLOW F4 FROM P1
READ FLOW F5 FROM D1
TO PERIVE FLOW F7 TO E2
```

使用功能描述处理程序对上述描述程序进行分析，得到该数据流图的邻接矩阵，由这个邻接矩阵来反映数据流图各元素之间的关系和各元素的编号，邻接矩阵中的值大于零，表示这两点之间有数据流，数据流的编号就等于该值。根据该矩阵和内容，由图形生成程序自动选点绘制出数据流图。绘图的方法是采用绘制网络图的方法，将数据流图中各元素看作是网络中的节点，数据流为网络中的工序。首先找出数据流图中所有无前序工序的节点，根据这些点的个数将显示器纵向划分为几个等分，然后再根据各节点的前后序关系横向的一个个画出。如图 1 所示。用计算机画出如图 2 所示。

**2、键盘交互式绘制数据流图** 交互式绘制数据流图的方法是由系统分析员选择进入绘制数据流图

状态,在这个状态下,分析员可以通过键盘上的光标键移动光标到一个选定的位置,按下所要画数据流图的符号所对应的字母键,该字母所对应的图形则显示出来,分析员再根据系统的显示信息输入对应符号的编码和名称。通过这样移动光标选点,可以画出该数据流图所包含的所有功能、实体和数据存贮。画好这些元素后,就可以按 L 键来画出数据流连接各元素,并输入各数据流的编码和名称。在交互式绘制数据流图方式下,可以对数据流图中的内容进行增加,删除和移动。

上述两种方式下所画的数据流图可以互相在不同的程序中被调用和使用。也可以对每一层数据流图进行进一步的分解。同时,在图形生成程序中,可以对所画的数据流图进行放大、缩小、平移和两层数据流图的合并,也可以同时显示两张数据流图,可以直接通过键盘命令对数据流图进行修改,也可以在图形显示状态下,调出该层数据流图的 FDL 描述程序进行修改后,再经由计算机处理重新绘制出修改后的数据流图。另外,数据流图还可以输出在绘图仪上画出。

**三、结束语** 本软件对于改变软件开发和维护文档时间长、速度慢、不规范、难查询和修改复杂等状况可以起到很好的辅助作用。它为系统分析员提供了许多附图:

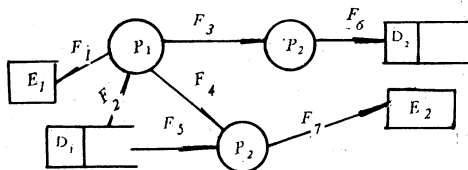


图 1

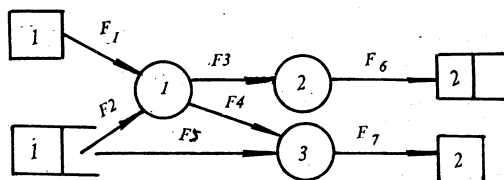


图 2

多好处,这包括:

- 1、通过标准化改进了文档图形的质量。
- 2、需要的图形可以随时显示在终端上,不需要在手册中查找。
- 3、图形便于增加、删除和更改。
- 4、具有冗余、矛盾等多种检验方式,保证了分层数据流图具有完整性、一致性和无二义性。
- 5、同其他类似的自动化图形工具相比,语言简炼、容易掌握、人工干预少和绘图方式灵活多样。

本软件具有层次性、信息共享性、可跟踪性和可扩充性等特点。同时,为了进一步减轻分析员的工作量,在进一步完善本系统的基础上,有必要开发图形生成工具的专家系统,使整个绘图过程在几乎不需人工干预的情况下完全自动化。

需软盘者,请同作者联系;需源程序者,同本刊联系。

#### 参考文献

- 1 曹锦芳,“信息系统分析与设计”,北京航空学院出版社,1987
- 2 Roger S. Pressman, “SOFTWARE ENGINEERING”, McGraw-Hill Book Company, 1987

## 常见 CPU 芯片数据

CPU 芯片型号	寄存器宽度(位)	数据总线宽度(位)	地址总线宽度(位)	物理寻址极限
8080	8	8	16	64KB
6502	8	8	16	64KB
Z80	8	8	20	64KB
8088	16	8	20	1MB
8086	16	16	20	1MB
65816	16	8	24	16MB
80286	16	16	24	16MB
68020	32	32	32	4GB
80386	32	32	32	4GB

(湖北省公路科研所 张红供稿)

# PC-XT 系统板维修一例

湖南省电子研究所 胡红

**故障现象:**开机无任何反应。

**分析与检修:**首先排除了电源、复位、时钟损坏的可能。问题显然是系统板损坏。开机,测 8088 的三根状态线  $\overline{S_0}\overline{S_1}\overline{S_2}$ ,发现 8088 总是在取指令和写 I/O 口,即 CPU 处于死循环。由 BIOS 中自诊程序可知,只有在基本 16K RAM 有问题时,才会使得 CPU 处于死循环,其它致命性的错误均会使 CPU 处于 HALT 状态。此时 8237 的 25 脚(DACKO)正常,说明在 RAM 检查以前的程序操作正常,由此确认为基本 16K RAM 故障。在这种情形,根据 8255 的 A 口来进行故障定位一般是行不通的。

我们知道,存储器部份由 RAM 存储体,地址译码器  $U_{44}$ (24S10)、行/列地址开关  $U_{39}$ 、 $U_{40}$ (LS158)、行列选通信号选择电路  $U_{55}$ 、 $U_{42}$ (LS138)、刷新电路、奇偶校验位产生/校验电路  $U_{20}$ (S280)、 $U_{86}$ (LS74)以及 RAM 数据收发器  $U_9$ (LS245)等七部份组成。在 CPU 总线周期由 CPU 对存储器部份进行读写操作,而在刷新周期由 8237 通道 0 控制存储器的刷新。

开机直接测量 BANK0 的任何一片 RAM 的出脚

波形,没有发现恒定电平和浮空现象,行刷新正常,但其 15 脚( $\overline{CAS}$ )有波形。由于 CPU 在死循环时,不对 RAM 进行任何操作,只有 8237 对它进行周期为 15.2 $\mu$ S 能有  $\overline{RAS}$ 信号而不应有  $\overline{CAS}$ 信号。见系统板图之 3。 $\overline{CAS}$ 由  $U_{42}$ 产生,它的工作条件:1)非刷新操作,高电平的  $\overline{ACKO BRD}$ 送至  $G_1$ 端;2)非刷新操作期间访问系统板上的 RAM,  $U_{24-6}$ 使  $G_{2A}$ 变低;3)存储器读写命令使  $\overline{G_{2B}}$ 成为低电平。只有当这三个条件都满足时, $U_{42}$ 才产生相应的  $\overline{CAS}$ 信号。由于本例中有  $\overline{CAS}$ 信号,故  $U_{42}$ 本身一般并无问题,只是三个控制端中有问题,追踪测量发现只有  $\overline{G_{2B}}$ 端可疑,再顺势前查发现 TD1 输出信号不对,如图。可见 TD1 输出特性变坏,关不死  $U_{42}$ ,并且使得 ADDRSEL 信号也不对。由于 TD1 不常见,可以用门电路搭。见下图:其中的门电路可以用 74LS04 或 74LS14。

其安装位置就用 TD1 的原位置,只不过需将不用的脚折断,才能插入。见下图。

将插座上打“×”的出脚折断,将 IC 上打虚线的脚短接。换掉后,故障排除。

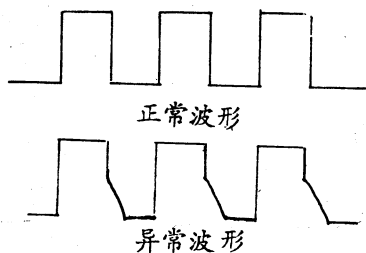


图 1.

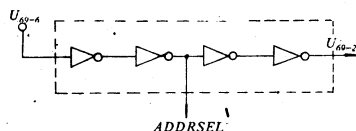


图 2.

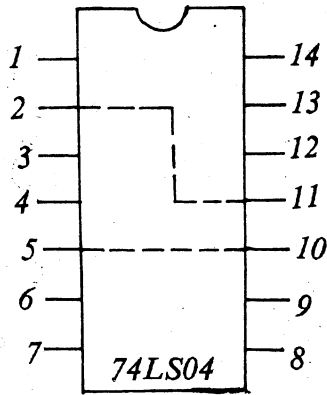
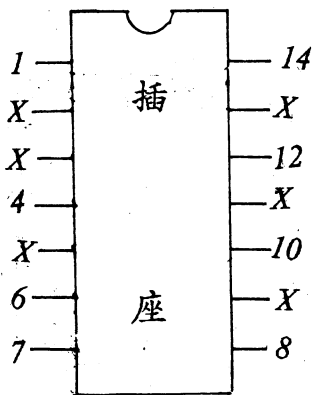


图 3.

# 友好的多窗口磁盘菜单操作系统

空军第一航空技术专科学校计算机教研室 李志伟

**摘要:**本文介绍了最新设计的一种多窗口磁盘菜单操作系统软件,它集成了常用的十几种 DOS 内部命令于一窗口,全自动地实现了各种 DOS 操作,并可随时显示磁盘文件状态,具备时钟、日期显示,磁盘及内存使用情况等辅助功能,是一种多窗口环境式、点键操作的系统软件,性能优异,坚固性好。在我校,本软件已为 95% 以上的 PC 机用户所使用,倍受欢迎。

## 一、软件简介

### (一)开发背景

传统的 DOS 状态一般都是单色操作,而为五光十色的应用程序“惯坏”了的 PC 机用户,对于这种单色操作早已感到淡而无味,而且要想熟练地完成各种操作,必须熟记那一条条枯燥无味的 DOS 命令,且要逐个地键入操作,用户好比在不发达地区作艰难的旅行,不断地倒车换船,对用户,尤其是 PC 机的初学者感到非常不便。随着 PC 机的普及和应用的日益深入,对于能满足广大用户五颜六色心理需要和能直接执行 DOS 操作的直达快车的呼声越来越高,那么本软件便满足了广大 PC 机用户的这种特殊需要,并且提供了丰富多彩的辅助功能,不但方便了用户,而且使用户一览这种新型软件的设计风采。

### (二)使用环境

1. 本软件是在 Turbo Pascal 4.0 系统上开发成功的,适用于 IBM PC/XT 及其兼容机,用于 DOS 及其它各种磁盘操作,并可实现一些辅助功能,操作简单,方便灵活。

2. 不但适用于对西文软件的操作,而且适用于对中文软件的操作。

### (三)磁盘文件

本软件共有二个磁盘文件:

DOS-WIND.EXE——运行文件

DOS-WIND.DOC——使用说明文件

用户可在 DOS 下键入文件名 DOS-WIND.EXE 或启动机器后由 AUTOEXEC.BAT 直接进入该系统。

### (四)软件特点

本软件吸取了最近几年推出的新版 PC-Tool 及 Turbo 系列的 Pascal、Prolog、C 等系统的优点,借鉴了其优秀的文件装载窗口及其它窗口功能,并优于这些软件的类似界面,提供了一个多窗口的下拉菜单,点键操作,使用方便,具有友好的用户性能。

## 二、软件功能

本软件由三个大的不动功能窗口(有些大窗口又由多个小窗口组成)和若干个活动执行窗口所组成,各种操作都在对应的窗口内进行。

### (一)DOS 功能窗口

位于屏幕顶部的是一个不卷动的功能菜单窗口,它预集成了常用的 14 种 DOS 内部命令和一个直接运行文件(仅对 COM、EXE、BAT 文件)的 Run 命令,使用者不必记住一系列繁琐的 DOS 命令便可在该窗口内执行各种 DOS 操作,甚至直接运行文件。用户可用上下左右等光标控制键移动光条,选中某项后按<Enter>,则执行相应的命令。

对于文件操作,则系统首先开辟一小窗口,要求用户键入路径名或掩码(系统默认为当前目录),键入后系统即将指定目录下的文件列于下拉窗口中,用户可在该窗口内利用所有的光标控制键控制光条搜索指定的文件,搜索到后按<En-

ter>即可对该文件进行相应的操作——删除、改名、拷贝、显示、打印、运行等。这种窗口功能有以下几个优点:

1. 用户不需键入文件名。
2. 用户不知道目录中有何文件时,也可以先浏览后选择。
3. 不会出现“File not found”之类的错误,因目录中列出的文件都是存在的。
4. 可以列出目录中的隐含文件、系统文件和只读文件。
5. 未选中文件为蓝底白字,选中文件为紫底黄字,非常悦目好看。
6. 各窗口中显示的文件名具有相同的宽度,显示格式比 Turbo 软件整齐。

若选点与文件操作无关的 DOS 命令,如建立、删除、改变、打印目录,以及设置日期、时间,显示 DOS 版本、驱动器卷标等,均将另辟一窗口,显示相应的信息,提示并执行对应的操作。

### (二)文件状态操作窗口

屏底两行也是一个不卷动、不消失的功能窗口,最底行用于显示系统名称、版本等信息,另一行为

F1——File State Esc——Exit

Esc:用户在任一操作窗口中按该键都将退出该窗口,回到上一级窗口,以至退出该系统。

F1:按该键则可在一窗口中显示出特定目录下所有文件的内部情况:文件名字、大小,建立日期、时间以及文件属性等,同样可用各种光标控制键快速查找到某一文件,其颜色搭配比 PC-Tool 悦目,显示状态比 PC-Tool 整齐,检索速度比 PC-Tool 快——优于 PC-Tool 的相应功能。

### (三)辅助功能窗口

位于屏幕右侧的窗口,由几个小的功能窗口所组成,主要用于显示系统的当前日期、时间、操作目录、磁盘及内存使用情况等,其显示内容可根据具体的操作而发生相应的变化。

## 三、设计技术

### (一)采用多窗口技术

软件设计采用了多窗口设计技术,使各功能模块分别集成于对应的窗口之内,各种操作都在窗口内进行,且用光条选点文件,执行对应的操作,改变了传统的手工 DOS 操作,实现了 DOS 操作的自动化。另外,软件设计还采用了下拉窗口和窗口覆盖等技术,各种窗口显示的颜色配置都新奇整齐,光怪陆离,给用户开辟一个崭新的天地。

### (二)采用单元设计技术

为了提高软件开发效率,提高软件开发质量,软件设计采用单元设计思想,并且充分利用了 Turbo Pascal 4.0 提供的单元功能,将程序中经常用到的比较复杂的功能模块设计到相应的单元中,不但增加了程序的可读性,缩短了主程序,更加体现了结构化程序设计思想,而且缩短了软件的开发周期,提高了软件的开发质量。

需要软盘者,请寄 18 元工本费至本刊读者服务部函索。

### 主要参考文献

- 1.《Turbo Pascal Owner's Handbook》Borland International Corporation, American
- 2.“一个友好的文件装载窗口”,唐常杰,“计算机应用研究”,No. 2, 1990

# dBASE III PLUS 中 MEMO 字段的格式化打印方法

吉林工学院计算机系 许建潮

MEMO 型字段是 DBASEPLUS 中非常有用的一种字段,在信息管理系统中,许多信息(例如图书管理中的摘要、人事档案管理中的简历、事故档案中的事故发生经过及原因等内容)都需用 MEMO 字段的内容只能用非格式化显示语句?、??、LIST 和 DISPLAY 等显示出来,不能用格式化显示语句显示、打印。这为按某一特定的表格形式显示 MEMO 字段的内容带来困难。作者在设计某应用软件的过程中,采用先将 MEMO 字段的内容转化成文本文件(.TXT)的内容,再将文本文件的内容复制到一个数据库文件(.DBF)中的字符型字段内,然后用打印字符型字段的内容的方法,较好地解决了 MEMO 字段的格式化打印问题,下面用一个例子来说明这一方法。

设含有一个, MEMO 型字段的数据库的名字是 AA. DBF, MEMO 型字段的字段名是 A1。用于打印的目标数据库名字是 BB. DBF, 只含一个字符型字段 B1。为了将一个 MEMO 字段的内容变成 BB 库中的多个记录,以便用 @ 打印,先要设法将 MEMO 字段的内容转变成 .TXT 文件的内容。但数据库文件复制命令

COPY TO<文件名>[<范围>][FIELDS<字段名表>][FOR/WHILE<条件>][SDF/DELIMITED][WITH<分隔符>]不能将 MEMO 型字段中的内容转换进 .TXT 文件,为此作者借用建立记录处理活动的文本输出文件的方法,先将一个 MEMO 型字段的内容通过屏幕显示记录到文本输出文件(设为 HH. TXT)中,再将 HH. TXT 中的内容转入 BB. DBF 中,方法如下:

设要打印 AA 中第 i 个记录的 MEMO 型字段 A1 中的内容。

...

sele I

use AA

Go i

set alternate to HH

set alternate on

display A1 off

close alternate

use BB

append from HH. TXT sdf

经过上面的处理, HH. TXT 中的内容被全部装入了 BB. DBF 中。BB...DBF 中的内容为:

记录号

B1

```

1
2      .display A1 off
3 * * ..... *
4 * * ..... *
5
n-1 * * ..... *
      n . close alternate
  
```

其中第 1、3 记录是空记录;第 2 个记录的内容是显示命令;最后一个记录的内容是关闭记录处理活动的文本输出文件的命令;其它记录中的 \* \* ..... \* 就是 AA 数据库第 i 个记录的 A1 字段的内容,用 @ 命令按表格要求打印 BB 数据库的第 4 ~ n-1 记录的内容,即可实现 MEMO 字段内容的格式化打印。

几点说明:

## 1、初始化设置

(1)在上述处理之前,先要按输出表格的宽度用 SET MEMOWIDTH TO <数值表达式> 命令设置 MEMO 字段的输出域宽,使得用 display 显示 MEMO 型字段内容时,能按小于等于表格要求的宽度显示,进而使 BB 数据库中每一记录对应输出表格的一行。B1 字段的宽度一定要大于等于 MEMO 字段的显示宽度,否则在信息由 HH. TXT 中转入 BB. DBF 的过程中,每一行内容将从左至右按 B1 字段的宽度截取后转 BB. DBF 中,每行其余的内容都将丢失。另外 B1 字段的宽度还必须小于等于输出表格的宽度。

在 DBASE 中,由于没有 SET MEMOWIDTH TO 命令, MEMO 字段的输出域宽隐含值是 50。所以输出表格的宽度不能小于 50。

(2)在程序开始部分要用 SET HEADING OFF 命令使得 display 操作中对被显示字段 A1 不显示栏标题,以减少 BB. DBF 中的无用信息。

## 2. display 命令要用 off 选项,避免显示记录号。

3. 第打印完一个记录的 MEMO 字段的内容,在要打印下一个记录的 MEMO 字段的内容之前,一定要用 ERASE HH. TXT 命令先除 HH. TXT 文件,用 ZAP 清除 BB. DBF 中的内容,否则在显示其它记录的 MEMO 字段时,将会使要显示的内容与已显示过的内容混在一起。

4. 若有多个 MEMO 字段的内容要打印,则要建立对应个数的只含一个字符型字段的用于打印的目标数据库。



## PC机软盘故障分析与维修

西北工业大学 牛海发

目前,广泛使用 PC/XT 和 PC/286 系列机及各种兼容机,系统都配有 A:和 B:软盘驱动器。由于用户使用不当,有的可能出现违犯操规程等,还有其它原因:诸如机房没有配备必要的电源设备、灰尘大而使磁头脏等。笔者的体会是:上述各种原因主要是,而在微机工作的近两三年内,真正的芯片故障只占少数。现就笔者几年来使用维修 PC 机 A:B:软驱中总结的经验体会介绍给同仁。文中最后列举了两个实例。

1、磁头偏移会导致数据读/写出错。一般情况下,应对驱动器的机械传动部分及磁头定位系统的各种参数进行检测后对钢带的调整使磁头定位在正确的磁道上,排除了故障。

2、当启动软盘工作时,正常是驱动器面板上指示灯点亮。否则应怀疑并检查驱动器选中信号是否有故障。

3、启动软盘,屏幕不出现提示符。可能是 DMA 地址出错,因软盘引导块装入到错误的 RAM 区,导致执行引导块程序错误,系统无法进入操作系统。也有可能是由于 74LS175 芯片失效,使 DMA 请求信号不能正确地发出而引起这样的故障。若是芯片失效,一般更换该芯片,故障就可排除。

4、软盘启停不灵。启动时间是一个重要的参数,若此参数控制不准就会导致启动控制不灵和定位出错。造成启停控制不灵和原因是:面板开关不灵;微动开关装置不准或接触不良;启动线路有故障;电机及伺服机构有故障;电源有故障。一般从上述几方面检查检修,故障可排除。

5、不能读/写或读写错误。软盘存储器的读写错误,主要表现为:读出/写入的信息出错或不能读写;或读不出内容或读出的内容不对。造成读写错误的主要原因是:磁层薄厚不匀;磁层划伤剥落;磁头太脏或间隙过大;接触不良;读写电路有故障或元件失效;电源电压或脉冲电流不正常;外界干扰使存储器的信息状态发生变化等。应着手从上述几方面检修,就可排除

故障。这类故障在 A:B:软盘中比率较高。

下面列举两个 PC 机 A:B:盘不能工作的故障分析检修实例:

例 1、有一台 PC XT 微机,出现不能读/写故障。

故障现象:开机自检通过后,不启动 A 盘而直接进入 C:盘 A:B:两盘都不能工作。当使用 DIR A:命令时,屏幕显示信息为:

Not ready error reading drive A

Abort, Retry, Fail?

故障分析与检修:这个故障有可能是磁头太脏,或信号线没有连接好等原因,当对磁头清洗后,并把软驱的各信号线重新按插好后,故障并没排除。经检查推断是  $U_{27}$ (73LS175)芯片损坏。因为  $U_{27}$  芯片本身构成一个延时电路,当软盘控制器(FDC)芯片  $\mu PD765$  请求使用 DMA 时,其输出  $DRQ=1$ 。同时要把这个信号送到  $U_{27}$  的 D 端,该信号经过几级延时后由  $Q_3$  端( $U_{27}$  第 15 脚)送到  $U_7$ (Mc3487)的其中一个输入端引脚 7,此时若数据寄存器  $U_{17}$  的  $Q_3$  端输出一个高电平,允许其输出端向系统发出 DMA 请求信号  $DRQ_2$  和中断请求信号  $IRQ_6$ ,由于  $U_{27}$  的失效,所以 DMA 请求信号不能正确地发出而引起了故障。更换  $U_{27}$  芯片,故障排除。

例 2、PC286 微机不识别 A:B:软盘故障。

故障现象:开机后系统认为没有配接 A:和 B:软磁般驱动器。

故障分析与排除:从该机的使用者了解到,该台 PC 286 机是用于学生有关机械设备信息分析处理实验,扩充槽装有专用卡,专用软件必须用 A:盘启动。笔者通过分析与观察,基本断定是磁头太脏和接触不良等原因引起的故障。经对磁头的清洗,故障并没排除。后经对各硬卡及接口用棉球沾工来酒精全部清洗完,待晾干、重新装好之后,加电开机启动, A:和 B:软盘工作均正常。专用软件从 A:盘启动,工作也正常。故障被排除。

# 长城 GW-0520DH 微机硬盘驱动器控制卡维修一例

四川省电子计算机应用研究中心 黄 岗

## 一、故障现象:

1.)开机后系统 640K 内存测试正常,硬盘无法自动引导 DOS 操作系统,屏幕显示如下:

```
640K  内存测试正常
Hard Disk 1:612 CYLS 4 HDADS
Emulatd CGA Mode ? (Y)↵ (回车)
```

Non—System disk or disk error Replace and strike any key when ready

然后死机。

2.)这时在 A:驱动器中插入 DOS 系统软盘能正常启动,但是无法进入 C:盘工作。

例 如: A>DIR C:

```
Seek error reading drive C
Abort, Retry, Ignore?
```

## 二、故障分析:

1.)打开机箱,拨下硬盘扁平连接电缆插头 J1,屏幕显示如下:

```
640K 内存测试正常
1701 (04)
0 Hard Disk
Emulate CGA Mode ? (Y)↵ (回车)
```

死机,A:驱动器指示灯常亮。

2.)同时拨下硬盘扁平连接电缆插头 J1 和 J2,屏幕显示仍然如下:

```
640K 内存测试正常
1701 (04)
0 Hard Disk
Emulate CGA Mode ? (Y)↵ (回车)
```

死机,A:驱动器指示灯常亮。

3.)拨下“硬盘驱动器控制卡”,屏幕显示如下:

```
640K 内存测试正常
Emulate CGA Mode ? (Y)↵ (回车)
```

DISK BOOT FAILURE, INSTEM  
SYSTEM DISK AND PRESS ENTER

4.)用更换相同型号的“硬盘驱动器控制卡”的比较法测试,硬盘能正常工作。经分析,硬盘工作正常,很显然,故障可孤立地确定出在“硬盘驱动器控制卡”上。

## 三、故障测试:

1.)测试“硬盘驱动器控制卡”上(+5V)供电电源正常。

2.)拨下“硬盘驱动器控制卡”,肉眼观察,电路板上所有元、器件无明显因过热或烧蚀现象造成的损坏;镀金插头、座完好无损。

3.)断开硬盘驱动器,带电测试“硬盘驱动器控制卡”上 J1 和 J2 的各驱动器控制逻辑信号,发现 J1—P34 脚(Direction/方向选择信号)始终为“高电平”;继续测试,发现与之相连接的芯片 7406(六倒相缓冲器/驱动器/OC,高压输出)的第(IV)组非门损坏;即:第 P9 脚输入正常,而第 P8 脚的输出因芯片损坏而始终置为“高电平”状态,人工输入脉冲触发信号,仍无反相输出信号。

## 四、故障排除:

由此可见,7406 芯片的损坏造成经第八脚反相输出后的“Direction”信号始终置为高电平状态,导致硬盘读/写磁头小车不能复位归“零”,无法正常地初始化,从硬盘上的第一扇区里引导 DOS 引导记录,造成硬盘既无法引导又不能使用的故障现象。

更换 7406 芯片,经两个月的实际使用,硬盘和整个系统工作一切正常。

## 五、注意事项:

1.)在进行维修之前,请先将 J1 插头的 P1、P3、……P33 等奇数引脚的“GND/共地线”与 P2、P4、……P34 等偶数引脚的各控制信号线短路,以避免损坏接口控制芯片。

2.)在使用普通的电烙铁和电热吸锡器时,一旦加热后,请务必从电源插座上拨下插头,然后再进行元、器件的更换操作,以避免因“静电感应”干扰而造成“硬盘驱动器控制卡”上的 MOS 大规模集成电路控制芯片的损坏。

## 吉林省实业技术开发公司

隆重推出:①让利最新 IBM 软件(含随盘资料)+②免费挂号邮寄  
+③免费赠送软件+④赠免费随盘软件目录

最新防病毒软件 SCANV62 系列,查解当前国际上发现的所有 86 种病毒,各中英文随盘资料,国内独家经营,价格全国最低。

T1:SCAN V62;其它公司的旧版本如 V52 与 V57 等售价 200—500 元,我处为最高版本只售 45 元,检测 86 种病毒,并可删除感染文件。

T2:VSHIELD V62(50 元):长驻内存的病毒监视软件,能防止带病毒软件被调入内存运行,常于 AUTOEXEC. BAT 文件中运行。

T3:NETSCAN V62(80 元):检测工作站和微机网络中的 86 种病毒感染,并可删除感染文件。

T4:CLEAN V62(60 元):清除 SCAN V62 发现的各种病毒,并可恢复原文件的功能。

T5:批文件工具集(2 张盘 40 元):含扩展批语言等几十种工具程序,如等待,音响,循环,搜索,选择等,有用法说明及教学演示,使批处理成为功能强大的批语言。

T6:一百个 BASIC 游戏(5 盘 80 元):既可用于学编程技巧或二次开发,又可用于娱乐,每个游戏不到 1 元钱,共上百万字节,种类繁多,丰富多彩。

T7:GRASP V1.10(50 元):编制带动画效果的电脑演示,带各种工具程序,随盘操作手册,演示和图形样品。CAP.COM 可储存屏幕图象建立图库,支持 CGA、EGA 和单显。

T8:PC-KEY-DRAW V3.33(2 盘 45 元):CAD+绘画软件,可用于画框图,地图,草图及建筑和电路板设计等。可建图库,支持键盘、鼠标,可画点、线、框、弧、椭圆、阴影、文字、融合、缩放、对映体、格栅等,有动画功能,各种色彩,支持 CGA,EGA,和 Epson 等类打印机。有手册。

T9:PC DESK V3.2(25 元):完整的办公桌系统,随屏提示,易学易用,操作迅速。

包括计算器、时钟、闹钟,日历,记事簿,电话号码本及拨号机,日历和日程表,打字机,打印机控制,DOS 命令等。

T10:Still Rever Shell V2.36(20 元):强大迅速的文件子目录管理工具。可减少 80% 的 DOS 工作量,包括选择操作,拷贝,移动,删除,寻找,修改,列出文件和子目录。可储 20 个先前用过的 DOS 指令供调用。隐藏文件,备份子目录,及用户再定义键,菜单色彩调整等。

T11:PKARCHIVES V3.5(20 元):文件压缩工具软件,可将文件压缩 30—60%,以便保存和传递。并可加密码、注释、及形成自动原式 EXE 文件等。

T12:FORMATQM(20 元):快速格式化 360K,720K,1.2M 软盘,自动计数,双盘工作。

T13:COPYQM V1.6(20 元):快速拷贝盘,格式化、拷贝、检验一次完成。自动计数,双盘工作,拷贝一张盘不到一分钟。

T14:Automenu (20 元):硬盘菜单,美国著名硬盘目录建立管理软件,只按一个键即可执行各种程序,批文件及 DOS 命令。支持网络和鼠标,有密码保护。易安易用。有随盘手册、随屏提示。

T15:SUM DOS (20 元):包括计算器,笔记本,文字处理,磁盘工具,阅读文本文件,通讯终端、电话拨号机,ASCII 码表,屏幕隐舍、储存,日历等等。

每购满百元者,免费赠以下一种软件,多买多赠,软件功能见《计算机世界》月刊 1990 年第 4 期 59 页。

1. DIR MAGIC; 2. PC-LOCK; 3. 文件保护工具盘; 4. PC-BOSS 5. DOS 提示。若购以上软件者,每购软件满 150 元者,可进一步优惠。

请汇款至:130051 长春市上海路 21—3 号,201 信箱 李柏 收

或银行转账至:吉林省实验技术开发公司,帐号:082—261041

开户行:长春市建行朝阳支行科技分理处。请务必注明欲购软件代号字样。

款到一周内发货。

# APPLICATION RESEARCH OF COMPUTERS

Vol. 8 No. 2 (Total 40)

## CONTENTS

### SOFTWARE

The Algorithm for Appraise by Specialist and The Supporting System by Software .....	Liu Dingjun and others (1)
Approaching about The Variable Model for Calculating Markets of Microcomputers .....	Zhang Hongming (5)
Principle and Technology for Package of Data Processing .....	Gao Weimin (7)
MOTHER—The Generating System of MIS .....	Zhang Xiaoqiang and Others(12)
Synthetical Applying with Varied Program Language .....	Zhong Tinjiao (14)
Displaying Program for Generalized Text Files .....	Wang Xiaowu (16)
Generalized Designing Environment for Internal Storage Staying Program .....	Xiao Junliang (17)
Designing and Realizing for Automatic Printing Program of Generalized Report .....	Lai Xiaoqiang (20)
Rapid Modify to chinese Base .....	Guo Jizhan (23)
The Software with Plotting Flowchart on PC—1500 Computer .....	Yang Dashun and Others(26)
Introducing A Plotting Program of Printer .....	Bie Shean and Others(29)
Designing and Realizing for Tools Software of Menu—MENUTOOLS .....	Wang Daoshun and Others (31)
The Method with Push—down Menu on dBASE III .....	Jiang Zhenbing(35)
Simplifying to Fixed Solution of Eight Emprss Problem .....	Liao Dayong(11)
Opposite Flip Encryption .....	He Qicai(36)
Let Playing Music by Computer .....	Pan Zuoqiu(37)
Friend and Many Aperture Operating System with Disk Menu .....	Li Zhiwei(59)
Auxiliary Generated Data Flowchart of Information System by Microcomputers .....	Fang Ming(56)
A Samll—scale Displaying Program .....	KeChangXian(16)
A Small—scale Utility Program .....	Huang Huan Ru(30)
A Uneneryption Method With "P" .....	ChenQinZhang(49)
The Method of printing Memofields by Formated way in dBASE III plus .....	XiuJianChao(60)

### HARDWARE

Designing Hardware for Controlled System by Single Chip Processor .....	Liu Xi and Others(40)
---	-----------------------

### SYSTEM

Designing A Practical System of speech Analysis and Compound .....	Liu Ronghe(45)
--	----------------

### MAINTENANCE

The Searching—error Technology and Dictionary of Algorithmic language .....	Niu HaiFa(42)
Once More Discuss of Maintenance Hard—disk .....	Li Jianhua(52)
The Methods of Maintenance for System Fault on Rigid Disk .....	Ren Ji(54)
An Example about Maintenance System Control Panel on PC—XT .....	Hu Hong(58)
Tow Examples with Overhaul Bug on 2024 printer .....	Qi Yi(55)
The Tinfoil Paper of Cigarette can Work Wonders .....	Zhang Weizhong(51)
The Mechanical Justify for Magnetic—tape Handler .....	Chen lanying(10)
Analysis & maintenace of Diskette Faults on Pc .....	NiuHaifa(61)
An Example of maintenance for cont roler of Rigid DiskDrive on Gw0520DH .....	Huang Gang(62)

### INFORMATION

The Pocket—size Computers be Faced with Problem and It' s Countermeasure .....	Cao Laifa(13)
News .....	(4,13,19,28,34,44,50,63)