

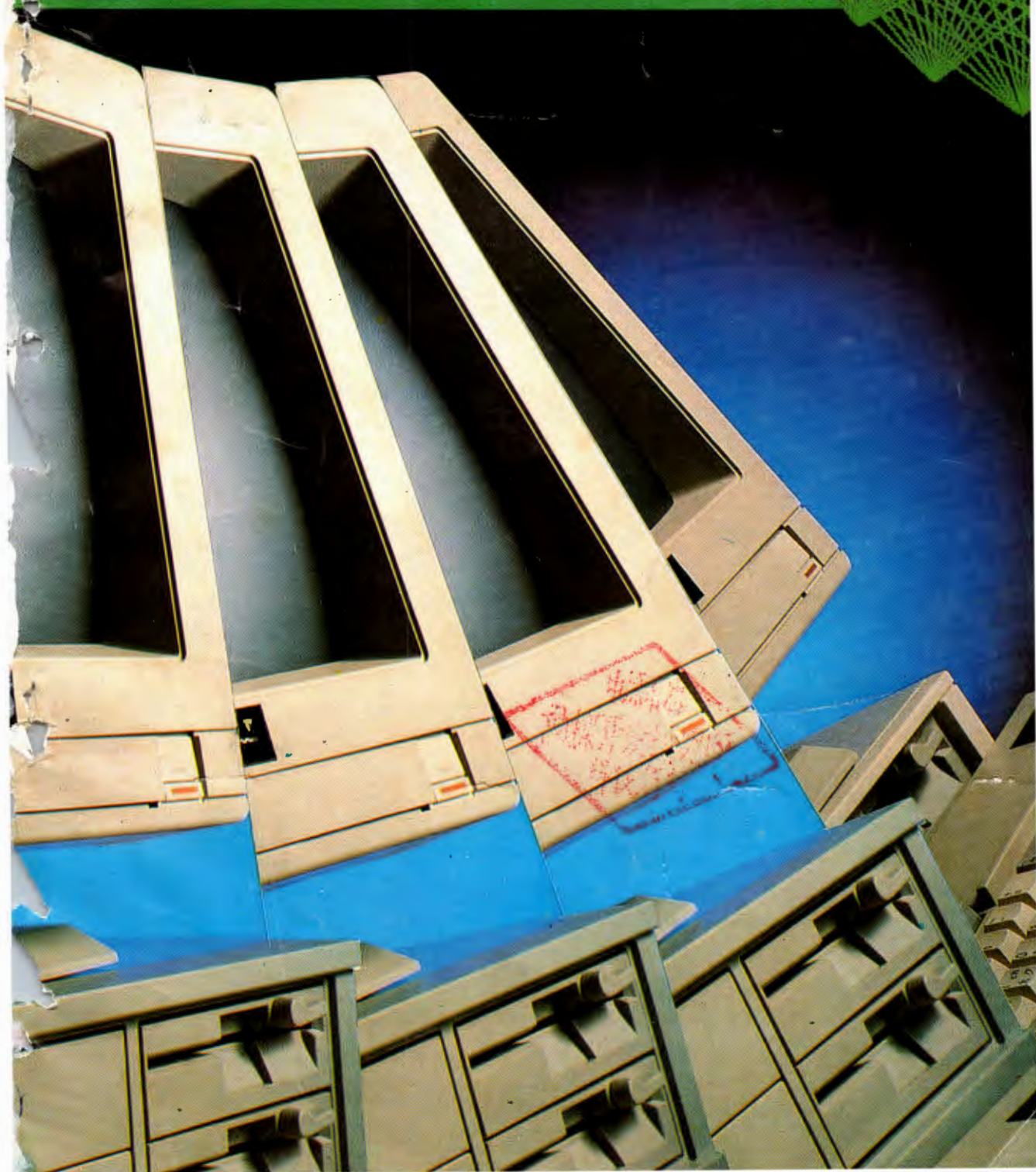
ISSN 1001-3695

計算機應用研究

1992

6

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



新兴通信——开拓新通信领域



XINXING COMMUNICATION—OPENING UP
NEW COMMUNICATION FIELDS

二哥大——大哥大的功能,对讲机的价格!
二哥大——个人通信,单位通信最优工具!
二哥大——高科技的标志!



FYA-903 手持机



YW-92 有无线转接自动控制台



C-904R 无中心选址自动中继台



NC-288B 热断自锁式
急速充电器
标准充电器



GC-904 无中心车载电台

(内容详见内文末页)

航空航天工业部国营新兴仪器厂

联系:国营新兴仪器厂通信测控系统工程处

厂址:四川成都厂北路1号

办公地址:成都一环路东一段138号

电话:331012-253,669

电传:60123 XIF CN

传真:(028)336298

电挂:7190

邮编:610055

厂休:星期六

武侯计算机网络工程公司开展微机系列合同保修业务

本公司前称西部电脑维修公司,拥有雄厚的软硬件技术力量,现有硬件维护人员 21 人,专门从事硬件维修多年,具有丰富的实际经验,在很多用户的要求和科委的支持下现特开展计算机保修业务,具体办法如下:

一、保全维修办法

- 1、由我公司维修技术人员上门办理保修手续。
- 2、单位参保的机器,由我公司技术人员检查、加封并双方认可后,办理保全手续。
- 3、凡参加保全维修的机器及设备,采用一次性缴费的办法(参见收费标准项)
- 4、我公司开展一年、二年两种保全维修业务,并长期开展续保业务。

二、保全维修方式

- 1、对参加保全维修的客户,我公司将提供优质服务,定期、分段对客户的机器设备进行保养,并提供技术咨询和完善使用功能。
- 2、成都市内上门服务,凡是送到我公司的机器,主机二天内交货,外设三天内交货,特殊情况(如停电、特殊器件等)例外。
- 3、参加保修的机器如因自然灾害或人为损坏。视情况处理。

三、保全维修收费标准(一年):(两年加倍减 20%)(含材料费)

PC 机及兼容机	200 元	打印机(打印头不宜加封,维修费另加)	UPS 电源(后备式)
PC/XT 及兼容机	300 元	低档 80 列打印机等	500W 100 元
IBM 286 及兼容机	350 元	中档 2024、3070、1600K 等	1000W 200 元
386SX-16,20	400 元	高档 3240 彩打等	2000W 400 元
386DX-25,33	500 元	数字化仪和绘图仪:	3000W 500 元
486,ISA	600 元	低档 200 元	在线式相应增加 80%
EISA	800 元	中档 300 元	
长城系列机相应地增加 100 元		高档 500 元	
AST 系列相应地增加 150 元			

注:主机配置不同,价格略有差异。未定机种参考以上价格收费。量大从优,凡是保修期内从未修过的机器,再免费保修半年。

本公司销售:各档次微机,所售机器均保修两年,终身维修。

对外承接如下工程:

- NOVELL 网的安装,调试
- NOVELL 网与 XENIX(UNIX)互联
- NOVELL 网与异种机互联(SUN HP SGI DEC 等各类工作站,各种大型计算机如 IBM)
- NOVELL 网与 NFS 网的互联
- 式数据库 ORACLE 安装,调试
- 发零售各种网络板及零配件

开户行:工商行成都跳伞塔分理处

帐号:67012804 邮编:610013

户名:成都武侯计算机网络工程公司

联系人:郭亮、代平武、况均、熊朝辉

地址:成都一环路南二段 11 号

电话:(028)-581252、580290(夜)

《计算机应用研究》杂志办刊单位

四川省电子计算机应用研究中心	新疆电子计算中心
贵州省科委计算中心	甘肃省计算中心
安徽省计算中心	广西计算中心
吉林省计算中心	山东省计算中心
河南省计算中心	青海省测试计算中心
云南省电子计算中心	四川省电子学会

《计算机应用研究》杂志社董事会

董事长:周赛渝

董 事:李天健 孙传江 陆慰椿 冯德成
乔中南 郑国基 叶大卫 闫长荣
黎 蓉 黎瑰常

《计算机应用研究》杂志编辑委员会

主任委员:张执谦

副主任委员:李泽民

委 员:贾洪钧 曾光初 龚宇清 张国栋
罗海鹏 刘铁军 崔振远 李文华
杨剑波 刘启茂 张湘金 余 凯

1992 年第 6 期(总第 50 期) 出版日期:1992 年 11 月 责任编辑:张 钢

计算机应用研究(双月刊) (公开发行)

刊 号:CN51—1196/TP
ISSN 1001—3695

邮发代号:国内 62—68
国外代号:BM4408

主 编:张执谦

副主编:李泽民

编辑出版:《计算机应用研究》杂志社

通讯地址:成都市人民南路 4 段 11 号
附 1 号

邮 政 编 码:610041

印 刷:新都一中印刷厂

订 阅 处:全国各地邮局

国内总发行:成都市邮政局

国外总发行:中国国际图书贸易总公司

广告经营许可证:川蓉工商广字 005 号

每册定价:1.80 元

《计算机应用研究》第9卷 第6期(总第50期)

目 录

软件篇

- 一种求任意切面轮廓算法 王学崇 李 剑 陈 皓(1)
- 正规文法生成算法及其实现 金之滨 仲肇铭 卢雷(5)
- 计算机递归分组排序算法 周建钦 赵志远(7)
- Occam2 并行处理程序在 Transputer 网络中的系统配置 马永强(9)
- 微机外部设备的虚拟接口方法 李桂青 赵占朝(12)
- 一种动态门限图像二值化技术 白 焱(14)
- Auto CAD 中汉字处理软件开发 尹文生 张志坚(15)
- 数据库管理系统文字型数据的类 HASH 编码技术与准顺序式紧缩存储结构
..... 张育民(18)
- 使用 LISP 编程空间不够的问题之解决方法 熊瑞萍 黄永宁(22)
- 一种窗口光条选择录入汉字域内容的方法 滕 锋(23)
- 一组巧妙的 DOS 实用例程 黄庆程(26)
- 微型机在医院病历资料管理中的应用 向瑞红 唐 承(29)
- 一个简单实用的 FoxBASE 函数计算器程序 陈来军(8)
- 西文 Turbo C 2.0 显示汉字的最简单办法 赵炳跃(17)
- 引导扇区型病毒通用清除软件—KV—BOOT 肖俊良(28)

系统篇

- IBM-PC/XT 与 MCS-96 系列单片机并行通讯系统的设计 ... 王杉杉 胡继胜 马修真(31)
- 点阵式液晶显示器与单片机的接口 赵广涛 盛翊智 吕汉兴(34)
- 对 IBM AS/400 系统没有给出 MIPS 值的几点看法 李春贵 刘铁军 马俊田(37)

硬件篇

- 微机过程控制系统中的接口与抗干扰问题 李均宜(42)

维修篇

- LQ-1000 打印机常见故障及维修 李红亮(46)
- TH3070SL 打印机故障排除一例 陈允西(47)
- M1724 打印机换行时打印头不能回到左端起始位置的故障维修 陈永红 穆大明(48)
- 电网强干扰损坏微机一例 陈德明(48)

信息篇

- RISC 结构及 SPARC 工作站 刘庆枏(49)
- 集中分散型综合控制系统(DCS)刍议 崔振远(51)
- 计算机辅助设计及其未来 张鸿鸣(54)
- 《计算机应用研究》一九九二年(第9卷)总目录 (57)
- 900MHz 无中心选址移动通信系统简介 (62)
- 书讯、启事、简讯 (25、33、41、45、50)

APPLICATION RESEARCH OF COMPUTERS

Vol. 9 No. 6 (Total 50)

CONTENTS

SOFTWARE

An Algorithm for Calculating Skeleton of Arbitrary Section	Wang Xuechong and et al(1)
Producing The Algorithm for Regular Grammar and It's Realize	Jin Zhibin and et al(5)
The Algorithm for Recursion and Grouping Sort on Computer	Zhou Jianqin and et al(7)
The System Configuration of Occam2 Parallel Processing Program on Transputer Netware	Ma Yongqiang(9)
The Virtual Interface Method with External Device of Microcomputer	Li Guiqing and et al(12)
A Dynamic Thershold Technique for Image Segmentation	Bai Yan(14)
Development Software for Processing Chinese Characters of AutoCAD	Yin Wensheng and et al(15)
Character Data Storage with Similar HASH Coding Technique and Quasi—sequential Condensed Storage Structure in Date Base Management System	Zhang Yumin(18)
The Method for Solving not enough Space by LISP Programming	Xiong Ruiping and et al(22)
An Method for Selecting Input Chinese Character with Aperture Optical—Bar	Teng Fong(23)
A Group Practical Example Program of DOS	Huang Qingcheng(26)
The Application for Management Medical—record in Hospital by Microcomputer	Xiang Ruihong and et al(29)
A Sample Program of Function Calculater for FoxBASE	Cheng Laijun(8)
A Sample Method for Displaying Chinese Character in Turbo C2. 0	Zhao Bingyao(17)
KV—BOOT— —The Software for Erase Virus of Bootstrap Sector	Xiao Junliang(28)

SYSTEM

Design with Parallel Communication System between IBM—PC/XT and MCS—96's Single Chip Processor	Wang Shanshan and et al(31)
The Interface of —liquid Crystal Display and Single Chip Processor	Zhao Guangtao and et al(34)
The Views for not Rendering MIPS Value of IBM AS/400 System	Li Chungui and et al(37)

HARDWARE

The Interface and Noise Immunity of Processing Control System by Microcomputer	Li Junyi(42)
--	--------------

MAINTENANCE

The Fault of The LQ—1000 Printer and It's Maintenanaced	Li Hongliang(46)
An Example for Debugging The TH3070SL Printer	Chen Yunxi(47)
Debugging for The Printer—head not coming Left of New—line on The M1724 Printer	Chen Yonghong and et al(48)
An Example of Power Circuit Interference Damage Microcomputer	Chen Deming(48)

INFORMATION

The RISC Structure and SPARC Workstation	Liu Qingping(49)
Brief Introduction about Synthetical Processing System with Centrattized and Dispersed	Cui Zhengyuan(51)
The Computer Auxilliary Design and It's future	Zhang Hongming(54)
Comprehensive Table of Contents of "Application Research of Computers" in 1992	(57)
The News	(25, 33, 41, 45, 50, 62)

一种求任意切面轮廓算法

王学崇 李 剑 陈 皓

四川大学计算机科学系 (成都, 610064)

摘 要 切面作为人们从任意角度观察人体组织位置的切口方法具有重要意义。而求任意切面轮廓既是求切面的基础, 又是为三维显示提供重构数据的基本方法之一。本文给出了一种求任意切面轮廓算法。

关键词 算法 三维 轮廓 切面 交点 切片

一、求任意切面轮廓算法的提出 重构三维立体象时, 必须先求出切面上的轮廓线^[1]。三维连续图象 $f(x, y, z)$ 在任意平面 $P: ax + by + cz + d = 0$ 上的 $f(x, y, z)$ 的分布, 称沿平面 P 的 f 的切面^[2]。而连续切片或多层 CT 断层象的轮廓线可在三维坐标系中表示为:

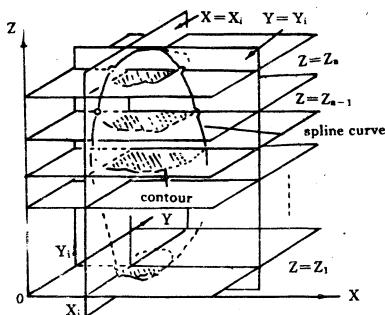


图1 轮廓线在三维坐标系中的表示

的各文件间的关系。然后, 再对有关参数进行定义。

二、求绕原轮廓线最大矩形框 由切片或断层扫描图象提取的轮廓线, 如果不考虑其 Z 坐标, 那么它是二维坐标系中第一象限里的闭合曲线。对于 PCVISIONplus 图象处理板而言, 其坐标范围为: $0 \leq X \leq 511, 0 \leq Y \leq 511$ 。而实际上每个轮廓线只占第一象限中的一部分。显然, 使用 $0 \leq X \leq 511, 0 \leq Y \leq 511$ 矩形框, 也能够表示轮廓线所在的范围。但是为了减少计算量, 只要求出所有原轮廓线中的最大的和最小的 X 及 Y , 即 $X_{\min}, X_{\max}, Y_{\min}, Y_{\max}$, 并定义绕原轮廓线最大矩形框为:

$$X_{\min} - 2 \leq X \leq X_{\max} + 2,$$

$$Y_{\min} - 2 \leq Y \leq Y_{\max} + 2;$$

这样, 在最大矩形框中, 就能够满足算法的要求。

这里, 求出轮廓线中最大的和最小的 X 及 Y 的算法如下:

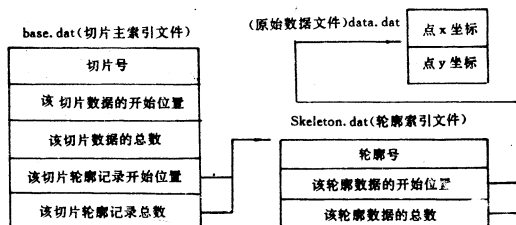
1、置切片或断层扫描图象号 $scile_no = 1$, 输入轮廓号 $skeleton_no$, $X_{\min} = Y_{\min} = 512, X_{\max} = Y_{\max} = 0$;

2、读切片或断层扫描图象的主索引文件 $base.dat$ 中切片或断层扫描图象号为 $scile_no$ 的记录 $base_record(scile_no)$;

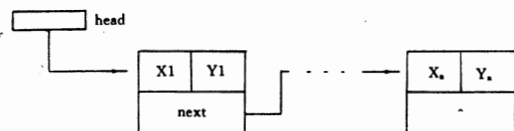
3、根据 $base_record(scile_no)$ 读切片或断层扫描图象的轮廓索引文件, 即 $skeleton.dat$ 中的记录 $skeleton_record$;

4、根据记录 $skeleton_record$ (轮廓号 $skeleton_no$) 读切片或断层扫描图象的原始数据文件 $data.dat$, 将轮廓上的离散点传送到数组 $Ax[], Ay[]$ 中;

5、使用三次 B 样条函数对数组 $Ax[], Ay[]$ 中离散点进行插值, 并形成数据结构为单链表形式的连续点的闭合轮廓线。其中: 头指针是 $head$, 格式如下:



切面作为人们从任意角度观察人体组织位置的切口方法具有重要意义。而求任意方向轮廓线既是求切面的基础, 又是为三维显示提供重构数据的基本方法之一。为了求任意切面的轮廓线(或切面与原轮廓线的交点), 首先由上图说明存储原轮廓线数据



从 head 起,把 X_i 与 X_{min} 及 X_{max} 比较,如果 $X_i < X_{min}$,则 $X_{min} = X_i$;否则判断 $X_i > X_{max}$,如果为真,则 $X_{max} = X_i$ 。同样 Y_i 与 Y_{min} 和 Y_{max} 也做上述工作,直到单链表尾;

6、切片或断层扫描图象号 $scale_no$ 加 1,如果该记录存在,则返回 2;

7、 $X_{min} = X_{min} - 2$, $X_{max} = X_{max} + 2$, $Y_{min} = Y_{min} - 2$, $Y_{max} = Y_{max} + 2$;算法结束。

三、在光栅扫描显示器上交点定义 由上节可知,所有原轮廓线都在其最大矩形框内。那么,直线在最大矩形框内的部分,才有可能与原轮廓线有交点。这里求交点是在离散平面中进行的,而很多对于模拟图象有定义的几何概念在离散图象(像素的集合)中是没有定义的^[4]。

3	2	1
4	P	0
5	6	7

图 2 为了对一个像素 P 定义其近邻像素的相对位置所用的符号

定义 1: 如果两个像素的相应单元共有一条边,那么称这两个像素为直接近邻(缩写为 d-近邻)。如果这些单元仅在一个角上相接触,那么称它们为非直接近邻(缩写为 i-近邻)。名词近邻表示这两种类型。 $0 \leq N \leq 7$ 的 N-近邻这一术语用于表示其位置是图 2 中以 N 标出的像素。可以看出, d-近邻是偶数 N 的 N 近邻,而 i-近邻对应着奇数 N。

定义 2: i-通路(或简称通路)是一个像素序列 A_1, A_2, \dots, A_n ,使得对 $K > 1$, A_{K-1} 是 A_K 的一个近邻。术语 d-通路指的是一个类似序列,但不是简单的近邻,而是要求这些像素为 d-近邻。简单通路是这样一种通路,在该通路中所有像素都不相同,而且每个像素都没有多于两个 d-近邻的通路。封闭通路是其中第一和最后一个像素重合的通路。

在光栅扫描显示器上,直线段与轮廓线相交的各种情况,如图 3 所示。

其中: X_1 ——直线段的起点; X_2 ——直线段的终点; ●——轮廓线;

S_x ——X 的增量; S_y ——Y 的增量。

在原轮廓线中,每条轮廓线都是封闭通路。这样,如果直线段通过轮廓线,那么,至少有一个交点。

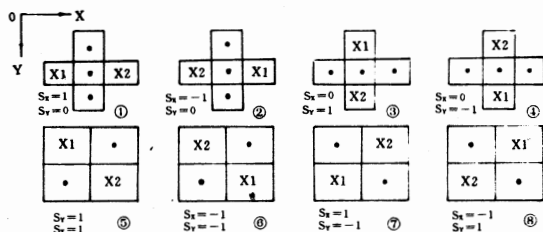


图 3 在光栅扫描显示器上直线段与轮廓线相交的各种情况

但是,曲线或直线(没有宽度)和平面区域(有一定宽度)之间直觉的几何上的区别在离散栅格上变得模糊不清。在这样的栅格上给定一像素集合,该集合是由一条细曲线的数字化所产生,还是由一完整的区域所产生,并不总是很明显的。由于像素点的连通性(这里指 8 连通),虽然直线段与轮廓线相交,但有时也存在无交点的情况。

通过分析以上八种情况,我们发现 5、6、7、8 情况,即: X, Y 的增量的绝对值都为 1,而且轮廓线如 5、6、7、8 情况走向时,由于像素点间的 8 连通性,显然,直线段与轮廓线在光栅扫描显示器上无交点。为此,在上述 5、6、7、8 情况时,定义直线段的始点 X_1 为直线段与轮廓线相交的交点。

四、求交点算法 正如大家所知道的, Bresenham 算法其构思巧妙,虽然它最初是为数字绘图仪而设计的,但它同样适用于 CRT 光栅显示器。算法的目标选择表示直线的最佳光栅位置。为此,算法根据直线的斜率确定或者选择变量在 X 或 Y 方向上每次递增一个单位。另一变量的增量为 0 或 1,它取决于实际直线与最近网格点位置的距离。这一距离称为误差。整数 Bresenham 算法采用整数算术运算和避免除法来加快算法的速度。使得每次只要检查误差项的符号即可^[4]。求交点算法是在 Bresenham 算法的基础上实现的。

将 Bresenham 算法中的写点语句改为检查该点的灰度是 0 还是 1, (0 表示背景, 1 表示轮廓线)。显然,当点 (X, Y) 的灰度 GREY 为 1 时,该点是交点。但是正如 §3 所分析的,只做上述检查是不够的。当出现无交点的情况时,就必须由新定义的交点作为所求交点。因此要对整数 Bresenham 算法做进一步的改进。首先,计算当前点 (X_1, Y_1) 的下一点坐标 (X, Y) ,并计算出它与前点在 X 和 Y 方向上的差 S_x, S_y ;然后,判断点 (X_1, Y_1) 的灰度值,当其为 1 时, (X_1, Y_1) 是交点;否则,判断 S_x 和 S_y 绝对值是否同

时为1?如是,则判断点 $(X_1, Y_1 + S_y)$ 和 $(X_1 + S_x, Y_1)$ 的灰度是否同时为1?如为1,则点 (X_1, Y_1) 是交点:这样得出求交点算法,可描述如下:

1、计算当前点 (X_1, Y_1) 的下一点坐标 (X, Y) ,并计算出它与前点在X和Y方向上的差 S_x, S_y ;

2、检查当前点 (X_1, Y_1) 的灰度是0还是1?是1转4;

3、检查 S_x, S_y 绝对值是否同时为1?如是,则判断点 $(X_1, Y_1 + S_y)$ 和 $(X_1 + S_x, Y_1)$ 的灰度是否同时为1?

如为1,则点 (X_1, Y_1) 是交点,转4;否则,转5;

4、保存交点;

5、 $X_1 = X, Y_1 = Y$;结束否? 否转1;

6、算法结束。

五、求任意切面轮廓算法 1、算法在二维坐标系中的结束条件 这里,我们首先讨论从B点切到D点的情况,假设切线(直线1)间为距离 d 。由图4可知,通过B点第一条切线的直线方程为:

$$y = \tan \alpha \cdot x + D_0 \quad (1)$$

其中, $D_0 = Y_{\max} - \tan \alpha \cdot X_{\min}$

再根据切线(直线1)间距离 d ,可以求出第二条切线的直线方程为:

$$y = \tan \alpha \cdot x + D_0 - D_1 \quad (2)$$

其中, $D_1 = d / \cos \alpha$;

以此类推,可以求出第 i 条切线的直线方程为:

$$y = \tan \alpha \cdot x + D_0 - (i-1) \cdot D_1 \quad (3)$$

其中: $i = 1, 2, \dots, N$;

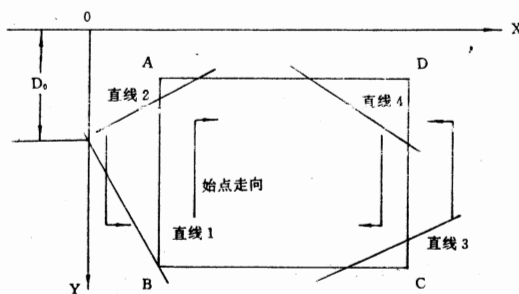


图4 长方体在二维坐标系中与直线相交的结束条件

这里点 $(X_s, y_s), (X_e, Y_e)$ 分别为切线段的开始和结束点,这时我们规定开始点先沿矩形框的BA边从B点向到A点移动,这时 $X = X_{\min}$,当 $y < Y_{\min}$ 时,再沿矩形框的AD边从A点向到D点移动,这时 $y = Y_{\min}$,当 $x > X_{\max}$ 时,切线段从B点切到D点的

算法结束。同样,在判断上述条件的同时,也要对结束点进行判断,这时我们规定结束点先沿矩形框的BC边从B点向C点移动;这时 $y = Y_{\max}$,当 $x > X_{\max}$ 时,再沿矩形框的CD边从C点向D点移动,这时 $x = X_{\max}$,当 $y < Y_{\min}$ 时,切线段从B点切到D点的算法也结束(直线1、2和直线3、4可通过程序设置标志区分)。

特殊情况下,即:直线2、直线3与X轴成 90° 和 180° 时(如图4)。直线2与X轴的夹角 $\alpha = 90^\circ$,方程简化为 $x = X_{\min} + (i-1) \cdot d$;条件简化为当 $x > X_{\max}$ 时,算法结束。直线2与X轴的夹角 $\alpha = 180^\circ$,方程简化为 $y = Y_{\min} + (i-1) \cdot d$;条件简化为当 $y > Y_{\max}$ 时,算法结束。直线3与X轴的夹角 $\alpha = 90^\circ$,方程简化为 $x = X_{\max} - (i-1) \cdot d$;条件简化为当 $x < X_{\min}$ 时,算法结束。直线3与X轴的夹角 $\alpha = 180^\circ$,方程变为 $y = Y_{\max} - (i-1) \cdot d$;条件简化为当 $y < Y_{\min}$ 时,算法结束。

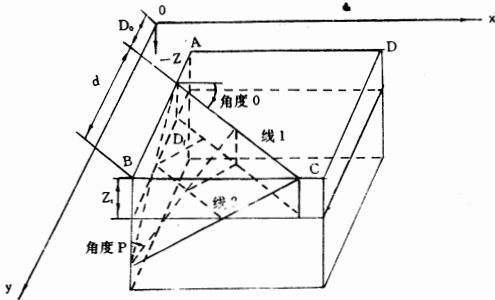
归纳所有四种情况,如下所示(不含特殊情况):

切线走向	始点	线段条件	终点	线段条件	方程
D → B	D	DC $x = X_{\max}$ $y > Y_{\max}$	A → D	DA $y = Y_{\min}$ $x < X_{\min}$	$y = \tan \alpha \cdot x + D_0 - (i-1) \cdot D_1$
B → C	B	CB $y = Y_{\max}$ $x < X_{\min}$	B	AB $x = X_{\min}$ $y > Y_{\max}$	
B → D	A → D	BA $x = X_{\min}$ $y < Y_{\min}$	D	BC $y = Y_{\max}$ $x > X_{\max}$	$y = \tan \alpha \cdot x + D_0 - (i-1) \cdot D_1$
	B	AD $y = Y_{\min}$ $x > X_{\max}$	B → C	CD $x = X_{\max}$ $y < Y_{\min}$	
A → C	A	AB $x = X_{\min}$ $y > Y_{\max}$	A → D	AD $y = Y_{\min}$ $x > X_{\max}$	$y = \tan \alpha \cdot x + D_0 - (i-1) \cdot D_1$
	B → C	BC $y = Y_{\max}$ $x > X_{\max}$	C	DC $x = X_{\max}$ $y > Y_{\max}$	
C → A	A → D	CD $x = X_{\max}$ $y < Y_{\min}$	A	CB $y = Y_{\max}$ $x < X_{\min}$	$y = \tan \alpha \cdot x + D_0 - (i-1) \cdot D_1$
	C	DA $y = Y_{\min}$ $x < X_{\min}$	B → C	BA $x = X_{\min}$ $y < Y_{\min}$	

2、三维坐标系中的结束条件及算法 如图5所示, Z_i 为切片或断层图象的距离, D_2 为切面间的距离, p 为切面与Z轴负方向的夹角,根据图5中的几何关系,可以求出同一轮廓上切线间距离 $d = D_2 / \cos p$ 。在三维坐标系中,设第一张切片或断层图象的Z坐标为最大,而最后一张的Z坐标为最小。

首先,考虑从B点 (X_{\min}, Y_{\max}) 开始切的情况。如图5所示,第1切面与第1和第2切片或断层图象所在平面的交线1和2的截距分别为 D_0 和 $D_0 + D_1$,其中, $D_1 = Z_i / \tan p / \cos \alpha$,而 $D_0 = Y_{\max} - \tan \alpha \cdot$

$X_{min}-d$ 。以此类推,与第 i 切片或断层图象所在平面的交线的截距为 $D_0+(i-1)*D_i, i=1,2,\dots,n$ 。这样, $D_0+(i-1)*D_i$ 作为切第 i 切片或断层图象的第 1 条切线的截距,按照 1 叙述的方法对同一切片或断层图象进行处理。作为三维结束条件,定义变量 $z, z=z_1, z_2, \dots, z_n; z_1 > z_2 > \dots > z_n$, 当 $z < z_n$ 时,算法结束。这里,要指出的是,当 $p=0^\circ$ 时, $D_z=d, D_i=D_z/\cos\theta$ 。从 A 点(X_{min}, Y_{min}), D 点(X_{max}, Y_{min}), C 点(X_{max}, Y_{max})开始切的情况,可类似地推导出来。



综上所述,求任意切面轮廓算法描述如下:

①置切面号 i , 切片或断层扫描图象号 j 为 1, $z=z_1$, 轮廓号为 $no(=1,2,\dots)$;

②输入角度 $\theta(0^\circ < \theta \leq 180^\circ)$, $p(0^\circ \leq p < 90^\circ)$, 及切面等间距 D_z , 并选择标志, 以区分是直线 1、2 或直线 3、4 中的一条;

③设置最大矩形框参数 $X_{max}, X_{min}, Y_{max}, Y_{min}$, 并初始化保存所求切面轮廓上特征点的结构

变量;

④由 1 叙述的方法, 求出所有有效切面与 j 层所在平面的交线和 j 层 no 轮廓线上的全部交点, 并保存这些交点到结构变量中, 其中: 切面号每次加 1;

⑤ $j=j+1, i=1$, 由 2 叙述的方法, 求出第 1 个切面与 j 层所在平面的交线的直线方程, 置 z 为 j 层 Z 坐标, 如 $z > z_n$, 则返回④;

⑥保存全部所求交点到磁盘, 算法结束。

六、结束语 通过上面的讨论, 得出了较为实用的求任意切面轮廓算法。在实际应用中, 我们先设 $\theta=180^\circ, p=0^\circ$, 切面间距离(等距)为 10 像素, 直线为直线 3; 然后, 用上述算法求出这些切面与原轮廓线的交点, 并保存下来, 最后, 通过三维网格图算法, 绘制出部分头部的三维网格图形^[5]。

参考文献

- [1] 金田 悠纪夫ほか“头部连续 CT 像からのパターン抽出と三次元立体表示”医用电子と生体工学 1981·8 第 19 卷 第 4 号
- [2] 鸟胁纯一郎 “最近の医用画像三次元立体表示の基本手法”医用电子と生体工学 第 24 卷 第 5 号 (Sept 1986)
- [3] [美] T·帕夫利迪斯 著《计算机图形显示和图象处理的算法》科学出版社出版 1988 年
- [4] [美] D.F. 罗杰斯 著《计算机图形学的算法基础》科学出版社出版 1989 年
- [5] 王学崇“三维重构软件的研制”四川大学硕士学位论文 1992.5

(上接 6 页)

6. 检查

因为 $L(V) = \{\varepsilon, prbtb, prbtbb, \dots\}$

而 $L(G) = \{prbtb, prbtbb, \dots\}$

所以 $L(V) = L(G) \cup \{\varepsilon\}$ 。

空句子 ε 在文法中有着特殊的作用。如果适当修改一下由 NFA M' 生成 G 的有关算法, 可以使求出的正规文法 G 满足:

$L(V) = L(G)$, 即 $\{\varepsilon\} \subset L(G)$ 。

另外, 根据实际问题, 给出正规式 V , 其表示形式不是唯一的。在上述实例中, V 也可以写成如下形式:

$$V = (prbt(b|bb|bbb))^*$$

根据上式执行算法, 其中间结果和最结果的表示形式一般是不相同的, 但总是可以做到满足:

$$L(V) = L(M') = L(M) = L(G)。$$

参考文献

- 1、左孝凌、李为镒、刘永才著,《离散数学》。
- 2、徐洁盘著,《离散数学导论》。
- 3、陈火旺、钱家骅、孙永强著,《编译原理》。
- 4、R. C. Gonzalez, M. G. Thomason,《Syntactic pattern Recognition》

正规文法生成算法及其实现

金之滨 仲肇铭 卢雷

山东工业大学计算机系 (济南, 250014)

摘要 本文给出了由正规式 V 生成有限自动机和正规文法的综合算法; 并用 PASCAL 语言实现了该算法, 具有很好的实用价值。

关键词 正规式 正规文法 有限自动机 算法

一、引言 根据实际问题建立相应的文法或自动机, 乃至自动生成, 是离散数学一个重要的应用方面。它在人工智能、模式识别和各种软件自动生成器中, 得到了广泛的应用。Chomsky 定义的四类文法和自动机之间的对应关系是:

- | | |
|-------|---------|
| 0 型文法 | 图灵机 |
| 1 型文法 | 线性界限自动机 |
| 2 型文法 | 下推自动机 |
| 3 型文法 | 有限自动机 |

其中, 3 型文法又称正规方法, 有限自动机分确定有限自动机 (Deterministic Finite Automaton; DFA) 和不确定有限自动机 (Nondeterministic Finite Automaton; NFA)。

本文综合文法和自动机方面的有关理论和方法, 给出了根据实际问题建立正规文法的算法。该算法可以手算, 人工建立正规文法; 也可以编出程序, 由计算机自动生成正规文法。

二、算法 由实际问题建立正规文法的算法可简述如下: [1], [2], [3]

- 1、根据实际问题, 给出正规式 V , 并检查 V 是否为正规式;
- 2、根据正规式 V , 求不确定有限自动机 M' (NFA M'), 使 $L(V) = L(M')$ 成立;
- 3、将 NFA M' 确定化成 DFA M , 使 $L(M') = L(M)$ 成立;
- 4、根据 DFA M 求正规文法 G , 使 $L(M') = L(G)$ 成立;
- 5、化简 G ;
- 6、检查 G 是否符合实际问题的要求。

该算法设计的理论根据是, 对任意正规式 V , 总存在 NFA M' 使下式成立:

$$L(V) = L(M') \quad (1)$$

进而存在 DFA M 和正规文法 G , 使下式成立:

$$L(V) = L(M') = L(M) = L(G) \quad (2)$$

在上述算法中, 第 1 步是关键。如果给出的正规式 V , 不能正确地表达实际问题, 以后的所有计算都是徒劳的。要避免这种错误, 根本的问题是给出的正规式, 必须是实际问题的充分必要条件。例如, 标识符被定义成以字母开头, 后跟任意个字母或数字的字符串。若以 a 表示字母, n 表示数字, 则正规式

$$V = a(a|n)^*$$

就是描述标识符的一个充要条件。而正规集 $L(V) = L(a(a|n)^*) = \{\text{标识符}\}$ 。

根据实际问题给出正规式, 一般要比根据实际问题直接写出正规文法容易得多。

根据描述正规式的文法

$$V \rightarrow V + T \quad (\text{其中 '+' 为正规式中的 '|'})$$

$$T \rightarrow TF | F$$

$$F \rightarrow F^* | (V) | i | \epsilon$$

检查 V 是否为正规式是不难实现的。

算法的第 2、3 步, 是有现成算法可寻, 一般不会产生错误或存在问题。

由于描述一个实际问题的正规文法不是唯一的, 因此, 要特别注意根据算法第 4 步求出的正规文法的具体形式。例如, 由标识符的正规式 $a(a|n)^*$, 根据上述算法求出的正规文法是

$$G1: S \rightarrow aA$$

$$A \rightarrow aA | nA | \epsilon$$

其中, S 表示 <标识符>。显然 $G1$ 是一个右线性的正规文法。众所周知, 标识符的正规文法, 常被写成左线性的, 即

$$G2: S \rightarrow a | Sa | Sn$$

$G2$ 是一个有直接左递归的正规文法。若消去 $G2$ 的直接左递归后得:

$$S \rightarrow aS'$$

$$S' \rightarrow aS' | nS' | \epsilon$$

显然, $G2$ 被完全化成了 $G1$ 的形式。故

$$L(a(a|n)^*) = L(G1) = L(G2) = \{\text{标识符}\}.$$

通过这个例子说明,只要求出的正规文法 G 与描述实际问题的正规式 V 之间满足: $L(G)=L(V)$ 即可。至于 G 是右线性的,还是左线性的不必追究。即是都是左线性的或右线性的,其形式也不一定完全一样。例如,上述右线性正规文法 G_1 ,也可以写成右线性正规文法 G_3 : $S \rightarrow aA$ $A \rightarrow aA | nA | a | n$

从文法 G_1 到文法 G_3 ,产生式 $A \rightarrow \epsilon$ 起了关键作用,型如 $A \rightarrow \epsilon$ 的产生式,在文法的研究中有特殊的作用。算法中的第5步,由于对文法的化简没有严格的定义,只要遵循公认的一些规定化简就可以了。

算法中第6步的作用,是进一步检查前五步的计算过程是否有误,特别是检查正规式 V 是否为实际问题的一个充要条件。至于如何检查,很难给出一个通用方法,只能具体问题具体分析。

如果要根据上述算法写成程序,由计算机自动生成正规文法,必须注意两点:1. 根据实际问题,给出正规式 V 仍然是人工完成;如何用计算机完成这一任务,尚待研究。

2. 由于没有一个通用方法来实现算法中的第6步,要根据具体问题具体分析。因此,提供丰富的人—机对话功能,是解决问题的方法之一。

如果问题不需要知道与正规式(V)等价的正规文法 G ,而只要知道能识别正规集 $L(V)$ 的有限自动机,则算法执行完第3步即可结束;也可以再追加化简 DFA M 的功能。在应用中,这是常遇到的情况。实际上可以把问题更一般化。由于式(2)中的 V, M' 和 G ,已知其中的任一个,其余三个均可求出来,且总使式(2)成立。

因此,仿上述算法,可以写出其它的算法。如已知 G ,求 M' 、 M 和 V ,且使式(2)成立的算法,或已知 G ,求 M' 和 M 的算法等。

三、实例 实例:^[4]正常心电图(ECG)如图(1)所示。

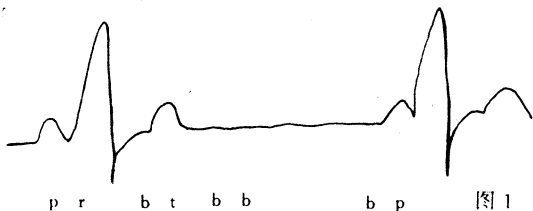


图1

心房首先产生 P 波,接着心室产生 r 波,之后是 b

波、t 波再 b 波。基本健康的心率允许 t 波之后的 b 波出现 1—3 个。即基本正常 ECG 由字符串 prbtb 或 prbtbb 或 prbtbbb 组成。如 prbtbprbtbb、prbtbbbprbtbprbtbbprbtbb 等,都是正常的。试求产生上述所有可能字符串的文法 G 。

执行上述算法的过程及相应结果如下:

1. 根据题目要求,给出正规式 V :

$V = (prbtb | prbtbb | prbtbbb)^*$

2. 根据 V , 求出 NFA M' , 使 $L(V) = L(M')$; 如图(2)所示。

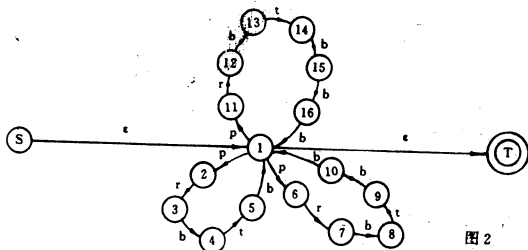


图2

3. 根据 NFA M' 求出 DFA M , 使 $L(M') = L(M)$; 如图(3)所示。

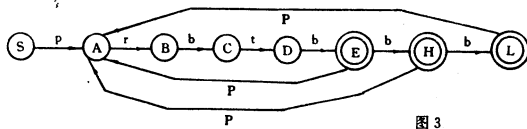


图3

4. 根据 DFA M 求出 G , 使 $L(M) = L(G)$ 。

$G = (V_n, V_t, P, S)$, 其中

$V_n = \{S, A, B, C, D, H, L\}$

$V_t = \{p, r, b, t\}$

$S = \{S\}$

$P = \{S \rightarrow pA, A \rightarrow rB, B \rightarrow bC, C \rightarrow tD, D \rightarrow bE, E \rightarrow bH, H \rightarrow bL, L \rightarrow pA, A \rightarrow rB, B \rightarrow bC, C \rightarrow tD, D \rightarrow bE, E \rightarrow bH, H \rightarrow bL\}$

5. 化简后的 G 为

$G = (V_n, V_t, P, S)$, 其中

$V_n = \{S, A, B, C, D, E, H\}$

$V_t = \{p, r, b, t\}$

$S = \{S\}$

$P = \{S \rightarrow pA, A \rightarrow rB, B \rightarrow bC, C \rightarrow tD, D \rightarrow bE, E \rightarrow bH, H \rightarrow bL, L \rightarrow pA, A \rightarrow rB, B \rightarrow bC, C \rightarrow tD, D \rightarrow bE, E \rightarrow bH, H \rightarrow bL\}$ (下转 4 页)

计算机递归分组排序算法

曲阜师范大学 周建钦(山东曲阜, 273165)

山东矿业学院 赵志远(山东泰安, 271019)

摘要 本文在介绍快速排序、桶排序算法基础上, 较为详尽地论述了计算机递归分组排序算法的算法描述及复杂性, 文末给出了实验结果。

关键词 快速排序 桶排序 递归分组排序

排序在计算机的系统软件和应用软件中使用非常广泛, 人们研究了许多排序算法。现在已经认识到, 若仅利用数据的序的信息设计出的排序算法, 平均工作量最少为 $O(N \log N)$ 。著名计算机科学家 Knuth 在文献[1]中将以前的主要排序方法——分析、比较, 认为排序算法中快速排序(Quicksort)的平均工作量达到下界 $O(N \log N)$ 。计算机应用中处理的数据, 均具有某种概率分布, 当 N 较大时更是如此, 若再利用数据的概率分布信息, 则排序的平均工作量可减少到 $O(N)$ 。文献[2]中分析了桶排序算法, 证明了在数据服从某些特殊概率分布的条件下, 排序的平均工作量为 $O(N)$ 。本文提出递归分组排序算法, 并证明在数据服从一般概率分布的条件下, 排序的平均工作量为 $O(N)$ 。

一、桶排序算法 为了后面叙述方便, 首先摘要介绍一下文献[2]的有关结果。桶排序算法描述如下:

已知数据文件 $F: y(i), 1 \leq i \leq N$, 开辟 N 个存贮单元 $R()$

D1, 求 $y()$ 的最大值 \max , 最小值 \min 。

D2, 将数据分成 N 组, 若

$J = [N(y(i) - \min) / (\max - \min)] + 1$, 其中 $[x]$ 表示取不大于 x 的最大整数, 则将 $y(i)$ 分入第 J 组。将 N 个数据(或其地址)重新存放在 N 个存贮单元 $R()$ 中, 组号小的数据放在前面, 同一组里的数据连续存放。

D3, 对 $R()$ 中的每一小组内数据排序。

若文件中的数据范围为 $[b_1, b_2]$, $B = b_2 - b_1 < \infty$; 数据服从概率分布, 其密度函数有界, 则桶排序的平均工作量为 $O(N)$ 。特别的, 当文件中的数据服从均匀分布时, 桶排序的平均工作量最少, 也为 $O(N)$ 。当文件中的数据服从正态分布时, 由于数据范围无界, 桶排序的平均工作量大于 $O(N)$ 。

二、递归分组排序算法 递归分组排序类似于 Quicksort, 它重新安排数据, 将数据文件分割成 N (N 是文件的数据个数) 个子文件, 使第 i 个子文件的数据小于第 j ($i < j$) 个子文件的数据, 然后递归地将每个子文件排序, 递归分组排序算法描述如下:

已知数据文件 $F: y(i), 1 \leq i \leq N$, 开辟 N 个工作单元 $W()$ 。

DD1, 令 $K = N, W(i) = y(i), 1 \leq i \leq K$

DD2, 求 $W()$ 的最大值 \max , 最小值 \min , 若 $\max = \min$, 则结束。

DD3, 若 $J = [(W(i) - \min)K / (\max - \min)] + 1$, 则将 $W(i)$ 分入第 J 组, 将 $W()$ 中数据放回原来的存贮单元中, 组号小的数据放在前面, 同一组内的数据连续存放。

DD4, 依次检查每个小组内的数据个数 r 。若 $r > 1$, 则令 $K = r$, 将这一小组内数据存放在 $W()$ 中, 实施 DD2—DD4。

DD5. 结束

下面分析递归分组排序算法的复杂性, 若文件 F 中的数据服从连续概率分布 $F(y)$, 则易证小组宽度 $(\max - \min)/N$ 趋向于 $O(N \rightarrow \infty)$ 。从而可证明, 第一次分组后, 每个小组内的数据服从均匀分布。第二轮分组后, 每个小组的数据个数基本上都为 1, 因而递归分组排序一般进行两轮分组后即结束。

第二轮分组, 即用桶排序算法排序每个小组(第一次分组后得到的小组)内的数据。由于第一次分组后, 小组内的数据服从均匀分布, 故排序第 k 小组数据的平均工作量小于 CN_k , 其中 N_k 是第 k 小组的数据个数, C 为一常数, $1 \leq k \leq N+1$

排序全部 $N+1$ 个小组内的数据平均工作量小于:

$$\sum_{k=1}^{N+1} C \cdot N_k = CN$$

由于第一次数据分组的工作量为 $O(N)$, 故递归分组排序的平均工作量为 $O(N)$ 。

若文件中数据服从离散概率分布, 即文件中有很多相等的数据, 递归分组的次数要少一些, 因而递归分组排序速度更快, 其平均工作量仍为 $O(N)$ 。

递归分组排序不要求数据服从某些特殊概率分布, 排序平均工作量为 $O(N)$, 从而优于桶排序。递归分组排序类似于 Quicksort, 数据结构简单, 易于程序设计, 具有很大的实用价值。

三、实验结果

数据个数	200	500	1000	1500
Quicksort	31	84	185	294
递归分组排序	18	44	87	130

表1 排序时间比较

表1是对 Quicksort 和递归分组排序在 IBM-PC 上做的一个比较, 算法语言采用 BASIC。表中的一列分别为对同一组随机数(由 RND 函数产生), 使用 Quicksort 和递归分组排序算法排序用的 CPU 时间(单位为秒), 实验结果充分证实了上述分析结论。

参考文献

- [1] Knuth, D. E., The Art of Computer Programming, Vol. 3, Sorting and Searching, 1973
- [2] Akl, S. G. and Meijer, H., Journal of Algorithms, 3(1982), 9-13
- [3] 复旦大学编,《概率论》,人民教育出版社,1979。
- [4] 周建钦、赵志远,“随机分组查找算法”,《科学通报》,35(1990),24:1905-1906。
- [5] 徐绪松、周建钦,“不等长记录的公式索引分组字典排序”,《软件学报》,2(1991),3:59-64。

一个简单实用的 FOXBASE 函数计算器程序

陈来军

四川三台县电力公司 (621100)

摘 要 本文介绍了一个简单实用的 FOXBASE 函数模拟计算器程序。

在实践中,笔者发现一些应用软件在使用时离不开计算器等额外的计算工具,有些应用软件虽配备有计算器功能,但也不方便,如操作麻烦、不能为编程人员在自己的程序中借用。另外,许多管理软件都是用 DBASE 系统完成的,如何在应用系统中随时调用 DBASE 的函数,也值得一提。笔者在完成《三台县电力公司供用电管理系统》过程中,开发出一个简单、实用、功能强的模拟 FOXBASE 函数计算器,程序清单如下。抛砖引玉,切望同行指正!

软硬件环境:FOXBASE2.1;AST 386/25。

程序说明:第3行:出错处理,若非 FOXBASE 函数或操作符的正确组合,则不显示结果。

4行:计算器在屏幕上开始行;

11-12 设置功能键;

29-34:将计算器清零;

39-35:退出。

应用一例:SET ESCA ON

ON ESCA DO JSQ

.....

注意屏幕的恢复,可使用 SAVE SCREEN 和 RESTORE SCREEN 命令。

再一例:SET FUNCTION 9 TO “[,”

HH=“10000”

@20,24 SAY “请输入编号(F9 计算器):”GET HH PICT“X9999”

READ

IF “[”≠HH

DO JSQ

ENDIF

需源程序清单者可同作者联系。

Occam2 并行处理程序在 Transputer 网络中的系统配置

马永强

西南交通大学计算机科学与工程系(成都, 610031)

摘要 本文对 occam2 并行处理程序在 Transputer 网络中的系统配置的若干问题作了较为详尽地介绍, 包括 occam2 语言程序在系统配置一级各并行进程的一些程序设计特殊要求, 系统配置语句描述及其说明等。本文还给出系统配置的一个例子。

关键词 Occam2 Transputer 并行处理 程序设计 系统配置

一、引言 用并行处理计算机体系结构借以提高计算速度是近年来人们一直所关注的课题。特别是 80 年代中期研制的用于并行处理的高性能新型晶片机 Transputer 及其并行处理程序设计语言 occam, 更加得到计算机界的极大兴趣和广泛应用。由于 Transputer 既是 occam 的硬件实现, occam 又可作为以 Transputer 为基本单元所组成系统的硬件描述语言, 使体系结构和高级语言一体化, 因此在以 Transputer 为基础的硬件系统产品上大多都配有 occam 编译。

用 occam 语言开发一个 Transputer 网络上的并行处理应用程序可以分两步进行: 第一步, 把一个串行处理的任务分解开来(或者针对问题本身的并行性)用 occam 语言以并行处理方式进行程序设计, 从而在逻辑上实现问题的并行处理; 第二步, 根据性能要求, 在所开发的程序中增加描述给定的 Transputer 网络的链接拓扑结构信息, 并描述在各个 Transputer 上运行的程序段。第一步为 occam 并行处理程序设计, 有关 occam 并行处理程序设计的方法和技巧作者已在文献[1]中作了详细地介绍。第二步称作系统配置(configuration), 它将使一个并行处理程序在 Transputer 网络上实现真正地并行执行。本文将以 TDS D700D 为运行环境, 详细介绍 occam2 语言程序在系统配置一级各并行进程的一些程序设计特殊要求, 系统配置语句描述及其说明, 并给出系统配置的例子。

二、系统配置级各并行进程程序设计的特殊要求 系统配置(configuration)将对 Transputer 网络中各处理机所执行的程序段作描述, 这些程序段所构成的并行进程是 Transputer 网络内最高层的结构, 并且实现真正的并行执行。因此, 各处理机所执行的程序段在程序设计时除了应满足 occam2 语言并行程序设计的一般要求外, 还有如下一些特殊的

要求:

(1) 各处理机所运行的程序(即系统配置级的各个并行进程部分)都必须是相对独立的程序段, 而且都必须写成一个过程的形式;

(2) 描述各处理机运行程序的过程, 其形参一般包含有通道。此外, 若有换名形参作为变量在过程体内使用, 则它所对应的实参变量一般只作指定存贮单元用的, 决不允许作为与其它处理机程序之间的公用变量;

(3) 描述各处理机运行程序的各个过程必须被分离编译, 生成独立的程序代码段, 以便装入相应的处理机中;

(4) 描述各处理机运行程序的各个过程都必须在系统配置内包含在一个 SC 折叠中, 不允许从一个库中取得。

上述这些特殊要求归结起来, 实质上就是 occam2 语言关于“变量用于存放值, 通道用于传递值”^[1]在系统配置一级的具体表现形式。由于各处理机程序所构成的并行进程是 Transputer 网络内最外层的结构, 它不会是其它进程的子进程, 因此, 各个并行进程之间不允许使用公用变量, 它们之间的一切数据交换都必须通过通道进行。换句话说, 系统配置一级的各个并行进程部分其内部不允许有自由变量存在。事实上, 如果在各处理机程序之间使用公用变量, 则当某一处理机所运行的程序改变某一公用变量的值时, 它并不能改变其它处理机上这一公用变量的值, 这就失去了多处理机之间数据的一致性。因此, 公用变量在系统配置一级是没有意义的。

由此不难看出, 把各处理机运行程序都表示成一个过程的形式并对它们进行分离编译是合乎情理的, 它不仅使得各个处理机都有一个相应的独立代码段, 也使得 occam2 编译程序在分离编译各程序段及系统配置时检测出不应存在的自由变量。

三、系统配置 1. 系统配置的描述

系统配置包括系统配置级的说明和处理机定位(或称处理机代码分配)两个部分,其形式为:

Configuration—level declarations

Placed PAR——处理机定位部分

(1)系统配置级说明(Configuration—level declarations)

系统配置级的说明可以包括:

- 只包含有一个过程的 SC 折叠
 - 使用 VAL 的常数说明
 - PROTOCOL 定义
 - #USE 行,引用只包含常数和协议定义的库
 - 用于任命处理机之间链路连接的通道的说明
- 其中,包含在 SC 折叠内的过程即上一节所介绍的将分配给处理机执行的程序段。

(2)处理机定位(Placed PAR)

处理机定位(代码分配)部分的语句是 occam 语言 PAR 结构的扩充形式,其一般形式为:

PLACED PAR

PROCESSOR number—1 transputer . type

Processor—level declarations

instance—1

PROCESSOR number—2 transputer . type

Processor—level declarations

instance—2

其中, number—i 为处理机编号,取整型值; transputer . type 为该处理机的型号,可以取 T8 (IMS T800), T4 (IMS T414) 或 T2 (IMS T212 或 IMS M212), instance—i 为一个过程调用语句。

处理机级的说明 (Processor—level declarations) 可以包括:

· 把系统配置级通道定位在处理机链路地址上,其形式为:

PLACE 通道名 AT 链路地址;

- 常数定义
- 变量说明
- 变量定位
- 简略与变量类型重定义
- 用作为该处理机“软通道”的通道定义

上述这种结构的意义是把各并行进程 instance—i 分别定位到编号为 number—i 的处理机上执行。

处理机定位还可以写成重复项形式,如:

PLACED PAR index=base FOR count

PROCESSOR index transputer . type

Processor—level declarations

instance—i

当然,若整个 Transputer 网络只包含一个处理机,则处理机定位时不必写出 PLACED PAR 语句。

2. 系统配置描述语句的说明

(1)处理机的编号 系统配置中各处理机的编号是该处理机的逻辑标识符,可以取整型数范围内的任意值。这些数只是用于标识从 TDS 软件来的处理机的信息,而对处理机定位是无关紧要的。

(2)处理机的代码分配 任何一个处理机其程序代码都是由包含在一个 SC 折叠内的一个单独的过程以及调用该过程的代码组成,这样一个过程便是系统配置级的一个并行进程部分。

(3)通道的定位 通道的定位用 PLACE 语句,它把 occam2 通道与处理机链路联系起来。一个通道若被定位在链路上两次,则该通道必须是被定位在一个处理机的一条输入链路上和另一个处理机的一条输出链路上,使得这两个处理机在逻辑上建立起通信通道。

一个只被定位在链路上一次的通道称为一条与 Transputer 网络以外的环境连接的“悬挂(dangling)”链路,网络程序通过这条链路可以与 TDS 内运行的 EXE 程序通信。

通道定位的描述必须与 Transputer 网络硬件所连接的链路的拓扑结构相匹配,否则在装入网络程序时将会失败。

(4)根处理机 当 Transputer 网络程序由宿主主机自举(booting)时,网络中必须有一个处理机与 TDS 连接,该处理机被称着“根处理机”(若网络由 ROM 自举,则包含 ROM 的那个处理机称着“根处理机”)。注意,从根处理机到网络的任何一个处理机都必须至少有一条路径通过处理机链路连通,使得网络程序能够被装入。

此外,TDS 提供的 occam2 编译程序要求网络中的根处理机在系统配置中必须总是第一个被说明的处理机。

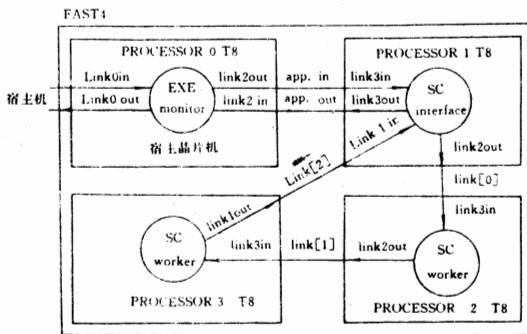
(5)软通道 软通道是只能在一个处理机内使用的通道。如果分配给一个处理机的进程其形式通道参数要求连接到该处理机自身,则它所对应的实参必须是一条软通道。这样的软通道必须在系统配置的处理机级说明中被说明为一条通道,它不被定位在一条链路地址上。软通道对提供一条流水线终结的循环返回以及对填充不会被使用的链路参数是很有用的。

四、例子 本节将结合一个流水线分类程序,

举例说明程序在一个 Transputer 网络上运行时配置语句的描述。关于流水线分类程序本身的具体算法可参看文献[1]、[2],本节不作介绍。

该流水线分类程序的功能是从根处理机的一条链路上输入一连串字符,通过按字典顺序分类后再从根处理机的另一条链路上输出。分类程序将被配置在 FAST4 中的 3 个处理机(T_1, T_2, T_3)上。FAST4 的剩余 1 个处理机(T_0)作为宿主晶片机,通过在 TDS 内运行一个 EXE 程序,完成分类程序与外界输入输出的驱动。

宿主晶片机及网络程序的软件结构如图 1 所示。为避免过多地分散注意力,把精力集中在网络程序的配置上,下面只对网络程序的配置部分作介绍。



网络程序的系统配置用一个 PROGRAM 折叠描述,其内容为:

```

{{{PROGRAM prog3
  {{{F prog3. tsr
  #USE header          一常数说明及协议定义
  ...SC PROC interface  一在网络中根处理
                        机上运行的过程的分离编译
  ...SC PROC worker    一在网络中其它处
                        理机上运行的过程的分离编译
  VAL number. of. transputers IS 3;
  CHAN OF string app. in, app. out;
  [number. of. transputers]CHAN OF letters link;
  ...link constants    一链路地址常数说明
  ...configuration     一系统配置语句描述
  }}}

```

```

}}

```

其中,链路地址常数说明的折叠的内容是:

```

VAL link0out IS 0;
VAL link1out IS 1;
VAL link2out IS 2;
VAL link3out IS 3;
VAL link0in IS 4;
VAL link1in IS 5;
VAL link2in IS 6;
VAL link3in IS 7;

```

系统配置语句描述的折叠的内容是:

PLACED PAR

PROCESSOR 1 T8

```

PLACE app. in AT link3in;
PLACE app. out AT link3out;
PLACE link[0] AT link2out;
PLACE link[number. of. transputers-1] AT
link1in;
interface (app. in, app. out, link[0], link[num-
ber. of. transputers-1])

```

PROCESSOR 2 T8

```

PLACE link[0] AT link3in;
PLACE link[1] AT link2out;
worker(link[0], link[1])

```

PROCESSOR 3 T8

```

PLACE link[1] AT link3in;
PLACE link[2] AT link1out;
worker(link[1], link[2])

```

五、结束语 本文在文献[1]介绍的 occam2 语言程序设计的基础上,对 occam2 并行处理程序在以 TDS D700D 为运行环境的多个 Transputer 的网络上系统配置的若干问题作了详细地介绍,并结合一个流水线分类程序举例说明系统配置的描述,目的是为了使得读者能够理解并掌握 occam2 程序在多处理机网络上配置的方法。本文所介绍的有关系统配置的这些问题同样也适用于运行在 TDS 环境以外的“独立程序”。

参考文献

- [1] 诸昌铃, 马永强, 《并行处理程序设计语言 OCCAM》, 西南交通大学出版社, 1990 年 9 月
- [2] INMOS Limited, TRANSPUTER DEVELOPMENT SYSTEM, Prentice Hall International (UK) Ltd, 1988

微机外部设备的虚拟接口方法

李桂青 赵占朝

武汉工业大学工程结构抗震研究所(武汉, 430070)

摘 要 本文给出一种外部设备虚拟接口方法。用户可在指定的外设接口文件中定义所使用的外部设备。系统运行时, 自动对外设接口文件中的信息进行加工处理, 将所定义的外部设备连接到绘图系统上。利用这种连接外设的方法, 使绘图系统中的绘图过程对外设采用统一的格式进行处理, 从而实现了外部设备的虚拟接口。

关键词 虚拟接口 接口文件 绘图系统 外部设备

一、前言 在开发计算机绘图系统时, 常常碰到一个系统与外部设备, 例如图形显示器、图形打印机、绘图机等连接问题。为了实现系统与外部设备的连接, 通常采用的方法是将常用的几种外设用程序来实现与外部设备的连接。使用时根据所用的外部设备的类型, 在指定的几种外设型号中进行选择。这种方法使得绘图系统被限制在几种外设上使用, 而没有提供给用户自行将不同型号的外设连接到绘图系统上的方法。因此不便于程序的推广, 特别是在计算机技术迅速发展的今天, 外部设备不断地更新发展, 这就要求绘图系统应该具有与外部设备的软接口, 可以让用户自行方便地定义所使用的外部设备。

本文给出一种外部设备虚拟接口方法。用户可在指定的外设接口文件中定义所使用的外部设备。系统运行时, 自动对外设接口文件中的信息进行加工处理, 将所定义的外部设备连接到绘图系统上。利用这种连接外设的方法, 使绘图系统中的绘图过程对外设采用统一的格式进行处理, 从而实现了外部设备的虚拟接口。这既益于程序系统的设计, 也便于绘图系统的移植推广。这种方法可适用于大多数外部设备的连接。

二、图形显示器虚拟接口的实现 对于通过适配器连接到计算机上的图形显示器, 有两种驱动方式: 一种是利用计算机系统提供的软中断进行作图; 另一种是对图形缓冲区直接进行设置, 完成图形操作。采用后一种方式具有较快的作图速度, 但对计算机的硬件依赖性较强。我们研制的系统 GSL 综合了上述两种操作方式, 在图形显示器的文件接口中

供用户选用。接口方法是根据所用计算机图形缓冲区的首址, 图形显示器的分辨率, 图形的软中断等数据, 按照接口文件指定的格式填写并存放至接口文件中。接口文件中可以有多个不同组不同类型显示器的接口数据, 使用时是根据所用显示器在接口文件中定义的顺序由图形处理过程取得相应的数据。图形显示器接口文件定义的格式为:

N, D_1, D_2, \dots, D_n

SX, SY, GM, GD

其中 N : 是图形缓冲区的个数。它表示一幅图形被存放在 N 个图形缓冲区中, 而 D_1, D_2, \dots, D_n , 给出了这 N 个缓冲区的首地址。例如 IBM-PC 微型机是由奇偶扫描线两个存区组成系统的缓冲区, 此时 $N=2$ 。

SX, SY: 是图形显示器的分辨率。

GM: 是使显示器进入图形状态的软中断号。

GD: 是在显示器上画一个点的软中断号。

上面给出的数据综合了图形显示的两种驱动方式。当 $N=0$ 时, 表示利用系统提供的软中断进行作图, 即利用中断 GD 实现图形显示。反之当 $N>0$ 时, 根据缓冲区的首地址 D_1, D_2, \dots, D_n , 图形处理过程直接对图形缓冲区进行操作。下面给出 IBM-PC 微型机显示器的数据接口:

2, 47104, 55296

640, 200, 6, 12

三、图形打印机的虚拟接口 现有计算机系统上只配有屏幕硬拷贝的功能, 它只能打印出固定大小的图形, 不能满足计算机绘图系统的需要。因此需要设置图形打印机的虚拟接口。

图形打印机一般都是智能式的,它提供了一组打印机命令,用来控制打印机的操作。由于各种类型的打印机,它们的命令格式是不完全相同的,因此,就需要将给定的打印机的命令格式通知绘图系统。图形打印机连接到绘图系统上的方式就是在打印机接口文件中定义指定的打印机命令的格式。绘图系统执行图形打印时,自动对定义的命令进行解释,并根据命令的格式组织要打印的数据,输出给打印机来完成图形的打印。打印机的数据接口格式为:

L,P

{打印机状态设置命令}

{打印机打印点阵的命令}

{打印机复位命令}

其中 L:是打印机可以打印的最大宽度;

P:是用来定义打印机是 9 针的还是 24 针的,当 P=9 时,表示是 9 针打印机;当 P=24 时,表示是 24 针打印机。

后面的三组指令可以有多个命令组成,来控制打印机的动作。这些命令可在打印机手册中查到。{打印机打印点阵的命令}规定了打印机打印点阵的命令格式。在这组命令的定义中,规定了二个参数 N,L。这二个参数用来定义要打印点阵的列数。其中 L 是一个整型数。当 $L > 0$ 时,说明用两个字节 n_1, n_2 来表示列数,列数的值由 $n_1 * L + n_2$ 给出。当 $L = 0$ 时,用 n_1, n_2, n_3, n_4 给出列数,这四个字节分别对应列数的千位到个位。上面给出的命令定义除规定的二个参数外,命令中的所有字符都用十进制的 ASCII 码来表示,并且在每组命令的最后用字符“%”作为结束标志。下面给出打印机 M=1724 的接口数据:

```
2176 24
27 74 18 % {设置行间距命令;ESC Jn}
27 71 N 256 % {打印点阵命令;ESC G n1 n2}
27 64 % {打印机初始化命令;ESC @}
```

四、绘图机的虚拟接口方法 绘图机与图形打印机有些类似,它也是智能式的外设,自身拥有一个命令集,用来控制绘图笔的移动。它是工程 CAD 系统的主要图形输出工具。

绘图机的接口方法与打印机的接口方法基本相同。它也是将有关绘图机的指令定义在绘图机的接口文件中,由系统执行绘图操作时,进行命令的解释,并按照绘图机规定的命令格式输出图形数据。绘图机的接口数据格式为:

XL,YL

C

{绘图机初始化命令}

{绘图机选笔命令}

{绘图机画线命令}

{绘图机归零命令}

其中,XL,YL:为绘图机的作图范围。C 是绘图机与计算机硬件接口的连接方式。当 $C = 'S'$ 时,表示绘图机通过串行插口与计算机相连。反之通过并行插口连到计算机上。后面的四组指令集定义了绘图机的操作命令。每组命令可以由多条组成,每组命令的最后以“%”作为结束标志。在命令的定义中用到了三个参数 X,Y,P。X,Y 表示绘图机上点的坐标,P 是选择笔号的参数。在这三个参数的前面都要加上“@”,以区别指令中的其它字符。在这几组命令中的{绘图机画线命令}是由两条绘图机指令组成的,第一条给出绘图机抬笔移动的命令,第二条定义了绘图机落笔移动的命令。

利用上面介绍的方法,可以将多种型号的绘图机连接到绘图系统上。连接时,只是将有关的几条命令存放到接口文件中,而程序系统无需作任何修改。因此这种外设的虚拟接口方法,使用方便,具有较强的实用性。作为一个例题,下面给出 DMP-42 绘图机的接口定义:

```
8410 5941
S % {通过串行插口连到计算机上}
:: A H O ECM
LO % {绘图机初始化命令}
PI % {绘图机选笔命令}
U @X, @Y
D @X, @Y % {绘图机画线命令}
H % {绘图机归零命令}
```

五、小结 我们应用本文所给出的虚拟接口方法,在开发微机软件系统中进行了应用,当购置新设备后,或移植软件时,软件无需改动,仅修改接口文件中的参数,方便易行,软件的维护费用低,收到了很好的效果。

参考文献

- [1] 张福炎,蔡士杰,“交互式图形系统的虚拟设备接口”,《计算机工程》,No. 4,1987, PP1-5.
- [2] 李桂青,罗持久著,《工程设计专家系统的原理与程序设计方法》,气象出版社,1991, 10.
- [3] Janes Foley, Defining Interfaces at a High Level of Abstraction, IEEE Software, Vol. 6, No. 1, Jan. 1989, PP25-32.
- [4] R. J. Hubbard, Computer Graphics and Displays, CAD Journal Vol. 16, No. 3, 1984.

一种动态门限图象二值化技术

白 焱

华中理工大学 数控技术中心(武汉, 430074)

摘 要 本文提出了一种通过求灰度图象直方图熵最大值的方法来实现图象二值化。这种方法使分割门限动态化, 并且易于编程实现, 实验效果良好。

关键词 图象分割 直方图 熵

一、引言 把灰度图象变换为二值图象, 常用固定门限对图象进行分割。这种方法的缺点是只能处理具有双峰直方图的图象, 对于对比度较小的图象就会损失信息。采用动态门限可克服上述缺点。文献[1]采用线性动态门限分割法, 即将图象的灰度均值与方差作为约束条件。这种方法虽然简单实用, 但在门限选择时, 仍然遇到参数不稳定问题, 门限系数只是一个经验参数, 没有与图象的灰度均值或方差成某种必然函数关系。这样图象的分割质量也就难以保证。

下面介绍一种方法采用动态门限来实现图象二值化。

二、图象分割方法 设 f_1, f_2, \dots, f_n 为图象的灰度直方图值, 则灰度的概率分布为

$$P_i = f_i / N, \quad i = 1, \dots, n$$

其中 N 为像素总数, n 为灰度级数, 并有 $\sum_{i=1}^n f_i = N$ 。

将这组数据划分为两组概率分布, 一组取灰度级 $1 \sim s$, 另一组则取 $s+1 \sim n$:

$$A: P_1/P_s, \quad P_2/P_s, \dots, P_s/P_s$$

$$B: P_{s+1}/(1-P_s), P_{s+2}/(1-P_s), \dots, P_n/(1-P_s)$$

$$\text{其中 } P_s = \sum_{i=1}^s P_i, 1-P_s = \sum_{i=s+1}^n P_i$$

则它们的熵为:

$$\begin{aligned} H(A) &= - \sum_{i=1}^s \frac{P_i}{P_s} \ln \frac{P_i}{P_s} \\ &= - \frac{1}{P_s} \left[\sum_{i=1}^s P_i \ln P_i - P_s \ln P_s \right] \\ &= H_s/P_s + \ln P_s \end{aligned}$$

同理

$$\begin{aligned} H(B) &= - \sum_{i=s+1}^n \frac{P_i}{1-P_s} \ln \frac{P_i}{1-P_s} \\ &= (H_n - H_s)/(1-P_s) + \ln(1-P_s) \end{aligned}$$

$$\text{其中 } H_n = - \sum_{i=1}^n P_i \ln P_i, H_s = - \sum_{i=1}^s P_i \ln P_i$$

定义 $H(A)$ 与 $H(B)$ 之和 $4(s)$ 则得

$$4(s) = \ln P_s (1-P_s) + \frac{H_s}{P_s} + \frac{H_n - H_s}{1-P_s}$$

此时 $4(s)$ 的意义为: 图象中物体与背景的信息量。当 $4(s)$ 取最大值时的 s 值即为所求的分割门限。

三、门限的选取 可以证明, 当 A 与 B 的分布相同或相近时, $4(s)$ 取得最大值。假设 A, B 的分布为均匀分布, 则

$$\begin{aligned} 4(s) &= \ln \frac{s}{n} (1 - \frac{s}{n}) + \frac{-\ln s}{n} \ln \frac{1}{n} + \frac{-(n-s)}{n} \ln \frac{1}{n} \\ &= \frac{(n-s)}{n} \ln s (n-s) \end{aligned}$$

此时可求得使 $4(s)$ 取最大值的 s 值为 $s = \frac{1}{2}n$ 。

对于 P_1, \dots, P_n 为对称分布的情形, 即

$$P_i = P_{n+1-i}, i = 1, \dots, n$$

也可证明当 $s = \frac{1}{2}n$ 时 $4(s)$ 取最大值。

这样, 在编程实现图象分割时, 只要对图象的灰度级数 n 进行统计, 就可获得分割门限 s 。

在实际情况中, s 的选取可根据具体情况进行稍微的调整。如在物体灰度低于背景灰度的情况下, 当分割所得的 1 像素(物体)远大于 0 像素(背景)时, 则使 $s = s+1$, 重新分割图象, 以避免二值图象向物体像素偏移; 反之, 当分割所得的 0 像素远大于 1 像素时, 则使 $s = s-1$, 重新分割图象, 以避免向背景像素偏移。

四、多门限的选取 当有多个不同灰度的物体出现在同一背景中, 即图象的灰度直方图为多峰时, 可将这一算法扩充为求

AutoCAD 中汉字处理软件开发

尹文生 张志坚

华中理工大学机械工程一系(武汉, 430074)

摘要 本文通过分析 AutoCAD 软件字库和 CCDOS 软件的特点, 建立了适合各种 AutoCAD 软件版本的矢量汉字库。针对 AutoCAD 软件文字处理慢的缺点, 提出了一个方法, 开发了一个能适合各种 AutoCAD 软件版本的汉字输入软件。

关键词 汉字处理软件 矢量汉字库 汉语拼音输入 汉字文件输入

1. 引言 AutoCAD 软件是功能很强的图形软件, 在我国流行比较广, 在很多地方尤其在工厂的设计绘图中应用广泛。但由于 AutoCAD 是西文软件, 在图中添加汉字不是很容易的。归纳起来, 以前在 AutoCAD 中添加汉字的方法有以下几种。

1.1 利用 AutoCAD 的形的概念, 建立汉字形文件

这个方法是按照 AutoCAD 形的定义将汉字矢量化, 即将汉字笔画用形来描述, 构成形文件。在 AutoCAD 的图中添加汉字是通过汉字的内码或其它代码去寻找汉字的形文件名, 然后调用该形文件在规定位置写出汉字。该法有以下几个缺点:

(1) 需要构造的形文件很多, 一般只能构造一些常用汉字。

(2) 查找不方便, 查找汉字一般是查找一张汉字编码表, 找到对应的形文件。

(3) 难以对一串汉字进行处理。由于汉字大多

是以串为单位进行处理的。单字处理显然不方便。例如要将一串汉字均匀地写在某点两边, 就得计算出各字的位置, 操作起来不太方便。

1.2 建立 AutoCAD 的汉字小字库

将 AutoCAD 形的定义也就是字的定义组合成一个小字库。国标一二级汉字有 6000 多个, 由于 AutoCAD 2.6 版本软件编译字体文件的容量有限, 形成不了大书库, 只能将汉字区位表中的汉字按一个区或两三个区形成小书库。显然这种方法查找汉字仍是不容易的。该方法的改善是采用参数化的方法。如我校研制的 FORAUT 软件, 就不一定要用户知道各个汉字的内码, 可在 CCDOS 下编译 FORTRAN 绘图程序, 运行这个程序就可形成一个演播文件, AutoCAD 可以接受该文件并将汉字写出。该法的缺点是具有参数化绘图所共有的缺点即调试困难。任何微小的改动都要重编程序。

1.3 使用汉化的 AutoCAD 软件

五、实验结果 上述算法用 FORTRAN 编程, 在 PC 微机上实现。对实验的十几幅图象, 二值化效果均较好。

由以上分析可以看出, 这种算法的优点在于它利用直方图的整体分布特性, 自动求取分割门限。并且算法易于编程实现, 是一种简单实用的图象分割方法。

参考文献

[1] 彭嘉雄, “图象的二值化边缘抽取技术”, 《电信科学》, No. 10, 1986。

$$4(s_1, s_2, \dots, s_k) = l_n \left(\sum_{i=1}^{s_1} P_i \right) + l_n \left(\sum_{i=s_1+1}^{s_2} P_i \right) + \dots + l_n \left(\sum_{i=s_{k-1}+1}^{s_k} P_i \right) + l_n \left(\sum_{i=s_k+1}^n P_i \right)$$

$$= \left(\sum_{i=1}^n P_i \right) - \frac{\sum_{i=1}^{s_1} P_i l_n P_i}{\sum_{i=1}^{s_1} P_i} - \dots - \frac{\sum_{i=s_{k-1}+1}^{s_k} P_i l_n P_i}{\sum_{i=s_{k-1}+1}^{s_k} P_i}$$

的极大值。这里 s_1, s_2, \dots, s_k 为灰度 $[0, n]$ 内的整数, 满足条件 $s_1 < s_2 < \dots < s_k$ 。选择 s_1, s_2, \dots, s_k 以求 $4(s_1, s_2, \dots, s_k)$ 的极大值。K 个门限将图象分为 K+1 个部分。

这是一个比较好的方法。汉化 AutoCAD 的菜单是汉字显示的,查找汉字按 CCDOS 的方法。汉字能作为 TEXT 命令的字符串使用,因此使用起来很方便,而且能实现 TEXT 的功能,如居中排列字符串,右对齐字符串等等。但是汉化的 AutoCAD 一般存在以下几个问题:

(1) 有的容易出现死机。这是由于汉化 AutoCAD 改动了 BIOS,西文 DOS 和 CCDOS 有出现冲突的地方,操作稍有失误即可造成死机,从而遗失数据。

(2) 难以跟上 AutoCAD 软件的版本更新速度。AutoCAD 版本一更新,就必须重新推出新的汉化 AutoCAD 软件。否则新的 AutoCAD 软件就不能使用汉字。

2. 本文的汉字处理方法 对以上方法的比较和对 AutoCAD 字库定义的研究我们提出一个建立大字库,用 AutoLISP 建立汉字输入软件的方法,即:首先按照 AutoCAD 字的概念定义一个大的汉字库,它容纳了国标一二级汉字,然后参照 CCDOS 汉字拼音输入方案建立一个汉字拼音输入软件。它的使用方法与 CCDOS 的汉字拼音输入方案基本一致。最后再对输入的一串汉字进行各种位置处理,如居中,对齐等。另外考虑到使用汉字的一些特殊要求,增加了一些汉字的特殊排列方式,增强汉字处理功能。

3. 建立汉字库 首先用一个转换程序将国标一二级点阵汉字库文件转化成矢量汉字库文件。该矢量汉字库文件是按 AutoCAD 字定义方法形成的,后缀为 .SHP。每个汉字的内码同 CCDOS 方案一样,是两个 ASCII 码的组合。然后用 AutoCAD 11.0 软件的字编译功能编译该文件,形成一个后缀为 .SHX 汉字库文件。前面我们指出 AutoCAD 2.6 软件形成不了大字库,但是 AutoCAD 11.0 软件形成的字库文件却适合于 AutoCAD 2.6 软件。这是一个很重要的结论。这就是说该汉字库适合于 AutoCAD 2.6 及以上各种版本软件,也就是说本文的建库方法和下面提到的汉字输入方案适合于 AutoCAD 软件的各种版本。

4. 汉字拼音输入方案 基于建立汉字库时得到的一个结论,本文力图使汉字拼音输入方案适合于 AutoCAD 软件各种版本。因此编制汉字拼音输入软件时是在 AutoCAD 2.6 版本软件上进行的,使用的开发工具是标准的 AutoLISP 开发语言。

4.1 汉字的查找

建汉字库时我们给每个汉字定了一个名即内码,它是由两位 ASCII 码组成。同 CCDOS 的机内码一样,是由区和位各加上 160 形成的。各种汉字输入

方案都遵守这点。在 AutoCAD 中写一个汉字很简单,因为一个汉字是大字库中的一个字,只要调用该大字库并给出这个汉字的内码即可写出这个汉字,因此写汉字的关键是查找它在大字库中的内码。

4.2 拼音方案

我们采用 CCDOS 的拼音方案查找汉字名。本文的声韵母代表字母同 CCDOS 一样。经过分析 CCDOS 的拼音方案,我们可以得到拼音扫描表。首先用 DEBUG 得到一个汉字输入扫描表文件,然后用 C 语言编一个程序将拼音码扫描表构造出来。这张去掉首尾码的表有 20 多 K 字节。虽说不大,但对于 AutoCAD 2.6 版本软件来说已经达到极限了。即使使用 AutoCAD 11.0 安装这张表也需要耗时 5 至 7 分钟。虽然调入内存中查找速度快,但安装过程实在难以忍受,而且装入的字绝大部分用不上,浪费内存。这个方案不太可行。

本文提出了一个方法解决了这个问题,即使使用 AutoCAD 2.6 版本软件也是行得通的。本文首先对拼音扫描表穷举各种可行拼音码的组合及每一个组合可以查到的字。这实际上是先建立一个汉字拼音输入的索引表(注:运行建立这个索引表所费时间相当长,这也从一个方面说明直接按拼音码搜索扫描表相当费时),然后进一步将这些组合按第一码分成 25 个索引文件(其中以 V 打头的文件不存在),这些文件平常不装入内存,只在需要时才调入。由于这些文件内部数据经过了排序,所以查字时读操作次数相当少。可以说按这种方法读取文字速度很快,基本解决 AutoCAD 软件影响汉字输入的主要问题。

4.3 拼音汉字的显示和选取

汉字同音字很多,输入相同的拼音码对应显示的汉字很多。必须加以显示和选取。本文认为 CCDOS 的显示和选取方式可以借鉴。仿造 CCDOS 的方式,本文设置了一条绿色小长条于图形区域底部,以红色字在上面显示同音字。但翻页符改为:下翻为“[”,上翻为“]”,这点同自然码的规定一致。选取则敲被显示汉字前面的 0,1,...9 数字。由于本文的字库建立时汉字的名或内码和 CCDOS 的机内码是一样的,对于选取的汉字,只要直接将该字的内码提取出来就可以写出该字。这一点也是今后开发汉字字符串应用程序的基础。

当原图结构复杂时,显示和擦除同音汉字必须谨慎使用 REDRAW 命令,防止重新生成原图。本文采用了局部 REDRAW 函数,只对局部实体作重画操作,因而不会产生原图的重新生成,能够节省时间。

5. 本文汉字处理功能 本文汉字拼音输入和处理系统(AutoCAD 汉字系统, 简化为 AHZ)已能取代 AutoCAD 中的 TEXT 命令。AHZ 的主选项共有 3 项: 直接插点输入, 文件输入和格式定义。其中直接插点输入方式相当于原来的 TEXT 命令。考虑到汉字的特点、使用者的习惯和实现起来的难度, 本文的文字(包括汉字和其它文字符号)的输入是先选插点按左对齐方式输入文字, 在文字输入完毕后再根据是否要重新排列来进行右对齐、居中等排列。

5.1 直接插点输入

这种方式包容了 TEXT 命令。当回答该选项选点后, 即可用拼音进行汉字输入, 也可用“%”开头进行其它 ASCII 的插入, 实现中西文的混合输入。当用回车中止输入选择后, 系统即提示是否再对所输文字进行重新排列, 然后可进行其它方式的排列。原有的 TEXT 命令的几种排列方式仍然有效, 本文又根据汉字造表的需要构造了两种排列方式, 即在横框或竖框内均匀排列文字, 这对于填写标题栏, 明细表等非常有用。

5.2 文件输入方式

汉字的输入速度与采用的软件有关, 用 CCDOS 的拼音输入方式输入汉字的速度是不快的。在 AutoCAD 内用拼音输入同样也是不理想的。对于低版本

的 AutoCAD 软件来说, 汉字显示的速度很慢, 因此当输入的汉字多时, 特别令人不满意。本文设置了一个文件输入方式。因为用其它汉字输入软件建立的汉字文本与 AutoCAD 的字的内码是一样的, 因此为了提高汉字输入速度, 可以先用其它快速汉字输入软件建立文本文件, 然后选用文件输入方式, 给定文本文件名, 本软件自动从该文件中一行一行地读入文字, 然后也可象直接输入方式一样选取排列方式进行排列。这样可大大地提高汉字输入速度。

5.3 文字输入风格

本选项主要用来选择书写方式和书写字体。

6. 容错处理 本软件由于建立了两个层, 用于显示要输入的字的背景和文字。在任何时候, 系统退出时都必须删除这两层, 并且恢复到调用前的状态。为此必须设置很多错误码, 记下当前的运行状态。在进行删除时还必须进行非空检验, 以防错删, 这是因为 AutoCAD 对不存在的实体作删除时会出错。

由于本软件是用 AutoLISP 编制的, 汉字库能适合 AutoCAD 各种版本软件的, 同时又进行了许多出错处理, 用它来处理汉字问题不会出现死机现象, 破坏数据, 而且它还是随 AutoCAD 版本更新而变化的。

西文 Turbo C 2.0 显示汉字的最简单办法

赵炳跃

济南军区指挥自动化工作站(250002)

在使用 Turbo C 2.0 进行编程时, 经常遇到怎样在 EGA 和 VGA 显示器上输出汉字的问题。尽管报刊上已有多种办法予以介绍, 但大都离不开编写一个小规模驱动程序的模式。虽然可以使用, 但仍然感觉不便。本文提供的方法只需改动一个字符, 问题便迎刃而解。

在 Turbo C 的控制台 I/O 头文件 CONIO.H 中, 有一个名为 directvideo 的变量, 通过它的值可以决定显示的内容送往视频缓冲区, 还是通过 BIOS 中断调用进行输出。其缺省状态是 directvideo=1, 即进行

直接向缓冲区写入 ASCII 码, 也就是通常所说的直接写屏处理。尽管速度很快, 但兼容性差(必须与 IBM PC 机 100% 兼容)。所以把头文件 CONIO.H 打开, 将其中的 directvideo 变量的值为 0 即可。即:

```
int __cs... directvideo=0;
```

这样编译后的程序通过 BIOS 中断调用输出, 屏幕上便可直接显示汉字。同时, 各种制表符号必须改用汉字的制表符, 而不能使用原来英文的制表符(即 ASCII 码大于 80H 的图形符号)。

该内容在东海 DH-0530B 型机上试验通过。

数据库管理系统文字型数据的类 HASH 编码技术

与准顺序式紧缩存储结构

张育民

西安医科大学第一临床医学院计算机中心(710061)

摘要 文字数据即字符串,它与图形、图像、声音以及其他形式的数据一样,属于非数值数据。在微型机的数据库应用中,对这类数据通常需要压缩(紧缩)存储,否则若照搬数据库原结构存储将会产生存储空间的冗余。本文介绍在关系式数据库 Foxbase(dBASE)应用环境下所设计的一套旨在提高系统存储性能的技术与结构——HASH 编码技术和顺序式紧缩存储结构。由于在实现时结合 Foxbase(dBASE)环境对传统的 HASH 技术以及顺序式紧缩结构进行了一些必要的修改,故此非正式地称它们为“类 HASH 编码技术(Similar HASH Coding Technique)”与“准顺序式紧缩存储结构(Quasi-Sequential Condensed Storage Structure)”。另外,本文还给出了类 HASH 技术的编码/解码以及准顺序式紧缩文件数据的存取算法,讨论了此项设计对系统效率的影响。

一、序言

在数据库管理中经常涉及文字信息的存储与操作,这种操作的特点是:操作对象为非数值型数据,即数据串,相应的操作称为串操作。现今的数据库系统[其中包括 Foxbase(dBASE)数据库系统]擅长于数值操作,而若要操作文字数据就要做一些辅助性操作,否则,不是直接操作困难就是要产生一些问题,其中最为严重的问题在于:在微型机上开展以上工作时,系统实际存储的信息量太少,不能满足管理的基本需求。微型机自身存储量小固然是影响文字信息存储的不利因素,但若设法在现有条件下对存储数据的方法加以改进,则无疑是现实、经济而又有效的解决途径。我们通过对 Foxbase(dBASE)的存储方式做具体分析之后发现以下规律:Foxbase(dBASE)的 DBF 结构总是以固定长度的记录字段存放数据的,而在文字型信息集合中,各数据项值的长度属性可以分成固定长度(定长)的和可变量度(变长)的两种,对于变长数据项的字段存储长度必须按项中信息可能出现的最大长度来设定,如此便造成大量存储空间的冗余(被空格所占用)。可以达到消除冗余的数据组织方法是:将各数据项按定长的和变长的两种属性分别存至两个子集合,定长信息组成一个定长库(库中记录称作定长记录),其存储在逻辑上依然采用 DBF 库结构,对于变长信息集合在逻辑上则采取先截去各数据项尾部空格,再将它

们依次连续存放的“字串顺序紧缩存储结构”(各变长项组成一个变长记录串,项与项、记录串与记录串之间用字符分界),如此构造成一个变长库(数据仍借用 DBF 库存放)。变长记录串在变长库中的存储位置作为定长库与变长库的联系而存放在定长库的相应记录中。这种定变长信息分离存储的方法虽然可以消除冗余,但也相应地带来了另一个问题,即:对变长的关键词项将如何处理。如果将它存于定长库中,虽然利于查找,但却会产生冗余,当变长关键词较多时冗余现象就更为严重;若将它紧缩后存于变长库中,则查找将非常不便。我们在做有关设计时有一种想法,即:设法找到一种代码,它直接由关键词转换而成,存取时将代码替代关键词,作为变长记录串的代表而与变长记录串存储地址构成一关系,此关系在变长数据紧缩存储时产生,在信息读取时将作为变长记录串的查找线索。在代码的生成过程中应要求其满足两个条件:1. 码长较短且长度固定;2. 生成速度快且方便。事实上这种代码转换是客观存在的,它就是本文所要着重讨论的 HASH 编码。

二、HASH 编码技术及其在数据库应用中的修正

HASH 编码法(又称散列法或杂凑法)是建立关键词与包含关键词记录的存储地址对应关系的一种方法,它对数据的组织如下:

设关键词 K 为数据库 Dv 中某一记录 R 的成员,

现构造一个函数 $f(K)$, 使得由 $f(K)$ 产生包含 K 的记录与某一自然数 A 的对应关系:

$K \xrightarrow{f(K)} A$, 其中 A 为 K 所在记录的存放地址[一般地直接有 $f(K)=A$ 并且 A 的值呈离散分布], 我们把上述关系的构造过程称为 HASH 编码映射, $f(K)$ 称为 HASH 函数, A 称为 K 由 $f(K)$ 而得到的映象, 由 A 构成的一维表称为 HASH 表(图 1)。

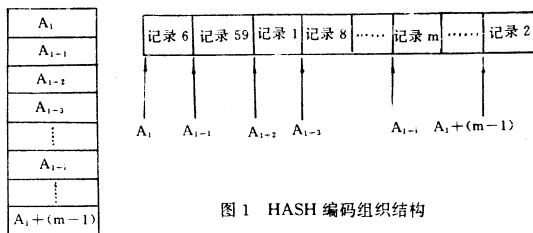


图 1 HASH 编码组织结构

(图中, A_1 为 HASH 表起始存储地址, m 为最大记录序号)

在采用 HASH 编码存储时有可能产生冲突[即 $K_1 \neq K_2$ 但 $A_1 = A_2$ 或 $f(K_1) = f(K_2)$]。因此, 应用好 HASH 技术的关键不仅要选择散列均匀的 HASH 函数, 还要设计一个恰当的处理冲突的方法。对于前者的选用无一定的规则可循, 实用中一般采取试选的方法, 在若干候选函数的实际运算结果中挑出散列效果最佳的一个作为应用中的 HASH 函数。对于后者的设计在理论上若有若干方法可资选用^[1], 其目的都是使设计符合相应的应用环境, 最终达到由键词集合到存储地址集合的一对一映射, 以避免冲突。关于以上两个方面的详细讨论已超出本文范围, 感兴趣的读者可查阅有关教科书^{[1][2]}。

为便于以下叙述比较, 有必要给出 HASH 技术的查找与处理冲突的算法描述:

FUNCTION hasearch(K ; keytype; hatable; ARRAY[1..m]OF rectype); integer;

{hasearch 为在 hatable 中查找包含键词 K 的记录存储地址的函数, keytype 为 K 所在域的类型, rectype 为记录所在域的类型, m 为 hatable 的表长}

BEGIN

$j := f(K)$;

IF hatable[j] = nilrecord {hatable 表中第 j 个位置上无记录}

THEN RETURN($m+1$) {函数返回 $m+1$ 表示未找到}

ELSE [IF hatable[j].key = K

THEN RETURN(j) · (查找成功)

ELSE [REPEAT

$i := \text{readjust}(K)$ {调用相应的处

理冲突运算 readjust 求下一地址}

UNTIL (hatable[j].key = K) or (hatable[j] = nilrecord)

IF hatable[j] = nilrecord

THEN RETURN($m+1$)

ELSE RETURN(j)]]

END; {hasearch}

由以上算法可以看出, HASH 方法对于冲突的处理是数据的存储阶段进行的。

若考虑 Foxbase(dBASE)数据库 DBF 结构的存储特点, 将 HASH 编码技术应用于其管理实践就需做两点改动。其一是存储地址的下标表示需做修改。由以上定义可知, HASH 地址下标为一上界确定的自然数区间(1..m)上的数(m 为计算机允许的最大整数), 并且下标是由 $f(K)$ 的值而产生的一个随机数; 由于数据库 DBF 的数据是按记录号递增的顺序存储的, 记录号为一个递增的自然数序列。这就在存储地址的产生方式上与定义方法相矛盾; 既然要用 DBF(变长库)文件结构实现各变长记录串的顺序紧缩存储, 则要将库记录定义成单一字段的结构(仅存储串值)并且在逻辑上把相邻记录看作是一个连续的存储空间(相邻记录间的逻辑联结靠算法加以实现, 详见三)。因为作顺序存储的各变长记录串长度不一, 因此应用中的 HASH 编码地址必定是由串在变长库文件的存储起址记录号(REC)和串起始字符在库记录中的位置(ADD)共同组成的一个二元组 $\langle \text{REC}, \text{ADD} \rangle$ 来决定。修正后的 HASH 编码成为:

$K \xrightarrow{f(K)} \langle \text{REC}, \text{ADD} \rangle$ 。如此, 修正后的 HASH 编码

把互不相关的 $f(K)$ 与 $\langle \text{REC}, \text{ADD} \rangle$ 构造成为一个三元组 $\langle f(K), \text{REC}, \text{ADD} \rangle$, 并将它记录于称做 HASH 索引表的数据库中(图 2)。

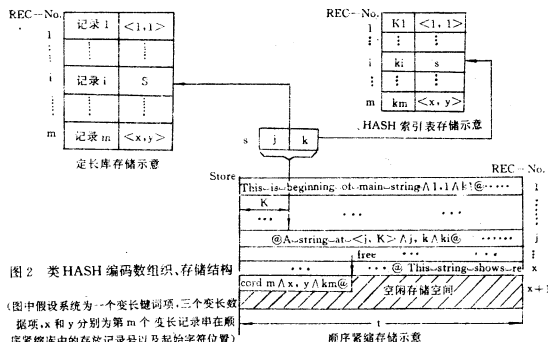


图 2 类 HASH 编码数组、存储结构

(图中假设系统为一个变长键词项, 三个变长数据项, x 和 y 分别为第 m 个变长记录串在顺序紧缩库中的存储记录号以及起始字符位置)

其二是处理冲突的时机与方法的改动。由于 f

(K)与(REC, ADD)并无内在关联,存储于 HASH 索引表中的三元组是即时构成的。这种顺序存储、即时登录(于索引库)的方法在数据存储时并不会察觉到冲突的产生(事实上也不需要冲突处理),与传统的 HASH 方法不同,修改后的方法在查找时先在 HASH 索引表中找到与 K 的 HASH 函数 $f(K)$ 相匹配的所有记录,再到变长库中读出所有对应记录串,该记录串集合中可能存在着 $f(K1)=f(K2)$ 但 $K1 \neq K2$ 的串,若将各串中的相关数据项与关键词比较就可以舍去不相匹配的串,最后若剩下相匹配的记录串则说明查找成功,否则查找失败。可见,修正后的方法是在查找时包含了对冲突的处理,这在时机上与 HASH 方法有所区别。

三、类 HASH 技术的编码/解码算法以及数据存储和其他运算的实现

1. 编码 coding(K):

(1)求 $f(K)$;

(2)在 HASH 索引库中查找包含 $f(K)$ 的记录,若查到,则返回记录中的另外两个项 REC 和 ADD 的值,否则返回空记录并结束;

(3)由 (REC, ADD) 在变长库中定位变长记录串的起址,定长库记录指针也按 (REC, ADD) 指向相应记录。

2. 解码 decoding(K, REC, ADD)除了将变长记录串中各项还原外,还要处理冲突;

(1)在变长库中第 REC 号记录的第 ADD 个字符位置读取同关键词 K 对应的项 K' ;

(2) K' 与 K 比较,若相等,则分解还原出变长记录串中的其他项,否则直接转向下一个信息的解码。按匹配的各变长记录串的 (REC, ADD) 在定长库中找到相应定长记录并与还原后的变长信息合并输出。

3. 变长信息顺序紧缩存储运算 appendrec:

(1)顺序式紧缩存储结构:它是一个连续的存储区域(有资料称之为“堆”^[1]),其逻辑结构是一向量 $store(1..maxsize)$, $maxsize$ 为此空间的容量。数据的存储方法是:设有一指针(算法中用 Free 表示)始终指向存储区中未被占用空间的低地址,每当系统提出存储申请时,都是从该地址起,划分出一块与申请等量的空闲空间供存储之用。(2)变长记录串的构成:①变长数据项 field 的结构化定义:

TYPE fieldtype = ARRAY[1...m] OF char;

field = fieldtype; 其中: field 为一变长数据项; m 是原始库中变长数据项(紧缩前的)字段定义长度。

②变长记录 recstr 的结构化定义及生成运算 trimcon;

TYPE rectype = RECORD

reclen : integer; {记录串长}

recch : ARRAY[1..n] OF fieldtype {组成记录串的各项}

END; 其中: n 为串中变长数据项的个数。

生成变长记录串的实质是属于同一记录的各项变长数据项紧缩后的链接。设 recstr 为长度可变的 rectype 型串变量, field 是最大下标为 n 的 fieldtype 型串数组。假设生成 recstr 变量的空间是足够的(Foxbase 和 dbase 均为 $\leq 64K$), 算法如下:

PROCEDURE trimcon (VAR recstr : rectype; field : ARRAY[1..n] OF fieldtype);

{将 field[i] 截掉尾部空格后链接至 recstr 尾部, ' ' 为 field[i] 的分界符}

BEGIN

FOR i = 1 TO n DO

[field[i] := TRIM(field[i]) + ' ']; {消除冗余并加分界符}

len-field[i] := LENGTH(field[i]);

movch(recstr.recch, field[i], recstr.reclen + 1, 1, len-field[i]);

recstr.reclen := recstr.reclen + len-field[i]

END; {trimcon}

其中, TRIM 和 LENGTH 分别是标准的截字符串尾部空格以及求串长函数, Foxbase (dBASE) 中均有其相应的函数。

(3)变长记录串存储运算 appendrec:

利用 DBF 库结构存储变长数据时需对 3. (1) 中结构略加修改。一方面, 3. (1) 中的存储区域在用 DBF 库实现时则与库记录以及记录中的字段长度都有关, 它在形式上表示为一个二维数组 $store(1..maxrec, 1..t)$, $maxrec$ 为 DBF 允许的最大记录个数 (≤ 10 亿), t 为 DBF 字段的存储长度 (≤ 254), 它们与 $maxsize$ 有如下关系: $maxsize = maxrec \times t$ 。另一方面, 由于变长记录串在变长库中存储起址是一个二维下标量 (startrec, startat), 在 HASH 索引表中存放的是 $f(k)$ 与该下标量的三元组 $\langle f(k), startrec, startat \rangle$, 图 2 中, S 为记录串地址的索引指针, 变量 free 含意同上且初值为 1。因此, 变长库的逻辑存储结构为:

TYPE stype = RECORD

startrec, startat : integer {变长记录串在 store 中的起址}

END;

设已存信息串名字为 main、类型为 fieldtype (字符数组)。算法 appendrec 描述如下:

```

PROCEDURE appendrec (VAR main; fieldtype;
rec, rectype);
(将变长记录串 rec 传至 main 末尾, '@' 为记录
串分界符)
BEGIN
CASE
free+rec. reclen-1>t;
WHILE rec. reclen>0 DO
[recl:=SUBSTR(rec*recch,1,t-free+1);
movch(main, recl, free, 1, t-free+1);
rec, recch:=SUBSTR(rec. recch, (t-free+1)+1, len-
rec-(t-free+1));
rec, reclen:=LENGTH(rec, recch);
rec.startrec:=rec.startrec+1;
rec.strartat:=1];
free+rec. reclen-1=t;
[movch(main, rec. recch, free, 1, rec. reclen);
rec.startrec:=rec.startrec+1;
rec.startat:=1];
free+rec. reclen-1<t;
[movch(main, rec. recch, free, 1, rec. reclen);
rec.startat:=t-free-rec. reclen+1];
END;
movch(main, '@', free, 1, 1)
END; {appendrec}

```

其中, SUBSTR 为标准的求子串函数, Foxbase(dBASE)中也包含有相应的函数。以上用到的 movch 过程是“字符串移动”算法:

```

PROCEDURE movch (VAR t; string; s; string; i, j, num;
integer);
(将位于 s[j] 起始的 num 个字符序列依次传送到由
t[i] 起始的位置)
BEGIN
FOR k:=0 TO num-1 DO t[i+k]:=s[j+k]
END; {movch}

```

读取变长记录串的运算与其存储运算互逆, 设计方法相似, 这里从略。

除了以上主要算法之外, 还有插入、删除、修改等其他运算, 现简述如下:

4. 插入运算: 若要在变长记录顺序存放的主串 main 中插入一变长记录串, 势必要挪动插入位置以后的所有信息, 这在信息量较大的情况下是无法实现的。因此, 插入运算只有在主串 main 的尾部进行才有意义, 而这实际上已变成存储运算了。

5. 删除运算: 由于对变长信息的操作需以 HASH 索引表为线索, 这就使得删除运算可不直接在变长信息库中进行, 同时考虑到读取信息时要处理冲突。因此, 删除运算的前半部分操作与编码/解

码操作的前半部分一样, 在按 K 找到匹配信息之后, 将它们在 HASH 索引表中的记录以及定长库中的定长记录删除即可。

6. 修改运算: 定长记录信息可以修改, 变长信息不宜修改, 其理同插入运算, 这里不再重复。对于变长关键词项的修改有一补救措施: 建立一个“参照库”, 其记录中包含两个数据项, 分别存放修改前的旧值与修改后的新值。每当对信息存取时, 先按关键词新值在“参照库”中找到旧值, 再分别用新旧值进行存取操作。当读出变长信息后, 将相应旧值换成新值。

7. 信息属性预定义: 在建立原始信息库结构, 说明数据属性(数据名, 类型, 长度)的同时应说明数据的变长属性, 用特定字符区分该数据项是定长的还是变长的。在原始库分裂成定长、变长两个库时, 区分符将作为数据项分离归类的依据。

四、对系统效率的影响

1. 运行时间: 由于原始信息由原来的一个文件分存至两个文件之上, 与常规的同类数据库系统相比, 影响类 HASH 编码系统运行效率的主要因素在于增加的一次对变长库的 I/O 操作。但类 HASH 技术对该文件中记录串的访问实现了直访地址的随机定位操作, 这对提高系统运行速度是一个很好的补偿。

2. 系统存储(压缩)率: 本设计系统存储性能的改善是依靠最大限度地消除存储空间冗余来体现的。为了更清楚地表达这个问题, 有必要涉及数据压缩率的概念。数据压缩率是压缩后的数据存储量与压缩前的定义量之比。现举例说明类 HASH 方法的数据压缩率:

设某变长数据项存储字段的定义长度为 100B (Bytes), 而数据实际所占用的平均存储长度为 50B, 若存储数据为一万个记录, 紧缩后的数据实际存储量为 $10000 \times (50 + 1) = 510000B = 510KB$ (KiloBytes) (1 是为各数据项所加的一个分界符)。系统定义的存储量为 $10000 \times 100 = 1000000B = 1MB$ (MegaBytes), 在用 Foxbase(dBASE)数据库实现时需要附加一个 HASH 索引库(假定有一个变长关键词项, 结构见图 2), 其占用的空间假定为 $(6 + 5 + 3) \times 10000 = 140KB$ (假设 f(K) 值占 6 位字长, REC 占 5 位, ADD 占 3 位)。类 HASH 技术的压缩率 = $(510 + 140) / 10000 = 65\%$ 。由此可见, 基于以上假设的字段的压缩效果是显著的。我们在一个医学专题文献情报检索系统中用本文方法, 在微型机条件下对两万多条文献(包括文摘, 硬盘容量 20MB)进行了压缩存储, 获得了小于 80% 的系统压缩率。通过实践, 我们还总结了以下提高存储效率的两个(下转 30 页)

使用 LISP 编程空间不够的问题之解决方法

熊瑞萍

黄永宁

广西计算中心 (南宁, 530022)

摘要 本文总结了使用 LISP 语言编程的实际经验,介绍了使用此种语言编程空间不够的问题之解决途径。

关键词 LISP 编程 空间分配 变量 函数

Auto CAD 作为一种高效的绘图工具,大大提高了在微计算机上进行各种图形设计和处理的效率,提供了很强的人——机交互绘图功能,特别是系统本身带有的内部高级语言 Auto LISP 更是给用户提供了强有力的二次开发手段。如何有效地使用系统提供的 LISP 语言进行编程,开发出灵活、实用、方便的应用系统是广大用户共同关心的问题。

LISP 语言在人——机交互绘图界面和图形处理方面具有许多其它高级语言所不能及的优势,但也存在着一个最令人头痛的问题,即程序运行过程中常常会出现下列显示信息:

Insufficient node space(节点空间不够)

这是因为 Auto LISP 是运行在最大为 45KB 的内存空间里的。在 Auto LISP 启动时,它需要两个很大的内存区域:一个称为“堆”(Heap),另一个称为“栈”(stack)。可根据应用系统的需要用 DOS 的“SET”命令,合理地分配堆栈空间,但总和不能超过 45000 字节。如:

C>SET LISPHEAP=35000

C>SET LISPSTACK=10000

但在编制较大的应用程序时,45KB 的空间往往不够用,致使程序无法执行下去。许多用过 LISP 编程的人都对此很苦恼,但对 LISP 又留恋不舍。在此,笔者总结了用 LISP 语言编程的一些经验,给出了一些避免出现空间不够问题的解决方法,希望对各位同仁能有所裨益。

1. 空间的合理分配

堆空间是用以存贮所有的函数和变量的,程序中拥有的函数和变量越多,堆空间也用得越多。栈存贮函数的变元和局部结果,函数的嵌套越深,执行递归的次数越多,所用的栈空间也越多。这二者中,堆空间占的比例往往比较大,应较多地将空间分配给它,但要保证二者之和不能超过 45000 字节,可根据

需要灵活调整二者的分配比例。

2. 使用虚拟函数分页

在装入 Auto LISP 函数之前,先执行函数 (VMON)即可激活虚拟函数分页功能。当执行此函数后,在 Auto LISP 程序占满内存时,会将不常用的函数分页出去,在需要时又自动读回。这种处理是自动进行的,对于你的程序是完全透明的,因而使用此功能,函数的定义可是无限多的。

3. 将程序化整为零

如上所述,使用虚拟分页后,一般能够解决空间不足的问题。但人们发现,在运行大程序时,仍旧常常会出现串空间不够问题。这是因为虚拟内存系统只是对函数进行分页,你仍然要有足够的空间来容纳当前程序运行所使用的数据和变量。在一个程序很大、变量很多时,空间就有可能不够,解决的办法是将一个大程序化为几个小程序,通过调用关系完成原来的功能。

4. 回收结点空间

对于一些短时使用的函数和变量,一旦没用时,即将它们设为 nil,以回收被占用的空间。如上述将程序化整为零后,当子程序执行完,不再使用时,即可设为 nil,这样,一边执行程序,一边回收空间,就可避免空间不够问题。

5. 减少变量/数据所占结点空间

可采用以下方法:

①变量名尽量短;

②变量名在不影响程序正确性的情况下重复使用;

③对于仅在当前程序中使用的变量应定义为局部变量,这样只要该程序结束,这些变量所占用的结点空间也随之释放。

6. 借用系统函数名

Auto LISP 提供了大量的预定义函数,在启动

LISP 时,它们将被装入内存,作为标准函数供用户使用。但在用户的应用程序中,往往有许多系统函数是根本用不上的,而它们却在内存中占据了一定的内存空间。可通过借用这些函数名来定义自己的函数,巧妙地夺取这些无用函数所占用的结点空间。只是,在借用系统函数名时,一定要慎重选择,否则可能影响应用程序的正确执行。

7. 压缩程序

提高程序设计质量,尽量使程序语句简洁,结构紧凑。这主要依赖于编程者对 LISP 语言的熟练程度,编程技巧的掌握及程序设计风格。

如何解决空间不够的问题,是使用 Auto CAD 从事二次开发工作的人所共同关心的,也是用 LISP 编程所要解决的首要问题。除以上所述,是否还有更好的解决方法,还有待大家探索。

一种窗口光条选择录入汉字域内容的方法

滕 锋

新疆电子计算中心(乌鲁木齐市,830011)

摘 要 作者采用在汉字文本栏录入处开窗口的办法,运用 FOXBASE+, 实现了窗口光条选择汉字快速录入,并给出了实用程序。

关键词 窗口 光条选择 汉字录入

在设计信息管理系统时,建立一个数据量充足的数据库在所难免,这就必然要涉及到大量基础数据的输入,而影响数据输入速度的主要栏目就是汉字栏目。当数据量较大时,这个矛盾就更为突出了。但是,只要我们仔细分析,即可看出,数据库中很多栏目(域)的内容取值是有一定范围的。如:人事档案信息管理系统中,性别域的取值不会超出“男”、“女”,籍贯域的取值,不外乎我国各省、市、县。若输入时,逐字地击键输入,对一个非专业操作员来说,速度之慢,使人生厌,因而往往使所开发系统的很多功能或束之高阁或半途而废。为解决此问题,以往编制的辅助汉字输入程序一般采用词组或代码输入方法,有的还加以屏幕提示,但以上两种常用的方法仍要键入词组代码或代码,而且,若无屏幕提示(所剩屏幕不够提示所有代码内容),代码的记忆也是一件非常令人头痛的事,况且很多栏目的内容千变万化,规律性差,无法实现“代码化”输入方法。

通过近年来编制信息管理系统的实践,使我想到了—种简便的快速录入方法:在汉字文本栏录入处开一个小窗口(首先将现有录入基本内容装入—内容数据库中),在输入汉字文本栏内容时,通过此窗口,采用光条选择方式进行录入。若遇到内容数据库中无所需内容时,按—键,则采用汉字直接录入,并将此内容记入内容数据库中,使内容数据库内容得以充实,下次即可用光条方式调用录入。并且内容数

据库内容扩充时,加之以重复性、合法性检查。遵循这种设计思想,笔者编制了在 FOXBASE+下实现的汉字栏目窗口光条选择录入程序。在 FOXBASE+环境下编制的信息管理系统,可根据实际需要随时建立相应内容库(见库结构),并调用此过程,就可方便地进行快速汉字文本栏目输入。使用时,根据屏幕下方亮区的操作提示,就可方便地进行汉字栏目的快速输入。由于该过程(或程序)具有学习功能,即可以不断地向内容库中加入以前没有的信息,日积月累,内容库会在应用程序运行中逐步完善,使汉字文本栏目的输入速度越来越快,工作效率大大提高。同时,在 FOXBASE+下建立内容库,由于建库与调用同时进行,所以方便灵活,建库可以通过内容库辅助管理程序进行,也可在圆点提示符下自行建立,辅助管理程序一方面可以辅助用户建立汉字文本栏目对应的内容数据库,另一方面可以对内容数据库进行管理(增删改),避免其过时性,同时也可进行重复性检查,以保证内容数据库的最小冗余性。

以上程序通过在新疆维吾尔自治区石油公司乌鲁木齐石油采购供应站网络信息管理系统的使用,效果非常显著。

以上程序在 SUPER-XT 型机上通过,可适用于 XT-286,386 机,在 FOX-BASE+, DBASE III 环境下使用。

程序清单见后。

库结构如下:

Field	Field Name	Type	Width	Dec
1	MEAN	Character	30	

记录:

Record # MEAN

- 1 新疆乌市
- 2 新疆昌吉
- 3 新疆阿尔泰
- 4 新疆伊犁
- 5 新疆吐鲁番
- 6 新疆都善

* Program.: XNAMEZK. PRG——光标选择录入汉字串
调用程序.

* 环境设置

@9,10 SAY "地名:"

STORE "JGB" TO INNAMEK

STORE "DWDM" TO OUTNAMEK

STORE 10 TO SMOL && 提示窗口大小(行).

STORE 9 TO HH && 提示窗口起始行数.

STORE 10+6 TO LL && 提示窗口起始列数.

STORE SPACE(20) TO GNAME

DO XNAME WITH INNAMEK,OUTNAMEK,SMOL,;

HH,LL,GNAME && 调用 XNAME

RETURN

* Program.: XNAME. PRG——光标选择录入汉字串程
序.

PROCEDURE XNAME

* 过程参数设置

SELECT A

USE &INNAMEK

SELECT B

USE &OUTNAMEK

SET COLOR TO N/W

@22,0 SAY SPACE(79)

@22,0 SAY'操作:PgUp—上页,PgDn—下页,↑—上,;

↓—下,←—选中,T—汉字录入,End—返回'

SET COLOR TO W/N

STORE 0 TO SPACE

STORE 1 TO SXX

STORE HH TO SCOL

STORE SPACE * SMOL+SXX TO SRECNO

STORE .T. TO RSPD

DO WHILE .T.

STORE HH TO SCOL

STORE SCOL TO LSCOL

STORE SMOL TO LSMOL

STORE SPACE * SMOL+SXX TO SRECNO

IF SRECNO(<=RECCOUNT0

GO SRECNO

ENDIF

IF RSPD

@SCOL,LL CLEAR TO SCOL+SMOL,79

ENDIF

DO WHILE LSMOL>0 . AND. . NOT. EOF0. AND. RSPD

SET COLOR TO W/N

STORE TRIM(MEAN) TO SUNIT

@ LSCOL,LL SAY SUNIT

SKIP

STORE LSCOL+ TO LSCOL

STORE LSMOL-1 TO LSMOL

ENDDO

STORE .F. TO RSPD

STORE HH TO SCOL

SET COLOR TO N/W

STORE SPACE * SMOL+SXX TO SRECNO

IF SRECNO (<=RECCOUNT0

GO SRECNO

ENDIF

STORE TRIM(MEAN) TO SUNIT

@SCOL-1+SXX,LL SAY SPACE(20)

@SCOL-1+SXX,LL SAY SUNIT

SET COLOR TO W/N

I = 0

DO WHILE I # 18. AND. I # 3. AND. I # 5. AND. I #

24.;

AND. I # 13. AND. I # 6. AND. I # 84. AND. I

116

I = INKEY0

ENDDO

@ 23,44 SAY SPACE(33)

DO CASE

CASE I = 6

SELECT A

APPEND BLANK

REPLACE UNIT WITH SUNIT

EXIT

CASE I = 18 . AND. SPACE>0

STORE 1 TO SXX

STORE SPACE-1 TO SPAGE

STORE .T. TO RSPD

LOOP


```

CASE I = 3 . AND. (SPACE+1) * SMOL < RECCOUNT0
STORE 1 TO SXX
STORE SPACE+1 TO SPACE
STORE . T. TO RSPD
LOOP
CASE I = 5 . AND. SXX >= 1
SET COLOR TO W/N
@SCOL-1+SXX,LL SAY SPACE(20)
@SCOL-1+SXX,LL SAY SUNIT
SET COLOR TO N/W
IF SXX-1>0
STORE SXX-1 TO SXX
ELSE
IF SPACE>0
STORE SPACE-1 TO SPACE
STORE 1 TO SXX
STORE . T. TO RSPD
ENDIF
ENDIF
STORE . T. TO FXZPD
LOOP
CASE I = 24 . AND. SXX(<=SMOL
SET COLOR TO W/N
@SCOL-1+SXX,LL SAY SPACE(20)
@SCOL-1+SXX,LL SAY SUNIT
SET COLOR TO N/W
IF SXX<SMOL
IF SPACE * SMOL+SXX+1,
(<=RECCOUNT0
STORE SXX+1 TO SXX
ENDIF
ELSE
STORE 1 TO SXX
STORE SPACE+1 TO SPACE
STORE . T. TO RSPD
ENDIF
ENDIF
LOOP
CASE I = 13
@23,44 SAY '选中'
STORE TRIM(MEAN) TO SUNIT
SET COLOR TO N/W
@SCOL-1+SXX,LL SAY SPACE(20)
@SCOL-1+SXX,LL SAY SUNIT
SET COLOR TO W/N
@23,44 SAY '选中;'
@23,50 SAY SUNIT
LOOP
CASE I = 116. AND. I = 84
STORE SPACE(20) TO SUNIT
@23,44 SAY "录入;"GET SUNIT
READ
STORE LTRIM(TRIM(SUNIT));
TO SUNIT
*合法性检查
LOOP
ENDCASE
ENDDO
RETURN

```

本刊启事

为了更快地推动、促进我国计算机产业的迅猛发展,及时地为计算机科研、开发、生产等部门提供计算机行业的最新发展动向,加速高新尖端技术、最新软硬件的开发、移植、引进,及时地为广大读者及计算机爱好者奉献更多更新的计算机专业技术资料,并使众多作者、译者脱颖而出,《计算机应用研究》杂志社现在开展优惠出版各类计算机专业技术资料业务,欢迎广大作、译者踊跃赐稿。具体出版业务欢迎来函来电来人商议。联系人:张钢(邮编:610041)

《计算机应用研究》杂志社启

推广应用开发研究计算机的良朋益友

传播计算机产品广告信息的忠实媒介

欢迎订阅《计算机应用研究》杂志

●刊号 CN51-1196/TP
ISSN1001-3695

●邮发代号 国内:62-68
国外 BM4108

●全年每套订价: 10.8元(国内)
\$12元(国外)

●全国各地邮局均可订阅

●国外总发行:中国国际图书贸易总公司

一组巧妙的 DOS 实用例程

——兼谈批处理文件使用技巧

黄 庆 程

福建省交通科学技术研究所 (福州, 350004)

摘要 本文应用一些鲜为人知或 DOS 手册上经常被忽略的批处理文件编程技术, 给出了一组应用批处理文件实现的 DOS 实用例程, 这些例程包括文件快速定位, 不同参数下重复执行一个 DOS 命令, 文本文件名忘却后的模糊查找、目录遍历、方便实用的软盘格式化例程等。利用本文提供的这些 DOS 实用例程, 可大大提高计算机操作的效率, 增强 DOS 系统的功能。

关键词 DOS 实用例程 批处理文件 编程技术

IBM-PC DOS 中提供了一个功能很强的批文件命令, 由在 DOS 下可以运行的键盘命令组成。一个批处理文件包含有一个或多个 DOS 命令, DOS 一次执行其中的一个命令。此外, 批命令本身还包括 8 个子命令, 这些子命令具有一定的编程能力, 可控制程序流程, 使得批处理文件具有很强的处理能力。但由于很多参考书仅对批处理文件作一般性的介绍, 而对如何利用批处理文件开发一些巧妙实用的 DOS 例程则很少涉及, 在很大程度上影响了批处理文件的进一步应用。为此, 本文将应用一些鲜为人知或 DOS 手册上经常被忽略的批处理文件编程技术, 给出一组用批处理文件实现的 DOS 实用例程, 希望能给广大电脑用户一些启迪, 从而开发出更多更实用的 DOS 例程。

一、文件快速定位 DOS 2.0 及以后版本推出层次目录结构, 给用户带来很大的方便, 但同时也使得查找不在当前目录下的文件变得既复杂又费时。如要在硬盘上查找一个名为 TOOL.COM 的文件, 将不得不进入每个子目录, 并在每个子目录中用 DIR 进行查找, 这样既复杂又费时。本文通过使用一个充分利用 CHKDSK 命令执行结果的批处理文件, 开发出一个 DOS 实用例程, 在复杂的层次目录结构中方便快速地定位文件。下面给出这个 DOS 实用例程的程序清单。

```
程序清单一 LOCATE.BAT
ECHO OFF
IF "%1"="" GOTO HELP
ECHO FILE LOCATOR COMMENCING... PLEASE
WAIT...
```

```
CHKDSK /V >TEMP. $$$
FIND "%1"TEMP. $$$
ERASE TEMP. $$$
ECHO END OF SEARCH...
GOTO END
: HELP
ECHO USAGE : LOCATE FILENAME
ECHO EXAMPLE : LOCATE TOOL.COM
: END
```

该例程的调用格式是: LOCATE FILENAME←
。运行后例程将给出文件 FILENAME 所在的目录信息, 实现了文件的快速定位。

该批处理文件例程巧妙地应用了输出重定向技术, 把 CHKDSK 命令的执行结果送到临时文件 TEMP. \$\$\$ 中, 然后利用 FIND 命令的查找功能实现了文件的快速定位。

二、不同参数下重复执行一个 DOS 命令 有时我们需要用不同的参数一次又一次地重复运行同一个命令, 例如用 DIR 查找所有扩展名为 EXE、COM、BAT 的文件时, 必须依次键入 DIR *.EXE←、DIR *.COM←、DIR *.BAT←, 这很麻烦。下面提供的批处理文件例程 REPEAT.BAT 将较好地解决这个问题, 提高工作效率。批处理文件例程 REPEAT.BAT 运行时将以不同的参数重复执行 DOS 命令。在使用该批处理文件时, 需键入 REPEAT 及一个 DOS 命令和参数表。例如键入 REPEAT DIR *.EXE * .COM.BAT← 就可列出当前驱动器中所有扩展名为 EXE、COM 和 BAT 的文件。具体程序清单如下:

```
程序清单二 REPEAT.BAT
ECHO OFF
```

```

IF "%1"="GOTO INSTRUCT
IF "%2"="GOTO INSTRUCT
SET CMD=%1
: DOCMD
ECHO %CMD% %2
%CMD% %2
SHIFT
IF NOT "%2"="GOTO DOCMD
GOTO END
: INSTRUCT
ECHO ERROR : MISSING PARAMETERS
ECHO USAGE : REPEAT COMMAND PARAMETER
[PARAMETER....]
ECHO EXAMPLE : REPEAT DIR *.EXE *.COM *.
BAT
: END
SET CMD=

```

该批处理例程巧妙地应用了几项批处理文件编程技术。其一是利用临时 DOS 环境变量 CMD 来存放 DOS 命令;其二是利用批命令的子命令 SHIFT 来变换参数,以遍历参数表中的所有参数;其三程序结尾处的“SET CMD=”用来释放临时环境变量 CMD 的空间,以节省内存,这是批文件编程中容易被忽略的技术。

三、文本文件名忘却后的模糊查找 在实际工作中常遇到这样的情况,需要一个文件,却忘记了文件名,只是还能大概记得文件中的一些内容,如其中的特殊字符串等。这时只能一个文件一个文件地用 TYPE 命令进行查找,费时又费力。下面介绍的批处理文件例程 FINDCHAR.BAT 将较好地解决这个问题。程序清单如下:

```

程序清单三 FINDCHAR.BAT
ECHO OFF
IF "%1"="GOTO INSTRUCT
IF "%2"="GOTO INSTRUCT
IF %3STR=/SSTR GOTO SAVE
IF %3STR=/SSTR GOTO SAVE
ECHO 检索结果为:
FOR %%A IN(%1)DO FIND /N%2 %%A
GOTO END
: SAVE
ECHO 正在生成检索结果文件 TEMP. $$$
IF EXIST TEMP. $$$ DEL TEMP. $$$
FOR %%A IN(%1)DO FIND/N%2 %%A>>
TEMP. $$$
GOTO END
: INSTRUCT
ECHO 调用格式为:

```

```

ECHO FINDCHAR 文件名“字符串”[/S]
ECHO 参数/S 表示要生成检索结果文件 TEMP.
$$$
ECHO 例:FINDCHAR TG *.*“交通运输”
: END

```

例如要查找.TXT 文件中含有“交通运输”字样的文件,只需键入:

```
FIND *.TXT“交通运输”←
```

该批处理文件不仅能查到包含所查内容的文件名,而且能指出这段文字内容在该文件中出现的位置及全句文字。

四、目录遍历 使用 DOS 时,常常希望执行非当前目录下的程序或访问非当前目录下的文件,了解盘上有什么目录及其正确的拼写都极有利于利用 DOS CD 命令来改变到另一目录。下面提供的批处理文件 ML.BAT 将列出盘上的所有子目录。程序清单如下:

```

程序清单四 ML.BAT
ECHO OFF
CLS
IF "%1"="GOTO HELP
ECHO ALL DIRECTORIES ON %1 DISK ARE:
ECHO-----
TREE %1|FIND “\”|MORE
ECHO-----
GOTO END
: HELP
ECHO USAGE : ML A : (OR B : OR C :)
: END

```

五、方便实用的软盘格式化例程 现在有很多微机配置了两种尺寸的软盘驱动器,A 驱动器用 5 英寸磁盘,B 驱动器用 3 英寸磁盘。这样每次格式化软盘时,往往需要翻阅 DOS 手册查找合适的命令开关。这对用户来说是很麻烦的一件事。本文提供的批处理实用例程 FORMAT.BAT 将较好地解决这个问题。具体的程序清单如下:

```

程序清单五 FORMAT.BAT
ECHO OFF
REM YOU MUST RENAME DOS'S FORMAT.COM TO
FORMAT!.COM
IF "%1"="360"GOTO 360
IF "%1"="12"GOTO 12
IF "%1"="720"GOTO 720
IF "%1"="144" GOTO 144
: HELP
ECHO FORMAT 360 [/S][/V][/B] Will produce a
360KB disk in drive A :

```

```

ECHO FORMAT 12 [/S][V][B] Will produce a
1. 2MB disk in drive A :
ECHO FORMAT 720 [/S][V][B] Will produce a
720KB disk in drive B :
ECHO FORMAT 144 [/S][V][B] Will produce a
1. 44MB disk in drive B :
GOTO END
: 360
FORMAT|| A : /N : 09/T : 40 %2 %3 %4
GOTO END
: 12
FORMAT|| A : /N : 15/T : 80 %2 %3 %4
GOTO END
: 720
FORMAT|| B : /N : 09/T : 80 %2 %3 %4
GOTO END
: 144
FORMAT|| B : /N : 18/T : 80 %2 %3 %4

```

```

GOTO END
: END

```

注:在使用这个批处理文件例程时,必须将 DOS 中的格式化程序 FORMAT. COM 更名为 FORMAT||. COM。

其中,FORMAT 360 用在 A : 驱动器上格式化 360K 软盘;FORMAT 12 用在 A : 驱动器上格式化 1. 2M 软盘;FORMAT 720 用在 B : 驱动器上格式化 720K 软盘;FORMAT 144 用在 B : 驱动器上格式化 1. 44M 软盘。当然,原 DOS 格式化命令 FORMAT 中的几个可选命令行参数如/S、/V、/B 在本例程中同样可用。

由于篇幅所限,本文仅给出以上几个实用的批处理文件例程,这些例程均在 IBM 及其兼容机上调试通过。

引导扇区型病毒通用清除软件——KV—BOOT

肖俊良

广州中山大学计算中心(510275)

笔者经过深入分析引导扇区型病毒的传播机理、操作系统引导过程中的自举程序及 8086/8088 微处理器的汇编指令,编制了引导扇区(即操作系统)型病毒通用清除软件,改变了尾追病毒的被动局面。

本程序具有如下特点:

- (1)使用方便,语法为:KV—BOOT[drive:]
- (2)提示清晰
- (3)短小精悍,执行代码仅 5K
- (4)功能强

无论病毒把原主引导程序、DOS 引导程序转存到磁盘的何处,都能自动搜索并予以恢复。在原(主)引导程序遭到破坏的情况下,只清除病毒不一定可以使其恢复工作。

本程序可以自动检测病毒是否已经驻留内存。如果是,那么本程序将显示求救信号并软死机。用户必须用清洁无病毒的 DOS 系统重新启动机器。

本程序提供了保存及恢复硬盘主引导扇区的手段。恢复硬盘主引导扇区时必须保证分区参数,即硬盘容量、分区数及分区大小均相同,否则结果不可预料。其中,/S 用于保存硬盘主引导扇区,/L 用于恢复硬盘主引导扇区,无论是保存还是恢复,数据均来

自 A 盘的 MAINBOOT. DAT 文件。

出现信息“No or invalid partition Table”表明所检测磁盘的主引导扇区内的分区表的内容无效,随后将出现信息“Repair Partition Table”,如果用户选择“Y”,则此时程序将全部硬盘空间分配给一个分区使用。如果原来已经把硬盘划分为多个分区,请选择“N”,否则后果难以预料。

出现信息“Non—standard DOS Boot Program!”表明所检测磁盘的引导扇区的内容不是由 FORMAT. COM 生成的,可能有错,请选择“Y”删除,如果经此处理后磁盘变得不能启动,建议使用 DOS 命令 SYS. COM 恢复。

出现信息“Virus may be in disk”表明所检测磁盘的引导扇区的内容值得怀疑,可选择“Y”回答,如果经此处理后磁盘变得不能启动,建议使用 DOS 命令 SYS. COM 或程序 NDD. EXE、DISKFIX. EXE 恢复。

本程序还可自动清除由病毒程序占用的虚假的“坏”簇,但下列情况可能例外:

- (1)不是可以自举的分区;
- (2)DOS 分区的大小超过 32MB。

欢迎提出宝贵意见。

需此软件者,请同作者联系。

微型机在医院病历资料管理中的应用

向瑞红 唐 承

四川省财政学校微机室(成都,610041) 航空航天部719厂设计所(成都,610055)

摘要 本文介绍一种医院病历资料微机管理软件,它能够实现对病历资料的登记、查询、修改、删除和统计,为分析各类病员、各种疾病之间的比例和分布、医生的诊断水平,制定相应措施,提供了一种科学依据和手段。

关键词 微型机 病历资料 统计管理

一、前言 医院的病历资料,不仅对病员,而且对医生来说都是十分重要的,它能协助医生了解病员的病情,作出正确的诊断,同时又能反映医生的诊断质量。而对病历资料作出不同时期、各年龄段、各类患者的统计,则能帮助医务工作者分析各种病因,及时制定出治疗和预防措施。因而,对医院病历资料的管理就显得十分必要,但面对大量的病历资料,要能快速、准确地完成上述管理工作,对于人工来说是相当困难的。随着微机在各个领域的广泛应用,笔者

在 IBMPC/XT 微机上开发出了这套医院病历资料管理系统,使得这一复杂的管理工作变得简单易行了。

二、功能及使用 本系统采用模块化结构,具有病历登记、病历查询,病历修改、病历删除和病历统计输出等功能。在设计上采用多层次菜单显示,充分利用屏幕的人机对话,使用者只需掌握简单的计算机知识即可方便地操作。本系统程序由 CDBASE II 编写而成,系统程序流程框图如图 1 所示。

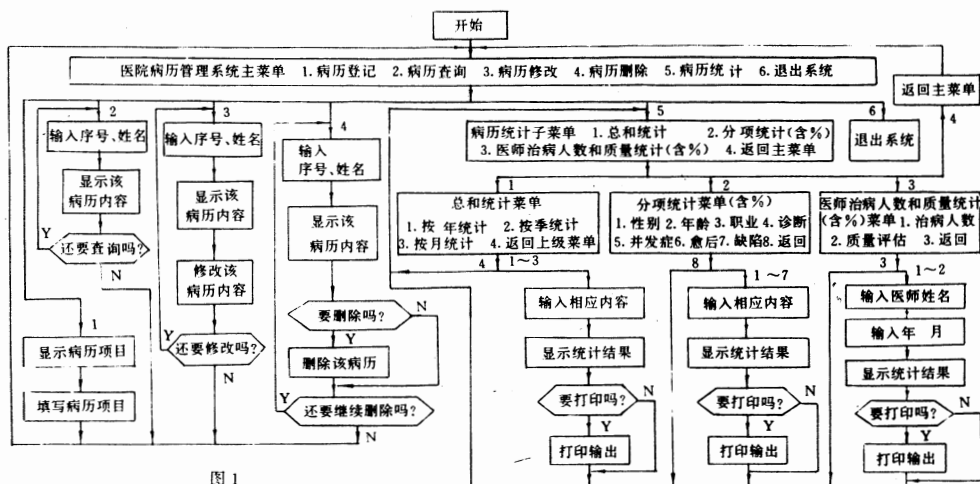


图 1

现将各功能模块分述如下:

1. 主控模块 在 CCDOS 和 CDBASE II 支持下,启动本系统,便可进入系统的主控模块,显示出系统

主菜单,它们是:1、病历登记;2、病历查询;3、病历修改;4、病历删除;5、病历统计;6、退出系统。按下相应的数字键,系统自动进入相应的功能模块。

2. 病历登记模块 本模块可根据屏幕提示,使用全屏幕编辑键方便地对病历资料进行登记。其病历资料由如下栏目组成:

序号、住院号、姓名、性别、年龄、入院时间、出院时间、住院天数、职业、体温、脉搏(次/分)、呼吸(次/分)、收缩压、舒张压、诊断1、诊断2、并发症、愈后、型别、缺陷、质量判定和医师。

其中,诊断1选择填写的内容是:呼、循、泌、血、内、代、结、免、理、神、精、传和其它等系统疾病。诊断2填写的内容是:各系统疾病下的具体疾病。愈后选择填写的内容是:3(治愈)、4(显效)、5(好转)、6(缓解)、7(无效)、8(加重)、9(残废)和10(死亡)。型别选择填写的内容是:5(a)、4(b)、3(c)和2(d)。缺陷选择填写的内容是:1、2、3、4和5。而质量判定则是根据:质量判定 = $(1 - \text{愈后} \times \text{型别} \times \text{缺陷} / 250) \times 100$ 公式,由计算机自动填写,分数高者,则说明医师医术水平高,反之,则医师医术水平低。

3. 病历查询模块 本模块用于查询病员的病历,了解基本状况,为医务人员和病员带来方便。进入本模块后,根据屏幕提示,正确输入要查询的是病员病历序号、姓名后,屏幕上便显示出该病员病历上的全部内容。若需要继续查询其他病员病历,正确输入其序号、姓名即可,否则返回主菜单。

4. 病历修改模块 本模块用于病员病历的修改更新。在登记病员病历的过程中,难免不出差错,一旦出现错误,就可以通过本模块进行修改更正。同查询模块一样,正确输入病员病历的序号、姓名后,屏幕上显示出该病员病历的内容,用全屏幕编辑键进行修改,修改完成后内容存盘,若需要修改其他病历,输入其序号、姓名即可,否则返回主菜单。

5. 病历删除模块 对于一个医院来说,病员病历资料的数量是相当大的,定期对其进行整理、删除一些没有保存价值的资料是必要的,这样能够提高

计算机的工作效率。本模块的工作过程与修改模块相同,按照屏幕提示即可完成删除。

6. 病历统计模块 本模块是本系统软件最具特色、管理功能最强的模块。它可以按年、季度、月对病历资料进行综合统计;对病历中的性别、年龄、职业、各类系统疾病、并发症、愈后、缺陷等情况按年月进行分项统计,并求出其占总数的百分比;对医生的工作情况作出评估,统计出医师某年某月收治病人数、占某年某月收治病人总数的百分比以及某年某月质量判定总数和平均数,并可根据需要用打印机输出其统计表。根据这些统计资料,可以分析出各类病员、各种系统疾病之间的比例和分布、医师的工作态度和医术水平,为管理决策人员提供科学依据和手段。进入本功能模块后,屏幕上出现病历资料统计菜单:1、综合统计;2、分项统计(含%);3、医师治疗病人数和质量统计(含%);4、返回主菜单。按下相应的数字键,屏幕上将出现相应的下一级子菜单,按照屏幕提示,完成相应的统计功能。统计完成后,返回上一级菜单,各类统计全部完成后,返回主菜单。

三. 结束语 本医院病历资料管理系统是为成都市第二人民医院急诊部设计的,它同样能适用于各类医院的病历资料管理。目前,IBM-PC/XT微型机及其兼容机的价格均在数千元以内,配上这套管理软件,投资不多,却能起到事半功倍之作用。需此软件者,请同作者联系。

参考文献

- 1、敬喜等,《DBASE III 新一代关系型数据库管理系统》,清华大学出版社,1986
- 2、詹前树等,《微型计算机 IBM PC/XT 常用软件上机操作与实践》,中山大学出版社,1990

(上接 21 页)条件,供读者参考:

(1)存放变长数据项的字段定义较长,库结构中应至少存在一个变长数据项,并且其字段长度应大于其实际存储数据的平均长度;或者

(2)原始数据记录数量大于记录中包含数据项的数量,并且记录中所有存放变长数据项的字段定义长度之和较大。

五、结语 关系式数据库应用环境下的类 HASH 编码技术及其数据的顺序紧缩存储方法以较小的运行效率代价换取了高效的存储空间,它可以明显地改善微型机的存储性能,这对微型机在文字信息检索与管理方面的应用是一个有力的帮助。所介绍的编码/解码以及存储运算由于在运行时对用

户透明,因此,在使用时不会感到有任何困难。本文方法不会对用户其他应用程序模块的并入产生任何影响;它可以用作 MIS 或 OA 领域中的一种通用的文字信息程序设计方法和数据压缩工具。此外,它对于各种汉字库系统的设计在方法上具有一定的借鉴作用。

参考文献

- [1] 严蔚敏,吴伟民,《数据结构》,清华大学出版社,1987。
- [2] N. Wirth, Algorithms + Data Structure = Programs, by Prentice-Hall, Inc., 1976。
- [3] 谭强明,《HASH 编码数据压缩》,《计算机应用与软件》,7(4),1990。

IBM-PC/XT 与 MCS-96 系列单片机并行通讯系统的设计

王杉杉 胡继胜 马修真

大连铁道学院 (116001)

摘要 IBM-PC/XT 和 MCS-96 系列单片机可以组成一个高性能的分布式检测控制系统。本文介绍了其中用 8255A 并行接口实现主机与前沿机数据通讯的方法,该方法实用性强,可靠性高,对大量双向数据通讯更具意义。

关键词 MCS-96 单片机 并行通讯 接口电路

一、前言 随着计算技术的飞速发展,计算机的应用愈来愈广泛。特别是单片机,由于其体积小,成本低,可靠性高和易于扩展等特点,在工业控制和检测领域取得了惊人的发展。本文针对参变量多,通讯数据量大,并要求检测具有同步实时性的检控对象,设计组成了以 IBM-PC/XT 为主机,多片 8097BH 单片机为前沿机的分布式检控系统。前沿机通过并行接口与主机相联,用户只需要通过主机向前沿机发出命令,前沿机就可完成对参变量的检测和预处理,并根据主机的要求传输大量数据,以便主机后处理分析、输出结果。由于主机与前沿机通讯采用的是由 8255A 可编程通用并行接口芯片组成的接口接路,提高了数据的传输速度和准确性,并且具有较强的抗干扰能力,因此系统能够保证在较复杂的应用环境下可靠准确地工作。检控系统结构框图如图 1 所示。

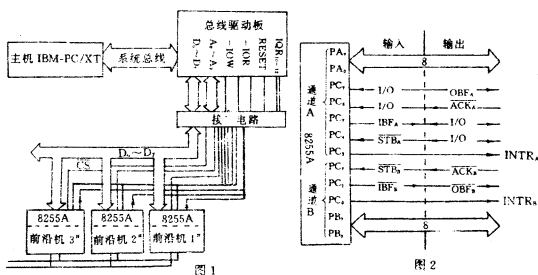


图 1

编程并行接口芯片具有三个 8 位 I/O 口: A 口、B 口和 C 口;并具有三种操作方式。本系统根据二机之间的通讯要求,选择了一种具有握手信号的选通输入输出方式,即方式 1; A 口和 B 口分别可定义为方式 1 下的输入或输出,这时 C 口作为联络信号口。

1. 前沿机与 8255A 通讯接口电路的设计

根据系统选用的前沿机 8097BH 的结构性能特点,该机共有五个 8 位 I/O 口, P_0 是只用于输入的; P_1 是准双向 I/O 口; P_2 是多功能复用口; P_3 和 P_4 可作为双向口和系统总线。我们在对 8097BH 单片机进行存贮器和外围电路的硬件扩展时,已确定 P_3 和 P_4 口作为前沿机用户系统的系统总线, P_1 口作为与 8255A 数据通讯的输入输出,而 P_0 口又与 A/D 转换器的模拟输入端共享引脚;作为多功能复用口的 P_2 ,由于 $P_{2.0}$ 、 $P_{2.1}$ 用于前沿机之间串行通讯的输入输出引线,而 $P_{2.3}$ 、 $P_{2.4}$ 和 $P_{2.5}$ 又被用户系统硬件所享用,所以只有 $P_{2.6}$ 和 $P_{2.7}$ 二个准双向口线可以设计作为前沿机与 8255A 通讯的联络信号线。由图 2 看出,在方式 1 下, A 口的联络信号线要比 B 口多出二条,因此在接口电路设计时,选用了与 A 口功能相同的 B 口作为数据双向通讯口。由 8097BH 单片机编程工作和前沿机所要执行的检控功能受控于主机的特点,决定了前沿机应以响应主机中断请求的方式工作;主机经 8255A 的 PC_4 口向前沿机发出的前沿上升的电平信号,根据 8097BH 单片机中断源的形式,该信号应接到 $P_{2.2}$ 引脚上作为前沿机的外部中断请求信号。接口电路的联线由图 3 右边所示。

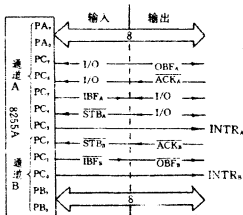


图 2

二、硬件设计 主机与前沿机是通过由 8255A 接口芯片所组成的电路实现通讯的。Intel 8255A 可

由于 8097BH 单片机的 P_1 口及 $P_{2.6}$ 和 $P_{2.7}$ 口均为准双向口,内部具有上拉电阻结构,在作为输入口使用时,应首先将该口拉到高电平。考虑到用改进外部驱动电路的设计来实现口线电平的上拉会造成 8097BH 芯片瞬态功耗增加,线路元件增多,所以在

系统中利用软件的改进解决了这个问题。

2、主机与 8255A 通讯接口电路的设计

主机与 8255A 接口芯片的联接是通过插在主机系统板上总线槽中的总线驱动电路和逻辑接口电路实现的。根据分布式系统所要求完成的任务和硬件环境的要求,主机与 8255A 之间的数据通讯是以硬中断、查询方式进行的。主机 PC/XT 的硬中断源是由主从二片 8259A 可编程中断控制器构成,为系统提供 16 级中断;在这 16 级中断中为用户在从片上保留了 IQR₁₀、IQR₁₁、IQR₁₂ 三个中断源,其中断类型码分别是 72、73 和 74。根据检控对象和检测参数的数量、同时性的要求,决定系统需要三台前沿机才能完成检控任务,主机所保留的三个硬中断源已能满足系统硬件的要求;若再增加前沿机或使用硬中断的外设,只需在原硬中断结构的基础上,再扩充一片 8259A 芯片即可增加 8 级中断。

并行接口芯片 8255A 共有四个通道地址,分别是 A、B、C 通道和控制寄存器地址;A₀、A₁、 \overline{CS} 、 \overline{RD} 和 \overline{WR} 这五个信号配合使用可对这四个通道地址进行寻址并作不同的操作。主机 80286CPU 具有 24 位地址线和 16 位的 I/O 地址线(A₀~A₁₅),并通过系统板上 I/O 总线插槽作总线延伸;CPU 在利用 I/O 口地址对外设进行寻址访问时,只用 16 位(A₀~A₁₅) I/O 地址线中的 10 位(A₀~A₉)对外设接口编址,高位地址信号 A₁₆~A₂₃ 均为 0。由 IBM-PC/XT 微机中的 I/O 地址分配表可看出,系统板上使用了口地址 00H~FFH,余下的口地址分配在 I/O 通道中使用。根据主机 I/O 通道地址的分配,结合 8255A 接口芯片控制信号的要求,主机与不同的前沿机进行通讯的 8255A 接口芯片通道地址分别设计为:1 号前沿机,3C0H~3CBH;2 号前沿机,3C4H~3C7H;3 号前沿机,3C8H~3CBH。

I/O 口地址分配表:

I/O 口地址	设备名称	I/O 口地址	设备名称
200H~20FH	游戏 I/O 口	378~37FH	并行打印口 1
210H~217H	扩展部件	380H~38FH	SDLC, 双同步 2
220H~24FH	保留	3A0H~3AFH	双同步 1
278H~27FH	并行打印口 2	3B0H~3BFH	单显和打印适配器
2F0H~2F7H	保留	3C0H~3CFH	保留
2F8H~2FFH	串行口 2	3D0H~3DFH	彩色/图形适配器
300H~31FH	样卡	3F0H~3F7H	软磁盘控制器
320H~32FH	硬盘适配器	3F8H~3FFH	串行口 1

根据 8255A 接口芯片各通道地址,通过由 74LS138 等芯片组成的译码电路,对被选中的 I/O 地址进行译码,可产生一个 \overline{CS} 信号, \overline{CS} 信号与主机

的 \overline{IOR} 、 \overline{IOW} 及 I/O 地址的低二位 A₀、A₁ 共同作用,来对 8255A 的通道进行控制和操作。由于本系统设计的主机所访问的 I/O 口地址范围在 3C0H~3CBH 之间,I/O 地址中的高五位(A₉~A₅)均为 11110,所以对不同的 8255A 接口芯片选通工作只取决于地址中 A₂、A₃ 和 A₄ 的组合。接口电路的联线图 3 左边所示。

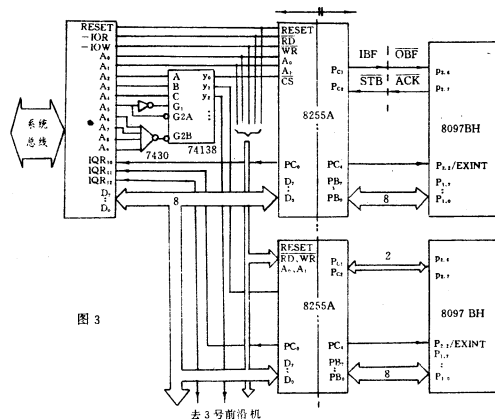


图 3

图 3、并行通讯接口电路图

三、软件设计 在分布式系统中,主机通过上述设计的接口电路与前沿机进行双向数据通讯,主机与前沿机通讯软件的主要功用是完成主机对前沿机工作选择的控制及前沿机调用功能子程序控制指令的传递;同时接收前沿机工作后传回的预处理过的数据。根据主机 CPU 的特点以及通讯程序所要求的数据。根据主机 CPU 的特点以及通讯程序所要求的数据。根据主机 CPU 的特点以及通讯程序所要求的数据。采用 8096 汇编语言编写。

根据并行接口电路的硬件设计,结合主机及前沿机的工作环境和特点,主机和前沿机通讯软件均采用中断、查询的工作方式。主机通讯程序响应的是由 8255A 产生的硬中断请求信号,为了克服由于外部干扰及内部中断优先权的差异造成的程序执行错误,保证通讯的连续性,在通讯程序执行到输入输出中断传输前,保存系统中断向量并关闭除工作的前沿机所用中断源外的所有中断源,在通讯结束后再恢复中断控制器的状态。前沿机通讯程序响应的是由主机经 8255AC 口发出的外部中断请求信号,由于 8097BH 单片机部中断优先权最高,因此只要前沿机履行与 8255A 接口通讯的握手协议,就可保证

数据通讯的顺利进行。

1、主机到前沿机通讯软件的设计

在分布式系统中,主机要对多个前沿机进行控制,因此高级语言调用通讯程序时,首先把那个前沿机工作的控制指令传递给通讯程序,主机通讯程序根据控制指令选取对应的 I/O 口地址,经所设计的接口电路译码后对工作机作出选择。对于前沿机,其通讯程序根据 8097BH 单片机结构特点,在启动或复位后,一些寄存器被初始化,消除 PSW,把 PC 寄存器的值置为 2080H,从 2080H 单元开始向下执行,在完成建立堆栈指针,定义复用口 P_{2.2},置中断服务程序中中断向量后,开中断等待由主机经 8255A 的 PC₄ 口发出的外部中断请求信号。下面根据 8255A 方式 1 输出时序及硬件接口连线介绍主机到前沿机的通讯过程:首先主机通讯程序对工作的 8255A 进行初始化,并通过 PC₄ 口(高有效)向前沿机 CPU 发外部中断请求信号。前沿机通讯程序进入了中断服务程序,便发出 ACK 响应信号(脉冲),使 8255A 输出的 $\overline{\text{OBF}}$ 信号无效、INTRB 有效变高,向主机 CPU 发出中断请求,此时前沿机判断 $\overline{\text{OBF}}$ 是否有效等待接受数据。主机 CPU 响应中断进入输出中断服务程序,判断 $\overline{\text{OBF}}$ 是否有效,若无效则 CPU 用输出指令输出数据,发出 $\overline{\text{WR}}$ 信号,把数据送到外部数据总线上同时使 $\overline{\text{OBF}}$ 有效,INTRB 无效变低,中断返回。前沿机 CPU 在获得 $\overline{\text{OBF}}$ 有效信号后,从总线上通过 P₁ 口读到数据并送指定单元贮存,随后又产生一个 ACK 信号通知主机 CPU 数据已取走,CPU 响应中断又开始新数据的输出过程。通过对通讯过程的分析,设计了主机到前沿机的通讯软件,程序框图见图 4 上半部。

2、前沿机到主机的软件设计

前沿机在接收到主机输出的数据后,根据主机的指示完成相应的检测工作,并把检测到的数据预处理后传输给主机。其通讯过程是:前沿机完成功能调用后发出 STB 信号,通过 P₁ 口把数据送到总线上,STB 有效 8255A 把总线上数据锁存进通道,并自动使 IBF 有效、INTRB 有效,向主机 CPU 发出中断请求。主机 CPU 响应中断进入输入中断服务程序,判断 IBF 是否有效,无效等待,有效主机 CPU 则用输入指令读取数据,并发出 $\overline{\text{RD}}$ 信号使 INTRB 无效、IBF 变低无效。前沿机在得到 IBF 无效信号时,才能再送

出一个数据,否则在此判断等待,前沿机到主机通讯程序框图见图 4 下半部。

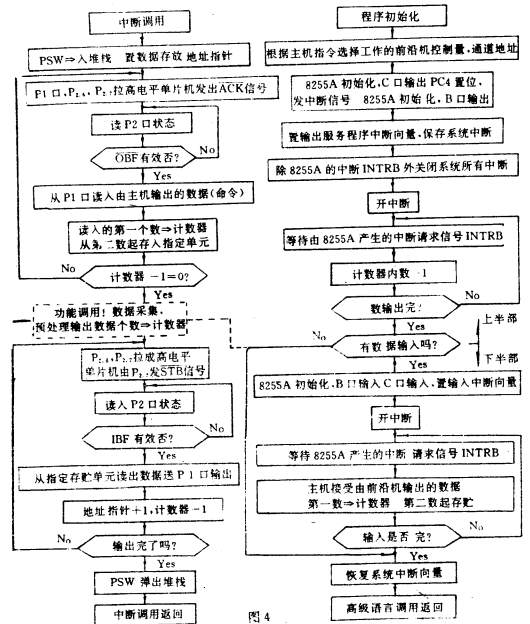


图 4

图 4 并行双向通讯程序框图

四、结束语 由主机和多个前沿机组成的分布式检控系统已经应用于机车柴油机性能参数的检测分析和对执行机构的控制,理论调试和现场试验,应用都证明了采用并行接口实现主机与前沿机的通讯是可行的,并具有易扩展、可靠性高的优势。

参考文献

- [1]周明德编,《微型计算机接口电路及应用》,清华大学出版社,1991年2月
- [2]朱传乃编,《80286 微机系统分析》,北京中科院计算所,1988年12月
- [3]周少柏译,《80286 汇编语言》,北京中科院希望电脑公司,1988年5月
- [4]徐君毅编,《单片微型计算机原理与应用》,上海科学技术出版社,1988年4月
- [5]孙涵芳编,《MCS-96 系列 16 位单片微型计算机》,北京航空航天大学出版社,1989年1月。

敬告作者

现本刊欢迎以下内容的稿件:①硬件设计经验、技巧;②主机、外设、系统、工作站、网络的维护维修;③多媒体计算机的开发应用;④各种单片机的开发应用;⑤各种汉卡、图形处理卡的应用技巧;⑥各种接口的设计、应用;⑦工作站的开发应用;⑧计算机开发应用的新动态、趋势。稿件宜用激光照排,标明字数,要求一稿一投。稿酬从优。

点阵式液晶显示器与单片机的接口

赵广涛

盛翊智

吕汉兴

华中理工大学自控系

(武汉, 430074)

摘要 本文介绍了笔者设计的点阵式液晶显示器(LCD)与单片机的接口电路,并介绍了显示自制字符的方法,最后给出了一实例。

关键词 点阵式液晶显示器 单片机 接口

一、引言 液晶显示器分段位式和点阵式。段位式 LCD 已广泛应用在电子表、计算器等上面,人们对它并不陌生,这种显示器显示的字符有限,应用因而受到限制。点阵式 LCD 是显示器中的后起之秀,它能显示更丰富的信息(包括部分汉字),能进行人——机对话,具有很强的功能。在国外,点阵式 LCD 已应用于便携计算机、文字处理机、办公自动化,直至大屏幕彩电。目前国内许多用户迫切需要了解如何使用点阵式 LCD,而这方面的资料却并不多见。笔者根据现有资料和使用中摸索的一点经验,总结出一种使用点阵式 LCD 的方法。

本文以 DMC40128LCD 为例。DMC 系列 LCD 的引脚及功能和指令系统详见文献[1]。

二、接口电路 文献[1]介绍了 LCD 与 80C31 单片机通过总线直接接口的方法,本文介绍一种 LCD 通过接口芯片 8255A 与单片机 8031AH 接口的电路。这种接口方法几乎适用于各种系列的微处理器和并行口扩展芯片,因此应用价值很高。

首先得注意 LCD 的“读”、“写”操作时序,如图一、图二所示。读 LCD 时,使能信号 E 有效后,有效数据才产生;写 LCD 时,有效数据必须在 DB 总线上稳定后,使能信号 E 才有效。LCD 的操作速度在 NS 级,足以与任何高速微处理器相配,而一般外围接口芯片的速度稍低一点,故当 LCD 通过外围接口芯片与微处理器接口时,不得不考虑到数据传输延迟。图三所示的接口电路正是基于上述考虑设计的。

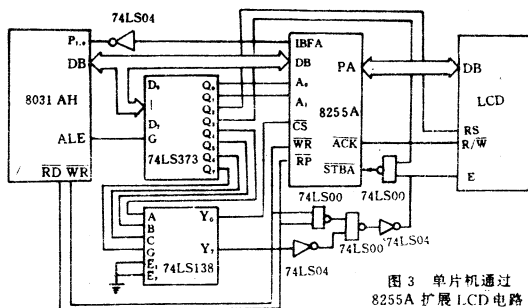


图3 单片机通过8255A扩展LCD电路

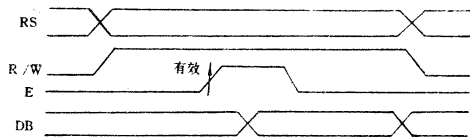


图1 “读”LCD操作时序

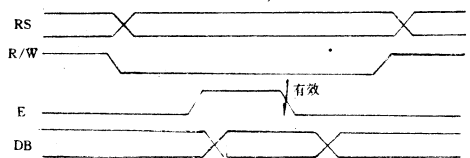


图2 “写”LCD操作时序

选择 8255A 工作于方式 2,这时 PA 口是双向方式,PC 口的高 4 位作为 PA 口的控制作用。具体是:

PC4,STBA,选通输入,输入低电平时将 PA 口数据打入 A 口锁存器。

PC5,IBFA,输入缓冲器满,输出为 1 表示数据已被打入输入缓冲器。

PC6,ACK,响应信号,输入低电平,开启 PA 口的三态输出缓冲器。

PC7,OFB,输出缓冲器满,输出低电平表示 CPU 已把数据写入 PA 口。

根据图三电路可以确定各口地址如下:

8255A: PA 口地址 80H, PB 口地址 81H,

PC 口地址 82H, 控制口地址 83H。

LCD: 写指令口地址 0F0H,

写数据口地址 0F4H,

读忙口地址 0F8H,

读数据口地址 0FCH。

下面简要分析一下电路的工作原理:

1、“读”操作过程: 单片机先发出 LCD 的口地址, 经 74LS373 锁存, Y7、RS、R/W 信号有效。单片机继而发出 RD 信号, RD 与 Y7 组合后作为 LCD 的使能信号 E。同时 E 与 R/W 信号“与非”后作输入选通信号 STBA, 将 LCD 发出的数据锁入 PA 口的锁存器。这时 IBFA 信号被置为高电平“1”, 单片机检测到 IBFA 为“1”, 再选通芯片 8255A 的 PA 口, 将数据读进单片机。

2、“写”操作过程: 单片机先选通 8255A 的 PA 口, 将要写入 LCD 的数据(或指令)先写入 PA 口, 再选通 LCD 的口地址, 同样 RS、R/W 信号先有效, R/W 信号同时作为 ACK 信号, 触发 PA 口放出数据(或指令), 当单片机发出 WR 信号时, E 有效, 而 LCD 的 DB 上数据已稳定, 于是被写入 LCD。

由以上分析可知, 单片机对 LCD 无论是“读”还是“写”, 都是分两步进行的, 第一步都是先将数据锁入 8255A 的 PA 口, 第二步才最后完成“读”或“写”操作。这种两步操作方式正好与 LCD 的时序相配合, 但需要数条指令完成, 操作时间长, 浪费了 CPU 时间, 然而却有很高的可靠性。

三、LCD 显示自制字符的方法 LCD 的字符显示分标准 ASCII 字符和自制字符, 显示标准 ASCII 码字符较简单, 只要在显示 RAM 的相应单元写入其 ASCII 码, 显示 RAM 起始地址为 80H, 而显示后者则在显示 RAM 相应单元写入自制字符代号, 字符代号为 0~7。

LCD 的字符发生器(CGRAM)共有 256 个单元, 其中 192 个单元已经固化了一些常用字符, 还剩 64 个单元让用户来自编字符。因为每编制一个 5×7 点阵字符需要 8 个单元存放数据(包括光标行), 故一次最多只能编制 8 个字符, CGRAM 的起始地址是 40H, 编制字符的代码就存放在 40H 开始的 64 个单元中。

下面以编制汉字“上”为例来说明如何编制字符。5×7 点阵每行有 5 个点, 对应于一个字节的低 5 位, 如果某点“亮”, 则对应位为 1, 而不亮位和高三位均作 0 处理。如图四所示, 一行点阵用一个字节表示, 故一个字符需要 8 个字节(光标算一行)。

	高三位	低	五	位	代码
第一行	0 0 0	0	0	1	0 0 0 4 H
第二行	0 0 0	0	0	1	0 0 0 4 H
第三行	0 0 0	0	0	1	0 0 0 4 H
第四行	0 0 0	0	0	1	1 1 0 7 H
第五行	0 0 0	0	0	1	0 0 0 4 H
第六行	0 0 0	0	0	1	0 0 0 4 H
第七行	0 0 0	1	1	1	1 1 1 F H
第八行	0 0 0	0	0	0	0 0 0 0 H
		↑	↑	↑	↑
		第	第	第	第
		1	2	3	4 5
		列	列	列	列

图四 汉字“上”的点阵图

假如把图四中右边表示“上”的 8 个字节代码依次写入 CGRAM 的 40~47H, 显示时, 先设定字符显示位置, 即确定在显示 RAM(DDRAM)的地址, 再将字符代号写入该 DDRAM 单元, 就可显示出自制字符。其中写入 DDRAM 的字符代号的低 3 位表示字符号, 高 4 位一定为 0, 因此 08H 与 00H 都是指第一个字符。虽然每次只能最多编制 8 个字符, 但一次可显示多个同样的字符。64 个 CGRAM 单元分配如表 1 所示:

表 1 CGRAM 单元分配表

字 符 号	CGRAM
0	40—47H
1	48—4FH
2	50—57H
3	58—5FH
4	60—67H
5	68—6FH
6	70—77H
7	78—7FH

四、应用举例 结合以上内容给出一个应用实例, 以进一步说明 LCD 的使用方法。

例: 显示“华中工大 celebrate the 43nd anniversary of the founding of CHINA”

分析: (1) 对 LCD 进行初始化。

(2) 必须在坐标纸上造出汉字“华中工大”, 确定其代码, 并写入到 CGRAM。

(3) 将字符 ASCII 码或字符写入到
DDRAM 中。

详细程序及注解见下述程序清单。

```

ORG      2100H
START: MOV    R0, #83H      ;8255 控制口地址
        MOV    A, #0C0H     ;8225 控制字
        MOVX   @R0,A        ;
        MOV    R0, #0F0H    ;LCD 指令口地址
        MOV    A, #38H      ;LCD 模式设置
        ACALL  WRITE        ;
        MOV    A, #01H      ;清 LCD
        ACALL  WRITE        ;

        MOV    A, #0CH      ;显示方式
        ACALL  WRITE        ;
        MOV    A, #06H      ;
        ACALL  WRITE        ;
        MOV    R1, #40H     ;CGRAM 首地址
        MOV    R7, #20H     ;字符代码个数
        MOV    DPTR, #2200H ;字符代码存放首地址
DD:     MOV    R0, #0F0H    ;LCD 指令口地址
        MOV    A, R1        ;
        ACALL  WRITE        ;先写 CGRAM 地址
        MOVX   A, @DPTR     ;
        MOV    在 R0, #0F4H ;LCD 数据口地址
        ACALL  WRITE        ;再写字符代码
        INC    DPTR         ;地址加 1
        INC    R1           ;
        DJNZ   R7, DD       ;
        MOV    R0, #0F0H    ;
        MOV    A, #80H      ;DDRAM 首地址
        ACALL  WRITE        ;
        MOV    R7, #28H     ;上行显示字符数
        MOV    R0, #0F4H    ;
AS:     MOVX   A, @DPTR     ;
        ACALL  WRITE        ;
        INC    DPTR         ;指针加 1
        DJNZ   R7, AS       ;
        MOV    R0, #0F0H    ;
        MOV    A, #0C0H     ;下行 DDRAM 首地址
        ACALL  WRITE        ;
        MOV    R7, #28H     ;下行字符数
        MOV    R0, #0F4H    ;
AS1:    MOVX   A, @DPTR     ;
        ACALL  WRITE        ;
        INC    DPTR         ;
        DJNZ   P7, AS1      ;
        AJMP   START        ;

```

```

WRITE:  ACALL  CBUSY        ;调查忙子程序
        XCH    A, R0        ;
        PUSH   A            ;保护 R0
        XCH    A, R0        ;
        MOV    R0, #80H     ;8255A 口地址
        MOVX   @R0,A        ;先进 PA 口
        XCH    A, R0        ;
        POP    A            ;
        XCH    A, R0        ;恢复 R0
        MOVX   @R0,A        ;写进 LCD
        RET                     ;
CBUSY:  PUSH   A            ;保护 A
        MOV    A, R0        ;
        PUSH   A            ;保护 R0
AGA0:   MOV    R0, #0F8H    ;LCD 查忙口地址
AGA:    MOVX   A, @R0        ;
        JB     P1.0, AGA     ;读进 8255 否
        MOV    R0, #80H     ;
        MOVX   A, @R0        ;
        RLC    A            ;
        JNC    AGA0         ;忙否
        POP    A            ;
        MOV    R0, A        ;
        POP    A            ;
        RET                     ;
ORG      2200H
        DB     0DH, 16H, 14H, 17H, 04H, 1FH, 04H, 00H ;字符代码
        DB     04H, 04H, 1FH, 15H, 1FH, 04H, 04H, 00H ;
        DB     00H, 1FH, 04H, 04H, 04H, 1FH, 00H, 00H ;
        DB     04H, 04H, 1FH, 04H, 04H, 0AH, 11H, 00H ;
        DB     00H, 01H, 02H, 03H, 20H, 63H, 65H, 6CH ;显示字符
        DB     65H, 62H, 72H, 61H, 74H, 65H, 20H, 74H ;
        DB     68H, 65H, 20H, 34H, 32H, 6EH, 64H, 20H ;
        DB     61H, 6EH, 6EH, 6AH, 76H, 65H, 72H, 73H ;
        DB     61H, 72H, 79H, 20H, 6FH, 66H, 20H, 20H ;
        DB     74H, 68H, 65H, 66H, 6FH, 75H, 6EH, 64H ;
        DB     69H, 6EH, 67H, 20H, 6FH, 66H, 20H, 74H ;
        DB     68H, 65H, 20H, 44H, 68H, 69H, 6EH, 61H ;
        DB     20H, 20H, 20H, 20H, 20H, 20H, 20H, 20H ;
        DB     20H, 20H, 20H, 20H, 20H, 20H, 20H, 20H ;

```

五、结语:文中设计的点阵式 LCD 与单片机的接口方法已成功应用于我们研制的智能仪器中。

参考文献

- [1]陈平,王靖,“字符式液晶显示器的应用”,《电子技术应用》,1991年第6期。
- [2]育乐,“精巧的液晶显示组件(一)”,《无线电技术》,香港,1991年215期。
- [3]盛翊智、余治安,《单片机及其应用系统》,湖北科学技术出版社,1988年3月。

对 IBM AS/400 系统没有给出 MIPS 值的几点看法

李春贵 刘铁军 马俊田 赵克
吉林省计算机技术研究所 吉林省科委(长春,130012)

摘要 本文从用户需求的各个不同侧面较详尽地论述了如何公正、全面地评估系统性能指标。

关键词 MIPS 响应时间 吞吐量 有效用户数 系统性能

AS/400 是美国 IBM 公司于 1988 年 6 月推出的面向事务处理的中小型计算机系统。这个系统问世,就在事务处理应用领域的激烈竞争中取得了令人瞩目的优势,迅速成为该领域范围内公认的优秀机型。仅 4 年时间,IBM AS/400 已更新了 4 代产品,从最早的 B 系列机发展到今天的 E 系列机。在国内(除台湾外),AS/400 机用户达数百家,在银行系统、财政系统、税务系统、保险业务系统、大中小型企业、学校、酒店、图书馆等众多领域中迅速广泛地得到了应用。

但是,在 IBM AS/400 的性能指标上曾经存在一些争论,在 IBM 所公布的 AS/400 系统的各项性能指标中没有包括目前国际上其它厂商所广泛采用、而且国内用户也普遍关心的每秒百万条指令数(MIPS)指标是引起这种争论的原因,有人据此怀疑 AS/400 系统的速度不够快,有的文章甚至直接批评 AS/400 系统的 MIPS 值低。其实,IBM 之所以不用 MIPS 值来衡量 AS/400 系统的性能,我们理解有如下原因:

一、不同的计算机厂商所生产的计算机系统往往都侧重面向某一方面的用户需求,该方面的性能指标可能很高,而其它方面的指标可能不是很高。一个各方面性能指标都特别高的计算机系统,其造价肯定会非常高,用户不一定接受。大多数用户在选择计算机系统时,主要考虑的是自己所关心的那些方面的性能要求。如面向科学计算的用户往往关心计算机的运算速度,如 MIPS 值、MFLOPS 值,而面向事务处理的用户,往往关心计算机的数据处理能力,而不是 MIPS 值。

二、众所周知,MIPS 描述的是处理机执行单条指令的速度,它只能体现系统的运算速度,特别是对复杂的科学计算,如处理上百阶矩阵、叠代,MIPS 值高是很有意义的。所以,MIPS 值确实曾作为衡量计算机运算速度快慢的一个指标而被广泛采用。但对于诸如事务处理,如信息检索,频繁地存取访问磁盘

数据,大量的排序查询检索数据文件等操作,MIPS 值就不能完全反映系统在这方面的性能。

IBM 认为,在对事务处理计算机系统的性能进行评价时,MIPS 不是一种适用的指标。IBM 公司在设计 AS/400 该项产品的硬件和软件时,认为从终端用户的角度来说,用下述三项指标来测量一个面向事务处理的计算机系统的性能是要比 MIPS 值更有实际意义。

1. 响应时间(Response Time):是从工作站用户角度所看到的,系统在完成一项交互式任务时所花费的平均时间,通常用秒来测量。响应时间小于等于 5 秒可以认为是合理的。当然要考虑到交互式任务的复杂程度,对于简单的任务来说,5 秒的响应时间未免太长了,但对复杂的任务来说就可能是合理的;

2. 吞吐量(Throughput):在维持合理的响应时间的前提下,在给定的时间范围内所完成的事务处理(Transaction)(或称“交易”)次数。对于交互式工作,吞吐量按每小时的事务处理(交易)次数来测量;

3. 有效用户数(Number of active users):是用给定的事务处理速度,在维持合理的响应时间的情况下,系统能同时支持的用户数目。有效用户数可由事务处理速率、平均响应时间、平均键入时间及操作人员的平均思考时间等推算出来的。假定平均响应时间为 5 秒,平均键入时间为 10 秒,平均思考时间为 15 秒,则终端操作员平均每分钟能输入 2 次新的事务处理(2 次/分钟)。若系统 A 每小时能处理 1,200 次这类的事务处理(即 20 次/分钟),则可推算出系统 A 对于给定的事务处理类型可支持 10 个有效用户($20/2=10$)。

三、为在事务处理及其它交互式应用环境中对不同的计算机系列进行性能测试,由包括有 20 几家大的计算机公司参加的一个国际性组织设计了一种工作负载,起名为 RA-MP-C,特征是很少或没浮点运算,执行应用程序的时间相当多,执行管理程序

的时间相当多,大量的 I/O 操作,大量的数据库操作(查询、修改数据文件等),大量的交互式操作。RAMP-C 的程序编制是独立于系统的硬件及软件的,这就保证了所有各种计算机系统都是在公平的、相等的条件下进行测试的。参加测试的厂商及系统

型号有 IBM AS/400B10-B60, DEC micro VAXII DEC、micro VAX 3600、DEC VAX 8530、HP micro 3000 XE、HP series70 等,被测系统的配置和测试结果如下面的图 1—图 6 所示。

1. 被测系统的配置

图 1

厂 商	IBM	IBM	IBM	IBM	IBM	IBM
系 统	AS/400	AS/400	AS/400	AS/400	AS/400	AS/400
型 号	B10	B20	B30	B40	B50	B60
操作系统	OS/400	OS/400	OS/400	OS/400	OS/400	OS/400
内 存	4MB	4MB	4MB	8MB	16MB	32MB
DASD	数量	2	2	2	2	2
	容量	630MB	630MB	800MB	1700MB	1700MB
	型号			IBM9332	IBM9335	
DASD 取数臂	2	2	4	4	4	4
DASD 控制器	集成式	集成式	集成式	集成式	集成式	集成式
线 速 度	3mbps	3mbps	5mbps	5mbps	5mbps	5mbps

图 2

厂 商	DEC	DEC	DEC	HP	HP
系 统	micro VAX	micro VAX	VAX	micro 3000	3000
型 号	II(H9642)	3000	8530	XE	series 70
系 统 软 件	Micro VMS RMS,FMS	VMS RMS,FMS	VMS RMS,FMS	MPE-V IMAGE,VPLUS	MPE-V IMAGE,VPLUS
操作系统版本	4.5	4.6	4.5	UB-D-2	UB-D-4
内 存	16MB	32MB	80MB	8MB	16MB
DASD	数量	1	4	3	4
	型号	RD54	RA84	7937	7937
	数量	2		1	1
	型号	RA81	RA82	7936	7936
DASD 取数臂 操作机构	3	4	7	4	5
DASD 控制器	1/RQ DX4	1/KDA 50	2/KDB 50	集成式	集成式
线 速 度	9.6kbps	9.6kbps	9.6kbps	9.6kbps	9.6kbps

2. 测试结果之一

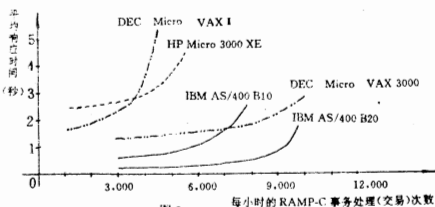


图 3

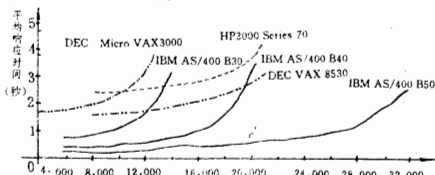


图 4

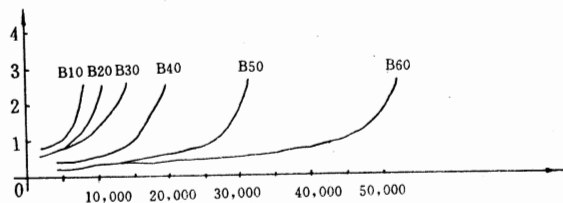


图 5

4. 相对性能直方图(在 70% 利用率时的 RAMP)

—C 吞吐量)

每一条高度代表一个系统的 RAMP—C 负载能力,它是在 70% 系统利用率下得到的。对于它的操作环境,希望系统的利用率不要超过 70%。

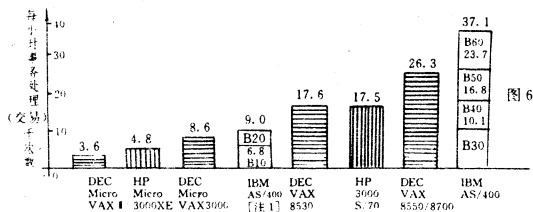


图 6

注 1:所有 AS/400 的数据都是在平均响应时间小于等于 2 秒的条件下得到的。如果响应时间大于 2 秒,则 AS/400 B10 的吞吐量还可提高。

注 2:IBM 参予上述测试的是 AS/400 机最早的 B10—B60 型,当时各项性能指标还比较低。当今 IBM 已推出 E 系列,AS/400 的 E02、E04、E06、E10、E20、E25、E35、E45、E50、E60、E70、E80、E90 共 13 个型号,在 RAMP—C 环境下进行测试,其相对性能分别是 B10 (内存为 16MB, DASD 容量为 945MB) 的 1.5、1.9、2.6、2.6、3.5、4.2、3.4、4.8、6.4、10.2、14.2、25.2、34.4 倍。

从测试结果中看到,AS/400 对上述 3 项指标 (响应时间、吞吐量、有效用户数) 上比其它系统都具有明显的优势。以 AS/400 B60 为例,当每小时的事务处理次数达到 45,000 次时,其响应时间仍不超过 1 秒。

四、用 MIPS 来对同一种系统的不同型号的机器进行性能对比是有意义的,但对属于不同生产厂家的不同系统进行比较则是无效的。因为各不同厂家的机器的体系结构和指令系统完全不同。因此,即使 IBM 公布了 AS/400 系统的 MIPS 指标,用户也无法根据它对 AS/400 系统和其它厂家的系统进行速度方面的科学比较。

五、过去采用计算机每秒执行的百万条指令数 (MIPS) 作为衡量计算机性能的标准,由于计算机系统结构中增加了许多新的内容,如多处理单元、专用微码以及为提高性能而设计的系统软件等,所以使得今天用此测量方法已不适合了。

今天, MIPS 指标已不能全面衡量一个计算机系统的速度性能。影响一个系统速度快慢的因素有很多,处理机的指令执行速度只是这众多因素中的一个,对于事务处理计算机来说更是如此。决定整个系统速度的其它因素至少应包括:

- 指令系统的效率和丰富程度;
- 存取主存的速度 (主存存取时间);
- 内部数据传送速度、通道和总线的数目、

速度及数据路径宽度;

- 存取磁盘数据的速度;
- 存取工作站数据的速度;
- 分布式的微处理器 (用于减轻主处理机的负载) 所造成的影响。

六、MIPS 的测试结果依赖于工作负载的类型。也就是说,即使是同一种系统,在不同的工作负载 (如侧重进行数值计算或侧重进行数据处理) 下测试的 MIPS 结果将是不同的。

七、IBM 在 AS/400 系统硬件设计上,采用了先进的处理器、先进的存储器和先进的硬盘技术来提高系统的处理速度和处理能力。

1. AS/400 采用分布处理器设计,系统具有多处理器结构 (见图 7、图 8)

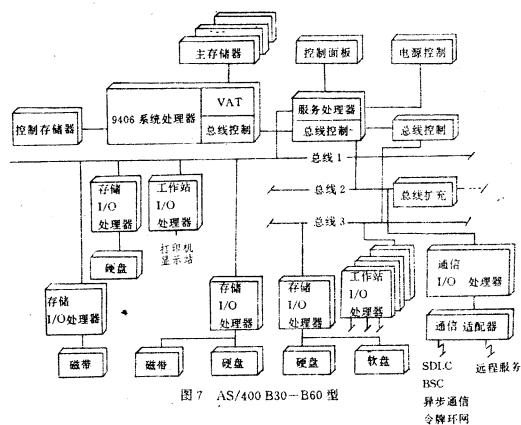


图 7 AS/400 B30—B60 型

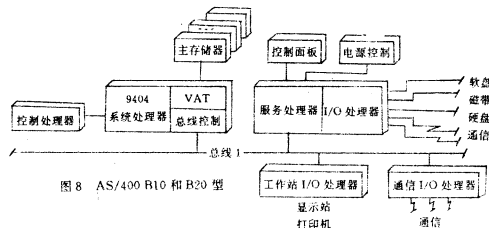


图 8 AS/400 B10 和 B20 型

除了系统处理器 (主 CPU) 外,各控制器和通信、I/O 都有独立的处理器。如服务处理器、通信 I/O 处理器、存储 I/O 处理器、工作站 I/O 处理器等。

系统处理器通过高速总线同各个功能独立的 I/O 处理器进行通信,直接存取数据,获得通信、磁介质存储器、工作站 (包括显示工作站和打印工作站)、I/O 设备等自成体系,从而减轻了系统处理器的负担,并实现了多个处理机的并行操作。

系统处理器对主存储器的存取是由虚拟地址转换 VAT (Virtual Address Translation) 硬件来实现的。主存储器中的地址转换表和硬件中的转换查找缓冲器提供从虚拟到实际主存储器地址映射的高速辅助。

系统处理器具有 32 位数据通道和 48 位寻址能

力,支持直接存取 281 万亿个字节的存储器,这种系统处理器所配置的软件和硬件结构能使存储扩充到 64 位寻址能力。

系统处理器从随机存取(RAM)控制存储器中取出 42 位水平微码(HMC)机器指令,进行 8 位、16 位或 32 位运算,大多数机器指令占用一个服务处理器周期。处理周期与机型有关,其变化范围为 60—120 毫微秒。

系统还包括一个服务处理器(service processor),作用是启动系统和对系统进行维护(即系统服务),例如故障隔离、错误检测、出错报告和运行系统处理器服务实用程序等。服务处理器还为控制面板和电源控制功能提供了接口。

较大的 AS/400 机型为磁带和软盘、通信线路和令牌环网线路、本地工作站和打印机以及服务处理器设计了专用的 I/O 处理器,从而提供了最佳的性能。这类机器的结构见图 8。

较小的 AS/400 机型的设计原则是降低造价和获得具有竞争力的性能。其基本结构是将 I/O 处理器和服务处理器做在一起合成一个 I/O 处理器。当要有其它 I/O 设备时,需另加 I/O 处理器以支持新增加的本地工作站和打印机或通信线路。这类机器的结构见图 7。

2. 为了使系统的输入输出吞吐量和响应时间最佳,IBM 公司采取了某些必要的新技术。最新的高密度存储技术是主存储器的基础。

主存储器采用了 IBM 最新的 16 兆位存储芯片。它有两种性能等级(65 毫微秒存取时间和 120 毫微秒存取时间)。主存储器的数据通道为 8 字节宽。根据不同型号,主存储器容量为 2 兆到 96 兆(E 系列为 8 兆到 512 兆)字节之间。在每个型号中,处理器的速度要同主存储器和控制存储器的存取时间及容量相协调才能获得最佳的性能。

控制存储器有 4K 和 8K 两种容量。控制存储器比主存储器贵,但速度却快得多。在比较便宜的 4K 控制存储器设计中,有一部分机器指令是存储在主存储器中的,处理器是从主存储器中而不是从控制存储器中获取这些指令的。

3. AS/400 硬盘存储器是 IBM 最先进的直接存取存储器设备(DASD),其上所有的寻道和定位工作由一个单独的微处理器来控制,从而使系统处理器可腾出空来运行更多的用户作业,以提高中心处理器的处理速度和效率。

4. AS/400 采用单级存储(单级寻址)和虚拟存储技术。系统将主存储器和辅助存储器(硬盘)看作为单一的一个存储区域(地址空间),使主存和硬盘存储逻辑上相互连接成一个连续的存储区域。用户可以在这整个存储区域(2^{28} 即 281 万亿个字节)里寻址,直接存取访问数据(见图 9)。程序员可以不必考虑程序的大小和数据放的地址。单级存储大大地

加快了数据存取速度和大大扩大了数据存放的范围。

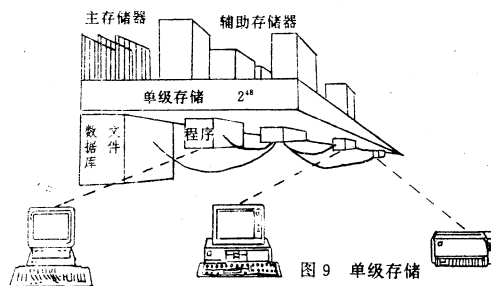


图 9 单级存储

将程序装入主存并将控制转到装入地址(主存和硬盘装程序的物理地址),是由 AS/400 虚拟存储的功能实现。虚拟存储将每个程序和数据区自动分做一些块(叫“页”),当程序运行时,AS/400 随时决定整个程序不一定需要全放入主存。主存里只保留这个时间需要的一些页,剩余的页仍存放在硬盘上。当需要程序另一页时,自动将该页纳入主存一个可用的块里,取代旧的已处理完的页。程序员不必关心一个程序所占的空间大小的限制,好象有足够的存储空间可用似的。

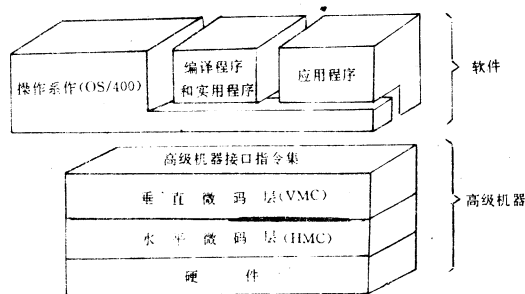


图 10 AS/400 分层机器结构

5. AS/400 采用分层机器结构和微码技术。处于低层的系统硬件与处于高层的水平微码(HMC)层和垂直微码(VMC)层组成了一个“高级机器”(见图 10)。水平微码层提供硬件指令集,垂直微码层提供系统的机器接口 MI 结构。微码比高级程序运行速度快得多,所以应用程序中使用的功能如果是用微码实现则运行得非常快。

高层机器除了按传统方式在操作系统中提供基本管理和资源管理以外,还为用户提供多任务操作、综合保密、程序和数据的自动调页、动态联编的内部语言调用、交互式调试 2^{28} (281 × 10¹²) 字节存储器寻址范围以及再入程序(所有用户共享程序指令的单个拷贝)等功能。

六、AS/400 提供了强有力的关系数据库和结构

式查询语言 SQL/400, 提供了完整的数据描述能力及应用程序存取数据的一致接口。

AS/400 数据库一部分是集成在垂直微码中, 另一部分集成在操作系统 OS/400 中, 或者说 AS/400 数据库与 OS/400 操作系统是一体化的。当用户安装 AS/400 时, 就同时安装了该数据库管理系统。数据库系统的许多基本功能则由机器微码(垂直微码)来实现。这一创新实现数据库系统的高性能, 为用户提供了高效的灵活的数据管理手段。

所有数据均存放且只能存放在这单一的关系数据库里, 或者说, AS/400 数据库用作系统的单一数据源, AS/400 数据库管理程序用作数据的唯一的管理程序。用户任何一种语言的程序或其他访问途径(如 SQL/400、IDDU、CL、Query、office、PC Support、ADT 和系统菜单等), 都可以直接访问系统上的所有数据(当然要在规定使用的数据权限内)。

用户可以使用 AS/400 上的数据库存储一个程序的数据, 然后在另一个完全不同的应用程序中使用这些数据。AS/400 关系数据库的一些指标为:

- 一个记录的字节数: 32,766
- 一个记录格式的字段数: 8,000
- 一个文件的关键字段数: 120
- 关键字的大小: 120 个字符
- 一个文件包含的记录数: 16,777,215
- 一个文件的字节数: 2,417,483,648

SQL/400 是 AS/400 系统上运行的结构式查询语言, 是系统内部集成的关系数据库的 SAA(系统应用体系结构)接口。它包含了非常强有力的数据定义和数据操作语句, 在许多场合中一些需要若干程序逻辑才能实现的操作, 往往只需一两个 SQL 语句就可以实现。SQL/400 在企业事务处理中对信息的查询提供了最快的数据排序及查询检索速度和很强的数据管理能力。

为了测试 IBM AS/400 数据库和 SQL/400 对于信息检索的能力, 1991 年 12 月下旬, 吉林省计算机

技术研究所用很短时间将中科院情报所计算中心提供的在 VAX 机输出的《中国学术会议数据库》其中一盘磁带(共 10,238 条记录, 每条记录非等长, 记录最长约 1,500 字节)的数据全部读入并转换到 AS/400 B10 机关系数据库中, 没有建立任何倒排文件和索引表, 未使用其他任何检索技巧, 直接使用 SQL/400 交互命令进行万条记录的顺序地进行“全文”检索实验。检索入口可以是任意一个字段或子字段, 如“工作单号”、“馆藏号”、“题名”、“个著者姓名”、“会议名称”、“会议届次”、“会议地点”、“会议起止日期”、“会议主办单位”、“个著者所在单位”(隶属机构)、“母文献名”、“文摘”、“主题词”、“自由词”等, 可有“与、或、非”逻辑运算和“=、<>、<=、>、<、>=、”等关系运算的任意组合。实验结果表明, 用 SQL/400 检索平均检索时间为 15 秒钟。而用 AS/400 Pascal 语言程序检索, 则长达 7 分钟。可见 AS/400 数据库加之使用 SQL/400 软件进行信息检索的速度相当快, 完全适用于图书、情报等信息系统的检索。如果采用 AS/400 更高档的机型特别是 E 系列机, 内存容量再扩大, 检索速度会更快。

AS/400 进行信息检索的速度之所以这么快, 正是由于 AS/400 具有先进的分布式处理器技术、先进的存储技术特别是单级存储和虚拟存储技术、先进的微码技术、先进的集成式数据库管理系统、先进的结构式查询语言等众多因素所导致的。

综上所述, IBM 没有采用 MIPS 指标对 AS/400 系统进行衡量, 这毫不意味着自 AS/400 系统在速度上不尽人意。恰恰相反, IBM 对 AS/400 系统进行的性能评价表示出 IBM 对该系统的各项性能指标(包括速度指标)的竞争力拥有充分的信心。选用 IBM AS/400 机器做为事务处理机, 无疑是一种最佳的选择。

注: 上述对 AS/400 机的描述都是基于 AS/400 最早的 B 系列而言, 现已发展到 E 系列, 其相对性能指标比 B 系列已有大大提高。

人类进入信息社会! 您要及早

订阅

《中文信息》

全国创办最早的推广应用中文电脑的科技期刊, 国内外中文信息界共识的信息中心, 它是——

- 人人学用中文电脑的良友!
- 搞办公自动化的技术参谋!
- 应用中文通联世界的向导!
- 您建功立业的现代化工具!

○“通联世界”向您提供海外最新技术信息 ○“文字处理技术”全国独有 ○“中文信息产业化”专为振兴信息产业服务 ○“计算机教育”为全国中小学开设必修课提供规范化参考教材

全国邮局订阅 邮发刊号: 62—56, 全年 10 元 直接订阅: (610065) 成都科技大学 263 信箱《中文信息》社

微机过程控制系统中的接口与抗干扰问题

李均宜

嘉应大学电子系 (广东梅州市, 514015)

摘要 本文主要从硬件设计上论述了微机过程控制系统中的接口与抗干扰问题, 有一定参考价值。

关键词 过程控制 接口 干扰 阻抗 缓冲器 光电耦合器 隔离

一、引言 微机过程控制系统的接口电路中, 与外部设备联系在一起的主要是传感器及其信号变换放大等电路、执行机构、显示器等等。在生产现场, 传感器执行机构与 CPU 所在地的控制台之间是有相当距离的, 并且往往是分散的。因此, 从系统结构的配置来说, 有可能接口电路紧靠 CPU, 这时 I/O 模拟信号线拉得较长, 如图 1(A); 也可能接口电路靠近外部设备, 这时三总线的数字信号线拉得较长, 如图 1(B); 如果是利用 IBM-PC/XT/AT 或 APPLE 等基本配置的系统机作主机, 自行设计接口电路, 则往往由于其主机箱内的 I/O 接口插座有限, 在外设又较多的情况下, 势必要将一部分接口电路及其三总线的有关部分移至主机箱外形成接口箱, 这时, 模拟信号线与数字信号线都可能拉得较长, 如图 1(C)。

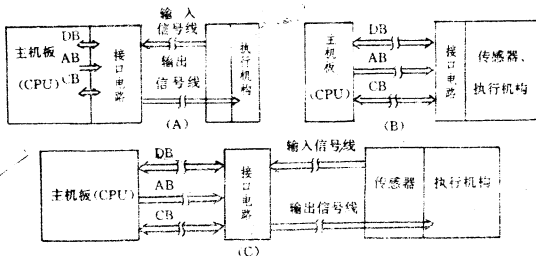


图1 微机过程控制系统硬件结构的三种可能方式

众所周知, 生产过程控制系统现场的干扰是比较严重的, 比较突出的干扰源是交流供电电压的突变, 传感器(如热偶, 热电阻等)在某些高温电热源间的漏电, 以及空间杂散电磁场等。这些干扰源可能作用于信号传输线间而产生差模干扰, 或分别作用于信号线和系统地线(模拟地线或数字地线)与大地间而产生共模干扰。一般地, 共模干扰途径较复杂, 不

易捉摸, 往往顾此失彼, 而且它们最终又将转化为差模干扰, 差模干扰出现在数据传输线上时, 会使检测量产生超过允许的误差, 若出现在系统的地址线或控制线上其强度又超过允许值时, 则会使计算机程序执行错误, 甚至根本无法执行(死机)。下面着重就共模干扰及其行之有效的几种抑制方法进行讨论。

二、硬件接口电路的结构与干扰 在一般工业过程控制系统现场, 常采用屏蔽线或扁平电缆作为信号传输线。出于安全保护等原因, 信号源端和主机箱、接口箱外壳都接大地(即使不接大地, 其分布电容、电感或漏电阻与大地之间也有耦合关系)。两个接地点间因为距离较长, 各接地点接地电阻又不可能一致, 或者分布电感电容耦合的干扰不同, 因而不可避免有较大的电位差 U_{cm} (有时可达几十伏), 尤其在电网电压突变时, 更为严重, 见图 2。 U_{cm} 是工业现场比较主要的共模干扰源, 它可以经由信号的来回线阻抗 Z_1 、 Z_2 及其与地之间的漏电阻、分布电容等效阻抗 Z_3 、 Z_4 之间形成的不平衡电桥, 转化为差模干扰 U_{de} :

$$U_{de} = \left[\frac{Z_3}{Z_1 + Z_3} - \frac{Z_4}{Z_2 + Z_4} \right] U_{cm} \quad (1)$$

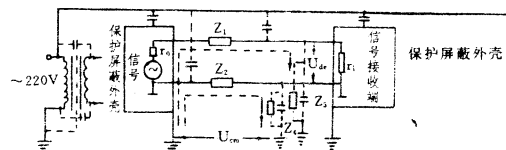


图2 电网地线、现场接地线形成的干扰信号 U_{cm} 和 U_{de}

应该指出, 以上讨论及式①, 对模拟信号传输线及数字信号传输线都是适用的, 而且对于干扰来说, 逆向的干扰即对信号源一侧的干扰, 也可用式①来描述。如图 1(B) 及 (C) 中 CPU 与接口电路间的三总

线传输线上出现的干扰,作用在 CPU 一侧时,如不加有力的防范措施,其造成的危害是致命的。

使用 IBM-PC/XT、AT 和 APPLE、Z80 单板机一类基本配置的系统机作为工业过程控制系统的主机时,其主机箱中都有一个或多个连接着总线的外设接口插座,总线同时又和主板上的 ROM、RAM 连接,如图 3 所示。系统工作时,其用户程序及缓冲数据都是存放在 ROM 或 RAM 中的,如果连接外设接口插座与接口电路间的连线上尽管只有一根地址线或控制线上出现了超过噪声容限的干扰(把“0”变为“1”或把“1”变为“0”),这干扰便畅通无阻地进入主机板上的地址线或控制线,导致寻址错误而“死机”。这是设计接口电路时,必须注意的。

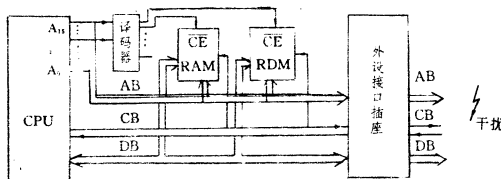


图3 系统机主板上的三总线
与RAM、RDM及外设接口插座的连接

同样,在模块化结构的工业控制机系统中,各模块间的联线,尤其是将总线拉得比较长距离的情况下,如无防范措施,也势必同样发生“死机”的问题。

因此,在外设接口电路不多时,应将其尽可能地集中在几个模块中并全部插在主机板 I/O 接口插座上,这可以使结构紧凑,避免将总线拉出主机箱外招致干扰。至于接口电路到传感器、执行机构之间的连接电路,仍应采取有效的抗干扰措施,如果接口电路较多,不得已将总线拉出(即接口箱与主机箱间有总线联线如图 1(c)),则应在总线传输线的结构上和总线两端的电路上,都必须有得力的抗干扰设计。

三、干扰的几个有力对策 对于微机过程控制系统来说,抗干扰的设计,首先要考虑的是抑制总线传输线上的干扰,其次才是保证测控数据的精度。因为对于测控数据干扰的抑制,除硬件上可以用屏蔽浮空等措施外,软件上还可以运用数字滤波等种技术予以有效的抑制。而当软件根本无法顺利运行时,就谈不上什么测控数据的精度了。

图 4 是常用的屏蔽浮空抗干扰电路,它可以用在模拟信号传输线上,也可以用在数字信号传输线上。显然它可以大大抑制住空间杂散电磁场造成的差模干扰,同时也可以大大削弱共模干扰 U_{cm} 转化为 U_{de} :

$$U_{de} = [Z_3 / (Z_1 + Z_3) - Z_4 / (Z_2 + Z_4)] (Z_5 /$$

$$Z_6) U_{cm} \dots\dots\dots ②$$

式中 Z_1 、 Z_2 为传输线阻抗, Z_3 、 Z_4 为传输线对屏蔽层的等效阻抗, Z_5 为屏蔽层通路阻抗, Z_6 为系统与外壳浮空时的等效阻抗。 Z_0 、 Z_0' 及 Z_i 、 Z_i' 分别为传输线出线端与进线端的等效阻抗。显然,在这里共模干扰 U_{cm} 被 Z_5 、 Z_6 分压,一般 $Z_5 \ll Z_6$, 实际的干扰 U_{de} 被大大削弱,因而效果是颇佳的。

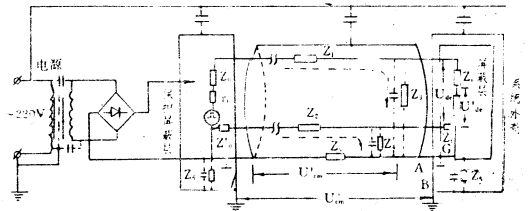


图4 屏蔽浮空的抗干扰电路

但是,图 4 要求每根信号线都使用双芯屏蔽线,不仅造价昂贵,而且结构臃肿。微机系统中习惯用结构简洁的、价格较低的扁平电缆或双绞线,它们对差模干扰均有一定的屏蔽作用,当然效果要差些,对于强干扰的袭击仍不得另加设防。

图 5 是在主机侧和接口电路侧的总线信号传输上加光电耦合的隔离、屏蔽、浮空电路。图 5(A)为电路结构示意图,图 5(B)为光电耦合的一种接法。可见,这时信号传输长线已被浮空,主机板和接口电路及其所连接的模拟信号电路,已被电气上完全隔离。光电耦合电路在信号线一侧又使用单独的电源,公共端(信号的公共地线)不接大地,也不和主机板(CPU)公共端及接口电路的公共端相连。这相当于图 4 中的共模干扰通路已被切断(Z_0 、 Z_0' 、 Z_i 、 Z_i' 均趋近于无穷大)。在传输线两端口上还有其它途径窜入的差模干扰也同样被光电器件等效的 Z_0 、 Z_0' 及 Z_i 、 Z_i' 隔断。这样,干扰不可能逆向地施加于 CPU 板的地址线和控制线(发光二极管与光敏三极管间的分布电容极小,约 1PF 左右)而造成控制错误或死机的危害。这种电路,在传输线使用扁平电缆或双绞线的情况下就可有令人满意的抗干扰能力。它还有一个突出特点是可以隔离接口电路一侧意外事故发生的电压对 CPU 一侧的袭击(一般可隔离 500V 以下)。

应该指出,这种光电耦合电路的一个主要缺点是延迟时间较 TTL 电路长得多,如 TLD-521 的延迟时间典型值为 6μs,比一般 TTL 电路大 3 个数量级,光电器件较多时,其特性不可能完全一致,为了可靠地工作,在硬件上光电器件的前后都得加数据寄存器或锁存器。软件上必须在每一个有关控制指

令或数据输入输出指令之后,都要调用延时子程序,以便等待数据可靠地经光电器件传出去。这就增加了电路的复杂性,同时程序运行时间也增加不少,这对于要求高速度数据采集的场合是不利的。

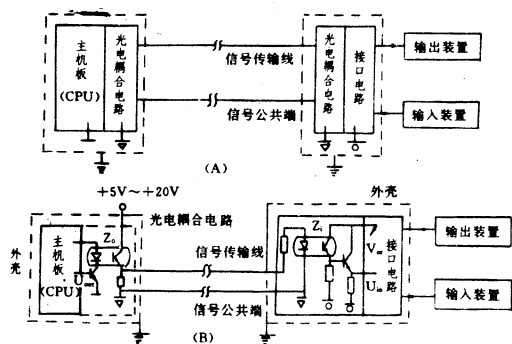


图5 用光电耦合作隔离、浮空的抗干扰电路

图6所示用TTL的三态缓冲器取代光电耦合器的电路。它在一定程度上具有光电耦合电路的抗干扰能力,却免除了光电耦合电路缺点。

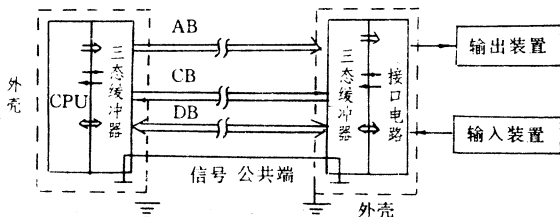


图6 用三态缓冲器取代光电耦合电路时的结构图

众所周知,三态缓冲器的使能端有效时,输入与输出间可以近似认为阻抗为零,信号可顺利向前传输如图7(A),但这时由输出端逆向地对输入端的干扰信号,例如由传输线向主板一侧三总线施加的干扰信号,则其逆向等效阻抗是缓冲器若干管子的PN结逆向阻抗,其数值可在 $M\Omega$ 级,即等效于图4中的 Z_i 具有很高的数值。信号线上的共模干扰通路接近于被切断,共模干扰转化为差模干扰后的实际作用值 U_{de} (见图4)也被 Z_i, Z_i' 所阻隔,这就首先有效地避免干扰造成死机的问题。至于干扰信号对接口电路一侧的影响,这时正常的总线信号和干扰信号都同处于正向传输的地位(相当于图4中的 Z_i 很小),抗干扰的能力则依靠其本身的噪声容限了。三态缓冲器的使能端为无效时,则传输线的两侧都使

传输线处于悬浮状态,其隔离等效阻抗 Z_i 都可在 $M\Omega$ 级以上,如前所述,干扰信号可有效地被阻隔。由此可见,即使在缓冲器使能端有效时,对CPU一侧的抗干扰能力也是可佳的。

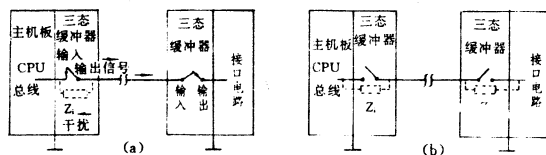


图7 三态缓冲器隔离作用示意图

如果我们用CPU发出的 \overline{IORQ} 信号作为三态缓冲器的使能端控制信号,那就可以仅当CPU和被连接的外部设备互相访问的那些时刻,随机出现的干扰信号,其强度又超过了缓冲器的噪声容限的话,才可能造成对接口电路操作的错误,而这时对CPU一侧来说,仍具有较高的抗干扰能力,不致造成灾难性的死机恶果。换句话说,这种传输线两端均采用三态缓冲器隔离的措施,极强干扰造成死机的危险机率已经很微小了。

显然,采用三态缓冲器作隔离时,它们的使能端(低电平有效)若固定地接地(有效)的话,那么系统运行的全部时间内,接口电路侧都有可能招致超过其噪声容限的强干扰危险,如无必要考虑其它特殊的问题,使能端以接 \overline{IORQ} 信号为好。这样,不仅使三态缓冲器起到扩大带负载能力的作用,而且更重要的是充分发挥其抗干扰的效能,提高了系统的可靠性。顺便指出的是,这种接法,由于一般TTL的缓冲器其延迟时间最大也不超过30ns,在机器时钟频率不大于10MHz的系统里,它不致于搞乱读、写周期的时序,工作是可靠的。

作为例子,图8为Z80CPU主板与A/D板间总线传输线的两端加接三态缓冲器74LS245与74LS244的电路。CPU侧用 \overline{IORQ} 及A7~A4译码输出“与非”后,作缓冲器的使能信号,A/D板侧则在 \overline{IORQ} 传输过来后作为缓冲器的使能端信号,74LS245的 $\overline{M1}$ 端接RD,以控制数据总线的读、写传输方向。A/D转换后的数据,采用查询方式读取。A/D板上还可接14个地址供接其它外设。这样,仅当地址为90H~9FH的16个外设与CPU互访时,主板上的三总线有关部分才与外界有联系。实践证明,它是有颇佳的抗干扰能力的。

以上主要在硬件结构上讨论了几个抗干扰的措施。而且旨在首先保证软件的正常运行,至于保证测控数据的精度,则可在此基础上采用常用的成熟可靠的软件方法。在此限于篇幅,毋庸赘述。

参考文献

- (1)王士元、吴芝芳编著,《IBM PC/XT(长城0520)接口技术及其应用》,南开大学出版社,1990。

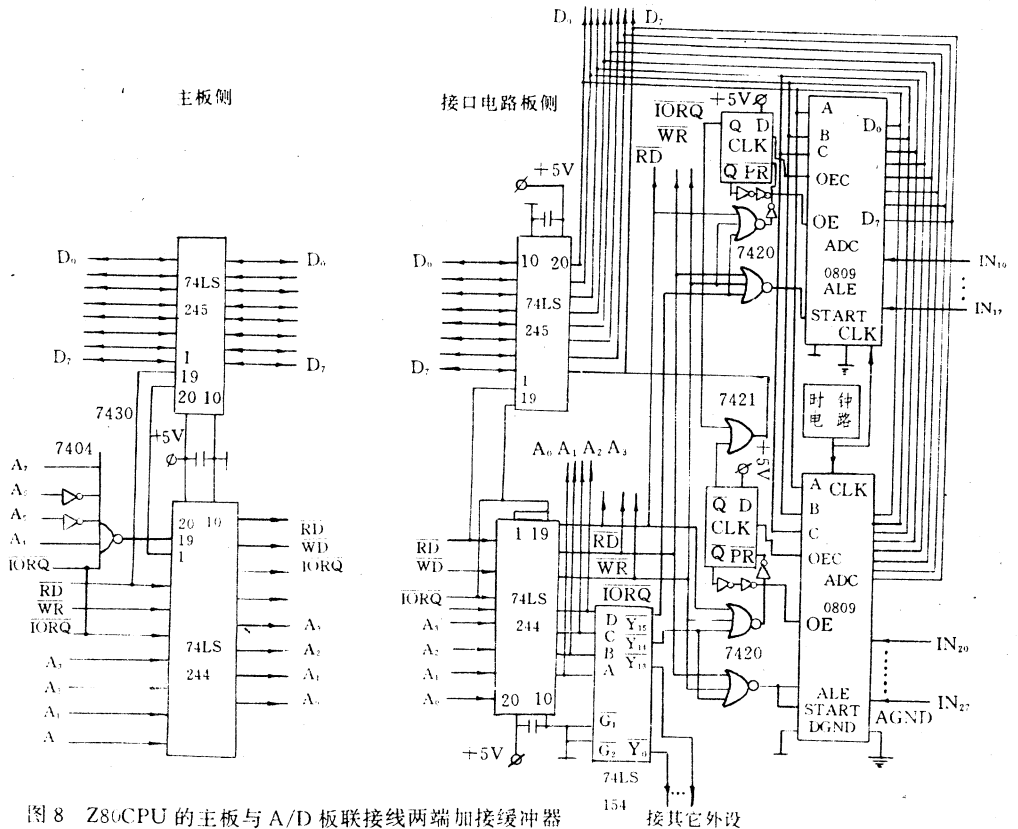


图8 Z86CPU 的主板与 A/D 板接线两端加接缓冲器

4, 第一版。

(2)《微机数据采集过程控制应用文集》,北京计算机配件五厂。

(3)潘新民编著,《微型计算机控制技术》,人民邮电出版社,1985. 8。

(4)PROGRAMMING THE 6502 RODNAY ZAKS SYBEX, 1983。

(5)TECHNICAL MANUAL AND DATA BOOK/STD BUS, SYSTEMS AND PRGM PROGRAMMERS PRO-LOG CORPORATION, 1985. 11。

(6)李大友主编,《微型计算机接口技术》,高等教育出版社,1988. 11。

(7)《IBM PC 硬件系统电路分析》,中国科学院希望高级电脑技术公司,1987. 8。

紧跟时代步伐 勇攀科技高峰

——电脑精品演示暨学术交流会在蓉城举办

《本刊讯》蓉城十月,金秋送爽,喜事重重。适逢党的十四大胜利闭幕之际,四川省计算机学会微机专委会、四川新潮计算机集团公司、《中国计算机报》社驻四川记者站于10月20日至21日在蓉城举办了一次别开生面、丰富多彩的电脑精品演示及学术交流会。会上,展示了新潮公司新近推出的多媒体企业广告演示计算机系统、彩色电脑画像系统、证券业电话自动交易系统最新成果,展示了'92新一代电脑精品——XC新潮系列微机,IBMPS个人系统计算机、IBMPS/2便携式电脑、COMPAQ DESKPRO/M系列电脑、Hpveetrace“无忧计算”U系列机,喷墨打印机,网络智能UPS电源,工业控制机开关电源,调制

解调器系列装置等30余项新产品,近两千余名计算机专业人员及管理人员观摩了演示。

演示会期间,三个主办单位组织了有关计算机病毒防治、工业多机控制网络系统,多媒体技术、证券业电话自动交易系统等方面的专题学术报告会,记者站向与会观众散发了《中国计算机报》宣传资料,受到广大观众热烈欢迎和一致好评。由于主办单位的共同努力,使会议开成了一个电脑精品新闻发布会,计算机应用学术交流会和《中国计算机报》驻川记者与广大用户、读者之间的沟通联谊会。《中国计算机报》驻川记者站)

LQ-1000 打印机常见故障及维修

李红亮

中国科学院广州电子技术研究所 (广州, 510070)

本文重点分析 LQ-1000 打印机的主控逻辑电路、接口电路、驱动电路及机械装置等部分的常见故障及维修。

例一

故障现象: 打印机上电后, 只有电源指示灯亮, 打印机没有任何其它反应。

故障分析: 这种现象一般是由主控逻辑电路部分发生故障引起的。

维修: 用示波器或逻辑笔测 CPU (μ PD7810) 的第 30、31 脚, 有时脉冲输入; 再查地址线、数据线, 均无脉冲, 怀疑复位电路有问题; 测 11D 的第 4 脚, 为低电平, 使得 CPU 始终处于复位状态, 程序无法执行; 往前查得 4A 的第 1、2 脚均为高电平, 用 CD4584B 替换 4A 后打印机工作正常。

例二

故障现象: 上电后, 字车一直往右端走, 碰撞右边挡板而不返回左边原点位置。

故障分析: 上电后, 原点位置检测电路工作, 当它检测到字车不在原点位置时, CPU 命令字车向左移动, 直至回到左边的原点位置; 当检测电路检测到字车上电前已在原点位置时, CPU 首先命令字车向右运动, 只有当检测电路测到字车已离开原点位置后, CPU 才命令字车向左移回原点。因此, 当检测电路发生故障, 始终发出“字车在原点位置”的信号给 CPU, 将会发生上述一直往右移而不返回的故障。

维修: ①上电后, 用逻辑笔测 4A 的第 5、6 脚, 用手移动字车过程中 (包含原点位置时刻), 第 5 脚有高/低电平变化, 而第 6 脚却恒为低电平, 怀疑 4A 已坏, 用 CD4584B 替换后故障排除。

②上电后, 用手移动字车 (包括回原点位置) 过程中, 用逻辑笔测得 CN5 的第 3 脚恒为高电平, 说明原点位置传感机构本身发生故障。用酒精擦净光电耦合器上的灰尘等脏物后, 故障排除。

例三

故障现象: 打印机连续打印一段时间后就停止打印, 联机指示灯 (On line 灯) 闪烁。

故障分析: 从现象上看, 这是由打印头温控电路

发生故障而引起的。LQ1000 打印机为了防止打印头的温度过高而损坏打印头, 在打印头上安装有一个热敏传感器, 当它检测到打印头温度过高时, 则停止打印, 此时联机指示灯闪烁, 如果热敏传感器发生故障, 会起上述故障。

维修: 换同型号的热敏传感器。或者干脆去掉过热保护, 这只要将主电路板上连 CN5 第 12 脚的线割断, 在断线处 (主板上) 与地之间接一个比 1.3K Ω 稍大一点的电阻 R_p 可。

另外, 有时出现开机后打印机就不打印, 联机指示灯不停地闪烁这种故障, 这一般也是由打印头温控电路发生故障而引起的。

例四

故障现象: 打印机字车移动正常, 但打印时不走纸。

故障分析: 一般情况下, 不走纸故障是由走纸电机驱动电路或走纸电机有问题而引起的。

维修: 打印过程中, 用示波器测 Q34、Q35、Q36、Q37 的基极, 均有脉冲; 再测 IC8B 的第 13 脚有脉冲, 集电极也有脉冲, 而 Q28 的 C 极却无脉冲, 用 2SB601 替换 Q28 后工作正常。

例五

故障现象: 上电后, 字车在原地抖动而不移动。

故障分析: 一般情况下, 这是由字车电机或其驱动电路发生故障引起的。

维修: 打印过程中, 用示波器测 IC1 (STK6982H) 的第 4、6、13、15 脚, 发现第 15 脚没有脉冲, 而测第 5、7、14、16 脚都有脉冲, 换 IC1 后故障排除。由于加到字车电机上的驱动信号缺相, 而引起字车抖动而不正常移动。

例六

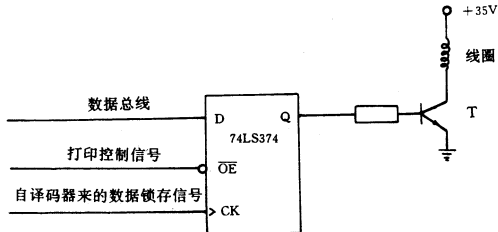
故障现象: 打印出来的字符笔划不全。

故障分析: 引起打印字符缺笔划的原因很多, 针数据出错、针驱动电路有故障、打印头太脏、断针或线圈被烧断等都会引起字符缺划, 下面仅就最常见的情况进行分析。

①断针 取下打印头, 从出针面上看到两列

针中露出白色针头的针不足 24 根,针面上可以明显地看到针孔,一般情况下说明打印头有断针,应换掉断了的针。

②驱动电路 分析:LQ1000 打印机的针驱动电路单元可用图一来简单表示。



数据锁存器(74LS374)的数据在来自译码电路的数据锁存信号控制下打入,由 CPU 来的打印控制信号控制输出的针数据脉冲宽度,数据输出后经开关三极管驱动打印头线圈。

故障维修:打印过程中,用示波器测 74LS374 的 D、Q 端和三极管的 b、c 极,都应该有脉冲,如果没有脉冲,则应替换相应的 74LS374 或三极管。

另外,如果 CPU 不能发出打印控制信号,将会使得打印机出现字车移动正常而打印全空白、听不到出针声的故障,这时应换 CPU。

例七

故障现象:自检正常,联机不打印,READY 灯不亮,锁死主机。

故障分析:锁死主机说明 BUSY 信号常高、打印机始终没准备好,应查 BUSY 信号通道。

维修:联机打印时,用逻辑笔测 1A 的第 4、5 脚均有脉冲,而第 6 脚却没有脉冲、它恒为高电平,换 1A(74LS32)后工作正常。

说明:由于 1A 的第 6 脚恒为高电平,致使 BUSY 也恒为高电平,READY 灯不亮。

例八

故障现象:自检正常,联机不打印,READY 灯闪亮,不锁死主机。

故障分析:此现象说明主机向打印机送出了数据,打印机不打印一般是因为数据选通通道出现故障引起的。

维修:联机打印时,用逻辑笔测 CN1 的第 1 脚有脉冲;而 4A 的第 11 脚恒为低电平,关机后用万用表测得该脚对地短路,换 4A(CD4584B)后故障排除。

TH3070 SL 打印机故障排除一例

陈允西

河南省计算中心 (郑州,450003)

故障现象:自检正常。联机时不走纸,字车总在一行处来回打印,而且打印字符有错。

分析与排除:打印机适配器和打印连接电缆都是好的。自检正常,联机出错,故问题出自打印机的并行接口电路。而字符打印出错情况是将字符 f 打印成 d、c 打印成 a 等等。其字符代码对照如下表:

字符	代 码	字符	代 码
f	01100110	c	01100011
d	01100100	a	01100001

从表中打印错误字符的代码可见是 8 位数据线倒数第二位出错。打印时将这位的代码“1”变成“0”而造成打印字符错。而换行的代码是 00001010。联机时是由计算机送换行代码给打印机的并行接口以实现走纸的。但该打印机将计算机送来的代码变成了 00001000(此代码在该打印机不用),从而导致联机时不走纸。

用示波器观察打印机接口插座的第 3 脚和 IC₁₈₋₃。发现 IC₁₈₋₃ 的脉冲幅度只有 0.9V。当用测试探头用力按 IC₁₈₋₃ 时,打印机联机恢复正常。故该点为虚焊点。重新焊好后打印机联机恢复走纸功能。

M1724 打印机换行时打印头不能回到左端起始位置的故障维修

陈永红 穆大明

黑龙江大学 (哈尔滨, 150080)

故障现象: M1724 打印机, 开机正常, 小车复位到左边第一个字符位置, 但在自检和联机打印时, 每打印一行, 小车的起始位置就向右移动一个字符位置, 结果使打印的数据资料成倒梯形。

故障的分析及解决: 拆机检查, 直流电机, 步进电机, 电路板上的各部及元器件均正常工作, 没有发现故障及损坏的情况。

拿下色带盒再开机自检, 空打了 5 分钟左右, 打印头可以回到左端起始位置, 打印正常, 说明问题在色带盒上。

由于色带是刚换过的, 而在没换新色带前没有发生这种故障, 对色带进行测量后发现, 国产色带较原装色带厚, 用原来的同样方法装上, 压带簧片就会压得较紧, 走带时需要的力矩增大。由于色带走动是由小车电机带动的, 因此影响了小车的正常运行, 使得换行时不能回到左端界。

打开色带盒, 可以看见左端的压带簧片太紧。拆下簧片进行校正调整, 然后重新装好, 开机自检, 则一切正常。

电网强干扰损坏微机一例

陈德明

国防科技大学二系 (长沙, 410073)

正确使用微机, 除了要求适当的环境外, 还要求防止电冲击。通常要求输入电源是经过稳压器输出的。平时关机后, 不要马上再开机, 要停一二分钟, 防止再加电使正在减速的马达又加速, 损坏马达, 在没有 UPS 时, 遇停电须立即关机。防止突然来电, 冲击机内部件。

这些注意事项都容易做到, 但对电网的干扰却容易忽视。笔者单位曾发生一起由于输入电源瞬时强抖动造成的损坏事故。今投书告戒同仁, 引起注意。

笔者单位的实验室从事高压储能方面的工作。平时作实验时, 用市电对电容器组并联充电, 然后串联放电, 产生几千到百万伏的高压脉冲, 脉冲持续几十到一百毫微秒, 放电能量几千焦耳。有一次, 将一台 IBMPC/XT 微机搬迁到高压实验室旁的一间房

子里使用, 开始几天均正常, 因为做高压实验时, 微机恰好没开。但有次微机工作时, 正好遇上做放电实验, 那边一响, 只见屏幕一闪, 什么也没有了。重新启动计算机, 只见硬盘指示灯暗闪几下, 不能自举, 再关机冷启动时, 只要电源开关一开, 硬盘指示灯就一直亮着, 颜色暗淡, 不能启动, 断开硬盘, 用软盘可以启动计算机, 进一步作互换性实验, 诊断为硬盘控制卡坏。

这次事故, 给工作带来了损失。经分析, 认为是电脉冲干扰所致。因为供给微机的电源地线与高压实验部分的地线是相通的, 由于那边高压信号太大, 可以通过地线传来一个高压脉冲。供给微机的电源虽经稳压器稳压, 但普通稳压器对如此短脉冲来不及反应, 因此一个脉冲电压侵入计算机, 损坏了部件。

RISC 结构及 SPARC 工作站

刘庆枰

铁道部 431 厂 CAD 室 (四川资阳, 641301)

摘要 RISC(精简指令集计算)结构以其高度的简洁性和极强的性价比而成为工作站的主流。本文首先介绍 RISC 的发展, RISC 的基本原理, RISC 的技术特点及其市场动态。然后介绍了 SPARC RISC 工作站的技术特色。并阐明对于国内欲开发 MIS 或 CAD/CAM 系统的一般企(事)业单位, 应该采取的策略。

关键词 RISC SPARC 工作站 编译 开放性 工业标准 超标量 并行处理 超级流水线 指令字 兼容 时钟频率 微处理器 客户/服务器

一、RISC 技术 回顾传统的 CISC(复杂指令集计算)计算机的发展历史, 为了软件编程的方便和提高程序的运行速度, 计算机硬件工程师采用可实现复杂功能的指令和多种灵活的编指方式。某些指令甚至可以支持高级语言语句归类后的复杂操作。硬件越来越复杂, 造价也相应提高, 为了实现复杂操作, 一般都采用微程序。事实表明, 复杂指令系统, 有助于提高计算机的性能。然而, 虽然处理机及其指令集越来越复杂, 但高级语言编译程序只使用了这些指令中的很少的一部分。应用程序的 80% 是由微码指令中的 20% 的指令实现的。那些最简单的指令, 却是最常用的指令。因之, 产生了 RISC 的基本原理。

70 年代中期 IBM 公司和加州大学伯克利分校及斯坦福大学开始研究 RISC。加州大学伯克利分校的研究专注于如何有效的利用 VLSI 宝贵的空间; 斯坦福大学的研究课题为“消除流水线各段互锁的微处理器 MIPS(Microprocessor Without Interlocking Pipeline Stages); IBM 则研究优化程序编译。伯克利分校的研究成果最后发展成为 SUN 微系统公司著名的 SPARC(Scalable Processor ARChitecture)结构; 斯坦福大学的研究成果则转让到 MIPS 公司的 RX000 系列; IBM 于 1990 年推出 RS/6000 系列。

作为一种新型的微处理器结构, RISC 通常具有以下特点:

- 指令系统精简, 格式统一
- 采用流水线, 增加指令执行的并行度
- 哈佛结构
- 编译优化

目前从以下三个方面来提高 RISC 的性能:

●提高时钟频率(CLK)。可由不同的半导体工艺来实现同一种结构的 RISC(RISC 的设计原则决定)。

●采用优化编译技术, 减少每个任务的指令数。

●减少 CPI。RISC 之所以能够做到结构简化而性能提高, 主要是在于减少每条指令的执行所需要的周期数 CPI。目前 RISC 的设计已实现每个周期执行多条指令。其技术途径是采用 SuperScalar(超标量), Superpipe line(超级流水线), VLIW(超长指令字)。超标量和 VLIW 都是采用多指令在多处理部件中并行处理的结构。其 CPI 可以做到小于 1, 而超级流水线则通过增加流水线的级数, 使 RISC 处理器的性能提高。作者看来, 超标量和 VLIW 的 RISC 产品更具优势。九十年代开发的高性能 RISC 微处理器都将采用超标量和并行处理技术。

由于先进的半导体技术、存储介质、软件技术和其它有关工艺技术的发展, 计算机的处理速度按指数曲线上升, RISC 芯片的性能已得到极大的提高。过去的巨型机可以作成现在的台式计算机; 过去用在大型计算机中的 Cache 存储、紧耦合处理、流水线和 CPU 并行处理等技术, 现已集成到单 RISC 芯片中; 而近来发展的数字信号处理、阵列处理、专用算法处理、自适应处理、神经网络计算机和模糊逻辑计算等新技术也已集成在 RISC 芯片中。以 RISC CPU 为基础的工作站不仅将覆盖过去传统的微型机到大型机的范围, 而且其性价比都将占优势, 九十年代无疑是工作站的时代。近年来, 各大计算机厂商都相继推出以 RISC CPU 为基础的工作站产品系列, 使得 RISC

工作站竞争异常激烈。

围绕 RISC 有如下的几个阵营:

- SPARC SUN 微系统设计
- R 系列 MIPS 公司设计
- POWER IBM 公司设计
- HP-PA HP 公司设计
- i860 INTEL 公司设计
- M88000 MOTOROLA 设计

在这六个阵营中,以 SUN,MIPS,HP,IBM 四个阵营最为重要,形成四雄并立的局面(表一为四个阵营的工作站市场份额)。近来,RISC 领域又发生了一系列重大事件:美 SGI 同 MIPS 合并;DEC 发表 Alpha 芯片;HP 发起成立了“PRO”;COMPAQ 退出 ACE 以及 SUN 正同 IBM 接近;使得围绕 RISC 的竞争越发复杂。

	SUN	HP	DEC	IBM	其它
1990 年	39.4%	20.4%	17.3%	4.4%	18.5%
1991 年	40.8%	17.2%	13.6%	5.6%	19.8%

表一 工站市场份额

据 IDC 估计,1995 年的 RISC 工作站五霸排名将是(一)SUN,(二)IBM,(三)HP,(四)DEC,(五)SGI。SUN 的 Gro11 则预测前几家公司将是 SUN,IBM 和 HP,随后是一家日本公司,再往后可能是 DEC。

二、SPARC 工作站 SPARC 是由 SUN 微系统公司开发的 RISC 结构。作为 SPARC 结构,SUN 规定了为确保软件兼容性所必需的指令系统和 MMU(存储器管理单元)的规格。对端口排列等则未作详细规定。SUN 于 1990 年 9 月将 SPARC 申请作为 IEEE 标准规格进行登记,IEEE 已把它作为 IEEE P1754 项目进行讨论。SPARC 工作站有以下技术特色:

小型、分布处理代替了“与主机直接相连接的集中式处理”,形成客户/服务器网络模式。

坚持单一的技术空间:

- 微处理器:SPARC(Super SPARC)

- 操作系统:SVR4(Solaris)

- 网络方案:ONC(开发网络计算)

- 图形用户接口:OPEN LOOK

实现系统开放,为异种系统提供支持;不同操作系统开发的软件,可在同一操作系统下运行;用户可在自己选定的硬件平台上,选择自己的系统软件。SPARC 的全部系列产品均严格按照已建立的硬件、软件、数据通信等工业标准进行设计。系统开放和标准化使得 SPARC 工作站得到众多的制造厂商和软件厂商的支持,产品品种丰富,性价比高,用户乐于采用。SUN 公司的开放式系统和工业标准的战略获得成功,给计算机界带来了巨大的冲击。IBM,DEC 也不得不调整战略,乃至进行公司改组以应付竞争。开放式系统和工业标准战略,还使得各种 SPARC 兼容机纷纷问世。我国的台湾 TUTANG(大同)公司也加入了 SPARC 兼容机厂商的行列且是最早通过 SCD 测试的厂家之一。国产华胜 4065,华胜 4075 系列也为 SPARC 兼容工作站。SUN 还鼓励为 SPARC 机器编写各种应用程序。据报道,目前世界上有一千多个应用软件商在为 SPARC 的机器编写软件。SUN 的这一举动,在工作站领域掀起了一个类似八十年代 PC 兼容机的浪潮,这时兼容机厂商特别是对经济力不强且要求自主开发的我国大陆是一不可多遇的机会。由于兼容机市场的形成,使得开发、生产和销售 SPARC 机器具有最小的风险,无论经 SUN 的成功或失败,该兼容机市场都不会受到致命的冲击。

考虑到我国的国情,国内欲开发具一定规模 MIS 或 CAD/CAM 系统的一般企(事)业单位,硬件平台选用 SPARC 工作站是理想的。经济状况较好的可以选用 SUN SPARC 工作站,经费较紧张的可以选用 TUTANG SPARC 工作站,也可以用国产华胜 SPARC 工作站(硬件网络可采用客户/服务器的模式)。这不管是从操作系统的角度还是从应用软件的数量来看,都是符合国情且是合拍于国际主流的,它不会有来自市场变动、公司间的兼并所带来的冲击。

· 书讯 ·

《DOS/BIOS 功能调用及程序实例》

DOS2.0~DOS5.0/EGA/VGA/CGA/CEGA/CMGA《DOS/BIOS 功能调用及程序实例》一书除功能调用口十分齐全外,重点放在程序实例上(在调用口后,给出一个或多个完整的如何使用该调用口的汇编语言程序实例)(程序实例有详细的汉字注释)。

书中除包含书名所列出的内容外,还含有:尚未公开的功能调用,网络功能调用,DOS 隐含功能调用,鼠标器功能调用,实时时钟功能调用,CCBIOS/GWBIO/CCBIOS2.013/CGE400 功能调用及扩展内存和扩充内存功能调用等,是同类书籍中内容最详细和全面的(各类调用口都包含了最新内容)!

要在 EGA/VGA/CEGA 等卡上进行图形操作,则必须要掌握其寄存器的正确控制。为此,书中详细列出了 60 多个寄存器的位定义,并给出了利用寄存器进行图形操作的程序实例,本书既满足了高级程序员的需要,又适合于初学者。

书中还附有“EGA/VGA/CEGA 图形压缩存储和还原显示”,“汉字和英文字符放大”,“硬盘故障诊断与修复”和“把汉字库放入扩展内存和扩充内存,然后在用户程序中调用”等汇编语言源程序。全书约 150 万字,定价 22.5 元。欲购者可直接汇款到(610041)成都人民南路 4 段 11 号附 1 号《计算机应用研究》杂志社读者服务部唐大利收,请务必注明收件人、汇款用途、通讯地址及邮编。

集中分散型综合控制系统(DCS)刍议

崔振远

河南省计算中心 (郑州, 450003)

摘要 本文较为详尽地介绍了集中分散型综合控制系统的概念、特点、国内外现状,探讨了我国发展 DCS 的策略。

关键词 DCS CIMS PLC PC 动态 发展策略

一、集散系统简介 集散系统是集中分散型综合控制系统 DCS(Distributed Control System)的简称。是伴随“4C”技术(Computer, Control, Communication, CRT)的发展,在小型工业控制机和模拟控制仪表的基础上发展起来的分级、分布式计算机控制系统。这种系统不仅实现了控制功能分散和操作管理集中,并且具有比模拟调节仪表灵活,比单纯的计算机系统可靠,且取代了常规的冗长的仪表盘,减少了现场接线和安装,能适应多样化现场和高度自动化需要等等优势。DCS 诞生十几年,迄今已发展到第三代产品,它是与计算机等现代电子技术同步而发展的,过程控制的高可靠性要求导致“危险分散”设计思想的萌发、集中处理信息则是计算机传统的工作方式,而两者结合形成的分级递阶式,分布式体系结构就是集散系统的脊柱。

由于 DCS 所特有的组态灵活,易扩展,易变更,高可靠,强功能等特点,而赢得了各行各业的欢迎。成为各行业自动化的优秀设备。DCS 操作简便,系统集成容易,人机界面友好。特别是近年来具有高可靠性、高性能价格比,采用 MAP 通信标准的新一代 DCS 产品相继问世,使之越来越吸引用户。无论是单台设备,还是大型联合企业都可以采用该类系统完成其控制作业,所以它是计算机集成制造系统(CIMS)的首选设备。在国外,已形成功能很强的第三代产品,它的理论、技术已趋向成熟,产品的市场销售始终保持上升趋势。

我们认为:DCS 在今后的十五年内,将成为我国过程控制领域内最有发展前途的设备,而由独立的集中控制发展到连成网络的分布、分级控制将是工业过程控制的必由之路。

DCS 控制功能、不仅适用于连续控制过程,而且也能满足断续批量控制的要求。除具有 DDC 反馈控

制功能外,还具备顺控功能和 BASIC 功能。DCS 为实现诸如前馈控制、采样 PI 控制、交叉限幅控制、解耦控制和 Smith 补偿控制等先进的控制方法提供了良好的环境,使之在实际中得到广泛的应用。

DCS 能将过程控制自动化、测试自动化和信息管理自动化结合起来,使计算机集成制造系统(CIMS)得以实现。据专家们预测:未来分散控制系统将取代绝大多数仪表控制而占统治地位,凡是不能与标准通信网络对接联网,不能与 CRT 等现代化人机接口设备联接对话的单台仪表,都将逐步淘汰。因此研究分散控制系统,发展 DCS 设备是我国自动化行业的一个重要方向。

二、国内外状况 分散型控制系统现今已成为过程控制的主流产品,国外发展大体可概括为以下三个阶段:

一九七五年至一九八〇年为初创期,出现了日本横河电机的 CENTUM,日立公司的 EX-1000,美国 Honeywell 公司的 TDC-2000,Foxboro 公司的 Spectrum, Bailey 公司的 N-90。这一时期产品有很大局限性,它们由简单的过程控制单元,数据集中装置, CRT 操作站,监控计算机、数据转换通道组成。过程控制单元控制功能单一,系统结构灵活性差,网络通讯协议非标准化,系统可靠性不是很高,智能化程度较低等。

一九八〇年至一九八五年为成熟期,出现了美国 Honeywell 和日本山武—Honeywell 公司的 TDC(S)-3000,日本西屋公司的 WDPF, L&N 公司的 MAX-1,日本横河电机株式会社的 CENTUM/B/D,日立公司的 EX-1000A 和 HX1000,西德西门子公司的 Teleperm-M 等,这一时期的产品经过不断完善,向着高精度,高可靠性,高可维护性及可用性,小型集成化、模板模块化,智能化、功能多样化等方向发展。

其特点是系统结构独立,控制功能多样化,数据通信标准化,人机接口智能化,系统结构模块化,应用系统设计面向问题透明化,对于工业控制过程,DCS的自动控制功能更加完善,但对于工业生产管理,还局限于生产过程自动控制的信息管理。

1985年以后为扩展期,其标志是计算机集成制造系统(CIMS)和计算机工厂综合自动化(TFA)的出现。CIMS是利用计算机网络技术将过程控制和信息管理系统紧密结合起来,形成控制管理一体化的集散控制系统。TFA是借助计算机网络技术把生产装置自动化、工厂管理自动化、办公室自动化联成网络而形成的一个完整的系统。这一时期的DCS产品把过程控制、监督控制、管理调度有机结合起来,开始采用专家系统(ES)、MAP(制造自动化协议)和TOP(技术与办公室协议),标准网络通讯协议及表面安装技术SMT。第三代DCS产品的性能价格比有极大提高。现场控制单元(CU)的控制与运算能力更加强大,普遍采用了32位微处理器,系统可靠性通过全面冗余措施和自诊断功能得到了大幅度提高。主要产品有:CENTUM—XL, WDPF I、II、TDC—3000LCN, Foxboro公司的I/A Series,及INFI。

虽然DCS问世不过十多年,但它的发展速度却是自动控制技术发展史上前所未有的。到目前为止,世界上绝大多数仪表公司几乎都卷入了这场空前激烈的技术竞争,推出了各种各样的DCS。可以说DCS已成为过程控制仪表装置的主流。在电力、冶金、化工、建材、轻工、纺织、医药、食品等各行业,都有采用DCS的厂家。他们利用DCS提高自动控制与管理水平、提高经济效益和社会效益,由于DCS可将控制与管理结合成一体,因此可以解决许多仪表系统无法实现的复杂控制、实现各种生产和经营信息管理自动化。

我国分散控制系统的研究开发起步较晚,七十年代末期开始开发,经八年多努力,研制出了现场控制站一级的装置,并在一些工厂推广应用。如重庆工业自动化仪表研究所DJK—F系统和上海工业自动化仪表研究所DJK—100工业控制计算机系统等。但从总体来看,这些系统都不具有分散系统综合控制和监视的功能。八十年代后期国内几个大型仪表厂相继从日本引进生产了单回路控制器,并相继批量生产投入市场。国家也向国内有关科研所、生产厂以及高等院校下达了一批重点攻关项目如“分散型控制系统与工业控制局部网络”、“工业过程控制系统应用工程开发研究”等,还有一批软件工程项目。同时,国内也引进了一些成套系统如:北京自动化技术研究所引进了美国Bailey Control的Network90,上海

调节器厂引进了美国Westinghouse的WDPF,西安仪表厂引进日本横河的Centum,四川仪表十八厂引进日本山武—Honeywell的TDC3000;以及上海—Foxboro合资厂共同生产英国Foxboro的spectrum和I/A系列等分散控制系统。目前正处在消化吸收的初始阶段。同时我国石油化工部门也引进国外十几个厂家的十几种型号的分散控制系统应用于工业生产。

为了推进工业控制计算机的应用和向高层次技术发展,目前正由重庆工业自动化仪表研究所筹建“机电部工业控制实时软件开发中心”。

三、DCS技术动态 目前,国际上DCS生产公司都沿着设计—提高、完善—重新设计—继续完善的发展道路不断创新,其主要动向如下:

1、系统开放

DCS发展初期,各公司独立开发自己的产品,缺乏统一的标准,致使产品之间不能互联,成为DCS向高级协调发展的关键障碍。其次,企业自动化要求DCS与PLC—programmable logic Controller(可编程逻辑控制器),及PC—personal Computer(个人计算机)之间取长补短、互相渗透、互相融合,特别是今天DCS被置身于企业总体自动化的CIMS背景中。系统开放,开放系统互联是对DCS提出的基本要求。因此,网络标准化、向上兼容、软件商品化,加强通讯功能,这些都反映了DCS的一种趋势——开放。当前,通讯及网络技术已不存在太大问题,可是关键需要统一口径,需要一个用于企业自动化的通讯标准协议。这是来自用户的要求,他们希望来自不同厂商的产品能熔于一炉。由于MAP(制造自动化协议)的问世,这个问题得到部分解决。国外已有近两千家厂商参加了MAP用户集团。这体现了他们的开放意识。目前,制造商都十分注意使自己网络靠近国际标准协议。如RS3、TDC3000、INFI等DCS系统已能够做到和Ethernet, MAP, TOP(技术与办公室协议)等标准协议相连接。

为了扩展自身的专有系统,不少制造厂(公司)将网关(gateway)加到专有网络与普通网络之间,有的在公共总线上建立一个系统,如DEC的VAX系统。另外,日本横河公司的CENTUM—XL, Foxboro的I/A Series等系统产品都有较强的系统开放互联能力。

2、系统规模

在分散系统发展过程中,似乎有两种截然相反的观点,有的人认为系统“越小越好”;也有人认为“越大越好”。笔者认为绝对说“越小越好”或“越大越好”,都有失偏颇。根据国内外企业自动化发展需要,

DCS 的规模也将全方位发展。事实上大小规模的设备系统是无法相互取代的。对于大规模现代化企业来说,大规模的 DCS 是必要的。但是在大型系统发展的同时,DCS 规模向小型化发展则是明显的趋势。其原因是中小企业数量及采用 DCS 的越来越多。因此高性能、低成本的小系统为更多用户所欢迎。随着硬件技术进步,半导体技术提高,出现了小型的性能价格比更高的处理器及其配套的芯片,为系统的小型化创造了良好的条件。这种体积小、规模小的小系统有可能具有功能强、高可靠的特点;并可容易地升级为大系统。另持系统“越小越好”观点者还认为:目前已进入实用阶段的称之为“启动设定品”的小型控制装置,允许用户从小规模系统入手,通过可以变更系统结构的 CRT 图示操作接口,组成规模较大的系统。这种需求来自以下几个方面:一是因资金关系无力购置昂贵的大型系统的中小企业。二是需要大型系统的用户出于以下原因,先购置小型系统:①出于技术练兵的需要;②出于了解 DCS 经济效益、技术性能和可靠性的需要。综上所述:DCS 将向大小两个方向发展。但从我国现状来看,今后数年内,小型系统的需要比重将占绝对压倒优势。与此同时以单回路、多回路调器为基础的初级系统作为 DCS 的一个分支也将得到适度发展。

3、吸收 PLC、PC 技术,嵌入 PLC、PC 产品

可编程控制器最初是为机械制造业顺序逻辑为主的控制任务而设计的产品。经十几年的应用发展,已具备了较强的通讯、网络和控制功能,它比 DCS 更灵活,价格更低廉,更能耐恶劣的环境,已占领了广阔的市场,赢得了大量的用户。因此,DCS 吸收融合 PLC 的技术特长是十分必要的。比如增强模块内批处理和逻辑功能,降低数字 I/O 成本即是吸收学习 PLC 技术特点的一例。其能否制造 DCS、PLC 之间的接口,解决相互通讯、将 PLC 嵌入 DCS 之中,都是 DCS 性能优劣的一个重要指标。

个人电脑以其有丰富的软件,众多的用户引人注目。大量的信息管理和办公室自动化工作也都是由 PC 完成的。DCS 发展,使其功能由单纯的过程控制向生产管理自动化发展。因此 DCS 能直接采用 PC 技术及其软件是十分有益的。为此,目前 DCS 大部分高级产品都能直接把 PC 嵌入系统。在 DCS 内发挥 PC 的优势。比如可以让 PC 完成离线和在线任务,也可充当上位机,充当操作窗口,甚至直接充当过程接口和控制设备。如 Foxboro 去掉了 PC 中的 DOS,加入实时和应用双操作软件,使之成为 DCS 的工程工作站。

4、DCS 的软件

经历十几年的应用积累,DCS 的软件功能十分

丰富。它除能完成数据收集、传递、加工、处理及复杂的控制外,目前较优秀的 DCS 数据库中,通常具备数据采集、数据趋势分析、数据史料处理、公共数据库管理、报警、网络接口、MRP 象元图形库、自整定、SPC/SQC(静态过程或静态质量控制)、AI(人工智能)、专家系统等功能。

5、产品兼容问题

近些年来,国外公司不断推出新的 DCS 和设备。但用户觉得无法适应这种快速变化,希望新、老产品之间,以及不同生产厂的产品之间有一个灵便的桥梁使之相互过渡,这就是所谓的产品兼容和连续问题。鉴于此,制造厂商面临着解决自己新老产品的兼容问题,更面临着解决与其它厂商产品的联接和通用问题。如前所述大部分厂商正在或已经部分解决了这一问题,比如研制网关(gateway)即是一例。

6、其它相关技术

DCS 产品的质量及性能已有相当水平,随着应用领域的扩展及相关技术发展,以下诸方面仍将受到 DCS 制造商的重视:提高用户界面的亲和性,实现操作简单化和效能化,例如语音输入/输出操作;提高 DCS 的智能化水平。如系统辨识、系统仿真、优化控制、智能化传感器及执行器;现场通讯的实现,系统间交互图示功能;加强标准化设计,实现不同机型物理联接和逻辑联接;人——机联系、系统扩展、机体尺寸等诸方面高度规范和标准化;高性能的 CPU 及海量存储器的采用,进一步提高 DCS 的性能和速度。

再如改进模块化硬件结构,做到一件多功能,降低造价,减少备件,维护方便容易;光纤通信的采用,提高数据传送速度;提高控制器的性能,使之具备自适应自调整功能,实现最佳控制;优秀的算法充分的采用等都是制造商注意的重点。

四、我国发展策略 国外 DCS 发展很快,而我国尚处于起步阶段,如何加快发展,迎头赶上,是需要探讨的问题。

1、制定积极优惠的政策,促进 DCS 迅速发展

因为 DCS 是一种多学科领域,高科技集约化技术产品,抓住了它的发展,即可实现高起点的技术飞跃,并带动多种行业高技术产品的全方位发展。对于这种关键技术,国家应有优惠政策给予重点支持。比如对于发展我国 DCS 为目的的技术改造及 DCS 国产化,给予积极支持,在投资、税收上给予优惠。

2、在技术开发方式上采取科研单位、高等院校与工厂联合攻关,联合试制上马。

3、在我国应以小型系统的开发与应用为主,可引进国外先进的第三代小型系统,结合应用,研制开发适合我国应用的成套产品,逐步形成 DCS 产业。对单纯地以应用为目的引进,应予以限制。对于大型企

业需要引进 DCS 产品可采取与科研单位联合引进, 在应用过程中联合研究, 进而发展自己产品, 联合推广应用, 效益共享。

4、由于我国 DCS 技术相对落后, 而与此相关的基础产业也无太大优势。为此短期内形成我们自己的 DCS 国产化尖端产品比较困难。但软件技术占 DCS 技术比重较大。而我国在软件开发上有较大优势。为此, 我们应把技术重点放在软件研究开发上。这样有可能形成我们的 DCS 技术优势, 并在世界 DCS 技术市场上占居一席之地。

5、发展配套产品和技术

DCS 有较多的配套产品和技术, 虽然在整体上我们处在劣势地位, 但在某些技术、某些产品上我们

有一定优势, 应充分发挥我们这方面优势, 使之进入 DCS 技术、产品市场。为此我们必须注意把这类产品及技术标准化, 易于和国际上流行的 DCS 产品配套。

6、发展与此有关的技术理论

如现代控制理论和大系统理论应用、系统仿真技术、实时软件和工程化软件技术、控制与管理集成系统设计技术、系统可靠性与适应技术、设备故障诊断与信号处理技术。这些多学科互相渗透而形成的新技术群体, 对 DCS 的发展革新具有重大影响, 且在我国有一定的研究基础。实际上不少研究成果被国外无偿采用了, 十分可惜, 今后应注意有针对性研究和促进其应用, 并高度重视经济技术情报的保密工作。

计算机辅助设计及其未来

张 鸿 鸣

吉林省计算机技术研究所 (长春, 130012)

摘要 本文概略介绍了 CAD 的算法、软件设计方法、最佳系统的选择, 展望了 CAD 的未来发展趋势。

关键词 CAD 算法 模型 软件 接口 最佳系统 VLSI CAM

计算机辅助设计最引人注目的是: 它在技巧和算法方面的发展。其中最著名的贡献是于 1963 年 I-van Sutherland 发展的“描绘部”技术及通用汽车公司在 1962 年所发展的 DAC-1 工程。八十年代开始, 国内有些技术领域的 CAD/CAM 计算机辅助设计软件也开始进入大发展的阶段。初期的计算机辅助设计软件的主要贡献在于它们显示出了计算机技术完全可以用于重要的设计和制造的环境之中去, 并且为未来的计算机辅助设计软件的研制和开发开辟了广阔的道路。

一、计算机辅助设计的算法 在辅助设计的软件中, 一个算法就是一组根据数学与几何方程式而建立起来的法则和程序, 并且用以解决问题或从一些可行的步骤中求找出一个已知的答案。最基本的法则是从一点开始(当有了点之后), 继之而生成的就是线。在图形学中, 一个设计的最简单模式就是所谓“基元”, 可以是一条线、一个矢量和一个点甚至可以是一个字。由点到线, 再到较高层次就是平面和多边形。一个多边形是由多条线端对端连接起来组成的。当我们将颜色涂于多边形的图案之中之后, 它就马上变成了所谓“实体”状的模型了。

1、曲表面。除了点、线和多边形之外, 还有曲表面的圆锥体和自由形式的曲面, 正如圆形、弧形、椭圆形、抛物线形和双曲线形等。自由形式的曲面是由绘图系统根据一串可以适应于一个算法的点所构成的图形。

2、几何属性。从表面来看, 在计算机显示器上绘图与传统的在绘图板上绘图并无任何区别。但是由于几何性能有了可变的属性, 使用计算机的工程技术人员在计算机的显示器上可以随意更改或加入新资料, 例如线阔、密度、颜色和可见度等等。

3、几何学上的变换。计算机的另一个辅助设计软件的特点就是可以作几何学上的变换:

- (1) 转换——物体的线性运动。
- (2) 旋转——物体的角度运动。
- (3) 定标——增加或减少物体的体积。
- (4) 坐标系统的变换。
- (5) 窗口(将一个矩形边界围住特定的区域)。
- (6) 剪辑——除去窗口以外的部分。

4、三维影象。三维影象可以运用独特的算法在计算机荧光屏上显示出来, 例如: 使用隐藏面算法可以依据几何学上的光线追踪原理创造出实体的影

象,光线可以根据数学上的定理将实体的影像在计算机屏幕上显示出来。通过进一步精心处理,它还可以提供明暗、反映及透明面等高水平视觉效果。只要工程技术人员需要,计算机辅助设计应用软件便可以增加新的特点满足各个领域用户的需求。

5、几何模型方法。初期的计算机辅助设计软件只能实现“线框”显示,在计算机显示器上以线显示实际图形。当用户想显示立体图形时,这个方法就产生一个问题—模型的后部分也会同时在屏幕上显示,自然使用户产生混乱。之后发明的隐藏线算法,可以解决这种不必要的混乱。因为运用了较简单的几何定理。“线框”便可以较快及很容易的组成,此法至今仍是最实用的技术。

6、表面模型。表面模型可以有效弥补“线框”及实态模型的不足。壳状体和薄的塑胶部件等这些复杂的三维几何图形可以用各种不同的表面模型来实现。表面模型要比线框和实体模型的成本低。运用光线追踪、切与和及积分的技巧,物体的质量可以利用表面模型计算出来。

7、实体模型。一个在交互式计算和绘图系统运行的实态模型系统中,我们可以很容易地设计出三维影像机械部件。在技术上,以计算机展示的全封闭三维形状完整而又清晰。运用实态模型,完全可以保证在计算时不会忘记设计某部分零件的问题,但这类问题在二维设计和线框设计时时常发生。

在发明实态模型方法之前,二维半设计和线框设计虽然可以改善绘图人员的绘图能力,但是工程技术人员要进行模拟(如数控方面的刀路分析)试验时,迫使他们往往要重新设计或加入新的数据,才可以开始试验。假若运用实态模型法,这些试验便可自动地进行了。

最基本最实用的实态模型方法就是构造性实态几何(CSG)及其边界表示法(B-REP)。构造性实态几何可以表现的形体有正方形、圆柱形和印模,锥切积木一般加入或抽出这些基本形体,一个新的组合体就是依据布尔逻辑指令创造出来的。运用边界表示法,物体的面、边和顶点就可以根据表面的各个数据,通过适当地连接(拓扑学)就可以构造成模型。

二、计算机辅助设计的软件 由于出现了更先进更新一代的计算机,它的功能强容量大,可以运行更先进的计算机辅助设计软件,工程技术人员可以将他们的设想、模型和算法存入计算机之中。从此,计算机辅助设计软件可以使工程技术人员最大限度地发挥计算机的功能。运用它工程技术人员不仅可以绘画模型而且还可以在屏幕上看到三维空间的物体,并且可以旋转、分析、组合和设计每一个零件。

工程技术人员并不必是计算机专家,他们完全可以在新一代计算机辅助设计软件的帮助下完成各

项工程设计工作。

Medusa系统和AutoCAD系统是当今市场上最流行的计算机辅助设计软件,也是比较典型的辅助设计软件,它可以用于二维或三维的设计、绘画和文件的编制。这些系统都由多个可以综合独立模块所组成,其中最基本的两个模块是绘图设计和模型设计。

(一)绘图设计。绘图设计(二维)模块可以提供几何建造能力,用户可以自如加入注释或变动设计的体积,从而可以减少花费在创作和修改图形需要的时间,大大地提高了绘图速度。

(二)模型设计。模型设计(三维)模块能使用户在设计很复杂的物件时,便于分析和看见实态的模型,从而大大地提高了工程技术人员的绘图能力。其特点如下:

1、可见性。

(1)真实态模型(边界表示法)所显示的部件。

(2)实体中的布尔式运算,包括合并、相交和差异。

(3)用户自定的视域,包括等距、正投影、辅助部分及透视。

(4)自动搬移隐藏线。

(5)影视观察,包括投影、透明和雾影观察能力。

2、绘图与设计。

(1)自动尺寸(英制或公制单位)标注控制,并有各种不同尺寸标注标准。

(2)产生部件目录、线目录或其他各类报告。

(3)翻译图纸上的文字,使之符合不同国家的需要。

(4)可以改动设计参数。

(5)标明了设计参数的图形意义。

(6)自动生成带平面或圆柱形弯头的金属片部件的平面模型设计图。

3、用户接口。

(1)备有支持它的操作系统和硬件。

(2)用户可以用C语言等高级语言编制和修改绘图数据库。

(3)灵活的项目单用户可以很容易制定。

(4)备有对其他应用软件包的接口,并可以用作绘图的数据控制。

4、其他接口。典型的计算机辅助设计软件资料完全可以传递到一个先进的图象式数控软件包,可以构成计算机控制多达五轴的机械工具。正在运行的机器可以在设计工作站中先行作模拟实验,经过后处理出来的数据可以输入图象式控制软件包或打印机(或磁盘机)之中。

还有其他辅助设计软件包的接口,用以减少错误设计。同时,工程技术人员还可以用来分析软件包,并可以一起进行设计。有些软件包还能提供结构

学、热力学和材料力学上的分析方法。

5、计算机集成制造。典型计算机辅助设计软件不仅可以同数控和分析软件包连接,它完全还可以同工厂控制、企业管理和信息管理连接。由于采用了模块设计的缘故,每个不同领域的计算机集成制造的软件也可以根据需要逐次加入。

三、计算机辅助设计最佳系统的选择 计算机辅助设计对我国来讲还是一项较新的技术,选择一个最佳的计算机辅助设计软件(系统)还是一个值得重视的问题,必须有计划地全面考虑。我们认为,要想选择一个最佳的计算机辅助设计软件(系统)应着重注意以下几个问题:

- 1、市场证明可靠的系统。
- 2、单一厂家(公司)的系统。
- 3、汉化实用,兼容性好,保证未来的发展。
- 4、模型设计完全可提供未来的扩展。
- 5、计算机集成制造接口要适用、可靠。
- 6、具备提供特殊应用的符号库。
- 7、开放系统可以应用产销需求来设计自己的项目单,执行程序 and 符号。
- 8、通用硬件可以提供其他用途。
- 9、培训技术力量强,文件齐全。
- 10、技术服务力量强,维修方便。
- 11、能提供全面的计算机辅助设计系统方案。
- 12、操作方便,便于维护。

四、计算机辅助设计的未来 在发达的国家中,计算机辅助设计技术在教育、能源、铁路、桥梁、矿产、机械制造、造船和航天等各个领域得到了广泛的应用。凡是需要图形设计的各行各业都用计算机辅助设计代替了手工制图,大大地提高了绘图质量和能力。不仅发达的西方国家需要复杂的计算机辅助设计软件(系统),在发展的亚太地区国家和中国也同样需要计算机辅助设计,将 AutoCAD 和 Medusa 系统(软件)作为高技术的绘图工具,并且在科学工程上开展了广泛的应用。

在国内,由于微型机的大量引进和国产化微型机、中小型机的投入生产,为计算机辅助设计的发展打下良好的基础。特别是引进了先进的 CAD 工作站更促进了计算机辅助设计的应用和发展。其次,由于国内的开放市场经济,竞争形势日益激烈,要求更高的生产效率和产品质量,以获得较高的经济效益,从而需要尽快开发出更多的计算机辅助设计软件(系统),以形成能力,研制出性能优异和兼容性好的计算机辅助设计系统产品,随着计算机辅助设计技术的广泛应用,使用单台微机和工作站的计算机辅助设计应用环境已不能相适应,已经开始提出联机、联网和建立完整系统的环境要求。我国电力工业的计算机应用已由提供计算的单机单项作业转向具有各

种目标的多机联网综合性系统化信息处理。同时,在电力设计方面已将现有的 32 位超级微机 CAD 工作站和各种微机组成三层计算机网,并配备有齐全的工程设计分析计算—CAD 应用系统、CAD 系统软件和支持软件,可以预测电力设计部门的联网计算机和 CAD 计算机辅助设计系统(软件)的需求量将会呈几倍的增长。

在计算机辅助设计技术的开发和应用的过程中,人们开始提出急需解决和改进的问题:

1、由于 AutoCAD 辅助设计软件的出现,必须具备各种专用于管理计算机辅助软件 AutoCAD 标准图形文件的实用程序。

2、提出计算机图形学的理论、方法和软件实用技术及其在计算机辅助设计中的应用。进一步开展研究图形算法、三维真实图形、几何造型、人机交互接口和图形软件的工具与环境,以及图形学在机械、数字系统计算机辅助设计中的应用。

3、必须进行人工智能计算机辅助设计的研究。目前,这项研究在有些领域中比较活跃,如大规模集成电路 VLSI 设计自动化的智能方法(VLSI 专家系统)。这种方法认为,人的知识是能掌握并可以存储到一个专家系统的知识库之中。知识形式的试探能处理 VLSI 设计问题中的真正复杂性,以调整与修改设计方法,减少试探次数。因此,随之而来数字系统或电路也就越来越复杂了,故障诊断和测试已不是传统算法所能解决的,必须探讨新的智能方法。当前,人们主要侧重研究专家系统方法的试验性测试生产软件,以及基于知识的内部自测试设计工具。另外,由于印制电路板的布线算法达不到令人满意的要求,人机对话时间太长,促使人工智能的 PCB 布线方法的研究提到议事日程上来了。

国内的计算机辅助设计技术发展的还很不平衡,有些技术领域发展得很快,但也有些领域还没有行动,CAD 软件发展得很快,CAM 技术还较落后。国内市场 and 大多数厂家(公司)开始注重技术和产品的开发,计算机辅助设计技术的发展将处于有利的地位,特别是在电子和机器加工行业将会迅速的发展,对 VLSI 的设计和加工的需求已不是遥远的未来了。

计算机辅助设计软件(系统)正在不断的发展,已由二维到线框,直到实态模型,全自动的由线到实态模型的转换将会很快实现。下一步的发展将是人工智能技术的运用,未来以分析与试验的软件将会更有效地发展起来,而且其操作步骤将会变得更为简易、方便。在将来的各个技术领域中,将会出现为机器人、数据控制及试验等项目设计的计算机辅助设计的专家软件(系统),将会使全自动控制的无人工厂的梦想变成现实。

《计算机应用研究》第 9 卷(1992 年)总目录

软件篇

用高级语言设计的数据库报表输出通用软件·····	向南平(1.1)
用高级语言管理 dBASE III 数据库结构·····	向南平(1.4)
人体肾血浆流量微机计算方法·····	张鸿鸣 刘铁军 田卫星 薛 赤(1.9)
一种交互式快速原型图式语言——RPSGL ·····	马 江 黄 玲(1.11)
选择组合查询的方法及其实现 ·····	田丽娃 刘鲜京(1.15)
多窗口用户界面的设计与实现 ·····	史品志 刘文明 牛连江(1.17)
通用函数曲线作图程序 ·····	谢进一(1.19)
中文菜单的设计方法 ·····	曾向阳(1.21)
IBM-FORTRAN 语言屏幕彩色字符显示功能的扩充 ·····	张 义(1.25)
一个 Turbo-c 的多窗口大程序框架 ·····	唐常杰 韩仲清(1.27)
在西文 DOS 下显示汉字的一种方法 ·····	李日林(1.29)
动画显示程序设计及在模态分析中的应用 ·····	朱才朝(1.30)
DBASE III 实现的中文屏幕游动显示 ·····	彭 立(1.32)
DOS 系统下的脱机输出 ·····	刘锦林(1.33)
FORTAN 语言图形功能的扩充 ·····	邵庆良 沈尔秩(1.34)
一种新的数据流图 DFD 划分方法 ·····	李 强(2.1)
基于平面网格划分的自由曲面成形算法·····	马在强(2.5)
KERMIT 的文件传输功能 ·····	符华儿 林 华(2.7)
绘制曲线的实用程序及其设计方法 ·····	班先林(2.14)
TANGO 绘图软件功能的进一步开发应用·····	李家武(2.16)
PC-1500 计算机通用框图绘图程序 ·····	穆安民 朱文嘉(2.20)
人体青光眼微机辅助诊断方法 ·····	张鸿鸣等(2.22)
利用 GAL 保护 EPROM 中的程序·····	赵英俊 胡 阳(2.24)
复杂图像生成处理显示的一种新方法 ·····	朱 鹰 朱 焱(2.27)
图像放大技术探讨 ·····	贾英江(2.30)
一个具有 PROLOG 推理机制的演绎数据库·····	梁庆龙 陈 康(2.32)
机床 CNC 中的定位软件设计方法 ·····	李天健(2.35)
自动纵横调整舍入误差的简捷软件 ·····	曲少宏(2.37)
化工设备信息管理系统 ·····	刘鲜京等(2.39)
网络工程中数据库应用系统设计技术 ·····	王敏生(2.42)
借毒加密法 ·····	符云青(2.45)
关系代数的分块自然联接运算·····	罗伟其(3.1)

关系数据库关系规范化支持系统	宾晓华(3.5)
数据库设计自动化中的一种最佳编码方案	李彦明 黄声烈(3.8)
GKD-PROLOG/SUN 模块系统的设计与实现	严静东、金 芝、吴泉源(3.10)
高级语言中越级返回的实现	谢果然、李 萍(3.15)
软磁盘文件加锁及高道局部特殊格式化扇区缝隙加密的技术	曹尔强、张 沂、李宝岩、潘继宏(3.17)
EGA/VGA 图形方式下 FOXBASE+ 屏幕叠加式菜单的实现	吴邦忠、夏 英(3.19)
一种用于自然纹理图像分割的并行算法	管旭东、刘健勤、王爱群(3.21)
微机上用 Turbo C 实现的动画技术	何汉武(3.23)
一个新颖生动的 XOR 动画实例	鄂大伟(3.26)
Auto CAD 与其它程序快速交换图形数据方法	王建华(3.29)
绘制逼真图形的捷径	江 涛(3.31)
利用华光交互式图表编辑排版软件绘制框图	陈 康(3.32)
0-1 整数规划的枚举算法	陶友传、唐泳洪、段正澄(3.33)
C 语言与软中断方式实现读写端口信息	黄声烈、吴庆妍、赵冬林、李志英(3.34)
2.13 汉字系统实用外词组文件建立与分析	瓮正科(3.36)
微机预测注射剂和滴眼剂有效期的方法	张鸿鸣、刘铁军、孙玉华(3.40)
从硬盘中恢复 2.13H24 点阵字库备份盘的一种方法	李修连(3.7)
通用论文稿打印程序	房红兵(3.28)
论 CIMS 的软件设计方法学	董 颖(4.1)
一种会计电算化的实现方法	楼永新(4.9)
微机辅助诊治人体骨伤的方法	张鸿鸣、刘铁军、张晓楠(4.11)
数据库 MIS 集成环境的设计方法	吴 春、刘 萍(4.14)
MILIS 系统软件的应用与更新设计	刘启茂、方 红、蔡红梅(4.17)
专家系统工具设计入门	卢中清(4.20)
汇编语言速成编程法	吴庆妍、赵冬林、李志英(4.24)
一个浏览数据库内容的通用程序	刘定君(4.26)
用 FoxDoc 自动生成数据库管理系统文档	王文涛 袁秋丽(4.28)
软磁盘零磁道信息的自动备份与恢复	蒲昌平、邓 波(3.30)
磁盘逻辑损坏后数据的恢复	邓 波、蒲昌平(4.34)
介绍一种新发现的屏幕图形拷贝命令	胡仲奎(4.36)
分布式数据库管理系统一种开发方法	成汝震 张 斌 阎立伟(4.38)
联通打印的一种分页技术	孟 萍(4.40)
计算机应用系统的画面生成技术	陶文正(4.41)
以鼠标器为热键的 TSR 程序的编制	刘传憬(5.1)
TANGO 软件汉字的快速安全输入方法	胡 阳 赵英俊 程必达(5.5)
用 Turbo c 实现应用系统软件中的汉字文本编辑器	任清珍 黄天戌(5.7)
冷冲模计算机辅助设计中的冲裁工艺性判定	李 秦(5.10)
微机预测人体乙脑流行趋势	张鸿鸣 刘铁军 薛 赤 程耀华(5.11)
实现 NOVELL 局域网定义的共享打印机与 CCDOS2.13H 全兼容	赵世荣(5.13)

产品决策系统应用软件的设计与研究	闫立恒 闫 英(5.15)
Cal Comp 数字化仪 ADI 驱动程序的安装方法	瓮正科(5.18)
汉字信息传递及检索问题探讨	蔡红梅 刘启茂(5.21)
扩充屏幕和扩充键盘控制全貌	张志林 川 子(5.24)
怎样快速搜索磁盘目录中的文件——文件查找程序 MATCH 的实现	李 智(5.26)
键盘绘图程序一例	徐芳之(5.28)
TANGO 软件图形的打印机无级缩放打印	李家武(5.31)
生物实验数据回归建模系统(MSED)的研究	王 东 张永禄 牟 东 向 军(5.33)
也谈多种程序语言的综合应用	李新华 张华磊(5.36)
一种求任意切面轮廓算法	王学崇 李 剑 陈 皓(6.1)
正规文法生成算法及其实现	金之滨 仲肇铭 卢雷(6.5)
计算机递归分组排序算法	周建钦 赵志远(6.7)
Occam2 并行处理程序在 Transputer 网络中的系统配置	马永强(6.9)
微机外部设备的虚拟接口方法	李桂青 赵占朝(6.12)
一种动态门限图像二值化技术	白 焱(6.14)
Auto CAD 中汉字处理软件开发	尹文生 张志坚(6.15)
数据库管理系统文字型数据的类 HASH 编码技术与准顺序式紧缩存储结构	张育民(6.18)
使用 LISP 编程空间不够的问题之解决方法	熊瑞萍 黄永宁(6.22)
一种窗口光条选择录入汉字域内容的方法	滕 锋(6.23)
一组巧妙的 DOS 实用例程	黄庆程(6.26)
微型机在医院病历资料管理中的应用	向瑞红 唐 承(6.29)
一个简单实用的 FoxBASE 函数计算器程序	陈来军(6.8)
西文 Turbo C 2.0 显示汉字的最简单办法	赵炳跃(6.17)
引导扇区型病毒通用清除软件—KV—BOOT	肖俊良(6.28)

系统篇

单片机录像遥控播放系统的设计	陈惠生 侯建新 王德来 白智鹏 王 平(1.42)
条形码技术在生产检测中的应用	姚景平 李 伶(1.46)
IBM PC—IBM PC 文件通信通用程序设计	郭继展(1.49)
8031 单片机 DMA 功能的扩充	周鸣争(1.51)
IBM —PC/XT 中断扩展的具体实现方法	赖红威(1.53)
图书馆自动化与 HP3000 系统	刘启茂 蔡红梅(2.51)
同步整周期数据采集器	潘俊启 陈允西(2.54)
条形码阅读器的设计实践	马在强、朱云、李尚明(3.42)
与 PC 机兼容的专用键盘的设计制作	周 立 赵以钰 王晓红(3.45)
伪彩 B 超中微机的应用和软件的编制	唐 丹 王 沐(3.46)
E ² PROM 在控制系统中的应用	崔正德(3.49)
微机人体心脏智能监护系统软硬件设计	张鸿鸣 邢洪发 田卫星 薛 赤(4.45)
两级分布式数据采集处理及远程有线通信微机系统	徐治国(4.48)

智能多路流量仪表的研制	余立建 王长林(4.52)
采用 8031 单片机的直流电治癌仪	李孔宁 倪华 卢布 陈永朴 王红(4.55)
介绍局域网软件 Star GROUP	符华儿(5.38)
单片机串行接口的应用研究	石志熹(5.44)
一种新颖的单片机开发技术	林立 林良德(5.46)
EPSON M1221 微型打印机的二次开发	林化(5.48)
IBM-PC/XT 与 MCS-96 系列单片机并行通讯系统的设计	王杉杉 胡继胜 马修真(6.31)
点阵式液晶显示器与单片机的接口	赵广涛 盛翊智 吕汉兴(6.34)
对 IBM AS/400 系统没有给出 MIPS 值的几点看法	李春贵 刘铁军 马俊田(6.37)

硬件篇

日历时钟芯片 M5832 与微机接口设计一种实现方法	卢正义(1.38)
一种实用的双机并行通讯方法	周立 王晓红(1.39)
PC 机与 TMS320 系列数字信号处理器的接口设计	王大可(1.40)
关于《具有 HMOS 结构的 MCS-51 系列单片机提供后备电源的方法研究》的探讨	沈新(2.46)
Compaq 386 与 Cosmo Graphic 数字化仪的连接	戴若兰 薛和华(2.47)
通用微机 62 总线扩展板	程黎(2.49)
使用 GAL 的硬加密特性保护 EPROM 中的软件	黄银彪(3.50)
失电后保存内存数据的方法	吴汉文(3.52)
IBM-PC/XT 18 插槽上 I/O 地址的“扩展”	赖红威、赖君利(3.53)
新型双路目标特性模拟信号源	高锋(4.43)
一种新颖的数字信号转换方法	张友生(4.47)
一种计算机光谱采集系统	何桂荣(5.50)
PC-1500 系列机的扩展应用	张明(5.52)
计算机光电隔离技术和双通道长线传输器的设计	左德立(5.55)
微机过程控制系统中的接口与抗干扰问题	李均宜(6.42)

维修篇

微型机的故障维修技术	牛发海 张选阁 张双才(1.55)
一种方便修复故障盘数据的实用程序	阮高华(1.57)
IBM-PC/XT 硬盘不能自举维修一例	李洪涛(1.59)
PC/XT 软盘驱动器故障二例	周力(1.59)
TH3070 点阵式打印机常见故障分析与检修	刘立华(2.57)
采用门阵芯片的 286 机的主板诊断	陈亮(3.55)
CE-150 绘图打印机电力检测电路分析及常见故障处理	潘国军(3.58)
SANKEN 1KVA UPS 的维修	郭毅(3.59)
高分辨彩色显示器故障维修一例	单昶贤(3.59)
解决实时时钟丢失故障一例	刘亮生(3.57)
软盘驱动器读错故障排除	宋靖涛(3.51)

利用冷却法修复微机一例	张 智(3.54)
AR3240 打印机故障维修	陈允西(4.56)
GW300 显示器常见故障维修实例	刘亮生(4.57)
硬盘驱动器故障分析及诊断方法	齐 毅(4.60)
对装有联想汉卡的 IBM-PC/XT 机硬盘启动失败的恢复及启动盘上 DOS 系统的恢复	李 霖(4.8)
谈谈高密软驱中的 Disk Change	胡 红 邹伟(5.57)
对 SANKEN SIO2 UPS 之重要改进	文显林(5.58)
M2024 打印机故障分析与维修一例	徐进敏(5.60)
CASPER 显示器维修分析一例	杨林安(5.61)
LQ-1000 打印机常见故障及维修	李红亮(6.46)
TH3070SL 打印机故障排除一例	陈允西(6.47)
M1724 打印机换行时打印头不能回到左端起始位置的故障维修	陈永红 穆大明(6.48)
电网强干扰损坏微机一例	陈德明(6.48)

信息篇

制鞋 CAD 开发应用刍议——兼谈 SMD-6410 技术特色	马在强 徐一清 张成武 刘 营(1.61)
简讯	(1.封二,8,50)
智能加密卡简介	吴洪宇(2.63)
MS-KERMIT 发展简史	符华儿 林 华(2.64)
推荐一种简易动画画面动态菜单	孙春秋等(2.19)
利用 Z80-CTC 中断实现视听兼容	周伟航(2.22)
简讯一则	(2.41)
计算机图形标准展望	汪厚祥 夏 静(3.60)
简讯 4 则	(3.封2,4,31,48)
重要公告	(3.封3)
决策支持系统的概念结构应用及其发展	王宗军(4.61)
简 讯	(4.29,39,40,42,53)
本刊启事	(4.64)
中国科技论文统计源期刊 28 家杂志名录	(5.封2)
900MHz 无中心选址移动通信系统简介	(5.62)
本刊公告	(5.62)
征订简讯	(5.9,49,56)
RISC 结构及 SPARC 工作站	刘庆枰(6.49)
集中分散型综合控制系统(DCS)刍议	崔振远(6.51)
计算机辅助设计及其未来	张鸿鸣(6.54)
《计算机应用研究》一九九二年(第9卷)总目录	(6.57)
书讯、启事、简讯	(6.25,33,41,45,50,62)

航空航天工业部国营新兴仪器厂

900MHz 无中心选址移动通信系统简介

GC-904 型 900MHz 无中心选址移动通信系统,是八十年代末,国营新兴仪器厂与机电部通信所,联合开发生产的一种新型移动通信系统。

该系统是在吸收国外多种先进样机优点的基础上,根据我国国情,自行设计、生产的国产化 900MHz 无中心选址电台。八九年曾获全国高新技术产品“火炬杯”优秀项目奖;九一年获机电部“科技进步”特等奖;九二年被国家科委、劳动部、国务院引进外国智力领导小组和国家技术监督局,联合评为九一年度国家级新产品。

900MHz 无中心选址移动通信系统与蜂窝公众系统、集群调度系统比较,具有无中心控制和无中心控制组网等优点。组网方便、灵活、投资省,建网费用可节省一半以上。该系统还有多信道共用,自动选取空闲信道,数字选呼与自动接续多址用户等特点。与点频道通信相比,频谱利用率提高 25 倍以上,用户不用申请频点指配。该系统还选用数字调制解调器与纠错编码,大规模集成频率合成和单片微机控制等当代先进技术,因此,编码容量大(除地区号、网号外,网内允许从 00000 至 99999 设置 10 万个身份码),接续速度快。另外,结合我国国情,作了 10 多项技术改进,增加了振铃、回铃、忙音和故障音等功能。同时,设有相应的标准接口,与有关转接器配接,可进入市话网。还可与计算机相连,具有数传、传真、传输静止图象、文字等功能。

900MHz 无中心移动通信系统,可广泛应用于工厂、矿山、油田、交通运输、公安、医疗卫生、汽车调度、旅游等城乡企事业单位,组成各种大、中、小型无中心控制通信网。由于是 900MHz 频段,在地下坑道、工事、隧道中使用,其通信效果明显优于 150MHz 和 450MHz 频段的通信机。

国营新兴仪器厂是航空航天部直属的大型复杂无线电设备制造厂,是国家定点移动通信产品生产厂,现有职工 5000 多人,技术人员 1000 多人,其中工程师近 400 名,高级工程师(含研究员级)近 200 名。工厂拥有国内外一流的测试手段和设备,生产设备精良,工厂管理严谨,质量保证体系完善,销售和技术服务及时、周到。目前,GC-904 型 900MHz 无中心选址电台,已形成规模生产,引进日本先进的 900MHz 无中心移动通信手持机生产线,92 年 9 月已批量投产。此外还开发了中继台、有/无线转接器和背挂式、遥控式、车载台等一系列移动通信产品及其配套设备,以满足不同场所、不同用户的需求。

在给用户提供高质量产品的同时,工厂还完善了售前售后服务制度。设立了“通信测控系统工程处”专门负责该产品的系统开发、销售、系统工程设计、组网、维修服务、技术培训和咨询等工作。热忱欢迎广大用户惠顾并提出宝贵意见。

航空航天部国营新兴仪器厂通信测控系统工程处

邮编:610055

电挂:7190

地址:成都市一环路东一段 138 号

电话:331012-253、669

电传:60123 XIF CN 传真:(028)336298

本刊公告

1、本刊为已登记注册的广告媒介,渠道畅通,覆盖面广,刊载有关电子、计算机产品、成果以及培训、书讯、招聘、展览、技术转让等广告信息,价格合理,效果良好,热忱欢迎各行各业单位、团体、个人踊跃惠顾!

2、本刊承办有关计算机技术资料的印发业务以及计算机开发、培训、咨询任务,竭诚欢迎各界友好、广大同仁光临指导,来函来电来人洽谈业务,开展科、工、技、贸各方面的广泛合作,发挥专业群体互补优势,实现资源与信息共享,收其“红了樱桃,绿了芭蕉”相得益彰之实效。

3、由于本刊已进入中国科技论文统计源,来稿数量迅速增加。为了使新颖实用、有独创见解的学术论文不成为“过时黄花”,得以尽快发表,本刊特辟出一定版面办理“优先快发”业务,有意于此者,请同本刊联系。

4、作者来稿中若有图纸和程序清单,请尽可能采用激光照排硬拷贝输出,以利编辑出版。

谢谢合作与支持!

本刊编辑部