

# 计算机时代

COMPUTER ERA



1991

2



# 计算机软件专业技术资格(水平)考试情况

浙江省软件技术开发中心

虞国秋 陈敏玲

通过“资格”考试取得专业技术资格,是深化技术职称改革,完善专业技术职务聘任制的一项重要措施。“水平”考试则是为了追踪“国际水平”,适应国际间人才交流与合作的有效办法。考试实行全国统一组织、统一大纲,统一试题、统一评分标准。“资格”与“水平”考试实行统一组织,分开进行的办法。

1990年,浙江省组织了首次资格考试和第4次水平考试,报名人数之多,应考率之高,都是历年来所没有的。考试结果如下表

级别	报名人数	应考人数	应考率	及格人数		及格率	
				资格	水平	资格	水平
程序员	694	544	78.39%	234	108	43.0%	19.9%
高级程序员	85	74	87.10%	24	1	32.4%	1.4%
系统分析员	2	2	100%	/	1		50%

根据国家有关文件规定,资格考试合格者授予人事部统一印制的《专业技术资格证书》,全国有效。水平考试合格者授予中国计算机软件专业技术资格(水平)考试委员会统一印制的《计算机软件专业水平证书》。在1990年以前通过程序员级和高级程序员级水

平考试的合格人员,对其中符合《工程技术人员职务试行条例》所规定的学历、资历者,可授予相应的专业技术资格证书。

1991年,计算机软件专业技术资格(水平)考试有新规定:获得计算机软件专业技术资格必须通过全国统一组织的考试。今后对中级职称以下(含中级)的计算机软件人员不再进行专业技术职务资格的评审工作。

资格考试级别分为初级程序员(相当于技术员)、程序员(相当于助理工程师)、高级程序员(相当于工程师)。

水平考试级别分为程序员、高级程序员和系统分析员。程序员、高级程序员级水平考试合格者同时具有相应级别专业技术资格。通过系统分析员考试的人员,其水平可作为评聘软件高级工程师职务的专业技术依据。

根据国家规定,由各省、市、自治区、电子办和职改办共同组织地方的考试实施机构。浙江省计算机软件专业技术资格(水平)考试实施办设在浙江省软件技术开发中心。今年的考试日期仍为9月的第一个星期日,报名日期在6月份。

## 计 算 机 时 代

(季 刊)

1991年 第2期

(总第30期)

编辑出版 《计算机时代》编辑部

通讯地址 杭州环城西路新5号

邮政编码 310006

主办单位 浙江省计算技术研究所

浙江省计算机学会

印 刷 余杭县文教印刷厂

总 发 行 杭州市邮局报刊发行科

订 购 处 全国各地邮电局

国内统一刊号 CN 33-1094

邮 发 代 号 32-81

出 刊 日 期 1991年5月20日

每 期 定 价 1.00元



# 计算机时代目次

1991年第2期

(总第30期)

• ~~~~~ •	洁霉素发酵罐二级计算机控制系统..... ( 1 )	
§ 开发与应用 §		徐川育 王道琴 张方强
• ~~~~~ •	酒精蒸馏过程计算机控制系统..... ( 4 )	
		蒋慧丽 蔡亦凡等
	微机应用于监护病房的探索..... ( 6 )	
		姚永金 王玉炜
	计算机医务管理系统..... ( 8 )	
		张 希 章寿发 相承基
	逻辑驱动器的建立及使用..... ( 12 )	
		霍云峰
• ~~~~~ •	分布式数据库管理系统的研究和实现..... ( 14 )	
§ 软件技术 §		谢驹谟
• ~~~~~ •	Auto CAD的快速图形输入..... ( 21 )	
		菇志富
	在Turbo C环境下开发遥感图像显示系统..... ( 26 )	
		高 峰
	dBASE FOXBASE数据库结构的优化设计..... ( 30 )	
		秦学礼
• ~~~~~ •	管理信息系统开发的新途径——原型化方法..... ( 34 )	
§ 技术探讨 §		靳 雷
• ~~~~~ •	TYPE命令的不足和改进..... ( 37 )	
		黄焕如
• ~~~~~ •	智能化信息系统中文本处理的词语表达式系统..... ( 39 )	
§ 专题报告 §		陈有刚
• ~~~~~ •		
• ~~~~~ •	硬盘的数据结构及软故障排除..... ( 43 )	
§ 维修经验 §		宋晓田 李广宇
• ~~~~~ •	RA81磁盘子系统的维护与检修..... ( 46 )	
		徐 力
• ~~~~~ •	计算机软件专业技术资格(水平)考试情况..... ( 封底 )	
§ 综合信息 §		虞国秋 陈敏玲
• ~~~~~ •		

%RA81-COMLETE-TEST; DIAG  
SUBTEST; 01

%RA81-COMLETE-TEST; DIAG  
SUBTEST; 07

·  
·  
·

%RA81 COMPLETE-TEST; DIAG  
SUBTEST; 20

表明运行诊断成功。

2. 马达旋转时, 对磁盘驱动器进行检查  
(“运行/停止”开关置于联接位置)

RA81>RUN DIAG TEST = 16

%RA81-COMLETE-TEST; DIAG  
SUBTEST; 02

%RA81-COMLETE-TEST; DIAG  
SUBTEST; 01

%RA81-COMLETE-TEST; DIAG  
SUBTEST; 07

%RA81-COMLETE-TEST; DIAG  
SUBTEST; 08

%RA81-COMLETE-TEST; DIAG  
SUBTEST; 0B

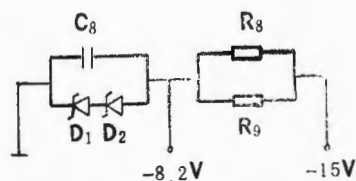
%RA81-COMLETE-TEST; DIAG  
SUBTEST; 03

%RA81-COMLETE-TEST; DIAG  
SUBTEST; 19

%RA81-TEST; DIAG SUBTEST; 1B  
ERROR; F7 UNIT; 001

%RA81-FRU-SERVO, RD/WT, HDA

这表明在诊断测试到子诊断测试1B时, 发现错误。测试“1B”是检查伺服板上的PLO电路, 从上述诊断基本可确诊为伺服板有故障, 检查J602的2脚电压, 结果为0V, 则确认伺服板有故障。查阅有关的电路(如图所示), 分析可能是稳压管D1和D2击穿或电容C8击穿



检查发现电容C8已击穿, 更换电容开机自检, 一切正常。

# 洁霉素发酵罐二级计算机控制系统\*

杭州自动化研究所 徐川育 王道琴 张方强

**摘要** 本文以工业大规模生化反应器中洁霉素发酵为对象,叙述了工艺过程结构模型及控制问题。本控制系统由PS/2 Model80和从Prc-log公司引进的STD PS1.22组成二级计算机控制系统,最后介绍它的实际应用。

**关键词** 发酵 动态模型 参数估计 流加补料 二级计算机控制系统

## 一、引言

洁霉素发酵作为二次代谢在发酵机理上是很复杂的,主要表现为检测难、建模难和控制难。

发酵过程是个微生物活体的生命过程,除了用热工参数外,还要用生化参数来描述,如PH、基质糖浓、氮浓、菌浓和效价。目前国内多数生物传感器缺乏,因此,必须积极开发或引进生物传感器。发酵工程面临四大任务——数据管理、生化反应物料平衡、能源管理和过程优化控制。数据管理指一次和二次生化参数获得、运算、比较和贮存;过程优化控制指对补料系统、搅拌系统和尾气系统的控制。为此,我们做的工作是研制预报控制模型和计算机流加补料控制系统。

## 二、模型和控制

洁霉素发酵是在发酵罐中进行纯种耗氧培养,是间歇式过程,其反应过程表示为



S、E、ES和P分别代表培养基、酶、中间产物和反应产物。

培养基参数有糖浓和氮浓,中间产物参数为PH,目标产物为发酵效价和体积。将培养因子、中间生化因子和目标因子组成一个系统,用结构模型描述递推关系,得到动

态模型。

建模采用两阶段最小二乘法和广义差分法(Durbin程序)<sup>[2]</sup>。

我们所建的工艺过程结构模型如下:

$$ES(t+1) = 2.31 + 0.61S_1(t) + 0.44S_2(t)$$

$$F = 33.08 \quad R^2 = 0.98 \quad DW = 1.96 \quad \hat{\rho} = 0.015$$

$$P(t+2) = 0.97 + 0.22ES(t+1) + 1.02S_1(t) - 0.38S_2(t)$$

$$F = 14.7 \quad R^2 = 0.8 \quad DW = 1.98 \quad \hat{\rho} = 0.01$$

$$S_1(t) = 1.05 - 0.3S_2(t) + 0.1S_1(t)$$

$$F = 17.86 \quad R^2 = 0.96 \quad DW = 2.05 \quad \hat{\rho} = -0.04$$

$$S_2(t) = 4.32 - 2.46S_1(t) - 4.19S_2(t)$$

$$F = 24.39 \quad R^2 = 0.77 \quad DW = 1.94 \quad \hat{\rho} = 0.02$$

F、R<sup>2</sup>、DW和 $\hat{\rho}$ 分别代表统计检验值、线性相关系数、杜宾瓦特森检验值和序列相关系数。

在洁霉素生产中,不是一次投料发酵到底,而是通过延长分泌期,使产量大幅度地增长。采取的措施是中间补料工艺。传统方式采取间歇式脉冲补料,参数波动大。计算机流加补料不仅可以稳定参数,而且在培养机理上与传统发酵不一样,它可以始终使微生物菌体代谢处于半饥饿状态,从而达到最大限度的分泌洁霉素。人为地改变微生物新陈代谢过程。

\*本项目为七五国家重点科技攻关项目75-53-01-13/02

控制的内容有向微生物提供的碳水化合物、蛋白质, 和反映糖代谢和氮代谢比例均衡程度的PH值。具体的参数为糖源流量、氮源流量和PH, 这些参数是新陈代谢中关键参数。

本控制系统的特点是在线检测和某些离线生化测定相结合; 可实现预报控制、反馈控制和优化控制; 真正实现了流加补料新工艺; 能群控6个百吨发酵罐, 使检测仪表智能化, 达到稳流作用。

本系统控制2个发酵罐, 由6个闭环和4个人机闭环组成。6个闭环是: 硫胺稳流闭环2个, 糖水稳流闭环2个, 氨水稳流闭环1个(055罐), PH参数与氨水流量双闭环1个(055罐)。4个人机闭环是: 培养基糖浓与补糖流量2个, 氨浓与补硫胺流量2个。

控制原理图见图1。

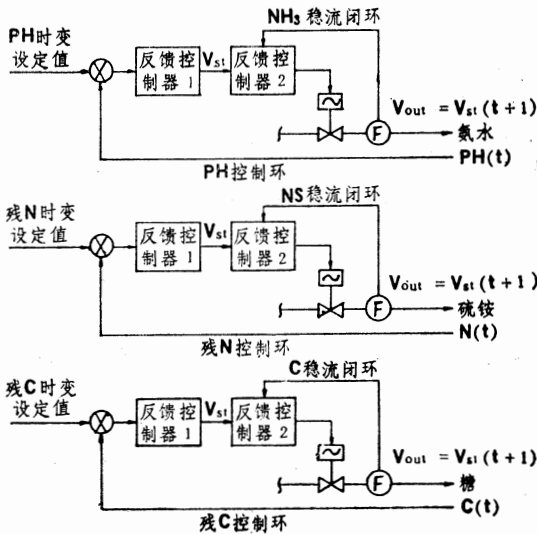


图1 洁霉素发酵罐控制原理图

### 三、二级计算机控制系统

二级计算机控制系统由上位机IBM的PS/2 Model80和下位机STD PS1.22系统组成。

#### 1. PS/2 的操作系统OS/2

OS/2 管理2M内存, 它是一个多任务

操作系统, 其中包含着DOS 3.3子集, 因而在DOS 3.3下支持PC机软件。另外PS/2的微通道结构, 使数据传输及运行快得多, 但只要恰当调整二台异型机的工作时序和通信速率, 应用RS-232C接口, 可以与下位机协调工作。PS/2采用中断工作方式。

STD PS 1.22是在STD DOS环境下工作, 它采用查询工作方式。上、下位机之间的通信经7314板实现。

#### 2. STD PS 1.22系统

PS1.22系统是指PS1.1系统的CPU板上装有实时钟器件, 板号为7863-4。它给用户提供了读写时钟, 等候规定的时间周期和预置时间标志等功能。时间标志最多达10个, 时间分辨率为1%秒。

该系统所带的库文件STD-LIB/V 2.1是美国Pro-Log公司1987年推出的最新版本, 其最大优点是将I/O驱动程序全部模块化, 有80多个程序模块供用户调用。库文件允许用C语言、BASIC语言和宏汇编写成的应用程序链接。因此, 编程时不用再写复杂的驱动程序。

基于通信等因素, 本系统采用BASIC语言。

本系统还采用7314板, 它为连接终端, 打印机和其它外设或系统提供双向RS-232C串行通道4个。每个通道可选择DCE/DTE配置, 采用标准的RS-232C线驱动器和接收器。

STD-LIB软件模块最多可支持32路RS-232C串行通道。每个通道可以分别设置波特率、数据位、停止位、握手信号和奇偶校验, 驱入驱出由中断驱动, 用户定义接收和发送缓冲区。

#### 3. 实际应用

① 实际应用系统的配置见图2。

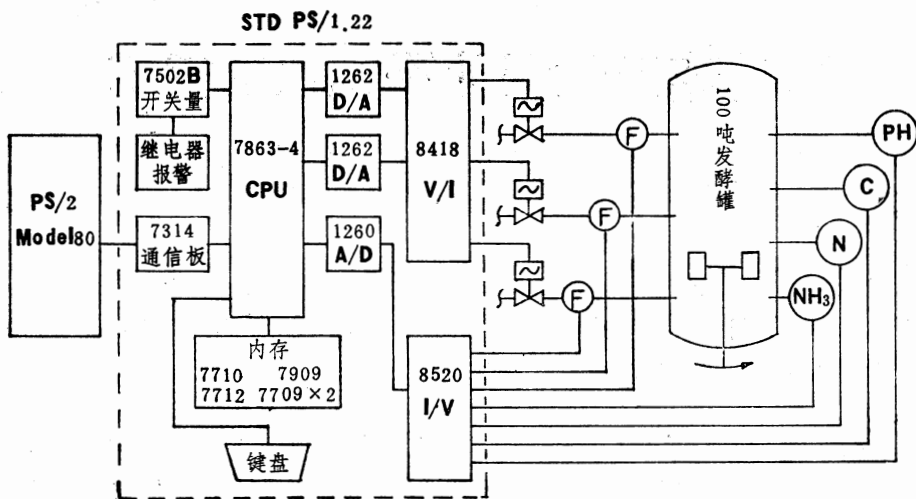


图2 洁霉素发酵罐二级计算机控制系统

② STD工业控制机按查询方式工作。每10秒钟为一工作周期，进行检测、控制和通信三件工作——检测糖流量、氮流量、硫酸流量和PH值，将这些模入信号作A/D转换，进行数字滤波；按工艺要求和控制律，

输出模拟信号，控制流量阀门；接收/发送信号，实现与PS/2通信。

③ PS/2作为上位机，起控制台作用。它对诸多操作项目作选择和切换；设置陷井并捕捉随机事件；显示采样信号，以数字和图形方法不断刷新流程图以及和下位机通信。

④ 应用软件的流程图见图3、图4。

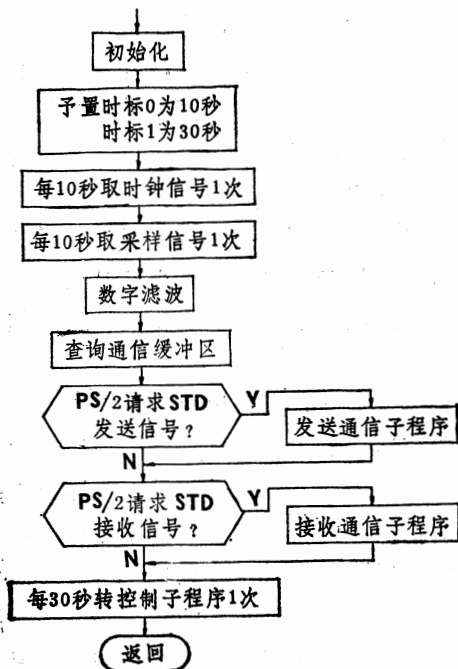


图3 STD PS/1.22机程序流程图

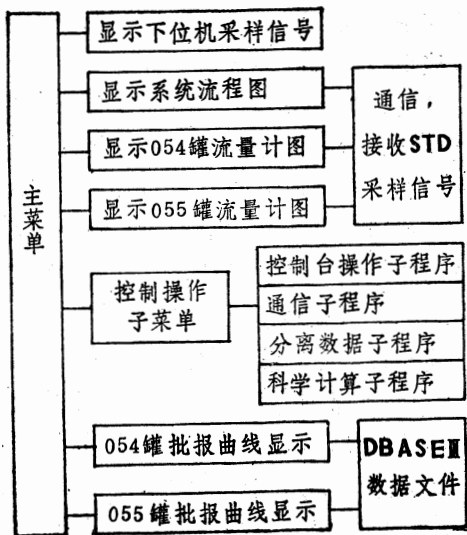


图4 PS/2程序模块图

# 酒精蒸馏过程计算机控制系统

浙江省计算技术研究所 蒋慧丽 蔡亦凡等

**摘要** 本文通过我们对菱湖酒厂酒精蒸馏过程计算机控制系统的研制和开发,介绍了该系统的控制方案及计算机硬件和软件的设计,指出了STD总线工业控制计算机系统的实用性、可靠性及其广阔的推广前景。

## 一、引言

酒精生产中的主要工序是蒸馏,由于当前大多数中小型酒精生产厂蒸馏工序的控制比较落后,有的采用老式的仪表显示,有的甚至还用手工操作。因此工艺参数难以保持稳定,酒精质量也难以稳定。手工操作还受人的情绪,责任心,技术高低等因素的影响,加剧蒸馏过程的不稳定。为了改变落后的蒸馏操作手段,提高中小型酒精厂的自动化水平,采用先进的计算机控制。本文根据蒸馏塔的结构特点和参数特性,采用多回路分散控制和简便的控制方法,正确地选择恰当的控制点及合理的被控参数,酒精蒸馏工段的整个生产过程实现了微机控制。

目前计算机技术在工业控制领域中的应用越来越显示出其优越性。尤其是近年来单片机及STD总线模板的普及和完善,其高可靠性和灵活性,使其应用更加广泛。酒精生产过程要求蒸馏系统长期,连续,稳定运行。因此,对控制微机的稳定性要求特别高。菱湖酒厂酒精蒸馏微机控制系统选用ZJ—STD总线标准的工业控制机。该系统配合国产的温度,压力传感变送器及气动调节阀。从而达到了稳定蒸馏的工艺参数,减少重复蒸馏,稳定成品酒精质量,降低能源消耗,提高设备的利用率和蒸馏效率,挖掘企

业潜力,降低生产成本,减轻操作工人的劳动强度。

## 二、控制方案

酒精生产的关键是蒸馏工序,它是提取成品酒精的最后一道工序,蒸馏系统控制的好坏,直接影响到成品酒精的质量与产量。

蒸馏工序的控制回路选择,根据工艺要求与现有设备的情况,参照原来手工操作过程的测量点与控制点,合理地加以确定。根据该厂长期的生产经验,为了确保酒精的质量,减少酒气,酒液的跑漏,降低能源消耗,必须稳定粗馏塔顶温和精馏塔中温,以及两个蒸馏塔的底温。为了简化控制系统的构成,必须找出成对的过程变量(测量点),与控制变量,(控制点),这些过程变量敏感于自身对应的控制变量,而减少受其它控制变量的影响。确立了以下7个相对独立的闭环控制回路。

- ① 以进醪量控制粗馏塔顶温;
- ② 用控制出酒量来控制精馏塔中温;
- ③ 用控制蒸汽流量来控制进气压力;
- ④ 用控制排废醪来控制粗底温度;
- ⑤ 用控制排废水来控制精馏塔底温度;
- ⑥ 用控制冷却水流量来控制一冷凝器酒精液的回流温度;



⑦ 用控制冷却水流量来控制二冷凝器酒精液的回流温度。

### 三、硬 件

主机采用浙江省计算技术研究所生产的 ZJ—STD 总线标准的工业控制机，输入通道采集蒸馏塔和冷凝器的温度和总管蒸汽压力，输出通道控制进醪电机的转速，及通过气动调节阀调节总管蒸汽量，出酒量和冷凝器冷却水的流量。ZJ—STD 工业控制机 CPU 采用 8031 芯片。配有 12 位 A/D 模板、8 位 D/A 模板、显示模板、键盘、打印机、主机电源及外设电源，还具有过程变量的显示与报警功能。温度和压力等参数定时由 80 行打印机打印输出。对于采样来的温度，压力和流量都能实现实时的动态显示，重新设置和修改实时时钟等。操作简单方便。为了适应多塔多级检测更多路的温度的需要，备有扩展接口端。系统硬件框图见图 1 所示。

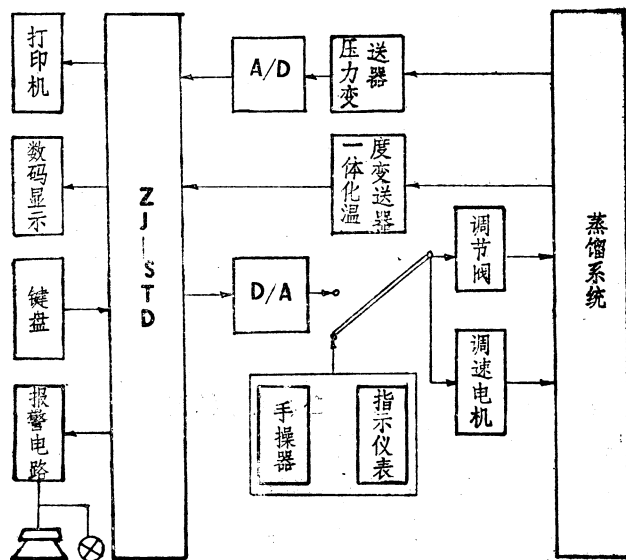


图 1 系统硬件框图

### 四、软件功能

#### 1. 软件框图

程序框图见图 2、图 3 所示。

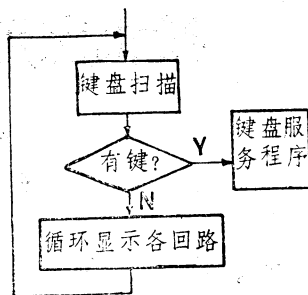


图 2 总控程序框图

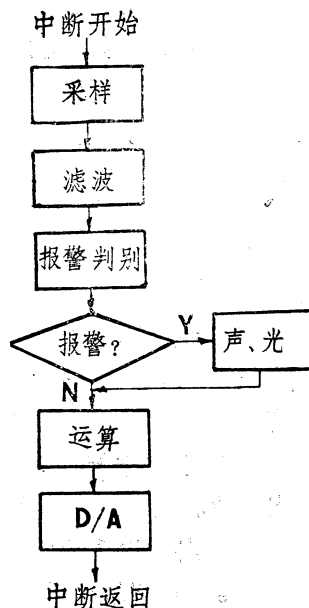


图 3 控制程序框图

#### 2. 数据处理

根据各个被控参数的不同特点，分别设置相应的采样调节周期，对于采样进来的数据首先进行数字滤波，然后进行上下限报警判别，最后以浮点格式存贮于计算机中，供操作人员随时查看，打印。

#### 3. 显示打印

在无键盘操作的情况下，系统进行实时时钟及各回路测量值的巡回显示，

(下转第 13 页)

# 微机应用于监护病房的探索

宁波市第二医院 姚永金  
宁波市计算中心 王玉炜

## 一、引言

危重病人的病情变化迅速,治疗十分棘手<sup>[1]</sup>,要求对病人的各项生理数据比普通病区有更及时、更准确地监测记录,以洞察病情演变,采取相应措施,抢救患者生命。为此,我们在宁波市第二医院监护科探索了用微机来解决这一问题。

## 二、微机系统概况

### 1. 硬件

采用美国HEWLETT-PARKARD公司生产的HP-85型微机,该机为我国医院常用的血气分析仪“Corning-178”的配套机,并带有16K存贮模块(HP-82903A)、只读存贮器(HP-82936A)和选择性系列接口(HP-82938A)各1块。该机还配有显示屏和打印机。临床测得的各项生理数据,值班人员及时由键盘输入。另配有高级磁带装置,作为资料的后备存贮器。计算机能自动将输入数据存入磁带。

### 2. 软件

本机规定用BASIC语言编写程序。作者在代号为“GEN”程序中共编了5个内容。

- ① 生命体征项(VITAL),包括血压、心率、呼吸率、中心静脉压;
- ② 血常规项(BRT),包括血色素、白细胞总数、白细胞分类和血小板;
- ③ 血电解质项(ELECT),包括血钾、血钠、血氯、血钙;
- ④ 肾功能项(RENAL),包括血肌

酐、尿素氮、血渗透压、尿渗透压;

- ⑤ 血气分析项(BGAS),包括血酸硷度、二氧化碳分压、实际硷、硷差、氧分压及氧饱和度。

现以生命体征项为例介绍软件的设计。

## 三、软件的设计

### 1. 框图设计

按需要画出软件框图如图1。

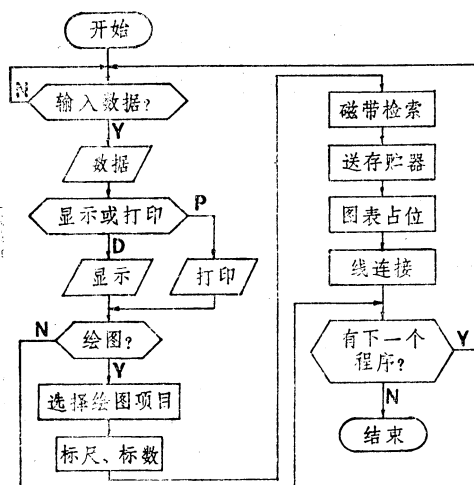


图1 软件框图

### 2. 编写程序

- ① 用功能键列出子程序号。
- ② 选择KEY\*1, 转子程序。
- ③ 显示日期、输入所需日期。
- ④ 从磁带读出原存数据至内存。
- ⑤ 输入新的临床资料。
- ⑥ 列出显示或打印项目。
- ⑦ 选择屏幕显示或打印。

⑧ 选择绘图。

⑨ 不绘图转结束。若需绘图问所绘项目。此项可任意选择全部或部分图形。

⑩ 自动标图。

⑪ 结束。

### 3. 微机打印、绘图

示例见图2、图3。

DATE	BP	HR	R	CVP	TEMP
1103/0700	120/60	100	16	6	37
1103/0900	130/70	110	17	7	38.4
1103/1100	140/80	120	18	8	39.6
1103/1300	100/50	72	13	0	36.8
1103/1500	150/90	130	20	9	40.1
1103/1700	160/86	90	22	8	36.7
1103/1900	90/50	118	21	13	39.1
1103/2100	130/90	120	18	16	40.1
1103/2300	140/80	72	16	4	37
1104/0100	130/70	78	14	15	38
1104/0300	125/69	110	10	10	39
1104/0500	112/90	100	20	7	38.5

图2 打印数据示例

DATE为日期, 1103/0700为11月3日7时;

BP为血压(mmHg), 120/60为收缩压/舒张压; HR为心率(次/分);

R为呼吸率(次/分);

CVP为中心静脉压(cmH<sub>2</sub>O);

TEMP为体温(℃)。

DATE 1103的 BP、HR、R如图3

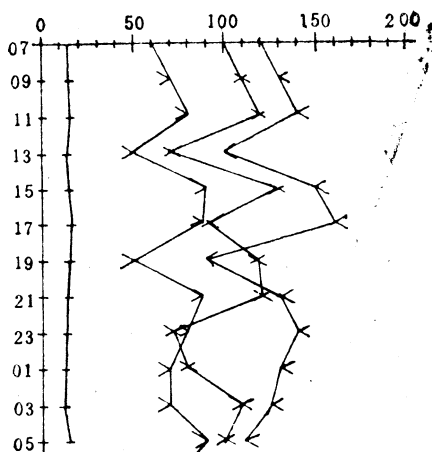


图3 绘图示例

X轴为刻度, Y轴为时间, 2小时记录一次。图中, <为收缩压、>为舒张压、×为心率。-为呼吸率。

## 四、讨 论

现代科学的发展使计算机几乎进入了一切领域<sup>[2]</sup>。微机能快速整理重危病人的临床资料, 为医生诊断病情提供丰富可靠的依据<sup>[3]</sup>, 这个功能正是监护病房所急需的。它可加速诊治和改善经治者的效率<sup>[4]</sup>。本微机原作为血气分析仪质量控制和自动记录血气分析结果用, 经我们编入软件后, 扩大了应用范围, 但不需增加经费支出且输入数据、显示打印的速度较人工书写快得多, 资料准确, 文字加图表可清晰地观察到病情变化趋势, 更有利于危重病病人的及时抢救。本系统用BASIC语言编写, 易懂易学, 数据和图表资料可长期保存。

### 参考资料

- [1] 金定炼 杭燕南主编, 重症监护手册, 第1版, 第1页, 上海科技出版社, 1988年。
- [2] 谭浩强 田淑清主编, BASIC语言, 第1版, 科普出版社, 北京, 1980年。
- [3] 华蕴博等, 电子计算机的整机结构和一般原理(讲义), 第55页, 1982年。
- [4] McDonald J et al, JAMA 259, 3433, 1988。

# 计算机医务管理系统

民丰造纸厂 张 希 章寿发 杨承基

## 一、概 述

我厂职工医院从1989年开始应用计算机,最初只是为进行职工医疗费用的统计结算。通过近两年的开发,现已形成了由医药用品管理、划价记帐、职工健康管理三个子系统组成的计算机医务管理系统。这一系统把医药用品管理和划价记帐结合在一起,实现两个子系统既在医药品数据方面高度共享,又是相对独立的子系统。

## 二、系统分析

我厂职工医院具有综合性医院的共性,科室齐全,各司其职,但又有联系。由于职工医院主要是服务于职工及其家属的医疗保健工作,因而又有其特殊性。例如,病员来源相对稳定,医药费实行定额管理,职工医院除了提供医疗服务外,还要提供劳动卫生、健康保健,卫生防疫服务。

经过对医院的调查分析,我们对医院的业务范围,信息流向,管理方式有了清晰的

了解。从系统调查分析的结果看,我厂职工医院的业务主要有6部分,即医药品库房管理,医疗管理,医药费管理,病房管理,劳动卫生管理,计划生育管理。我们把开发工作的重点放在医药品库房管理、医疗管理、职工医药费记帐、职工健康档案管理和病房管理上。

## 三、系统设计与实施

### 1. 系统功能

我们将计算机医务管理系统设计成由5个子系统组成,各子系统均含有多个不同的功能模块,系统功能图见图1。

① 医疗用品药库管理子系统。它的功能有

A. 医疗用品和制剂的入库登记;

B. 领用登记,即各科室领用的医疗用品及时登记入帐,对各药房(门诊药房、医疗点药房、病区药房)的各种药品进行实时更新,各药房的各种药品数量补充是通过其领用登记完成的。

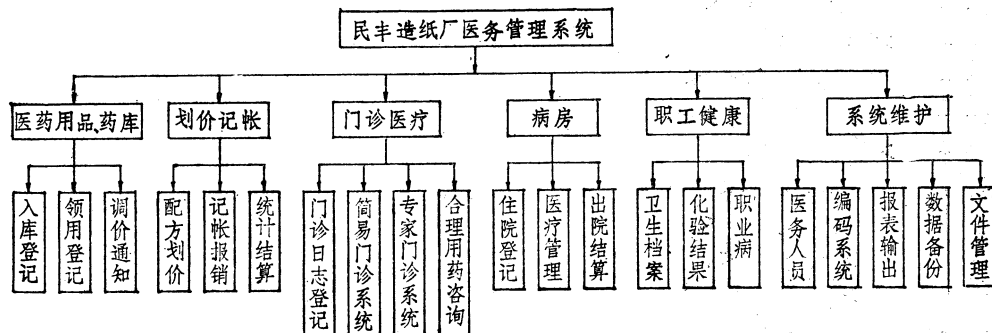


图1 系统功能图



C. 调价通知, 将医药公司的医用品调价通知及时输入并自动对库房和各药房的相应药品进行价格调整。

② 划价记帐子系统。它是整个系统的核心, 它的功能是

A. 配方划价。即划价和记帐。

B. 记帐。对职工在厂外医院就诊的记帐单转帐和职工的化验、检查等费用及出差在外的治疗费用报销等业务。

③ 门诊医疗管理子系统。它的功能有

A. 门诊日志登记, 以便获得流行病资料。

B. 简易门诊。对于常见病, 多发病, 通过本系统即可开出协定处方, 以杜绝病人点医、点药而且减轻医生的工作量。

C. 专家门诊。引进国内比较成熟的专家系统, 对疑、难、杂症进行咨询。

D. 临床用药咨询。有常用药物的各种临床资料, 参数和药物配伍变化资料, 其中的药物配伍变化可被划价记帐子系统直接调用。

③ 病房管理子系统。它的功能是

A. 住院登记。输入住院人员病历, 并根据职工卫生档案管理的要求自动将该病历归入职工卫生档案。

B. 治疗管理。记录病人整个住院期间的病情发展过程、用药情况、护理医嘱、治疗结果。

C. 出院结算。结算住院期间的医药费用, 对本厂职工自动实现转入医药费的统计结算, 对外来的住院病人按结算收费。

④ 职工健康管理子系统。它的功能是

A. 职工卫生档案管理。对全厂 5 千多名职工的健康史进行管理, 将个人的嗜好史、既往病史、配偶病史、家族疾病史、月经史、生育史、职业史、工伤事故记录、急性职业病记录、尘肺病记录、结核病记录、其

它慢性职业病、肿瘤记录、其它现患疾病记录等输入到计算机, 便于查阅、增加、修改、归类及统计。

B. 体检、化验结果管理。把工厂对职工实行定期检查的结果输入计算机。如自觉症状、体检结果、化验结果、主检医生意见等, 并将医生意见和结果及时反映给有关部门。同时也是职工卫生档案管理的一部分。

C. 毒害工种、职业病管理。对从事有毒有害工种的工人实行专门管理的模块, 我厂主要毒害类别有粉尘、噪声、酸、碱等, 定期对这些职工的检测记录, 对职工的劳动保护起到了积极的作用。

⑤ 系统维护子系统。它是管理系统的后援系统, 它进行医疗用品等的编码管理, 数据文件和程序文件的管理。

## 2. 数据库设计

在进行数据库设计时我们采取代码化设计方式, 这里仅以医疗用品药库管理子系统中的数据库设计为例。

我厂的医务管理系统是在多用户系统上开发的, 各个子系统既互相联系又相对独立。医用品数据库和一码多用数据库是两个通用的数据库, 他们和一个内存变量文件在各子系统的运行过程中提供它所需的中文信息, 既满足实际使用又缩短文件长度, 使得系统运行时降低计算机内存和时间的开销。这一数据库设计方法的优越性在划价记帐子系统中尤为突出, 这一系统运行中产生的数据量非常大, 平均每天有 4 百条左右记录, 每月的记帐文件字节数都在 340~380 KB 之间。另外, 我们在系统设计中采取每个月的数据文件分别自动建立管理, 这也使得文件的记录不至于过长。数据库结构见表 1 至表 5。

表 1 医用品数据库

段 名	DH	MC	GG	DW	DW <sub>1</sub>
含 义	医疗用品编号	医疗用品名称	规格	计 量 单 位	处方用药单位
结 构	C(6)	C(14)	C(10)	C(2)	C(2)
索引段 BH					

表 2 一码多用数据库

段 名	DH	MC	ML	GH	XM
含 义	编 号	院内单位名称	核 算 项 目	来往单位名称	医务人员姓名
结 构	C(2)	C(12)	C(10)	C(16)	C(8)
索引段 DH					

表 3 购买医用品发票数据库

段 名	DWFPH	RQ	ML	BHPH	XS	SL	PFE	GME	YHE
含 义	单位号和发票号	发 票 期	核 算 代 码	品 号 和 批 号	装 量	购 买 数 量	批 发 价 总 金 额	购买金额	优惠金额
结 构	C(8)	C(6)	C(2)	C(12)	N(2)	N(4)	N(10,3)	N(10,3)	N(10,3)
索引段 DWFPH									

表 4 医用品管理数据库

段 名	BHPH	XS	DJ	SL	S <sub>24</sub>	S <sub>26</sub>	S <sub>31</sub>
含 义	医用品编号及其批号	装 量	单 价	库房现存的药品数量	门诊药房现存药品量	病区药房现存药品量	医疗点药房现存药品量
结 构	C(12)	C(4)	N(8,3)	N(4)	N(8)	N(8)	N(8)
索引段 BHPH							

表 5 五类药品数据库

段 品	CFH	DW	BHPH	XS	DJ	CS	YS
含 义	处方号	药房号	医用品编号及其批号	装 量	单 价	处 方 数 量	医师号
结 构	C(8)	C(2)	C(12)	C(4)	N(8,3)	N(8)	C(2)
索引段 CFH							

### 3. 划价记帐的实现

用计算机进行划价记帐是我厂计算机医务管理系统的一大特色。

#### ① 划价记帐的主要过程

A. 药品在采购入库时按其编号、批号、价格、单位装量进行归类存贮在库房段中,这时的数量是大单位数量,各项内容全一致时数量相加后存入,否则产生一条新记

录。

B. 各药房领用时按批号先后顺序自动领出,并转化为处方用药单位与现有的数量相加后存贮于相应的字段中,这一过程记录数不会有变化。

C. 配药划价时只需从键盘上输入药品编号,然后计算机自动查找到第一条编号与之相同,而相应药房名下的数量不为零的记

录上,把有关信息显示出来(如药品名称、规格、装量、现存数量等)接着输入数量,输入数量的值若比机器的数值大,则机器进行屏蔽。在处方中如有5类药品(麻醉、精神、放射、贵重药品和毒品)则自动记入5类药品数据库。

D.处方上的药品输入完毕,如有必要可调用药物配伍变化检验程序,发现配伍禁忌记录该处方,并作退方处理。

E.处方输入完,计算机仅把日期、就诊单位、处方医生、配药单位、就诊人员工号及处方金额按一张处方一条记录存贮,以备统计核算用。

#### ② 划价记帐的程序设计

上面表4(医用品管理数据库)给出了划价记帐中使用的数据库的结构,这一数据库的主索引文件是对字段BHPH进行的,所以其排列型式即已符合药品配药时的先进先出要求。

以划价为例,结合程序加以说明。

```

do While .t.
  bha=
  @ 1, 0 get bha
  read
  → if bha=
    exit
  → endif
  select 3
  use Kfwj index Kfwj
  set relation to left(bhph, 6) into ypml
  find' &bha'
  → do While left(bhph, 6)=bha.and.S&dwa
    =0 Skip
  → enddo
  → if left(bhph, 6)=bha
    exit
  → endif
  @ 21, 60 say d&dwa.1'无此药品'
→ enddo

```

在给出药品编号后,首先通过执行find命令快速定位(快查)于所要配出药品的第一条记录(同一种药品很可能有数条记录),然后通过循环指令进行逐一检查定位(微查)。这一过程结束后,进行判断是否找到所要配的药品,接着输入数量,并根据单价进行记帐。

## 四、系统特点

本系统通用性、兼容性强。系统选用FOXBASE+数据库,在XENIX多用户操作系统GW—286型(带4个终端)微机上开发而成。由于FOXBASE+与dBASEⅢ完全兼容,故本系统既能在XENIX多用户下运行,也能在CCDOS下运行,可实现多用户,也能满足单用户。

我厂职工医院虽然规模较小,但业务范围、管理方式与专业医院基本一致,所以该系统也适用于各专业医院。

经过一段时间的使用,医院的有关人员反映其快速、准确、方便,随时可以输出打印统计报表以及其他报表文件,极大地提高了工作质量。这一系统的使用可避免错付购货款;避免医疗用品的积压;严格执行医疗用品价格政策;及时、准确地进行职工医药费结算;控制非法医疗费用每年可达5~15万元。



经验交流

# 逻辑驱动器的建立及使用

机械电子部兰州石油机械研究所 霍云峰

本文介绍如何在只有一个硬盘驱动器上,建立多个逻辑驱动器,并说明如何使用这些驱动器(这里所说的逻辑驱动器不是用内存的部分区域作为存储介质的仿真磁盘驱动器的虚拟磁盘,而是具有物理磁盘驱动器功能的逻辑驱动器)。

## 一、条 件

计算机应具备一个硬盘驱动器,机型为IBM、长城、AST等系列均可;应具备DOS 3.3以上版本的磁盘操作系统,并装在软盘上。

计划一下你的硬磁盘有效空间多大。准备建几个逻辑驱动器(最多能建23个逻辑驱动器,即:D、E、F……Z)。每个驱动器分配多大空间,以连续空间的起始值和终止值来定义每个逻辑驱动器的大小。并列出分配表。下述分配表仅建立三个逻辑驱动器D、E、F,其格式为

分 配 表

名 称	驱动器 符号	起 始 (MB)	终 止 (MB)	逻辑驱动器 (MB)	各 区 间 (MB)
硬 盘 可用空间		0	40		40
初 级 DOS分区	C	0	20		20
扩 展  DOS分区	D	20	30	10	20
	E	30	35	5	
	F	35	40	5	

## 二、准 备

首先将DOS3.3以上版本的软盘插入A

驱动器并启动。用后备文件命令BACKUP将硬盘上的文件复制到你的空软盘上。然后运行FDISK程序,并选择屏幕上相应的选择项来删除硬盘中以前建立的DOS分区。必须先删除扩展DOS分区,后删除初级DOS分区。

## 三、建 立

按下列程式进行。

1. 运行DOS3.3以上版本操作系统中的FDISK程序。屏幕上将显示主菜单,选择(1),即“建立DOS分区”,并按Enter键。将显示二级菜单。

2. 选择二级菜单中的(1),即“建立初级DOS分区”。按Enter键,屏幕将显示“你想用DOS初级分区最大的可能空间吗(Y/N)?”,回答N后,屏幕提示“要求回答初级DOS分区空间大小”,你按空间分配表回答后。按Enter键,初级DOS分区建成。系统自动将字母C定义给初级DOS分区。按ESC键,回到二级菜单。

3. 选择二级菜单(建立DOS分区菜单)中的(2),即“建立扩展DOS分区”,按Enter键,屏幕提示“要求回答扩展DOS分区空间大小”,按你的分配表回答,按Enter键就成。按ESC键,回到二级菜单。

4. 选择二级菜单中的(3),即“建立活动分区”按Enter键,屏幕提示“回答要置活动分区的数目”,回答数字1。按Enter键,这个1就是你上电或执行系统复位时要激活的分区号,该分区为活动分区。



在某一时刻只有一个分区有效,有效分区控制着系统,要启动驱动器C时,该DOS分区为有效的。

5. 按ESC回到主菜单,选择(1),即“建立DOS分区”,按Enter键,显示二级菜单后再选择(3),即“在扩展DOS分区中建立逻辑DOS驱动器”,按Enter后屏幕显示“回答逻辑驱动器空间大小”,按你的分配表回答空间大小,按Enter键,系统则自动为此空间定义D驱动器并显示在屏幕上,而且屏幕上重复显示“回答逻辑驱动器空间大小”,然后继续按分配表回答,并按Enter键,如此循环下去,系统则自动分别定义各空间为E、F、G、……等逻辑驱动器。

重复按ESC键,返回到主菜单。选择(4),即“显示各分区信息”。对照自己的分配表,其空间大小及符号是否符合?如不

符合可重复上述操作进行修改。

#### 四、使用

逻辑驱动器建立在扩展DOS分区上,当DOS被启动后,每个逻辑驱动器指定一个驱动器符号,如,D、E、F……。通过驱动器符号来存取该驱动器信息;就如同它们都是各自分离的硬盘一样,在各逻辑驱动器上可装入不同版本的操作系统,装入前应该用相应版本的DOS系统对各逻辑驱动器进行格式化。格式化时可建立各系统的卷标参数,然后装入各系统文件。在格式化C驱动器时应该加/S开关项,以便让IBM BIO.COM、IBM DOS.COM及Command.COM拷贝到C驱动器盘上,以使用硬盘启动系统。

最后,用RESTORE命令恢复硬拷贝文件到你所希望的各逻辑驱动器上。

(上接第3页)

#### 四、应用结果

本系统已成功地应用于华北制药厂的两个洁霉素百吨发酵罐,并于1990年5月通过了由机电部仪表司组织的专家鉴定。

计算机流加补料控制系统投入运行后,洁霉素发酵单位(u/ml),发酵指数和小时增长率分别提高11.3%、8.64%和5.84%,效果显著。经浙江省科技情报所1990年5月2

日国际联机检索结果,未发现同类研究。鉴定委员会认为本项目达到了80年代国际水平。

#### 参考文献

- [1] D.G.Mou and C.L.Cooney. Biotechnol. Bioeng. 25, 225, 1983.
- [2] 张寿 于清文编著, 计量经济学, 第一版, 上海交通大学出版社, 1984年。

(上接第5页)

通过键盘操作,系统具有

① 可对系统内7个检测点的数据进行查看,并可根据需要随时对这些数据打印出中文报表,供工艺员及管理人员使用。

② 可对控制回路给定值进行设定和修改,并可随时查看设定值。

③ 可进行手动/自动,自动/手动无扰动切换,并设有报警切除/恢复报警的软

开关控制。

④ 可对实时时钟进行调整,修改。

#### 五、结束语

本系统各回路中除2个特殊要求以外(粗底、精底)均采用PID控制算法,各参数均经过现场整定,用简单而成熟的控制方法,实现了对复杂对象的控制。回路的控制精度在 $\pm 1^{\circ}\text{C}$ 以内,完全满足实际生产的要求。

# 分布式数据库管理系统的研究和实现

中国纺织大学计算机系 谢驹谟

**提要** 本文概括地介绍一种分布式数据库管理系统在欧米网络中的实现方法。该系统基于网络通信软件、站点局部数据库及操作系统,其工作流程由总控、全局会话、局部会话等16个工作模块组成。文中对数据模式、数据分布策略、查询处理、并发控制等内容作了较详细的分析研究。

## 一、引言

分布式处理系统是当今计算机科技领域中发展最为迅速的一个分支,其中分布式数据库的研究,更成为一个日益被重视的课题,并且可以预见,它的重要性还在不断增长。这是由于分布式数据库比集中式数据库具有许多优点,诸如分布性、逻辑关联性、可扩性、灵活性、可靠性和可用性等。

随着微、小型计算机的最新发展,能够以最低廉的价格提供以前大型机所提供的能力,再加上计算机网络技术,数据库技术的进展,为分布式处理、分布式数据库实现提供了广阔的领域。国外,对DDBMS(分布式数据库管理系统)的研制较早,如较有名的系统有:SDD—1<sup>[1]</sup>、R\*<sup>[2]</sup>等。国内近年来随着计算机网络的普及和应用,不少单位亦开始这方面的研究。

本文在欧米局部网络和各工作站点的本地数据库管理系统上,建立起关系型分布式数据库系统(ODDBMS)。

该系统的设计目标和要求为

1. 提供分布式数据库系统的所有功能。
2. 分配合理协调,发挥局部网络控制方便、传输速率高等优点,使整个系统保持良好的运行性能和可靠性。

3. 保持系统今后可扩充、可移植的特性。

4. 对用户来说,要求使用本系统如同使用集中式一样方便,故应向用户提供4种透明性,即

①位置透明性——访问数据与物理位置无关。

②并发透明性——多个用户同时访问系统时,具有冲突检测和并发控制性能,与用户无关。

③故障透明性——局部故障不会导致整个系统瘫痪,整个系统将继续工作。对非致命故障有自恢复功能。

④分布透明性——对用户的整体访问,由系统映射到一个适当的局部访问。

## 二、系统构成

本系统的功能开发采用综合的体系结构,将全部功能、系统功能分散到网络中各个工作站上,即以联合和匀质为条件的分布系统。联合是指以每个结点上现有的DBMS为基础,向上扩展,设计成整个系统。匀质是指各个DBMS、数据模型和语言、命令都相同。因此全局只设一个统一翻译器,各个子系统间不必再作翻译。充分利用原有数据库的管理功能,因此,本系统的设计重点可放在分布式数据库系统部分,见图1。

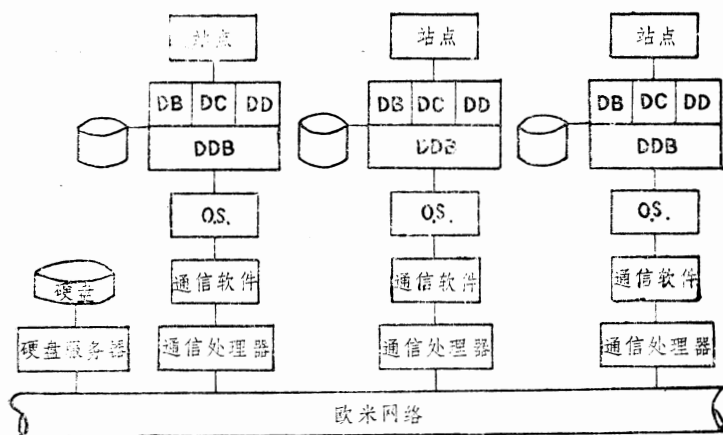


图 1 系统结构

其所需的软件为

- ①局部数据库管理系统（LDBMS），图 1 中的 DB。
- ②数据通信部分（DC）。
- ③数据词典（DD）。
- ④分布式数据库管理系统（ODDBMS），图 1 中的 DDB。

每一站点均有上述诸部分，LDBMS 由原微机 IBM—PC/XT 中数据库管理 DBASE—II 构成。ODDBMS 是本系统实现的主要环节，每个站点都保持它的一个副本。较中心站点控制方式可靠性高、通信费用少，亦不易产生瓶颈拥塞现象。

### 三、ODDBMS 的构成及流程

图 2 为 ODDBMS 的模块构成及流程。从微机局网的特点出发，有明显层次界限，因而设计成一种较为清晰的工作流程。

作为用户、事件和分布数据库之间的软件接口，ODDBMS 应具有 3 个主要功能。

#### 1. 负责描述数据库的逻辑和物理结构

把其信息以完全或局部形式存放在分布数据字典（DD）内。

## 2. 管理数据库

①它控制整个数据库的运行，接受用户的请求，由查询子系统查询、分析数据，作优化处理。

②由完整子系统负责维护整个数据库的完整性、一致性和保密性。

③负责调度子系统的执行及调度局部处理、数据传送等命令。

④具备翻译功能。

## 3. 维护数据库的可靠性

由可靠性子系统提供全网络范围内的各种后备和恢复功能，保证系统的安全运行。

为了完成以上功能，其工作流程及 16 个模块构成见图 2，各模块的作用分述如下。

#### 1. 总控模块（ODDBMS）

两种用户的访问及外部数据请求均由总控进入。当系统确认用户合法，则进入由它调用的全局会话、局部会话、响应请求三个模块之一。

#### 2. 全局会话模块（GCON）

由全局用户 GUSER 或全局数据库管理员 GOBA 进入，负责对 GUSER 或 GOBA 的会话，查询名称、通行字，如为非法码，则拒绝之。

#### 3. 局部会话模块（LCON）

由局部用户（LUSER）进入。查询名称、检查口令字并确认合法后直接转到局部数据库系统模块（LDBMS）。

#### 4. 响应请求模块（REQU）

当在指定单元查询到有信息时，即转到 REQU，处理从其他工作站发来的外部数据

请求命令。

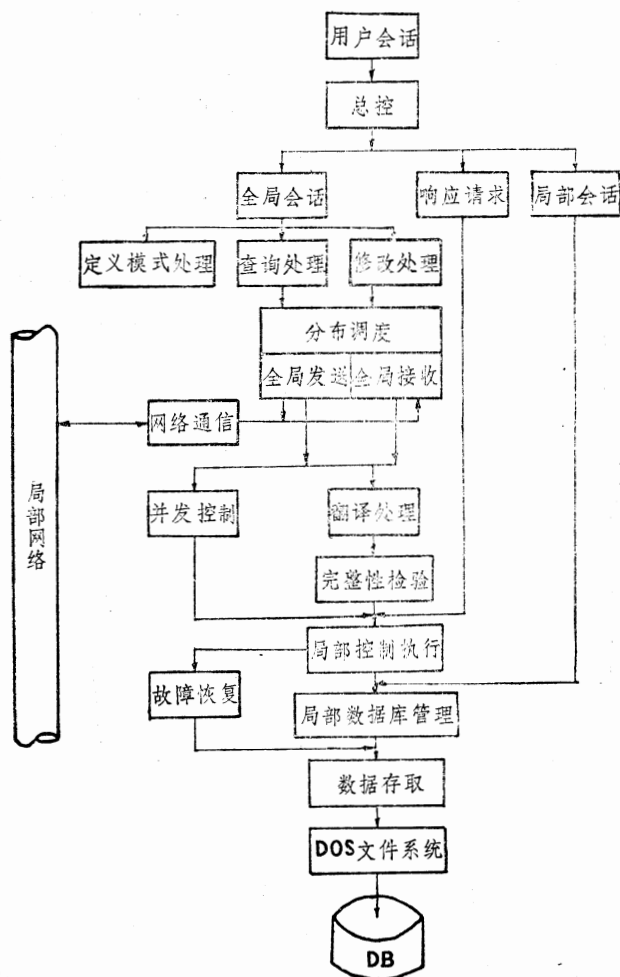


图2 工作流程及模块构成

### 5. 定义模式处理模块 (DDL)

由GOBA所进行的用户和数据库作增、删、维护情况下，它定义数据库的全局概念模式和内部模式。

### 6. 查询处理模块 (QUERY)

进入查询分析，形成操作总表，分解操作表，数据定位，得出最佳调度策略。

### 7. 修改处理模块 (UPD)

包括增加、删除、追加、插入、修改等功能，予分析和调变策略类似查询处理，

不同的是结果不返回源站点。

### 8. 分布调度模块 (DSCH)

对上述QUERY及UPD两模块获得的操作结果进行处理、发送局部操作，并发送到各站点的请求缓冲工作区。其数据格式为

标识	类型	顺序号	操作原语	源关系
目的关系	发向站号	目的站号		

当类型 = 0，则为向各站点发送命令。

标识——运行事件的名称。

顺序号——规定本次操作的顺序。

操作原语——发送命令原语，如SEND（发送）、RECEIVE（接收）、SELECT（选择）、PROJECT（打开）、JOIN（连接）、UNION（合并）等。

源关系——参与操作的关系名。

目的关系——取得运算结果的关系名。

发向站号——原语命令发向的站号。

目的站号——原语操作的目的站号。

当类型 = 1时采用

标识	类型	源站点	目的站号
----	----	-----	------

一般用于向指定站点传送数据，如用SEND操作发出，各行意义上同。

### 9. 并发控制模块 (CONCR)

和集中式数据库一样，危害数据完整性的一个重要原因是并发操作。许多DBMS可同时支持多个站点用户事件运行，为防止因竞争而引起冲突，系统必须解决并发更新问



题。它所提供的并发独立性应使每个用户以为自己独自使用数据库，并能和其他并发用户的动作隔绝。本文采用予分析时标法，保证对运行部相同数据项作串行化调度处理，并送回其结果。

#### 10. 网络通信模块 (COMM)

利用欧米网络的通信原语，计有 7 条，分别为

- ① INITIALIZE TRANSPORTER 传输器初始化。
- ② SEND-PACKET 发送信息包。
- ③ SET-UP-RECEIVE 建立接受连接。
- ④ END-RECEIVE 接受结束。
- ⑤ WHO AM I 查本站点号。
- ⑥ ECHO 回声 (查联网站点号)。
- ⑦ PEEK/POKE 装入/卸出。

本系统采用的入口数据格式为

主 被动方式	命令	源站号	目的站号	发送名
-----------	----	-----	------	-----

#### 11. 翻译处理模块 (TRNSPC)

完成全局数据库命令向局部数据库命令的翻译转换，这是通过命令处理器实现的，将经网络通信的全局命令译成站点上局部数据库 LDBMS 可执行的 DBASE—II 命令文件形式。

#### 12. 完整性检验模块 (CHECK)

为确保数据的正确性、相容性和合理性，防止错误数据的存贮和操作。系统提供保证在任何时候数据库都具有完整性检验功能。作为一种数据保护策略，完整性约束条件是完整性检验和控制的依据，它可以约束数据的允许值，数据之间的联系，可包含单个或多个关系，当一个数据库应用所执行的修改违背了完整性约束条件时，系统就拒绝这个应用，从而保证了数据的正确性。

#### 13. 局部控制执行模块 (LEXEC)

按分布式调度算法所确定的顺序，选择

执行发送或接收等数据库命令。若正确执行，则正常结束，否则转入故障恢复模块。

#### 14. 局部数据库管理模块 (LDBMS)

采用 DBASE—II 2.4 版原型，由 LEXEC 或 LOCON 调用。

#### 15. 故障恢复模块 (RECOV)

当出现网络或站点主机故障，将发现接收方在数据格式的标记单元中。收到“NO”，经过一定延时，若再次收到“NO”，则令其转入 RECOV，由该模块将出错站点上文件自动转存到网络硬盘的相应盘体，待恢复后再将其送回原站点。

#### 16. 数据存取模块 (DACES)

对数据库的一次应用操作，总会涉及到对数据库中数据的存取。本模块通过对缓冲区的管理和 DOS 文件系统功能调用，完成对数据库文件的读出和写入。

## 四、数据分布策略

设计分布式数据库系统的数据分布时，为了分布恰当，有利于查询、存取和故障恢复，又不使存贮开销太大，一般采用数据冗余技术，使数据库数据在若干站点存有复制付本。

在 DDBM 系统中，常用的有两类分布策略。

#### 1. 重复型

这种类型又可分为完全重复型和子集重复型，前者每个站点数据库都是总体数据库的重复；后者每个站点数据库只是总体数据库的一个子集。

在重复型中，如果某些数据的付本因故障受到破坏，仍可用其他付本继续处理。这种类型所化的检索通信费用较低，但更新时开销增大。

#### 2. 分割型

它亦可分成重复分割型和子集分割型两

种,前者没有设置总体数据库,所有站点复制了全部数据,后者亦没有总体数据库,而每个站点只复制部分数据。可见,子集分割型的冗余度较低,存取更新开销小,但通信费用稍大。

本系统基本上采用子集分割型方式,但在系统硬盘上采用总体重复型方式,这样既照顾站点上微机存贮容量有限,又考虑到控制方便,通信费用满足要求等条件。

按ANSI/X<sub>3</sub>/SPARC建议,数据描述有下列3个层次。

1. **外部模式** 可有一个或几个,每个包含数据库的一部分描述,供使用此数据库的用户使用。

2. **概念模式** 定义各个关系的逻辑结构,供数据库管理员在设计或修改数据库时使用。

3. **内部模式** 它包含物理数据的位置和格式的描述,以表示概念模式描述的各个数据类型。具体说,如分布类型、分割条件及站点信息等,供数据库管理员精调,以更好发挥该数据库效能。

上述3种模式放在一个称为数据词典(DD)的专用数据库中,使用数据描述语言(DDL)来对该3种模式进行表述,如提供关于数据库本身信息,模式定名,定义数据项记录项,建立记录间逻辑关系,指出安全控制要求,指明数据完整性约束条件等功能。

## 五、查询及优化

在分布式系统的查询过程中,各站点之间的通信造成了明显的时间延迟,其开销甚大,必须采用良好的查询策略以减少其时间延迟。

本系统进行各种控制性操作,如查询检索、插入、删除、更新、用户命令检查,打关闭数据库等功能,均可由数据操纵语言

DML来表示。DML由一组命令组成,一条命令完成一种操作,这里仅讨论查询功能操作的实现过程。

为使查询的响应时间和总代价性能优越,本系统采用静态的有限探索策略,以取得较小总代价为目标,该方法是比较有效的。

查询优化过程的步骤是

### 1. 词法分析,语法分析

查询语言经词法分析得到单词符号串,利用有限状态自动机原理,通过语法分析,接受这些单词符号,自下而上归约,化解为子句形式,并提供给查询机构。

其间形成语法定义表

```
SELECT <target list>
FROM <relation>
WHERE <boolean condition>
```

### 2. 完整性约束,合法权检查

判别用户访问合法性,数据存在性。

### 3. 子句化解

按定义表中指出的所有关系为单位,把查询操作化解成对一个关系的操作,并加上必要的操作符。

设关系以符号R表示,数据在两个站点之间传输的代价C是传输数据量X的线性函数。

$$C(X) = C_0 + C_1 \cdot X$$

$C_0$ ——每一个传输所需的起动时间。

$C_1$ ——大于0的常数。

假定站点间数据传输的代价远大于局部处理和局部范围内存贮设备之间数据交换的代价。

化解的方法为

①每一 $R \in \text{FROM}$ 。

②从R的每一指针,读出WHERE(条件)。

③若条件是单目标,加上SELECT操

作符。

若条件是双目标,则在连接域上加 PROJECT操作符。

④当查到 R是结束关系,则在输出域加上“find”。

⑤直到全部指针读完, FROM 表处理完成。

3个语法定义表的格式如下:

SELECT	FROM指针	WHERE
输出域	关系	条件

化解方法是为获得查询操作代价 C 最小,做了必要准备。

#### 4. 站点定位

使所有全局关系都获得具体物理位置,在分割分布策略下,首先定位站点是分割涉及的分布站点集,记作

$$R_i \rightarrow \{S_{i1}, S_{i2}, \dots, S_{ii}\}$$

然后,对分布集  $\{S_{i1}, S_{i2}, \dots, S_{ii}\}$  实行区域优化算法,删除不必要站点。

#### 5. 最小代价优化算法

若每个关系处于不同站点,在分布式查询处理代价图中可表示为  $\{S, R, E, M\} Q$ ; 其中 Q——本次查询; S——分布站点集;

R——站点关系集; E——指针集;

M——连接域集。

对分布查询代价图按代价函数  $C(X)$  实现优化调度算法,选择当时最小代价  $C(X)$  进行合并,合并的数据量小者向大者传送,删除两个旧关系,生成一个新关系。然后,图形中参数重新计算,直到取得最佳的调度结果。

### 六、并发控制机构

为防止多用户操作时带来数据完整性被破坏,本系统采用改进的时标法以实现并对并发事务的控制和管理,其算法是:事

件除了要记下时标  $t$  外,还要记下发源站点 ID,操作符 OP,数据集 SET 及后继符 SUCC,形成一元组:  $\langle t, ID, OP, SET, SUCC \rangle$ 。并始终以  $\langle t, ID \rangle$  作为关键字。对于并发的两个事件  $T_1, T_2$ ,先按  $t_1, t_2$  时间关系,再按源站点  $ID_1, ID_2$ ,决定其优先数大小,即依据  $\langle t, ID \rangle$  的大小,将事件元组编排成等待队列,每次取队首元组投入运行,保证了可串行化工作。另外,系统在共享盘体上设置了处理日志和事件处理集,以提高并行处理效果。

处理日志是记录从开始运行到当前为止所有对数据库的执行操作。一个元组运行后就放入处理日志,元组的后继符表示本处理对该数据集的操作是否结束,当后继符等于 0,表明操作结束,把处理日志关闭。遗留指针指向等待队列的队首元组,但该指针没有指出新元组的投入,会不会发生冲突,故从避免重新启动出发,不把该元组立即运行,再设置一事件处理集,当新事件产生,先对之进行予分析,对操作的数据集进行登录,当发现该事件的数据集与另一事件的数据集相交时,则判断它们将发生冲突,而不让新事件投入,否则将使它运行。

改进时标法的形式描述如下

分布调度登录

BEGIN

READ (message) from query or  
updata;

CASE message cf

trans begin;

BEGIN READ ( $t, ID$ ); READ  
every set in transaction  
into workspace  $\langle \rangle$ ;

END;  $\langle \text{begin} \rangle$

trans end;

BEGIN READ first workspace  
 $\langle \rangle$ ;

delete  $\langle \rangle$  from TRAN-  
SACTION SET;

```

END, {end}
tuple;
BEGIN READ (operation coad,
            set, succ) into next
            workspace < >;
< > = ID + t + site + < >;
IF WAIT QUEUE is empty THEN
BEGIN put < > into WAIT
        QUEUE; N := 1;
END;
ELSE BEGIN i := 1;
        WHILE < t, ID > < < t(i),
            ID(i) > AND i <= N
        DO i := i + 1;
        insert < > into WAIT QU-
            EQE; N = N + 1;
        END;
END < tuple >
END

```

并行控制模块描述

```

BEGIN
GET first < > from WAIT QU-
    EQE;
serch TRANSACTION SET accorand
    to REMAIN POINT;
IF < t, ID > is min THEN
BEGIN execut < >;
    write < t, ID, OP, SUCC > into
        LOG
    accorand to set;
IF REMAIN POINT not in end of
    LOG
THEN wait < >
ELSE for every transaction (i)
    with t > t(i) OR
    t = t(i) AND ID > ID(i) in
        TRANSACTION SET;
IF set(i) & set = empty
THEN BEGIN execute < >;
    write < t, ID, OP, SUCC >
        into LOG
    accorand to < set >;
IF SUCC = "0" THEN
    move REMAIN POINT
        to end of LOG;

```

```

END
ELSE wait < >;
EDN

```

END

本算法是会话时标法和静态予分析法的结合,比较它们的性能,本算法的响应时间优于会话时标法,在网络站点少于20个时,更是如此。其次本算法亦避免了基本时标法在全部数据项的时标,节省了开销。还具有可消除重启动造成的系统不稳定,减少会话及延迟等待时间等优点。

## 七、结 束 语

ODDBMS 的出现标志着局网技术与数据库技术的结合和进一步发展,虽目前国内的研究工作还刚兴起,但已表明它具有明显的优越性和实用性。

本文仅介绍 ODDDBMS 的几个内容,对诸如数据库操作,数据库实施,安全性控制和故障恢复等问题,限于篇幅,均未列入。

由于本系统硬件环境为欧米网络及个人微机 IBM-PC,通用性强。软件设计合理、有效,用户使用操作方便灵活,可靠性较好,可适合多种应用场合需要。

## 参 考 文 献

- [1] C. J. Date, 《An Introduction to Database Systems》, Third Edition Addison—Wesley, 1983, 3.
- [2] T. Landers and R. L. Rosenberg, 《An Overvien of MULTIBASE》, Distributed Databases, H. J. Schneider, ed., North-Holland, 1982.
- [3] 王以和,涂小平编,《分布式数据库系统》,电子工业出版社,1988.
- [4] 谢驹谟,汤秋萍,局部网络中分布式数据库的设计,《计算机技术》,1986年第5,6期。



# Auto CAD的快速图形输入

杭州电子工业学院计算中心 茹志富

**摘要** 本文阐述利用在Auto CAD菜单文件中设置Auto Lisp 自定义函数的方法来实现从图形输入板快速输入图纸。文中叙述了输入图纸所必须解决的问题及提高输入速度和精度的措施。这对于开展CAD方面的工作具有实用参考价值。

## 一、问题的提出

把图纸输入到计算机,常常是用户着手CAD工作时首先感兴趣的事,这也是CAD的一项基础工作,是一个CAD应用系统的组成部分。

Auto CAD是当前微机中广为流行的通用绘图软件。在Auto CAD的TABLET方式下,使用其绘图命令就可以把图纸从图形输入板输入到微机中。然而,用原始的Auto CAD命令来输入图纸有两个缺点:其一是要求操作者必须对Auto CAD相当熟练,这要化费很多的时间和精力;其二是由于Auto CAD的每条命令只能绘制一个简单

实体,而每条命令执行时要求操作员的干预又甚多,故输入图形的效率很低。

由于每个行业都有自己常用的几何图形及其习惯画法。因此,若用Auto CAD内嵌的Auto Lisp语言设计完成某一特定功能的函数,再把这些函数及常用的文字组织到一个菜单文件中,通过指定菜单项来输入图形,就能实现形象输入、集成输入,从而有效地克服上述两个缺点。

## 二、系统结构

本文以机械行业为对象,图纸输入系统的逻辑结构框图如图1所示。

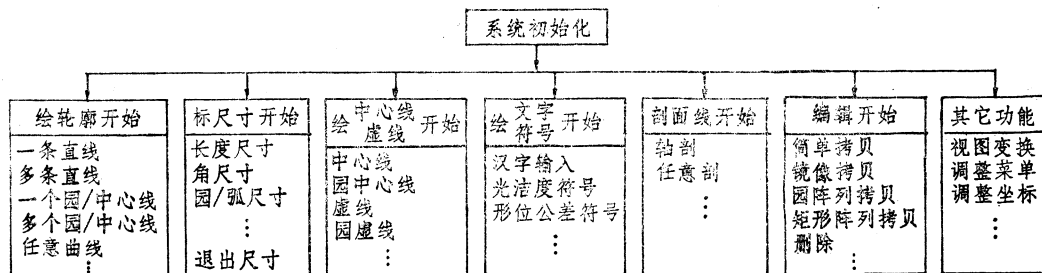


图1 图纸输入系统的逻辑结构

系统初始化框是一个名为“KS”的自定义函数,一张新图纸要求输入时,首先必须执行此函数。该函数的功能如图2所示,其中每一个功能框其实也是一个自定义函数。

定义“绘图单位”是输入图形的基础。Auto CAD中的绘图单位是一个抽象的长度

单位。一个绘图单位所代表的实际长度,随应用行业的要求而异。在机械行业中,用mm作单位,若把绘图的X边界定为0到390,Y边界定为0到287,则就把一个屏幕的图形区定义为3号图纸的大小。这里,一个绘图单位为1mm。包含在KS函数内的1号框就是用来执行此任务,其函数名为

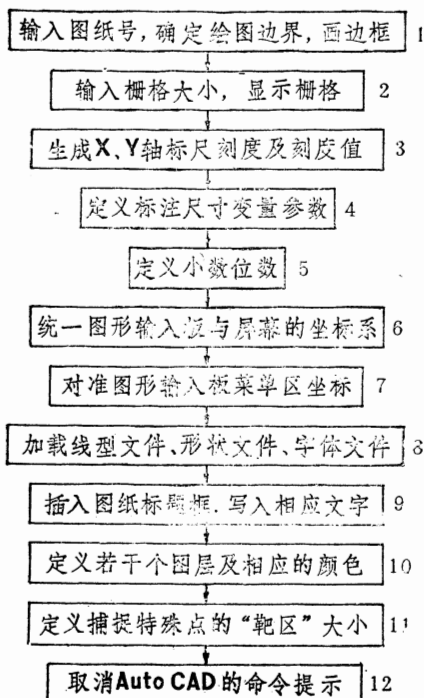


图2 KS函数的功能

“OUTSIDE”的程序如图3所示。

如此定义之后, Auto CAD 提供的标准线型文件、标准字体文件、尺寸变量值、显示栅格及标尺大小等等长度数据都应作相应的修改, 以便与用户定义的长度单位的实际值相适应, 并且符合本行业的制图规范。例如, 代表尺寸线箭头长度的尺寸变量 DIMASZ 原始值为 0.18 长度单位, 应按机械制图规定, 改为 5 长度单位, (即 5mm)。框 2、3、4、8、11 就是完成此类工作。

框 10 定义了若干个图层, 不同类别的实体被置于不同的图层上, 并使用不同的颜色。例如, 图的轮廓线在 0 层上, 用黄色; 文字及标注尺寸在 1 层, 用红色等等。分层绘图的好处是: 屏幕上显示的图形清晰; 编辑图形时不致发生破坏临近其它图形等不必要的麻烦; 绘图机绘图输出时便于选用不同粗细的笔, 使图纸符合要求。

```

(DEFUN OUTSIDE (/ NO)
  (SETQ NO (GETINT "\nEnter number of picture(0,1,2,3,4):"))
  (COND ((= NO 0) (SETQ X 1154 Y 821))
        ((= NO 1) (SETQ X 806 Y 574))
        ((= NO 2) (SETQ X 559 Y 400))
        ((= NO 3) (SETQ X 390 Y 287))
        ((= NO 4) (SETQ X 267 Y 200)))
  (COMMAND "PLINE" '(0 0) (LIST X 0) (LIST X Y) (LIST 0 Y) "CL")
  (COMMAND "ZOOM" "W" '(0 0) (LIST X Y))
  (COMMAND "LIMITS" '(0 0) (LIST X Y))
  (COMMAND "LIMITS" "ON")
)
  
```

图3 1号框程序

执行了 KS 函数后, 就可以进行图形输入。但在进入任意一个子模块进行绘图之前, 必须先执行该子模块的“开始”程序, 以

便保证每一类型的绘图都在相应的图层中进行。这里仅举尺寸标注开始函数 DIM—KS 程序例子予以说明, 其程序如图 4 所示。

```

(DEFUN DIM-KS ( )
  (COMMAND "LAYER" "S" "2" "")
  (COMMAND "LINETYPE" "S" "BYLAYER" "")
  (SETQ N (GETREAL "\nEnter scale factor:"))
  (SETQ C (GETSTRING "\nUsing dimension of system?(Y/N):"))
  (COMMAND "DIM")
  (COMMAND "DIMSCALE" N)
)
  
```

图4 尺寸标注函数DIM—KS程序

### 三、提高输入速度措施

集成化输入是提高图形输入速度的有效途径。为此, 用户应该充分地分析本行业几何图形的组成, 通过归纳总结来设置各个自

定义函数应完成的特定功能。例如, 在机械行业中, 可分别设置“园及其中心线”、“直线及与其相切的弧”、“正六边形及其内切园和中心线”等等为各自定义函数。显然, 如何设置这些函数的功能, 直接影响系统的

规模、使用的方便及输入效率,用户必须精心考虑。

尽量减少函数执行过程中操作员干预,也能大大加快输入速度。为此,首先是在设置自定义函数时,其功能应尽可能单一。例如,本系统把画一个圆及其中心线为一个函数C1,而画若干个同心圆及其中心线为另一个函数CN。其次,在设计自定义函数时,应尽量利用系统本身的计算来取代人工操作,以获取“点”及其它实体的几何参数。例如,函数LINE—GJD 是用于把光洁度符号垂直于直线 $P_1P_2$ 加到 $P_0$ 点上,如图5所示。为此需要计算光洁度符号的转动角A。

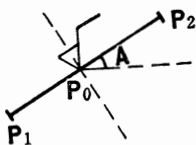


图5 光洁度符号加到 $P_0$ 点

显然, $P_0$ 点必须由操作者指定,而计算角A的 $P_2$ 点若也是人工指定,则程序就大为简化。可是为了省掉这一操作,用ENSEL标准函数(而不是用GETPOINT函数)在操作者指定 $P_0$ 点的同时,获取 $P_0$ 点所在直线的两个端点,其中在 $P_0$ 点右方或者上方的点就是计算角A的第二个点。函数LINE—GJD程序如图6所示。

```

(DEFUN LINE-GAI ( )
  (SETQ N (ENTSEL "\nSelect a point on LINE:"))
  (SETQ P0 (LAST N) EN (CAR N))
  (SETQ ET (ENTGET EN))
  (SETQ P1 (ASSOC 10 ET))
  (SETQ P2 (ASSOC 11 ET))
  (IF (> (NTH 1 P1) (NTH 1 P2))
    (SETQ P3 (CDR P1))
    (SETQ P3 (CDR P2))
  )
  (IF (= (NTH 1 P1) (NTH 1 P2))
    (IF (> (LAST P1) (LAST P2))
      (SETQ P3 (CDR P1))
      (SETQ P3 (CDR P2))
    )
  )
  (SETQ A (/ (* (ANGLE P0 P3) 180) P1))
  (IF (> (LAST P3) (LAST P0))
    (SETQ A (- A))
    (SETQ A (- 360 A))
  )
  (COMMAND "SHAPE" "GAI" P0 2 A)
)
  
```

图6 函数LINE—GJD程序

画轴的剖面图函数ZB,只要求操作员指定轴的两个端点 $P_1$ 和 $P_2$ 即可,如图7所示。中点 $P_0$ 、三条弧及为指定画剖面线区域的两条闭合弧,都可以设法通过计算后直接自动画好,从而明显地减少人工操作次数。

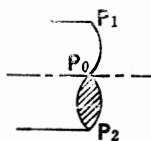


图7 轴剖面图

提高输入速度的又一个措施是在函数执行过程中必要的操作员干预,尽量用指定图形输入板上的菜单项来代替键盘的击键,使得操作单一化。为此,可在菜单文件中设置一些常用的字符,如数字、符号、Yes、No及单点捕捉的关键字END、CEN、MID等等。而在图形输入板菜单上,则应该设置相应的菜单项。

#### 四、提高输入精度措施

提高输入图纸的精度,也是设计系统及操作员工作时应予充分考虑的。

在本系统中设置了视图变换函数,采用在图纸上开窗口的办法,把复杂而稠密的图纸区域充满整个屏幕,以便于观察图纸输入的精确性。

在小号图形输入板上输入大号图纸时,则首先应该把图纸分成若干个小块,并把整张图纸作为一个坐标系在每一块图中标上任意两个非正交的点的坐标值,然后分块把图纸贴在图形输入板上,用块内两个已知点的坐标来统一图纸与屏幕的坐标,即可输入其上的图形。这样,输入的精度仅取决于图上所标坐标点值的精度及用游标对准此坐标点的精度。

对于对称图形,应尽量采用拷贝、取镜像等方法。在操作过程中,凡是要求输入坐标点的地方,都可以根据需要选用特殊点捕

捉功能,例如端点、交点、圆心等等,以便确保输入的点准确地落在要求的位置上。

“正交”功能可以保证输入水平线及垂直线的准确性。在用自定义函数取代Auto CAD原始命令输入图形时,Auto CAD的正交功能失效。用户可以自己设计程序来实现正交

功能。其策略是:由两点所定的直线与水平及垂直四个方向比较,若误差小于某一角度,例如 $4^\circ$ ,即 $0.07$ 弧度,则认为是输入误差,对该直线的终点予以校正,使之成为正交线,反之则不予校正。正交函数ORTH程序如图8所示。

```
(DEFUN ORTH (/ D)
  (SETQ A (ANGLE P1 P2))
  (SETQ D (DISTANCE P1 P2))
  (COND
    ((OR (AND (> A 0) (< A 0.07)) (> A 6.21)) (SETQ A 0))
    ((AND (> A 1.50) (< A 1.64)) (SETQ A (/ PI 2)))
    ((AND (> A 3.07) (< A 3.21)) (SETQ A PI))
    ((AND (> A 4.58) (< A 4.78)) (SETQ A (+ PI (/ PI 2)))))
  )
  (SETQ P2 (POLAR P1 A D))
)
```

图8 正交函数ORTH程序

此函数在画多条连续直线及尺寸标注函数等都有应用,为了简化程序设计,我们把此功能单独设计为一个函数,称之为“公用函数”。

在程序设计中,无论工程计算还是绘图,“数组”是经常需要的,而Auto Lisp没有提供这类数据结构。本系统设计了定义数组的公用函数DMS及调用数组变量的公用函数CALL,其程序如图9所示。图10所

```
(DEFUN DMS (P I)
  (READ (STRCAT P (ITOA I)))
)
(DEFUN CALL (P I)
  (EVAL (READ (STRCAT P (ITOA I)))))
)
```

图9 公用函数DMS和CALL

示的函数CN,是用户输入园心及n个同心园的园周上任意一点,画n个同心园及其中心线,以示DMS函数及CALL函数的应用。

```
DEFUN CN (/ N PO I R)
  (SETQ N (GETINT "\nEnter number on circle:"))
  (SETQ PO (GETPOINT "\nEnter center of circle:"))
  (SETQ I 1)
  (SETQ R '( ))
  (REPEAT N)
    (SET (DMS "P" I) (GETPOINT "\nEnter a point on circle:"))
    (SET (DMS "R" I) (DISTANCE PO (CALL "P" I)))
    (COMMAND "CIRCLE" PO (CALL "R" I))
    (SETQ R (CONS (CALL "R" I) R))
    (SETQ I (+ I 1))
  )
  (SETQ R (CONS 'MAX R))
  (SETQ R (+ (EVAL R) 5))
  (SETQ P1 (POLAR PO PI R))
  (SETQ P2 (POLAR PO 0 R))
  (SETQ P3 (POLAR PO (/ PI 2) R))
  (SETQ P4 (POLAR PO (+ PI (/ PI 2)) R))
  (COMMAND "LAYER" "S" "1" "")
  (COMMAND "LINETYPE" "S" "CENTER" "")
  (COMMAND "LINE" P1 P2 "")
  (COMMAND "LINE" P3 P4 "")
  (COMMAND "LAYER" "S" "0" "")
  (COMMAND "LINETYPE" "S" "BYLAYER" "")
)
```

图10 函数CN程序

通过指定若干个输入点来输入图纸中任意曲线,也要使用数组函数,此处不详述了。

在设计函数及输入操作时, 还要注意克服绘图随意性问题。例如, 在标注较小的长度尺寸时, 所绘的尺寸线及尺寸文本就可能与原图不一致; 又如若用园心及弧的两个端点画弧时, 操作者若不按逆时针方向输入弧的起点和终点, 所绘之弧就不一定是需要的。用Auto CAD画图案的方法画复杂图形的剖面线, 也会产生混乱。

## 五、实 现

把包括公用函数在内的所有自定义函数组织在 LISP 文件中。把直接用于图形输入的函数的函数名及常用字符按功能分类组织到菜单文件中, 如图11所示的ACAD.MNU示意菜单文件, 开机后 Auto CAD 将自动加载它。在图形输入板的有效图形输入区的周边, 写上与菜单文件中各项的顺序和功能相一致的均匀排列的菜单项功能, 在菜单项旁边的无效图形输入区, 可以画上执行每一个菜单项的所绘之图及输入要求。对于绘制新图, 在进入 Auto CAD 之后, 首先应对准菜单坐标, 本系统是用游标按钮 4 来调用 (TABLET—CFG) 函数实现。其后贴上图纸, 指点与 (KS) 函数相应的菜单项, 就可以使用这些菜单来实现图纸输入了。为了扩大屏幕上图形显示的有效面积, 建议重新配置Auto CAD, 取消系统缺省设置的屏幕菜单区。

## 六、结束语

综上所述, 笔者就开发图形输入板的图形输入系统中遇到的几个问题作了简要的论述。也许未必完美, 但确能保证图纸输入的精度和大大提高工作效率, 而且使用者只需

经稍许学习就能应用。笔者认为, 按画法几何原理及行业制图规范, 加入人工智能方法来输入图形, 必将进一步提高输入的速度和精度。

```

TABLET1 ; 板菜单区1
(KS) ; 绘图开始
(OUTSIDE-KS); 轮廓线开始
(L1) ; 一条直线
(LN) ; 多条正交/非正交线
(C1) ; 一个圆及中心线
(CN) ; 多个同心圆及中心线
(LAT) ; 直线及其相切弧
(POLY-PN) ; 任意曲线
(RM) ; 螺母
(CD-KS) ; 中心线/虚线开始
(CENTER) ; 中心线
(HIDDEN) ; 虚线
***TABLET2 ; 板菜单区2
0
1 ; 数字
. ; 小数点
R ; 半径R
%%c ; 直径φ
%%d ; 角度°
END ; 捕捉端点
MID ; 捕捉中点
Y ; 回答“是”
N ; 回答“非”
***TABLET3 ; 板菜单区3
(DIM-KS) ; 标尺寸开始
(HVA) ; 标长度尺寸
(CORNER) ; 标角度尺寸
(R-D) ; 标直径/半径
(DIM-EXIT) ; 退出标尺寸
(FH-KS) ; 符号开始
(LINE-GJD) ; 直线上标光洁度符号
(CIRCLE-GJD); 圆周上标光洁度符号
(B-KS) ; 剖面开始
(ZB) ; 轴剖
(RB) ; 任意剖
(EDIT-KS) ; 编辑开始
(COPY1) ; 拷贝
(MIRR) ; 镜像
***BUTTONS ; 按钮菜单
(COMMAND "ERASE" "L" "" ); 按钮2
(VIEW) ; 按钮3, 视图变换
(TABLET-CFG); 按钮4, 对准菜单区座标

```

图11 ACAD.MNU 示意菜单文件

# 在 Turbo C 环境下 开发遥感图像显示系统

浙江大学地球科学系89研 高峰

**摘要** Turbo C 的集成开发环境是十分友好的,特别是其1.5以上版本提供的图形函数功能,为遥感图像显示系统的开发提供了十分方便的工具。以C语言编写的显示系统大大缩短了软件开发周期,增加了系统的可维护性,提高了系统的效率,本文论述了其软件的设计及实现。

## 一、问题的提出

随着国内外卫星遥感技术的迅速发展,大量的卫星遥感数据源源不断地送回地面,如何使之成为资源调查、环境测量、区域开发、规划及决策的重要工具,更有效地发挥其经济与社会效益,则是遥感图像应用处理工作的重要任务。

遥感图像应用处理的主要技术之一是计算机数字图像处理,它与光学处理、目视判读一起可以完成多种应用工作。目前国内遥感图像显示及处理是在专用图像处理系统上进行的,如美国 IIS 公司生产的M75 图像处理机。但这些设备和配套软件的价格都很昂贵,国内拥有这些设备的单位不多,这给遥感资料的有效利用带来了困难。随着微机性能的不断提高,越来越多的单位开始利用微机开发遥感专用微机系统。这些系统软件是以 Fortran 和汇编语言编写的,其不足之处在于每次处理的图幅较小,除去附加的图例及说明,显示区会更小,这对遥感图像处理是很不方便的;而且其处理速度慢,软件开发周期长,其可理解性和可维护性差,难于扩充功能。

C语言是一种结构化、模块化的语言,

具有良好的可移植性,其空间和时间效率都很高。Turbo C 1.5 以上版本提供了图形功能,它支持CGA卡、MCGA卡、EGA卡、VGA卡、IBM8514 卡,其中支持 EGA 卡的EGAHI模式,分辨率为  $640 \times 350$ , 16 种颜色、分 2 页显示;支持VGA 卡的VGAHI模式,分辨率为  $640 \times 480$ , 16种颜色,分 1 页显示;支持IBM8514 卡的 IBM8514HI 模式,分辨率为  $1024 \times 768$  象元,它支持256种颜色的调色板,并提供了红、绿、兰、三色合成函数。Turbo C的图形功能主要包括图形系统控制、色彩控制、绘图和着色、图形屏幕管理、图形模式下的正文输出、图形模式中的错误处理以及状态询问等功能。Turbo C提供的这些功能为遥感图像显示系统提供了极为方便的开发工具。

## 二、系统设计

对一幅遥感图像进行图像操作主要包括图像的变换操作、取子图、取窗口,放大、取控制点以及显示小块区域的灰度值等。为方便以后的拍照成图或打印输出,还应具有专用系统的特色,可以改变图像上某类象素的颜色、加或去对焦线、加或去栅格线、加或去图例的功能。为了能显示大幅图形,还



需要具有卷轴功能, 并可以任意选取一定的区域进行图像操作, 本系统的设计考虑到不同的图形卡配置而设计为 EGA 卡和 VGA 卡

上的通用系统。综合上述显示系统的功能, 根据数据流程图, 设计出系统主控流程及模块如下图所示。

### 三、软件实现

根据以上的设计, 分别对各模块进行详细设计并编码。现分述其主要功能模块和主要技术问题。

#### 1. 卷轴 (scroll) 功能的实现

本系统设计的卷轴功能可以使用户一次对一幅  $640 \times 960$  (VGA) 或  $640 \times 700$  (EGA) 图幅大小的图形进行操作。其实现原理为将区域位图像保存到内存中, 然后再回放出来。Turbo C 的图形处理库函数提供了将一指定区域位图像保存到内存中去的函数。

Void far getimage(int left, int top, int right, int bottom, far \* bitmap)

它可将一幅由 left、top、right、bottom 所定义的矩形区域的位图像存储到主存储器中。bitmap 指向主存储器中将存储这个位图像的区域。在保存之前先要为此区域分配存储的主存。

unsigned far imagesize (int left, int top, int right, int bottom)

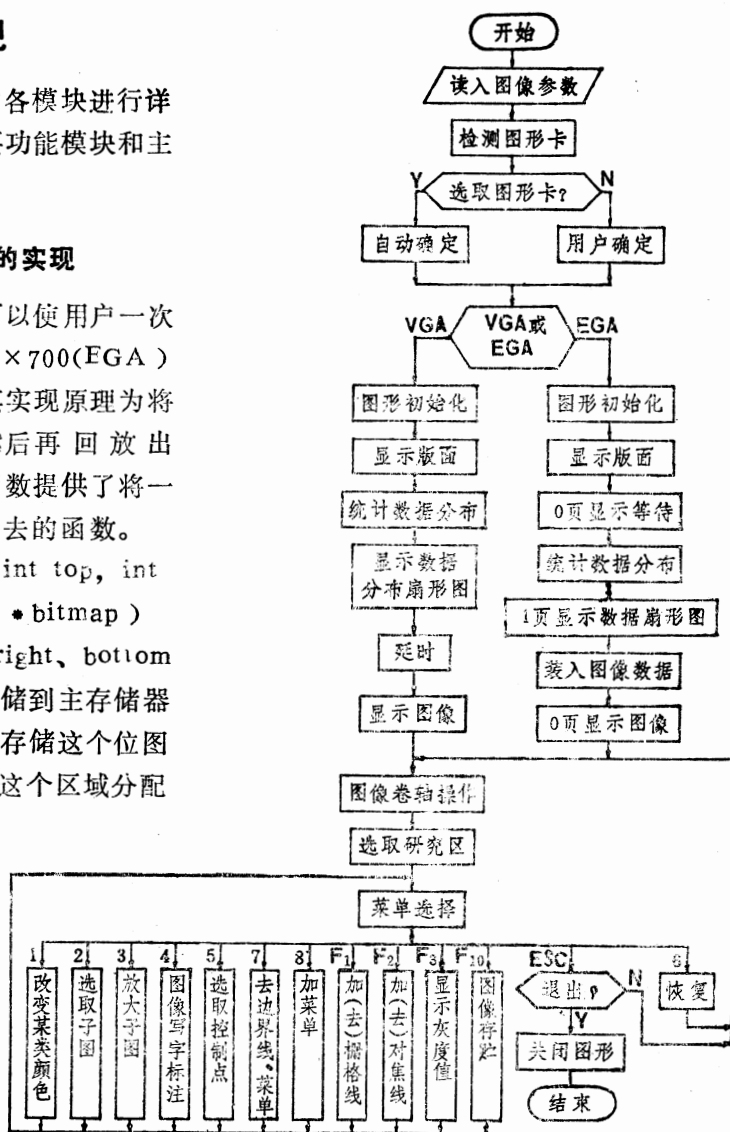
这个函数将返回存放指定矩形所需的字节数, 再用 void \* malloc (unsigned size) 分配主存空间。保存

位图的一个条件是位图像的大小必须小于 64K, 否则 getimage 和 putimage 将返回空。

图像重新回放可调用函数

void far putimage(int left, int top, void far \* bitmap, int op)

它将回放图像的左上角定在 (left, top) 处, op 是组合操作, 它提供了复制、



“异或”、“或”等操作功能。

Turbo C 提供了 6 种存储模式——极小模式、小模式、中模式、紧凑模式、大模式和特大模式, 其中紧凑模式的代码最多为 64K, 而数据可为 1MB; 大模式的代码和数据均用远指针, 均可达 1MB。由于遥感图像数据量很大, 故可采用紧凑模式或大模式

来完成设计, 这样可将整个屏幕分成若干个块进行存贮、回放, 而使每一块的大小均小于64K, 对于VGA卡在VGAHI模式下可将屏幕分为6块, 每块大小为 $640 \times 80$ ; 对于EGA卡在EGAHI模式下可将屏幕分为5块, 每块大小为 $640 \times 70$ , 这样设计一个保存两页大小图像的系统就分别需要12块和10块, 只要总数据量小于1MB, 还可使一次显示的图幅更大。由于保存位图到内存的时间和由内存再回放的时间都很短, 恢复一幅保存的 $640 \times 480$ 的6块位图像在IBM PC 286微机上仅用1—2秒就可完成。这样基于Turbo C就完全有可能完成卷轴功能, 而不至于使用户感到时间上的难耐。图像上卷一次则将保存的位图依次向上移动一块; 下卷一次则将保存的位图依次向下移动一块。每次卷动都要计数, 向上卷加1, 向下卷减1, 这主要是为控制卷动次数和以后取控制点的需要。

## 2. 选取图像控制点

对遥感图像进行处理时, 一般首先要对遥感图像进行几何精纠正, 这是因为卫星对地面进行扫描时, 由于扫描器、地球等因素的影响, 使图像有畸变。校正的方法是从遥感图像上选取显著地物象素, 得到其坐标值, 再从地形图上找到其对应物体, 求得其坐标值, 经坐标转化后, 用一次或高次曲线拟合, 根据求出的对应关系再用重取样的方法对图像重新进行空间位置和灰度的调整。

对图像上某些特殊的控制点(如水库、水坝、桥梁、高大建筑物等), 我们要求找出它在图像上的行、列坐标。本系统设计了可以上下左右移动的光标, 光标点移动一个位置, 就返回它所在的行、列坐标及颜色值, 获取一点的颜色灰度值可调用`int far getpixel (int x, int y)`函数的返回值来完成, 有时可能分辨不清其具体位置, 这时还可求助于`boxprint`功能, 选取该象素周围 $24$

$\times 25$ 个象素, 显示其灰度值, 从而可以通过其灰度值进一步确认其精确坐标位置。

## 3. 图像的放大功能

有时, 需要对图像的某些部分进行更详细的观察, 可以通过设计的放大功能来实现。其实现原理是取放大区域上一象素的灰度, 将一个象素放大为若干个象素, 以同样的灰度显示出来, 根据需要可以放大为4个象素(2倍)、9个象素(3倍)、16个象素(4倍)或更大。功能流程是首先通过移动+字光标确定放大区域, 然后恢复初始图形模式, 回放放大区域, 最后显示放大图像。

## 4. 改变某类象素的颜色

在显示中有可能相邻类区颜色的界线分辨不清; 或是显示色彩不合要求, 一般视觉总是想以蓝色表示水体, 绿色表示植被; 有时为提取某种信息, 则要求其它信息隐去。这些可以通过改变调色板的方法来实现颜色的改变。函数`void far setpalette(int index, int actual-color)`为我们提供了方便, 一旦改变了一种调色板颜色, 屏幕上出现的所有那种颜色的象素都将改为新的颜色。如果`size`是当前调色板的项数, 则`index`可在0到(`size - 1`)间变动, EGA的硬件允许显示64种颜色(0...63), 故而EGAHI模式中调色板有16项, 每项的值为0...63, 因此通过设置调色板改变颜色还可使图像的色调变得丰富起来, 用设置调色板的方法改变颜色, 其速度十分快, 可实时完成。提取某种信息, 只需将图像上除该信息外的其它所有信息的颜色设置成底色即可。另外利用这一功能, 还可将图像上的若干类合并, 使用十分灵活。

## 5. 在显示的图像上写字标注

Turbo C为图形模式下的正文输出提供了十分灵活的多种字体、字形、大小和方向的字库和输出函数, 利用它所提供的输出函数, 设计系统的写字功能可以在屏幕图像上

书写多种颜色, 1—10倍于字符大小(8×8像素), 多种字体、垂直和水平两个方向的字符。

在图形方式下输入数字和字符时, 存在的一个问题是正文到达屏幕底部时, 图像也会随之滚动, 而Turbo C中并没有提供图形方式下的正文定点输入功能, 为解决这个问题, 编制了两个图形方式下的新的输入函数int readnum(int x, int y)和int readstr(int x, int y, char str[ ]), 用以在(x, y)定点处开始读取数据和字符串, 如果设置了颜色void setcolor(int color), 那么以后的输入数据及字符将以该颜色回显, 这两个函数实现的原理很简单, 现以readstr函数为例说明:

①先置str内容为空, 以getch()获取一字符, 置i为0。

②多路判断为何字符。若为退格键←且i大于0, 则清除最后一个字符, 在该位置加字符串结束标志'\0', 屏幕上以底色在最后字符的位置画一小矩形, 消去该字符, 并且i--, x=x-8, 否则break退出。

若为其它可显示字符, 则以函数void far outtextxy(int x, int y, char far\* textstring)在(x, y)处回显, 将该字符加入到字符串中, strcat(str, "字符"), 且i++, x=x+8。

若为回车键则表示字符串输入结束, 否则继续等待输入。

③返回i

str为输入的字符串数组, int readstr(int x, int y, char str[ ])返回的是字符串长度。int readnum(int x, int y)则返回图形方式下定点输入数。

## 6. 调试及运行

系统设计所采用的图形卡可以是自动检测, 也可以是自己规定, 自动检测将选取最高分辨率的图形工作方式。对于EGA在

EGAHI模式下可以分2页显示, Turbo C提供了为图形输出设置活动页的图形函数void far setactivepage(int pagenum), 它使pagenum页变为活动页; 以及设置可见图形页号函数void far setvisualpage(int pagenum), 它使pagenum页显示, 这样可以快速显示屏幕后的图形。本系统设计利用了它的这一特点, 0页显示时1页活动, 等1页运行完毕, 再转换, 这样可以使用户不感到运行中系统装入数据时的“等待”。

在图像进行放大操作后, 考虑到其图像位置的变动, 系统限制了进行菜单选择5、F1、F2的运行, 对于显示小区域的灰度值后, 系统限制了仅可运行8(加菜单)、6(重恢复)操作。

系统采用菜单选择, 具有模块化结构, 充分体现了交互性原则。在对单个模块功能测试后, 又进行了联合测试, 运行状况良好, 达到了遥感分类图像显示操作的需要。试用结果表明它的运行速度较以往系统有所提高。

## 四、结 论

在Turbo C集成环境下开发出的遥感图像显示软件较以前的显示系统具有一些新的特点。程序设计的模块化和优良的接口, 使该系统的可读性及可扩充性明显提高, 其执行速度要较Fortran调用汇编子程序的系统快得多, 其可执行代码与源程序代码所占的空间比为1:8, 用C语言编制的系统其可移植性强, 更重要的是它所提供的一些图形函数大大缩短了我们开发软件的周期, 这比以汇编语言开发优越得多。

在调试中发现, C语言的安全性不强, 但只要在系统中加以控制是完全可以避免这种不安全状态的。

尤其是Turbo C还支持IBM8514卡, 它

# dBASE FOXBASE 数据库 结构的优化设计

衢州化学工业公司计算机中心 秦学礼

**提要** 本文根据笔者应用 dBASE、FOXBASE 数据库系统设计开发管理信息系统的经验和体会,对数据库结构的设计进行了分析和研究,给出设计中的优化方法及应用效果。

**关键字** 数据库系统 管理信息系统  
数据库结构 优化设计

## 一、引言

随着微型计算机在我国的普及应用,适用于微型机的关系数据库系统也被用于开发各种管理信息系统,如目前普遍使用的 dBASE、FOXBASE 关系数据库系统。由于受到微机内、外存贮器容量的限制,在编程和使用维护诸方面都存在不尽人意之处。因而设计者一般采用牺牲某一或几方面来弥补其中要求较高的方面,使得设计的应用系统能在固有硬件环境下,取得较满意的效果,这就是本文所要阐述的内容。

## 二、数据库结构设计的优化

在开发管理信息系统中,数据库结构设计在系统开发中占据非常重要的位置。数据库设计的好坏直接影响着整个系统的效率及资源占用。确定数据库结构则是数据库设计中的关键环节。

一个完备的数据库结构,往往要经过需求分析,既通过对对象业务活动进行分析,了解用户环境,确定系统边界和分析系统内部的结构。关系数据库结构的逻辑设计的有力工具是 E. F. Codd 提出的“关系规范的

理论”。

范式(Normal Form)的概念是 Codd 在 1971~1972 年提出了 1NF, 2NF 和 3NF, 讨论了进一步规范化的问题。1974 年, Codd 和 Boyce 共同提出了 BCNF (Boyce Codd NF), 1976 年 Fagin 提出了 4NF, 现在又有人提出了 5NF。

关系数据库的关系要满足一定的要求, 满足不同要求的为不同范式。满足最低要求(关系的 6 条性质)的为第一范式(1NF)。在 1NF 基础上再满足一些要求的为第二范式(2NF), 以此类推。一般如果一个关系 R 是 4NF, 它一定也是 3NF, 2NF。这种关系可以用集合表示法表示为

$$5NF \subset 4NF \subset BCNF \subset 3NF \\ \subset 2NF \subset 1NF$$

由此可见, 一个低一级的范式的关系模型, 通过投影分解可以转换为若干高一级范式的关系模型。

规范化的基本思想是逐步消除数据依赖中的不适合部分, 使模式中的各种关系模型达到某种程度的“分离”, 即“一事一地”的模式设计原则。让一个关系描述一个概念、一个实体与实体间的一种联系。若多于一个概念, 就把它分离出去。因此规范化的实质是概念的单一化。关系规范理论的另一个

问题是模式中的冗余问题。规范后的关系模式在某种意义上冗余应该最小,即所谓最小冗余原则。从理论上讲,经规范后的关系最明显的特点有

(1)数据的冗余度小。

(2)数据的一致性好(没有二义性)。

由于dBASE、FOXBASE数据库系统没有提供数据字典管理功能,所以在设计一个应用系统时,首先要采用关系规范理论设计数据模式。根据笔者几年来实际应用体会,完全规范化的数据模式在具体应用中存在着很多问题,归纳起来是数据库统计速度、编程的复杂性、占用磁盘空间等。

### 三、优化设计实例及方法

以笔者在“化工健康监护管理信息系统”的设计开发为例,探讨数据库结构设计的优化问题。

设有职工个体特征关系R。

R[GH(工号),XM(姓名),XB(性别),CCRQ(出生日期),DWBM(单位编码),...]

R关系有反映职工个体特征的信息项29个。

职工健康问答及病史关系S。

S[GH(工号),WDRQ(问答日期),ICD1(疾病一),ZDRQ1(诊断日期一),...,XYSJ(吸烟开始时间),XYCD(吸烟程度),...]

S关系有反映职工病史的信息项22个。

职工体检关系T。

T[GH(工号),TJRQ(体检日期),TZ(体重),...,XL(心率),XDT(心电图检查结果),...]

T关系有职工体检信息项33个。

从R, S, T三个关系我们可以清楚地看出,它们之间都不存在传递依赖关系,其候选码是GH(工号),因而它们是标准的第

三范式。

对于这标准范式数据库结构,在实际应用中,确有很多不尽人意之处。例如我们要按单位、性别、体检日期、大于35岁、心率大于80、每天吸烟大于等于20支和病史问答日期去统计,在FOXBASE和dBASE环境下,实现的算法,运算时间和占用存贮空间等都各不相同。

#### 1.同时关联打开的三(二)个数据库文件

FOXBASE, dBASE提供了

SET RELATION TO [〈表达式〉

INTO 〈别名〉]命令,在FOXBASE环境下可以把一个打开的数据文件按关键字表达式与二个打开的库文件关联,关键字表达式是二个数据库共有的,用别名标识的数据库必须按关键字表达式索引过,每当指定数据库的记录移动时,相关联库的记录指针也移到对应于关键字的记录。dBASE环境下只能实现与一个库的关联。当按要求的条件进行统计时,其相关内容被分别存放在R、S、T三个数据库中,在FOXBASE环境下用F1.PRG程序统计是速度最快,数据库设计最合理,占用的存贮空间最省,可以说是最优的数据库结构。

经试验,R库有992条记录,S库有943条记录,T库有357条记录,在16MHZ的286AT机上运行13秒。

F1.PRG程序清单如图1所示。

#### 2.在dBASE III环境下的统计

在dBASE III环境下,一个打开的库只能与另一个打开的库关联,根据统计条件,要把结果得出来还比较麻烦,有几种统计方法。

```

1: * F1.PRG 1990.11.1
2: ? TIME()
3: SET TALK OFF
4: SET SAFE OFF
5: MDW="04"
6: MXB="男"
7: MSJ=1988.10
8: SELECT 3
9: USE S
10: INDEX ON GH TO S
11: SELECT 2
12: USE R
13: INDEX ON GH TO R
14: SELECT 1
15: USE T
16: INDEX ON GH TO T
17: SET RELATION TO GH INTO B
18: SET RELATION TO GH INTO C ADDITIVE
19: COUN ALL FOR MDW=B->DWBM .AND. ;
   MXB=B->XB .AND. TJRQ<=MSJ .AND. ;
   MSJ-B->CCRQ>=35 .AND. XL>80 .AND. ;
   20>=C->XYCD .AND. C->WDRQ<=MSJ ;
   TO MSR
20: ? MSR
21: ? TIME()

```

图 1

A. 采用 F2.PRG 的统计程序  
F2.PRG 程序清单如图 2 所示。

```

1: * F2.PRG 1990.11.1
2~16句同F1.PRG
17: UPDATE ON GH FROM B REPLACE ;
   XB WITH B->XB, DWBM WITH ;
   B->DWBM, CCRQ WITH B->CCRQ
18: UPDATE ON GH FROM C REPLACE ;
   WDRQ WITH C->WDRQ ;
   XYCD WITH C->XYCD
19: COUN ALL FOR MDW=DWBM .AND. ;
   MXB=XB .AND. TJRQ<=MSJ ;
   .AND. MSJ-CCRQ>=35 .AND. ;
   XL>80 .AND. 20>=XYCD .AND. ;
   WDRQ<=MSJ TO MSR
21: ? MSR
22: ? TIME()

```

图 2

采用 F2.PRG 程序, 就把标准第三范式的关系库 T 扩充为第一、二范式库, 增加 XB, DWBM, CCRQ, WDRQ, XYCD 5 项。程序中的 17, 18 句是把 R、S 库中的数据替换到 T 库, 解决数据的一致性问题。与 1 同样统计条件和原始数据时, 运行结果

为 42 秒。

### B. 将 T 库与 R 库关联

在 T 库增加 WDRQ, XYCD 两个项, 使得原来标准第三范式的关系也变成了非标准。采用 F3.PRG 程序, 运行结果为 53 秒。

F3.PRG 程序清单如图 3 所示。

```

1: * F3.PRG 1990.11.1
2~16句同F1.PRG
17: UPDATE ON GH FROM C REPLACE;
   WDRQ WITH C->WDRQ.;
   XYCD WITH C->XYCD
18: SET RELATION TO GH INTO B
19: COUN ALL FOR MDW=B->DWBM ;
   .AND. MXB=B->XB .AND. ;
   TJRQ <=MSJ .AND. ;
   MSJ-B->CCRQ>=35 .AND. ;
   XL>80 .AND. 20>=XYCD .AND. ;
   WDRQ<=MSJ TO MSR
21: ? MSR
22: ? TIME()

```

图 3

### 3. 采用数据库的联结功能, 在统计时根据需求生成新的临时工作库

联接过程是在一个数据库(当前工作库)中顺序抽出各个记录, 每次抽出一个记录后, 就在另一个以别名指示的数据库中寻找符合某种条件的记录, 每找到一个就与前数据库中抽出的记录进行联结, 形成一个新的记录, 新记录中包含了两个数据库中的字段。在联接操作中, 要求有充裕的磁盘存贮空间, 运行时间还特别长。一般可采用二种方法。

A. 用程序 F4.PRG, 运行结果是 35 分 32 秒。

F4.PRG 程序清单如图 4 所示。

其第 17 句是把 T 库与 S 库的 GH, TJRQ, XL, WDRQ, XYCD 5 项联接成一个新的库, 再与 R 库相关联, 运行时间太长。

B. 用程序 F5.PRG, 同样是采用联结, 但在程序中稍加了一点技巧, 运行时间为 54 秒。



```

1: * F4.PRG 1990.11.1
2~16句同F1.PRG
17:JOIN WITH C TO T4 FIELDS GH,;
    TJRQ,XL,XYCD,WDRQ FOR GH=C->GH
18:USE T4
19:INDEX ON GH TO T4
20:SET RELATION TO GH INTO B
21:COUN ALL FOR MDW=B->DWBM ;
    .AND. MXB=B->XB .AND. TJRQ<=MSJ ;
    .AND. MSJ-B->CCRQ>=35 .AND. ;
    XL>80 .AND. 20>=XYCD .AND. ;
    WDRQ<=MSJ TO MSR
23:? MSR
24:? TIME()

```

图 4

F5.PRG程序清单如图 5 所示。

```

1: * F5.PRG 1990.11.1
2~10句同F1.PRG
11:COPY STRU TO S11
12:SELE 4
13:USE S11
14:SELECT 2
15:USE R
16:INDEX ON GH TO R
17:SELECT 1
18:USE T
19:INDEX ON GH TO T ,
20:COPY STRU TO T11
21:JOIN WITH D TO T5 ;
    FOR GH=A->GH
22:USE T5
23:INDEX ON GH TO T5
24:APPE FROM T
25:UPDATE ON GH FROM C ;
    REPLACE WDRQ WITH ;
    C->WDRQ,XYCD WITH C->XYCD
26:SET RELATION TO GH INTO B
27:COUN ALL FOR MDW=B->DWBM ;
    .AND. MXB=B->XB .AND. ;
    TJRQ<=MSJ .AND. ;
    MSJ-B->CCRQ>=35 .AND. ;
    XL>80 .AND. 20>=XYCD ;
    .AND. WDRQ<=MSJ TO MSR
29:? MSR
30:? TIME()

```

图 5

设计思想是先把 S 库的结构(没有记录)考贝成临时文件,再把SII, T两个库联结为一个新的数据库,然后只要把有用字段的内容追加到新库,语句24, 25, 再与R库关联,实现统计。

从上述几种方法看, F1的数据库结构是标准的第三范式, 程序运行时间最短, 但是如果统计条件与三个以上的数据库相关时, 设计的数据库不可能采用标准第三范式。从  $F_2 \sim F_5$  的实现方法可以看出, 当记录条数较少, 机器的运行环境较好时, 运行速度没有明显的差别, 但  $F_4$  的运行时间让人无法忍受, 所以在设计中应避免采用数据直接联接的方法。如果要联接几个数据库, 应采用  $F_5$  的程序设计技巧。

## 四、几点体会

用dBASE, FOXBASE这样的工具开发较大型的系统, 应在数据库结构的设计中采用优化方法, 否则设计的系统将会出现运行速度慢、占用存贮空间大, 数据容易出现不一致性等, 所以笔者认为, 在设计阶段应做到以下几点。

(1)数据库结构设计应先遵循第三标准范式。

(2)根据系统需要, 主要是统计、查询的要求, 扩充成第一、二范式数据库程序设计中应采用相应的技巧和方法。

(3)根据硬件的环境, 考虑设计的结构是提高运行速度, 还是节省存贮空间。

根据笔者几年的实践经验, 认为只要精心设计和采用相应技巧, 设计的系统可以达到最优。

# 管理信息系统开发的新途径 —原型化方法

36430部队指挥自动化工作站 新 雷

**摘要** 传统的软件生命期开发方法存在一些问题,本文从分析解决这些问题入手,介绍了软件开发的原型化方法,并探讨了将这种方法与SA、SD、FTO方法结合使用的技术。

## 一、引言

软件开发已由原始的个体手工艺演变成集体化、有组织的生产方式。“软件工程”就是研究如何应用科学理论和工程技术,指导软件开发,从而达到用较少的投资获得高质量软件的目的。为此,软件开发的方法有数种,较常见的是“软件生命期”(SA、SD等)方法,这种方法一般要分几个阶段,即用户需求分析、设计、编码、测试、运行、维护。每个阶段所花费的精力见图1。各阶段可能出现错误的大致比例见图2。

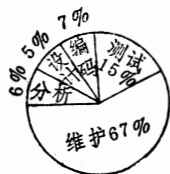


图1 所花精力比例

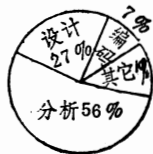


图2 出错比例

由此可知,绝大部分精力用于改错和由于环境变更而进行的程序修改;绝大部分错误及程序修改则出在用户需求分析阶段。

由于传统的软件开发方法,要求开发人员努力按严格定义的模式,用户也要尽可能提出详细、准确的需求,但这样做实际上是很

困难的,因为

①对开发人员和用户在系统开发的早期要求较高、工作量太大。

②各阶段划分太细、不统一,界面太多,图表复杂。

③用户和开发人员各自的知识互相扩散不够,尤其是有的用户对计算机究竟能提供何种服务、解决什么问题不清楚。

④工作中的疏漏,传递信息媒介的局限性,通信中的误解等,都可能造成规格说明书的不完全。

随着用户对计算机的了解,他们的要求常会发生变化。尽管在传统的软件开发方法中,通过加强复审与确认可减少上述问题,但都未能很好地解决。特别是在开发的中、后期,是绝对不容许用户需求再有变化。对于这些问题,只有用新的技术和方法才能解决。

## 二、原型化方法

上面的分析表明,传统的软件生命期开发方法确有一些缺点。于是人们思考:传统的系统分析方法能得到准确的规格说明书吗?光凭细心能避免各种错误吗?因此,提出所谓“原型化方法”。

### 1. 原型化方法的概念

“原型”最早是指模仿某种产品的原始

雏形。把这一概念引用到信息系统的开发中,“软件原型”就是该系统的一个“早期版本”,含有最终系统的主要特征。它要求经过简单分析,迅速构造出一个小型软件系统,满足基本要求,使用户在试用过程中受到启发,并与开发人员一起,加强通信和反馈,针对具体原型提出改进意见,量化需求分析。同时逐步纠正分析中的错误,确定各种细节。

## 2. 几种原型方式

由于系统的特点和各个不同的开发阶段,通常按要达到的目的,可分为三种不同的原型。

①探索型 其目的是搞清用户对系统的要求,激发出好的设计思想。在系统开发的前期,开发人员对应用领域了解不多,用户对计算机也知之甚少。这种原型能给出一个功能的实际演示,在开发人员与用户之间架起一座易于沟通的桥梁,共同探讨多种方案的可行性。

②实验型 在决定用以上方案去全面开发系统之前,用来进一步检查方案的正确性,看其规格说明是否可靠。

③演变型 它强调使系统逐步地适合变化着的环境,在改进原型的过程中从最基本的核心入手,不断改进充实,将其演变成合乎用户要求的最终应用系统。

## 3. 建立原型的求与方法

原型的建立是开发人员与用户紧密配合的成果,是反复进行的过程。从一个只能完成基本功能的样本开始,根据用户的要求,不断完善,直至最后达到用户满意。由于原型不同于最终系统,它要求迅速实现,投入运行,故要注意“原型功能”和“系统性能”的区别,从而正确取舍。既要忽略一切暂时不太重要的部分,又要让用户充分地认识原型系统的作用。

通常,原型的建立过程如图3所示。

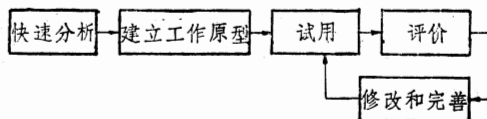


图3 原型的建立过程

①快速分析 在开发人员与用户的通力合作下,迅速确定用户对系统的基本要求,根据原型所要表明的特性,(如系统的主要性能、界面形式、总体结构及原型性能等),使用户易于接受的通俗语言描述基本规格说明,以满足开发原型的需要。这里特别要注意围绕运用原型的目标,集中力量,确定局部的需求说明,为尽快建立原型打下基础。这一阶段类似于传统开发方法的需求“分析”阶段,但在分析的层次上有所不同。

②建立工作原型 根据上述分析,尽快建立一个可运行的系统原型。这里要求具备有力的软件工具和开发环境(如MISAD系统工具等),忽略某些细节及一切次要内容(如安全性、出错处理、意外处理等),主要考虑原型能否充分反映用户所关心的主要问题。

③原型的试用 这一工作主要应由用户亲自操作,开发人员进行指导。由于在建立原型时,强调了快速,忽略了许多内容,用户在试用过程中努力发现各种不符合要求部分,提出具体的改进意见。用户和开发人员在这一过程中进一步加深了对系统的认识和彼此之间的理解。这是发现问题,解除误解的重要阶段。

④原型的评价 原型试用后,各类人员分析试用效果,检查规格说明是否与原型相符,以及规格说明能否满足用户的需求,商定全面、详细的修改意见和具体步骤。

⑤原型的修改和完善 根据前一阶段的工作结果,具体动手迅速修改。若发现规格

说明书模糊或未完全反映用户的意见、有疏漏等,应不怕麻烦充分尊重用户意见,迅速修改。甚至果断地忍痛割爱,废弃旧的,重建新原型。这一过程类似于最初的快速分析,因此可以说是原型开发的循环。由于用户不止一次地提出修改意见,所以可行的原型一经建立,只是第3至第5过程的反复进行,直至不断地趋近最终系统的要求。

### 三、使用原型化方法的几个问题

#### 1. 原型法与传统的软件开发法的关系

原型化方法打破了传统方法中开发的一致性 & 严格的线性要求,更好地发挥了用户的积极性,相比之下,前者是后者的扬弃,而不是一味抛弃,所以这种方法也可与生命期开发方法、“FTO”(四图三表一报告)法结合使用。

①将原型化方法运用到传统开发方法中的某个阶段。有时为达到严格的开发软件,在整体上仍采用传统方式,只是在某个阶段使用原型化方法。比如,

在分析阶段 把原型作需求说明的补充或细致的量化,尽可能使需求说明完整、一致和准确,以代替只用反复审核、确认的方法来提高需求说明的质量。

在设计阶段 有时分析与设计是交织在一起、互相联系的。快速分析与初步设计出系统只有一步之遥。在分析的同时也考虑到了设计的要求和目的。用户与开发人员密切合作,不再象以前仅以书面上的说明相联系,纸上谈兵,面面俱到。不敢轻易动手,而是通过共同使用原型 边说边干,结束后得到双方都较满意的需求说明与设计说明。

②将原型化方法运用于传统开发方法的整个阶段,在传统的方法中,用原型化方法代替分析、设计、编码与维护阶段。从图3中可看出,也就是第3步与第5步反复的过

程。但是这需要强有力的软件工具和工作环境的支持。

#### 2. 原型与最终系统的关系

①若只是在分析阶段使用了原型化方法,则原型及其文档资料可看成是传统方法中分析阶段的过渡文档。以此产生出更有效的详细文档资料和系统雏型。最终系统建成,就可把原型废弃。

②若在整个开发过程中的多个阶段或整个阶段使用了原型化方法,基本或完全放弃了传统的开发方法。从开始的基本原型入手,经过与用户的反复修改完善,由达到开始的基本需求,到满足系统的一切要求,那么最终的整体系统就是由原型演化而成的。

#### 3. 如何确定修改的次数

原型化方法实际上是一个不断地对原型进行修改的过程。通常修改一次,用户的满意程度就加深一次,但也可能某次修改降低了用户的满意程度。不可能达到用户百分之百的满意。因此,修改的次数可以这样来确定:

①限制用户的满意程度(80%或90%)及开支、效益控制。

②按第一步修改,详细记下修改的时间、内容、出处和用户的满意程度。

③继续第二步修改,直至达到第一步的要求。

### 四、结束语

原型化方法确是更有效的软件开发方法,但这一方法还只是处于探讨阶段,发展尚未成熟,比如缺乏丰富且强有力的软件工具和开发环境;没有有效的管理研究;安排不当会造成文档不一致、工作混乱等。但从方法本身和设计思想来说,却不失为一种信息系统开发的好方法,为提高软件的开发效

# TYPE命令的不足和改进

南昌 江西拖拉机发动机厂 黄焕如

TYPE命令是DOS的一条内部命令,其功能是显示指定盘或默认盘的ASCII码文件的内容。所谓ASCII码文件即文本文件,例如,各种编译型语言的源程序、解释型语言的源程序、系统结构设置文件、批处理文件以及由字处理建立或编辑的各种文件等等。当使用TYPE命令显示非ASCII码文件时(例如扩展名为EXE、COM、DBF等文件),则可能使屏幕显示发生紊乱,或显示其他一些杂乱无章的符号,甚至会破坏操作系统本身的软件,造成机器“锁死”而不得不重新启动系统,这不能不说是TYPE命令的不足。

TYPE命令不能正常显示非ASCII码文件的主要原因是ASCII码为31以下的某些字符,该命令难以处理。例如,12号ASCII码显示的结果是清屏。为了使TYPE命令既能正常显示ASCII码文件,又能在显示非ASCII文件时能随时中断退出,尤其能特别指出EXE、DBF文件性质,笔者利用C语言(Ver 4.0)编制了一个小程序TYPE.EXE,用来替代原DOS的内部命令TYPE,具体方法如下。

## 一、生成TYPE.EXE文件

利用字处理软件编辑程序TYPE.C,然后编译和链接成TYPE.EXE文件,文件清单为

```
/* TYPE.C */  
#include "stdio.h"  
main (argc,argv)  
int argc;
```

```
char *argv [ ];  
{  
FILE *fp; int i, il, i2;  
char c;  
if (argc<2) {  
printf( "格式: TYPE<文件名>\n");  
exit(1); }  
if((fp = fopen(argv[1],"r")) = 0) /*  
打开文件*/  
{ Printf( "%s ", argv[1]);  
printf("文件不存在! \n");exit(1);}  
il = fgetc(fp); i2 = fgetc(fp);  
if(il = 'M' && i2 = 'Z') {printf( "%S ",  
argv[1]);  
printf( "是EXE文件! \n " ); }  
if(il-3 || i1-0x83){printf("%s",argv[1]);  
printf( "是DBF文件! \n " ); }  
fseek(fp,0L,0);  
while((i = fgetc(fp)) != EOF) {  
c = (char)i;  
if(i>31 || i = 10) printf("%c",c);  
else printf( ". " ); }  
fclose(fp); /* 关闭文件 */  
}
```

该文件实际上可作为DOS的外部命令执行。但是,由于人们已习惯使用TYPE命令显示文件,且内部命令优先于外部命令,故可将COMMAND.COM文件内命令字TYPE改为其他名字。

## 二、修改COMMAND.COM文件

将COMMAND.COM文件内命令字

TYPE为任何其他名字,这样在执行TYPE命令时将不再执行DOS中内部命令,转而执行外部文件TYPE.EXE。以DOS 2.00为例:

```
C ) DEBUG COMMAND.COM
- RCX
  CX 4580
- S100 L4580' TYPE' (查命令字
    TYPE地址)
- XXXX: 3B05 (XXXX为段地址)
- E3B05' TYPP' (非TYPE即可)
- W
```

### 三、建立公用子目录

如果在硬盘上已建有公用子目录,可直接将TYPE.EXE文件拷入该目录,否则新建一子目录,例如

```
C>MD DOS (假定子目录名为DOS)
```

```
C>COPY TYPE.EXE C:\DOS
```

然后在自动批处理文件中末尾加一条PATH C:\DOS即可。

上述步骤完成后,必须重新启动机器,在硬盘任何子目录下均可方便地使用新的TYPE命令。对于非ASCII码文件,31号以下的字符均显示为“.”,当显示字符为结束符“1Ah”时自动退出,否则可按Ctrl+Break键退出,无论它是否被改名,都能特别指出被显示的文件是由LINK链接成的EXE文件还是DBASE数据库的DBF文件。事实上对于任何在文件头有特定标志的文件,该程序很容易扩充,都能指出显示的文件类型。如果您想粗略地看一下某个DBF文件,再也不用进入DBASE解释程序了,用新的TYPE命令即可。

---

(上接第29页)

提供了6位红、绿、蓝彩色合成函数,显然它对遥感显示是十分有利的。因此,我们完全可以建立起一套微机遥感图像专用系统,其图像操作部分是相似的,所以系统中再扩充支持IBM8514卡的软件是很方便的。本设计考虑到目前大多数单位微机配制的是EGA卡和VGA卡,而这两种图形卡可以胜任一般遥感分类图像的显示操作,故而设计了基于这两种图形卡上的通用系统。本系统是微机遥感图像处理专用系统的核心软件,是用户与图像数据的接口,它直接关系到遥感资料的有效处理和应用。同时该系统的开发也为遥感专业人员介绍了一种微机遥感图像专用

系统的开发工具和方法。

### 参 考 文 献

- [1]潘金贵等, Turbo C程序设计技术, 南京大学出版社, 1990年1月。
- [2]赵锐, 遥感数字制图原理与方法, 测绘出版社, 1990年2月。
- [3]刘连硕等, 微机遥感图像应用处理系统, 《环境遥感》, 1989年第4期。
- [4]Williams, Davis, Systems Analysis And Design a structured Approach, Addison—Wesley publishing Company, 1983.



# 智能化信息系统中

## 文本处理的词语表达式系统

东北财经大学人工智能研究所 陈有刚

**摘要** 由单词和词语运算符 $/\&\sim?*$ 等按一定语法规则所联结成的式子称为词语表达式。使用词语表达式和文本句子相匹配的方法查询文本中句子的系统,称为词语表达式系统。本文给出处理文本的词语表达式系统及其算法。

### 一、引言

智能化信息系统中的信息库,通常包括数据库、知识库、算法库、文本库、模型库、图形库和语音库等。文本库中的文本内容可以是语言文字所表达的法条、文件、档案和资料等。文本库的功能是使用计算机按某种要求查询或“理解”文本中的内容。

处理文本的方法很多,例如,受限的自然语言理解,领域专用关键词的查询,条款的查询,文本的形式化描述及其推理系统等。本文给出处理文本的词语表达式系统及其算法。

### 二、词语表达式系统

在词语表达式系统中,文本库是若干个文本的集合,文本是若干个句子的有序集合,并使用词语表达式匹配查询或激励有关的句子。词语表达式是由单词和词语运算符 $/\&\sim?*$ 等按下列语法规则联结而成的式子。

$\langle \text{文本库} \rangle ::= \langle \text{文本} \rangle | \langle \text{文本库} \rangle \langle \text{文本} \rangle$

$\langle \text{文本} \rangle ::= \langle \text{文本主题} \rangle \langle \text{文本内容} \rangle$

$\langle \text{文本内容} \rangle ::= \langle \text{句子} \rangle | \langle \text{文本内容} \rangle \langle \text{句子} \rangle$

$\langle \text{句子} \rangle ::= \langle \text{不含 } \$ \text{ 和 } \# \text{ 的字符串} \rangle \#$

$\langle \text{词语表达式} \rangle ::= \langle \text{词或式} \rangle$

$\langle \text{词或式} \rangle ::= \langle \text{词与式} \rangle | \langle \text{词或式} \rangle / \langle \text{词与式} \rangle$

$\langle \text{词与式} \rangle ::= \langle \text{词元式} \rangle | \langle \text{词与式} \rangle \& \langle \text{词元式} \rangle$

$\langle \text{词元式} \rangle ::= \langle \text{词组} \rangle | \sim \langle \text{词组} \rangle | (\langle \text{词或式} \rangle)$

$\langle \text{词组} \rangle ::= \langle \text{单词} \rangle | \langle \text{词组} \rangle * \langle \text{单词} \rangle | \langle \text{词组} \rangle \langle ? \text{串} \rangle \langle \text{单词} \rangle$

$\langle \text{单词} \rangle ::= \langle \text{不含 } \sim / \& ? * \$ \# ( ) \text{ 的字符串} \rangle$

$\langle ? \text{串} \rangle ::= ? | \langle ? \text{串} \rangle ?$

对于一个句子G和两个词组a、b,词语表达式的语义是:

a: G含有a

$\sim a$ : G不含有a

$a/b$ : G含有a或G含有b

$a\&b$ : G含有a而且G含有b

$a*b$ : G含有 $a\cdots b$ ,其中a和b间可以含有任意多个字符,即\*可以占有任意多个(包括0个)字符位置。

$a? \cdots ? b$ : G含有 $a\cdots b$ ,其中a和b间的字符个数是?的个数,即每个?占有1个字符位置。

例如  $ab/cd\&\sim ef/g*b$

这表示若G含有ab,或G含有cd但不含有ef,或G含有 $g\cdots h$ ,则为真,否则为假。

这就是说,一个词语表达式对于某个句子 $G_1$ 可能是真的,对于某个句子 $G_2$ 可能是假的。在查询时,只激励使这个词语表达式为真的那些句子,或输出显示这些句子,或调用相应的功能子程序,或理解并处理这些句子,使这个系统具有一定的智能性。

在使用词语表达式处理文本时，首先由用户任意输入一个词语表达式。然后由计算机将这个词语表达式转换为关于/ & 和 ~ 的后缀式，相当于逆波兰表达式。

词语表达式后缀式的语法是

〈词语表达式后缀式〉 ::= 〈词组〉 | 〈词语表达式后缀式〉 〈词语表达式后缀式〉 / | 〈词语表达式后缀式〉 〈词语表达式后缀式〉 & | 〈词语表达式后缀式〉 ~

例如 (a/bc) & de / ~ fg & h 的后缀式是 a, bc, /, de, &, fg, ~, h, &, /

将词语表达式转换为其后缀式的递归过程的pascal算法如下

```

procedure H (var X:C);
var y,z:C;
begin E(y);
  while NS = "/" do
  begin
    scan, E(z);
    y := y + z + "/";
  end;
  x := y;
end;
procedure E (var x:C);
var y,z:C;
begin B(y);
  while NS = "&" do
  begin
    scan, B(z);
    y := y + z + "&";
  end;
  x := y;
end;
procedure B (var x:C);
begin
  if NS = "~" then
  begin scan;
    X := NS + "~";
  end else
  if NS = "(" then
  begin scan;
    H(X);
    if NS ≠ ")" then error
  end else

```

X := NS;

scan

end;

在上述算法中，C是字符串类型，过程scan每一次或取一个词语运算符/ & ~ ( )，或取一个词组，并送入NS中。

最后使用一个布尔栈ST(j)计算其后缀式之值。其算法是由词语表达的后缀式，每次扫描一个词语运算符或词组。若是~，则将栈顶反演；若是/或&，则将次栈顶和栈顶进行相应的运算∨或∧，并将结果存入次栈顶后栈退1；若是词组，则将词组和句子相匹配，并将匹配结果true或false加进栈。

词组和句子间的匹配过程是这个算法的核心。匹配的目的在于判断这个句子G是否含有这个词组。匹配过程是一个搜索过程。

例如，当词组a \* bc ? d和句子

a...bcde...bcd...d...

匹配时，第1次试探第1个b...，不成功；

第2次试探第2个b及其后继ccd，成功。

再如，当词组a ? ? bc \* d和句子

aaaaabced...

匹配时，第1次试探第1个a...，不成功；

第2次试探第2个a...，不成功；第3次试探第3个a...，成功。

为了实现匹配的搜索过程，使用一个栈的数据结构表示词组。例如，对于词组

a \* bc ? ? ? de

其栈结构如下图所示。

栈顶针 T	算符 S	单词 D	长度 C	起点 Q	? 个数 W
1	*	a	1		0
2	*	bc	2		0
3	?	de	2		3
:	:	:	:	:	:

相应的匹配算法如下

M := length(G); y := G; T := 1;

```

13: if S(T) = “#” then 为真并结束;
    if S(T) = “*” then
        begin
            DC := D(T); N := Q(T-1) + C(T-1);
            Y := copy(Y, N + W(T), M);
            P := pos(DC, Y);
            if P = 0 then 为假并结束;
            Q(T) := P; T := T + 1; goto 13;
        end;
    if S(T) = “?” then
        begin
            N := Q(T-1) + C(T-1);
            if copy(Y, N + W(T), C(T)) = D(T)
                then begin Q(T) := N + W(T);
                           T := T + 1; goto 13
                        end;
            else begin while S(T) = “?” do
                           T := T - 1;
                           W(T) := W(T) + 1;
                           goto 13
                        end;
        end;

```

### 三、处理词语表达式的优化算法

在处理词语表达式

$a_1 \& a_2 \& \dots \& a_n$

时, 其后缀式是

$a_1, a_2, \&, a_3, \&, \dots, a_n, \&$

只有当 $a_1, a_2, \dots, a_n$ 的相应值计算完后, 才能得到其结果。事实上, 只要有一个 $a_i$ 的相应值是假, 其结果便为假, 为了提高计算速度, 而不必计算其它 $a_i$ 的相应值, 特别是 $> i$ 的 $a_i$ 。对于 $a_1 / a_2 / \dots / a_m$ , 也有类似的情况。

为优化处理词语表达式的算法, 可以采用文献〔1〕中所给出的计算布尔表达式的方法。

例如  $a \& b * c ? ? de / f \& \sim g$

所相应的三元组是

	S(N)	V(N,1)	V(N,2)
①	#	1	
②	a	3	4
③	b * c ? ? de	7	4
④	f	5	6
⑤	g	6	7
⑥	#	0	
⑦			

其中 {# 1} {# 0} 分别表示将结果值置于1(true)或0(false); {a 3 4} 表示若G含有a转第③号三元组, 否则转第④号三元组, 即后两个元素分别表示真转和假转的序号。

将词语表达式转换为三元组的算法如下。

```

procedure H(var TC, FC: integer);
var CC: integer; label lp;
begin
    TC := 0; FC := 0;
    lp: E(CC, FC);
    MG(TC, CC);
    if NS = “/” then
        begin
            FL(FC);
            FC := 0;
            scan; goto lp
        end
    end;
procedure E(var TC, FC: integer);
var CC: integer; label lp;
begin
    TC := 0; FC := 0;
    lp: B(TC, CC);
    MG(FC, CC);
    if NS = “&” then
        begin
            FL(TC);
            TC := 0;

```

```

                scan, goto lp
            end
        end,
    procedure B(var TC, FC : integer);
        var K : integer;
        begin
            if NS = "(" then
                begin scan,
                    H(TC, FC);
                    if NS = ")" then scan
                        else error,
                            exit
                    end,
                if NS = "~" then
                    begin scan,
                        B(TC, FC);
                        K := TC; TC := FC; FC := K;

```

```

                exit
            end,
            TC := N+1; FC := -TC;
            N := N+1; S(N) := NS;
            V(N, 1) := 0; V(N, 2) := 0
        end;

```

关于反填过程FL, 合并拉链过程MG, scan等, 此不赘述。

将词语表达式转换为三元组后。再解释执行三元组。易见, 解释执行三元组的过程, 实质上是个有序的产生式系统。

词语表达式系统在研制智能化信息系统中是有意义的。

本文所给出的全部算法, 是杨晓帆调试的。

### 参 考 资 料

- [1] D.Gries, 〈编译程序构造〉。
- [2] 陈有刚, 智能化信息系统中的算法库设计, 《计算机时代》, 1990年第2期。
- [3] 陈有刚, 智能化信息系统中的离散结构, 《计算机时代》, 1991年第1期。
- [4] 陈有刚, 智能化信息系统中的广义推理系统, 《计算机研究与发展》, 1991年第1期。

(上接第36页)

率开辟了新途径。

### 参 考 文 献

- [1] 潘锦平, 软件开发技术, 第一版. 上海科普文献出版社, 1985。
- [2] 冯玉琳 赵保华, 软件工程, 第一版, 中国科技大学出版社, 1988。
- [3] 张维业 杨芙清 唐世谓, 软件生存期与速成原型方法, 《计算机科学》, 1989年第2期。
- [4] Tamara Tagloy and Thomas A. standish, "Initial thoughts on rapid Prototyping techniques", ACM, SIG-SOFT software engineering notes, DEC, 1982。

# 硬盘的数据结构及软故障排除

南京炮兵学院 宋晓田 李广宇

**摘要** 本文在介绍硬盘的内部结构、数据组织形式的基础上,分析几种常见硬盘故障产生的原因并提出排除方法。

## 一、硬盘的物理性质

### 1. 结构

硬盘由一个或多个盘片组成,它们安装在由一个小马达驱动的轴杆上,小马达带动盘片高速旋转,每个盘片的双面都可存取数据,每面有一个读写磁头。

磁盘的每面由若干磁道组成,由外向内依次编号为0、1、2、...各面上具有相同道号的磁道组成一个柱面,每道又分成若干个扇区。例如,典型IBM—PC/XT10MB固定硬盘有两个盘片,每面305条磁道,每道有17个扇区,每个扇区存放512字节的数据。

### 2. 物理格式化(低级格式化)

低级格式化可由LOWFORM完成,它的主要工作是将每道划分为扇区(每扇区由地址场和数据场两部分组成),写好识别每扇区的地址场信息,并用某常数写入数据场。

### 3. 硬盘的读/写

经过低级格式化后,磁盘上每个扇区由三个参数(柱面号C,磁头号H,扇区号R(1~17))确定,可利用硬盘ROM BIOS提供的读写操作访问硬盘。例如读2号磁头,3道的9号扇区,可使用如下的汇编程序实现。

```
MOV AH,00H
MOV DL,80H
INT 13H      ;1硬盘复位
MOV AX,0201H;AH=02读操作,
              AL=01读一个扇区
```

```
MOV CX,0309H ;CH=03 3道,
              CL=09 9扇区
MOV DX,0280H ;DH=02 2号磁头
MOV BX,E000H
MOV ES,BX
MOV BX,0000H ;ES:BX数据缓冲区地址
INT 13H
```

## 二、硬盘数据空间的逻辑组织

### 1. 硬盘分区

进行低级格式化后,可以使用DOS盘上的FDISK程序对硬盘进行分区,例如,可以分成4个分区,这时硬盘数据空间划分为:

主引导扇区	分区1	分区2	分区3	分区4
-------	-----	-----	-----	-----

主引导扇区占0头0道1扇区,由分区代码程序和分区信息表两部分组成,分区代码程序是LOWFORM程序在做完低级格式化后写入的,在开机时完成引导任务,分区信息表是在执行FDISK程序时填写的,包含有关每个分区情况的信息,如首地址,大小,活动分区标志等。

### 2. MS—DOS分区

各个分区用于装入不同的操作系统或数据,可以将MS—DOS装在某个分区,例如2分区中。假如2分区空间已划定由3号到203号柱面组成,在物理上,硬盘空间是不连续的,每个扇区需由三个参数决定,扇区号不能唯一决定对应的扇区,为了处理方便,DOS的文件管理模块,在逻辑上将DOS分区空间看成

是由连续扇区组成的, 给予每个扇区一个唯一的逻辑扇区号L, 这样就在分区数据空间上建立了一个由物理地址(H,C,R)三参数组到逻辑地址L的一对一的映射, 对应可以有不同方式, 下表所示即为其一。

	0	1	2	3
3	0,1,... ,16	17,18,... ,33	34,35,... ,50	51,52,... ,67
4	68,69,... ,84	85,86,... ,101	102,... ,118	119,... ,135
⋮	⋮	⋮	⋮	⋮
203	13600,...	...	...	...,13667

一旦建立了由逻辑号到扇区的一对一的对应关系后, 只要给出逻辑号, DOS就可以读写硬盘上相应的扇区。例如, 用DEBUG程序读逻辑号为18的扇区的操作如下:

A> DEBUG

-L 100 2 18 1

### 3. 逻辑格式化 (FORMAT C:)

MS-DOS分区必须经过逻辑格式化后, 才能由MS-DOS使用, 这个工作由DOS盘上的FORMAT程序完成, 执行命令 FOMAT C:, 格式化后, DOS分区的数据格式为:

引导 扇区 BOOT AREA	文件分 配表 1 FAT #1	文件分 配表 2 FAT #2	文件根 目录扇区 FILE DIRECTORV	文件数 据区 USEDATA AREA
--------------------------	-----------------------	-----------------------	----------------------------------	------------------------------

该格式与软盘上的数据格式一样, FORMAT程序对C: 盘格式化主要做了以下几方面的工作。

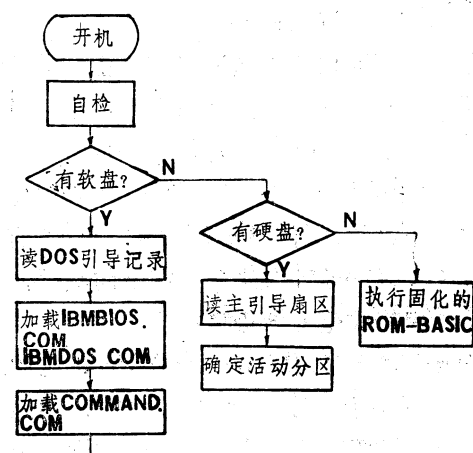
- ①校验DOS分区各扇区的读/写好坏情况, 记录在FAT表中。
- ②DOS引导记录写入BOOT AREA。
- ③写入FAT表。

如果用FORMAT C: /S形式格式化C: 盘, 除完成上面工作外, 还将MS-DOS的两个隐含文件(IBMBIOS.COM, IBMDOS.COM)和COMMAND.COM文件写入文件数据区。

引导扇区占一个扇区, 内含引导MS-DOS的代码程序, 文件分配表FAT记录磁盘文件的空间占用状态。文件读/写都要使用FAT#1表, FAT#2是FAT#1的备份。

## 三、DOS启动过程

计算机在自检以后, 首先试图从A盘启动, 若不成功, 则转从硬盘C: 启动。使用硬盘启动时, 首先将主引导扇区代码程序装入内存以确定那个分区是DOS分区; 然后将DOS分区的引导扇区代码程序 (BOOT AREA) 读入内存, 并执行, 启动过程见下图所示。



## 四、故障分析及排除

硬盘故障的原因来自硬件和软件两个方面, 本文只讨论由于记录在硬盘上的引导程序等重要信息出错而造成的软故障, 根据能否从硬盘直接启动, 可分以下两种情况。

### 1. 系统不能从硬盘启动

这时又可按系统从软盘启动后能否进入硬盘分两种情况。

①从软盘启动后能进入硬盘, 而从硬盘启动时屏幕无显示或显示

Non-system disk or disk error

Replace and strike any key when ready

这提示DOS分区中两个隐含文件有错或没有两个隐含文件, 可以先用A盘启动, 然后打入命令

A>SYS C:

重新拷入两个隐含文件, 若不能排除, 再打入命令

A>FORMAT C: /S

重新格式化硬盘, 即可排除。

如屏幕显示

Bad or missing command interpreter

这提示COMMAND.COM文件可能有错, 可以用A盘启动, 然后打入命令

A>COPY COMMAND.COM C:

即可排除。

## ②从软盘启动后不能进入硬盘

这可能是主引导扇区信息有错, 对于AT机, 还可能是硬盘设置有错,

A. 主引导扇区信息有错

从软盘启动后, 要进入硬盘C: 时, 出现如下信息

Invalid Drive Specification

这是由于主引导扇区有错, 用FORMAT程序无法排除这种故障, 须使用LOWFORM程序对硬盘进行低级格式化, 然后用FDISK程序分区, 再用FORMAT程序逻辑格式化C: 盘。

## B. AT机硬盘设置错

AT机硬盘类型是登记在CMOS配置存储器(CMOS CONFIGURATION MEMORY)中的, CMOS使用系统带的电池工作, 由于某种原因造成CMOS信息丢失, 也将导致硬盘无法使用。

可以运行AT机高级诊断程序中的SETUP程序, 重新设置硬盘类型, 即可排除,

## 2. 能从硬盘启动, 但读写出错

这种故障可能有以下三种原因。

### ①文件数据区信息破坏

读盘时会发出反复定位的咔咔异常声,

显示出如下信息

Seek error reading drive c:

Abort, Retry, Ignore?

这是由于某扇区信息破坏了。

### ②文件分配表信息破坏

读写时屏幕上出现

File allocation table error or bad

说明文件分配表信息破坏了, 这将导致硬件上的文件无法读写。

### ③文件目录区信息破坏

如不能读写文件且用命令c: dir列文件目录有错, 则应怀疑是文件目录信息破坏。

上述故障一般均可用打入命令

A>FORMAT C: /S重新格式化硬盘的方法排除。

通过以上处理不能排除的故障, 一般是由于硬盘控制卡或磁盘等硬件有物理损坏, 需送专门的维修部修理或更换部件。

## 五、建 议

为了减少操作失误造成的磁盘故障, 应做到

### 1. 养成复制数据和程序的良好习惯。

由于硬盘是共享设备, 及时将硬盘中的数据复制到软盘上, 防止别人破坏你的信息, 以及当硬盘有故障或误操作时造成大的损失。

2. 应避免频繁开关电源, 一般关机后应至少等待1分钟以上再开机, 切忌在硬盘工作时开关电源。

3. 在关机和机器搬运时都应运行磁头锁定程序(PARK程序或SHIPDISK程序)。

4. 机器应处于良好的使用环境。

### 参考资料

[1] MS-DOS设备驱动程序剖析与实现,

夏东涛译, 北京科海培训中心。

[2] IBM PC/XT微型计算机系统原理及维修, 苏州电子计算机厂, 1987.6。

[3] 微型计算机系统原理分析及维修, 傅在益等编, 科学出版社, 1988.6。



# RA81磁盘子系统的维护与检修

浙江省公安厅信息处 徐 力

**摘要** 本文介绍RA81磁盘驱动器的维护与检修方法,先概述RA81磁盘驱动器的结构与原理,再分析RA81磁盘本身所具有的几种诊断手段——联机软件诊断,磁盘前面板指示灯,用诊断终端脱机诊断等。根据笔者的实践,结合具体的故障事例,详细地介绍如何利用诊断手段、快速、准确地分析和排除RA81磁盘驱动器故障的方法。

## 一、引言

DEC公司的VAX系列计算机在我国已广泛使用并受到用户的好评。其外存设备之一的RA81磁盘驱动器与主机系统及其它外部设备相比,质量上略有逊色,故障率较其它设备高。因此,尽量避免发生故障和快速、准确地诊断故障并尽快排除,是保证系统正常运行的重要条件。

## 二、RA81磁盘驱动器的

### 结构与原理

RA81磁盘驱动器是一种采用温切斯特(Winchester)技术制作,随机存取,磁头可移动的高性能磁盘驱动器。它使用全密封的磁头/磁盘组合件(HDA),可靠性好,记录密度高,是RA81的关键部件。RA81的数据存贮容量为456兆字节(以16位字格式)并通过标准磁盘互联SDI总线联到磁盘控制器上,任何符合SDI总线标准的控制器均可使用RA81磁盘驱动器。

密封的HDA包含记录介质(四片磁盘),14个读/写磁头,回旋定位器和前置放大电路。7个磁盘面用来存贮数据,第8个盘面用作伺服面,它包含编码的伺服信息。每个数据盘面有两个读/写磁头,而伺服面只有

一个只读磁头来读出伺服信息,专门用来定位。使用伺服板的从属微处理机锁定伺服系统来控制HDA中的回旋定位器,伺服控制回旋定位器按着记录表面上的圆弧形状来移动磁头,这样把磁头定位在数据地址上,RA81磁头定位采用了专用伺服面和嵌入式伺服两种方式,专用伺服面作为磁盘的粗定位而用嵌入式伺服突发信息作为磁盘的精定位,从而使得磁头的定位显得更加快速、准确。

RA81磁盘所有逻辑都包含在驱动器内部,有三块主要的电路板。

伺服板(Servo Module):用来控制磁头定位;

微处理器板(Microprocessor Module):进行磁盘读写及CPU控制;

个性板(Personality Module):微处理器板和磁盘控制器的接口

RA81在ROM中存放了自诊断程序,用户可以通过它对RA81本身各部件进行检测,如有故障,则把故障限定在一个或几个现场可替换部件(FRU)上,从而保证了RA81维修更加快速、简捷。这些FRU包括:微处理器板、伺服板、个性板、读/写板、控制台前面板、20针数据电缆(个性板上)、40针数据电缆(个性板上)、50针读/写电缆(读/写板上)、20针数据电缆(读/写板上),控制面板电缆、电源和HDA组件。

### 三、RA81诊断

RA81具有完善的诊断功能,其诊断有三种方法:

①系统提供的联机软件诊断;

②在加电序列期间,利用控制台前面板指示灯的显示自动运行的一组诊断以检查启动操作;

③在脱机方式下,通过诊断终端运行的诊断。

#### 1. 联机的软件诊断

这种诊断,通常是在RA81能正常启动时,对RA81作定期维护或对磁盘控制器作诊断时才使用。

首先,必须把诊断程序从存贮设备(磁盘或磁带)中引导出来;

第二,用诊断命令ATTACH和SELECT将要诊断的设备联结好;

第三,运行诊断程序,并正确回答诊断程序所提出的问题;

最后,根据诊断程序所显示(或打印)出来的结果进行分析。

此外,在VMS操作系统下,利用系统对磁盘平时出错信息的记录进行分析,这样可随时监视磁盘的运行状态。具体的命令如下:

```
$ SET DEF SYS $ERRORLOG
$ ANALYZE/ERROR/INCLUDE = DISK
  ERROR.LOG
```

#### 2. 利用前面板的指示灯进行检查

RA81磁盘驱动器前面板上的每一个开关都有一个显示驱动器状态的指示灯,它们依次是:

运行/停止	( RUN/STOP )
故障	( FAULT )
设备号/准备好	( Unit/READY )
写保护	( WRITE PROT )
A 口	( A )
B 口	( B )

当驱动器检测到一个错误之后,“故障”灯亮,驱动器一直停留在当前的操作下,直到错误被纠正。某些故障会自动纠正,

“故障”灯自动熄灭;某些故障不会自动纠正,“故障”灯仍然亮着,则必须按“故障”开关。第一次按“故障”开关,驱动器进入脱机状态(相对于控制器),这时前面板指示灯由驱动器控制显示出一种闪烁的错误码(在显示错误码期间,驱动器一直维持脱机状态),记录错误码;第二次按“故障”开关以后,磁盘驱动器试图清除故障,贮存错误码并进入可用的工作状态。如果该故障为偶然性故障,则前面板指示灯恢复正常,并继续工作,否则,重复上述步骤,继续检查驱动器有无其它故障。

由于RA81驱动器本身具有加电测试序列和空循环的测试序列,因而在加电时驱动器本身就对它作了各种诊断测试。当“运行/停止”、“A口”和“B口”三个开关均处断开时,给驱动器加电,则运行加电测试序列,它包括各种硬码测试及静态主逻辑和伺服测试。在执行硬码测试时,前面板指示灯全亮,约3—4秒以后,如果指示灯全灭,表示加电测试序列完成;如果还有灯继续亮着,则表明加电测试序列失败。在顺利完成加电测试序列后约30秒开始执行空循环测试,空循环测试要运行30秒并通过前面板指示灯以及微处理器板的指示灯来报告故障状态。如果一切正常,微处理器板的指示灯为十六进制的“E7”。例如:在加电测试序列和空循环测试时,前面板的“故障”灯和“B口”灯闪烁,查维修手册为“加速错”,同时微处理器板的故障指示灯的错误码为“04”,即在加速周期间40秒内主轴速度未达3420转/分,替换马达,故障即除。

#### 3. 利用诊断终端进行诊断

利用现场服务诊断终端是RA81区别于其它磁盘驱动器一大优点,它采用标准的

ASCII键盘和RS232接口,这样可以使使用任何EIA兼容终端设备和驱动器通信(300波特是驱动器逻辑的缺省参数)。

### ①诊断终端的安装

位于微处理器板中间的联接器通过接口电缆联到联接器的底座上,该座的联接器即标准的RS232接口,用来联接诊断终端。在诊断时要求“**A口**”、“**B口**”开关处于断开位置,驱动器才能响应键盘,联好电缆后,立即按终端键盘的 $\wedge$ C键,终端提示符出现之后,用户可以打入十个诊断命令中的一个去进行诊断测试。

### ②诊断命令的运行

下表列出了诊断命令和它们的缩写:

命 令 名 称	缩 写
运行诊断	RUN DIAG
运行寻址	RUN SEEK
运行再定标	RUN RECA
运行读/写	RUN READ
运行伺服	RUN SERV
运行接口	RUN INTE
运行交替	RUN ALTE
运行随机	RUN RAND
运行递增	RUN INCR
运行诊断测试 = $\times \times$	RUN DIAG TEST = $\times \times$

在上面10个诊断命令中,运行诊断命令最常用,因此一般我们总首先运行该命令。该命令可以在两种不同的情况下使用,即驱动器已经加电,但未按“运行/停止”开关,驱动器处于停止状态;驱动器加电,并按下“运行/停止”开关,驱动器处于旋转状态。

在此基础上,可通过其它命令进行诊断,以进一步确定故障的范围,并根据提示更换FRU部件。

当把故障范围确定在几个FRU部件后,我们还可从电压指示灯和几组电缆联接线进

一步确诊FRU的故障范围。

### A.指示灯

在电源上的4只指示灯(+10V, -10V, +24V, -24V)中的任一只灯不亮,则表明电源板有故障;伺服板上的两只指示灯(+12V, -12V)不亮,则表明伺服板有故障。

### B.联接线电压

J205的第1,2,3脚的电压为+5V;

J306的第15,16脚的电压为-5.2V;

J403的第11脚的电压-24V;

J403的第13脚的电压+24V;

J601的第7,8脚的电压为-4V;

J601的第9,10脚的电压为+6V;

J602的第2脚的电压-8.2V;

如果+5V, -5.2V, -24V, +24V中的任一电压不正常,则表明电源板有故障;如果-4V或+6V电压不正常,则检查读/写板上的电压调节器;如果-8.2V电压不正常,则表明伺服板有故障。

## 四、RA81维修实例

### 故障现象

开机时,前面板指示灯的“运行/停止”和“故障”灯亮,按“故障”开关,“故障”、“写保护”和“B口”灯闪烁。

### 故障排除

从前面板的错误现象看,是在伺服精定位时出错。打开机柜,从微处理器板的错误指示灯看,其错误码为“F7”,即在“从”再定标超时,FRU可能的故障有伺服板,读/写板和HDA。

为了进一步分析故障,用诊断终端脱机对其进行诊断。

1.马达不旋转时,对磁盘驱动器进行检查(“运行/停止”开关置于断开位置)

RA81>RUN DIAG TEST=17

%RA81-COMLETE-TEST: DIAG

SUBTEST: 02

%RA81-COMplete-TEST; DIAG  
SUBTEST; 01

%RA81-COMplete-TEST; DIAG  
SUBTEST; 07

·  
·  
·

%RA81 COMplete-TEST; DIAG  
SUBTEST; 20

表明运行诊断成功。

2. 马达旋转时, 对磁盘驱动器进行检查  
(“运行/停止”开关置于联接位置)

RA81>RUN DIAG TEST = 16

%RA81-COMplete-TEST; DIAG  
SUBTEST; 02

%RA81-COMplete-TEST; DIAG  
SUBTEST; 01

%RA81-COMplete-TEST; DIAG  
SUBTEST; 07

%RA81-COMplete-TEST; DIAG  
SUBTEST; 08

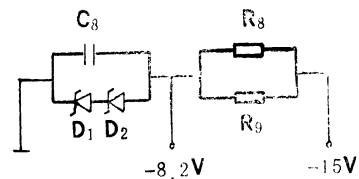
%RA81-COMplete-TEST; DIAG  
SUBTEST; 0B

%RA81-COMplete-TEST; DIAG  
SUBTEST; 03

%RA81-COMplete-TEST; DIAG  
SUBTEST; 19

%RA81-TEST; DIAG SUBTEST; 1B  
ERROR; F7 UNIT; 001

%RA81-FRU-SERVO, RD/WT, HDA  
这表明在诊断测试到子诊断测试1B时, 发现错误。测试“1B”是检查伺服板上的PLO电路, 从上述诊断基本可确诊为伺服板有故障, 检查J602的2脚电压, 结果为0V, 则确认伺服板有故障。查阅有关的电路(如图所示), 分析可能是稳压管D1和D2击穿或电容C8击穿



检查发现电容C8已击穿, 更换电容开机自检, 一切正常。