

计算机时代

COMPUTER ERA



1991



征 稿 启 事

本刊是通过邮政系统公开发行的技术类季刊，它以传播计算机开发应用新信息，交流计算机应用新经验，推广计算机应用研究新成果，介绍国内外计算机动态，培养计算机开发应用人才，促进计算机科学技术尽快地转化为生产力，加速我国“四化”建设为宗旨。

主要栏目 开发与应用、软件技术、专题报告、技术探讨、国内外动态、知识园、修经验、综合信息、少量译文以及计算机开发应用成果、产品（包括智能仪器）、新等。欢迎全国广大从事计算机教学、研究、设计、开发、应用人员踊跃投稿。

来稿注意事项

1. 来稿须论点明确，内容新颖，文句精练，数据可靠。每篇稿一般不超过6000字（含图表）。凡多于3000字的论文，起始应有150字左右的摘要。

2. 来稿一律用16开方格稿纸（20×20或20×15）楷体书写，要求字迹工整，标点准确，每体一字（标点也占一体）。油印稿或非稿纸打印稿将退回作者誊抄。

3. 文题须确切简明。稿中外文字母、符号须分清大、小写，正、斜体，对易混淆的要用铅笔在旁注明文种、大小写或数字。上、下角码和符号的位置高低应明显区别。

4. 所用的计算单位一律采用《中华人民共和国法定计量单位》。

5. 图表（如系统结构、软件流程、模块结构、系统配置、电路、波形等）要少而精，绘制比例适当，布置紧凑。图中文字和数字须书写清楚，并注明图（表）序号，图（表）题名。除了必不可少的补充正文用的局部源程序外，一般不必附上源程序清单。

6. 参考文献须是公开出版的书、期刊、报纸并且要精选，按文中出现先后依次编号，序号用小方括号括起。在正文引用处，带小方括号的序号标在右上角。文献格式如下：

书 序号，著（编）者，书名，版次，出版地，出版社、出版年，首页码或起讫页码。

刊（报） 序号，作者，题名，刊（报）名，出版年，卷号（期号），首页码或起讫页码。

7. 文后务必注明作者详细通讯地址、邮政编码、联系电话。

8. 来稿一经刊出，即致稿酬，并赠送当期期刊两本。未录用的稿件会适时通知作者，原稿不退，请自留底稿。

计 算 机 时 代

（季 刊）

1991年 第4期 （总第32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 浙江省报刊发行局
订 购 处 全国各地邮电局
国内统一刊号 CN 33—1094/TP
邮 发 代 号 32—81
出 刊 日 期 1991年11月20日
每 期 定 价 1.00元

计算机时代目次

1991年第4期

(总第32期)

§§§§§ 开发与应用 §§§§§

PLC在烟草打叶线自动控制系统中的应用 (1)

汪洪

环境地质质量多级模糊综合评价预测系统 (3)

陈振武

稳定可靠的自动复位抗干扰技术 (8)

王攀峰

微机在数字电路调试中的应用 (10)

张胜

打印机状态设置在报表程序设计中的应用 (12)

王敬农 应好仁

§§§§§ 软件技术 §§§§§

基于结构化分析的数据字典工具 (15)

钱涛 金连甫 胡上序

交互式图形程序库的设计方法 (20)

景红

高级语言与汇编语言混合编程技术 (24)

庄真 胡家富

XENIX环境下FOXBASE+文件与DOS环境下

dBASEⅢ文件的兼容 (29)

章寿发

在INFORMIX·4GL关系数据库语言环境下

实现表格的灵活打印 (31)

王余星

dBASEⅢ与C语言间数据传递新法 (33)

符云清

§§§§§ 专题报告 §§§§§

改善企业MIS的可维护性策略 (34)

于功第 路枝

信息处理的映射和链接技术 (38)

杨宪泽

§§§§§ 维修经验 §§§§§

AR—3240打印机常见故障及维修 (43)

吴可 卢岩文

长城0520—CH微机015板维修二例 (45)

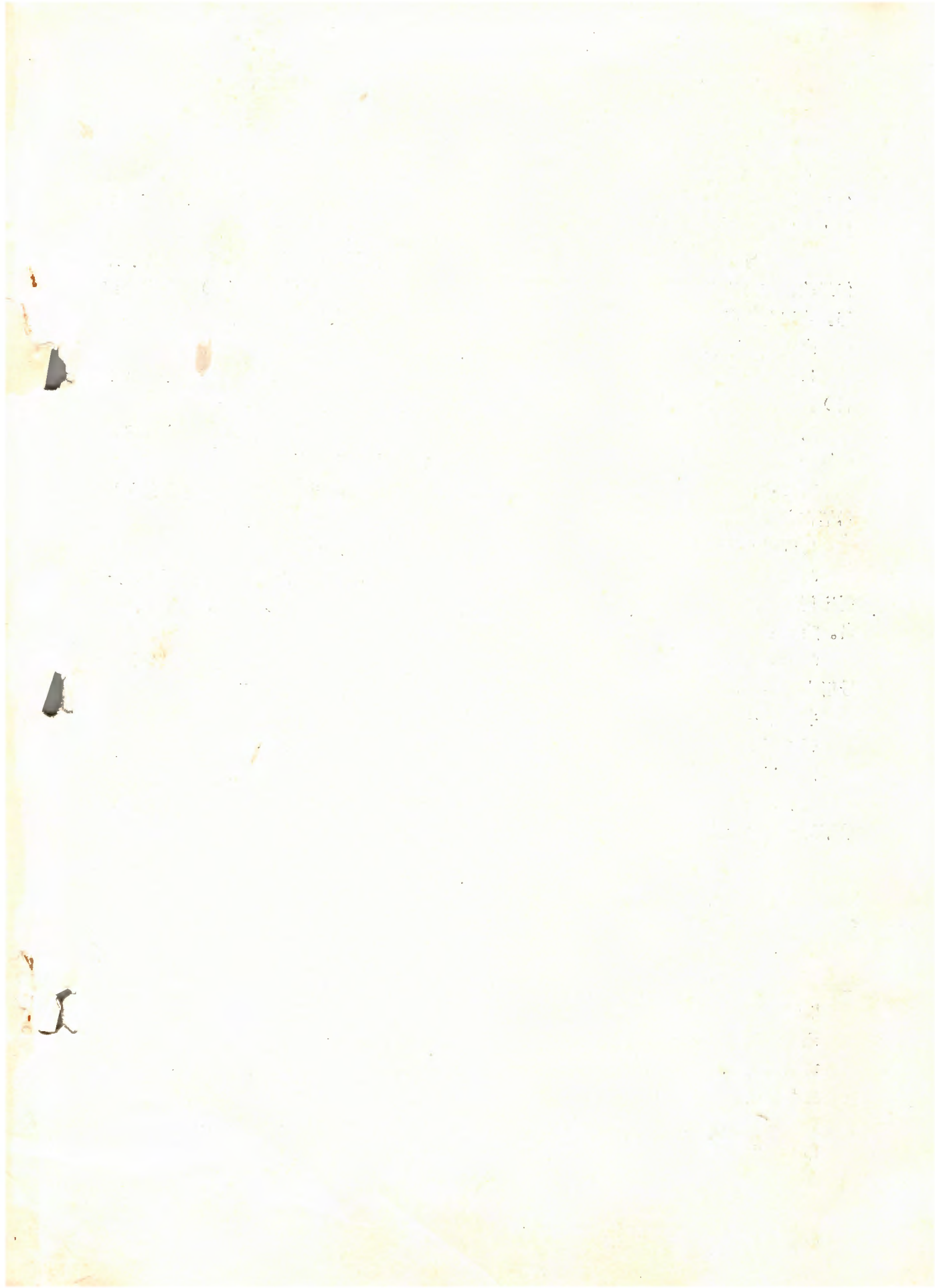
胡红

§§§§§ 知识园地 §§§§§

条形码识别技术及其应用 (46)

史红霞

征稿启事 (封底)



PLC在烟草打叶线 自动控制系统中的应用

航空航天部第15研究所 汪 洪

一、引言

打叶线是烟草生产加工中的一条生产线。它在烟草生产过程中占据较重要的地

位。我所为海林卷烟厂研制的3000kg/h打叶线按工艺可分为四打十分。全线设备组成如图1。在生产过程中,烟叶首先经过双路切尖解把机,完成叶梗分离,叶尖部分通过翻箱和

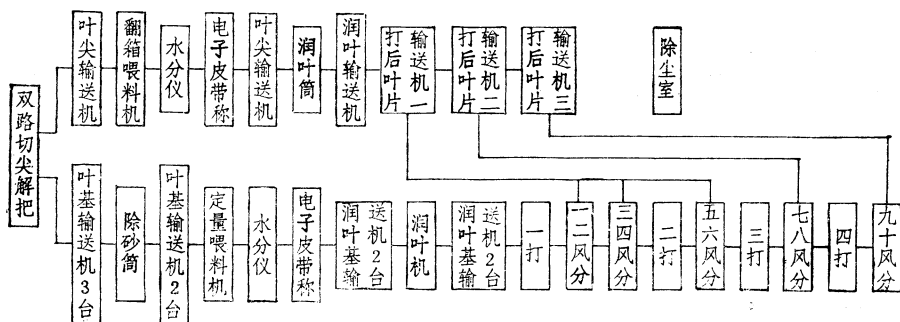


图1 3000kg/h打叶线设备框图

喂料机,进行水分测定以及流量控制。然后送往润叶筒内加高压蒸汽和一定水量,最后通过输送机送至叶处理线。而叶梗部分经过除砂筒除砂,通过定量喂料机,进行水份测定和流量控制。在润叶机内润湿,加高压蒸汽,经过四打十分,完成打叶,风分任务,最后通过输送机送至梗处理线,除尘室则要完成空气净化。全线大小设备共计30余台。在过去的打叶线上,其控制系统由传统的继电器组成,不仅控制设备庞大,故障率高,调试及维修不方便,而且控制系统自动化程度差,生产效率低,系统抗干扰能力也差。80年代,国外推出可编程控制器即PLC应用于工业控制系统中,逐渐取代以继电器为中心的工业控制。PLC的特点是适应恶劣工业环境,具有较高的抗干扰能力。在工业控制中

容易构成控制系统模块化,使控制设备体积大大减小。采用软件技术可方便地实现现场中的工艺调试。且编程简单,维修方便。因此,在海林烟厂的打叶线上,我们采用PLC作为控制主机,采用美国Westhouse公司产品PLC—700,将其应用于打叶线自动控制系统。

二、系统主要功能及构成

1.主要功能

- ①控制全线105台电机,含单向电机、双向电机、15kw以上大电机(需Y/△启动)。
- ②接收全线各种传感器信号。
- ③控制全线27个电磁阀。
- ④对启动、停车、事故及工作情况下的

安全连锁控制。

⑤控制模拟屏灯显, 监视全线工作状态。

⑥完成同上位机间的通信。

⑦各种声光报警, 故障指示。

2. 硬件设计

根据控制系统的要求, 输入到PLC的信号达406点, PLC输出的信号达339点, 4路模拟量输入。系统又分为24VAC输入, 220VAC输出和5VAC输出。在进行I/O地址分配时, 结合本控制系统特点。即控制对象多, 工艺过程复杂。将全线分成叶梗段, 叶尖段, 打叶段, 除尘室段。在具体分配I/O地址时, 按输入及输出信号类别、设备自身、设备与设备、段自身、段与段, 采用分块集中, 以便管理如图2。在接触器输入口中, 按段、设备排列, 即入口地依次为叶尖段、叶梗段、打叶段、除尘室段。其中输入/输出口地址依次可推出。

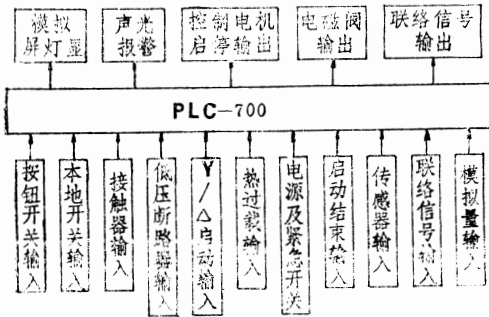


图2 I/O口地址分配框图

确定I/O口后, 配置扩展电源便可组成硬件系统。

3. 软件设计

系统控制程序按各段分成相对独立程序块。如图3所示。开机后, 系统进行初始化。通过人机对话, 主机开始进行自动循环前的准备工作。首先判断是自动还是手动,

若手动, PLC程序不运行; 当自动开关合上

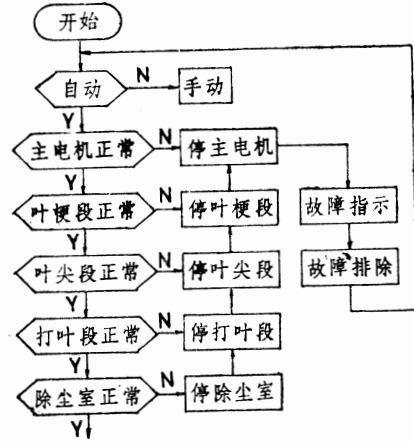


图3 主程序流程图

后, 程序判断主电机是否正常启动, 若无法启动, 诊断程序进行诊断, 并进行故障指示, 人工加以排除。主电机正常工作后, 则开始逆向分段定时启动, 即叶梗段、叶尖段、打叶段和除尘室段。若某段某设备工作不正常, 软件根据输入逻辑组合进行连锁控制, 并给予故障报警, 指示某台设备出现的故障。全线设备正常工作后, 程序进入自动循环工作状态, 同时对系统运行情况进行现场监视并显示, 随时将本线工作状态通过RS-232口告诉上位机, 以便对整个生产线进行管理。

三、系统特点

1. 全线只选用PLC-700主机一台作为控制机。这与意大利COMAS公司产品不同, 该公司选用了两台主机(西门子公司产品PLC)。PLC-700构成的硬件系统, 将控制系统中的部分输入点/输出点, 按16点一组构成寄存器输入/输出方式, 满足系统中输入信号为406点, 输出信号为339点的要求, 软件功能完全满足控制系统要求、因此不仅该控制系统成本降低, 而且也提高了系统可靠性。

2. 本系统由56块I/O板组成, 需要提供
(下转第19页)

环境地质质量多级模糊 综合评价预测系统

地质矿产部915水文地质大队 陈振武

摘要 该系统应用模糊数学多级综合评判的原理,用GW BASIC语言在高分辨GW BIOS3.0系统支持下在长城0520—CH微机上运行。采用汉字主菜单模块覆盖技术。该系统具有建立数据文件,计算作用系数,多级模糊评价,总评价,退出系统返回GW BASIC状态五种功能。主要用于对环境地质质量综合评价预测,对工程地质质量综合评价。

引言

1987年12月11日第42届联合国大会的169号决议确定1990~2000年为“国际减灾10年”。我国是世界上受地质灾害影响最为严重的国家之一,地质灾害防治工作必须贯彻以防为主,防治结合,综合治理的方针。要在全中国范围内开展地质环境区划,查明我国地质灾害的发育,分布规律,迅速加强重点地区的地质灾害勘查,监测和防治工作,尽可能减少灾害造成的损失⁽¹⁾。所以,开发了“环境地质质量多级模糊综合评价预测系统”,为保护环境,减少灾害服务。该系统第一次应用在10°拉萨幅区域水文地质普查报告中,评价工程地质质量,环境地质质量,取得了满意结果。在报告评审验收会上被评为质量优秀的水文地质普查报告。

一、运行环境

系统名为“HJDMP”,其运行环境为长城0520—CH微机及其兼容机,紫金3070—24针汉字打印机。采用GW BASIC语言,GW BIOS3.0系统及汉字字库1,汉字字库2(长城微机开发公司1985)。

二、HJDMP系统原理及功能

应用模糊数学多级综合评判原理,视环境地质质量为一系统工程,按层次分成若干子系统,每个子系统由若干因子组成。用寻找“安全岛”的理论将测区分成若干小区(分区),分别评价各小区的环境地质质量。用模式识别原理对全区(亚区)评价预测。当然,对建筑工程质量、城市环境质量等均可评价预测。

1. HJDMP主控模块(ZKMK)

功能 在屏幕上显示主作业选择提示信息。当键入主作业选择编号后,可进入相应的子模块。详见图1主控模块结构图。

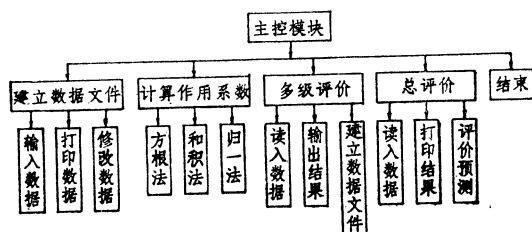


图1 主控模块结构图

2. 建立数据文件子模块(JLSHJ)

功能 按子系统因子的隶属度建立数据文件并存盘。具有输入数据, 输出数据, 修改数据等功能。

原理 采用随机文件OPEN〈文件名〉AS〔#〕〈通道号〉LEN〈长度〉命令。文件名用一个字符串变量代替, 用人机对话形式建立你需要的数据文件。在修改数据功能中巧妙的利用数组与记录数的对应关系, 为修改数据提供方便。

图中 ZX——要建立的文件数;

N——变量数;

D——计数器。

模块结构详见图2。

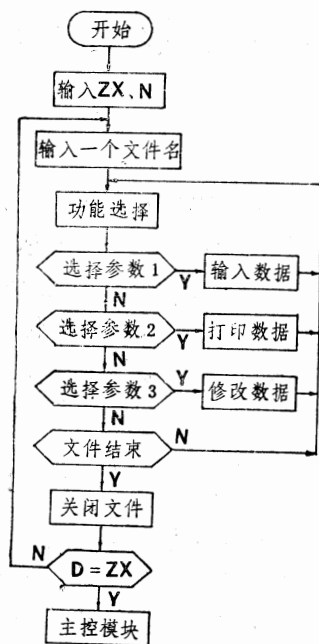


图2 建立数据文件子模块框图
数据文件结构如下:

字段	变量名	长度
1	YS	2B

含意: 单因子隶属度

3. 计算作用系数子模块 (ZWEYX)

功能 计算子系统对系统 (因子对子系统) 的作用系数。

原理 根据层次分析模型建立判断矩阵, 判断矩阵表示同一层次的子系统对系统的重要程度或同一层次的因子对子系统的重要程度。假设A系统中有n个子系统 B_n , B系统中有n个因子 C_n , 则可建立 B_n 对A与 C_n 对B的判断矩阵⁽²⁾, 详见图3、图4。

A	B_1	B_2	...	B_j	...	B_n
B_1	b_{11}	b_{12}	...	b_{1j}	...	b_{1n}
B_2	b_{21}	b_{22}	...	b_{2j}	...	b_{2n}
...
B_i	b_{i1}	b_{i2}	...	b_{ij}	...	b_{in}
...
B_n	b_{n1}	b_{n2}	...	b_{nj}	...	b_{nn}

图3 A系统判断矩阵

B	C_1	C_2	...	C_j	...	C_n
C_1	c_{11}	c_{12}	...	c_{1j}	...	c_{1n}
C_2	c_{21}	c_{22}	...	c_{2j}	...	c_{2n}
...
C_i	c_{i1}	c_{i2}	...	c_{ij}	...	c_{in}
...
C_n	c_{n1}	c_{n2}	...	c_{nj}	...	c_{nn}

图4 B子系统判断矩阵

B_i 、 C_j 为行, B_j 、 C_i 为列, b_{ij} 、 c_{ij} 为元素。i、j等于1, 2, ..., n, B_i 、 B_j 、 C_i 、 C_j 为排序序号, b_{ij} (c_{ij})表示 B_i (C_i)相对 B_j (C_j)的重要程度。当 $B_i - B_j > 0$, B_i 比 B_j 重要。 $B_i - B_j = 0$, B_i 与 B_j 同等重要。 $B_i - B_j < 0$, B_j 比 B_i 重要。上述关系用公式(1)表示之, 以此计算各元素。

$$\begin{cases} b_{ij} = B_i - B_j + 1 & \text{当 } B_i - B_j \geq 0 \\ b_{ij} = 1 / (|B_i - B_j| + 1) & \text{当 } B_i - B_j < 0 \end{cases} \quad \dots (1)$$

将同一层次中的因子对子系统的重要程度称为作用系数。其计算方法归纳为三种。

①方根法

$$A_i = \overline{\varphi_i} / \sum_{j=1}^n \overline{\varphi_j} \dots\dots\dots (2)$$

$$\overline{\varphi_i} = \sqrt[n]{M_i}; M_i = \prod_{j=1}^n b_{ij}$$

②和积法

$$A_i = \sum_{j=1}^n \overline{b_{ij}} / \sum_{j=1}^n \overline{\varphi_j} \dots\dots\dots (3)$$

$$\overline{b_{ij}} = b_{ij} / \sum_{k=1}^n b_{kj}$$

$\overline{b_{ij}}$: b_{ij} 正规化处理后的值。

③归一法(最大特征向量法)

$$A_i = b_{in} / \sum_{i=1}^n b_{in} \dots\dots\dots (4)$$

b_{in} 矩阵的行向量, 每一行的N列是最大的特征向量。程序运行框图从略。

4. 多级模糊评价子模块(DJMPJ)

①功能 对子系统分别评价, 打印结果, 建立中间性数据文件。

②原理 选用模糊评判主因子突出型模型, 记作 $M(\cdot, V) \dots \text{III}^{(3)}$ 。

给定因子集合 $V = \{V_1, V_2, \dots, V_n\}$, 模型 II 在评价时要考虑所有的因子, 因为各因子对评价所起的作用不同, 所以, 得到一个模糊子集 \tilde{A} 。 $\tilde{A} = \{A_1, A_2, \dots, A_n\}$, A_i 是作用系数。

给定评语集合 $U = \{I, II, III, VI\}$, I 好、II 较好、III 较差、VI 差, 分别取值为 0, 1, 2, 3。根据因子集合与评语集合的关系, 得到模糊关系矩阵 \tilde{R} 。

$$\tilde{R} = \begin{pmatrix} r_{11} & r_{12} & \dots & r_{1n} \\ r_{21} & r_{22} & \dots & r_{2n} \\ \vdots & \vdots & \dots & \vdots \\ r_{m1} & r_{m2} & \dots & r_{mn} \end{pmatrix}$$

根据模糊子集得到模糊矩阵 $\tilde{A} = (A_1, A_2, \dots, A_n)$ 。 $\tilde{B} = \tilde{A} \cdot \tilde{R}$, \tilde{R} 为评价矩阵。

③判别因子的选择及单因子隶属度计算
环境地质特征(判别因子), 决定于该区的自然因素和人为因素。自然因素包括气候、水文, 地形及地质条件等; 人为因素包括人口密集程度, 人民的生活和生产方式, 工农业经济发展状况, 地区开发方针政策和有关自然环境保护法的制度和执行。以 $1:10^6$ 拉萨幅环境地质质量评价为例, 该区与环境地质质量有关的单因子选取 16 个, 分为 9 个子系统。系统中每个单因子的性质, 分级标准及取值详细列表 1。

单因子隶属度利用概率统计的方法确定。各因子的数据来源, 一收集有关单位的观测资料, 如气象子系统降雨量, 平均气温, 冻害等。二是实地调查, 如地貌子系统中的海拔高程、切割深度、坡度角等。根据诸因子的数据变化范围与自然生态环境和人类生存发展的关系, 确定等级区间。如年平均降雨量 600~800mm 最适合, 划为 I 类; 600~500mm, 降雨量就少了一些, 划为 II 类; 500~400mm 造成干旱, 划为 III 类; 小于 400 及大于 800mm, 不是干旱就是水灾, 定为 VI 等。现以地貌子系统中海拔高程因子 $x(4)$ 调查统计资料为例详细说明单因素(子)隶属度计算。根据因子 $x(4)$ 的数据对照分级范围与等级, 所在的等级取值为 1, 其余等

表1 环境质量系统因子分级取值表

系统名	子系统名	因子名	因子符号	变量性质	因子分级区间与取值			
					I	II	III	IV
					0	1	2	3
环境地质质量系统	气象	降雨量(mm/r)	x(1)	连续值	800~600	600~500	500~400	<400>800
		平均气温(°C)	x(2)	连续值	10	10—5	5—0	0
		冻害(强度)	x(3)	定性	无	弱	较强	强烈
	地貌	海拔高程(米)	x(4)	连续值	<4000	4000~5000	5000~6000	>6000
		切割深度(米)	x(5)	连续值	<500	500~1000	1000~1500	>1500
		坡度角(度)	x(6)	连续值	<30	30~40	40~50	>50
	构造与地震	地震裂度(度)	x(7)	连续值	<7	7—8	8—10	>10
		活动断层(强度)	x(8)	定性	弱	较弱	较强	强
		断裂(密度)	x(9)	定性	稀疏	较稀疏	较密	密集
	地质灾害	斜坡作用(个/百km ²)	x(10)	连续值	<1	1—5	5—10	>0
		砂漠化(km ² /百km ²)	x(11)	连续值	<1	1—3	3—6	>6
		盐碱化(土中易溶盐%)	x(12)	连续值	<1	1—5	5—10	>10
	水文	面蚀度强(%)	x(13)	连续值	<10	10—30	30~50	>50
	植被	植被覆盖率(%)	x(14)	连续值	>50	50~30	30~10	10~0
	土地	人口密度(人/km ²)	x(15)	连续值	<1	1—4	4—10	>10
	开发	利用程度(%)	x(16)	连续值	无—20	20—50	50~80	>80
	地球化学	地方病(%)	x(17)	连续值	<10	10—30	30~50	>50
	人为工程	道路交通(密度)	x(18)	定性	无一很疏	较疏	较密	密
		水利工程(密度)	x(19)	定性	无一很少	较少	较多	多

表2 x(4)数据统计与隶属度计算

样品号	x(4)	>4000 I	4000~5000 II	5000~6000 III	>6000 IV
1	3700	1	0	0	0
2	3800	1	0	0	0
3	4500	0	1	0	0
4	5200	0	0	1	0
5	6500	0	0	0	1
6	3650	1	0	0	0
7	4900	0	1	0	0
8	4750	0	1	0	0
9	3900	1	0	0	0
10	5100	0	0	1	0
11	3950	1	0	0	0
Σ11		5	3	2	1
频数		0.45	0.27	0.18	0.019

级为0。将对比结果形成一个矩阵,将各列求和,形成一个行矩阵,(5 3 2 1),各元素除以样品总数11,得到各等级中的频数;将频数最大的等级为x(4)的隶属度。I类频数最大为0.45,根据表1中规定第174小区x(4)取值为0。

5. 总评价预测模块(ZHPJ)

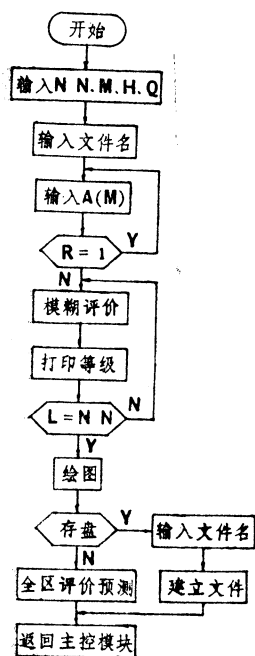
功能 根据多级评价所建立的数据文件,对测区环境地质质量评价预测。具有打印计算结

果,打印图(表),全区环境地质质量评价预测,也可以将计算结果存盘。

原理 同多级模糊评价。

评价预测:环境地质质量评语集合中的四个等级,分次赋予不同程度地质问题的内涵。Ⅰ类好,可以预测自然环境保持良好。Ⅱ类较好,预测地质环境产生了轻微的地质灾害(地质问题)。Ⅲ类较差,预测自然环境遭受破坏,且产生了较严重的地质灾害。Ⅳ类差,预测自然环境遭受严重破坏,并产生了严重地质灾害,自然环境呈恶性循环。

程序见图5。



其中
 NN——分区数;
 M——子系统数;
 H——行数;
 Q——列数。
 R = 1 表示作用系数输入有错,则重新输入;
 R = 0 表示作用系数输入正确。

图5 总评价子模块框图

系统运行结束后,返回到GW BASIC状态。

三、应用效果

现以拉萨幅1:10⁶环境地质质量评价为例,将测区分成192个长方形小区,评价结果可由计算机自动打印成图。

从评价图知分区环境地质质量好的和较好的占54.76%;差的和较差的占45.24%,因此,全区环境地质质量评价是好的。该地地处高原,地广人稀交通不发达,自然生态保持较好这个评价是客观的。

测区环境地质质量等级(好,较好,较差,差)在平面上有明显的规律性,反映了区内不同地区自然地质环境和人类活动营力的差异。现按等级分述如下:

1.环境地质质量好的小区主要分布在藏北湖区和西藏南部边缘浅切割地区,人类活动影响微弱,以自然生态环境为主。预测自然环境好,地质灾害单一且强度不大。

2.环境地质质量较好的小区主要分布在怒江源头、拉萨河上游高海拔地势平缓地带,这里人类活动影响轻微,自然环境保持较好。预测自然因素引起的环境破坏较小。

3.环境地质质量较差的小区主要分布在怒江下游、拉萨河、雅鲁藏布江谷地之上部及喜马拉雅山一带,自然环境差,预测地质灾害严重。

4.环境地质质量差的小区主要分布在拉萨河下游,雅鲁藏布江曲水一泽当段,隆子河等第一个谷肩以下的于旱半干旱河谷地区及雅鲁藏布江下游大弯峡一带,人类活动对自然环境破坏程度大,至使生态环境恶化,地质灾害严重。

四、结束语

1.HJDMP运用了软件工程的理论,设计目的明确,准备充分,选择数学模型慎重,因此,编程调试时间短而使用时间长,经运行证明计算正确,使用方便,效果好。软件容易维护,可移植性强,并建立了齐全的资料。各模块均可单独使用,也可和其它程序配合使用。

2.在评价预测中应用了智能字段,分别对四个等级的自然环境预测,实践证明是成

稳定可靠的自动复位抗干扰技术

华北电力学院 王攀峰

摘要 本文针对MCS—51系列单片机提出了一种新的自动复位抗干扰技术，与传统的抗干扰技术相比，它能在干扰较强而且持续时间久的情况下强行复位，使单片机更可靠地工作。

一、引言

在工业控制系统、机电一体化产品和智能仪表中，往往由于中断信号丢失或电磁场干扰，使程序计数值PC出错或取出错误代码，从而程序失去原有的控制功能。目前一般采用如下两类抗干扰方法。

1. 硬件方法

利用单片机的一个I/O口（如P1.7）和单稳集成电路（如74LS123）设计一个单稳电路，在程序正常运行时，由P1.7口不断地发出脉冲。一旦程序发生故障，P1.7口不再发脉冲，此时自动复位电路向单片机的复位脚发出一个复位脉冲，迫使单片机复位^[1]，但该方法占用了单片机的一个输出口，而且，当干扰出现时，单稳电路仅发出一个复位脉冲，如果复位不成功，就不再发脉冲了。该方法不能排除长时间干扰对微机的影响。

2. 软件方法

用软件设置一个计数器^[2,3]（如秒计数器），在程序正常运行时，不断地对计数器清0；出现故障后，程序不再对计数器清0，当秒计数器超过设定值时，由中断程序利用软件对其复位。该方法的缺点是当出故障的程序若关掉中断源（计数器），或长期连续出现干扰时，也就失效了。

由此可见，现有的自动复位抗干扰技术对软硬件的依赖性很强，而且无法克服连续长时间干扰给微机带来的影响，所以在干扰较强而且持续时间久的环境下，或对控制过程要求万无一失的场合，使单片机可靠地工作将十分困难。为此，我们提出了一种无法用软件停止其计时的自动复位抗干扰技术，它的最大特点是能在长期连续干扰的情况下，强行复位，具有很强的抗干扰能力。

二、基本原理

功的。在编写随机文件时，把记录号与数组结合起来，解决了各子系统字段不同的矛盾，既减少了占用内存，操作又方便。

3.该系统针对小比例尺环境地质勘察而开发的，对单因子评价用概率统计方法，对大比例尺也可应用。只是对小区划分要细，因子数据调查加密测网即可。

参 考 资 料

[1]侯海生。“以防为主防治结合综合治理

——朱训（地矿部部长）同志谈地质灾害防治工作，《中国地质矿产报》，1990—10—08，第一版。

[2]王碧泉 陈祖荫著。《模式识别理论方法和应用》，地震出版社，1989，349—354页

[3]黄克中 毛善培编著。《随机方法与模糊数学应用》，第二篇模糊数学及其应用，同济大学出版社，1987

微机在数字电路调试中的应用

广东商学院计统系 张 胜

摘要 介绍通用微机在数字电路调试时作为各种调试脉冲发生器的编程方法。这种方法无须改变微机的任一软、硬件结构,只须在其打印机接口上插上一引线插头,配合编制的程序,就可得到所需的各种脉冲——宽度可调的正脉冲、负脉冲、沿、负沿等。脉冲产生的方式和时序由键盘控制,脉冲输出可单步可连续,可一路可多路,全由编程设定,非常灵活方便,在调试仪器缺乏时尤为实用。

一、硬件结构及I/O口地址

利用打印机接口,对其编程以获得各种脉冲信号用作调试。IBM PC/XT机的打印机适配器有两种:一种是打印机适配器独立的,其I/O口地址为378H~37AH;另一种是单显适配器和打印机适配器做在一起的,它的I/O口地址为3BCH~3BEH。前者系统配置的显示器是彩显而后者是单显。以前者为例,说明编程方法,有关的硬件结构如图

1。在打印机适配器上可以直接编程又能在25脚D型插座上输出脉冲的有两个地方:一个是数据口74LS374,其输出对应D型插座的2—9脚,选通地址为378H;另一个是控制口74LS174,它的输出要经OC门反相,再分别接至D型插座的1、14、16、17脚,选通地址为37AH。D型插座的18~25脚接地。这些脚可作为输出脉冲信号的地线(即参考点)。下面的例子都以控制口为例。

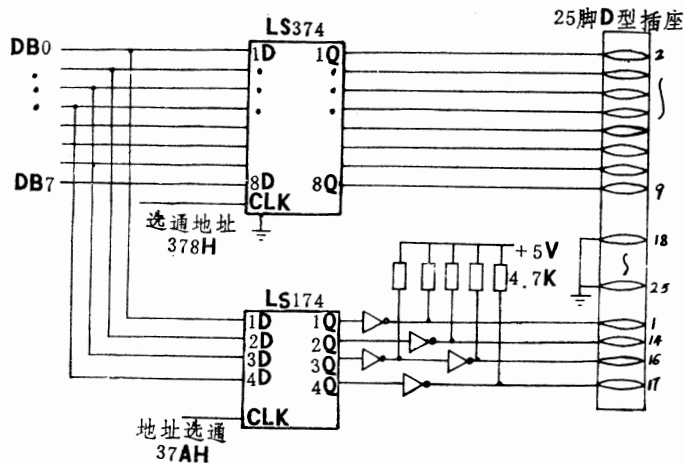


图 1

二、单步(沿)脉冲输出

参照图1,假如起始时令数据线 $DB_0 \sim DB_3$ 低四位为“1”,写入控制口LS174,

1脚上得到的是低电平“0”。若再令数据线为低,写入控制口,这样,1脚上就可得到一个由低向高变化的上升沿。编程如下:

1. MOV DX, 37AH ; DX寻址I/O口
2. MOV AL, OFH ; 初始化、使1脚为低电平
3. OUT DX, AL ; 输出
4. MOV AL, OEH ; 使1脚变高电平
5. OUT DX, AL ; 输出

如果使1脚再变低,就可获得一个正脉冲。

6. MOV AL, OFH
7. OUT DX, AL

这个正脉冲的宽度为执行6、7两条指令的时间,执行指令6需4个时钟周期,执行指令7需12个时钟周期,则脉冲宽度为: $16 \times 210\mu\text{s} = 3.36\mu\text{s}$ 。这里210μs是PC/XT机的时钟周期。这个宽度是PC/XT机上用此方法获得脉冲的最小宽度。下降沿、负脉冲的编程与此类似。实际应用时,总得寻找一种方法,使得脉冲在需要时产生,不需时停发。下面这段程序,任意击一键在1脚上输出一宽度为10ms的正脉冲。

```

MOV DX, 37AH ; 初始化,使1脚低
MOV AL, OFH
OUT DX, AL
J1: MOV AH, 1 ; 调用BIO判键符功能码
INT 16H ; 有否键
JZ J1 ; 无键; 再判
MOV AL, OEH ; 有,则使1脚变高
OUT DX, AL
MOV CX, 3179 ; 置1脚高时的延时计数
J2: LOOP J2
MOV AL, OFH ; 1脚复低
OUT DX, AL
JMP J1

```

上述例程与前面相比增加了两个功能:其一调用BIO判键符功能(DOS判键符功能也一样),以实现脉冲输出由键盘控制;其二加入LOOP延时以获得脉冲的所需宽度。

已知脉冲宽度T求CX寄存器的计数值的近似公式为:

$$CX \approx \frac{T}{17 \cdot 210\mu\text{s}}$$

计算时T的单位为μs。

三、周期性脉冲输出(脉冲发生器)

上述程序产生脉冲的方式都是单步的,即按一次键产生一个脉冲。同样道理,可以编制程序产生连续脉冲。下面这段程序,即能实现此功能。程序一执行,就能在1脚上获得周期为38KHZ,宽度为3.36μs的正脉冲,任按一键脉冲停止产生。

```

MOV DX, 37AH ; I/O口地址
J1: MOV AH, 1 ; 调用BIO判键符功能
INT 16H ; 有否键符
JNZ J2 ; 有、则返回
MOV AL, OEH ; 无、则输出脉冲
OUT DX, AL
MOV AL, OFH
OUT DX, AL
JMP J1
J2: RET ; 返回

```

四、多路脉冲输出

多路脉冲输出在数字电路调试时是很有用的。比如说,一路脉冲给数字电路的CP端,另一路给数字电路的 \overline{R} (复位)端,然后在键盘上控制这二路脉冲的输出来模拟实际的工作过程,观察数字电路的输出是否与预计的一致。多路脉冲输出时,脉冲的产生可以是同步的,也可以是异步的。同步脉冲的输出比较简单,只要给一个口的某几位同时编程就可以了。异步脉冲的输出编程因脉冲时序要求的不同而难易有别。下面给出一个完整的程序,经ASM(MASM)汇编,(下转第28页)

打印机状态设置 在报表程序设计中的应用

浙江省计算技术研究所 王敬农 应好仁

摘要 为了使计算机程序设计员编制出有特殊要求的既美观又实用的报表,就必须熟练地掌握打印机状态的动态控制,只有这样,才能随心所欲地设计出各种美观实用的报表。本文通过对打印机状态设置的探讨,提出了一整套报表程序设计中动态控制打印机状态的方法。

一、打印机的状态设置

一个计算机系统,一般都包括主机、终端、打印机等设备,无论你选择的打印机型号如何(如LQ系列,3070,1724,M2024等),都有其工作状态的设置。一般来讲,打印机状态可通过主机发命令或对打印机《选择按键》的操作来完成。本文只介绍主机发命令的方式。

1. 打印机的初始状态

任何型号的打印机在出厂时都设置了初始状态。也就是说,打印机在出厂时,厂家依据一种国际通用标准,将打印机的各种状态设置好,存放在打印机的ROM中。这样打印机一旦和主机接通后就可按标准方式输出报表文件。

2. 打印机的状态改变

任何型号的打印机都有多种状态,用户可根据需要向打印机发出命令来改变打印机的工作状态。改变的方式可通过主机向打印机发出状态设置命令,通常人们比较喜欢用BASIC语言,PASCAL语言,C语言和shell过程控制语言等编制一段改变打印机工作状态的程序。例如

```
LPRINT CHR$(27);CHR$(48) ;
```

指ASCII码27和48送到打印机的BASIC语言程序,它执行后可使打印机选择1/8时列间距。但这种改变是一种静态的,一旦通过命令设置后,打印机就处在设置下的状态,只有到状态重新设置或打印机关机后才能改变。这种设置方式显然在报表中是不够的。因为只有采取动态的打印机状态设置方式,在报表中不断改变其工作状态,才能实现一些特殊报表的输出,以满足用户的要求。

二、在报表程序中打印机 状态的动态设置

1. 打印机的状态参数

为了更好地在报表程序中设置打印机的状态,首先要查阅厂商的打印机手册中的指令格式说明,看一下所选用的打印机有那些功能可在用户报表中应用。然后根据报表的要求,重点查阅有关功能的参数,为在报表程序中对打印机作动态设置准备。值得注意的是,在查阅参数时一定要查阅打印机预设命令的参数。

2. 报表程序中打印机状态的动态设置

知道打印机状态的有关参数后,就可在报表程序中应用了。以LQ—2500K打印机为

例, 介绍设置打印机状态的两种方法。

① 报表程序语言设置打印机状态

利用报表程序语言本身所提供的功能来动态控制打印机状态是最理想的方法。根据打印机在计算机系统中所处的地位不同, 一般可分为系统打印机和终端打印机两种。这两种打印机都可用报表程序语言来实现动态控制。方法是利用它的打印命令(一般是PRINT命令)来向打印机送控制码。下面分两种不同的动态要求来介绍, 第一种是逐行进行动态控制, 实现步骤如下:

步骤1 在打印机状态要改变的行开始前, 用打印命令送打印机状态控制码。

例1 `print ascii28, ascii14; {打印字体放大一倍}`

`print "打印内容" {打印实体}`
它可将“打印内容”放大一倍输出, 经常用于表头部分。

例2 `print ascii28, "-", ascii1;`
`{打印字体带单底划线}`
`print "打印内容" {打印实体}`
它可将“打印内容”划上单底划线。

步骤2 在恢复和更改打印机状态行开始前, 用打印命令送恢复或更改控制码。

例3 `print ascii28, ascii14; {打印字体放大一倍}`
`print "打印内容" {打印实体}`
`print ascii27, ascii64; {恢复打印机出厂参数}`
`print "新打印内容" {打印实体}`

它可将“打印内容”放大一倍输出后, 恢复打印机的出厂参数后再打印“新打印内容”。

例4 `print ascii28, "-", ascii1;`
`{打印字体带单底划线}`
`print "打印内容" {打印实体}`
`print ascii27, ascii64;`
`{恢复打印机出厂参数}`
`print ascii28, "-", ascii2;`
`{打印字体带双底划线}`

`print "新打印内容" {打印实体}`

它是将“打印内容”划上单底划线输出后, 更改成双底划线输出“新打印内容”, 中间有一句恢复出厂参数的命令很重要, 因为双底划线命令不能取消单底划线命令, 如果不用中间恢复出厂参数命令时, 就不能打出双底划线。故如搞不清后面的打印命令能否取消前面的打印命令时, 建议均采用恢复出厂参数命令来取消前面的打印命令。这样就不会引起控制混乱。

步骤3 在程序结束时不论打印机处在什么工作状态, 都需用打印机恢复出厂参数的命令来恢复打印机状态, 以免影响别的用户使用。

第二种是在一行中用多个状态来控制打印机输出。我们用下例来说明实现步骤。

例5 `print "打印内容1", "打印内容2",`
`"打印内容3"`

要求“打印内容1”是单底划线状态, “打印内容2”是标准状态, “打印内容3”是双底划线状态。其实现步骤如下:

- (1) `print ascii28, "-", ascii1;`
`{打印字体带单底划线}`
- (2) `print "打印内容1", ascii13;`
`{将进入打印缓冲区的“打印内容1”输出}`
- (3) `print ascii27, ascii64;`
`{恢复打印机出厂参数}`
- (4) `print " ";`
`{空格数与“打印内容1”等长, 将打印头定位在“打印内容2”的起点}`
- (5) `print "打印内容2";`
`{标准状态打印“打印内容2”}`
- (6) `print ascii28, "-", ascii2;`
`{设打印字体带双底划线}`
- (7) `print "打印内容3";`
`{“打印内容3”带双底划线输出}`
- (8) `print ascii27, ascii64`
`{恢复出厂参数}`

由上可见,不论是逐行控制还是在一行中控制都能根据用户的要求在报表程序中动态控制打印机的状态。作者曾以打印机状态动态控制方式,以底划线替代表格线,将一张需两页打印纸的表格打印在一页打印纸上,实现了用户的特殊要求。另外,对于终端打印机来讲,用户还需查厂商的终端手册找到打开、关闭终端打印机的参数,用打印命令在程序的开头和结尾处分别送出

```
print ascii27, "[5i";
{启动终端透视打印}
print ascii27, "[4i";
{设终端打印机为脱机状态}
```

其余程序无需作任何修改,就可在终端打印机上使用。

②非报表程序语言设置打印机状态

凡是有打印命令的语言都可用来设置打印机状态,但是和系统的工作环境有着直接的联系。如采用 BASIC 语言,就需要运行报表程序的同时可运行 BASIC 语言程序的系统环境,所以用非报表程序语言设置打印机状态,必须考虑到系统的工作环境,所以通常我们选用 C 语言,因为不论 DOS 操作系统还是 UNIX (XENIX) 操作系统环境,一般都可使用 C 语言的目标程序,方法是先用非报表程序语言将打印机状态设置的程序编制好,一种设置编一个小程序。然后在报表程序中分别调用。现有两个 C 语言程序,通过它们来说明非报表程序的打印机控制程序在报表程序中的应用

```
程序 1 scham = setrow—column . c
{程序名 setrow—column . c}
#include <stdio . h>
main ( )
{
    FILE *fp;
    fp = fopen ( "/dev/ttyH15",
        "w" ); /* 打开打印机 ttyH15 */
    fprintf ( fp, "\034S%c%c\r",
        0, 1 );
```

```
fprintf ( fp, "\034T%c%c\r",
    0, 1 );
fprintf ( fp, "\034U\r" );
fprintf ( fp, "\0333%c\r", 25);
}
```

这是一个将打印机设置成以中文方式输出,行、列间的距离均为一个点针,这样可使打印出来的表格线上下左右互相连接,使表格外观美丽。

```
程序 2 scham = setrecover . c
{程序名为 setrecover . c}
#include <stdio . h>
main ( )
{
    FILE *fp;
    fp = fopen ( "/dev/ttyH15",
        "w" );
    fprintf ( fp, "\033\100" );
}
```

该程序是使打印机状态恢复出厂参数。分别对它们进行编译生成 setrow—column 和 setrecover 两个可执行文件。

在报表程序设计时,可将它们作为操作系统命令来用。若权限不对时可用操作系统命令更改它们的权限。若报表设计语言有调用操作系统文件的命令的话(一般是 call 命令或 run 命令),可在调用的地方以 call setrow—column, call setrecover 来实现对打印机的状态设置。若没有调用操作系统文件的命令,在 UNIX (XENIX) 操作系统下,用 sh setrow—column, sh setrecover 来实现调用;在 DOS 操作系统下,可在调用处用暂停命令,使程序运行暂停,然后用 command /C tetrow—column, command /C setrecover 来实现调用,再用继续命令使程序继续运行。由此可见用非报表程序设计语言来动态控制打印机也是可以实现的,但受到环境的影响和报表程序设计语言本身功能的限制,做起来比较复杂。因此;我们认为还是尽可能使用报表程序语言本身来控制打印机比较方便,也可两者结合使用。

基于结构化分析的数据字典工具

浙江大学管理信息与计算技术研究所

钱 涛

金连甫

胡上序

摘要 本文阐述了数据字典(DD)工具基于的理论思想——快速原型法和结构化分析方法,并结合数据库理论中数据字典的概念,运用盒子结构分析方法,给出了管理信息系统(MIS)原型建造环境中的DD工具的设计。

关键词 数据字典 结构化分析方法 盒子结构

一、引言

MIS原型建造环境是为提高MIS软件生产率和质量而研制的。MIS原型建造环境由需求分析计算机辅助工具和系统设计工具组成。数据字典工具是机辅设计的重要工具,它起到承上启下的作用,所谓承上,即对调查得到的信息进行分析 and 总结;所谓启下,即为下阶段管理信息系统的设计提供依据。所以说,数据字典工具是MIS建造环境的瓶颈。

二、DD工具理论思想

1. 快速原型法

传统的MIS开发模型是建立在系统开发人员能够获得稳定不变的用户需求的基础上。实际上,只有等到用户对开发中的MIS有了更多的了解和经验,开发人员才有可能得到正确和稳定的需求,因此,需求是不断变化的。这样,快速原型法便被引进到MIS开发中来了。

原型是一个建议系统某些特征的可执行的具体模型,快速原型法是对一系列原型快速建立和评价的处理方法。

快速原型法的出发点,一是及早发现以致减少开发各阶段的错误,二是能在较短的时间里完成整个系统。为了达到第一个目

的,这种方法允许在开发各个阶段不断反馈和修改,同时尽量缩小与用户在需求理解上的差距。快速原型法是首先建立一个能够反映用户主要需求的模型,让用户实际看一看未来系统的概貌,以便判断哪些功能符合要求,哪些方面还需要改进,然后将该模型反复修改,最终建立比较符合用户要求的新系统。

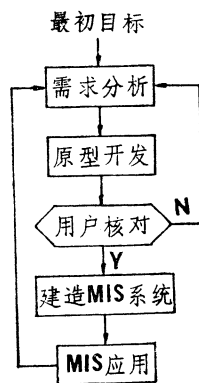


图1 快速原型法

但是,快速原型法在MIS开发中的应用受到一定程度的限制,因为原型不是最终的MIS,需要经常和反复的修改。而在人工开发MIS的情况下,快速原型法是不能够充分发挥其潜在的优势,实现真正意义上的“快”。MIS原型建造环境便是提供解决这种问题的途径。

2. 结构化分析方法

倘若管理信息系统的成功是以用户满意为评判标准的话,那么需求分析是开发管理信息系统的第一步和最重要的一步。需求分析的方法有多种多样,主要有结构化分析法(Structured Analysis),结构化分析和设计技术SADT,软件需求工程法SREM,PSL/PSA,有限状态机FSM和Petre网等技术。由于应用广泛,成熟和简便,结构化分析方法独领风骚。结构化分析SA是以“结构化”方式进行系统定义的分析方法,它的基本手段是“分解”和“抽象”;它的基本思想是采用“自顶向下”的分解方法,这种方法使用结构化工具对一个复杂的系统逐渐地向下分解成尽可能独立的子系统,模块和子模块等,直到足够清楚地简明地描述一个功能。SA不是用繁琐的语言,而是用图解的方法简明地表达一个系统的逻辑模型。

结构化分析方法用到的工具为数据流图、数据字典、结构化英语、判定表和判定树。

数据流图DFD(Data Flow Diagram)用于表示通过现有系统或推荐系统的数据流动;数据字典DD(Data Dictionary)用于定义流经系统的数据;而结构化英语、判定树和判定表则用于描述数据流的动作过程。

三、数据字典概念

虽然很多人从不同角度对数据字典进行了描述,但至今尚没有一个统一的定义,本文集大家之说概括如下:

定义1 数据字典就是有关信息集中的库,以及管理这些信息的软件。

按照与数据库管理系统的联系程度来划分,数据字典可以分为主动型(active)和被动型(Passive)两种。

定义2 在运行时间里,数据库管理系统使用了数据字典中的定义,那么这种数据字典称为主动型数据字典(active data dictionary)。

定义3 如果数据字典是独立于数据库管理系统的,则该种数据字典被称为被动型数据字典(Passive data dictionary)。

主动型数据字典必须包含对物理数据库精确的最新描述的数据,数据库管理系统能够访问这些数据。而且在主动型字典里只有唯一的形式化的定义供软件各组成部分分享。

对于被动型字典,定义有多义。数据库管理系统所用的定义独立且区别于数据字典中所存的定义和各种不同程序所用的定义。

主动型数据字典和被动型字典有较大的区别。

表1 主动型和被动型字典的区别

主 动 型 字 典	被 动 型 字 典
1.只有一份定义	1.冗余定义 ——存在不一致性 ——需要仔细维护 ——需要额外的存储空间
2.系统取决于字典有效性	2.系统不受字典失败影响
3.字典成为系统的瓶颈	3.一个系统不影响另一个系统

四、数据字典设计

为了达到减少冗余,实现一致性等目的, MIS建造环境的数据字典应按主动型字典设计, 下面给出DD工具的模式。

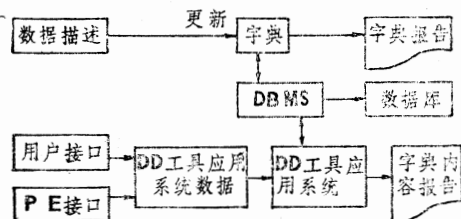


图2 DD工具系统模型

1. DD工具的功能设计

本文引入盒子结构来对DD工具进行分析和设计。有关盒子结构方法 (Box-Structured Design) 在此不赘述, 详参有关资料。

从功能上来讲, DD工具可以划分成四个模块——编辑、维护、管理和输出。

①DD编辑

鉴于DD的重要性和应用广泛性, 作者所设计的DD工具具有两重性, 一是DD工具作为MIS建造环境的一个组成部分, 对整个系统有依赖性; 二是可以把数据字典工具视为独立的工具, 可以单独使用。用盒子结构方法中的盒子描述如下:

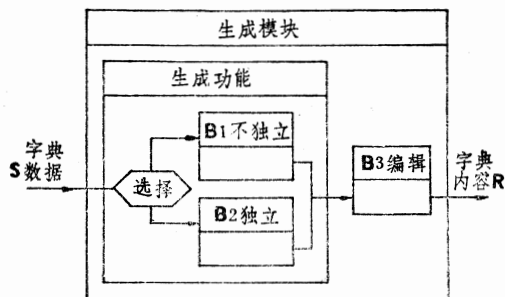


图3 DD生成模块黑盒

B1对已在需求调查部分出现的DD信息, 数据字典工具可以自动地生成数据字典中相应的条目及其它一些信息。自动生成的

信息不能进行修改和删除, 以确保一致性, 但可以用于编辑其它一些信息, 以便更加详细地规定出各个成分的定义, 组成和说明。这种自动生成的功能在数据字典的编辑模块中完成。B2增强了数据字典工具在应用方面的广泛性和灵活性。B3提供一定的编辑规则, 详细地规定出各个成分的定义、组成和说明。

用BDL复合语句表示图3的黑盒子结构的描述如下:

define BB字典生成

stimulus

MIS描述信息: 数据

response.

字典内容: 数据

proc

if DD工具为独立工具

then use B2 人工生成条目

else use B1 自动生成条目

fi,

use B3 编辑详细字典内容

crop

考虑到篇幅, 下面不再给出相应模块的黑盒结构。

②DD维护

维护功能可分为两大部分, 一是对数据实体维护, 二是数据字典内容的一致性检查。对于前者, 数据插入和删除是在充分考虑具体字典间相互关联的基础上设计的; 对于后者, 则要对整个数据字典的内容检查其余性和不完整性。所谓余性, 即数据字典中的数据元素不属于任何数据流和数据存储。所谓不完整性指数据流与数据存储中含有未定义的数据元素。如果数据字典作为MIS建造环境中的工具, 则还要检查DD中是否含有在调查报告中未出现的条目。

③DD管理

对DD中所有条目进行分类和排序, 另外

还向用户提供多种查询功能, 根据用户给定的条目名或编号, 读出该条目的详细定义以及相关信息。

④DD输出

参见MIS建造环境DD工具模型图。输出设计成两部分, 一部分是对数据字典本身结构特征的输出, 这类输出报告主要是供DD工具设计人员用的, 便于对DD工具的维护和修改; 另一部分是DD内容报告输出, 这类输出报告面向MIS系统分析和设计人员, 为设计一个完整的特定的MIS提供依据。

2. 数据字典内容

形式上, 一个数据字典DD可以由六元组描述。

$DD = \langle DE, DS, DF, DB, PL, EE \rangle$

其中 DE——数据元素描述集合;

DS——数据结构描述集合;

DF——数据流描述集合;

DB——数据存储描述集合;

PL——处理逻辑描述集合;

EE——外部实体描述集合。

以下, 用小写字母表示集合中的元素, 用小写字母带括号表示集合中的元素, 且其本身又是个集合。

①数据元素de描述

$de ::= (de-name, longname, type, length, abp())$

其中 de-name——数据元素名称;

longname——数据元素别名;

type——元素类型;

length——元素取值范围和含义;

abp()——有关的处理逻辑集合。

数据元素别名的设立考虑到了不同数据库管理系统的特点, 事实上, 数据元素在物理存储时取了不同名称, 在逻辑上是相同的, 数据元素别名可以很好地表示这种关系。

②数据结构ds描述

$ds ::= (name, str, abp())$

其中 str——数据结构组成, 它表示数据之间的关系;

定义 4

(1) 若 e_i 是数据元素, $i = 1, \dots, m$,
(e_1, e_2, \dots, e_m) [E];

(2) 若 s_1, s_2, \dots, s_n 是str, $j = 1, \dots, n$, (s_1, s_2, \dots, s_n) [str];
str满足 $str = (e_1, e_2, \dots, e_i) U (s_1, s_2, \dots, s_j)$,

则str为一个数据结构ds的组成。

③数据流df描述

$df ::= (df-name, sdf, ddf, str, rf, prf)$

其中 sdf——数据流的来源;

ddf——数据流的去向;

rf——流通量;

prf——峰值流通量。

sdf和ddf或是某个处理逻辑, 或是某个数据存储, 或是某个外部项。

④数据存储db描述

$db ::= (db-name, idf(), odf(), str, id())$

其中 idf()——流入的数据流集合;

odf()——流出的数据流集合;

id()——立即存储要求集合。

定义 5

若db满足上面的组成, 且 $idf() U odf() \subseteq db$, 则db为一个数据存储。

⑤处理逻辑pl描述

$pl ::= (pl-name, idf(), odf(), prg)$

其中 prg——功能描述。

定义 6

dp是一个处理逻辑, 则idf()和odf()必须同时存在。

⑥外部实体ce描述

ce:: = (ce-name, idf(), odf(),
number)

其中 number——作为该外部项具体对象的数据。

定义7

记外部实体流入的数据流为term1, 记外部实体流出的数据流为term2,

则外部实体描述满足 $term1 = term2$

3. 各子数据字典关系

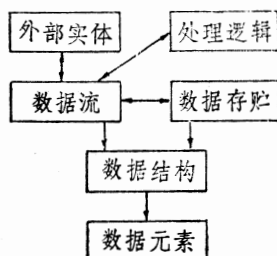


图4 数据字典逻辑层次关系

图4反映了相对同一层数据流图的各个具体数据字典之间的相互关系, 它决定了编辑各个具体数据字典的先后次序, 同时也给数据字典的一致性检查提供了依据。

五、结 束 语

如果说MIS原型建造环境是为了提高

上接第2页

一定功率的扩展电源。根据理论计算需要3台扩展电源才能满足56块I/O板同时工作。但是, 在实际工作中, 所有I/O板中的控制点并不是同时工作的。根据工艺要求, 每次同时工作的最大点数只占输入/输出总数的60%~70%。因此可减少一台电源。实践证明, 这一方案完全可靠。

3. 程序分成相对独立程序模块, 在调试中可按先调设备本身→设备与设备→段本身→段与段→全线联调, 一条复杂的控制线的调试便可成为方便而简单的工作。

MIS生产效率, 那么数据字典工具作为环境的瓶颈, 在数据传递, 存储和频繁变更方面提供了方便, 它的灵活性也将扩大它的应用面。

参 考 书 目

- [1] Victor Weinberg, Structured Analysis, Prentice-Hall, Inc. 1980
- [2] Charles J. Wertz, The Data Dictionary: Concepts and Uses, QED Information Sciences, Inc. 1986
- [3] Luqi, Software Evolution Through Rapid Prototyping, Computer, Vol.12, No.5, May 1989
- [4] B. Boehm, A Spiral Model of Software Development and Enhancement, Computer Vol.21, No.5, May 1988
- [5] J. J. Martin, Data Types and Data Structures, Prentice-Hall International, U.K., 1986
- [6] 杨文龙. 《软件工程》, 科学技术文献出版社, 1989
- [7] 王燮臣等. 《管理信息系统》, 浙江大学出版社, 1989

四、结 论

1. 本系统于1990年11月正式在海林烟厂投入运行, 半年来, 运行正常, 无故障发生。

2. 与国外80年代同类产品相比较, 性能相同而价格低。

3. 生产效益显著提高。同时, 操作简单, 维修方便; 大大减轻了工人的劳动强度。

4. 抗干扰能力强。周围不仅有15kw大电机17台, 而且环境温度高, 但PLC至今工作仍然稳定可靠。

5. 系统模块化, 扩展方便灵活。

交互式图形程序库的设计方法

西南交通大学 景 红

摘要 本文着重论述了在微型机上运用高级语言与Auto CAD系统开发建立交互式图形程序的设计思想和设计方法,并对Auto CAD与高级语言程序的链接、数据交换等设计中的若干重要的技术问题的处理方法作了详尽地介绍。可广泛地使用于桥梁、土建、岩土、机械等多种领域的计算机辅助程序设计中。

一、前 言

目前,在我国广泛流传的Auto CAD软件,是一种在微型机上运行的辅助绘图软件包,由于它的绘图功能甚强,开放性好,在建筑、机械、轻工等多种行业的计算机辅助设计中有着较广泛的应用价值。然而Auto CAD缺乏对图形进行分析和设计的功能,计算能力也很差,很难独立完成诸如铁路桥梁之类具有大量复杂的分析设计与工程计算的辅助设计任务。但是,只要我们将高级语言的分析计算能力强和运算速度快的特点与Auto CAD强有力的绘图功能结合起来,再借助数据库存取管理数据,以此充分发挥各自的优点,即可组成一个功能很强的软件系统。同时,也可由此方便地与其他工程计算软件进行数据传输。

因此,在计算机辅助设计中,解决高级语言与Auto CAD的链接与数据交换,即“接口程序”的设计,实现“设计计算——绘图”一体化,不仅十分重要而且具有很大的实际意义。

二、交互式图形程序库的设计

硬件环境 IBMPC系列微型机

DPX-2000平板式绘图仪

技术方案框图如图1示。

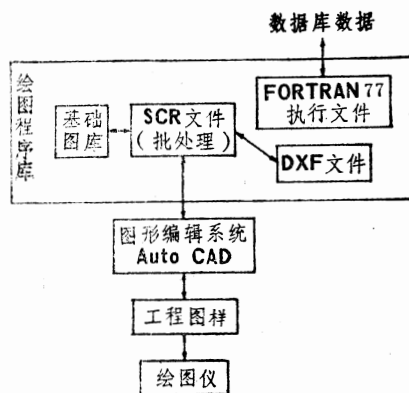


图1 技术方案框图

执行程序框图如图2示。

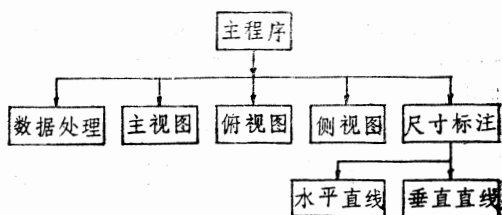


图2 执行程序模块框图

程序结构模块化,便于扩展和维护,实现参数制图,加快绘图速度。

通过实参与形参的替代和开辟公用区(COMMON)数据共享两种方法完成模块间的数据联系,尽量减少所占用的内存空间,提高运行速度。

由于大型工程图都是由各种各样的单元

图形组成,而每个单元图又有多种类型,因此,使得其描绘文件相应地有比较复杂的逻辑判断结构,这时可充分利用FORTRAN77的块IF语句。

三、图形设计中几个重要问题的处理

1. 软件接口的处理

Auto CAD提供了两个能与外部通讯的文件:一个是图形交换文件.DXF——它是图形生成环境参数和一系列具有固定格式的图形实体参数在一定格式下的堆积:一个是命令文件.SCR——它是Auto CAD各种命令及其参数在一定格式下的堆积。使用.DXF文件速度快,但格式复杂,且只有十多种基本作图实体,对于常用的画阴影线、标注尺寸等命令则无法转换。使用.SCR文件连接计算程序和Auto CAD中的绘图命令类似于一种批处理方式,比较简便,但处理速度慢,因此综合引用两者,取长补短,效果会更佳。

尽管再复杂的工程图形,其全部图形终究可分为三类基础图形:(1).基本图形(直线与线段、圆与圆弧等);(2).几何相交图形(两直线相交、直线与圆相交等);(3).参数化图形(这类图形本身只有大小相似的区别,没有形状上的差别)。对前两类基础图形可按Auto CAD中的复合线实体进行处理,而对第三类图形(包括图中的各种符号,剖面线、地面线等阴影线,所造的新字,固定的图形及附注等)按Auto CAD中的块进行处理极为方便。

一个完整的.DXF文件应由标题段、表格段、块段、实体段及文件结束段五部分组成。其结构极其繁杂。但实际上描述图形的参数都在实体段(该段含有图形实体,包括每一块的标记)及块段(该段含实体的块定义,所描述的实体是由图形内的各块组成)

中,故只需使用高级语言按实体段及块段的格式去生成.DXF文件,即可在Auto CAD中引用DXFIN命令读入此文件,绘出所设计的图形。

根据以上分析,笔者采取以基础图库取代.DXF文件中的块段,完成实体块定义任务;同时使用.SCR文件完成“批处理”任务。这样,不仅进一步将.DXF文件的所用格式压缩到最简程度——只含实体段,克服了.CF文件的基本作图实体少以及诸如阴影线不能被转换等缺点。大大提高了绘图速度。而且解决了在引用汉化版Auto CAD作为图形软件支撑系统时带来的将图一律按旧图处理,至使图形交换文件中的块段不能被读入,导致该文件中的相应块实体不被接收的问题。

基础图库由单位块图组成。调用该库中的某一块实体时,其相应图形的实际尺寸单位即为该块图X,Y比例因子。基础图库的建立方法:在Auto CAD系统下,进入绘图新图状态,然后将剖面线等图形分别用绘图命令绘成|*|的图形,用BLOCK命令建成单位块实体存入基础图库中。

.SCR文本文件可用DOS系统下的EDLIN命令来编辑。.SCR文件对空格、分号、斜杠等特殊键有着特殊的定义,不能随意增减。

当第一次装入Auto CAD时,用C>HCAD HQT WW(或C>ACAD HQT WW)命令调用文本(WW.SCR),系统就将置缺省图形名HQT,并从WW.SCR文件中去读命令。WW.SCR文件产生一个用缺省图形名为名字的新图形,并依次发出所置的读图库,读DXF文件等命令,最后自动绘制出所设计的工程图样。其一次完成多项任务,就类似于DOS系统的批处理。

2. 图形交换文件的生成

只含有实体段和结束段的DXF文件的格

式:

```
0
SECTION          实体段开始
2
ENTITIES          实体段标志
( 图形实体 )
0
ENDSEC           实体段结束
0
EOF              文件结束
组成图形实体的各种图形元素的DXF格
```

```
CHARACTER*10 DXFNAME
WRITE(*,*) '请输入DXF文件名:'
READ(*,*) DXFNAME
OPEN(8, FILE=DXFNAME, STATUS='NEW', ACCESS='SEQUENTIAL')
BACKSPACE(8)
WRITE(8, 10)
WRITE(8, 30) 3, 5, 3, 25
WRITE(8, 30) 3, 25, 53, 25
WRITE(8, 30) 53, 25, 53, 3
WRITE(8, 30) 53, 3, 13, 3
WRITE(8, 30) 'PMX' 3, 5, 50, 20
CALL DIMH(3, 26, 53, 26, 30, 50, 0, 0)
CALL DIMH(3, 4, 13, 4, 1, 10, 1, 0)
CALL DIMV(3, 5, 25, 53, 57, 20, 1, 0)
WRITE(8, 40)
10 FORMAT(' 0'/'SECTION'/' 2'/'ENTITIES'/)
20 FORMAT(' 0'/'INSERT'/' 8'/'0'/' 2'/'A3'/' 10'/'F9.6'/' 20'/'F9.6'/'
1 ' 41'/'F9.6'/' 42'/'F9.6'/' 50'/'0')
30 FORMAT(' 0'/'POLYLINE'/' 8'/'0'/' 66'/' 1'/' 40'/'0.04'/'41'/'
1 '0.04'/' 0'/'VERTEX'/' 8'/'0'/' 10'/'F9.6'/' 20'/'F9.6'/' 0'/'VERTEX'/'
2 ' 8'/'0'/' 10'/'F9.6'/' 20'/'F9.6'/' 0'/'SEQEND'/' 8'/'0')
40 FORMAT(' 0'/'ENDESC'/' 0'/'EOF')
ENDFILE(8)
COLSE(8, STATUS='KEEP')
END
```

其中 'PMX' 为基础图库中的一单位块实体。

3. 其它

为了使所用的图形进一步明朗化、美观、易读, 可进行环境设置。同时可迅速地把某层分离出来并对其单独进行显示、查询等, 提高图形程序的调试速度。

把三个视图的原点和绘图比例均处理为

式, 在Auto CAD的使用手册中有详细的说明, 本文不再赘述。

这里需要我们解决的问题是图中尺寸的标注。为此笔者特设计了两个常用的尺寸标注通用子程序——水平尺寸线标注和垂直尺寸线标注。程序见附录。

有了以上基础, 用高级语言编写生成所设计图形的.DXF文件就比较容易实现了。在此, 我们仅以一简单图形的绘制来说明。

可变, 即通用的图形交换文件生成程序自动地依据图框尺寸和图形的一些有关参数设定三个视图的原点坐标, 同时, 自动地根据图形的具体参数变化, 挑选绘图比例。这样既统一了绘图图型, 又保证了图形版面的美观。

环境设置

层名	状态	颜 色	线 型
0	ON	7(WHITE)	CONTINUOUS
1	ON	5(BLUE)	HIDDEN
2	ON	3(GREEN)	DASHDOT

四、结 束 语

本设计方法为笔者在长期的计算机辅助设计与绘图软件包的研制工作的总结。在很多工程绘图运用中效果甚佳。

附录 水平尺寸标注通用子程序

```

SUBROUTINE DIMH(X1, Y1, X2, Y2, YZ, TT, IP, IQ)
40  FORMAT(' 0'/'LINE'/' 8'/'0'/' 10'/F9.6/' 20'/F9.6/' 11'/F9.6/
1 ' 21'/F9.6)
50  FORMAT(' 0'/'SOLID'/' 8'/'0'/' 10'/F9.6/' 20'/F9.6/' 11'/F9.6/
1 ' 21'/F9.6/' 12'/F9.6/' 22'/F9.6/' 13'/F9.6/' 23'/F9.6)
60  FORMAT(' 0'/'TEXT'/' 8'/'0'/' 10'/F9.6/' 20'/F9.6/' 40'/'0.30'/
1 ' 1'/'13'/' 50'/'0'/' 41'/'0.85'/' 51'/'10.0'/' 7'/'HZ'/' 72'/'
2 ' 1'/' 11'/F9.6/' 21'/F9.6)
IT=NINT(TT)
IF (IP.EQ.0) THEN
WRITE(8,40)X1, Y1+0.0625, X1, YZ+0.18
WRITE(8,40)X2, Y2+0.0625, X2, YZ+0.18
ELSE
WRITE(8,40)X1, Y1-0.0625, X1, YZ-0.18
WRITE(8,40)X2, Y2-0.0625, X2, YZ-0.18
ENDIF
IF ((X2-X1).LT.0.6) GOTO 70
WRITE(8,40)X1+0.18, YZ, X2-0.18, YZ
WRITE(8,50)X1+0.28, YZ-0.045, X1+0.28, YZ+0.045, X1, YZ, X1, YZ
WRITE(8,50)X2-0.28, YZ-0.045, X2-0.28, YZ+0.045, X2, YZ, X2, YZ
XX=(X2-X1)/2+X1
IF (IT.LT.100) GOTO 65
WRITE(8,60)XX-0.2, YZ+0.15, IT, XX, YZ+0.15
RETURN
65  WRITE(8,60)XX-0.49, YZ+0.15, IT, XX, YZ+0.15
RETURN
70  WRITE(8,40)X2+0.18, YZ, X2+0.40, YZ
WRITE(8,40)X1-0.18, YZ, X1-0.40, YZ
WRITE(8,50)X1-0.28, YZ-0.045, X1-0.28, YZ+0.045, X1, YZ, X1, YZ
WRITE(8,50)X2+0.28, YZ-0.045, X2+0.28, YZ+0.045, X2, YZ, X2, YZ
IF (IQ.EQ.0) THEN
WRITE(8,60)X2+0.30, YZ-0.1, IT, X2+0.30, YZ-0.1
ELSE
WRITE(8,60)X1-1.2, YZ-0.1, IT, X2-1.2, YZ-0.1
ENDIF
RETURN
END

```

高级语言与汇编语言混合编程技术

云南大学地球科学系 庄 真 人民银行昆明市分行 胡家富

摘要 本文论述IBM微机的FORTRAN、PASCAL、C等高级语言调用汇编过程的机制,并通过实例介绍混合编程的方法。这种混合编程既能保持高级语言的高效、简捷和模块化的特点,又能发挥汇编语言对声音、图象、色彩有较强的处理能力的长处。

一、问题的提出

高级语言通俗易懂,使用高级语言编写程序方便简捷、容易调试、可读性强。但是,像FORTRAN、PASCAL、C等语言对声音、图象、色彩方面的控制能力很有限(C语言稍强一些),而使用汇编语言编程却能有效地提高程序在这些方面的处理能力。因此,有时需要“混合编程”,即高级语言写的主程序与汇编语言写的子程序分别编译之后进行链接及运行。这既保留了高级语言程序的模块化设计能力和高效简捷的特点,又弥补了汇编语言在声音、图象、色彩及硬件控制功能的不足。对于一些频繁调用的子程序,采用汇编语言编写还能进一步提高程序的运行速度。

二、高级语言程序调用

汇编语言子程序的机制

MICROSOFT的FORTRAN、PASCAL、C等语言已被设计成在目标文件(.OBJ)这一级上与汇编语言兼容,所以高级语言源程序经过编译生成的目标文件(例如A.OBJ)与汇编语言源程序经过汇编生成的目标文件(例如B.OBJ)只要链接在一起(例如LINK A+B)就可生成一个可执行的映象文件(例如A.EXE),实现混合编程。

当高级语言程序调用汇编语言书写的外部过程时,高级语言源程序中的调用子例程

语句或引用函数语句不必作任何改动,就如同调用同一种高级语言书写的过程一样。另一方面,用汇编语言书写外部过程时则必须遵循某些特殊的要求,以便正确地实现控制转移和参数传递。

在汇编语言里,调用一个过程(PROCEDURE)有两种方式:一种是远程(FAR)调用,即跨段调用;另一种是近程(NEAR)调用,即段内调用。从FORTRAN或PASCAL调汇编过程时只使用远程调用,而从C调汇编过程时可以是远程或近程调用。参数传递也有两种方式:一种是地址传送(又称为引用方式),即远程调用时需要传送4个字节的段址及偏址,而近程调用时每个参数只要传送2个字节的偏址;另一种方式是值传送(又称为值方式):即直接传送参数的值。从FORTRAN调汇编过程时一律使用“地址传送”,即使实参是常数或表达式,也是传送这些常数或表达式值的临时存贮单元的地址。从PASCAL调汇编过程时一般使用“值传送”,但当实参是以VAR说明的变参时,则使用“地址传送”。从C调汇编过程时使用“值传送”方式。

每当控制执行到调用程序中的调用外部过程语句时,系统会自动地做下述两件事:

第一,系统会自动地把调用程序传递来的各个实参(值或地址)按排列的先后依次压入栈中。另外,在函数调用的场合,如果

函数返回值是 4 字节或 8 字节的实型值，则该值将存贮在由调用程序生成的一个临时变量中，这个临时变量（值或地址）也将跟在上述实参（值或地址）之后压入栈中。

第二，接着，系统会把调用程序中紧跟在调用语句之后那个语句的代码地址（即“返回地址”）也压入栈中，以便在过程调用结束时能正常地返回调用程序。

在书写汇编过程的源程序时，过程的名称必须说明成 PUBLIC，以便其它语言的程序可以调用它。与汇编过程共用的其它数据段也必须说明成 PUBLIC，以便实现数据共享。汇编过程本身则要根据远程或近程调用适当地说明成 FAR 或 NEAR，以便进行正确的寻址。此外还要注意下述三个方面：

① 书写汇编过程时，首先必须把 BP 压入栈中，以便保留基址寄存器中旧的 BP 值（旧的帧面指针），然后把这时的栈顶位置移入新的基址指针 BP。这个新的 BP 相对于栈中各个参数的位移是不变的，于是栈中各个传递参数都可以很方便地由离开新 BP 的某个相对位移来寻址，而且当栈中继续压入更多的量使 SP 不断改变时也不影响对各传递参数的寻址。在汇编过程返回之前，只要把旧的 BP 弹出栈，就能恢复旧的帧面指针，保证调用程序的运行环境复原。

② 如果汇编过程要使用 AX、BX、CX、DX 之外的其它寄存器，最好先把这些寄存器原先的值压入栈中保存起来，等到即将退出汇编过程时，再按相反的次序弹出恢复之，以保证调用程序的原局部数据空间不受破坏。

③ 汇编过程的 RET 语句起控制返回的作用。它有两种形式：如果先前的调用程序未向汇编过程传递过任何参数，则汇编过程结束时只要使用 RET 语句，系统就会自动地将栈内的 4 字节返回地址弹出，控制将根据此地址返回到调用程序去执行；如果调用程序先前曾向汇编过程传递过若干参数（值或地址），那么汇编过程结束时必须使用 RET n

语句，系统则先把栈内 4 字节返址弹出，再把原先压入栈内的全部参数（总共 n 个字节）弹出，使栈腾空，然后控制才返到调用程序去执行。这样就保证了在经过多次调用之后栈不会溢出。

三、FORTRAN 程序调汇编过程

FORTRAN 调外部过程时，参数传递均采用地址传送方式，并且都是远程调用。无论整型或实型数据，FORTRAN 的缺省存贮单元都是 4 字节长。

1. FORTRAN 程序调汇编子例程

例 1 此子例程将屏幕设置成 640×200 的图形模式，从主程序接收三个参数（X 坐标、Y 坐标、颜色号），在指定位置按指定颜色显示此像素点，并把此位置记录在数据区的两个变量中供主程序将来查询这个位置。

```
主程序
CALL PUTDOT(300,100,3)
END
子例程
DATA SEGMENT PUBLIC 'DATA'
CURSORX DW ?
CURSORY DW ?
DATA ENDS
DGROUP GROUP DATA
CODE SEGMENT 'CODE'
ASSUME CS:CODE,DS:DGROUP,SS:DGROUP
PUBLIC PUTDOT
PUTDOT PROC FAR
PUSH BP
MOV BP,SP
MOV AH,0
MOV AL,6
INT 10H
LES BX,DWORD PTR [BP+14]
MOV CX,ES:[BX]
LES BX,DWORD PTR [BP+10]
MOV DX,ES:[BX]
LES BX,DWORD PTR [BP+6]
MOV AX,ES:[BX]
MOV AH,12
INT 10H
MOV CURSORX,CX
MOV CURSORY,DX
MOV SP,BP
POP BP
RET 12
PUTDOT ENDP
CODE ENDS
END
```

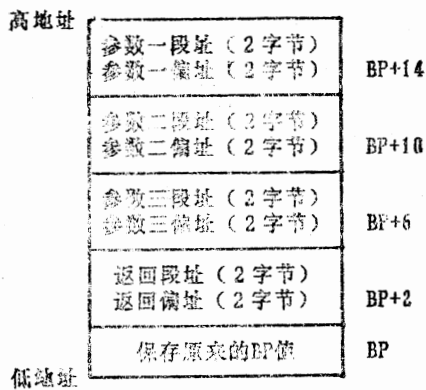


图 1 例1的栈结构

2. FORTRAN程序调汇编函数子程序

当汇编过程是一个函数子程序时，函数返回值的存放地点分三种情况：

①如果返回值是 2 字节的整型或逻辑型量，就存在 AX 寄存器中带回去。

②如果返回值是 4 字节的整型或逻辑型量，其高位存在 DX 中，低位存在 AX 中。

③如果返回值是 4 字节或 8 字节的实型量，则存放它的临时变量的 2 字节地址将跟在所有实参地址之后被压入栈中，其位置总是 BP+9。

例 2 此函数子程序接收两个 2 字节的整型变量，相加后的 2 字节整型函数值通过 AX 带回到主程序中。

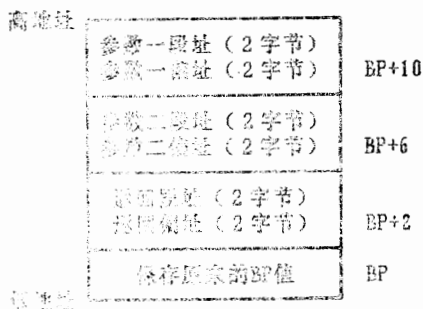


图 2 例2的栈结构

```

主程序
INTEGER*2 IA, IB, IADD
IA=3
IB=5
WRITE(*,*) 'IA+IB=', IADD(IA, IB)
END

子程序
CODE SEGMENT
ASSUME CS:CODE
PUBLIC IADD
IADD PROC FAR
PUSH BP
MOV BP, SP
LES BX, [BP+6]
MOV AX, ES: [BX]
LES BX, [BP+10]
MOV CX, ES: [BX]
ADD AX, CX
MOV SP, BP
POP BP
RET 8
IADD ENDP
CODE ENDS
END

```

四、PASCAL程序调汇编过程

PASCAL的过程调用都是远程调用。PASCAL程序中的汇编过程名应说明成 EXTERN 外部过程。参数传递的缺省方式是按值传送。但是，如果 PASCAL 过程调用语句中的实参之前有 VAR 限定词，则是按地址传送。按地址传送又分为两种：若实参之前是 VAR，则参数地址只按 2 字节传送，返址是 4 字节；若实参之前是 VARS，则参数地址按 4 字节传送，返址仍是 4 字节。PASCAL 的整型及实型数据的缺省存储单元都是 2 字节长。

1. PASCAL程序调汇编子例程

例 3 此子例程是将两个整型变量互换，注意此例的参数是按地址传送。其栈结构与例 2 相同，见图 2。


```

主程序
PROGRAM TEST11(INPUT, OUTPUT);
VAR
X, Y: INTEGER;
PROCEDURE SWAP(VARS A: INTEGER
; VARS B: INTEGER); EXTERN;
BEGIN
X:=3;
Y:=5;
SWAP(X, Y);
WRITELN('X AND Y ARE: ', X, Y);
END.

```

```

子程序
CODE SEGMENT
ASSUME CS: CODE
PUBLIC SWAP
SWAP PROC FAR
PUSH BP
MOV SP, BP
LES BX, [BP+10]
MOV AX, ES: [BX]
LES SI, [BP+6]
MOV CX, ES: [SI]
MOV ES: [BX], CX
MOV ES: [SI], AX
MOV SP, BP
POP BP
RET 4
SWAP ENDP
CODE ENDS
END

```

2. PASCAL 程序调汇编函数教子程序

汇编函数返回值的存放地点与 FORTRAN 一节所述相同。

例 4 此函数子程序接收两个 2 字节整型变量，相加后通过 AX 返回主程序。注意按缺省规定此例的参数是按值传送。

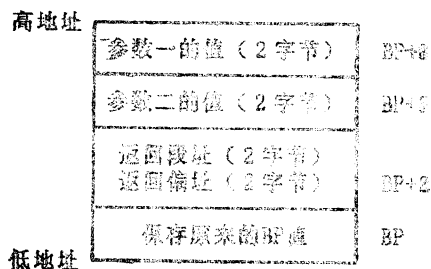


图 3 例 4 的栈结构

```

主程序
PROGRAM TEST13(INPUT, OUTPUT);
FUNCTION MADD(A, B: INTEGER);
INTEGER; EXTERN;
BEGIN
WRITELN(MADD(3, 5));
END.

```

```

子程序
CODE SEGMENT
ASSUME CS: CODE
PUBLIC MADD
MADD PROC FAR
PUSH BP
MOV BP, SP
MOV AX, [BP+8]
MOV CX, [BP+6]
ADD AX, CX
MOV SP, BP
POP BP
RET 4
MADD ENDP
CODE ENDS
END

```

五、C 程序调汇编过程

C 语言只提供“函数调用”方式。数据缺省的存储单元是 2 字节长。MICROSOFT 的 C 语言提供了小模式、中模式、紧凑模式、大模式、巨模式等五种内存模式。我们若使用小模式，则把汇编过程说明成 NEAR，参数传递是按数值传送，2 字节的函数值由 AX 返主，4 字节的函数值由 AX 和 DX 返主。若函数值大于 4 字节，汇编过程必须分配返回值的空间，然后把它的段址存入 DX，偏址存入 AX。建立返回值空间的最简单办法是在数据段中说明它。一旦建立起返回值空间，把返回的函数值存入该存储单元内，就能在调用结束时返主。要注意 C 语言编译程序对字母的大小写非常敏感，在编译汇编程序时要使用 MX 开关指令。另外，由于 C 语言编译程序在所有的变量及符号前面加了一条下划线，因此在汇编子程序中必须

在过程名前面加一条下划线，否则就无法链接。另一点也值得注意，C语言把实参压入栈的顺序与FORTRAN、PASCAL的相反，它是先把最后一个参数压入栈，其次才压入倒数第二个参数，依次类推，最后压入返回地址。被C调用的汇编过程的RET语句即使不写栈内参数的总字节数，过程返回时也会自动退栈。

例5 此函数子程序接收两个2字节的整型变量，相加后通过AX寄存器返回主程序。因为是近程调用，返址只用2字节的偏址。参数传送是按值传送。

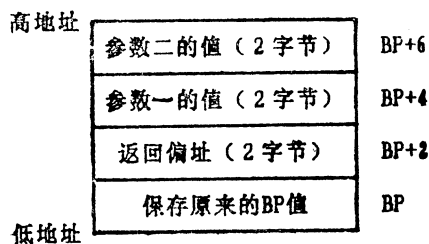


图4 例5的栈结构

主程序

```
main()
{
    int a,b;
    a=2;
    b=3;
    printf("%d \n",nadd(a,b));
}
```

子程序

```
_CODE SEGMENT
ASSUME CS:_CODE
PUBLIC _NADD
_NADD PROC NEAR
    PUSH BP
    MOV BP,SP
    MOV AX,[BP+6]
    MOV CX,[BP+4]
    ADD AX,CX
    MOV SP,BP
    POP BP
    RET
_NADD ENDP
_CODE ENDS
END
```

(上接第11页)

再由LINK链接生成EXE文件后便可使用。该程序功能如下：在键盘上按一次“A”键，1脚上输出一正脉冲；按一次“B”键，在14脚上输出一负脉冲；奇数次按“C”键在17脚上得到的是一上升沿；而击“C”键的偶数次则在17脚上输出一下降沿。阅读程序时请参看图1。另外，按ESC键返回DOS、其余键全被禁止。程序附后。

```
CODE SEGMENT PUBLIC 'CODE'
PROC FAR
    PUSH DS
    MOV AX,0
    PUSH AX
    ASSUME CS:CODE
    MOV DX,37AH
    MOV AL,00001001B
    OUT DX,AL
    MOV BL,1
    J1: MOV AH,0
        INT 16H
        CMP AL,1BH
        JZ J6
        CMP AL,'A'
        JZ J2
        CMP AL,'B'
        JZ J3
        CMP AL,'C'
        JNZ J1
        CMP BL,1
        JZ J4
        JMP J5
    J2: AND AL,11111110B
        OUT DX,AL
        OR AL,00000001B
        OUT DX,AL
        JMP J1
    J3: OR AL,00000010B
        OUT DX,AL
        AND AL,11111101B
        OUT DX,AL
        JMP J1
    J4: XOR BL,BL
        AND AL,11110111B
        OUT DX,AL
        JMP J1
    J5: MOV BL,1
        OR AL,00001000B
        OUT DX,AL
        JMP J1
    J6: RET
PULS ENDP
CODE ENDS
END PULS
```

XENIX环境下FOXBASE+文件与DOS 环境下dBASEⅢ文件的兼容

民丰造纸厂 章寿发

一、概述

在我国、绝大多数的中、小单位在计算机应用于企业管理、信息处理等方面都使用了dBASEⅢ数据库。随着计算机应用的深化,大多正在从单用户微机逐步向多用户系统发展。目前,多用户微机系列在XENIX环境下应用FOXBASE+的用户日渐增多。在新用户中,因没有掌握正确的复制方法或命令语句使用不当,用DOSCP命令复制的dBASEⅢ命令文件不能在FOXBASE+下运行。另外,由于目前处在由单用户向多用户过渡阶段,在特定条件约束下,仍有大量的在FOXBASE+中编制的命令文件需要复制到DOS下运行,同样存在复制后不能运行的问题。

本文通过对复制和运行过程中产生现象的分析,介绍如何进行操作,以达到FOXBASE+与dBASEⅢ的完全兼容。也许对使用多用户系统的新用户有一些帮助。

二、XENIX环境下FOXBASE+

对dBASEⅢ的兼容

《计算机时代》1990年第4期刘曙光在“在XENIX系统下复制DOS环境下dBASEⅢ文件应注意的问题”一文中介绍了一个正确的复制方法。文中提到的foxpget实用程序,是一个极为方便的软件。运行该程序后,只要指明被复制文件的驱动器盘号以及文件存放的子目录名,则软盘中所有dBASEⅢ文件都被自动地转换到XENIX系统下。

有时只需要复制盘上的某个文件,下面介绍怎样复制单个文件。

所有DOS的正文文件用CR+LF(回车符+换行符)来表示换行,而XENIX仅仅用LF表示换行。当用DOSCP命令转换DOS文件为XENIX文件时,原DOS文件中CR并未去掉。所以,当在命令行中含有CR的命令文件,在XENIX环境下的FOXBASE+中运行时,屏幕就会出现“句法错误”提示,命令文件不能执行。用显示文件的命令是不能发现CR存在的,但用modify command命令进入编辑状态就不难发现了。例如在程序中有这样一个命令行:

```
do while .t.
```

按下ctrl-B将光标移到该命令行的末尾,就可以看到这样的现象:

```
do while .t. ■
```

在光标前有一个空格,这就是CR。用左移键将光标移至前一列,按ctrl-T就能去掉CR,该命令行就能执行了。用这种方法操作,对一个大型文件来说是极其繁琐的。好在多用户FOXBASE+有一个转换程序dtx。其操作过程是:

doscp驱动器号: dBASEⅢ文件名 过渡
文件名

dtx<过渡文件名> FOXBASE+文件名

例如,将dBASEⅢ文件zc.prg转换到XENIX下,软盘在B驱动器:

```
doscp B: zc.prg p
```

```
dtx <p> zc.prg
```

经过dtx的转换,zc.prg就能在

FOXBASE+中正常运行。

三、DOS环境下dBASEⅢ 对FOXBASE+的兼容

用DOSCP复制的FOXBASE+命令文件，在DOS的dBASEⅢ中同样不能直接执行。其原因是在XENIX的文件中表示换行的只有一个换行符LF，少了一个回车符CR，用DOSCP复制时并没有加入CR。这个现象用DOS的TYPE命令可以发现。

由于没有CR，所有的命令行连接成一个命令行。用Edlin查看，发现行号没有规律，编辑无法进行。如果试图用dBASEⅢ的modify command命令编辑，则将会丢失程序的大部分命令行。但是可以在WORDSTAR的非文本文件编辑状态下编辑，操作方法如下：

在非文本文件编辑状态下读入被编辑的命令文件，这时可以看到在屏幕的右方最后一列上，每行都有一个“J”。该符号是LF在屏幕上的映像。将光标停在第二行的行首，并打开插入键，顺序反复按Del键、回车键、下移键。直至屏幕最后一列的每行都出现“<”，这是CR+LF在屏幕上的映像，表示在命令行中加入了CR。该命令文件就能在DOS的dBASEⅢ中正常运行了。

四、互为兼容

多用户FOXBASE+与dBASEⅢ是否完全兼容？一方面取决于两个软件的性能，另一方面还得看使用者的应用方法。对一些非执行文件，一般不太重视，其兼容性通常考虑较少。但对可执行文件来说，其兼容性就显得重要了。

.MEM、.DBF本身就具备互为兼容的条件DOS拷贝后能直接使用。多用户FOXBASE+对DOS环境下dBASEⅢ的.ndx文件兼容，在FOXBASE+中使用了.ndx文件后，自动产生.idx文件，以后即抛弃.ndx

文件。而dBASEⅢ对FOXBASE+的.idx文件却不兼容。由于两个操作系统的不同.prg文件，表示命令行换行的方法不一样，因而互不兼容，但只要设法去掉或插入回车符，就达到了.prg的互为兼容。

由于两个操作系统的差异，FOXBASE+与dBASEⅢ的某些相同命令也会不兼容。多用户FOXBASE+的所有资源都是面向多用户的，对数据库的任何写操作如append，pack，reindex，replace，browse，update等命令，只有当获得独占方式使用数据库的用户才能进行。例如用append命令对名为YWF数据库中某条记录的FY字段进行替换操作，下列程序：

```
use YWF
:
replace FY With x
:
```

在DOS的dBASEⅢ中能顺利执行，可是在多用户FOXBASE+下却显示错误信息：

```
Record is not locked
```

这是因为use YWF是以共享方式使用数据库的，在多用户下共享方式是不能对数据库进行写操作的。要使程序顺利执行，可在使用数据库命令后加一个独占子句，即

```
use YWF exclusive
:
replace FY With x
:
```

由于加入了一个EXCLUSIVE的可选项，程序就能在FOXBASE+执行了。但同时也带来两个不太理想的问题：第一，因为XENIX是面向多用户的，而DOS仅一个用户，所以EXCLUSIVE子句在dBASEⅢ中得不到承认，运行时又会出现“变量没找到”的出错信息，造成“对数据库的写操作，两个系统不完全兼容”的假象；第二，即使在多用户环境下，EXCLUSIVE子句将数据库的共享属性关闭，其它用户就不能对

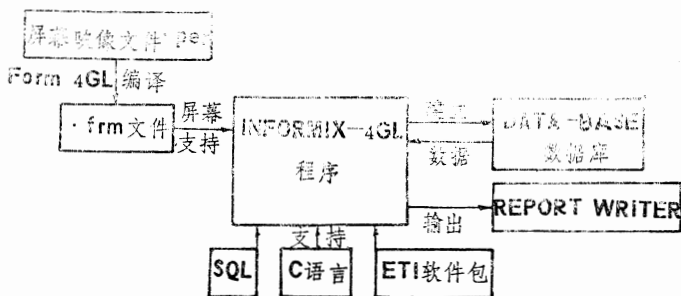
在INFORMIX. 4GL关系数据库语言 环境下实现表格的灵活打印

浙江省台州绣衣厂 王余星

一、INFORMIX—4GL语言简介

INFORMIX—4GL1.10.02关系数据库

多用户操作系统UNIX SYSTEM V RELEASE 4.0支持下，第四代多用户关系数据库语言的系统结构如下：



数据库由TABLE组成，每个TABLE由不同字段（COLUMNS）组成。打印输出由REPORT函数完成，但此语言函数之间的参

REPORT 函数名（参数表）

```

OUTPUT
FIRST PAGE HEADER
I' PRINT"XXXXXX" * 打印第一
    页页首 *
I" PRINT"XXXXXX"
PAGE HEAGER
II' PRINT"XXXXXX"
II" PRINT"XXXXXX" * 打印其余
    页页首 *
ON EVERY ROW
III' PRINT"XXXXXX" * 打印每行
    内容 *
III" PRINT库记录内容
ON LAST ROW
IV' PRINT"XXXXXX" * 打印最后
    一行内容 *
IV" PRINT ASCII 14
END REPORT
    
```

数传递不太方便，只有传值一种方式，因而给程序的通用性带来困难。简单的REPORT过程格式如下：

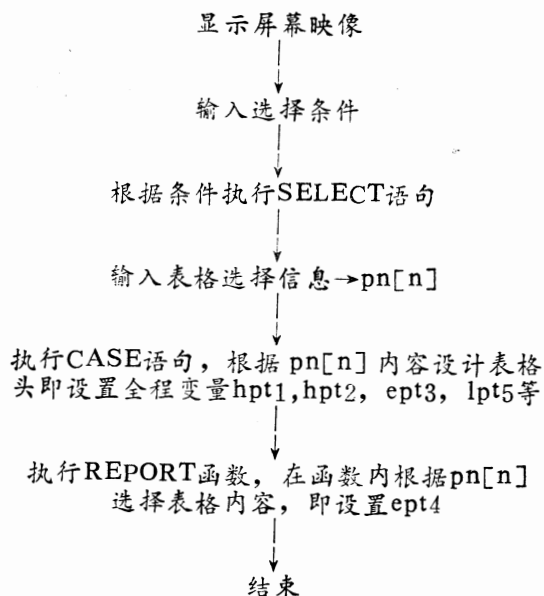
二、问题的提出及解决

我们要解决的问题是打印的表格列的长度可以改变，内容可以选择，用户可以对一个TABLE中的字段进行选择打印出满意的信息，特别对有很多字段的TABLE更有用。先取最简单的表格格式，假设第I与II组的内容相同，则我们可以设置两个全程字符变量（hpt1、hpt2），相应地III与IV也设置全程字符变量（ept3、ept4、lpt5），现在设法用一个简单的输入接口来改变（hpt1、hpt2、ept3、ept4、lpt5）这五个全程变量。我们的解决方法是运用一个整数数组pn[n]，与两个主要由case语句组成的过程。pn[n]内容与case语句的条件相对，

以达到对以上五个全程变量的控制。为更好的地说明问题，以下举个具体例子。

三、具体的实现

某厂的人事档案管理系统的打印程序流程图如下：



因为表格内容每行都在变化，所以 ept4 的内容要在report 函数内设置。这个程序解决了打印信息的冗余与用户的要求之间的矛盾。

四、更灵活的方法

从前一个例子看出，尽管可以选择栏目打印，但其表头的每一栏目都在case语句里被固定起来，对一些经常要改变表头格式的表格仍不适用。解决的方法是建一个表格格式TABLE，由以下字段组成

no integer * 打印位置识别 *

```

prt1 char(60) |
prt2 char(60) |
prt3 char(60) | * 表格单元内容 *
prt4 char(60) |
prt5 char(60) |
coluno integer * 库里字段编号 *
  
```

- no 数字 1 表示 prt1—5 为第一张表的表头设计用的制表符基元；
 数字 2 表示 prt1—5 为其它的表头设计用的制表符基元；
 数字 3 表示 prt1—5 为每行的制表符基元；
 数字 4 表示 prt1—5 为表末行的制表基元。

prt1—5 为供表格设计的基元 如 “|——”、“——|”、“序号”等。

coluno 数字控制着每行打印的库字段内容，如用数字 2 表示打印部门编号等，因为这被建成一个库记录，所以应用人员即可对其增加，删除和修改，程序里根据对库记录的选择设计表格的表头，表末及表格每行内容，运行这个程序，用户即可自己设计满意的表格。但这方法的缺点是操作人员在打印表格前必须设计表头，不象前一种方法那样操作起来很方便。

五、结 论

以上程序在UNIX操作系统下运行后，应用操作人员对此功能非常满意，尽管例子是用INFORMIX.4GL关系数据库语言编写，但其思想也可适合其它类似的大型关系数据库语言，只要略作修改，其它类型的表格也能完全成功地设计出来。

dBASEⅢ与C语言间数据传递新法

重庆大学计算机系 符云清

众所周知, dBASEⅢ 是一种深受广大计算机工作者喜爱的数据库语言, 而 C 则是一种有力的系统开发工具。若能使 C 程序和 dBASEⅢ 程序互相调用, 这无疑将提高软件开发的效率。要实现 C 和 dBASEⅢ 混合编程, 关键要解决两种语言参数传递问题。目前, 许多期刊上发表过关于 dBASEⅢ 与 C 之间参数传递的方法^[1], 概括起来, 可以分为两大类: 1. 通过文件进行的数据传递方法, 如通过双方都能访问的 .TXT 文件进行数据传递。这需对文件进行操作, 速度慢, 仅适于大量、批处理方式处理数据; 2. 不通过文件进行的直接数据传递, 如利用 dBASEⅢ 中的 CALL 命令, 将参数传递给汇编程序, 再转给相应高级语言程序。本文介绍一种新的数据传递方法, 即直接内存法。该法简单易行, 而且执行速度快。

在 dBASEⅢ 中有这样几条命令:

1. set segment—设置段地址。
2. poke 命令—向内存指定单元写一数。
3. peek 命令—从内存指定单元读一数。

在 C 语言中, 以 turbo C 为例, 也有类似函数:

1. poke (段地址, 偏移量, 数)。
2. peek (段地址, 偏移量)。

当然, 我们完全可以不使用这些函数, 而利用 C 语言灵活的指针类型和强制转换机制来实现。

利用上述函数, 便可以很方便地实现 C 和 dBASEⅢ 间的数据传递。调用方将要传递的参数转换成以字节为单位的数据区, 然后写入内存中指定单元。而被调用方从先约

定好的单元读出数据转换成参数。这便可以实现传值调用。如被调用方最后将参数的值再放入对应单元, 便可以实现变量参数调用。现举一例详细说明。

假设 dBASEⅢ 程序要调用一个 C 语言子程序。该 C 语言程序在屏幕上指定位置画一方框。需要的参数是左上、右下两点坐标, 即 (x-left, y-left) 和 (x-right, y-right)。相应 dBASEⅢ 程序段和 C 语言程序段为: 插入附录程序段, 其中 ×××× 表示段地址, u1, u2, u3, u4 表示地址偏移量。把 C 语言程序编译成 graph-rect . EXE 形式即可。

这只是一个简单的例子, 通过一些转换, 可以实现更灵活、更复杂的参数传递。上述方法对其它高级语言及汇编语言同 dBASEⅢ 传递参数同样适用, 而且执行速度快。

dBASEⅢ 程序段

```
set segment to ××××
poke u1 x-left
poke u2 y-left
poke u3 x-right
poke u4 y-right
RUN graph-rect
:
```

C 语言程序段

```
x-left = peek (××××, u1);
y-left = peek (××××, u2);
x-right = peek (××××, u3);
y-right = peek (××××, u4);
rectangle (x-left, y-left,
           x-right, y-right);
/* 画框子程序 */
:
```

参考文献

- [1] 吴坚, dBASEⅢ 与 C 语言进行数据传递的方法, 《中国计算机用户》, 1991(1)

改善企业MIS的可维护性策略

西南交通大学计算中心 于功第 路 枝

摘要 本文简述了企业MIS可维护性的重要性,并结合实际设计,重点叙述了为提高系统的可维护性应采取的策略。另外,还讨论了系统可维护性定义与分类及开发策略等问题。

一、问题的提出

目前随着计算机应用的普及,各行各业都开发出许多适用于不同管理目标的管理信息系统(MIS)。据大量资料表明,现在许多MIS的开发工作都把绝大部分的精力和费用耗费在系统的开发方面,而忽略了系统可维护的工作。现有的大多数系统工程和软件工程学的方法和工具也都只涉及到新系统的开发技术,而忽略了维护技术,而且系统维护软件工具也相当缺少。过去在MIS研究课题和费用分配上的趋势都表明,系统维护这个领域还没有引起人们足够的重视。事实上,系统维护工作非常重要。1978年美国对制造工业和非制造工业的数据处理管理人员进行了一次调查,结果表明管理问题比技术问题更重要,而维护现有系统比开发新系统更重要,大约65%的数据处理人员已投入到维护工作中而不是投入到开发工作中。调查结论是:除非找到崭新的解决办法,不然,需要维护的工作量还会大大增加,以至于很可能再也没有余力去从事新的系统开发工作。工业方面的调查报告也指出,在典型的工业应用中,系统维护期间所花的费用占总费用的70%,而劳力占总劳力的40%—95%。来自美国国防研究部(Department of Defense study)近期报道说,空军的航空电子控制系统中,开发软件的费用是平均

每条指令75美元,而该软件的维护费用却是每条指令4000美元左右。

从上述可见,目前各类软件系统的维护问题日益突出,而且维护费用直线增加,必须引起人们足够的重视。实践表明,在开发系统时就重视改善系统的可维护性,是解决上述矛盾的一种有效方法,它可使系统具有通用性和灵活性,且可减少后援维护费用。

二、系统可维护性定义与分类

1. 可维护性定义

一个管理信息系统的可维护性指标⁽¹⁾定义如图1所示。

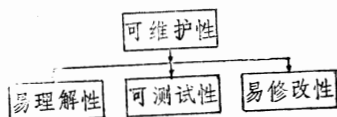


图1 可维护性层次示意图

其中 易理解性——指人们能够容易地阅读和理解该系统的结构、程序和文献。

可测试性——允许改变系统的测试方法,使人们能够容易地验证变更系统之后的系统正确性,从而确保系统的可靠性。

易修改性——对用户新要求的提出或环境的变化能较容易地对系统进行修改,以延长系统的寿命。

2. 可维护性分类

一个MIS可维护性可分成几类⁽²⁾。

①改正性维护——指在系统的使用期间发现错误，提出错误报告，诊断并改正错误的过程。

②适应性维护——为了和变化系统环境适当接口而进行的修改系统的工作。这种变化的环境，可以是增加新的硬件设备或更换旧的系统，也可以是新的操作系统或新的版本出现。

③完善性维护——当一个系统投入运行后，用户会希望增加新的功能，修改现有的功能等。这类工作往往占全部系统维护工作量的大部分。

④预防性维护——为了改进系统未来的可维护性或可靠性所作的工作。或为系统未来的改进提供更好的基础而修改系统的活动。

以上几种维护中，系统的完善性维护是最主要的，所占工作比重最大。

三、提高系统可维护性应注意的五项工作

我们在开发地区级电业局MIS时，把系统的可维护性放在重要的地位，并充分考虑了提高整个系统的可维护性策略。首先把下列五项工作列入系统的开发过程中。

1. 确定系统的质量和首要目标

在系统开发前就系统质量和工作重点以书面形式提出要求，弄清整个系统实现的主要目标。

2. 定出各种规范标准并加以实施

标准有：①系统说明书的标准；

②编程的规范；

③系统变更的控制标准；

④系统文献的制作标准；

⑤系统质量的控制标准。

上述标准是我们参考软件工程规范与用户共同协商制定的，并且在系统的开发过程中始终坚持并贯彻，确保了系统的稳定性和完整性。

3. 形成系统开发进程文献

我们将开发人员的目标、工作重点、基本设想、开发工作的指导思想，设计难点和算法等等都记录在系统的开发日志上，将容易忘记的意图记录下来。有了这些文献就会使开发和维护工作简单得多。

4. 坚持进行阶段验收和评审

我们遵循了一条软件工程学原理——用一系列生存期计划来管理软件课题的思想——来提高系统的可维护性。在系统的每个开发阶段都坚持对系统进行验收和评审，使用户能够定期检查自己的要求和愿望是否达到，开发人员也能够较早核实自己原先对系统重点、要求和说明书的理解，维护人员也能对正在开发的系统定期进行可维护性检查。

5. 让维护人员参与系统的开发进程

随着系统的规模越来越大，维护人员需要有时间来研究熟悉他们今后要维护的系统，需要有时间来做维护的准备工作。如果等到运行和维护阶段再来考虑系统的后援，那么其后援的“性能/价格”就会增加。维护人员参与系统开发的任务和作用是向开发人员提供建议、意见、工具和方法；要求开发人员提供维护后援所必需的信息；给开发人员定出可维护性标准等。这些都有助于开发人员注意改善整个系统的可维护性。

四、系统的可维护性策略

在地区级电业局MIS的开发过程中，我们采用了下列改善系统可维护性的策略，即

1. 让维护人员自始至终参加 系统的各阶段工作

在系统分析阶段,有一个突出的问题,即用户的要求难于用技术性语言来描述,而用户的这些要求所提出的质量和完整性指标都将较大地影响到今后该系统的质量和可维护性。为了确保系统的可维护性质量。在这个阶段,我们一方面深入理解用户的要求,另一方面明确维护人员参与的工作,以及系统维护范围。逐条审查系统要求,针对四个方面的问题重点进行分析。

①检验在系统开发之后对关键资源进行扩展的可行性。例如,我们是利用 MU/FO 多用户关系网络数据库系统开发上述 MIS 的。由于 MU/FO 数据库系统规定最多可分别定义 240 个不同的记录、索引、表格和报表。为此,我们需检验在系统功能基本实现后,还能剩下多少数量未定义的记录、索引、表格和报表资源,以便今后系统扩展的可行性和进行完善性维护。

②明确那些要求可能会造成系统进行收缩和扩充。例如,当用户提出某些要求后,由于限于目前管理的需要及对计算机功能的不了解,往往仅提出整个管理业务中的部分要求。随着计算机管理功能的实现和投入使用,则他们在重新理解计算机功能的基础上,必然会提出管理业务中的其它部分要求。这些要求势必造成系统的扩充,若等系统设计实施完成后再扩充,将可能造成系统出现难以估计的问题。为此,在分析用户要求时,要注意那些要求将会引起系统扩充,以便在系统分析设计开始时,就与用户协商,将这些要求包括到系统中,进行全面设计实施,以增加系统的完整性和可维护性。

同样,当用户提出某些不合实际业务管理的要求或理想化的要求,系统实现后无实用价值,最终则必然会删掉,使系统收缩。为此,在分析用户要求时,应拒绝用户的这类

要求,不予系统实现。避免不必要的系统维护工作。

③针对用户提出的系统要求,全面研究运行和维护该系统所需的后援工作。例如,用户在用电管理子系统中要求在计算、统计、分析时,要去年同期值,这样则必然造成系统中要同时存放两年的数据。这样一来势必与其它管理子系统的数据库争存贮空间,为此,要制定出运行各个管理子系统的数据库后备策略,统一安排后援工作,以便进行系统数据库维护。

④为了维护该系统的不同版本,还要确定那些性能和设备是必需的,那些可能要变更。例如,在全面分析用户要求和基于用户现有设备条件后,我们确定此 MIS 有两个版本,第一版本实现地区电业局局机关计算机辅助管理,第二个系统版本通过扩充实现地县两级电业局管理。对于前者在现有 IBM PC/AT 机上实施,先投入使用,后者由于对设备处理速度、数据通信和存贮空间提出了新的性能要求,故要更新设备,购置高档微机联网后才能投入开发使用。这种一开始就根据设备性能统筹规划系统的不同版本,避免了盲目开发,造成由于设备限制带来系统版本的维护困难。

通过上述工作给今后整个系统的可维护性打下了基础。

系统说明书是否完整和正确是关系到整个系统开发的成败。在形成说明书阶段,我们坚持和维护人员一起审查说明书,同他们一起对系统未来结构、目标功能模块和可维护性加以评价;向维护人员说明新系统需用那些模块,以及在测试中要用那些方法,听取他们的建议和要求,以求改善系统的可维护性。

MIS 设计中采用的自顶向下和自底向上的混合设计方法,模块化系统结构和程序结构化设计方法构成了设计阶段的规范⁽³⁾。为

了确保系统的可维护性，我们还增加了

①让维护人员参与设计的审查工作，以使他们较早地弄清实现设计的可行性，及对系统的影响，以及运行中可能产生的问题。

②注意使系统能灵活地扩展、压缩和变更，为了测试它的灵活性，在设计中做一些变更试验。

③与维护人员一起研究反复调用的关键系统软件和将来系统可能变更的被修改措施，充分考虑系统的应变能力。

总之，我们在设计中尽量多考虑一些系统的可维护性问题，并注意系统原先的计划日期和费用的约束条件。

测试是MIS开发的最重要阶段之一。常规的测试仅注意发现系统的设计和实施工错误，发现不能正确实现说明书的错误所在。而我们在测试过程中，让维护人员参与决定测试计划、测试数据以及形成最后的测试报告的工作。当修改系统时，让他们再用实施过的测试实例重测一下，比较结果估计系统的可维护性质量。由于难以测试的系统是难以维护的，而由维护人员主持的测试更容易发现系统错误和难维护性，并可促进他们较早地熟悉需维护的系统。

2. 系统变更的控制

显然一个系统的应变能力是其可维护性的关键，而系统变更的副作用是使系统质量下降的重要因素。有人估计，系统每变更一次可能导致系统出现一个错误的概率是20%左右。如果系统经常变更，不仅会产生大量错误，而且最终会导致整个系统的全面退化，使系统失去效用，性能/价格也不合算。为此必须对系统变更进行控制。

在MIS开发期间的变更通常有三种，即用户要求的变更；说明书的变更；程序的变更。一般讲后两种变更更容易得到控制，而第

一种变更不宜控制。为此，每当用户提出变更要求时，我们对其认真分析研究后，再作出决定。不是用户一提出要求就马上修改系统。因为即使是简单的变更，在系统使用期中也可能有错综复杂而非常严密的细节，稍不注意就会产生错误。还有不能单凭可用性有提高或性能得到改善，就决定变更是合理的。例如，有时用户常常过分夸大他们如何需要增加某种特殊功能，可一旦实现后，他们却又很少用它，而随之带来的副作用却难以尽快解决，另外，我们还注意考虑每个变更要求与其他变更要求之间的关系以及与系统生存期规划的关系，来对变更加以控制。并明确处理变更要求的规程，当变更要求拒绝时给出理由，采纳时记录下变更要求，制订系统变更实施方案，并把这些变更包括到维护任务的规划中去。

通过上述为改善系统可维护性的策略，使我们开发的地区级电业局MIS具有较好的系统可维护性。

五、结束语

目前，综合MIS的可维护性问题还没有引起人们的足够重视，笔者在两项较大型的企业综合MIS的开发实践中，参考软件工程的开发规范和一些文献资料对此作了一些探讨工作，供同行们参考。

[1] Carma L·Mcclure 《MANAGING SOFTWARE DEVELOPMENT AND MAINTENANCE》Van Nostrand Reinhold, 1981

[2] 董士海编。《计算机软件工程环境和软件工具》，科学出版社，1988

[3] 于功弟等。地区级电业局MIS的设计与开发，《计算机时代》，1991(1)

信息处理的映射和链接技术*

西南民族学院 杨宪泽

摘要 本文围绕信息处理中如何提高速度和节省存贮空间两个基本问题,介绍了很有用的映射和链接技术。文中涉及到排序、检索等计算机信息处理的基本操作,给出了一些高效方法及其研究现状和发展前景。

一、引言

在计算机信息处理工作中,涉及到两个基本问题:一是速度,包括信息的处理速度和查询速度;二是占有空间,即怎样节约存贮空间。实际上,速度和空间是一对矛盾的统一体。人们时而附加存贮空间来换取速度,时而考虑最小存贮空间而不考虑速度,视具体情况而定。为此,产生了大量有效的方法和技术,目的都是在特定场合下兼顾空间与速度。本文根据笔者近几年的研究,以信息处理中几个重要的操作为模型,既介绍如何应用映射技术和链接技术来提高速度,或者节省存贮空间,还介绍这几个重要操作的研究进展。

二、排序中的映射技术

在计算机信息处理中,花费在排序上的时间占系统CPU运行时间很大比重。有人认为在商用计算机中,批处理系统的15%—70%的CPU时间用在排序上;也有人认为排序工作占了计算机工作量的50%左右。笔者不敢肯定上述两种提法的正确性,但据我们的研究、考察资料表明,排序工作占了微机应用工作量的20%—30%。因此,为了提高计算机工作效率,研究出更有效的 $O(N)$ 的排序方法,是计算机应用科学的一个重要

方面。

比较交换排序是人们经常使用的一类排序方法。其基本思想是比较信息关键字大小而交换位置,最后排成人们需要的有序序列。这一类方法有算法容易设计,不附加存贮空间的特点。早期的比较交换排序算法时间复杂性一般是 $O(N^2)$ 。Quicksort方法把时间复杂性提高到了 $O(N\log_2 N)$ — $O(N^2)$;Heapsort方法把时间复杂性提高到 $O(N\log_2 N)$ 。已经证明,比较交换排序法时间复杂性下界是 $O(N\log_2 N)$,这限制了信息处理的速度。

排序中如果使用映射技术,时间复杂性可以提高到 $O(N)$ 。早在1956年,Isaac, E. J和Sing—leton, R. C提出了用地址计算排序,这是排序中使用映射技术的初期研究,但并未得到足够重视:其一,当时的计算机内存很小,以内存开销换取速度不可取;其二,初期的研究有很大的局限性,没有跳出反复比较和交换关键字方式的束缚。80年代后期,许多人都认真的研究映射技术应用于排序,并投入实践。其原因是

①随着微机在事务管理中应用的普及,需要处理的信息量越来越大,寻求时间复杂性是 $O(N)$ 的排序方法成为当务之急。

②微机在事务管理中的某些应用具有特殊性。如高考分数排序,竞赛名次排序,全

*国家民族委员会科研基金资助课题,编号90010。

面质量管理评估得分排序等关键字最大值和最小值在一定范围内,使用映射式排序可获得高效率,这就促进了映射式排序方法的研究和应用。而其它一些工作又借鉴这些工作的成功基点,使映射式排序方法得到扩充,发展。

③微电子技术飞速发展,微机内存容量大大增加,从而使映射式排序方法以牺牲部分存贮开销换取速度成为可能。

映射式排序方法的基本思想是:给定一组数据 D_1, D_2, \dots, D_n ,可知或可以估计这组数据中的最大值 D_{\max} ,最小值 D_{\min} ,那么,若开辟一个数组 $R(L)$ (如果 D_{\min} 为正, $L=D_{\max}$;如果 D_{\min} 为负, $L=D_{\max}-D_{\min}$),即可把数据送入数据值与 R 数组下标相等的对应元素中,该方法原理结构见图1。

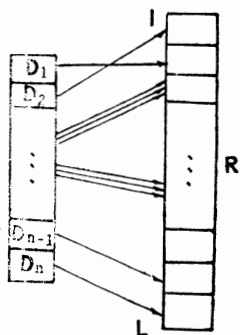


图1 映射式排序结构图

显然,若 $D_1=50$,它对应 $R(50)$; $D_2=500$,对应 $R(500)$ 。相同数据落在同一数组元素中,用计数方式可知有 n 个。由于数组元素的下标是有序的, $500>50$,数组元素的下标自然把数据一次定好了位置,最后只要将非零元素按规定的方式打印出来,相同元素按计数值打印多次,排序即可完成。

映射式排序彻底抛弃了将数据一一反复比较而交换的方法,速度很快。但提高速度是以牺牲部分存贮开销换取的。

三、检索中的映射技术

字符检索是计算机信息处理的关键技术之一。信息量越大,使用什么样的检索算法尤为重要。因此,一个检索算法的好坏,不仅决定计算机使用效率,也影响信息的获得和处理效率。

现有的检索算法可归纳为两大类:一类完全建立在比较基础上,检索依赖于信息关键字个数和所进行的比较次数,效率不高;另一类称之为散列算法,核心是构造Hash函数使信息关键字通过计算而落在预先确定的值域,要么再比较查找完成检索;要么再构造另一个Hash函数,直到冲突不发生,检索完成为止。散列检索算法实际上部分使用了映射技术,因此提高了效率。但它的缺点是:实践证明,要想在不同情况下分析关键字设计出一个好的Hash函数困难,不好的Hash函数效率不高。迄今为止,虽然人们在改进散列算法,构造新的Hash函数方面进行了不懈的努力,但上述问题依然存在。

笔者提出一个字符检索的新方法,不是部分而是始终使用映射技术,利用字符计算机内ASCII码可转换成十进制数表示的特点,使之能生成一个通用计算程序使关键字(字符)与信息记录有直接对应关系。这一方法包括预处理与检索实施两大部分。

1. 预处理

预处理建立信息关键字(字符) K_i 与信息记录的存贮位置之间的直接映射关系 F ,步骤如下:

①对每一信息关键字 $K_i(i=1, 2, \dots, N)$,切分成单一字符 $K_{i1}, K_{i2}, \dots, K_{id}(d$ 为字符串长度)。

使用命令

$LEN(K, \$)$; 求字符串长度。

$MID \$ (K, \$, j, 1)$; 切分成单一字符。其中 $j=1, 2, \dots, d$ 。

②求 $K_{i1}, K_{i2}, \dots, K_{id}$ 的ASCII码十

进制数, 最大值255。

使用命令 $ASC(K_i \text{ } \mathcal{S}(j))$ 。

③求和 $F_d = (K_{i1} + K_{i2} + \dots + K_{id}) \text{ ASCII}$

$F_{d-1} = (K_{i1} + K_{i2} + \dots + K_{id-1}) \text{ ASCII}$

\vdots

$F_1 = (K_{i1}) \text{ ASCII}$

求和的含义是: 如果 $K_i \rightarrow F_d$ 映射是单值的, 那么, 可建立 $K_i \rightarrow F_d \rightarrow$ 信息记录的映射关系。反之, 有相同ASCII码十进制之和的信息关键字。这时我们试探 $K_i \rightarrow F_{d-1}$ 的单值性……, 只要关键字不同, 一定可以在 $K_i \rightarrow F_j (j \geq 1)$ 找到这种单值关系。如果直到 $K_i \rightarrow F_1$ 都是多值的, 为相同关键字。

④让 F_d 值与附加存储空间 R_d 空间地址对应, 即 $R_d(L)$ 中 $L = F_d$, 计数器 $PL_d \leftarrow PL_d + 1$ (PL_1, PL_2, \dots, PL_d 初值赋0)。并作如下处理:

A. 若 $PL_d > 1$, 让 $R_{d-1}(L)$ 中 $L = F_{d-1}$, 且 $PL_{d-1} \leftarrow PL_{d-1} + 1$, 若 $PL_{d-1} > 1$, 让 $R_{d-2}(L)$ 中 $L = F_{d-2}, \dots$ 。如果 $PL_1 > 1$, 为相同关键字, 处理方法参阅文献。

B. 若 $PL_1 = 1 (j \geq 1)$, 把 K_i 的信息记录原始存贮地址填入 R_j 中, 至此, 预处理结束。

综上所述, 预处理可以不分析信息关键字状况, 依赖编写一个通用程序实现。

2. 检索实施

预处理完成后, 信息关键字与 R 空间形成了唯一对应关系, 为检索提供了捷径, 见图2。

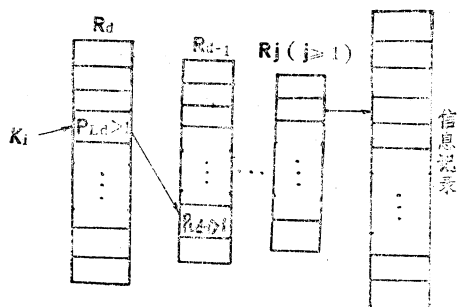


图2 检索结构图

关键字检索实际上是预处理工作逆运算, 流程见图3。

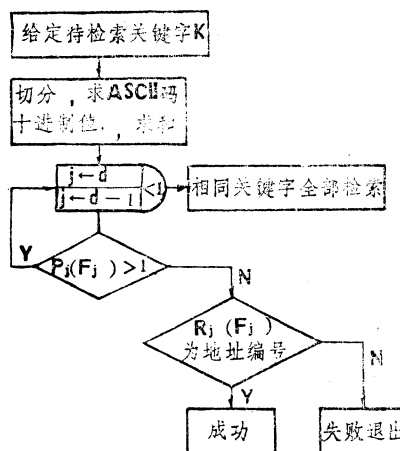


图3 检索流程图

可以证明, 此方法附加存贮开销 $128d(d+1)$ 。由于信息处理工作中 N 很大, 而检索的关键字要求越简短越好, 因此 d 一般很小, 所以附加存贮开销即使是微机也能承受。

预处理是在检索前完成的, 它应包括在信息记录处理部分, 不作为关键字检索所需时间。因此, 检索时间复杂性为 $O(1)$ 。

这一检索方法, 不象散列算法必须去分析信息关键字状况, 实践证实比构造Hash函数的检索方法效率高6—8倍。但必须认识到, 效率的提高与始终采用映射技术和牺牲部分存贮开销来换取速度有关。

四、使用链接技术构造排序算法

图1展示的排序方法, 对均匀分布的数据最有效, 许多细节还待改进, 具有相同的地址关键字怎样处理是待改进的问题之一。在多分量记录的排序算法构造中, 我们使用链接技术予以辅助, 处理相同关键字。

多分量记录排序算法的提出基于这样的背景: 近年来, 计算机处理的信息规模越来越大, 其记录分量越来越多。这类信息排序

移动记录所花时间多于关键字比较所花时间,而现有的不移动记录的排序算法时间复杂度是 $O(N^2)$ 。为此,笔者使用映射排序的基本思想,以链接技术予以辅助,构造了一个时间复杂度达到 $O(N)$ 的排序算法,基点如下:

(1) 给定 N 个待排信息记录,含有分量 $D_{1i}, D_{2i}, \dots, D_{mi}$ (m 为分量个数, $i=1, 2, \dots, N$), 确定某一分量为关键字, 记为 K_i , 求 K_{\max} 。

(2) 开辟链指针空间 R , 容量 N ; 记数空间 P , 链首指针空间 Q , 链当前指针空间 W , 容量均为 K_{\max} 。

(3) 从 $i=1$ 开始, 输入 K_i , 让 $P(K_i) \rightarrow P(K_i) + 1$; 即完成映射工作, 记录相同关键字出现个数。

(4) 若 $P(K_i) = 1$, 作 $W(K_i) \leftarrow i$ 和 $Q(K_i) \leftarrow i$, 转(6); 即据关键字映射所对应不同的单元, 构造链当前指针和链首指针。链当前指针将为(5)出现相同关键字提供链接地址, 链首指针是映射单元对应信息记录首地址。

(5) 若 $P(K_i) > 1$, 作 $R(W(K_i)) \leftarrow i$ 和 $W(K_i) \leftarrow i$; 即有相同关键字, 据当前链指针地址链接起来, 此外, 还要再记录当前链指针, 为链接出现多个相同关键字作准备。

(6) $i \leftarrow i + 1$, 直至 $i = N$ 为止, 实施(3)~(5)。

(7) ($Z = 1$), 从 $J = K_{\max}$ 开始, 若 $P(J) = 0$ 转(8); $P(J) \neq 0$, 作递减排序: (A) $T \leftarrow Q(J)$; 链首指针送 T 。(B) 输出 $D_1(T), D_2(T), \dots, D_m(T)$ 。(C) $Z \leftarrow Z + 1$, 若 $Z \neq P(T)$, $T \leftarrow R(T)$ 后转(B); 否则, 转(8)。

(8) $J \leftarrow J - 1$, 实施(7), 直至 $J = 0$ 结束。

算法流程见图4。

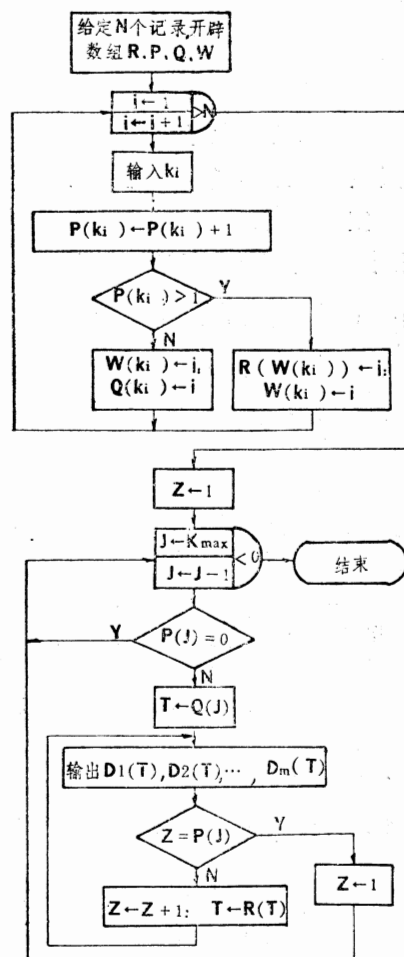


图4 多分量记录排序算法流程图

实践证明, 这一排序方法效率高, 但附加了存贮开销 $3K_{\max} + N$ (K_{\max} 为关键字最大值)。请注意, 常见的信息处理中 $K_{\max} < N$; 若有 $K_{\max} \gg N$ 的情况, 可扩充算法。

这一排序方法所以有高效率, 原因也是不实施对信息关键字反复比较和交换两种操作, 附加了一定的存贮开销换取速度, 以映射关系完成排序。此外, 如果不采用链接技术处理相同关键字, 这一排序算法难以构造。

五、库房进货管理中 使用链接技术

库房进货管理是库房管理的一个子课

题。有的库房需要计算机保存多年的每一天的进货信息。而每一天所进物品不定，多则上百件，少则没有，这给设计造成困难。最简单的方法是据历史上每天进货的最高纪录设计一天最大信息容量，但这样做将造成极大的内存空间浪费。如果设计中采用链接技术，问题可以得到圆满解决，见图5。

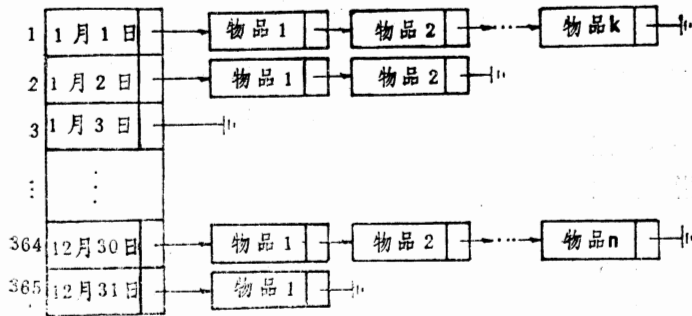


图5 库房进货管理链接结构图

图5对一年中的每天采用顺序索引，

为查询提供了方便。每一天所进物品采用链接技术，有则增加，无则为0。这不仅把线性表中链接技术插入容易的优点应用，而且在这一特定环境中，极大地节约了软件占有的内存空间。

六、结束语

笔者在信息处理研究和实践中认识到，映射和链接方法是很实用的技术。这些技术，如果不注意发掘，很难了解它们的许多有价值的特性，但若注意把它们融于不同的软件设计环境，就可以在特定场合下发挥很大的作用。笔者认为，本文介绍的方法

仅是这些技术应用的一个侧面，它们还有更广泛的用途，这需要我们不断再发掘，再探索。

(上接第30页)

该数据库进行访问了，降低了多用户系统的效果。

我们可以从软件提供的命令语句中找到解决的方法，在FOXBASE+或者dBASEⅢ中，对数据库写操作的大部份命令都带有

〔范围〕可选项。如append的完整句法是：
replace〔范围〕〈字段1〉with〈表达式1〉[，
〈字段2〉with〈表达式2〉……]
〔for/while〈表达式〉〕

句中的〔范围〕可缺省。在dBASEⅢ中，缺省表示对当前的记录进行替换操作，通常都是缺省。在FOXBASE+中如果以USE YWF exclusive的方式使用数据库，则对当前记录的替换操作也通常缺省。如果在对当前记录操作的命令中加入一个范围next1，则在dBASEⅢ中，next1和缺省是等价的；在FOXBASE+中，不是以exclusive的方式使用数据库，next1会自动锁住当前的记录，一旦操作完成，记录加锁就会自动释放，也不影响其它用户对数据库的访问。如：

```
use YWF
:
replace next 1 FY with x
:
```

这样就能使程序在两个系统中互为兼容，同时使多用户数据共享得到了保证。

五、结束语

由于多用户与单用户的环境条件而决定了FOXBASE+与dBASEⅢ不可能完全兼容。但是，在编程技巧上，巧妙地使用命令语句，使多用户FOXBASE+在对数据库写操作时能自动地加锁和解锁，那么FOXBASE+与dBASEⅢ能实现互为兼容。另外，两个软件版本也不同，FOXBASE+比dBASEⅢ的功能强，指令也丰富。因此，在不考虑多用户环境的前题下，FOXBASE+对dBASEⅢ是完全兼容的，而IBASEⅢ对FOXBASE+是不完全兼容的。总之，要使命令文件在不同的环境下运行，达到兼容性最高、修改度最低，这主要取决于使用者对系统了解的深入程度、操作方法是否正确、命令语言的熟悉程度和编程的技巧。

AR-3240打印机常见故障及维修

浙江省计算所技术开发部维修部 吴可 卢岩文

一、AR—3240打印机简介

AR—3240 打 印 机是当前较为流行的一种高点阵硬字库超高速汉字打印机，它采用了最新研制的高速高精度打印头，机内配置24点阵宋体、32点阵宋体两套硬字库，还可加配40点阵黑体、宋体和楷体三套未经压缩的高点阵精密硬字库。控制码采用了中国计算机发展公司统一制定的指令序列。整机性能较以前的2463有很大提高。

该机CPU为TMP90C841,用来控制整个打印机系统,CPU和门阵列(μ PD65006)把

输入数据编辑处理成为信号，经由驱动电路转换成可驱动打印头，字车和送纸电机的信号进行打印。

随着 AR—3240 打印机的普及使用,对机器的维修也越来越需要,根据我们维修工作中遇到的一些故障,归纳为以下几点。

二、AR—3240常见故障及维修

1. 电机驱动部分故障

电机驱动部分的故障率比较高,主要是字车驱动管损坏。该部分电路如图1所示。

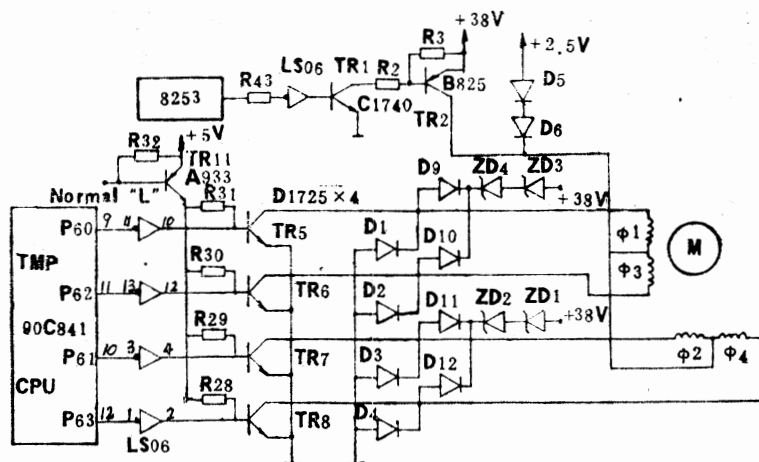


图 1 字车电机驱动电路

电机驱动部分出现故障时，一般表现为：开机后面板指示灯正常，但打印头不归位，用手拨打印头时，发现打印头很紧，不易拨动；或者打印头微动后即发出“咔咔”声，这时一般可断定为电机缺相所致。检查IC3（LS06）正常，8253也发出信号，但四

只驱动管TR5—TR8却未按正常情况顺序导通。用万用量表四只驱动管,发现全部或部分烧坏,用好的管子替换即正常。(注:若无D1725管,可用BU406代替)。

在此,要特别指出的是:电机驱动管的损坏,常常伴随着TR2(B825)也损坏,或

1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021, 2022, 2023, 2024, 2025, 2026, 2027, 2028, 2029, 2030, 2031, 2032, 2033, 2034, 2035, 2036, 2037, 2038, 2039, 2040, 2041, 2042, 2043, 2044, 2045, 2046, 2047, 2048, 2049, 2050, 2051, 2052, 2053, 2054, 2055, 2056, 2057, 2058, 2059, 2060, 2061, 2062, 2063, 2064, 2065, 2066, 2067, 2068, 2069, 2070, 2071, 2072, 2073, 2074, 2075, 2076, 2077, 2078, 2079, 2080, 2081, 2082, 2083, 2084, 2085, 2086, 2087, 2088, 2089, 2090, 2091, 2092, 2093, 2094, 2095, 2096, 2097, 2098, 2099, 2100, 2101, 2102, 2103, 2104, 2105, 2106, 2107, 2108, 2109, 2110, 2111, 2112, 2113, 2114, 2115, 2116, 2117, 2118, 2119, 2120, 2121, 2122, 2123, 2124, 2125, 2126, 2127, 2128, 2129, 2130, 2131, 2132, 2133, 2134, 2135, 2136, 2137, 2138, 2139, 2140, 2141, 2142, 2143, 2144, 2145, 2146, 2147, 2148, 2149, 2150, 2151, 2152, 2153, 2154, 2155, 2156, 2157, 2158, 2159, 2160, 2161, 2162, 2163, 2164, 2165, 2166, 2167, 2168, 2169, 2170, 2171, 2172, 2173, 2174, 2175, 2176, 2177, 2178, 2179, 2180, 2181, 2182, 2183, 2184, 2185, 2186, 2187, 2188, 2189, 2190, 2191, 2192, 2193, 2194, 2195, 2196, 2197, 2198, 2199, 2200, 2201, 2202, 2203, 2204, 2205, 2206, 2207, 2208, 2209, 2210, 2211, 2212, 2213, 2214, 2215, 2216, 2217, 2218, 2219, 2220, 2221, 2222, 2223, 2224, 2225, 2226, 2227, 2228, 2229, 2230, 2231, 2232, 2233, 2234, 2235, 2236, 2237, 2238, 2239, 2240, 2241, 2242, 2243, 2244, 2245, 2246, 2247, 2248, 2249, 2250, 2251, 2252, 2253, 2254, 2255, 2256, 2257, 2258, 2259, 2260, 2261, 2262, 2263, 2264, 2265, 2266, 2267, 2268, 2269, 2270, 2271, 2272, 2273, 2274, 2275, 2276, 2277, 2278, 2279, 2280, 2281, 2282, 2283, 2284, 2285, 2286, 2287, 2288, 2289, 2290, 2291, 2292, 2293, 2294, 2295, 2296, 2297, 2298, 2299, 2300, 2301, 2302, 2303, 2304, 2305, 2306, 2307, 2308, 2309, 2310, 2311, 2312, 2313, 2314, 2315, 2316, 2317, 2318, 2319, 2320, 2321, 2322, 2323, 2324, 2325, 2326, 2327, 2328, 2329, 2330, 2331, 2332, 2333, 2334, 2335, 2336, 2337, 2338, 2339, 2340, 2341, 2342, 2343, 2344, 2345, 2346, 2347, 2348, 2349, 2350, 2351, 2352, 2353, 2354, 2355, 2356, 2357, 2358, 2359, 2360, 2361, 2362, 2363, 2364, 2365, 2366, 2367, 2368, 2369, 2370, 2371, 2372, 2373, 2374, 2375, 2376, 2377, 2378, 2379, 2380, 2381, 2382, 2383, 2384, 2385, 2386, 2387, 2388, 2389, 2390, 2391, 2392, 2393, 2394, 2395, 2396, 2397, 2398, 2399, 2400, 2401, 2402, 2403, 2404, 2405, 2406, 2407, 2408, 2409, 2410, 2411, 2412, 2413, 2414, 2415, 2416, 2417, 2418, 2419, 2420, 2421, 2422, 2423, 2424, 2425, 2426, 2427, 2428, 2429, 2430, 2431, 2432, 2433, 2434, 2435, 2436, 2437, 2438, 2439, 2440, 2441, 2442, 2443, 2444, 2445, 2446, 2447, 2448, 2449, 2450, 2451, 2452, 2453, 2454, 2455, 2456, 2457, 2458, 2459, 2460, 2461, 2462, 2463, 2464, 2465, 2466, 2467, 2468, 2469, 2470, 2471, 2472, 2473, 2474, 2475, 2476, 2477, 2478, 2479, 2480, 2481, 2482, 2483, 2484, 2485, 2486, 2487, 2488, 2489, 2490, 2491, 2492, 2493, 2494, 2495, 2496, 2497, 2498, 2499, 2500, 2501, 2502, 2503, 2504, 2505, 2506, 2507, 2508, 2509, 2510, 2511, 2512, 2513, 2514, 2515, 2516, 2517, 2518, 2519, 2520, 2521, 2522, 2523, 2524, 2525, 2526, 2527, 2528, 2529, 2530, 2531, 2532, 2533, 2534, 2535, 2536, 2537, 2538, 2539, 2540, 2541, 2542, 2543, 2544, 2545, 2546, 2547, 2548, 2549, 2550, 2551, 2552, 2553, 2554, 2555, 2556, 2557, 2558, 2559, 2560, 2561, 2562, 2563, 2564, 2565, 2566, 2567, 2568, 2569, 2570, 2571, 2572, 2573, 2574, 2575, 2576, 2577, 2578, 2579, 2580, 2581, 2582, 2583, 2584, 2585, 2586, 2587, 2588, 2589, 2590, 2591, 2592, 2593, 2594, 2595, 2596, 2597, 2598, 2599, 2600, 2601, 2602, 2603, 2604, 2605, 2606, 2607, 2608, 2609, 2610, 2611, 2612, 2613, 2614, 2615, 2616, 2617, 2618, 2619, 2620, 2621, 2622, 2623, 2624, 2625, 2626, 2627, 2628, 2629, 2630, 2631, 2632, 2633, 2634, 2635, 2636, 2637, 2638, 2639, 2640, 2641, 2642, 2643, 2644, 2645, 2646, 2647, 2648, 2649, 2650, 2651, 2652, 2653, 2654, 2655, 2656, 2657, 2658, 2659, 2660, 2661, 2662, 2663, 2664, 2665, 2666, 2667, 2668, 2669, 2670, 2671, 2672, 2673, 2674, 2675, 2676, 2677, 2678, 2679, 26

这类故障往往比较难发现。它的特点是后打印头正常归位,但不能联机打印,认为是联机接口门阵列损坏。这可以用简便的方法来区别,即打印机进行自检。若自检正常,一般为接口片损坏;若为保护电路损坏,则自检时,在打印头归位,会有一声长鸣,且不打印,该部分电

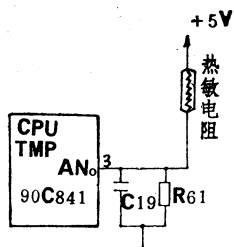


图2 打印头温度保护电路

3. 复位电路故障

接通电源后，单稳IC7（M51953）应输出一个约34ms的RESET信号，（ $T = 0.34 \times C_{32} \text{ (PF)} \mu\text{s}$ ），该信号使CPU复位，从而使TR11截止，屏蔽所有发予字车电机、送纸电机和打印头的驱动信号。检查这类故障只要在上电时用示波器观察IC7的输出端的波形，或直接检查CPU的RESET端波形。另外也可用示波器观测IC13（LS11）的第三只脚的波形，正常情况下该脚在开机时会有一负脉冲，然后为高电平。（待续）

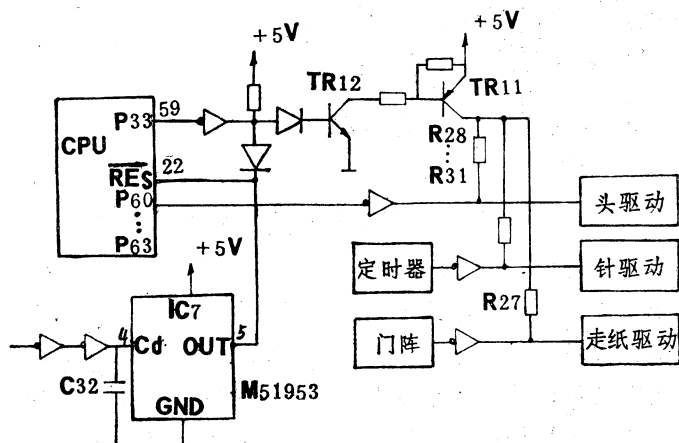


图 3 复位电路

长城0520—CH微机015板维修二例

湖南省电子研究所 胡 红

故障 1. 开机即死机，用插拔法确认为 015 板有问题。

分析与维修 015 板是长城 0520—CH 微机采用的中分彩卡，类似于 PC/XT 的彩卡，但具体线路不同，所以两者互不兼容。不过它们的维修思维方法是一样的。本故障显然是彩卡与主板之间的介面信号有问题。拔出 015 板，用透明胶纸将其地址线插脚贴住，然后插回，再开机，仍然死机，说明地址线没问题。再用透明胶纸贴住其数据线引脚，开机，由机箱前面的显示窗可以看出，已经不死机了。显然只要检查 U_{65} 和 U_{66} 器件即可，见图 1。静态测量发现 U_{65} 的第 5

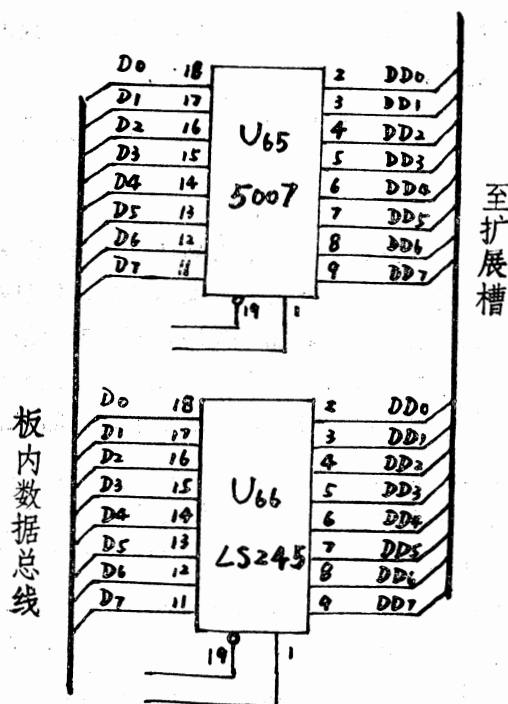


图 1

脚对地电阻只有 $30\ \Omega$ ，换掉 U_{65} ，一切正常。如果数据线也没问题，那么就要重点检查 A_{10} 脚，($\overline{I/O\ READY}$)，很可能是它恒低而造成主机死锁，继续向前查是不难找到故障原因的，见图 2。

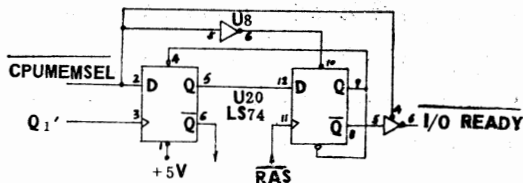


图 2

故障 2 上电，屏幕全黑，但不死机。

分析与维修 在 014 板正常，将开关打到 015 板显示，便无任何信息显示，显然问题出在 015 板。这种故障就要通过测量 U_{41} (6845) 的三个输出信号——行同步 (39 脚)、帧同步 (40 脚)、视频允许 (18 脚) 来进行大致的故障定位。如果三者都正常，则说明 U_{41} 初始化正确，在上电自检过程中，只有在 VRAM 读写正确的前提下，才对 U_{41} 进行初始化，因此在这种情况下，VRAM 的数据通路、CPU 访问的地址及其同步时序信号是正常的，状态寄存器也是正常的，主要是字符/属性通路中断传输、显示时序电路损坏或者方式选择寄存器等故障引起的，依照电路图逐级检查即可。而本例是三个信号都不对，也就是说 U_{41} 没有初始化，因此显然是 VRAM 或 6845 部份有问题，当然还有时序控制。还是先检查扩展槽中的 A_{10} 脚，即 $\overline{I/O\ READY}$ 来大致判断 CPU 与 015 板的时序配合，上电用示

条形码识别技术及其应用

杭州商学院计算中心 史红霞

摘要 本文着重介绍条形码的原理, 条形码标签的制作方法和识别技术, 并通过几个实际应用的例子, 以总结出它的应用规律。

一、引言

随着计算机应用的不断普及, 条形码作为人与计算机联系的一种中间代码而相应发展起来, 并成为一项新的产品编码技术受到各界重视。条形码技术的发展为计算机应用系统提供了一种行之有效的数据录入手段。

二、条形码简介

条形码是一种唯一可以印刷的、由一系列黑、白垂直条纹符号序列组成的机器语言, 如图①所示。

CODABAR CODE



* 23456 *

图 1

条形码是一种信息存储形式。条形码中

各条条纹及其中间的间隔或条纹的相应宽度包含着存储的信息。

条形码是一种网络技术, 通过识别条形码标签, 利用计算机把分布在各个地方的物品联系起来, 成为比较便宜的通讯网络。

典型条形码编码结构如图②, 从条形码编码结构可知, 起始边界和结束边界保证了条形码阅读的完整性, 起始位和停止位不对称, 使阅读器能正, 反向扫描条形码, 这样为阅读输入提供了方便。编码区是条形码的核心部分, 该区内包含了所要传送的信息。检查和具有自动纠错的功能, 以提高阅读的准确性。

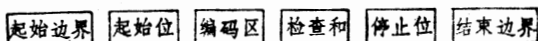


图 2

条形码具有适合于各种场合使用的码制。各种码制的字符集、编码方法、信息密度等都各不相同。如UPC码, 它主要应用于销售行业, 2/5—CODE码是一种国际代码, 读出率比较可靠, 广泛用于货运部门包

波器测量, 发现在读盘前恒高, 不断热启动总是如此。我们知道, VRAM的刷新由字符时钟CCLK同步, 而CPU存取VRAM是由CPU I/O CLOCK同步, 因此CPU直接采用MEMW命令信号会造成和CRTC刷新VRAM竞争, 所以必须让CPU在读/写总线周期插入等待状态, 让时序电路在一定时间

换成同步的 \overline{WE} 信号才可存取VRAM。现在在自检中, 015板的 $\overline{I/O\ READY}$ 恒高, 就使CPU认为VRAM总有问题而不对6845进行初始化, 造成无显示。经查发现 U_2 。(LS74)的第9脚浮空, 造成 $\overline{I/O\ READY}$ 恒高, 见图2。换之, 故障排除。

装箱标签, CODABAR码为一种可变长码制, 常用于银行、血库, 另外, 还有CODE—39码、I—2/5—CODE码等。

三、条形码标签打印技术

在国外条形码标签由制造商直接印在商品的包装上面, 使用者不必自行制作。但就我国目前情况而言, 一般来说条形码标签都由使用者制作, 这样势必带来一个用什么样设备制作条形码标签的问题, 归纳起来主要可分为以下4类。

(1) 用激光打印机打印条形码标签, 它能在一般的打印纸及涂有不干胶的打印纸上打印出高分辨率的条形码标签, 但由于其打印成本高及打印码制的限定, 目前只有在要求高的场合应用。

(2) 用普通24针点阵式打印机(如M1724、3070等)打印条形码标签, 其最大的优点是打印成本低, 操作方便, 适合于大批量条形码标签的制作, 如图书馆, 仓库等地方, 其不足之处只能打印出低、中分辨率的条形码标签。

(3) 用热敏打印机打印条形码标签, 这种打印机打印出来的条形码标签质量较好, 但它所用的热敏打印纸价格贵, 故不宜大批量生产。

(4) 采用照相制版技术制作条形码标签, 但制作过程十分繁琐, 制作价格昂贵, 只有在要求很高的情况下使用此方法。

四、条形码标签识别技术

条形码是光学形式代码, 故不能用简单的计数方法来识别和译码, 必须借助于条形码阅读器和译码器才能对其进行识别。图③是条形码阅读器, 译码器与计算机的联接图, 由图可见, 条形码标签通过阅读器扫

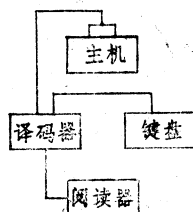


图3

描, 把信息传送到译码器, 然后再由译码器将信息转换成计算机能接受的符号, 以完成数据输入工作。(实际应用时一般都把阅读器和译码器集成在一起, 统称阅读器, 在此为了说明方便起见将其分成两个器件)。

阅读器就其类型主要可分为移动式与固定式两大类, 移动式阅读器是通过移动阅读器达到扫描的目的, 适用于比较笨重, 不易挪动而需现场扫描的物品。固定式阅读器是通过移动条形码标签达到扫描的目的, 适用于扫描借书证、卡片等轻巧物品。

条形码阅读器的原理是: 光源发射出一束光束, 当光束扫描条形码时, 宽窄不一的黑白条纹产生不同强度、不同宽度的反射光, 传感器接收到反射光后, 产生一系列相应的电脉冲, 这些不同幅度, 不同宽度的电脉冲, 经过处理转换成为数字信号, 经校验后, 传送到计算机系统。阅读器扫描使用的光源可分为激光和其他光源二种, 若采用激光光源, 激光束只要对准要输入的条形码, 让其自动扫描, 即可正确读入。采用其他光源的阅读器, 输出光束的光源部分和接收反射光的传感器都安装在一起, 象支钢笔, 简称光笔, 操作者手握光笔笔杆, 将笔头轻轻划过条形码, 即可读入。

五、条形码技术的应用

实现“人机”的方式很多, 除了最常用的键盘输入以外, 有关人士曾通过OCR(光学字符识别系统)、MS(磁条卡片识别系

统)等方法完成数据的录入工作,但这些方式与条形码比较,有其局限性,如效率低、错误率高,有的效果虽然较好但造价高、制作不便、读出机构复杂。条形码系统集廉价、简便、高效、出错率低、容易识别等特点于一体,它的独到之处引起了许多领域的反响,下面就目前应用趋势作一介绍,从中研究出条形码的应用规律。

1. 图书流通管理

众所周知,在知识更新如此迅速的今天,图书馆是各位有识之士常往的地方,这样必定产生图书流通量与日俱增和图书流通管理人员应接不暇的矛盾,解决这个矛盾的有效办法是把条形码技术应用到图书流通管理系统中去。在这个条形码管理系统中,只要对所有图书、借书卡都用条形码编码分类,读者借书时,用光笔阅读器在借书卡和所借书籍的条形码上各扫描一下,即完成了借阅登记的全过程。此系统在图书流通领域的应用,可大大提高图书流通的效率,改善流通管理的服务质量,用及时掌握图书流通率,为书刊的采购,增订提供一定的依据。

2. 仓库管理

由于仓库的物品进出频繁,有些货物十分笨重,用传统的手工管理有诸多不便。采用条形码管理后,只要将有关物品的参数制成条形码形式,每每进出物品时,扫描该条形码标签,商品的一系列数据就输入了计算机,然后再由计算机对这些数据进行处理,以实现由计算机对仓库物品的系统管理。有些体积庞大、笨重的物品可用便携式阅读器对其现场扫描,以减少不必要的功耗,提高

经济效益。

3. 车辆自动识别管理

在每辆车的固定位置粘上能表示各车辆特点的条形码,然后在车辆进出口安置激光扫描阅读器。车辆进出时,阅读器对条形码进行扫描,并把此扫描获得的信息传送到计算机管理系统,由计算机对全局车辆进行科学的调度、分配,从而使车辆的调度最合理,利用率最高。

4. 超级市场及商业零售点

UPC代码是超级市场及商业零售点应用的通用码制,由于商品价格和商品制造商等信息都用条形码表示,使计算机的零售管理具体化了,顾客只要把欲购买商品的条形码标签在阅读器上划一下,该系统就能自动对每位顾客进行现金结算,动作迅速、准确。

以上点滴仅是条形码在实际应用中的一个侧面,它还有待于我们进一步开发、研究。可以预计条形码技术在我国的应用将有广阔的前景,不久将会有更多、更优秀的条形码系统投入实际运行。

参 考 文 献

- [1] Harry E. Burke 《HANDBOOK OF BAR CODING SYSTEM》
- [2] D. Saviv and G. J. Lawrer 《The Characteristics and decodability of the universal product code symbol》
- [3] D. R. Percival 《EMERGING IMPACT OF AUTOMATIC IDENTIFICATION SYSTEMS》
- [4] 史树民. 光笔条形码技术及其在信息管理系统中的应用, 《现代图书情报技术》, 1987(1)