

計算機應用研究

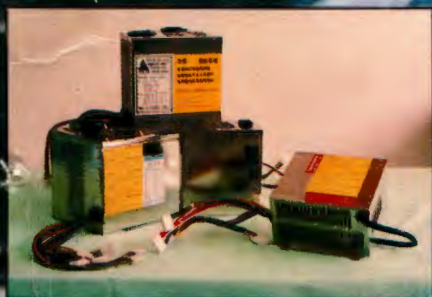
APPLICATION RESEARCH OF COMPUTERS

1994 3

證券交易計算機處理系統
 計算機自動播音系統
 多媒體應用系統
 微機系列軟件
 新潮系列微機
 工業控制計算機
 開關電源
 傳感器
 微機皮帶秤

新潮電腦

時代新潮



ISSN 1001-3695



05>



四川新潮計算機產業集團公司
 中國四川成都新鴻路四號 郵編: 610051
 電話: 4448100, 4444670 傳真: 4444115

“安”和“稳”的故事……



“凯联”追求稳定的电压和自身的稳定！

我厂专业生产稳压电源，SBW、DBW系列三相、单相全自动大功率补偿式稳压器，是专为电网电压波动而设计的，是国内目前最为先进的节能型稳压电源。

SBW、DBW系列具有：功率大、损耗低、稳压精度高、输入范围宽、能承受瞬间超负载……。它广泛应用于邮电、工业、医院、交通、科研、宾馆……等一切需要电压稳定的场所，也是所有进口设备的必配电源。

主要技术指标及规格：

指标：稳压精度：额定电压 $\pm 1 \sim 5\%$ (可调)
稳压范围：额定电压 $\pm 20\%$ 损耗：1.5% (平均)
波形畸变：0.1% 应变时间： $\leq 0.1s$ 有过压、故障自动保护
规格：三相 10, 20, 30, 50, 100, 180, 225, 300, 400, 500, 600, 800, 1000 (KVA)
单相 3.5, 10, 20, 30, 50, 100, 180, 225, 300 (KVA)

“凯联”服务宗旨：上门调试 代办托运 来询即答 终身服务

SBW、DBW——系列 全自动大功率补偿式稳压器



凯联

稳压



上海凯联稳压设备厂

地址：上海市常德路379号 电话：2537960 2553368 传真：021-2553368 电挂：1454 邮编：200040

经 销 商：
• 山东省机电设备总公司 地址：济南市经五路225号 电话：733723
• 陕西省机电设备总公司 地址：西安市西五路22号 电话：775249
• 湖南省机电设备总公司 地址：长沙市五一中路6号 电话：420645
• 南京市机电设备总公司 地址：南京市太平南路514号 电话：2215084
• 浙江省机电设备公司 地址：杭州市延安路217号 电话：555646
• 闽东电机(集团)公司 地址：福州市五一一路144号 电话：524452



MOTOROLA (半导体) 中国代理

香港庆成集团

成都庆成电子有限公司

成都庆成电子有限公司是香港庆成集团公司合资企业,是美国 MOTOROLA 公司(半导体)中国代理,提供 MOTOROLA 全系列半导体产品

RF 器件:全系列 MHW、CATV 放大模块。

功率器件:长期为中国市场配套使用的各个门类的功率器件

MCU 单片机:MC6805 系列、MC68HC05 系列、MC68HC11 系列

有线电视:专为中国有线电视用户开发的专用芯片,有线电视机上变换器最佳配置:MC3006、MC44802、MQS3001P。

电话机:全系列各种功能电话机 IC,首推中国优选之 MC34018、MC34114、MC3363、MC3361、MC2833

程控交换机:各类程控交换机芯片

通用集成电路:74LS 系列、74HC 系列、74HCT 系列、74F 系列、MC1400 系列、线性电路

另配套经营国外各名厂半导体器件

公司地址:成都市人民南路四段十一号附1号(省电子所内)

经营部地址:成都市一环路南二段九号

电话:(028)5582666-2018,2087

传真:(028)5542088

电话:(028)5541255

SBW、DBW 系列 三相、单相大功率电力稳压器

SBW·DBW系列三相、单相大功率自动补偿式电力稳压器是我公司引以为自豪的荣誉产品。多年来在引进欧洲技术的基础上,结合我国实际情况,使该产品性能不断完善、质量不断提高,销量不断上升,在强手如林,竞争激烈的市场中立于不败之地,深受广大用户信赖。该系列产品设有报警装置、自复装置、避雷保护、缺相保护、相序指示等多种功能,还可根据用户的特殊要求,三相分调、双速调节、配备隔离变压器,以适应任何国家引进的设备所需电压。欢迎联系,欢迎咨询。



指标: 稳压精度: 额定电压 $\pm 1\sim 5\%$ (可调)
稳压范围: 额定电压 II 型 $\pm 20\%$ (304~456V)
III 型 $\pm 25\%$ (285~475V)
波型畸变: 0.1%
损耗: 1.5%(平均值)
应变时间: $\leq 0.1s$ 有过压、故障自动保护
规格: 三相: 10~1200kVA
单相: 3~300kVA



来函必复·备有现货·上门调试·服务完善

上海仪表(集团)公司系统工程部制造

地址: 上海市大名路65号 电话: (021)3562757 3062998 邮编: 200080

传真: (021)3230561 电挂: 2599

开户银行: 工行虹办提分处 帐号: 212-04082840

Voltage Stabilizer

深圳金华轮电子有限公司

四川海韵电子公司

经销下列电子元件

intel	8031	80C35	8032	8272	8751	4164
	80C31	8039	8098	8748	8752	4416
	8035	80C39	8257	8749	D2914	2732

BOURNS	3296W/X	3386P/W	3862	3400	3520
	3006P	3323P/W	6674	3437	3362

MAXIM	A/D 转换器	模拟数字转换器	UP 监控器/电源检修器
	D/A 转换器	数字模拟转换器	显示驱动器
	放大/缓冲/比较器	计数器/定时器	
	模拟开关/模拟多路调节器	有源滤波器	
	电源直流—直流转换器	设计软件等	

National	MN3102	MN3101	MN3101	AN5435
	MN3207	MN3008	MN3007	AN5123

MOTOROLA	MC1330	MC1596	MC3242	MC3446
	MC1468	MC1741	MC3302	MC3460

SONY	CXA1238S. M	CXA1116	CX20172
	CXA1634P	CXA1109S. M. P	CX20023
	CXA1032	CXA1101S. M. P	CX20106

继电器/小型继电器牌子:三菱、松下、欧姆龙、富士通、日电

适用于:通信设备、程控交换机,自动系统,防盗器等。

固体继电器更适用于自动控制系统,如:影印机、传真机、包装机温度控制,电动玩具等。

世界名牌集成电路、电阻、电容、二、三极管、整流器、电位器、TDK 片状器件等,以及民品、工业品及军品。

联系地址:

香港金宝轮(国际)发展有限公司

香港海韵国际企业公司

Tel:0852-6885133 Fax:0852-6885069

深圳市爱华路沙埔头西 76 号 401 室

深圳金华轮电子有限公司

Tel:0755-3350962

四川成都市一环路南二段二十四号

四川海韵电子公司

Tel:028-5558770

计算机应用研究 第11卷 第3期(总第59期)

目次

综述评论

一种新的计算机应用体系结构——客户机/服务器	李文华 徐雅君 杜秀春 赵燕清(1)
微机数据通信	方友良(4)
未来CAD展望	胡德森(58)

研究探讨

可手写计算机识别码的研究	吴庆祥(6)
手写体数字识别的一种模糊联想记忆神经网络方法	何奉道(8)
集成电路最小全测试集矩阵算法的特点及程序实现	郑晓薇(11)
图像旋转的快速显示技术	李伟青(13)
DOS设备驱动程序原理浅析	周永利 席春红(17)
微机市场预测联合利润最大模型	张鸿鸣 刘铁军 王秀兰(21)

应用实践

哥德巴赫猜想等式的个数——软件水平考试程序设计练习	罗海鹏(27)
如何从二维图像中得到主动脉中胶原纤维的空间位置	高云钦(19)
步进电机智能控制卡	孙传铮(24)
一种双终端不同画面的显示控制器	周立 王晓红(14)
GAL器件在智能仪表设计中的应用	周鸣争(28)
电子广告显示屏	王水清(30)
VGA256种颜色的显示	李龙澍(31)
防止MIS意外退出系统的方法及实现	蔡青华 胡毅(32)
利用EPROM编程卡固化51单片机程序	王晋远(34)
激光印字机的文本打印程序	陈德明(35)
如何用多功能超卡代替串并卡	叶富乐(36)
EGA/VGA/TVGA屏幕特技技术	迟鲲鹏(37)
游戏机程序设计要点	王让定 王小牛 王联国 索国瑞 张召琪(39)
Foxpro下拉菜单系统的设计	侍永新(42)
实现矢量汉字的一种方法	单洪 李长喜(44)
主从分布式监控系统的设计	余立建 王长林(46)
只用二张360K的低密盘也可以使用WPS	朱广忠(48)
双积分型A/D片与单片机接口大全	孙庆梁 孙建国(50)
如何在应用程序中使用鼠标器	黄建新 于长海(52)
新颖的动画技术——颜色流动法	符云清(7)
舍位平衡算法	魏鹰(55)
从Pascal到C的自动转换	周韬 魏志毅(59)

维护维修

CTX-3型单色显示器异常故障维修一例	王建中 刘刚(10)
打印机两个典型问题的分析排除	饶增仁(38)
用DDD盘检测和调整软盘驱动器	梁学禹 孙吉权(41)
给硬盘“贴上”写保护	戴水贵(49)
提高静态存储器掉电保护可靠性的方法	宋颖慧 马平 谢生(54)
微机硬磁盘自举功能恢复一例	周宏(57)

动态简讯

简讯广告	(封二、封三、57)
本刊公告	(26、36)
新书简讯	(10、59)
新技术、新产品荟萃	

一种新的计算机应用体系结构——客户机/服务器

山东省计算中心 (济南 250014) 李文华 徐雅君 杜秀春

山东省科学院情报研究所 赵燕清

摘 要 本文介绍了客户机/服务器的产生背景、工作原理及其开发工具。

关键词 客户机 服务器 开发工具 数据库

1 客户机/服务器出现的背景

进入九十年代以来,客户机/服务器(Client/server)在信息管理领域内成了一个热门话题。在全世界范围内越来越多的用户单位开始逐步接受这一全新的结构体系,据统计,在北美已采用 Client/server 的应用系统已由 1990 年的 14% 上升到 1992 年的 34%,估计 1993 年会有更大幅度的提高,目前尚未见到正式报告,但 1993 年曾对 100 位信息管理系统的业务主管做过调查,80% 的信息部门主管表示已经开发了或正准备开发这种体系结构的信息管理系统。无怪乎目前世界上最有影响力的《计算机世界》、《软件》、《PC》等杂志几乎众口一辞,称之为在计算机领域中如果说七十年代是数据库的年代,八十年代是微机/网络的年代,而九十年代将是 Client/server 的年代。

那么,产生这样一种体系结构的背景是什么呢?

一般说来,我们把计算机按照其大小,功能,价格粗略地分为微型机,小型机和大中型机。而按用户和中央处理器(CPU)的关系来划分可分为单一式和宿主式,小型机,大/中型机,带若干个终端,我们称之为宿主式。而一般说来,微型机,它的 CPU 为一个用户所专用,完成用户所指定的单一任务,我们将其称之为单一式。微型机的优势毋庸多说,但它最大的弱点是用户之间不能共享数据资源,这就严重地影响了用户群体的工作效率。对于宿主式计算机来说,用户的应用程序及全部数据都存放在主机上,由主机 CPU 进行调度和管理用户的数据,这样在保证用户数据的完整性,安全性方面都显示出优越性。但是由于用户界面是建立在非智能化的终端上,一切的处理全部压在了主机的 CPU 上,尽管许多大/中型机或小型机的说明书上都宣称自己可以接多少个终端,实际上,系统效率的下降与用户的增加绝非呈线性关系。因此,为了追求高效率,只能最大限度地提高主机 CPU 的处理速度,这无疑增加了宿主式计算机的造价,造价并非与处理速度呈线性关系递增,而几乎是按令人可怕的指数关系递增。毫无疑问,这将给一个大型的信息管理系统的建设带来极大的风险。一次投资量大,开发也并非渐进的,而计算机技术发展方向的调整和应用界面的活跃变化,使得这种系统的生命周期比人们所预期的短得多,所谓“船大

难调头”,目前相当一部分这样的系统处于欲罢不能的尴尬局面。早在八十年代初期,人们基于微型机技术的弱点而开发的本地局域网(LAN)以及后来的广域网(WAN)技术,在一定程度上解决了数据共享问题,使得微型机的应用进入了一个新的时代,已从原来意义上的“Personal computer”进入大厅广众面前,承担起了较为复杂的事务处理任务。这期间涌现出了多种多样的网络管理软件,其中最成功的莫过于 Novell 公司的 Netware 以及 Microsoft 的 LAN Manager。但是,这类系统与我们上述的单一式处理似无本质上的区别,服务器只是被动地为来自工作站的数据存取要求提供服务,而对数据的处理如检索,更新,计算等仍然由工作站自己来完成,多个用户对相同数据不能做同步更新,文件的共享只能以轮流的方式来实现,即某一时间内被锁定于某一台计算机上,同时由于服务器只能简单地管理文件,所以当用户提出对文件的处理要求时,它只能将整个文件在网络中传输给用户,即所传来的文件中可能很大成分对于用户来说是不需要的,这无疑增加了网络的传输负担。这些问题严重地困扰着计算机用户和系统的开发者,尤其是对实时处理要求比较严格的应用场合,如证券交易,期货市场,多个用户在很短的时间里访问同一个文件,而且交易系统必须在极短的时间对用户的要求给出响应。宿主式系统或网络系统的负担过重,对数以百计的用户要求做到这样一点力不胜任。

2 客户机/服务器的工作原理

面对上述问题,人们便设想,存在一种合理的体系结构,既保留了上述系统的优点,又避免了其不足,那就是客户机/服务器(Client/Server),它将 PC 工作站的处理能力和大/中或小型计算机上的数据库管理系统(DBMS)二者之长处有机地结合在了一起,从逻辑上讲,它是由文件服务器进一步发展而成,即文件服务器不再仅仅是被动地管理文件,而成为一个数据库服务器。这种思想的核心是一分为二,无论是宿主式还是单一式,过去都是集数据存取与应用程序于一身,而现在则是分别由 Client(客户机)和 Server(服务器)分别来执行。数据库管理系统是安装在一台超级微机或小型机上,它接受到工作站上应用程序传来的数据要求,由 Server 的 DBMS 执行诸如选择,检索及索引排序等数据处理任务,然后仅仅把经过处理后的工作站上用

户需要的那一部分数据,而不是整个文件,传输到工作站上,由工作站完成对其所需数据的加工,如用户接口菜单处理,数据的一致性检查等。因此,这样一个体系结构减轻了中央处理机的负担,而充分利用了这种机器对大型数据库的管理优势,保证了数据的完整性,同时,又减轻了网络的负担,兼之充分利用了 PC 机的处理能力。这是一种思想的革新,设备依旧,配构方法不同,效果就大不一样。

归纳起来看,我们可以得出 Client/server 具有如下几点优越性:

1) 显而易见,系统的运行性能得到了极大的改善,不仅仅是 LAN 上的流通负担大大地减轻,而且服务器(一般以小型机来充当)上的 CPU 全部用于管理数据库,直接管理到数据库文件的记录级,因此,它提供了更好的并发控制以确保多个用户在同一时间内存取相同的数据,这就显著地改善了运行性能,特别是那类以高频率更新数据的运行环境。上海证券交易所用一台 HP-9000 系列的小型机,采用 Client/server 体系结构,连接上百个工作站,达到每秒处理 1800—2000 笔交易速度就是一个例子。

2) 所有数据都由中央 server 进行管理,这就为数据的完整性,可靠性,安全性以及易维护性提供了保障。

3) 由于系统功能明显地分为 Client 和 Server 两部分,这样从原则上讲,只要 Server 端数据库系统保持不变的前提下,对其基础硬件,软件加以更新换代,将不会影响前端的应用,保证了用户在应用开发及培训方面的投资,使设备的更新,升级,扩充变得极为容易。同样,由于任何传统的数据库都提供有一定形式的用户接口(API),用于开发针对它的应用,在 Client 端用户完全可以应用自己熟悉的开发方式,自如地开发针对它的应用,这就在极大程度上方便了用户需求的改善和提高,极利于系统的剪裁和分步实施,极利于系统的集成,如果选择方案合理,还可以在在一定程度上保护了前端以往的开发投资。

3 开发工具简介

Client/server 体系一经问世,如上所述,得到了全世界范围的广泛注意,许多著名的数据库开发商都声称自己的产品已经适应了这种体系的需求,像著名的 Oracle, Informix, Ingres 以及 SYBASE 等。而在前端 Client, 原来致力于 PC 工作站上工具软件开发的厂商,也都纷纷做出保证,其产品可以适用或将会适用于数据库 Server, 这类软件通常称为群件(Groupware), 即那种供 LAN 环境中一群 Client 端用户使用的软件。为了开发一个基于 Client/server 体系的应用软件,其工作平台软件应由以下三部分构成:

1) 一个运行在多任务操作系统上的后援数据库系统。

2) 一种功能强大的客户工具软件,如目前计算机专家所倡导的具备面向对象技术的第四代语言。

3) 一种开放性接口软件,即为了使后援数据库系统和前端应用之间传输命令和信息的命名管道接口,它独立于网络软件,但网络与 DBMS 软件的所有通讯都经由这种命名管道接口。

我们在评价这些软件系统的优劣的时候,要求后备数据库系统针对 Client/server 的特殊要求具有稳定性高,联机存取事件控制能力强,即面对相当数量的并发用户,大型数据库存取要有很高的事件流通率,快速响应,以及随着系统的增长在不明显衰变的情况下继续维持高流通能力。

我们对接口软件的要求是,对于一个 Client,后备数据库的数据资源在它看来都是透明的,而对于 server 而言,各种 Client 对于它的数据需求都是可以接受的。归根结底一句话,不同类的 Client 可以与各类后端环境的 Server 相连接,达到真正的异型机与异种数据库群体的透明联接。

对于前端开发工具,可以照顾到两个方面,一是专业开发人员开发技术水平增长的连续性,即在其原有开发手段的基础上稍加提高,即可开发新的软件;二是对于非专业技术人员对这种开发技术很快适应性。

从这个意义上讲,真正具备上述要求的数据库 Server 当首推 SYBASE 公司推出的 SYBASE SQL Server,而前端 Client 的开发工具则首推 Uniface 公司推出的 Uniface,而 Power Soft 公司推出的 Power Builder 也占有一席之地。而无论是 Uniface, 还是 SYBASE 都带有自己的 Open Server 和 Open Client, 作为其接口软件,完成前后端的信息转换。

SYBASE SQL Server 之所以很快在市场上走俏,其年销售额近年来都在成倍地增长,即使在中国,1993 年进入中国市场也销出了近一百套,原因在于该系统从它开始设计就是面向 Client/Server 这种体系结构的。我们一般是在普通局域网的基础上将系统体系结构改造为 Client/Server 结构,而 SYBASE SQL

Server 为了满足现在广泛应用的 Netware 网络专门开发了 SYBASE For Netware 版本,这就为用户的体系改造提供了极大的方便。除此之外,这种数据库系统还采用了以下一系列的先进技术,使其更具特色:

① 多线索单服务器进程体系结构。它将所有访问数据库的用户进程都连接在单一服务器进程上,采用类似于并行处理操作系统中所使用的先进技术,将服务器进程又细分为多个可并行的线索,对其做并行处理的调度,加上它独到的优化策略考虑,使它能保障在几百个或更多的并发用户的联机事务处理(OLTP)系统中既能保持事务处理的大吞吐量,又能使系统响应时间随着用户数的不断投入呈线性,而不是指数方式的增长。

②支持存贮过程。存贮过程是一组预编译过的 T—SQL 命令,经定义命名后,作为一个数据库客体登记在动态数据字典表中。T—SQL 是 SYBASE 对标准的 SQL 的扩展,能支持流程控制逻辑,允许申请局部变量等,用户的每个存贮调用都可以发送多种 SQL 请求,其中的 SQL 语句都是预编译过的,因而省去了解释及验证合法性的时间,成组传递数据访问的请求和处理结果则有利于最有效地进行网络传输打包。存贮过程的使用能有效地加快数据库访问速度和明显地减少网络传输量。

③支持 VSA,即虚拟服务器体系结构,能有机地与对称多处理技术结合,以充分发挥多 CPU 硬件平台的高性能。同时支持 Log 成组提交以降低每个事务的 I/O 次数。

④值得一提的是 SYBASE SQL Server 率先把数据完整性控制等事务逻辑由传统的放在客户一方转为由服务器一方完成,它能明显地改善控制质量并减少应用开发和维护的开销。

⑤由于 RPC(远程处理调用)技术和两段式提交技术的应用,它允许将数据处理放在网络的多个节点上,在应用程序的同一事务中允许访问多个服务器节点的数据,任何客户可以直接或经过本地服务器间接访问远程服务器,它仅需要一个本地服务器接口,而无需了解各部分数据物理上是散布在哪个节点上。

另外,SYBASE SQL Server 在容错能力和数据安全方面都有新的举措和提高。

值得一提的是,当前软件世界的泰斗 MicroSoft 公司购买了 SYBASE 的技术,推出了 Microsoft SQL Server for windows NT,这本身就是一个 SYBASE 关系数据库的 PC 版本,而 Client 端目前盛行的开发工具,大多是基于 MicroSoft 的 Windows 系统的,就为用户开发提供了更大的便利,由于 Windows 系统的基于图形的多任务多窗口,以及支持应用程序间的动态数据交换等特性,使开发的软件质量也大有提高。

最后,谈一下 Client 端的开发工具。在去年的评比中 Uniface 独占榜首,得了 76 分,位居第二的即 Power

Builder,其实这两个软件各有其长。Uniface 除了一般的开发工具所具备的功能外,它最大的特点是对于多种数据源体的兼容性,它将执行系统和数据管理系统的驱动程序常驻在 Client 端,并提供一个名叫 Poly Server 的通用 DBMSI/O 接口,它可通过数据库管理系统驱动程序把 I/O 需求转换为目标数据源体所能理解的数据管理语言,最终将所需信息传送回 Client 端,从应用的角度看,所有的数据存取都是透明的。而 Power Builder 则充分发挥了面向对象的开发技术的优势,极大地方便了管理信息系统的开发,它方便易学,尤其适用于非计算机专业技术人员为适应不

断变化的用户需求更新软件,它被 MicroSoft 公司推荐为标准的 Client/server 应用开发工具。这种开发工具的不足之处是它只能与关系型数据库相连接,而且也只能运行在 Windows 操作系统上,而不能运行在为中国广大用户所熟悉的 DOS 系统上。

对于熟悉 DOS 系统,尤其是在微型机上是沿着 DBASE,FOXBASE,FOXPRO 这样一个开发工具发展过程走过来的用户来说,Microsoft 不失时机地推出了适应这类用户的 Client/server 体系结构,前端的 Client 用户仍然用自己所熟悉的 FOXPRO for DOS 进行开发,只是将访问后援数据库的 SQL 语句嵌套在 FOXPRO 程序中间,在对 FOXPRO 进行预编译时,将 SQL 语句识别出来,通过一个 FOXPRO 连接软件 FOXPRO CK 将其传递给服务器上的 SQL Server,服务器上的 SQL Server 即我们上边提到过的 SQL Server for windows NT,它识别并接受 SQL 语句,经过处理,将用户需求的数据再通过它自身的一个连接软件传输给前端的 Client。实际上,SQL Server for windows NT,遵从开放的数据库连接规范(ODBC),它可以将嵌套在 FOXPRO 中的对于任意数据源体的需求进行识别,如对 Oracle,Ingres,Informix 的需求,传递给各自的数据库系统,从而使得这些数据源体对于 Client 端都是透明的。

4 结束语

本文从一般性原理上介绍了 Client/server 技术,具体实施中,还会遇到许多问题,在选择各自实现的方案时,存在着多种多样的考虑,如原来已开发的软件资源的保护,已有的硬件投资的保护,原来所采用的技术及后续技术的一致性,更重要的要考虑到系统组合的先进性和有效性,都需要设计人员保持清醒的头脑,而不是生搬硬套,或推倒重来。只有这样,用户才能从这种堪称当今世界上崭新的技术体系中,以最省的投资,获得预期的效益。

我国计算机技术发展到了今天,已经在很大程度上具备了接受这项新技术的物质基础和技术上的潜在的能力,SYBASE 公司去年一年在中国宣称卖出了成百套系统就是一个例证。本文的目的就是希望引起我国计算机界,尤其是已经为用户开发了不少管理信息系统的同仁们的注意,将该项已在国外取得极大成功的技术引用到我们正在进行或准备进行的工作中来。除了上述交易系统应用外,像厂矿企业的管理信息系统,机关的办公自动化系统,高档次的宾馆管理系统,都可以采取这种体系结构。在进行这种系统的设计时,不要继续沿着八十年代国外走过的老路,而是接受最新的结构设计思想,只有这样,才能使我们的计算机应用水平缩短和世界先进水平的差距,可望在不久的将来,能与世界先进水平并驾齐驱。

微机数据通信

甘肃省计算中心 (兰州 730030) 方友良

摘 要 本文介绍了微机远程数据通信的外部设备调制解调器(MODEM)的原理和操作,以及目前常用的汉字微机通信软件包的使用,说明了计算机数据通信的各要素及全过程。

关键词 计算机数据通信 调制解调器 RS-232 标准

1 概述

随着世界信息业的迅猛发展,人们对信息的共享要求越来越强烈。目前,在我国进行计算机数据远程通信,共享信息的用户越来越多,计算机数据通信在我国已逐步达到了成熟发展阶段。下面,综合一些国内外的有关资料及本人在计算机远程数据通信建设中的经验,对此作一剖析介绍。

2 计算机数据通信

计算机通信(也称数据通信或微机数据通信或点对点通信)就是在两台或多台计算机之间,经由传输链路传输各种类型的数据。一般过程如下:

A 计算机以位序列方式将二进制数据(数字数据)传送到调制解调器中,由调制解调器将这些位转换成能模拟出 1 和 0 的模拟信号,然后,通过电话线传输这些模拟信号。

在接收端,连接在 B 计算机上的调制解调器再将模拟信号转换回二进制信号,并且这些信号同 A 计算机向调制解调器发送的那些信号基本一致,然后,调制解调器将二进制信号发送到 B 计算机。

此时,B 计算机的内存贮器中就有一份 A 计算机原先发送给调制解调器的数据拷贝。当然 B 计算机也可以向 A 计算机发送数据。

由此可以看出,计算机通信的主要成份有:

- 数据(用户所要共享的信息)
- 数据终端设备(计算机)
- 数据通信设备(调制解调器等)
- 数据通信链路(公用或专用电话线等)

如图 1 所示。

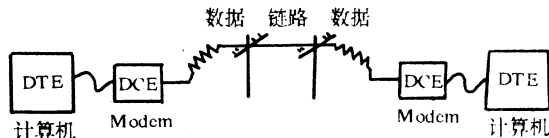


图 1 数据通信结构示意图

3 调制解调器

调制解调器实际上是调制器和解调器的简称。它的组成如图 2。

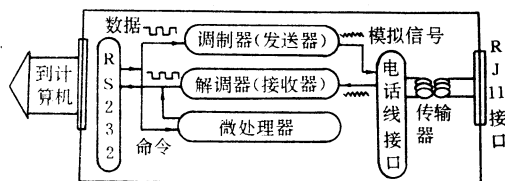


图 2 调制解调器结构图

调制器(发送器)把数字信号转换为模拟信号,通过电话线发送出去;

解调器(接收器)把模拟信号转换为数字信号,通过 RS-232 接口送到计算机里。

调制解调器选用的主要参考因素有以下五种:

- ①质量;②速率;③标准;④兼容性⑤其它。

3.1 工艺质量

同样技术指标的调制解调器,不同厂家其质量有很大差别。在噪音电话线或远距离电话线上,特别是在交替载波的线路上,不同厂家的产品质量存在着明显的差别,所以,调制解调器的选用在经济许可的情况下要特别注重选购名牌产品。

3.2 调制解调器的速度

调制解调器的速度是一项重要的技术指标,在长途电话线上,传送大量数据所占用的通信时间是不可轻视的。如果经常性在长途线上传送大量数据,购置 9600 波特率的调制解调器比购置 2400 波特率的调制解调器要合算得多。

3.3 调制解调器的标准

调制解调器的标准越高,功能就越强。对于同一速度的调制解调器,标准是可以不同的,由此导致它的价格也相差很大。例如,9600 波特率的调制解调器,有些只有 V. 32/V. 32bis, V. 22/V. 22bis 标准,有些还有 V. 42/V. 42bis。具有 V. 42/V. 42bis 的调制解调器有很强的纠错能力和 4:1 的数据压缩力。但这也导致了价格上的差异,用户可根据情况选用。

3.4 调制解调器的兼容性

AT 指令集是调制解调器兼容的手段,在选用时这些务必必要考虑到,因为不兼容的调制解调器通讯是不可能的。有些高标准的调制解调器还具有扩展 AT

指令集。一般随机手册中都列出了 AT 指令集,用时可查阅。

3.5 其它

由于调制解调器技术发展很快,有些产品具有很多新的功能,如:智能适应(自动测试和适应线路品质,调整效率等),自动拨号,自动应答,存储号码和自动重拨等,用户可参考选用。

目前生产的调制解调器一般都有前面板指示灯,以便用户观察调制解调器的工作状态:

SD:发送数据; OH:摘机;
TD:接收数据; TR:终端(计算机)
CD:建立载波(即接通); 准备就绪;
HS:高速率; ANS:自动应答;
LS:低速率; ORG:手动应答。

4 通信软件包的使用

Xcom 是目前经常用到的汉字菜单驱动式通信软件,适用于各种 IBM 及其兼容机,各种不同型号、不同速率、不同标准的调制解调器,适用于公共电话交换网及专线网。可在任何汉字 DOS 下运行。驻留内存只占极小的空间。支持两个串行口及 AT 指令集。

该软件可在任何驱动器上运行,启动命令格式为:

路径,Xcom[/Sn][Pn]

其中:Sn—通信速率

• n=0 19.2K bps
• n=1 1200 bps
• n=2 2400 bps
• n=3 300 bps
• n=4 4800 bps
• n=7 7200 bps
• n=9 9600 bps

Pn—串行口端口号

• n=0 串行口 1(COM1)
• n=1 串行口 2(COM2)

当省略 Sn、Pn 参数时,系统默认为 2400 波特率,串行口 1(COM1)。

系统的主菜单为:

现在是等待状态

- ①发送文件 ②获取文件
- ③列本地目录 ④列对方目录
- ⑤看本地文件内容 ⑥看对方文件内容
- ⑦专线呼叫 ⑧专线应答
- ⑨自动拨号 ⑩数据/语音转换
- ⑪点对多点自动通信 ⑫断点续发
- ⑬断点续取 ⑭实时屏幕对话
- ⑮后台呼出

按<ESC>返回 DOS 0 禁止(允许)登录

Xcom>>(通信提示符)

• 呼入方式

在远端计算机采用自动拨号和后台呼出功能呼入本地时,系统将在允许情况下自动进入通信屏。

• 呼出(建立通信链路)

若用户设置调制解调器为手动应答,则须先手动拨通对方的电话,然后双方同时一个选 7 回车,一个选 8 回车,建立载波接通即可通信。

若对方调制解调器设为自动应答,则选 9 回车,然后键入对方的电话号码并回车,自动建立载波,或进入功能 14,先打 AT 回车,显示 OK,进入 AT 命令方式,然后根据自己的电话是双音频还是脉冲式,用 AT 命令集的 ATDT 电话号码回车或 ATPT 电话号码回车,建立载波,即可通信。

若连接成功,调制解调器的 CD 指示灯亮。

• 数据获取及发送

在主菜单下,选择 1 至 6 号功能,即可根据菜单提示,完成你欲实现的通信操作。

• 实时屏幕对话

选择实时屏幕对话功能即进入对话屏,左半屏为你的发话区,右半屏为你的收话区。

• 数据语音转换

在数据通信时,选择此功能,可拿起电话与对方通话,如:比较传输的文件长度是否正确。

5 错误诊断及处理

联机问题的最佳办法就是在申请注册前就解决。

1). 发出了拨号命令,但什么也没有发生

• 检查计算机和调制解调器的所有接插件及电源是否接好;

• 试一次手工拨号,以保证电话线路正常;

• 检查通信软件,要确保根据需要发出了脉冲或音频拨号命令;

• 是否内线电话没有拨外线;

• 要保证调制解调器的所有开关都已按拨号需要正确设置(可参阅调制解调器手册);

• 计算机上串行端口是否不止一个,串行口是否选择正确。

2). 电话铃响了,但没有任何应答

• 检查所拨的电话号码是否正确。

3). 听到了对方系统发来的信号,但计算机不作任何响应

• 检查调制解调器的连接情况,如此还不能解决问题,就关掉调制解调器和计算机的电源,然后重新启动计算机,如仍不能解决问题,则要请教专业技术人员。

可手写计算机识别码的研究

福建师大实验中心 (福州 350007) 吴庆祥

摘要 为了使得用手填写的数码能让计算机百分之百地自动识别,本文研究了一种新的可手写计算机识别码,该码是由标有数字的小方格和识别标志组成的二维图形符号,只需简单地在相应的方格上涂黑,即可得到计算机可识别的数码。文中也阐述了以这种码为基础研制计算机数码自动识别系统的方法,及其在财会,商业票据和普查表格数据自动输入中的应用。

关键词 手写识别码 计算机自动识别系统

文字的计算机自动识别是当今计算机科学中的重要课题,特别是对数字的识别要求更高,例如在财会金融和商业等领域,对于数字的识别必须保证百分之百正确。因此,目前国内许多单位研制的手写字识别系统,虽然具有 95% 以上的准确率,但还不能投入实际应用。然而许多领域还是相继采用条形码,国际商品条码化就是一例。但是条形码不能用手写,在需要成千上万人填写数据的地方,如邮政编码,普查表格等,就无法使用条形码。因此设计一种新的可手写的计算机识别码具有广泛的应用领域。

1 可手写识别码的设计

新码要达到的两个重要指标的既能用手写又能让计算机百分之百识别。然而我们把它设计成什么形状才能满足要求呢?不妨让我们先来分析一下条形码和手写字,也许就会清楚了,条形码可百分之百识别,它的特征是一维图形并且形状大小有严格的规定;手写字能手写,它的特征是二维图形而大小形状随机性非常大,由此可看出形状大小的严格规定是百分之百识别的关键所在,码形维数的增加意味着表示容量的扩大,因此我们决定采用二维方格来表示新的码,这就规定了严格的形状,为了便于填写在方格上用很淡的颜色打上数字(或者用不同的颜色加以区别),如图 1 所示,用一行 0—9 的方格表示一个数字,若要表示某个数字只要在相应的数字上涂黑即可,图 1 表示的数为“3578.19”。

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13
R1													
R2													
R3			0	1	2		4	5	6	7	8	9	
R4			0	1	2	3	4		6	7	8	9	
R5			0	1	2	3	4	5	6		8	9	
R6			0	1	2	3	4	5	6	7		9	
R7			0				4		6	7			
R8			0		2	3	4	5	6	7	8	9	
R9			0	1	2	3	4	5	6	7	8		
R10													

图 1

此外,为了扩大新码的表示容量,不但要表示 0—9 的十进制数,还要表示 ASCII 标准字符集,我们对新

码还作如下规定:

1). 若每行 0—9 的方格只有一个方格被涂黑,则表示一位十进制数。

2). 若 8,9 两个方格都被涂黑,则 0—7 的八个方格按八位二进制的方法表示 ASCII 字符码。例如图 1 中的小数点就是以 ASCII 码“2E”即二进制码“00101110”,将与“1”相对应的位涂黑来表示。

2 识别技术及其实现方法

为了识别时便于定位与提高可靠性,在图 1 中上下与左右两边设计了一些条纹。利用这些条纹采用以下方法就可实现自动识别。

2.1 定位输入识别

每一列对应一个光电输入器,共十三个。当十三个输入器都处于“暗”信号时,说明起始位置已正确。接着向下扫描,并根据 C1 列光电信号进行输入数据,若 C1 和 C13 列的信号变为“亮”时,校正 C2, C4, C6, C8, C10, C12 的信号是否为“暗”, C3, C5, C7, C9, C11, C13 是否为“亮”,否则进一步调整直到正确为止。然后,继续向下扫描,根据 C1 和 C13 的亮暗交替变化,读入对应的数据,直到第 R10 行的 C2—C12 读入器全为“暗”时,表示读完一个数据。

在硬件设计上可分为两个部分,第一部分是数码图形的输入,主要通过光电读入器读取图形数码;第二部分是单板机或单片机为核心的数码转换器,将读入的图形码转换为计算机的键盘码通过计算机的键盘接口[1]送入计算机,并且在不用识别器时,又能让计算机的原有键盘自动接通,用原键盘输入。

2.2 图形扫描输入的识别

首先用扫描器将图形码输入计算机,然后在图形上找出起始行,根据 R1 黑杠的长度计算出列的宽度,并设立 13 组检验点作为上述的光电输入器。最后以上述的方法沿着垂直于黑杠 R1 的方向扫描,将数据识别出来。使用这种方法可以识别不同大小的图形数码。

在硬件方面,可在计算机中装上 GS—4500SCANNER^[2],利用它的图形扫描功能将图形码输入计算机,再设计一套转换软件,将这图形码转换成数据存到计算机的数据库中。

新颖的动画技术——颜色流动法

重庆大学 12-27 信箱(630044) 符云清

目前,计算机动画技术有很多种,常见的有:
一、擦写法,即在某个位置显示画面后,然后将它擦除,再迅速移到新的位置,如此反复。由于程序的快速执行,给人的感官印象便是画片在连续移动;
二、页面技术。将一系列要连续显示的画片存贮在显示 RAM 的不同页面中,然后,通过改变活动页面达到动画效果。这些方法都有一定的局限性,如擦写法可能引起速度慢,连续感不好等,而页面法则因显示 RAM 有限,不能存放较多的画片等。

笔者在此介绍另一种动画显示技术——颜色

流动法。现以模拟回路流水彩灯显示为例,在流水彩灯显示中所有彩灯仅颜色不同而已,其它特性均一致。所以动画设计中,保持灯位不变,而灯的颜色去流动,即每流动一次,相当于颜色阵列循环移动一次,其效果等于整个灯序列循环移动一次,从而达到动画效果。使用该法设计简单,显示速度快。笔者用此法编写了回路彩灯模拟程序,语言为 turbo c2.0。经编译,链接便可在带彩显的机器上实现其效果不错,而且流动速度可以控制。当然,读者不难借此发挥,设计其它动画技术,如属性流动法等。同时,在你的软件界面中加上这些技术,则将使软件添色不少。需实例程序清单(流水彩灯显示)者可同作者联系。

收稿日期:1993年3月11日

3 新码设计的几点考虑

3.1 可靠性措施

因为本码采用二维图形表示,在扫描识别时,扫描线应始终保持与起始黑条保持平行,否则会导致误码,因此,在 C1 和 C13 列设置了黑白相间的行变化标志,在扫描过程中,使 C1 和 C13 列黑白变化同步,则可使扫描线与起始黑条平行,而不误码。

C2 列黑条的设计可在扫描识别过程中随时检测扫描运动方向是否与起始黑条垂直,从而保证扫描识别的正确性。

3.2 方便性的考虑

码的线度允许较大时,可在方格上打上数字,因此填写非常方便。若码的线度较小,则用不同的颜色来区分方格,同样也很方便。

另一方面,有了 C1 和 C13 的行变化标志后,也使得本码的纵向宽度(行的宽度)可以不等,只需保持 C1, C13 变化同步就可正确读取数据。因此可采用纵向不等距的图码,在表格的设计中带来许多方便,如表 1 就是一例。

3.3 容错方面的考虑

因为码的填写严格遵循上述的两条规定,所以在码受污损时,很快即可被发现,如在码中发现 8,9 方格未涂黑而 0—7 方格有两个被涂黑,则可肯定码有受污或受损。对于要求更高的地方,可在每个数码的最后一位设置奇偶检验位,来保证读入正确。

4 应用意义

本码可以应用于所有的信息处理部门,尤其是需要成千上万人填写数据的领域,如信件的邮政编码识别,财会商业的票据,商品的编码,普查表格等等。若使

用这种码,并由计算机自动识别,将会节约大量的录入投资和人力。因此有很高的应用价值。

使用本码后制作的表格形式如表 1 和表 2 所示。

表1 体质调查表

姓名	编号
性别(1为男2为女)	011287-6738
年龄	
工作单位	
职称编号	
身高	
体重	
血压	收缩压
	舒张压
跳高	

表2 收款收据

科目 19 年 月 日	
缴款单位 或缴款人	编号
缴款项目 摘要	金额
金额 大写 万 千 百 拾 元 角 分	
小写	
主管 会计 出纳 经收入	

参考文献

- 1 IBM PC 硬件技术手册,中国科学院希望电脑技术公司,1989
- 2 GS-4500 USER'S MANUAL, KUN YING ENTERPRISE CO. LTD, JAPAN, 1989

手写体数字识别的 一种模糊联想记忆神经网络方法*

西南交通大学神经网络与信息技术研究所 (成都 610031) 何奉道

摘 要 手写体数字识别是当前神经网络应用研究最为活跃的领域之一。本文提出一种基于模糊联想记忆神经网络模型的识别方法,并获得较为满意的计算机模拟结果。与常用的 BP 网络算法相比,具有学习速度快、算法简单、网络规模小、拓扑结构简单等优点。

关键词 神经网络 模糊联想记忆 手写体数字识别

1 引言

手写体数字识别是解决计算机数据自动输入的关键技术之一。对一般的实用系统,往往要求具有很高的识别率和识别速度。传统的识别方法多采用基于模式识别和人工智能的 OCR(光学字符识别)技术,目前已提出数十种方法。但不论是基于决策理论的统计模式识别还是基于形式语言的结构模式识别,都难以满足实用要求,被认为是传统人工智能发展面临危机的表现。近几年来,迅速发展的人工神经网络由于具有高度的并行运算能力和分布式存储方式,很强的容错性和自适应学习功能,被广泛地应用于信息处理诸领域,其中最主要的领域就是模式识别。人工神经网络研究的兴起为手写数字识别提供了一条新的,很有希望的途径。

目前,应用人工神经网络进行手写数字识别方法的研究主要是采用 BP 神经网络模型。由于 BP 模型和算法自身的特点,学习速度慢、易陷入局部最优、网络规模大和结构较复杂,特别是如何确定隐节点的个数目前尚缺乏理论上的指导,一般都是以输入层和输出层节点数或由经验来选择,这无疑也给网络输入层和输出层节点的设计施加了限制等,是该种网络难以克服的困难。各种改进算法也不能从根本上解决这些问题。

从模拟人的智能特征看,人工神经网络侧重于对大脑内部结构的模拟,而模糊集合理论则突出对人的外在表达方法的描述,二者之间必然存在着密切的联系。Bart Kosko 提出的模糊联想记忆神经网络是把神经网络理论和模糊集合理论相结合而形成的最具代表性的一种模糊神经网络。由于该网络采用模糊赫布型学习规则而不能有效地进行多个模糊值模式对的联想存储,从而极大地影响了它的应用。

本文针对模糊联想记忆神经网络学习算法的缺点,提出一种自适应优化学习算法,从而使网络能有效地联想记忆多个模糊值模式对,并将其应用于手写数

字的识别,获得了较为满意的计算机模拟结果。

2 模糊联想记忆神经网络和模糊赫布型学习规则

模糊联想记忆神经网络是一种能联想存储任意模糊值模式对 (A, B) , $A = (a_1, a_2, \dots, a_n)$, $B = (b_1, b_2, \dots, b_m)$ 的两层人工神经网络,输入层和输出层神经元广泛连接。输入层第 i 个单元对应于输入模式 A 的第 i 个分量 a_i , 输出层第 j 个单元对应于输出模式 B 的第 j 个分量 b_j 。 a_i 和 b_j 之间的连接权为 W_{ij} 。网络拓扑结构如图 1 所示。

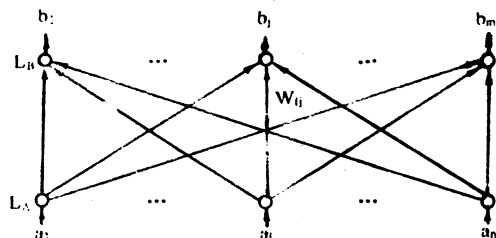


图 1 FAM 拓扑结构

Kosko 采用如下学习规则建立网络关于学习模式对 (A, B) 的连接权矩阵

$$W_{ij} = a_i \wedge b_j, i=1, 2, \dots, n, j=1, 2, \dots, m. \quad (1)$$

式中 \wedge 为模糊极小运算,它反映出神经元 a_i 和 b_j 之间的联系随它们的同步兴奋而增强,即是一种模糊赫布型学习规则。

$$\text{网络的回想和联想由方程 } B = A \circ W \quad (2)$$

完成。如定义 \circ 为模糊极小——极大合成运算,即有

$$b_j = \bigvee_{i=1}^n (a_i \wedge w_{ij}), j=1, 2, \dots, m. \quad (3)$$

上式一方面反映了输入刺激量 a_i 和连接权 w_{ij} 之间的限制性修正作用,同时也反映出神经元之间的主因素突出型竞争方式,因而可认为是一种主因素决定型合成关系。

为了在网络中联想存储多个学习模式对 (A^k, B^k) , $k=1, 2, \dots, p$, Kosko 提出一种学习规则

$$W_{ij} = \bigvee_{k=1}^p W_{ij}^k = \bigvee_{k=1}^p (a_i^k \wedge b_j^k), i=1, 2, \dots, n, j=1, 2, \dots, m. \quad (4)$$

网络的回想或联想方程为

$$B^k = A^{k0}W, k=1, 2, \dots, p. \quad (5)$$

$$\text{即 } b_j^k = \bigvee_{i=1}^n (a_i^k \wedge W_{i,j}), j=1, 2, \dots, m; k=1, 2, \dots, p.$$

不难看出,模糊联想记忆神经网络学习规则可归结为寻求使(5)式成立的模糊关系矩阵 W 。由模糊集理论可知,在大多数情况下,这样的矩阵是不存在的。即使存在,由(1)或(4)式获得的权矩阵也不一定是解。另一方面,由学习模式对集合通过一次学习便形成网络连接权矩阵,使得网络的回想和联想特性直接取决于学习模式对的性质,而要求给出理想的学习模式对往往是不现实的。为了使网络具有较好的回想和联想特性,一个有效的方法是对由(4)式形成的连接权矩阵进行自适应优化。

3 网络的优化学习算法

设 $(A^k, B^k), k=1, 2, \dots, p$ 为 p 个训练模式对。 B^k 的第 j 个分量为 b_j^k , 网络对应单元的实际输出为 \bar{b}_j^k 。第 k 个模式对的回想误差函数为

$$E_k = 1/2 \sum_{j=1}^m (\bar{b}_j^k - b_j^k)^2, k=1, 2, \dots, p. \quad (6)$$

全部训练模式对的全局误差为 $E = \sum_{k=1}^p E_k$

$$\text{即 } E = 1/2 \sum_{k=1}^p \sum_{j=1}^m [\bigvee_{i=1}^n (a_i^k \wedge W_{i,j}) - b_j^k]^2 \quad (7)$$

调整网络连接权使 E 最小,这可由梯度下降规则来实现 $\Delta W = -\alpha(t) [\frac{\partial E}{\partial W_{i,j}}]_{n \times m}$ (8)

其中, $\alpha(t)$ 为学习率,应设为迭代次数 t 的单调递减函数。

(8)式中梯度的计算含有模糊合成运算的微分,可采用如下规则运算:

(1) 设 a, b 为两模糊变量, \wedge, \vee 分别表示模糊极小、极大算子,定义

$$\frac{\partial}{\partial a} (a \wedge b) = \begin{cases} 1 & \text{若 } a \leq b, \\ 0 & \text{否则。} \end{cases}$$

$$\frac{\partial}{\partial a} (a \vee b) = \begin{cases} 1 & \text{若 } a \geq b, \\ 0 & \text{否则。} \end{cases}$$

(2) 设 $X = x_1, x_2, \dots, x_n, Y = (y_1, y_2, \dots, y_n)$ 为两 n 维模糊向量, 0 表示模糊极小——极大合成算子,由(1)我们有

$$\frac{\partial}{\partial x_i} (X \circ Y^T) = \frac{\partial}{\partial x_i} [\bigvee_{j=1}^n (x_j \wedge y_j)]$$

$$= \begin{cases} 1 & \text{若 } x_i = (x_i \wedge y_i) \geq \bigvee_{j=1}^n (x_j \wedge y_j), \\ 0 & \text{否则。} \end{cases}$$

(8)式表明,网络连接权矩阵的变化应正比于训练模式对集合各模式对对应的负梯度之和。在实际应用中,为简化计算,可每给网络提供一个模式对即计算误差函数 E_k 的梯度并调整连接权矩阵。这虽偏离了全局误差函数 E 上的梯度下降,但当 $\alpha(t)$ 较小时,这种偏离是可以忽略的,给予连接权矩阵以合适的初值,网络

将收敛于全局误差函数的最小值点。

综上,模糊联想记忆神经网络优化学习算法可描述如下:

(1) 初始化:由(4)式设置连接权矩阵初值,给定精度 ϵ 和迭代次数上限 u 。

(2) 对每一训练模式对 $(A^k, B^k), k=1, 2, \dots, p$ 调整连接权矩阵:

$$W(t+1) = W(t) - \alpha(t) [\frac{\partial E_k(t)}{\partial W_{i,j}}]_{n \times m}$$

$$\frac{\partial E_k(t)}{\partial W_{i,j}} = [\bigvee_{r=1}^n (a_r^k \wedge W_{r,j}(t)) - b_j^k] \cdot \delta_{i,j}(t)$$

$$\delta_{i,j}(t) = \frac{\partial}{\partial W_{i,j}} [\bigvee_{r=1}^n (a_r^k \wedge W_{r,j}(t)) - b_j^k]$$

$$= \begin{cases} 1 & \text{若 } W_{i,j}(t) = (a_i^k \wedge W_{i,j}(t)) \geq \bigvee_{r \neq i} (a_r^k \wedge W_{r,j}(t)), \\ 0 & \text{否则。} \end{cases}$$

(3) 计算 $E(t+1)$ 。

(4) 若 $E(t+1) < \epsilon$ 或 $t > u$ 则结束,否则 $t = t+1$, 返回(2)。

实验表明上述优化算法是非常有效的。当存在使(5)成立的模糊关系矩阵,经过数次甚至仅一次学习使得到解矩阵;若不存在使(5)成立的模糊关系矩阵,网络也能收敛到全局误差函数 E 的全局最优点,学习次数一般不超过 40 次,极大地提高了网络联想记忆多个训练模式对的性能。

4 手写数字的特征提取

应用人工神经网络识别手写数字,网络的输入模式一般有两种选择:特征输入和直接输入手写数字的图像的点阵。本文的输入模式由字符的交叉特征构成。所谓特征是计算若干条(10条以上)水平、垂直和斜向扫描线与字符的交点数并通过归一化处理而获得。

令 h_i', v_i', o_i' 分别表示水平扫描、垂直扫描和斜向扫描时 i 次相交的点数。归一化的各次交点数分别为 $h_i = h_i' / \sum_{j=1}^4 h_j', v_i = v_i' / \sum_{j=1}^4 v_j', o_i = o_i' / \sum_{j=1}^4 o_j', i=1, 2, 3, 4$ 。

由于 0—9 十个数字与水平、垂直、斜向扫描线的交点数最多为 4,因而当扫描时交点数大于 4 时,表明字符严重形变或有干扰,可按 $i=4$ 处理。

采用扫描线与字符的交叉点次数并经归一化处理作为字符的特征抽取,能较好地体现出字符的结构拓扑特征,有效地减小了网络规模,基本上消除了字符位置、大小、形状的影响。如图 2 所示的两个字符的位置、大小有明显的差异,但所抽取的特征向量

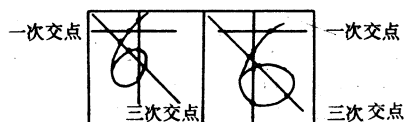


图 2 字符的特征抽取

CTX-3 型单色显示器异常故障维修一例

化工部沈阳化工研究院计算机室(110021)王建中 刘 刚

(CTX-3 型)单色显示器是几年前我室 S47-STD 工业控制机的外设,在使用中出现故障,经修复又恢复正常。

CTX-3 型单色显示器是深圳爱华电子公司的产品,虽然现在微机一般都配置彩显,但有些家用普及电脑还仍用单显,所以有必要总结。

故障现象:

开机后显示屏左边出现向右大幅卷边,调行幅旋钮作用甚微。关闭主机,故障依旧,说明原因在单显本身。重新开机并测电源电压,低于正常值($V_D=16\text{v}$)为 12v 左右,电流 $I_D=1.8^A\sim 2^A$ ($I_D\approx 1.2^A$) 去掉机内电源用直流稳压源代替,现象依旧,后又更换新行

输出管和新定做的高压包,现象依旧,又查过逆程电容和几乎所有的与行有关的电容,都没有发现故障点。这时行部分就剩偏转线圈没查了。

CTX-3 型单色显示器的显象管和偏转线圈均为前南朝鲜(信陵公司)产品,线圈外形美观,漆封严密,质量表面上看应无可怀疑。在无奈之际,用一国产 12 吋黑白电视机偏转线圈代替,除电流略微大些,线性度在四角略差些外,屏幕基本恢复正常。

结论:

此现象为行偏转线圈匝间短路所致,在确定故障点时,不能因有些部分故障率低或产品外观印象好而忽略它们。

收稿日期:1993 年 3 月 23 日

5 计算机模拟结果

网络输入层节点为 12 个,对应于字符特征向量的 12 个分量;输出层节点 10 个,分别对应于 10 个数字 0~9。

实验样本为来自 40 个人的手写体数字 40×10 个。随机选取 10×10 个数字以及一组标准体数字作为训练模式集,取 $\alpha(t)=t^{-0.5}$,经过 38 次学习网络收敛。识别方法为待识别字符的输出向量与标准模式的输出向量按模糊贴近度进行分类。训练样本的识别率为 95%,非训练样本的识别率约为 91%。

6 结语

本文提出的手写数字识别的神经网络方法,综合了模糊赫布型联想记忆神经网络和 BP 型神经网络的优点。利用误差逆传播梯度下降技术使用网络连接权矩阵收敛到误差曲面的最小值点,同时保持了模糊赫布型联想记忆神经网络的输入量与连接权之间的限制性修正作用和主因素决定型模糊合成关系的特点,使网络的回想和联想能力得到极大的提高。与目前广泛采用的 BP 神经网络方法或其改进算法相比有如下优

点:

- 1) 学习速度快,训练次数比 BP 算法降低 1~2 个数量级;
- 2) 网络规模小,拓扑结构简单;
- 3) 学习算法和回想算法简单;
- 4) 综上,该方法更便于软件或硬件实现,这对实用是有意义的。

参考文献

- 1 Kosko B. Fuzzy Associative Memory. In: Kande A ed. Fuzzy Expert Systems Reading, MA: Addison-Wesley. 1987
- 2 Kosko. B. Neural Networks and Fuzzy Systems: A Dynamical Systems Approach to Machine Intelligence. Volumes I-IV. Prentice-Hall. 1991
- 3 王晓蒲、末田正次郎。“文字と走直线、との交差情報を用いた手书文字认识”。日本电子情报通信学会 1992 年日本全国大会讲演论文集第七分册。
- 4 新蕃、范俊波、谭永东。神经网络与神经计算机。西南交通大学出版社。四川成都。1991.12

新书征订简讯

应计算机行业长期以来强烈渴求,四川辞书出版社特邀多年从事计算机维修业务方面的专家,新近编撰出《计算机故障维修 1000 例》,并于今年 8 月份出版发行,全方位介绍计算机各种故障的分析诊断及维修经验,内容翔实,举例众多,特别是新一代电脑产品 386、486 等维修实例,均有涉猎,新颖实用。每一实例均分故障现象、故障分析、故障排除三个部分,维修者可按实例对号入座。本书深入浅出,既适合专业人员使用,非专业人员亦可按实例进行检修。其读者对象为计算机专业人员、家庭电脑使用者;也可作为计算机维修技术培训班的教材。需此书者,请将书款(每册 13 元,批量预订折扣 25%)通过邮局汇至本刊编辑部哲明收,请勿在信中夹寄现金。

集成电路最小全测试集矩阵算法的特点及程序实现

辽宁师范大学计算机科学系 (大连 116022) 郑晓薇

摘 要 最小全测试集技术是对数字集成电路进行逻辑功能检测的方法之一。采用计算机编程实现的最小全测试集技术,涉及到特殊的矩阵算法。本文利用数据库语言的特殊构造形式以及强大的数据处理能力,较好地解决了矩阵变换、矩阵运算等程序算法问题;并用 FOXBASE 加以实现,使得最小全测试集可以由计算机自动建立,测试参数可以自动产生,达到了使用其它高级语言不易达到的效果。

关键词 故障表 等价 相容 测试子集

对数字集成电路进行逻辑功能测试的方法之一是**最小全测试集技术**。它是基于故障表法产生的。它可以用最少的测试次数检测出电路中所有的单故障。而故障表的形式与真值表相类似,可以看作二维表。对于组合逻辑电路来说,测试 T_k 就是加在电路上的输入向量序列。 $X_{k1}, X_{k2}, \dots, X_{kn}$ 和相应的该电路的输出向量序列。 S 为测试长度。这些测试的集合称为**全测试集**。我们用公式表示为

$$T_k = \{X_{k1}, X_{k2}, \dots, X_{kn}, Z_{k1}, Z_{k2}, \dots, Z_{kn}\}$$

当含有故障 f_i 时,产生的输出向量序列与 Z_{k1}, \dots, Z_{kn} 有不同,则 T_k 检测故障 f_i 。在全测试集的基础上产生最小全测试集其计算工作量非常大,因此采用计算机辅助计算是解决问题的手段之一。在计算机程序设计上可以有多种方法。由于可以把故障表看作二维表,那么在计算机语言的选定上选择关系型数据库语言就比较方便。因为关系数据库的数据结构是二维表的形式,并且它可以进行各种关系运算,对数据的处理能力极强。本文选用关系型数据库语言 FOXBASE 设计的最小全测试集数据库在软件实现上具有独到的优点。

以两输入或门电路的故障表为例,其故障表如下:

X_1	X_2	Z	f_{10}	f_{11}	f_{20}	f_{21}	f_{30}	f_{31}
0	0	0	0	1	0	1	0	1
0	1	1	1	1	0	1	0	1
1	0	1	0	1	1	1	0	1
1	1	1	1	1	1	1	0	1

它共有四个测试 $\{(00), (01), (10), (11)\}$ 。一个正常输出 Z , 六个故障输出。其中 f_{10} 为 X_1 固定为 0 的故障输出函数,其余类推。从上面故障表中获得最小全测试集要进行以下几个步骤。①去掉等价故障。②得到故障检测矩阵 G 。③去掉劣势行,优势列后获得最小全测试集矩阵 G^* 。

我们将故障表进行改变得到如下两个矩阵,一个

是输出向量矩阵 Z , 另一个是故障矩阵 F 。其中

$$Z = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \end{bmatrix}, \quad F = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 & 1 \end{bmatrix}$$

则上述工作变为对矩阵的计算。下面对矩阵算法的特点及程序实现分几部分加以描述。

1 合并 F 矩阵中等价故障的算法

对于一个 (m, n) 矩阵 F , 若 $F^{\alpha i} = F^{\beta j} (i=1, 2, \dots, n, j=1, 2, \dots, n, \alpha=1, 2, \dots, m)$, 则称为等价。将 $F^{\alpha i}$ 和 $F^{\beta j}$ 合并, 得到矩阵 E 。E 矩阵是去掉等价故障后的故障矩阵。

$$E = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \end{bmatrix}$$

设矩阵 F 已存入数据库 Zhen 1.DBF 中。用类 FOXBASE 语言描述等价算法如下:

```
P=1
do while p<=n
use zhen1
if f&a. &i=F&a. &j
do while not eof()
if F&a. &i=F&a. &j
bz=1, skip
else
bz=0
exit
endif
enddo
if bz=1, use jian1
dele all for field _name="F&a. &j."
endif, endif
p=p+1, a=a+1, i=p, j=p+1
enddo
```

上述算法中利用数据库指针下跳作为矩阵行的移动, 用 not. eof() 作为行结束判断标志, 在比较的过程中, 只要有一次不相等, 则说明不等价, 跳出后去比较下一

列(下一故障)。假若直到比较结束,两列仍相等,就说明此两列 F^{α} 和 F^{β} 等价。将存放在间接库 jian1 中的 zhen1 库的库结构 F^{α} 以删除记录命令 `delete all for field name="F&alpha.&j"` 删除掉。最后就获得了矩阵 E。

2 故障检测矩阵 G 的产生算法

已知输出向量矩阵

$$Z = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \end{bmatrix}, \text{故障矩阵 } E = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \end{bmatrix}$$

将 $Z(X^i) \oplus E^{\alpha}(X^i) =$

$$\begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \end{bmatrix} \oplus \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \end{bmatrix} = G = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

故障检测矩阵 G 的每列 $j(j=1, 2, \dots, n)$, 表示被检测的一个可区分的故障, G 的每行 i 表示所有检测各故障的完全测试集的和集所对应的输入组合 X。若 G 中的 $G^{ij}=1$, 表示 i 行对故障 j 是一个测试, 否则为 0。

矩阵 G 产生的程序算法为:

先将 Z 矩阵作为一个字段并入故障矩阵 E 的数据库 zhen2 中。这样, 将两个矩阵进行异或运算就变换为对一个数据库字段间的关系运算。

```
m1=1
do while m1<=n
use zhen2
do while .not. eof()
if Z<>E&m1
repl E&m1 with 1
else
repl E&m1 with 0
endif;skip;enddo;m1=m1+1
enddo
```

在本算法中, 根据异或运算是两者相等其值为 0, 两者不相等其值为 1 的特点, 用 Z 字段代表输出向量矩阵, 用 E&m1 代表故障矩阵 E 的各字段, 将两者进行比较后, 用 `repl E&m1 with 1` 和 `repl E&m1 with 0` 两条命令表示异或运算后的结果。如此产生的数据库就是故障检测矩阵 G。

3 获得最小全测试集矩阵的算法

对故障检测矩阵 G 进行去掉劣势行, 优势列的变换。获得的矩阵 G^* 为最小全测试集矩阵。对于矩阵

$$G = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \text{若 } i \text{ 行} \subset j \text{ 行, 则称 } i \text{ 行为劣势行; 若 } \alpha$$

列 $\supset \beta$ 列, 则称 α 列为优势列。在本矩阵 G 中, 可以看出 $i^4 \subset i^2, i^4 \subset i^3$, 因此 i^4 行称为劣势行。而 $\alpha^4 \supset \alpha^1, \alpha^4 \supset \alpha^3$, 则称 α^4 列为优势列。将 i^4 行和 α^4 列去掉, 产生了矩

$$\text{阵 } G \text{ 的一个子集 } G^*. G^* = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \text{ 该子集中的各行}$$

所对应的输入向量的组合 X^i 就是最小全测试集。

对矩阵中的两行或两列间进行相容性判断在程序算法中采用判断是否满足下列一组算式:

$$\begin{cases} G_i, \alpha = G_j, \beta \\ G_i, \alpha = G_i, \beta \end{cases} \text{ 且 } \begin{cases} G_i, \alpha \neq G_i, \beta \text{ 或} \\ G_j, \alpha \neq G_j, \beta \end{cases}$$

的方法。如满足, 表示两列或两行不相容, 即不存在优势列或劣势行。如不满足此算式, 继续与下一行或下一列进行比较, 假如直到所有的行(或列)全比较完毕仍不满足上式, 则表示此两列或此两行相容, 有劣势行或优势列存在。进而判断, 若 G_i 中 1 的个数 $< G_j$ 中 1 的个数, 则 $G_i \subset G_j, G_i$ 为劣势行; 若 G_α 中 1 的个数 $> G_\beta$ 中 1 的个数, 则 $G_\alpha \supset G_\beta$ 为优势列。

```
use zhen3
do while .t.
p1=1;p2=p1+1
do while .t.
if r&i(p1)<>r&i(p2).and.r&i(p1)=r&j(p2).and.r&j(p1)=r&i(p2)
bz=0
exit
else
p2=p2+1
if p1=n;bz=1
exit
endif;endif
enddo
if bz=1
delete r&i(p1) for c1<c2
endif
i=i+1
if i>=m;exit
endif
enddo
```

在本算法中, 用 $r&i(p1)$ 代表 $G_i, \alpha, r&i(p2)$ 代表 G_j, β 来进行判断。用 C1 和 C2 分别代表两行中 1 的个数。

本文从数据库的关系式数据结构角度出发, 提出了实现矩阵运算的几种算法, 并在数字集成电路最小全测试集的自动建立过程中加以应用。也可以用于其它类似的矩阵运算中。

参考文献

- [1] [美] H·Y·张, E·曼宁 G·梅茨著. 数字系统的故障诊断. 国防工业出版社. 1979. 7
- [2] 李建勋著. 数字电路与逻辑设计. 科学出版社. 1981.

图像旋转的快速显示技术

浙江大学 CAD&CG 国家重点实验室 (杭州 310027) 李伟青

摘要 在图像处理及图形、图像编辑等领域中,图像的旋转变换是较常用的一种操作。本文从理论上推导了图像旋转变换的递推公式,并给出了旋转变换的具体算法。算法完全消除了费时较多的乘法运算,代之以常数增量的加法运算,因而具有快速的特点。

关键词 图像旋转 快速显示 算法描述

1 引言

在许多计算机图像处理以及图形、图像编辑系统等应用领域中(如广告辅助设计系统、排版系统等)图像的几何变换是最基本的操作。其中,对平移、缩放变换的处理较为简单,只需做一些加法运算就能得到最终结果。对于旋转变换,情况就有些复杂了。

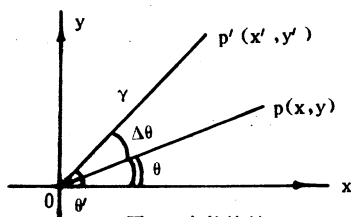


图1 点的旋转

如图1所示,点 $P(x, y)$ 绕坐标原点 O 旋转角度 $\Delta\theta$,至点 $P'(x', y')$, θ 为线段 OP 与 x 轴的夹角, θ' 为线段 OP' 与 x 轴的夹角,线段 OP 的长度为 r 。

从而有

$$\begin{cases} x = r \cos \theta \\ y = r \sin \theta \end{cases} \quad (1)$$

以及
$$\begin{cases} x' = r \cos \theta' \\ y' = r \sin \theta' \end{cases} \quad (2)$$

已知点 P 的坐标 (x, y) ,由(1)可求得 r 和 θ ,又 $\theta' = \theta + \Delta\theta$,由此,(2)式可得到旋转之后的点 P' 的位置 (x', y') 。然而上述过程涉及乘法和三角函数计算,过程较复杂且费时。

2 理论推导

由于 $\theta' = \theta + \Delta\theta$,式(2)可以变为:

$$\begin{cases} x' = r \cos(\theta + \Delta\theta) = r \cos \theta \cos \Delta\theta - r \sin \theta \sin \Delta\theta \\ \quad = x \cos \Delta\theta - y \sin \Delta\theta \\ y' = r \sin(\theta + \Delta\theta) = r \sin \theta \cos \Delta\theta + r \cos \theta \sin \Delta\theta \\ \quad = y \cos \Delta\theta + x \sin \Delta\theta \end{cases} \quad (3)$$

由此看出,点 P' 的位置仅与其对应的变换前的点 P 的位置和旋转角度 $\Delta\theta$ 有关,而与 r, θ 及 θ' 等无关。在旋转角度 $\Delta\theta$ 一定的情况下, $\cos \Delta\theta, \sin \Delta\theta$ 也变成了常量,因此,对于给定的一点 P ,经旋转变换后,由式

(3)可立刻计算出其新的位置 P' 。

由于计算机所处理的图像是以各个像素(pixel)的像素值来表示的,而且各个像素的位置是离散的,通常用整数形式来表示。对于一个位置处于 (x, y) 的像素点 P ,经变换以后的新的位置 (x', y') 未必正好落在一个像素点上,经取整运算以后,就可能使得新图像中的一些像素点在原图像中找不到对应点,这种现象被称为“空穴”。

为了克服这一缺陷,可以采用逆向变换的方法,即对于新图像中的每个像素点,经过逆变换计算出原图像中相应的点的位置。由于新图像中的各点都得到了处理,因此,“空穴”现象自然也就不存在了。这种逆变过程可表示如下:

$$\begin{cases} x = r \cos \theta = r \cos(\theta' - \Delta\theta) = r \cos \theta' \cos \Delta\theta \\ \quad + r \sin \theta' \sin \Delta\theta = x' \cos \Delta\theta + y' \sin \Delta\theta \\ y = r \sin \theta = r \sin(\theta' - \Delta\theta) = r \sin \theta' \cos \Delta\theta \\ \quad - r \cos \theta' \sin \Delta\theta = y' \cos \Delta\theta - x' \sin \Delta\theta \end{cases} \quad (4)$$

然而,式(4)须做4次乘法运算,对于变换之后的大小为 $M \times N$ 的图像,其乘法运算次数将猛增至 $4 \times M \times N$ 次,显然难以达到快速的效果。

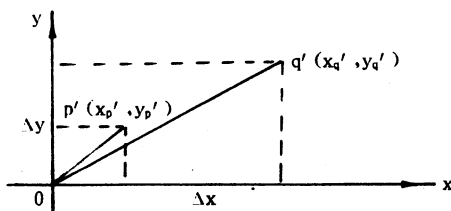


图2 任意两点的位置关系

如图2所示,设 P', Q' 分别是变换以后的图像中的两点,其位置有如下关系式:

$$\begin{cases} x_{q'} = x_{p'} + \Delta x \\ y_{q'} = y_{p'} + \Delta y \end{cases} \quad (5)$$

由式(4)可知, P', Q' 在原图像中的相应的点 P, q 的位置分别为:

一种双终端不同画面的显示控制器

东南大学热能研究所(南京 210018)

周 立

南京师范大学物理系

王晓红

摘 要 本文介绍了采用 MC6845 制作双终端显示控制器的技术,两片 62256 作双终端的显示存储器,可同时显示不同的画面,以满足大量显示曲线及汉字的需要。

关键词 双终端 CRT 控制器 单片机

收稿日期:1994 年元月 4 日

$$\begin{cases} x_p = X_p' \cos \Delta \theta + y_p' \sin \Delta \theta \\ y_p = y_p' \cos \Delta \theta - x_p' \sin \Delta \theta \end{cases} \quad (6)$$

和

$$\begin{cases} x_q = x_q' \cos \Delta \theta + y_q' \sin \Delta \theta \\ y_q = y_q' \cos \Delta \theta - x_q' \sin \Delta \theta \end{cases} \quad (7)$$

将式(5)代入(7)可得

$$\begin{cases} x_q = (X_p' + \Delta x) \cos \Delta \theta + (y_p' + \Delta y) \sin \Delta \theta \\ \quad = X_p' \cos \Delta \theta + y_p' \sin \Delta \theta + \Delta x \cos \Delta \theta + \Delta y \sin \Delta \theta \\ \quad = X_p + \Delta x \cos \Delta \theta + \Delta y \sin \Delta \theta \\ y_q = (y_p' + \Delta y) \cos \Delta \theta - (x_p' + \Delta x) \sin \Delta \theta \\ \quad = y_p' \cos \Delta \theta - x_p' \sin \Delta \theta + \Delta y \cos \Delta \theta - \Delta x \sin \Delta \theta \\ \quad = y_p + \Delta y \cos \Delta \theta - \Delta x \sin \Delta \theta \end{cases} \quad (8)$$

如果 P', q' 为同一行中相邻的两个像素点,即 $\Delta x = 1, \Delta y = 0$, 那么

$$\begin{cases} x_q = x_p + \cos \Delta \theta \\ y_q = y_p - \sin \Delta \theta \end{cases} \quad (9)$$

如果 P', q' 为同一列中相邻的两个像素点,即 $\Delta x = 0, \Delta y = 1$, 那么

$$\begin{cases} x_q = x_p + \sin \Delta \theta \\ y_q = y_p + \cos \Delta \theta \end{cases} \quad (10)$$

由此可知,只要知道了新图像中当前行的第一个点所对应的原图像中的点的位置,就可以通过累加一个常数增量求得该行中其余各点在原图像中的对应位置,以及下一行中的第一个点在原图像中的位置。

3 算法描述

如前面所提到的,图像的位置必须以整数形式表示,而式(9)、(10)的计算结果为实数形式,必须对它们做取整运算。对于大小为 $M \times N$ 的图像,要做 $2 \times M \times N$ 次取整运算,也将会影响到最终的显示速度。

为此,可以预先设置四个数组 $rowx[N]$ 、 $rowy[N]$ 、 $colx[M]$ 和 $coly[M]$,其中, $rowx$ 和 $rowy$ 一组保存新图像中同一行上相邻两点所对应的原图像中的点的位置差, $colx$ 和 $coly$ 一组保存新图像中同一列上相邻两点所对应的原图像中的点的位置差。上述各数组均以整数形式保存。这样,若已知新图像中的点 (i, j) 在原图像中的位置 (x, y) , 则点 $(i, j+1)$ 在老图像中的位置为 $(x', y') = (x + rowx[j+1], y + rowy[j+1])$, 点 $(i+1, j)$ 在原图像中的位置为 $(x', y') = (x + colx[i+1], y + coly[i+1])$ 。

上述四个数组的计算过程为:

考虑新图像中第一行各像素点。置 $rowx[0] = rowy[0] = 0$, 它对应于第一个点,由式(9)知,第二个点与第一个点在原图像中的位置之差为 $(\Delta x = \cos \Delta \theta, \Delta y = -\sin \Delta \theta)$, 从而 $rowx[1] = (\text{int}) \Delta x$, $rowy[1] = (\text{int}) \Delta y$, 并且 $\Delta x = \Delta x + \cos \Delta \theta$, $\Delta y = \Delta y - \sin \Delta \theta$, 同理,分别计算各点与第一点在原图像中的位置差,然后, $rowx[i] = rowx[i] - rowx[i-1]$, $rowy[i] = rowy[i] - rowy[i-1]$ 便是两邻两点在原图像中的位置差。

参考式(10),用同样方法可以求得 $colx$ 、 $coly$ 数组。

由上述过程可以看出,取整运算只做了 $(M+N-2) \times 2$ 次,且排除了乘法运算。

图像旋转变换的算法描述如下:

① 计算 $\sin \Delta \theta$ 和 $\cos \Delta \theta$ 。

② 求出图像中心点的位置,并建立以中心点为原点的坐标系。

③ 求出经过变换之后的图像的大小 $M \times N$, 值 M 、 N 可以通过四个角点旋转之后的位置求得(利用式(4))。

④ 计算 $rowx[N]$ 、 $rowy[N]$ 、 $colx[M]$ 和 $coly[M]$ 四个整形数组的值。

⑤ 利用式(4)计算新图像中左上角点所对应的原图像中的位置。

⑥ 对于每一行 $i (i=0, \dots, M)$, 由 $colx[i]$ 、 $coly[i]$ 及上一行第一点所对应的原图像中的位置计算该行第一点在原图像中的位置;由 $rowx[j]$ 、 $rowy[j] (j=0, \dots, N)$ 计算该行中各点所对应的原图像中的位置,并将相应的像素值置于显示缓冲区中,显示该行中有效部分的像素值。

4 结论

本文所提出的图像旋转的生成算法,在两次循环过程中(算法第 6 步),用位置增量的加法运算取代了求取新位置时的乘法运算,将每点处的取整运算改为预先求取位置差,使取整次数由 $M \times N \times 2$ 次减为 $(M+N-2) \times 2$ 次,并取消了三函数运算,其时间复杂度与图像大小呈线性关系,因此具有快速的特点。

1 引言

本文介绍了一种单色 CRT 显示控制器,采用一片 MC6845 控制两个显示终端的显示,并且同时显示不同的画面,既可显示英文字符,又可显示汉字字符,还可以显示图形曲线。此 CRT 显示控制器,采用了显示 RAM 的地址重叠复用技术,因而双终端双画面的控制并没有占用更多的 RAM 地址空间,对于计算机系统其余模板没有丝毫影响。单 6845 同时驱动双负载,各个显示器的扫描与刷新也和单个显示器的情况完全一样,各个显示器互不影响,而是独立工作。计算机通过软件自由访问所要访问的显示器,软件上和单个显示器时的唯一区别,就是当要从当前访问的显示器 RAM 转而去访问另一个显示器 RAM 时,需要一个切换指令,确定一个为主显示器,一个为副显示器,计算机的常规显示操作都将在主显示器上有所反映。

此双终端 CRT 显示控制器,已用于电站锅炉、汽机测控系统中,采用 8031 作主控 CPU,一个终端显示

锅炉的参数及过程曲线,另一个终端显示汽机的参数与过程曲线,达到了同时显示大量工艺参数的目的。它和单终端显示相比有着突出的优点。单终端显示大量参数时需要人工切换画面,看到一幅就必然看不到另一幅,影响操作人员对工艺流程的及时、准确地判断。实际工程应用中也有采用多个计算机同时显示多个画面的方案,其缺点是成本较高,一般需要计算机间相互通讯以交换数据,这使系统复杂起来,也使系统整体的可靠性下降。

2 硬件设计

此 CRT 控制板的电路图如图 1 所示。

图中未画出 CPU,相关的总线信号已标出,其中:

WR:8031 写信号

RD:8031 读信号

A15~A0:8031 系统的地址总线

D7~D0:8031 系统的数据总线

C9:地址译码所得的片选信号

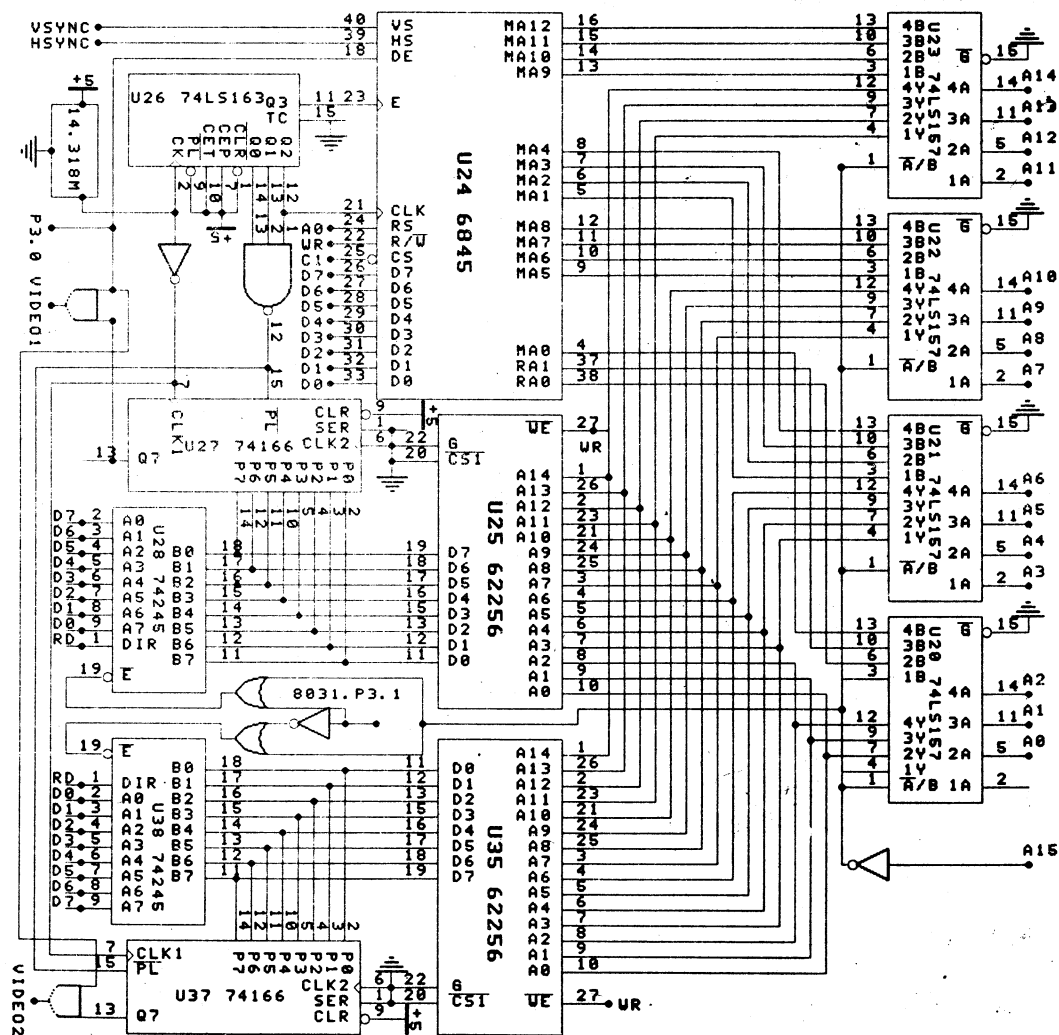


图 1 CRT 控制板电路图

2.1 CPU 对 6845 的控制 CRT 显示控制器的核心元件是 MC6845,它是可编程器件,内部集成有 18 个参数寄存器,1 个编号寄存器,屏幕存储器地址逻辑,扫描地址计数器,同步信号发生器,状态与控制逻辑以及光标与光笔逻辑等许多电路。其工作方式由 CPU 对其参数寄存器初始化置数后确立,6845 依此参数完成其各种功能。

用于连接 6845 与 CPU 的信号为:8 条数据线 D7~D0,片选(\overline{CS})信号,寄存器选择(RS)信号,读/写(R/W)信号和使能(E)信号。R/W 接至总线 WR,RS 信号连接到系统的最低地址位 A0,这样,6845 就占据两个 I/O 地址。当 A0=0 时,访问地址寄存器,给它装入需要的参数寄存器的数;然后将 A0 置于高电平,访问指定的那个寄存器。因此一个典型的寄存器存取操作应该由两个邻接写周期或者由一个写周期紧跟一个读周期组成。E 是 6845 所需的同步时钟信号,用于启动内部 I/O 缓冲器,并且通过数据缓冲器来同步输入至内部寄存器的数据和从内部寄存器输出的数据,E 接至计数器 163 的 Q3 端,即为 6845CLK 输入的 2 分频的连续时钟信号。

2.2 62256 的被访问 CPU 和 6845 都要访问 VRAM62256,不同的是 CPU 对 62256 可读可写,而 6845 对 62256 只读不写,而且是假读。6845 读 62256 时数据并不流向 6845,而是流到 166 的数据输入端。为了解决 CPU 和 6845 同时访问 62256 时造成的“竞争冒险”,采用的措施是 CPU 避让 6845 的办法。因为 6845 对 62256 的访问是由硬件时钟控制的,而 CPU 对 62256 的访问是软件可调节的。实际做法是 CPU 读 6845 的 DE(DISPEN,显示使能输出)端,若 DE=1,则表明 6845 正在访问 62256;若 DE=0,则处于回扫,6845 未访问 62256,此时 CPU 可访问 62256。具体实现方法是采用 4 片四 2 选 1 数据选择器 74LS157, CPU 控制 157 的选择($\overline{A/B}$ 端,若 $\overline{A/B}=0$,则 CPU 的地址线与 62256 的地址线相通,若 $\overline{A/B}=1$,则 6845 的扫描地址线与 62256 的地址线相通。U28 为三态收发器 245,其作用是当 CPU 对 62256 读、写(\overline{RD} 决定数据流向)时导通,此外保持高阻态,起总线隔离的作用。当 CPU 不对 62256 操作时,62256 应始终处于被读状态,这样,它就能始终输出对应于 6845 所送地址的数据。图中 62256 的 G 和 \overline{CS} 接低电平, \overline{WE} 接 WR,当 WR=1 时,62256 处于被读状态,当 WR=0 时,62256 处于被写状态。当 CPU 对 62256 读或写时,A15=1,则有一片 245 被选通,若 A15=0,则两片 245 皆不通。

2.3 视频输出控制 基本点时钟采用 14.318MHz 晶体振荡器输出,该点时钟经计数器 163 的 8 分频后送至 6845 的 CLK,提供 6845 的字符时钟信号。在显示期间内,每经过一个字符时钟,6845 提供一个新的屏存储器地址(MA0~MA12)和光栅扫描线地址信号

(RA0~RA1),62256 的数据线输出也就随之更新;每经过 8 个点时钟(即一个字符时钟),由计数器 163 的 Q0、Q1、Q2 经与非操作的信号就为移位寄存器 166 的置数信号提供一个负脉冲,将 62256 提供的数据置入 166 的寄存器内,并开始移位,移位时钟信号由点时钟经倒相后提供。倒相后,166 的 CLK₁ 的时钟就比点时钟相位相差 180°,使得 166CLK₁ 的上升沿正好落在 166 \overline{PL} 信号下降沿后面半个点时钟周期,从而保证了在 166 置数后半半个点时钟周期时开始移位输出。具体时序见图 2。

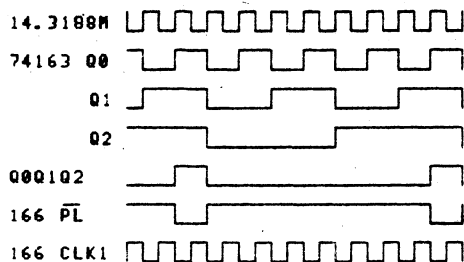


图 2 时序图

6845 的 DE 与 166 的 QT 相与构成视频输出信号。在 6845 的电子束回扫期,166 仍不断移位输出数据,此时 DE=0,视频输出为低电平,无电子发射,使回扫期在屏幕上不留痕迹。水平同步信号和垂直同步信号由 6845 的 HS 和 VS 提供。

2.4 双视频输出 双终端显示需要两个不同的视频信号输出,同单端显示相比,增加了部分电路,包括 62256,245 和 166 等。增加的一片 62256 用来存放第二个显示器的显示内容,两片 62256 的地址相重叠,采用 8031 的 P_{3.1}脚来控制 CPU 将对那个显示器内容进行操作。增加的 245 功能仍是总线隔离。增加的 166 承担同样的串行移位操作,其置数信号和移位时钟信号与另一个 166 的信号并联。这样,一片 6845 就同时控制了二个显示器的显示,这两个显示器工作方式一样,但显示内容不同。

3 软件设计

软件编程包括设置屏幕显示方式和设定屏幕显示内容两部分。屏幕显示方式由对 6845 寄存器的初始化完成。在本电路设计中,62256 的地址线中用了两根光栅扫描线地址信号 RA₀、RA₁,因而每字符行有 4 线,每个字符是 4×8 的点阵图块。对于 640×400 分辨率的显示,可按如下方式对 6845 的寄存器初始化:

```
R0=73H   R1=50H   R2=5AH   R3=0AH
R4=7FH   R5=03H   R6=64H   R7=6AH
R8=03H   R9=03H   R10=06H  R11=08H
R12=00H  R13=00H  R14=00H  R15=00H
```

设定屏幕显示内容是通过 62256 的内容作更改来实现的。由 P3.1 指明当时可被 CPU 访问的 62256,由程序将要在屏幕上某坐标处显示的点(下转 23 页)

DOS 设备驱动程序原理浅析

济南市 55152 部队计算中心 (250031) 周永利 席春红

摘 要 本文分析了 DOS 设备驱动程序的工作原理, 对 DOS 中各主要功能接口的功能和调用约定进行了详细的讨论。

设备驱动程序是任何操作系统的必不可少的部分, 它们实现了计算机系统所有附属设备的一个标准接口。换句话说, 设备驱动程序提供了存取各种不同的硬件设备的公用机制, 是链接操作系统内核与系统外部设备 I/O 驱动之间的关键部件, 通常被认为是操作系统中最保密的一个组成部分。它包含与硬件直接相关的设备驱动, 不同的计算机系统都有其制造商提供的特定的设备驱动程序包。

DOS 是具有一套分层接口的操作系统, 每一个接口都有其特定的功能和调用约定 (Calling Convention)。下面是 DOS 的主要功能接口:

- 应用编程接口 (Application Programming Interface 简称 API)
- 设备驱动程序接口
- 基本输入/输出系统 (BIOS) 接口
- 硬件设备接口

DOS 设备驱动程序与 DOS 的这些功能性接口交互发生关系。可以这样说, DOS 设备驱动程序就是通过逐层利用以上各层接口而实现对硬件设备的设备驱动的。因此, 要搞清 DOS 设备驱动程序的工作原理, 就必须讨论以上每层接口的功能和调用约定, 经过对各接口功能和调用约定的讨论之后, DOS 设备驱动程序工作原理也就昭然若揭了。我们先讨论一下软件分类的观点以及这些功能接口是如何与软件接口体系相关的, 然后对每一层功能接口进行详细分析。

1 软件接口体系结构

接口表示一些抽象的功能或功能集的实现。软件接口的体系结构有两个主要组成部分, 第一个组成部分是可以通过接口存取的功能性抽象集。但是, 当实现时这些抽象又变成允许我们创建, 存取和删除文件的文件管理接口。第二个组成部分是用于存取基础功能性抽象的标准化调用约定。虽然在一个编程环境中存在有无穷多种调用功能的方法。但是体系化了的软件接口代表调用这些基础功能的简单 (标准化) 方法。

接口的定义隐含着软件功能的层次。DOS 中的每一层接口都有其特定的目的。应用编程接口 (API) 处理 DOS 提供给在 DOS 环境中执行的应用程序的服

务。这些服务 (或称为操作系统的功能调用) 是非常抽象的, 或者说是独立于硬件的。例如, 应用程序可以要求 DOS 创建一个名为 MYFILE 的文件。这种类型的操作系统服务是与硬件毫无关系的。该服务只是表示与操作系统的这一特定层次相关的高级抽象。

这并不是说应用程序打开 MYFILE 的请求不会引起一些与硬件有关的操作细节的出现。不过这些与硬件有关的操作只出现在从应用编程到硬件有关接口层的请求的主要部分。

我们可以把系统的体系结构看作三角形。而应用程序的 DOS 服务请求位于三角形之项 (见图 1)。DOS 服务管理程序接收并处理 DOS 服务请求。每一 DOS 服务向 DOS 设备驱动程序发出后继的请求。DOS 设备驱动程序依次向 BIOS 层发出请求。最后, BIOS 层向硬件自身发出请求 (见图 1)。

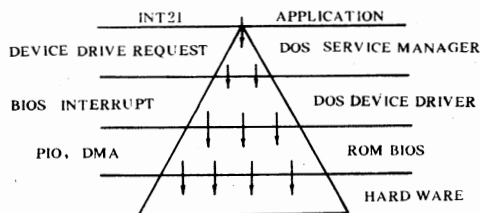


图 1 系统请求三角形图示

在系统体系结构中作为高层的请求结果, 向后继层发出的请求数目被称为“输出端数” (fan-out), 输出端数由请求三角形的底边宽度来形象描述。

当请求三角形的输出端数减少时, 系统中的额外开销 (Overhead) 就增加。这是因为当一层的系统体系结构中的后继层作请求时, 请求被编码为适合后继层调用约定的形式。后继层为了执行操作必须对请求进行译码。当输出端数逼近某一个时, 系统就接近只作编码和译码接口请求的那一状态。

2 应用编程接口

应用编程接口 (API) 是 DOS 服务或功能调用的常用名称。具体地说, API 指的是 INT21 服务。80X86 系列处理器提供了软件中断 (Software Interrupt) 功能。该功能的形式是中断指令 (INT) 后加所要的中断号。因此, INT21 只是引起中断向量位置被存取的中断指令。

DOS API 为应用程序员提供了大量的服务过程。如果应用程序员在其程序中自己写这些,则不知要花多少时间。即使应用程序员能够大其程序中写这些服务过程,但是这些服务过程中的许多要被连续更新以反映 DOS 版本的修改。由于这个原因,应用程序员应尽量使用 DOS 提供的各种功能。

如前所述,DOS API 提供可以通过 INT21 接口存取很多有用的服务。该接口(或调用约定)要求 CPU 的 AH 寄存器表示所请求的 DOS 服务。任何附加信息通过其他 CPU 寄存器提供。

3 设备驱动程序接口

DOS 设备驱动程序接口与应用编程接口类似,不过还是有一些重要的区别。DOS API 是应用程序和 DOS 服务管理程序之间的接口,相反,设备驱动程序是 DOS 服务管理与 BIOS 或硬件自身之间的接口。这一区别不是很重要,因为 DOS 是一个不可重入操作系统(reentrant)。因此,当应用的请求从 DOS API 到设备驱动程序接口,到 BIOS,最终到硬件的在系统中进行时,在可以执行什么类型的操作中,是很少有灵活性的。

另一观察方法是当请求到达设备驱动程序接口时,设备驱动程序不可能通过 DOSAPI 发出请求。简言之,越接近硬件,对请求和服务越不抽象。

3.1 DOS 设备驱动程序结构与管理

DOS 设备驱动程序是一内存映象文件(.COM),包含实现设备连接或实现所要求的所有逻辑。虽然设备驱动程序是标准类型的文件,但它有一主要区别。一般,.COM 文件要求从位置 100H 启动。这一要求使得 DOS 可以在加载 .COM 文件之前先在内存中创建 256 字节的程序段前缀(Program Segment Prefix,简称 PSP)。如果 .COM 文件从位置 00H 启动,则当 DOS 加载文件时会覆盖 PSP 而使程序不能正确运行。

DOS 设备驱动程序不是从位置 100H 启动,而是从位置 00H。可能会奇怪 DOS 为什么允许设备驱动程序而不允许一般的 .COM 文件从位置 00H 启动呢?答案很简单,因为 DOS 设备驱动程序表示 DOS 核心的延伸。因此,DOS 已申请了内存和指定内部数据结构以管理系统中每一设备驱动程序的位置和操作。进一步说,一旦 DOS 设备驱动程序被加载,它们的内存地址不改变。不过 .COM 文件经常是被加载执行然后从内存中删除,它们被重复执行时不能保证它们占据同一内存位置。下面讨论 DOS 如何管理这些设备驱动程序。

一个 DOS 设备驱动程序必须由“设备头”,“策略过程”以及“中断过程”三部分组成。

所有 DOS 设备驱动程序都必须有一个位于位置

0 的 DOS 设备驱动程序头。它是区别于其它设备驱动程序标志,含有设备属性,逻辑设备名或设备部件数。它类似于 .COM 文件的 PSP,DOS 用设备驱动程序头把所有设备驱动程序连接为单向链表。因此,可以找到 DOS 设备驱动程序的链表头,就所看到系统中的所有设备,下面是 C 结构描述的 DOS 设备驱动程序头的特定形式:

```
struct DDH_struct
{
    struct DDH_struct far * next_DDHI;
    unsigned int ddh_attribute;
    unsigned int ddh_strategy;
    unsigned int ddh_interrupt;
    unsigned int ddh_name[8];
}
```

策略过程将 DOS 或应用程序传送的“I/O 请求头”的双字指针保存在驱动程序代码段可寻址的双字变量中,不作任何的 I/O 操作。通常,DOS 或应用程序执行设备的输入或输出操作时,都必须向驱动程序提出“I/O 请求头”,因此,首先通过调用策略过程将该请求头的指针予以保存,然后再调用中断过程,根据双字指针指向的“I/O 请求头”的信息,由相应的子过程完成实际的 I/O 操作。

中断过程才是设备驱动程序的核心。它真正完成驱动程序实际 I/O 的各种功能:依据“I/O 请求头”的信息,转入内部相应的功能,完成 I/O 操作后在“I/O 请求头”中返回有关的信息和状态。通常,它由以下三个部分组成:

(1)一个总的入口处用于保存所有在程序中要修改的寄存器,然后从“I/O 请求头”中获取指定的命令码,转向相应子程序。

(2)包含所有 DOS 请求功能的子程序包。对应每一个命令码都必须有一个相应的可执行的子程序,不管该子程序是否完成有实际价值的工作。

(3)一个总的出口处用于返回状态和其它有用信息到“I/O 请求头”,并且恢复由入口处保存的所有寄存器。

3.2 DOS 设备驱动程序分类

DOS 中每一个设备必须具备相关的设备驱动程序才能被存取,因此,所有通讯设备(COM1,COM2,……)以及打印设备(设备 LPT1,LPT2,...,PRN,...)必须有相关的设备驱动程序头。

要搞清 DOS 设备驱动程序的分类就必须理解字符设备和块设备之间的区别,字符设备被设计为执行可变长度的 I/O 操作。这意味着字符设备可以操作单个字符或任意数目的字符。字符设

如何从二维图像中得到主动脉中胶原纤维的空间位置

华西医科大学计算中心(成都 610041) 高云钦

摘要 本文探讨了如何从二维图像中得到主动脉中胶原纤维空间位置的原理,并试图用数字方法解决这一问题。实践证明,这一方法是行之有效的。

关键词 空间位置 二维图像 胶原纤维

1 概述

心血管疾病是人类死亡的重要原因之一。因此,弄清楚心血管组织的结构、病变的变化,从而找出预防和医治心血管疾病的有效方法是造福于千百万人的重要课题。

从已发表过的研究文章中已经知道,血管结构的主要组成成分有胶原纤维(Collagen Fibers),弹性纤维(Elastin Fibers),平滑肌细胞(Smooth Muscle Cells),成纤维细胞(Fibroblasts)和其它(Ground Substance)等等。然而,这些成分各自在血管组织结构中所占的比例,它们在血管的外膜(Adventitia)、中膜

(Media)、和内膜(Intima)中的分布情况,以及这些纤维在血管组织结构中的空间位置如何排列……一系列问题尚需进一步地进行研究和探讨。

目前,普遍使用的生物医学图像处理系统大多属于二维图像处理系统,并且采用二维图像系统进行定量分析已取得了一定的成果。因此,无论从经济上或技术上来说,探讨二维图像系统的进一步应用都是合理的。本文仅就如何从血管组织结构切片的二维图像中研究胶原纤维的空间分布的方法作一探讨,供感兴趣的读者参考。

从近年来的许多组织学实验中已知,单根的胶原纤维在血管组织结构中,既不平行于血管的轴线,也不与血管壁相垂直,而且也不成直线状。如图1所示:

收稿日期:1994年元月6日

备一般支持打印机,显示器,键盘,局域网和通讯设备之类的设备,每一个字符设备都有一个设备驱动程序和一个名字。

块设备被设计为执行固定长度的I/O操作。块设备一般支持软盘和硬盘。不同于字符设备驱动程序。DOS中的块设备驱动程序设有名字。块设备驱动程序可以负责或控制一个以上磁盘设备。

4 BIOS接口

基本输入输出(BIOS)系统接口是驻留在系统板以及相关的适配器板上的ROM中的软件。BIOS提供系统中大部分I/O设备的设备级控制,ROM模块可能被定位在加入的适配器板上,以提供该特定选项板的设备级控制。

BIOS子程序使应用程序可以执行块或字符级I/O操作而不必关心设备寻址或设备的功能特性。类似地,物理硬件设备可以被一根根本不同的设备所代替而不影响BIOS所维护的系统的实际性能。

BIOS的目标是给系统提供实际的接口并使程序员减少考虑与硬件设备的特定性质相关的问题。BIOS接口把应用程序与硬件以及其他绝缘开来。这样,新设备可以被加入到系统中而不影响其他设备。用这种方

法,硬件设备对于应用程序只是逻辑设备,而应用程序也不受硬件修改或改进的影响。

5 硬件设备接口

硬件设备接口是系统中最低级的接口。该接口允许软件直接控制与系统相连的硬件适配器或设备,硬件的直接控制一般是通过下面两种方法中的一种来完成的。

硬件控制的第一种技术是编程I/O。编程I/O使用INTEL80X86系列的IN和OUT指令把数据传送到I/O适配器或设备,或I/O适配器或设备传输过来。因为指令IN或OUT需要与设备通讯,CPU必须被编程为在适当时间执行这些指令。

硬件控制的第二种控制技术是内存映像I/O。该接口技术要求硬件设备译码特定的内存地址。每当这些特定内存地址在地址总线上存在时,硬件设备读取写到数据总线上并且执行I/O操作。

虽然硬件的直接控制看上去十分简单,但是应注意每一硬件设备都有不同的体系结构,因此,两个不同的硬件设备可以被设计为用同样方法(编程I/O或内存映像I/O)与系统接口,同时有完全不同的控制要求。



图1 人的胶原纤维的横切面和纵切面电镜图

(×100,000)

从图1中还可以看到,在多根胶原纤维之间也不是相互平行的。因此,无论横切或是纵切血管组织的剖面都无法直接确定胶原纤维在血管组织结构中的空间位置。

作者通过大量的实验研究,并从数学方法上提出了这一问题的如下的解决方法。

首先假定,在局部范围内,可以近似地将胶原纤维看成是一圆柱体状。那么,当垂直于血管轴线作血管组织的切片时,在该切片上的胶原纤维截面大多数是椭圆形。世界著名的生物力学创始人冯元桢博士认为,这一假定是合理的。

在这一假定的前提下,胶原纤维在血管组织结构中的空间位置就可由两个相互垂直的平面上的两个角度 α 和 β ,和X-Y平面上的圆心坐标值X,Y决定。如图2所示:

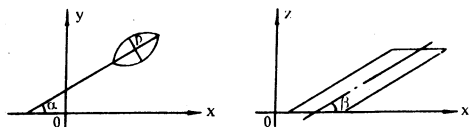


图2(a)水平面内的参数 图2(b)

从图2(a)中可以看出,胶原纤维在血管组织结构中空间位置的三个参数:

α —胶原纤维横切面(椭圆)的长径与X轴的夹角。

X,Y—胶原纤维横切面重心P的X,Y坐标值。

可以较容易地从二维图像处理系统中测得。但是,另一表征其空间位置的角度 β ,就不可能直接地从同一切片的二维图像中测得。

将图2(b)的图形补充如图3所示。

图中:L—胶原纤维任意角度截面的长径。

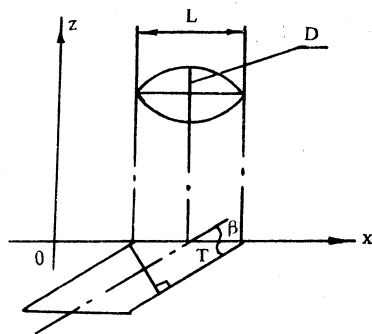
D—胶原纤维的直径。

从图3中知, $L\gamma = L\beta$

$$\sin\gamma = \sin\beta = \frac{D}{L}$$

$$\text{即 } \beta = \arcsin \frac{D}{L}$$

式中L和D两个参数可以从X-Y平面的二维

图3 β 角的计算图

图像处理系统中直接测得,然后,经过上式的计算就可求出该胶原纤维另一个表征其空间位置的角度 β 。

对许多根胶原纤维都求出上述的一组参数后,就可按统计分析的方法,找出该切面前后胶原纤维在血管组织结构中的空间分布情况,包括胶原纤维在血管切面外膜、中膜和内膜各部分的分布数量、密度,以及其主要的空间走向情况等参数。如果将整根血管的不同部位进行如此的分析,就可弄清楚这一组成血管的重要成份在血管组织结构中的情况,进而就可弄清高血压和其他心血管疾病的血管组织结构组成成份的变化情况。

作者在冯元桢博士的实验室,采用BioScan公司的OPTIMAS二维图像处理软件,自动地从大白鼠主动脉电镜切片的测量中获取了如下的一些参数:

D—胶原纤维横切面(椭圆)的短径,也就是胶原纤维的直径。

L—胶原纤维横切面的长径。

A—胶原纤维横切面(椭圆)的面积。

α —胶原纤维横切面长径L与X轴的夹角。

X,Y—胶原纤维的X,Y坐标位置。

显然,这些参数已足以对描绘在血管组织结构中胶原纤维的空间位置和走向所需要的指标进行必要的统计分析和计算了,并可对胶原纤维在血管组织切面外膜、中膜和内膜中的分布数量和密度等指标进行统计和计算。

实践证明,上述方法是成功的,同样也可适用于其它领域的类似情况。

参考文献

- 1 徐建华. 图像处理与分析. 科技出版社. 1992
- 2 L. C. Junqueira et al: Basic Histology. LANGE Medical Publications 1977
- 3 Image Compression, Medical Imaging V; Image Capture, Formatting, and Display, Y. Kim (Ed), Proc SPIE 1444: 240—290, 1991
- 4 Medical Imaging VI: Image Processing, MH Loew (Ed), Proc SPIE 1652, 1992

微机市场预测联合利润最大模型

张鸿鸣

刘铁军

王秀兰

吉林省计算机技术研究所(长春 130012)

哈尔滨师范大学

摘 要 本文重点研究微机市场预测联合利润最大模型和它相关的卡特尔理论。

关键词 联合利润最大模型 卡特尔理论 纯质双头 奏伯霖妥协解

前两篇文章已经对微机市场预测猜测变量模型和对策论模型进行了研究,猜测变量模型是假定骨干厂家之间无勾结行为时的模型。虽然很多国家的法律都禁止骨干厂家就产量和价格进行妥协的行为,但是,聪明的骨干厂厂家常常能以不同的方式达成不同程度的妥协、勾结或共谋,以便实现自己获利的目的。

我们在本文就骨干厂家的联合、勾结行为研究微机市场预测联合利润最大模型。

1 纯质双头骨干(垄断)厂家联合最大利润模型

现实中的纯质双头骨干(垄断)厂家不一定那样单纯,也不一定象有人假定的那样以降低价格同对方争夺市场,更未必象猜测变量模型所描述的那样只关心自己的利益而去占对方厂家的便宜。也许是他们出于各自的利益而对对方采取妥协的方法。这种复杂的独占行为无法一概而论。为此,奏伯霖(E·Chamberlin)提出了妥协解,为这一目的带来了极大的可能。他主张为使纯质双头骨干厂家的各自利润最大,就必须应首先使他们的联合利润最大,于是得出各厂家决定产量的标准为:

$$\begin{cases} \frac{\partial(V_A+V_B)}{\partial q_A}=0 & (1) \\ \frac{\partial(V_A+V_B)}{\partial q_B}=0 & (2) \end{cases}$$

又因为联合利润为

$$\begin{aligned} V &= V_A + V_B = P \cdot q_A - TC_A + P \cdot q_B - TC_B \\ &= [90 - (1/3)(q_A + q_B)]q_A - ((1/6)q_A^2 + 60q_A + 50) + \\ &\quad + [90 - (1/3)(q_A + q_B)]q_B - ((190/3)q_B + 100) \end{aligned}$$

$$\text{故令 } \frac{\partial(V_A+V_B)}{\partial q_A} = 30 - q_A - (2/3)q_B = 0 \quad (3)$$

$$\frac{\partial(V_A+V_B)}{\partial q_B} = (80/3) - (2/3)q_A - (2/3)q_B = 0 \quad (4)$$

由(3)和(4)联立方程组便可解得

$$\begin{cases} q_A = 10 \\ q_B = 30 \end{cases}$$

因此,市场价格便为

$$90 - (1/3)(q_A + q_B) = 90 - (1/3)(10 + 30) = (230/3)(\text{元})$$

每个厂家的利润分别为

$$V = (230/3) \times 10 - (1/6) \times 10^2 - 60 \times 10 - 5 = 100(\text{元})$$

$$V = (230/3) \times 30 - (190/3) \times 30 - 100 = 300(\text{元})$$

这就是奏伯霖的妥协解,其价格比猜测变量模型解的任何情况都高,行业的产量最小,利润之和最大。

2 价格领导模型

在许多国家中,对骨干厂家之间就产量和价格进行勾结和达成君子协定是非法的(除政府允许的)。这促使骨干厂家去寻求更不正规的勾结方式——暗中默契。价格领导模型便是暗中默契的主要方式。一个价格领导者就是这样的一家公司(厂家),其价格变动为行业中的其它厂家所一致遵从。

如果一个厂家是本行业中的价格领导者,这个厂家的需求曲线就不会出现弯折。因为一旦他决定了某一产品的价格,竞争对手厂家都会跟随。而如果一个厂家不是行业中的价格领导者,他就可能遇到弯折的需求曲线,除非他对全行业来说十分微小,以致他的定价活动大家都不予以理睬。当一某行业中出现一个实力强、规模大,在行业中处于举足轻重地位的厂家时,这一厂家就可能成为该行业的价格领导者——主宰厂家的价格领导模型。或者出于骨干厂家们的共同利益,他们一致推举,公认某一厂家作为价格领导者——指标式的价格领导模型。

由于反垄断法的限制,主宰厂家的价格领导模型的前提假设是,必须能维持其它厂家的生存。虽然该主宰厂家具有决定价格的能力,但它不能凭借自己的优势消灭所有的其它厂家。它所制定的价格乃是在其它厂家能维持生存的前提下使自己的利润达到最大。换言之,此主宰厂家销售量乃是由其它厂家供给不足而得来的。

在价格领导模型中,其它厂家无决定价格的能力,只是(领导)价格的接受者。因此,除价格领导者外,其它厂家都有供给曲线存在,他们的总供给为每个厂家的供给(边际成本)曲线的水平加总,设为 ΣMC_i 。市场的需求量减去这些其它厂家的总供给量,便是主宰厂家可销售的数量。或称主宰厂家所面临的需求曲线,以 D_d 表示,即 $D_d = D - \Sigma MC_i$ 。

设某一行业为主宰厂家的价格领导。如图 2.1 所示,若当市场价格为 P_0 时,主宰厂家无法销售,且市场的需求量又恰好等于其它厂家的供给量,故这时主宰

厂家只能把价格定在 P_0 以下。若当市场价格在 P_1 时, 其它厂家无供给量, 而只能由主宰厂家销售。据此可断言, 主宰厂家的需求曲线 D_d 必须通过点 P_0 和点 H 。由此可知, 主宰厂家所决定的价格应在 P_0 和 P_1 之间。具体的价格还需依主宰厂家的边际成本曲线 MC_d 和边际收益曲线 MR_d 的位置来确定, 也正如图 2·1 所示, MC_d 和 MR_d 相交于点 E , 那么主宰厂家将把市场价格定在 P_E 。

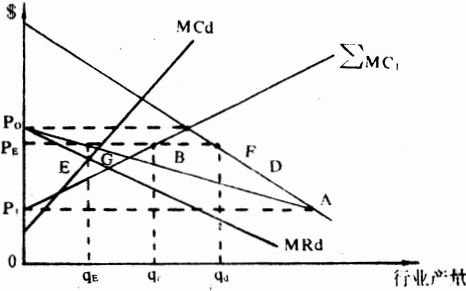


图 2.1 主宰厂家的价格领导模型

因为在同一价格之下, 主宰厂家销售量为市场需求量减去其它厂家的销售量, 故 $BF = P_E F - P_E B = P_E G = q_E = q_d - q_1$ 为主宰厂家的销售量, 而其它厂家的销售量为 $P_E B = q_1$ 。

现在设某行业的需求曲线为 $P = 380 - 2Q$
价格领导厂家的成本函数为 $TC(1/2)q^2 + 10q + 500$
另有 5 家小厂家, 其成本函数皆为

$$TC_i = q^2 + 20q + 100 \quad (i=1, 2, \dots, 5)$$

试求: (1) 价格领导厂家的产量和利润, 以及它所确定的市场价格;

(2) 每个小厂家的产量及利润。首先求 (1)。为此, 给出一个主宰厂家的价格领导模型。每个小厂家的供给曲线为

$$P = MC_i = dTC_i/dq = 2q + 20 \quad (i=1, 2, \dots, 5)$$

即 $q = (1/2)P - 10$

于是 5 个小厂家的供给曲线 ΣMC_i 为

$$Q = 5q = (5/2)P - 50$$

由此可知, 当 $P > 20$ 时, 小厂家才有供给量。因此可得价格领导厂家的需求曲线 D_d 为

$$q = 240 - 3P \quad \text{即} \quad P = 80 - (1/3)q$$

而领导价格必在 $20 < P_E < 80$ 范围内。因为价格领导厂家的边际成本和边际收益分别为

$$MC_d = dTC/dq = q + 10 \quad MR_d = 80 - (2/3)q$$

令 $MC_d = MR_d$, 即 $q + 10 = 80 - (2/3)q$

可解得价格领导厂家的产量为 $q_E = 42$ 单位。于是价格领导厂家把市场价格定为

$$P = 80 - (1/3)q = 80 - (1/3) \times 42 = 66 \text{ (元)}$$

价格领导厂家的利润为

$$V = 66 \times 42 - TC(42) = 2772 - 1802 = 970 \text{ (元)}$$

下面求 (2)。为此, 把领导价格代入需求曲线便可得整个市场的需求量为

$$Q = 190 - (1/2) \times 66 = 157 \text{ (单位)}$$

每个小厂家的产量为

$$q_i = (157 - 42)/5 = 23 \text{ (单位)} \quad (i=1, 2, \dots, 5)$$

每个小厂家的利润为

$$V_i = 66 \times 23 - TC_i(23) = 1518 - 1089 = 429 \text{ (元)}$$

如果一个行业中出现一位公证的价格领导者, 或是骨干厂家为了避免彼此间的竞争而发生损失, 而由他们中间推举一个代表厂家, 以该厂家所定的价格作为他们价格的指标——指标式的价格领导模型, 那么就会发生一种更加狡猾的共谋。这个代表厂家 (此厂家可能是主宰厂家, 也可能不是) 敏锐地注视着市场需求和各厂家的成本水平, 以便定出适当的价格, 然后其他厂家跟随。并且在该价格下, 所有厂家都将获得利益。

3 卡特尔理论

骨干 (垄断) 厂家为了各自的利益进行勾结的典型方式是卡特尔。它是目前流行的寡头垄断理论中最重要的内容之一。卡特尔在资本主义国家是一个垄断生产厂家的组织, 参加这个组织的每一个成员通过协议的方式, 共同确定市场价格, 分配产量及市场份额; 其目的是使这个组织得到更大的垄断利润, 而使每个成员从中获利。

在寡头垄断行业中, 垄断者为了避免彼此相互竞争而使他们造成损失, 常常通过正式或非正式的甚至是口头上就价格、产量等达成“君子协议”而形成卡特尔。就每一个垄断厂家达成卡特尔协议的动机来看, 完全是出于他们各自对垄断利润的追求。但是, 他们各自的利益必须通过共同的利益来实现, 这是形成卡特尔组织的最强有力的内在动力, 便促使卡特尔协议在资本主义国家中普遍存在。石油输出国组织被认为是典型的国际间的卡特尔组织。

卡特尔组织内的每一个厂家不是由单一的公司所拥有, 每个厂家的产量与售价也不容易得到准确而有效地控制。在卡特尔组织中, 整体的利润最大, 未必是每个成员的最终目标。因为自我利益往往驱使他们不遵守卡特尔协议而进行欺骗。

形成卡特尔组织是有困难的。在经济学的角度来看, 要使许多不同骨干生产厂家形成卡特尔组织以求整体的最大利益, 它和完全竞争市场改为多工场完全垄断有些相似, 亦即要有下面两个步骤: (1) 一些厂家必须关闭; (2) 每个厂家必须减少产量。换言之, 卡特尔组织要达到整体利益最大, 它希望成本较高的厂家能退出此行业, 并且在整个行业内减少产量,

提高价格。但有谁愿意退出呢?因此,卡特尔组织及其成员必须从未来所预见能获得的利润中付出一笔适当的补偿费给予退出的厂家。但筹集这笔费用又是一件不容易的事情。因此,那个厂家退出和补偿费用的筹集,是形成卡特尔组织的两大困难。

另外,每个厂家的产品质量可能不同,每个厂家所拥有的产品资源也具有差异。这些差异会导致每个厂家的行为及市场价格和产量的看法不同。产品质量较好的厂家因容易找到更多的用户而希望把价格定高些,资源较富的而且可长久经营的厂家,希望市场价格不要太高。因为市场价格太高,将会加速新技术或新产品(替代品)的开发。从长远角度来讲对他们是非常不利的。因此,形成统一价格是有困难的。另一方面,即使是卡特尔形成了,它又面临着是否能够长期稳定地维持下去的难题。在卡特尔组织的配额下,虽然可以达到整体的利润最大,但对每个厂家本身而言,未必也是利润最大。因此,他们很可能自己悄悄地增加产量,并把价格降低些。特别是协议的市场价格超过厂家的长期边际成本时,厂家更会如此。尽管有些厂家不改变价格,但他可以改善产品质量,扩大产量,以争得更多的买主。这些都可以使他们获得更多的利润。虽然卡特尔也能依产品质量的不同来确定出一个较详细的价格表及市场分配的界限,但无论如何对其成员的欺骗行为有时是很难发现和加以控制的。不执行卡特尔协议的厂家会大大增加利润,因而将会导致其他厂家也不执行卡特尔协议。假若有相当多的厂家为了自家的利益而不执行卡特尔协议,那么卡特尔组织必将会崩溃。其

结果使每个厂家都不能得到好处。于是,出于他们各自的利益还会勾结成新的卡特尔。从上述的分析,我们可以得出卡特尔理论中的两个既重要又相互矛盾的定理:

定理 3·1:若卡特尔的每个成员厂家都遵守卡特尔协议,那么通过垄断利润的实现,每个卡特尔厂家都可以从中获益。

定理 3·2:若其他成员都不欺骗,那么每个卡特尔成员厂家便可以通过不遵守卡特尔协议的办法获益。

除非有政府或法律的支持,大多数卡特尔才会长久存在。不然,由于卡特尔组织的松散和成员厂家的私心,执行卡特尔协议是非常困难的,所以卡特尔组织是不稳定的,不会长期存在。

总之,上面对微机市场预测联合利润最大模型以及和它有关的卡特尔理论进行了较详细的研究,笔者希望能很好地运用这个模型及微型计算机对市场进行科学预测,使其对社会主义市场经济有一定的指导意义,并希望它能产生巨大的经济效益,为国家的改革开放做出更大贡献。

参考文献

- 1 中国科学院计算中心概率统计组编著. 科学出版社, 1979年4月
- 2 中山大学数学力学系常微分方程组编. 人民教育出版社, 1978年12月
- 3 A. 拉尔斯登, H. S. 维尔夫著, 徐献瑜等译. 数字计算机上的数学方法, 上海科学技术出版社, 1963年

(上接 16 页) 阵图形转化为对应 62256 一定地址的数据, 将此数据写入 62256。当分辨率为 640×400 时, 每字符行含有 4 线, 屏幕上共有 100 字符行; 每字符为 8 点阵宽, 每行有 80 字符列。若以屏幕左上角为坐标原点, 原点对应于数据的 D0 位, $0 \leq x \leq 639$, $0 \leq y \leq 399$, 则点 (x, y) 对应于显示 RAM 相应字节的地址的高位(A14~A2)为:

$80 \times \text{int}(y/4) + \text{int}(x/8) \text{int}$ 意思为取整数。低位(A1、A0)为 $y/4$ 的余数, 显示 RAM 中相应字节的位数为 $x/8$ 的余数。

例如, 若要在 (313, 218) 处显示一个点, 则地址高位为

$$80 \times \text{int}(218/4) + \text{int}(313/8) = 80 \times 54 + 39 = 4359 = 1000100000111\text{B}$$

(218/4) 的余数为 2, 则 A1、A0=10B

(313/8) 的余数为 1, 则对应位为 D1 位。

则写点程序为先读地址 A15 ~ A0 =

1100010000011110B=C41EH 的数据, 将所得数据 D1 位置 1 后, 再送入此地址; 若擦去此点, 只要将 D1 位清零即可。

屏幕某处写点是最基本的操作, 在此基础上, 可完成画线及显示汉字等操作。因为汉字是一个点阵图块, 只要将汉字点阵码送至 62256 即可, 汉字点阵码的获取方法很多, 在此不再多述。

4 结束语

如此的双终端设计成本很低, 却很实用, 解决了工程应用中的多参数、多曲线同时显示的问题, 具有强大的生命力。

参考文献

- 1 [美]. Gerry Kane 著. CRT 控制器手册. 成都市兴隆电子仪器厂情报室, 1988年4月
- 2 何立民编著. 单片机应用系统设计. 北京航空航天大学出版社, 1990年

步进电机智能控制卡

上海交通大学动力机械工程系 (200030) 孙传铮

摘要 智能控制接口是一般单功能控制接口的进一步发展,步进电机智能控制卡是这样一种接口,该卡上利用 8031 单片机和专用控制芯片等构成了 PC 机总线接口卡,能对三台步进电机同时进行实时、并行、多功能控制,从而减轻了主机的负荷,在软件设计上,用户只要使用简单的命令,不用复杂的编程就能实现所要求的控制。

关键词 步进电机 智能控制接口 分频系数 步数

步进电机是一种将数字信号直接转换成相应的角位移或线性位移的控制驱动装置。步进电机具有快速启动、停止的特点,其驱动速度和控制脉冲能严格同步,具有较高的重复定位精度,能实现线性加、减速,运行速度和步距不受电源波动及负载变化的影响,所以在许多工业领域中得到了广泛的应用。以往的微机及单片机步进电机控制系统是采用软件的方法,控制输出端的输出脉冲来驱动步进电机的每一相,这种方法虽然硬件简单,但要占用主机 CPU 较多的时间,控制软件复杂,难以实现对多台步进电机进行同时控制和变速控制。为了能实现同时对多台步进电机进行多功能的控制,并且能根据主机的命令安排,由步进电机控制系统独立完成全部的控制,减少主机的负担,设计了 PC 总线工业控制微机的步进电机智能控制卡,用于同时对三轴(X、Y、Z 向)的步进电机进行多功能的控制。

1 步进电机智能控制卡的基本构成和性能

步进电机智能控制卡是由 8031 单片机基本系统、步进电机控制接口和 PC 总线接口等所构成,如图 1 所示。8031 单片机基本系统由 8031CPU、32K~64KRAM、ROM、扩展 I/O 口和译码电路等所构成,是智能卡的核心,用于对整个智能卡进行独立控制,完成 PC 机的全部命令和要求。步进电机控制接口是直接产生步进电机控制所需要的工作脉冲信号和正反转控制信号。PC 总线接口是和 PC 机进行命令、状态传送的接口,使步进电机智能控制卡能直接插入 PC 机插槽。

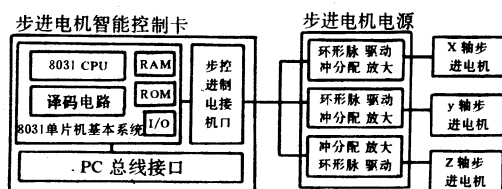


图1 步进电机智能控制卡的结构框图

步进电机智能控制卡的主要性能:

- 1) 三个独立的步进电机控制通道,可以同时或单独控制 X、Y、Z 三台步进电机。
- 2) 每个命令的步数为 1~65535 步。
- 3) 步进电机控制速度为 1~10000PPS。
- 4) 输出步进电机控制脉冲和正、反转信号、全部信号由光电耦合元件与步进电机电源相连接。
- 5) 外部有每一通道紧急停止按钮接口,用于手动紧急停止。
- 6) 对每一通道提供二种控制模式:连续模式和选择模式。连续控制模式是连续设定的各段加、减速、恒速等过程所组成的一条复杂工作曲线,而运行可以从初始设定的第一段一直连续工作到最后一段结束,这种模式用于较为复杂的多段加工过程,曲线情况如图 2(a)所示。选择模式是预先设定好一组工作曲线,根据不同加工情况从其中选择一条曲线进行加工。这种模式适合于简单加工和多种类的加工要求,每条曲线相对简单,由较少加工段组成,曲线情况如图 2(b)所示。

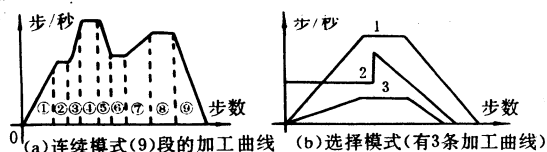


图2 二种控制模式

7) PC 机接口命令及驱动软件:PC 机使用的主要命令有模式设定、工作段方式设定(速度时间方式:初速、终速、总时间、旋转方向,速度步长方式:初速、终速、总步长、旋转方向)、运行、循环运行、停止、暂停、连续、清除、复位、等待等命令。

2 基本接口电路设计

基本接口电路是指步进电机控制接口和 PC 总线接口。

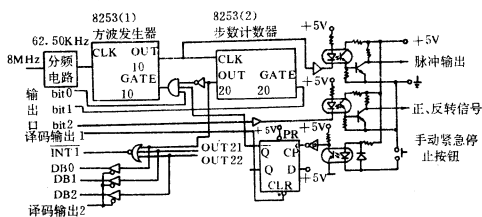


图3 步进电机控制接口

图3表示了步进电机控制接口的一个通道的逻辑图,该接口的主要功能是在8031CPU的控制下,产生控制步进电机转速的脉冲信号、正反转信号、并对外部手动紧急停车由硬件自动响应。

1)产生控制步进电机转速的脉冲信号方法

利用二片8253可编程间隔定时器,每片有三个16位定时/计数器,每一通道使用二只16位定时/计数器。一只作为方波发生器,对62.5KHz信号进行分频产生所要求频率的方波信号,此方波信号经驱动和光电耦合后输出,输出方波的频率对应着步进电机的转速,而方波数即是步进电机转动的步数。另一只16位定时/计数器作为步数计数器。在加、减速过程中,根据计算的分频系数和对应的步数写入这二只定时/计数器,在步数到后,由步数计数器的OUT20端输出高电平,经过非门和三与门后控制方波发生器的GATE10端,暂停产生方波,同时产生1个8031的外部中断信号和识别标志,在中断处理中可重新取出数据区中下一个分频系数和步数,重写这二只定时/计数器,设置后步数计数器的OUT20变低,打开GATE10,产生下一频率的方波,实现加、减速过程。

2)正反转信号的方法

控制正反转只需要一位的高低电平就可以了,所以每一通道只要利用8031基本系统的输出口中的一位,经过驱动、光电耦合后输出。

3)外部手动紧急停车的处理

每一通道外部有一手动按钮,一按此按钮该通道就停止脉冲输出,从而使步进电机停止。此按钮通过光电耦合产生一CP脉冲,使D触发器的Q端变低,通过三与门控制方波发生器的GATE10端,暂停方波输出。可用软件由译码器输出端1使该D触发器复位,Q端为高,重新打开GATE10。

另一个基本接口是与PC机总线的接口,如图4所示。

该接口的主要功能是8031系统和PC机之间的信息交流,PC机侧向8031侧传送命令和数据,并从接口读取工作状态,虽然接口的传送数据量少,但要求接口能快速响应,从而能立即实行暂停、停止、继续等命

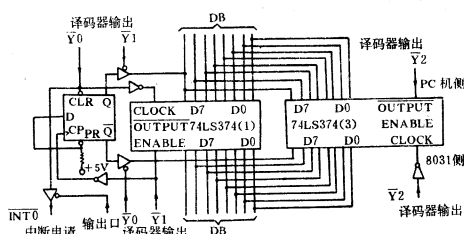


图4 8031系统和PC机的接口

令。所以在硬件设计上采用中断、设置交互查询标志、锁存数据等,再配合软件的查询数据传送方法。这种方法硬件所使用的芯片少,电路简单,实用性强。

图4中74LS374(1)是用于锁存PC机侧传送的命令和数据,374(2)用于锁存8031侧传送的工作状态标志。D触发器是用于设置交互查询标志的,其Q端是PC机侧查询是否可以传送数据的标志,Q为“1”时,表示可以传送,而Q端是8031侧查询是否可以读取数据的标志,Q为“1”时,表示可以读取。当PC机侧送入数据时,由PC机侧的Y0清Q置Q,而当8031侧读取数据时,由8031侧的Y1作为CP脉冲置Q清Q,从而使PC机侧可以再传送下一个数据。

①PC机侧的处理:PC机侧先由译码地址Y1有效,读取Q标志,如Q为“1”表示可以写入命令数据,然后Y0有效,此信号经非门后作为374(1)的数据锁存信号,锁存命令数据,同时清D触发器,使Q为“0”,Q为“1”,此信号还作为8031外部中断申请信号INT0,在传送命令数据后发送一结束符0FFH。PC机用Y2直接读取374(2)的状态数据。

②8031侧的处理:当8031响应INT0后,其译码地址Y0读取Q标志,如为“1”,表示可以读取送入的命令数据,再由Y1读取,同时使D触发器的Q为“1”,Q为“0”,允许PC机侧送下一字节命令数据,直至读到结束符返回。8031侧用Y2锁存状态标志在374(2)中,供PC机侧读取。

3 软件设计

步进电机智能控制卡的软件由二部分组成,一部分是PC机接口软件、另一部分是智能控制卡上8031的系统软件。

1)PC机接口软件:

PC机接口软件主要是为PC机传送命令数据的接口软件,采用函数或子程序调用的形式和接口软件连接。这接口软件的主要功能是:①传送命令数据、②命令数据合法性检查、③读取智能卡状态标志、④命令数据出错信息提示和状态标志分析提示等。

2)8031系统软件:

这部分软件相对复杂,除了8031本身有关的基本

软件外,主要完成以下控制有关的功能:

①读取 PC 机命令数据有关的外部中断处理功能。

②线性加、减速运行控制、外部按钮紧急停止处理的外部中断处理功能。

③命令数据处理、工作状态输出功能。

④线性加、减速过程的计算。

⑤随机动态工作数据的存储分配。

工作数据区的结构:工作数据是控制步进电机工作的预定曲线计算的数据,这些数据是根据不同通道、不同模式、不同曲线、不同曲线段、不同加、减速过程经计算得出的。由于不同用户、不同的工作情况,这些数据是一种随机、动态、不同数据量的许多组数据。为了能更加合理、有效地在有限的内存空间中存储这些数据,而且便于加减速过程中取出,能快速查找这些数据,实现线性加减速过程,采用了分级指针表方式的存储结构和分级工作指针进行快速查找取出数据的方法。图 5 表示了这种分级指针表方式的存储结构情况。

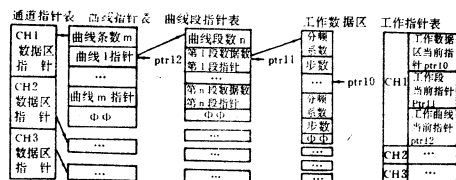


图 5 分级指针表数据区结构

线性加、减速过程的计算:

为了实现线性加、减速控制,必须计算出控制所需的分频系数和相对应的步数值。根据线性加、减速运动规律,有以下基本公式:

$$V_T = V_0 + aT \quad (1)$$

$$N = V_0 + \frac{1}{2}aT^2 \quad (2)$$

其中: V_0 为初始速度、 V_T 为终速度、 N 为从 V_0 到 V_T 的总步数、 T 为从 V_0 到 V_T 的总时间、 a 为线性加速度。

从(1)(2)式可以得出:

$$a = (V_T^2 - V_0^2) / (2N) \quad (3)$$

在从 V_0 到 V_T 的过程中在 t 时刻, S 步时,速度为 V ,则此速度 V 和 t 可以得到以下计算公式:

$$V = \sqrt{\frac{V_T^2 - V_0^2}{2N} \times S + V_0^2} \quad (4)$$

$$t = \frac{2N(V - V_0)}{V_T^2 - V_0^2} \quad (5)$$

$$S = \frac{N}{V_T^2 - V_0^2} (V^2 - V_0^2) \quad (6)$$

在从 V_0 开始,经过 m 阶速度的增(减)量后,达到

速度 V 时,实际所用的时间为 $\sum_{i=1}^m \frac{ni}{vi}$,实际走过的步

数为 $S_m = \sum_{i=1}^m ni$,而理论上应该用的时间为 $\frac{2Sm}{V_0 + V}$ 。由

于步长是步进电机精确定位所必须的,所以通常选择步长为目标函数,即保证总步长为设定的步数 N ,而总时间只能为一个近似值,但计算中可将总时间的误差控制在 0.00001 秒以下,这完全能满足工业控制的要求,

所以要求 $\min \left| \sum_{i=1}^m \frac{ni}{vi} - \frac{2Sm}{V_0 + V} \right| \leq 0.00001$ 秒。采用曲线拟合的方法,根据不同的总步长,先选取一个合适的预置初值,进行 V_i 的逐级计算,并通过迭代,修正 ni 后,得到符合要求的一组分频系数和步数。

4 结束语

采用单片微机的智能接口能使 PC 机方便地实现实时、并行、多功能的智能控制,是一般单功能控制接口卡的进一步发展。步进电机智能控制卡是这种接口中的一种,由于此卡采用了专用的定时/计数器芯片,使步进电机的加、减速过程变得很容易实现,又采用了与 PC 机的简单的并行接口,用户只要使用提供的命令进行简单编程就能实现所需的各种控制,而且在步进电机控制的同时,PC 机还可以进行其他的各种处理,实现并行工作。这种智能控制接口对于其他不同控制的接口设计有一定的实用参考价值。

由于智能卡上的 8031CPU 的处理速度和内存 RAM 容量的限制,所以对于复杂的加、减速过程的计算需要一定的时间,只能预先设定并计算完后,才能进行运行。对于计算数据量,还应根据设定曲线复杂程度和曲线条数等进一步智能化,以能自适应这些变化,控制一定的数据量,从而能为尽可能多的不同用户服务。

参考文献

- 何立民编著. 单片机应用系统设计. 北京航空航天大学出版社, 1990. 5
- 王士元 编著. IBM PC/XT(长城 0520)接口技术及其应用, 南开大学出版社, 1990. 4.
- 李伯成等编著. IBM PC 系列微机开发与应用, 西安电子科技大学出版社, 1992. 8

公告

经正式批准,《计算机应用研究》杂志已成为中国计算机学会会刊,从而本刊又登上新的台阶,成为中国一级学会会刊。

本刊全体办刊同仁暨各联合办刊单位,衷心感谢中国计算机学会赋予的殊荣,并继续矢志不移地办好刊物,为繁荣信息产业,为全中国和全人类做出更大贡献!

特此公告

《计算机应用研究》编辑部 1994 年 2 月

哥德巴赫猜想等式的个数 ——软件水平考试程序设计练习

广西科学院(南宁 530003)罗海鹏

摘要 全国软件水平考试下午试题是程序设计,题目的形式是:给出程序,但留出几个关键的地方,让考生填上。本文即是这样的一道 FORTRAN 程序设计试题,给有兴趣的读者练兵用。

关键词 哥德巴赫猜想 软件水平考试 FORTRAN 程序设计

任何大于 4 的偶数都可以表示为两个奇素数之和,例如: $6=3+3$, $8=3+5$, $10=5+5$,...,类似的等式可以很容易地这样一直列下去。这个看似简单的命题,若要严格地证明它,却是非常困难的,至今仍未解决。由于哥德巴赫在 1742 年首先提出这个命题的等价命题,因此,人们把这个未解决的问题称作哥德巴赫猜想。

实际上,对于一个确定的偶数,常常可以列出不只一个的不同的满足要求的等式,例如: $10=3+7$, $10=5+5$, $22=3+19$, $22=5+17$, $22=11+11$ 。对于偶数 1994 来说,存在着 27 个不同的等式: $1994=7+1987$, $1994=43+1951$,..., $1994=997+997$ 。随着偶数的逐渐增大,等式的个数忽多忽少,但总的趋势也是逐渐增多的。

下面的 FORTRAN 程序,给出了 1994 可以表示为两个奇素数之和的所有的等式。这个程序把 1994 换成任何其他的偶数,则可以求出这个偶数可以表示为两个素数之和的所有等式。请您仔细阅读程序,填出其中的①、②、③、④、⑤应该换成 FORTRAN 哪些字符,这个程序才能正确运行。

程序清单

```
PROGRAM PARAMETER
```

```
M=1994
```

```
DO 20 L=2,M/4+2
```

```
MP1=L*2-1
```

```
CALL PRIME(MP1,NB)
```

```
IF①GOTO 20
```

②

```
CALL PRIME(MP2,NB)
```

```
IF① GOTO 20
```

③

```
WRITE(*,10) N,'')'M,'=' ,MP1,'+',MP2
```

```
10 FORMAT(1X,I2,A2,I4,A,I3,A,I4)
```

```
20 CONTINUE
```

```
STOP
```

```
END
```

```
SUBROUTINE PRIME(MP,NB)
```

```
NB=1
```

```
MT=SQRT(FLOAT(MP))+0.5
```

```
DO 30 I=3,MT,2
```

```
IF(NB.EQ.0)GOTO 40
```

④

```
IF(MU*1.NE.MP)GOTO 30
```

⑤

```
30 CONTINUE
```

```
40 RETURN
```

```
END
```

(答案请在本期内找)

输出结果

1) $1994=7+1987$

2) $1994=43+1951$

3) $1994=61+1933$

4) $1994=127+1867$

5) $1994=163+1831$

6) $1994=193+1801$

7) $1994=211+1783$

8) $1994=241+1753$

9) $1994=271+1723$

10) $1994=331+1663$

11) $1994=337+1657$

12) $1994=367+1627$

13) $1994=373+1621$

14) $1994=397+1597$

15) $1994=463+1531$

16) $1994=523+1471$

17) $1994=541+1453$

18) $1994=547+1447$

19) $1994=571+1423$

20) $1994=613+1381$

21) $1994=673+1321$

22) $1994=691+1303$

23) $1994=757+1237$

24) $1994=823+1171$

25) $1994=877+1117$

26) $1994=907+1087$

27) $1994=997+997$

GAL 器件在智能仪表设计中的应用

安徽机电学院电气系 (芜湖 241000) 周鸣争

摘要 本文针对 GAL 器件的结构特性和智能仪表设计的特点。介绍了几种利用 GAL 器件实现仪表设计加密的方法。

关键词 GAL 智能仪表 加密 硬件设计

1 引言

众所周知,在智能仪表的硬件设计中,大都采用标准的组合电路,通用集成芯片和专用 ASIC 芯片来实现系统功能。近年来,不少厂家和单位开发了很有市场前景、具有较强竞争力的智能仪器仪表,由于在设计中大都采用标准器件和集成芯片,使得这些产品一旦问世,就会立即被别人仿制,保密性很差。在目前有关法规不太健全的情况下,对仪表实行加密是一种保障设计人员合法权益的有效方法。但要实现系统硬件级上的加密,通常是一种比较麻烦的事情。这是由于常用的集成芯片其功能,引脚及内部结构都已由生产厂家硬连接定型,设计人员无法改变,使得这些器件很容易被识别。笔者在实践中感到,GAL 内部虽只含有门电路和触发器,但它们并非硬连接,可由用户通过编程来实现软连接。若我们在仪表设计时,利用这种器件内部矩阵阵列的熔丝,将其内部的门电路按要求以一定的逻辑关系连接起来,来代替一些标准器件的功能或对仪表中 CPU 的数据线、地址线重新定义后实现重新连接,而实现这种变换之间的逻辑关系用 GAL 器件来实现这样 GAL 器件好象一个“黑匣子”。使得仿制者无从下手,以达到系统硬件保密的目的。

2 GAL 器件工作原理

GAL 器件是 1985 年由美国德州仪器公司推出的新器件,是第二代 PAL,它的阵列与 PAL 相似。它与 PAL 器件的主要不同点在于输出可编程和可设置永久保密位。

GAL16V8 为最常见的一种 GAL 器件。其外部结构如图 1 所示。内部点阵输出采用宏单元法,允许用户定义每个输出的结构和功能。输出逻辑宏单元中包含

有 8 个输入或非门,极性选择异或门,D 触发器,四个多路开关,三套输出缓冲器。内部共有 36 个唯一地址可供用户使用。其中 0~31 行,每行包含 64 位输入项数据。可编程码点为 $32 \times 64 = 2048$ 个,这是用户阵列,用户可根据设计要求,制定所需的逻辑功能,在开发系统的支持下,将逻辑功能写入 GAL 器件。第 32 行是电子标签行,包含 64 位供用户编程。其程序器可保存用户所定义的数据,这个标签不受保密位的控制。60 行为一 82 位结构控制字其定义为:

32位 与项无效	4位 XOR(n)	1位 SYN	8位 AC1(n)	1位 ACO	4位 XOR(n)	32位 与项无效
12 13 14 15	12 13 14 15	12 13 14 15 16 17 18 19	16 17 18 19			

此行包括输出结构和输出极性信息。这个 82 位结构控制字用来设计所需用途的器件。61 行为保密单元,仅有一位,一旦该位编程完毕,所设计的逻辑阵列永远不会被读写,这可防止对此阵列结构的非法复制但标签读出受影响。第 63 行为整体擦除,擦除时将电子标签和保密单元也擦除。GAL 器件回到初始状态。

GAL16V8 共有 8 个宏单元。其管脚编号为 n,每一宏单元由结构控制字中的四位 SYN、AC0、AC1、XOR 来规定宏单元所具有的功能。其关系为:

SYN	AC0	AC1	XOR	功能
1	0	1		输入模式
1	0	0	0/1	输入逻辑组合输出
1	1	1	0/1	有反馈组合输出
0	1	1	0/1	钟控有反馈的组合输出
0	1	0	0/1	钟控选通有反馈的组合输出

其中 XOR=1,表示输出高电平有效。XOR=0,为输出低电平有效。

3 加密方法

综上所述,GAL 器件的加密对用户来说是透明的,即用户不必知道其内部的加密机制,只需利用 GAL 编程器就能完成加密操作。这是因为 GAL 器件内部有一加密位,一旦这位编程,就绝对读不出移位的原始配置。利用 GAL 器件的这种特性,可实现系统中的硬件加密。其设计步骤为:1)按要求确定逻辑关系。2)根据逻辑关系,生成编程软件的输入文件

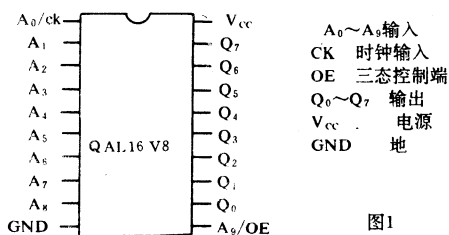


图1

(GAL. PLD)。3) 编译或汇编该输入文件生成内部编程器熔断 (GAL. JED)。4) 将 JED 文件写入 GAL 器件并加密。

一般来说, 在一个智能仪表的结构中, 信息的流动不外乎通过数据总线、地址总线和控制总线来完成, 硬件通常也是通过各种组合逻辑或时序逻辑电路来实现连接。因此, 利用 GAL 器件进行系统硬件加密可有以下几种常用的方法。

① 硬件代替法。该方法主要通过对 GAL 的编程, 用 GAL 器件实现某一特定的功能, 来代替硬件中用常规标准器件所实现的逻辑电路。以实现对硬件结构的加密, 防止对仪表硬件的仿制。

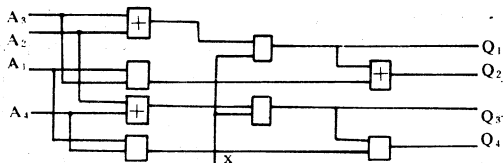


图2 逻辑电路图

如在一仪表中, 有一用图2所示的逻辑电路, 来实现系统硬件之间的连接。则我们可用 GAL16V8 来实现这逻辑关系。代替用常规门电路来设计的方法。其编程源程序见程序一。

* 程序一

PLD16V8

A₁, A₂, A₃, A₄, NO

X, NO, NO, NO, GND

Q₁, Q₂, Q₃, Q₄, NO

EQUATIONS

NO, NO, NO, NO, VCC

$Q_1 = A_2 * X + A_3 * X$

$Q_2 = A_2 * X + A_3 * X + A_4 * A_1$

$Q_3 = A_2 * X + A_4 * X$

$Q_4 = A_1 * X + A_4 * X + A_2 * X + A_4 * X$

END CHAN-JIAN-MING

② 连线错位法。该方法主要是针对带有微处理器 (MPU) 智能仪表中, MPU 的地址线进行错位连接或进行逻辑运算错位, 使得 MPU 的逻辑地址与存储器的物理地址之间形成相应的地址映射表。而这种映射关系由 GAL 来实现。该方法既可实现对硬件结构的加密, 又可对 EPROM 中的系统程序实现加密。但在应用这种方法时, 应注意逻辑地址与物理地址之间的对应关系。避免产生存储器单元地址的重叠和无定义映射。

如某一 CPU 的地址线为 16 根 A₀~A₁₅, 其逻辑地址与物理地址之间映射关系如图 3(a) 所示。

这种对应映射关系可用 GAL16V8 来实现, 其编程源程序如程序二所示。硬件连接图如图 2(b) 所示。

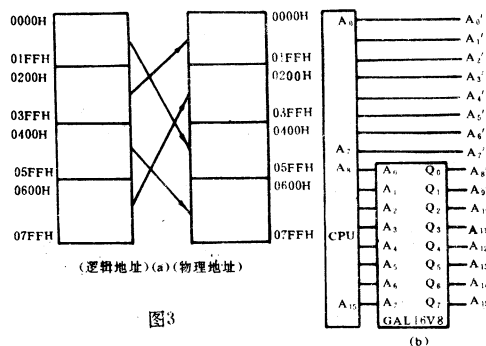


图3

* 程序二

PLD16V8

A₀, A₁, A₂, A₃, A₄, A₅, A₆, A₇, NO, NO

Q₀, Q₁, Q₂, Q₃, Q₄, Q₅, Q₆, Q₇

EQUATIONS

$Q_7 = A_7$; $Q_6 = A_6$; $Q_5 = A_5$; $Q_4 = A_4$

$Q_3 = A_3$; $Q_2 = !A_1$; $Q_1 = A_2$; $Q_0 = A_0$

END CHAN-JIAN-MING

③ 数据变换法。该方法主要用于对系统中 CPU 的数据线进行重新定义和连接, 使得 EPROM 中的数据或指令代码具有与本身不同的含义, 使得反汇编后的源程序与原程序不同, 防止应用软件的复制, 实现对其系统软件的加密。

图4就是利用该方法对数据线为八位的 CPU 进行行数据变换的连接图。其 GAL 编程源程序如程序三所示; 但在应用该方法时, 要求在对 EPROM 进行固化时, 数据就必须进行这种变换。

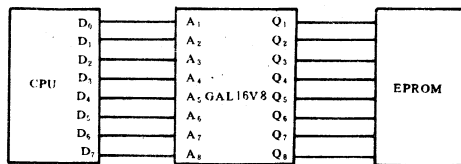


图4

否则, 使数据无法还原, 系统不能工作。

* 程序三

PLD16V8

A₁, A₂, A₃, A₄, A₅, A₆, A₇, A₈

Q₁, Q₂, Q₃, Q₄, Q₅, Q₆, Q₇, Q₈

EQUATION

$Q_1 = A_3$; $Q_2 = A_5$; $Q_3 = A_7$; $Q_4 = A_1$

$Q_5 = A_8$; $Q_6 = A_2$; $Q_7 = A_4$; $Q_8 = A_6$

END CHAN-JIAN-MING

4 结束语

利用 GAL 器件对智能仪表实现硬件、软件加密是完全可行的。笔者在多个仪表设计中, 通过使用上述所介绍的方法, 证明是行之有效的。在智能仪表的设计中, 若利用 GAL 的时钟端, 构成 GAL 时序工作方式或综合利用上述几种加密方法, 将会取到更好的加密效果。

电子广告显示屏

中国科学院近代物理研究所 (甘肃兰州 730000) 王水清

摘要 本文介绍了一种商业化的电子广告显示屏,采用 8031 单片机系统,用发光二极管组成点阵显示屏,对制做电子显示屏具有一定的参考价值。

关键词 电子广告 显示屏 单片机

现在市场上使用的电子广告显示屏越来越普遍,商店、车站、机场用来做商品广告,车次和航班状况,证券市场用来显示股票行情。本文介绍的是商店中使用最多的 16×16 点阵字形的广告屏,它可以显示英文、数字和汉字信息。根据这种方案可以做到更大的规模。如果采用 8031CPU,主频 12MHz,最大分辨率可做到 192×96 点阵,分辨率再高就会因发送数据时间延长,而使整个屏幕出现闪烁。

1 主板结构

见图 1 所示(略) CPU 采用 8031,主频可选晶振 6MHz 或 12MHz。程序存储器用 4 片 16KB 的 27128 EPROM;其中地址 0000H—3FFFH 内储存程序部分,地址 4000H—FFFFH 储存汉字、图形库。1 片 32KB RAM 的 62256 作为数据存储器,储存被显示的内容。1 片 74LS139 作地址译码。外部 I/O 接口和外部 RAM 统一编址, RAM 储存区的地址为 0000H—7FFFH,扩展一片 8155 并行接口芯片作显示数据的输出和显示板点亮驱动信号源,因点亮显示板的功率较大,用了 8 只晶体三级管作功率驱动。本文针对显示板为 8 个 16×16 点阵汉字显示板的功率需求设计的,如果显示板面更大,可采用达林顿管驱动。

8155 片 A 口的 8 只反相器的作用是:标准字模点阵中凡是 1 点亮,凡是 0 为暗点,加上反相器则显示点阵与字模点阵一致。

2 显示板结构

见图 2 所示(略)这是一个 16×16 点阵的显示板,整个显示屏由同样的 8 块显示板拼接而成,上下分别由 8 位数据线 AD0—AD7 和 BD0—BD7 在板与板之间串行连接,一级一级送下去。AD0—AD7 与 BD0—BD7 在主板上并联。数据打入脉冲 PCA 和 PCB 从 8031 的 P1.0 和 P1.1 发出。由于上下数据线共用,当打入 PCA 脉冲时,上半部的数据锁存器 74LS374 接收到数据 AD,反之当 PCB 脉冲有效时,下半部数据锁存器接收到数据 BD。注意:这里的关键是,因数据传输是通过 74LS374 一级一级传下去的(由右向左),如果 PCA 或 PCB 信号同时向各显示单元发出时钟信号,则各单元上的数据无法保证正确无误,各级锁存器的数据锁存必须有个时间差。只有当左边的一个锁存器将数据锁存之后,它前一个(右边的)锁存器才能允许

数据更新。各级锁存器之间的数据更新的时间差是依靠在相邻两个锁存器的时钟端插入两个反相器或一个同相驱动器来延时的。从主板上送来的时钟脉冲 PCA 和 PCB 首先送到最后(最左边)一块显示板的左端 (PCA—IN 或 PCB—IN),这样时钟脉冲是反向传送的。

显示板上的发光二极管每个纵列点亮由一个三级管驱动。8 块显示板的点亮信号是并行连接,一次点亮两列,发光二极管组成四个方阵,每次的数据传送和纵列点亮都以小方阵为单位。

3 软件设计

标准汉字库的汉字字形点阵如图 3,在显示流动的文字信息时,大多数情况下是文字由显示屏右边进入,向左方流动并逐步移出显示屏,标准字模点阵在这种显示方式下存储有诸多不便。采用 MASM V5.0 汇编语言编制了 GZK。ASM 程序对标准字库进行了修改。程序中计算方法如下:

将第 1、3、5、7、9、11、13、15 字节的 2^7 位组成新字模的第 1 字节,其中第 1 字节的 2^7 位变为 2^0 位,第 15 字节的 2^7 位仍为 2^7 位。第 17、19、21、23、25、27、29、31 字节的 2^7 位组成新字模的第 2 字节,其中第 17 字节的 2^7 位变为 2^0 位,第 31 字节的 2^7 位仍为 2^7 位,依次类推,第 18、20、22、24、26、28、30、32 字节的 2^0 位组成新字模的第 32 字节。如此组成的新字模形式如图 4。

第 1 字节	第 2 字节
第 3 字节	第 4 字节
$2^7 \dots 2^0$	$2^7 \dots 2^0$
...	...
第 31 字节	第 32 字节

图 3

第 1 字节	第 3 字节	2^0		第 31 字节
		.	.	
		.	.	
		.	.	
		2^7		
第 2 字节	第 4 字节	2^0		第 32 字节
		.	.	
		.	.	
		.	.	
		2^7		

图 4

VGA256 种颜色的显示

安徽大学计算机系 (合肥 230039) 李龙澍

摘要 本文介绍了 VGA 显示器 256 种颜色的显示方法,给出了用 Turbo C 语言编写的精巧的显示程序。

关键词 彩色显示 VGA Turbo C

VGA 显示器是 286 及兼容机上一一种较普及的显示器,它最多提供 256 种颜色。彩色图像是调用操作系统的中断实现的,BIOS 的 10H 号中断 0 号功能是设置屏幕显示模式的,对于 VGA 显示器,13H 模式是 200×320 点阵的彩色图形模式,共有 256 种颜色,其图形映象信息位于 A0000H 开始的视频 RAM 中,每个字节代表一个图像点,共 64000 个字节,字节的取值为 0~255,每个数值代表一种颜色,共 256 种颜色。

人们在编写绘制彩色图像程序时,很难记住 256 种颜色,每种颜色也很难用语言给予描述,下面的 C 语言程序 Color256.C 可以在 VGA 显示器上显示所有 256 种颜色,每种颜色为一个矩形块,分 8 行显示,每行 32 块,彩色块大小一致,256 块恰好占满整个显示器。每种颜色对应的值为:Line * 32 + col (0 ≤ line ≤ 7, 0 ≤ col ≤ 31), line 代表行号, col 代表列号, * 是乘号。

用户可以用 Turbo C 将 Color256.C 编译成执行文件 Color256.EXE。生成的执行文件与 DOS 外部命令文件一样。用户编写程序需要查阅颜色时,随时都可以在 DOS 提示符下键入 Color256。

源程序清单如下:

```
#include<stdio.h>
#include<dos.h>
#include<conio.h>

unsigned char far *buffer=MK_FP(0xa000,0x0000);
/* buffer 开始指向图形映象位置首址 */
Void main(Void)
{
    int color,high,wide,col,line;
    _AX=0x0013;
    geninterrupt(0x10);/* 设置 13H 模式 */
    for(line=0;line<8;line++)
        for(high=0;high<25;high++)
            {color=line*32;
              for(col=0;col<32;col++)
                  {for(wide=0;Wide<10;wide++)
                      *buffer++=color;
                    color++;
                  }
            }
    getch();
    _AX=0x0003;
    geninterrupt(0x10);/* 回到文本模式 */
};
```

收稿日期:1993 年 12 月 18 日

将需显示的汉字点阵字库经以上方法加工后,制成文本文件并存入首地址为 4000H 的 ROM 区。

运行软件在开机启动后,查找到第一汉字的一个字节,并将这个字节的数据送到 AD 数据线,字节地址加 1 后,把第二个字节的数据送到 BD 数据线。之后字节地址加 15,将第 17 个字节的数据送到 AD 线,18 字节送到 BD 线,如此 16 次循环后将前 8 个汉字的两列数据传输完毕,这时发出列点亮信号,延时 2MS 后,关闭列点亮信号。这时字节地址为 242,该地址减去 239 后地址为 3,将 3 作为首地址重复以上工作,但点亮第 2 列,如此下去我们看到一幅 8 个汉字出现在显示屏上了。当 8 个汉字显示完成后,分别将 3、5、7、...,每次加 2 的奇数作为前地址重新开始,于是我们看到一系列的汉字开始在显示屏上由右向左地流

动。

至于向其它方向的流动,在运行软件中要根据具体情况调整字模点阵的显示次序,如向上流动。在显示一幅 8 个汉字的完整画面之后,将整幅 8 个汉字的全部奇数字节的 2° 位屏蔽掉,字节右移 1 位后,将偶数字节的 2° 位并入作为 2' 位。然后将偶数字节同上处理,并将下一幅 8 个汉字的全部奇数字节的 2° 位并入作为 2' 位。显示这样处理后的一幅画面,就会看到画面向上移动了一行点阵,连续下去,则一幅接一幅的画面由下向上源源不断地滚动起来。

按照这样的思路我们可以做出各种花样的显示动画来。

注:需硬件电原理图者,可寄 10 元工本费向本刊邮购。

防止 MIS 意外退出系统的方法及实现

山东工业大学计算机系 (济南 250012) 蔡青华
 山东工业大学计算中心 (济南 250012) 胡毅

摘要 本文针对 FOXBASE、DBASE 等数据库语言在开发 MIS 时的不足之处, 提出一个完善其功能的方法, 并给出实用程序, 供开发者参考。

关键词 MIS 中途退出 用户系统 BIOS 中断 功能码

1 问题的由来

随着计算机技术的发展, 计算机在信息管理方面的应用日益普及。FOXBASE、DBASE 等数据库语言的出现, 无疑为用户方便地开发 MIS (管理信息系统) 提供了良好的工具。但由于这些语言本身功能所限, 在系统执行过程中, 无法对可能出现的意外情况做妥善处理, 致使用户系统被迫中途退出。若继续运行则需重新进入系统, 有时会使前面的操作和处理前功尽弃。

在 MIS 中, 造成系统意外退出的原因有多种。其中常见的是由于外部设备状态不明而引发的。而 FOXBASE 之类数据库语言本身对外设的状态无法全面检测。为此, 笔者对引起 MIS 意外退出的因素进行整理归纳, 提出一个统一解决的方法。

2 引起意外退出的几种因素

2.1 打印机

输出打印是 MIS 的基本功能之一, 为保证系统正常运行, 需要对打印机以下状态进行检测。

- (1) 打印机的电源是否接通;
- (2) 打印机是否在联机状态 (即是否准备好);
- (3) 打印纸是否安装好;
- (4) 打印机的 I/O 口是否错 (如连线是否接好等)。

2.2 软盘驱动器

数据的获取和存储都不可避免地要与磁盘打交道, 所以使用前也应了解以下状态。

- (1) 驱动器中是否装入软盘;
- (2) 装入的软盘是否为写保护状态;
- (3) 磁盘中的剩余空间是否够用。

以上几点为用户系统访问外设时经常遇到的情况, 如运行中这些情况不能满足用户系统要求的状态, 就会出现中途退出用户系统, 并且 DOS 的错误提示信息在屏幕上的光标处显示出来。DOS 的错误提示信息一旦出现在屏幕上, 虽然按相应的提示键可以使程序继续运行, 但在未清屏前屏幕上始终保留着错误提示信息。这样, 不仅扰乱了正常的屏幕显示, 而且 DOS 的错误提示信息简单而不明确, 易造成误操作。

3 解决的方法

3.1 系统提示、用户检测

这是一种目前常用的方法。开发系统时, 在程序中

适当的地方 (如启动打印机或磁盘之前) 给出各种提示信息提示用户, 届时由用户来检测外部设备的状态, 保证系统正常运行。但是, 如果用户无视这些提示信息或进行了误操作, 未能使外设达到系统运行所要求状态, 则仍然会引起中途退出用户系统 (见图 1.a)。另外, 还有些状态是人工检测不到的, 如磁盘的剩余空间等。所以, 这种方法对用户的要求高, 并且不能从根本上解决问题。

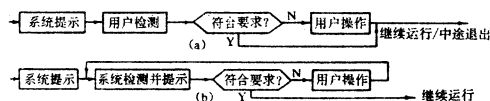


图 1 两种提示与检测的方法

3.2 系统提示、系统检测

提示可用上述方法完成, 检测则是通过用户系统中的测试程序来完成。用户根据系统提示完成操作后, 仍要通过系统检测, 如用户无视提示信息或进行了误操作, 则检测通不过, 仍给出提示信息, 直到外设达到系统运行所要求的状态 (见图 1.b)。测试程序可由下列方法实现。

(1) 打印机的状态测试有两种

①利用 FOXBASE 中的应用函数 SYS()。此函数 SYS(16)用于测试打印机的状态。若打印机尚未待命, 传回“OFFLINE”信息, 其返回值为 F(假); 若已准备好 (打印机处于 ON-LINE 状态), 则传回“READY”信息, 其返回值为 T(真)。这种方法简便易行, 但测试的内容不够全面。

②用 BIOS 中断 17H 打印机控制服务功能。将功能码 02H (打印机状态请求) 装入 AH 寄存器中, 将要测试的打印机号装入 AX 寄存器中, 执行 INT 17H

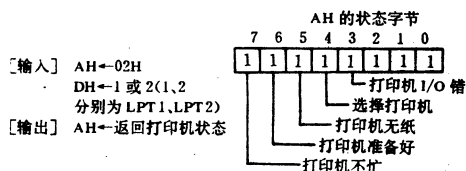


图 2 检测打印机状态

后, 打印机的状态返回值在 AH 寄存器中。若 AH 的第

六位置 1, 为打印机准备好, 若为 0 则为未准备好; 若 AH 的第五位置 1, 为打印机无纸, 若为 0 则为纸已准备好; 若 AH 的第三位置 1, 为打印机 I/O 口有错 (见图 2)。这种方法可全面了解打印机的当前状态, 根据提示, 由用户做相应处理。

(2) 软盘驱动器的状态, 在 FOXBASE 中无法直接测试, 只有通过中断功能来解决。用 BIOS 中断 13H 软盘服务功能, 将功能 04H 装入 AH 寄存器中, 将了解的驱动器号码值装入 AL 寄存器中, 然后执行 INT 13H。盘片没有装入驱动器时, 在 AH 寄存器中返回 6XH; 贴着写保护胶片时, 在 AH 寄存器中返回 1XH (6XH、1XH 中的 X 值不定)。

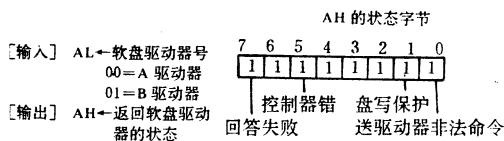


图 3 检测磁盘及驱动器的状态

计算磁盘的剩余空间是利用功能调用 21H, 将功能号 36H (读取磁盘剩余空间) 装入 AH 寄存器, 将要读取的磁盘驱动器号装入 DL 寄存器, 然后执行 INT 21H, 执行后的结果可在 AX、BX、CX、DX 寄存器中得到。

[输入] AH ← 功能调用号 (36H) DL ← 驱动器号
00 = 当前驱动器号 01 = A 驱动器号 02 = B 驱动器号

[输出] AX ← 一簇的扇区数 BX ← 可用的簇数
CX ← 一扇区的字节数 DX ← 一个驱动器的簇数

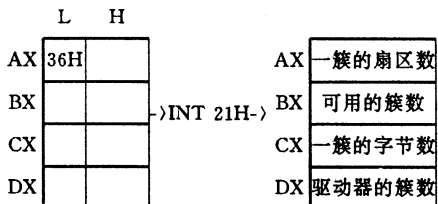


图 4 计算磁盘剩余空间

磁盘的全部容量 = 一簇的扇区数 * 一扇区的字节数 * 一个驱动器的簇数 = AX * CX * DX *

磁盘剩余空间量 = 一簇的扇区的字节数 * 一扇区的字节数 * 可用的簇数 = AX * CX * BX

如果指定了无效的驱动器号, 则会将 FFFFH 返回到 AX 中。

4 实现

在上述提及的方法中大多数为调用中断的形式, FOXBASE 本身无法完成, 但 FOXBASE 提供了调用其它高级语言的命令。对于外部可执行文件可以直接用 RUN 命令启动运行, 而对于汇编语言的程序模块经 EXE2BIN 转化为二进制代码文件后, 可用 LOAD、CALL、RELEASE 三个命令对其装入、运行、释放。在

此, 我们就软盘及软盘驱动器的测试为例, 用 C 语言来实现上述功能。

```
#include<dos.h>
int fd_check();
fd_check(fd)
unsigned fd;
{
union REGS inregs,outregs;
int intno,status;
intno=0x13;
inregs.h.ah=0x04;
inregs.h.al=fd;
int86(intno,&inregs,&outregs);
switch (outregs.h.ah & 0xf0)
{
case 0x80:
printf ("\n 软盘未准备好");
break;
default:
if ((outregs.h.ah | 0xfd) == 0xff)
{
printf ("\n 请去掉软盘的写保护");
}
else
{
printf ("\n 软盘已准备好, 进行数据操作");
}
break;
}
return()
}
main()
{
unsigned a,ss;
int n;
/* 对 A 盘进行测试 */
a=fd_check(0x02);
}
```

5 结束语

在对 MIS 多年的编程实践与应用中, 针对 FOXBASE 本身不易解决的问题, 用 TURBO C 编制了一些辅助程序, 取得了比较满意的效果, 也使 MIS 更加完善, 提高了用户系统本身的可靠性。

参考文献

- 戴水贵编著. DOS、BIOS 功能调用手册. 陕西电子杂志社
- 刘赢译. 汉字 Foxbase+ 2.10 使用手册. 中国科学院希望高级电脑技术公司

利用 EPROM 编程卡固化 51 单片机程序

太原市国营汾西机器厂研究所 (030027) 王晋远

目前,单片机在我国各行各业得到了广泛的应用,尤其是 MCS-51 单片机应用最为普遍。而一般的 MCS-51 单片机仿真系统所带的 EPROM 写入卡同 PC 机通用的 EPROM 写入卡相比来说,仿真系统所带的 EPROM 写入卡写入速度低、可靠性不高、固化 EPROM 的种类有限,为了克服上述缺点,笔者编制一个软件利用 EPROM 编程卡固化 51 单片机程序。

通用的 EPROM 编程卡只能固化以二进制存盘的文件,不能固化由 MCS-51 仿真系统编译软件所产生的目标文件,这是由于编译软件所产生的目标文件是以文本方式存盘的,这样,就为利用 EPROM 编程卡固化 51 单片机的目标文件带来困难。为此,笔者编制了一个程序,把以文本方式存盘的目标文件转换成以二进制方式存盘的文件,这样,就可以利用 EPROM 编程卡固化。在这里笔者首先采用的是以 BASIC 语言编制的程序(见程序 1),在使用当中发现运行速度慢,编译后运行速度也不理想,因此,笔者又以 Turbo C V2.0 语言改编了该程序(见程序 2),从而使运行速度大幅度提高,读者不妨一试。

另外需要说明的是:51 单片机编译软件形成的目标文件中:1—4 字节是文件的开始地址,5—8 字节是文件的结束地址,其后才是真正的文件部分。对于固化程序来说,这部分不起作用,所以,在本程序中不进行转换。

程序 1、程序 2 已在 AST P286、LX-286、LX-386、COMPAQ 386 机器上运行通过,笔者已使用三年多,效果良好,特推荐给读者,供读者参考。

程序 1:

```
100 PRINT "INTEL MSC51 File. OBJ(ASCII) TO File.
OBJ(BINARY)":PRINT
110 INPUT "INPUT FILENAME=",A$
120 INPUT "OUTPUT FILENAME=",B$
130 PRINT:PRINT:PRINT "Please wait....."
140 OPEN A$ AS #1 LEN=2
150 OPEN B$ AS #2 LEN=1
160 FIELD #1,2 AS N$
170 FIELD #2,1 AS M$
180 QQ=INT(LOF(1)/2)
190 I=5
200 GET #1,I
210 X$=MID$(N$,1,1):Y$=MID$(N$,2,1)
220 C=VAL(X$):D=VAL(Y$)
230 IF X$="A" THEN C=10
240 IF X$="B" THEN C=11
```

```
250 IF X$="C" THEN C=12
260 IF X$="D" THEN C=13
270 IF X$="E" THEN C=14
280 IF X$="F" THEN C=15
290 IF Y$="A" THEN C=10
300 IF Y$="B" THEN C=11
310 IF Y$="C" THEN C=12
320 IF Y$="D" THEN C=13
330 IF Y$="E" THEN C=14
340 IF Y$="F" THEN C=15
350 ZZ=C*16+D
360 LSET M$=CHR$(ZZ)
370 PUT #2,I-4
380 I=I+1
390 IF I>QQ THEN CLOSE #1:CLOSE #2:
PRINT "The end";SYSTEM ELSE GOTO 200
```

程序 2:

```
#include<stdio.h>
#include<conio.h>
#include<dos.h>
main()
{
char namein[14],nameout[14]
FILE *filein,*fileout;
int k,kk,bitnum[2];
long unsigned int filebit;
char chl;
struct date today;
struct time now;
textmode(c80);
textbackground(BLUE);
clrscr();
kk=6;k=25;
while(k!=3){
sleep(0.5);
clrscr();
textcolor(kk);
gotoxy(2,k);
cprintf("MCS-51 ASCII(*.OBJ) FILE TO BINARY
FILE Version 2.0 Copyriht(c) 1991.1992,1993\n");
gotoxy(13,K+2);
cprintf("BY WANG JIN YUAN 1993-7-18 884
PLANT TAIYUAN\n");
K=k-1;
KK++
}
gotoxy(10,11);
cprintf("Please input souce file name:");
gets(namein);
(下转 51 页)
```


激光印字机的文本打印程序

国防科技大学二系(湖南长沙 410073) 陈德明

激光印字机的打印效果虽然很好,但用 DOS 命令却难以得到理想的效果,表现在:(1)打印出来的样张左空太小;若纸张大小小于面板设定大小,则会出现右边漏字,底下漏行且无底空的现象。(2)只有送往印字机的内容满 1 页后才输出,对于多页的文件,其最后一页要等到下次打印满一页时才被输出,且内容混在同一页面。为克服这个不足,充分利用印字机的优越性,可以根据打印命令编一个打印程序,达到好的效果。通常的印字机的复位指令为 CHR\$(27)+"E",发送它,可使当前页(不论是否满)被打印,指令 CHR\$(27)+"&1#L"可将左边界置为第 # 列,CHR\$(27)+"1#F"可将页长设为 # 行,每页满这么多行后即换页,其中 # 为一数字串。笔者用 TURBO C 2.0 编了一个打印程序 HPTYPE.C,将它编译连接成一执行文件 HPTYPE.EXE 后,可以用来方便地打印文本文件。命令格式为:HPTYPE filename[length][width]

页长 length 和行长 width 为可选参数。本程序对 HP 系列印字机运行通过。

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
main(int argc,char *argv[])
{
    int width=70,m=0,length,c;
    char s[10],ss[5];
    FILE *fin,*fou;
    if(argc<2 || argc>4){
        puts (" * * * * *");
        puts (" * HP Laser Jet Hard Copy *");
        puts (" * by Chen Deming *");
        puts (" * * * * *");
        puts (" Usage: HPTYPE filename [length]
[length]");
        puts(" Default: length=60,width=70");
        exit(1);}
    strcpy(ss,"60");
    if(argc>=3) strcpy(ss,argv[2]);
    if(argc==4) width=atoi(argv[3]);
    length=atoi(ss);
    printf("page length =%3d, width =%3d\n",
```

```
length,width);
    fin=fopen(argv[1],"rb");
    if(! fin || length>100 || width>80){
        printf("\7 File not found or parameter invalid ! \n");
        exit(1);
    }
    fou=fopen("PRN","w");
    if(! fou){printf("Laser Jet is not ready ! \n");
        exit(1);
    }
    strcpy(s,"&1");
    strcat(s,ss);
    strcat(s,"F");
    fputc(27,fou);fputc('E',fou);
    fputc(27,fou);fputs("&a7L",fou);
    fputc(27,fou);fputs(s,fou);
    c=getc(fin);
    while(! feof(fin)){
        fputc(c,fou);
        m=m+1;
        if(c==10) m=0;
        c=getc(fin);
        if(m>=width&&c!=13&&c!=10){
            fputc(13,fou);fputc(10,fou);m=0;
        }
        fclose(fin);
        fclose(fou);
        fou=fopen("PRN","w");
        fputc(27,fou);fputc('E',fou);
        fclose(fou);
        printf("printing Completed!");
    }
}
```

《哥德巴赫猜想等式的个数》参考答案

- ①(NB.EQ.0)
- ②MP2=M-MP1
- ③N=N+1
- ④MU=MP/I
- ⑤NB=0

如何用多功能超卡代替串并卡

浙江水产学院(宁波 315010) 叶富乐

摘要 本文介绍对多功能超卡在磁盘选中时使它的数据线为高阻状态,从而解决了超卡与原机的磁盘专用卡在磁盘工作时数据冲突的矛盾,达到用户使用超卡中的串并口的目的。

IBM PC/XT 及其兼容机或配有 5 吋硬盘的一些 286 微机,它们都有独立的串并卡,当这些串并卡损坏时,由于是早期产品,故现在许多城市都很难买到这些单独的串并卡,使得配有这些卡的微机无法打印输出或进行通讯传送。而目前市场上一些电脑商店都有多功能超卡出售(卡内包括一个 3 吋硬盘接口,一个软驱接口,二个串口和一个并口及一个游戏口),并且价格也很便宜。能否用这种超卡来代替损坏的串并卡呢?回答是肯定的,因为这种卡包括一个并口和二个串口,可以用于打印和通讯,但是不能直接把这种卡插入带 5 吋硬盘的微机中使用,因为这些机器都配有自己专用的磁盘接口卡,而 3 吋硬盘的接口与 5 吋硬盘接口又不能通用,故多功能超卡上的硬盘接口不能代替原机上的硬盘接口卡,所以二卡同时插入,在选中磁盘时,必定会造成地址及数据冲突,使硬盘与软盘都无法使用,因此我们必须解决造成二种卡地址冲突的影响后,才能使用多功能超卡。下面我们谈一下如何解决的方法。

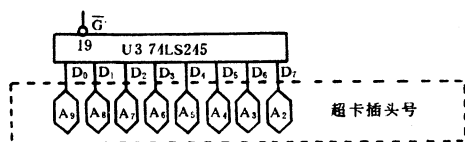


图1 多功能超卡的数据输入/输出接口

表(1)输入输出设备地址分配表

输入输出设备	地址(16进制)	地址<二进制>															
		A9	A8	A7	A6	A5	A4	A3	A2	A1	A0						
并行打印口1	378-37F	1	1	0	1	1	1	1	×	×	×						
并行打印口2	278-27F	1	0	0	1	1	1	1	×	×	×						
RS-232 串行口1	3F8-3FF	1	1	1	1	1	1	1	×	×	×						
RS-232 串行口2	2F8-2FF	1	0	1	1	1	1	1	×	×	×						
软盘控制器	3F0-3F7	1	1	1	1	1	1	0	×	×	×						
硬磁盘	1F0-1F8	0	1	1	1	1	1	0	×	×	×						
									1	0	0	0					

注:×为0或1任意

从图(1)中可见多功能超卡数据输入/输出线是通过 U3(74LS245),而 74LS245 是非反相三态输出的八

总线双向传送接收器。只有当 U3 的 19 脚为“0”时才能选通 U3,数据才可以输入或输出,否则它的 8 条数据输入/输出线为高阻状态。因此我们只要在磁盘读写时使这脚为“1”就能解决上述问题。

现在我们来看一下表(1),从表(1)可知若 A9=1 时硬盘就不能选中,A3=1 时软盘控制器就不能选中,也就是说当 A9=1 并且 A3=1 时,只能选中串并口,因此我们只要把 U3 的 19 脚断开,按如图(2)所示连接,就可以达到我们的目的

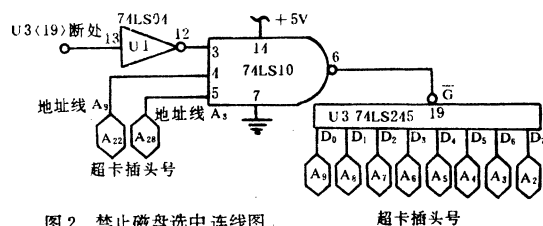


图2 禁止磁盘选中连线图

图中 U1 的 13 与 12 脚在原线路板中是空脚,我们在此可以利用这个反相器。74LS10 三与非门是我们另外加上去的。从图中可知凡多功能超卡中任意一接口地址有效时 U3 的 19 脚断处,即 U1 的 13 脚就是低电平“0”,通过 U1 反相输出为“1”,使 74LS10 的 3 脚为“1”,前面我们讲过只有 A9 与 A3 等于“1”时才能选中串并卡,而当磁盘选中时,由于 A9 或 A3 等于“0”,使 74LS10 的 6 脚为“1”,使 U3 的数据线为高阻状态,不影响磁盘专用卡上的数据输入/输出。另外我们还得把超卡上的软盘中断请求线 IRQ6 及硬盘中断请求线 IRQ14 二线割断,不使它产生中断请求。经过上述改变后超卡线路板就可以替换损坏的串并卡,用多功能超卡的串并口可以打印或进行通讯。

敬告订户和读者

现邮局正在办理今年下半年报刊征订手续,请广大订户和读者勿错失良机。本刊邮发代号为 62-68。

上半年未订到本刊的订户和读者可直接寄款到本刊编辑部,补订上半年刊物。本刊每份全年订价为¥12元。

本刊编辑部 1994 年 5 月

EGA/VGA/TVGA 屏幕特技技术

华中理工大学 (武汉 430074) 迟 鲲

现在的一些游戏机软件经常使用一些屏幕特技技术,以产生良好的视觉效果,下面给出常用的三种特技技术,并给出 C 语言源代码。

1 屏幕打开、关闭技术

当从一个比较大的文件读入数据来生成一个比较复杂的图形时,由于计算机速度不够快,给人的视觉效果是图形一部分一部分的生成。这时可采用此技术,在图形生成以前,先关闭屏幕,屏幕上不显示任何图形,待图形生成完毕以后,再打开屏幕,给人的感觉是整个屏幕一下子弹出来。EGA/VGA/TVGA 的操作定序寄存器(3C4H, 3C5H)的时钟寄存器(索引号 1)第 5 位提供了开/关显示器的功能,开/关显示器的 C 语言代码如程序段 1;这一部分 C 语言源代码在 TVGA 上运行无误,在 EGA/VGA 上运行时,要将 Evgaclosecrt() 中的第二个 while ((vert&8) == 0) 改为 while ((vert&8) != 0) 即可。

2 连续颜色变换技术

这种技术是对屏幕上的一种或几种颜色进行连续颜色变换,比如可以将红色连续变换为 16 种 EGA 标准颜色,造成一种霓虹灯闪烁的效果。程序段 2 就是一个例子,其功能为将屏幕上的白色按 16 种颜色进行连续变色,直到按下任一健才停止进行颜色变换。Turbo C 为实现这一功能提供了标准函数 setpalette(int colorm, int color),其功能是将调色板的 colorm 项变为 color,同样,只要不断变换 color 项,就可以得到连续颜色变换的效果。

3 屏幕的谈入谈出技术

这种方法的基本原理是:修改 DAC 寄存器的方法实现谈入,谈出;将 256 个 DAC 寄存器一次一次减 1,一次一次再设置,直到 DAC 寄存器全部减为 0,就产生了慢慢消失的效果;而后将其一次一次加 1,一直加到和原表相同,就实现了慢慢恢复显示的效果。程序段 3 先将屏幕慢慢消隐,再慢慢恢复。

根据以上介绍的方法,读者可以根据需要加以灵活应用,比如可以将三种方法结合起来使用,以达到更为理想的效果。

程序清单如下:

;程序段 1

```
void Evgaclosecrt();关闭显示器
{
```

```
    unsigned char vert,b;
    do
    {
        vert=inportb(0x3da);
    }while((vert&8)==0);
    do
    {
        vert=inportb(0x3a);
    }while((vert&8)==0);
    outportb(0x3c4,1);
    b=inportb(0x3c5);
    b1=0x20;
    outportb(0x3c5,b);
    return;
    }
    void Evgaopencrt();打开显示器
    {
        unsigned char b;
        outportb(0x3c4,1);
        b=inportb(0x3c5);
        b&=0xdf;
        outportb(0x3c5,b);
        return;
    }
}
```

;程序段 2

```
while(1)
{
    if(! bioskey(1))
    {
        i=1;
        for(i=1;i<=16;i++)
        {
            setpalette(WHITE,i);
            if(bioskey(1))
            {
                sign=0;
                break;
            }
            for(j=0;j<=10000;j++)
            for(k=0;k<=50;k++);
        }
        if(sign==0)
            break;
    }
    else
        break;
}
```

打印机两个典型问题的分析排除

兰州大学计算中心 (730000) 饶增仁

问题 1 现象:

FX-100 打印机, 加电自检正常, 但不能联机打印。

问题 2 现象:

M1724 打印机, 每当加电启动时, 面板指示灯仅闪亮一下, 随即打印机无任何反应, 打印头也不回左边界。

问题 1 分析排除:

打印机不能联机打印, 一般是接口问题, 要么是主机内并口, 要么是打印机侧接口的故障引起, 也有可能是“病毒”造成。用“无毒”DOS 启动后, 再联机打印时, 故障依然存在, 这说明并非由于“病毒”造成。进一步用交换法确诊是打印机内接口原因, 单独对打印机接口信号静态测试, 发现 STROBE 选通信号始终为低 (对地直流电压仅为 1.14V), 而选通信号是由主机并口送入打印机的 IC8 (LS14) 芯片的 ① 端, 输出是 IC8 的 ② 脚, 为 +5V, 见图 1 示, 显然 IC8 (74LS14) 六反相器有问题, 或是其输入 ① 脚外围电路有问题。经过换 IC8 为新的六反相器 74LS14 后, 问题解决。

问题 2 分析排除:

据面板指示灯能闪亮, 说明问题不在电源部位, 而在逻辑控制电路部分, 首先应检查打印机主 CPU 复位

信号 CPURST, 逻辑电路图见图 3 示。

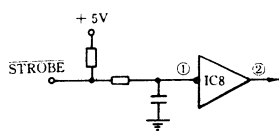


图 1

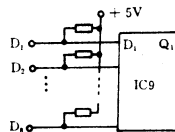


图 2

用示波器或万用表测量 #G3 的 RESET 输入电压波形 (或电位值) 见图 4(a), 正确的应为图 4(b) 而 RESET 信号是由 #G3 (LS04) 的 ⑧ 脚输送的, 再测 #G3 ⑨ 脚电位到正常 (+5V), 通过换 #G3 后, 问题解决。

以上两个故障都属于打印机接口问题, 造成这种故障原因很多, 这里要强调的是为防止人为造成, 在需要拔插打印机电缆线时一定要先关掉电源。

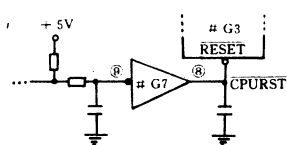


图 3

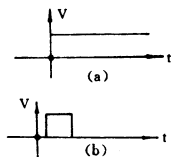


图 4

收稿日期: 1993 年 7 月 4 日

```

}
;程序段 3
void changecolor()
{
int i,j,color,sign=1;
char color1[768],color2[768];
for(i=0;i<768;i++)
{
outportb(0x3c7,i/3);
color=inportb(0x3c9);
color1[i]=color2[i++]=color;
color=inportb(0x3c9);
color1[i]=color2[i++]=color;
color=inportb(0x3c9);
color1[i]=color2[i]=color;
}
while(sign!=0);
{
sign=0;
for(i=0;i<256;i++)
{
for(j=0;j<3;j++)
{
if(color1[i*3+j]!=0)
{
sign=1;
color1[i*3+j]--;
}
}
outportb(0x3c8,i);
outportb(0x3c9,color1[i*3]);
outportb(0x3c9,color1[i*3+1]);
outportb(0x3c9,color1[i*3+2]);
}
}
sign=1;
while(sign!=0)
{
sign=0;
for(i=0;i<256;i++)
{
for(j=0;j<3;j++)
{
if(color1[i*3+j]!=color2[i*3+j])
{
sign=1;
color1[i*3+j]++;
}
}
outportb(0x3c8,i);
outportb(0x3c9,color1[i*3]);
outportb(0x3c9,color1[i*3+1]);
outportb(0x3c9,color1[i*3+2]);
}
}
return;
}

```

游戏机程序设计要点

王让定 王小牛 王联国 索国瑞 张召琪

西北师范大学计算机科学系开发应用室(兰州 730070)

摘要 任天堂家用电脑游戏机在我国已相当普及,如何在游戏机上进一步开发,使其更加适合我国家庭教育,是目前所探讨的课题。本文将作者在开发游戏机的过程中所总结的程序设计要点,介绍给读者。

关键词 图像信息 声音处理 键处理 中断

1 图像信息的设计

1.1 游戏机地址分配

6527 CPU 把 PRAM、PPU 和 PROM 作为单独的外部设备进行统一编址,其中主机板上 PRAM 占用空间 \$0000H \sim \\$1FFFH, 共 8K; PPU 占用空间 \$2000H \sim \\$3FFFH, 共 8K; 声音口及手柄占用空间 \$4000H \sim \\$5FFFH, 共 8K; \$6000H \sim \\$7FFFH 共 8K 为机内扩展区; PROM 占用空间 \$8000H \sim \\$FFFFH, 共 32K。其中 VRAM 地址分配如下:

\$0000H \sim \\$0FFFH: 字模低区, 共放 4K 字节字模;

\$1000H \sim \\$1FFFH: 字模高区, 共放 4K 字节字模;

\$2000H \sim \\$23BFH: 第一背景码区, 存放所要显示的图像码字;

\$23C0H \sim \\$23FFH: 第一背景码区所对应的 64 个色块;

\$2400H \sim \\$27BFH: 第二背景码区;

\$27C0H \sim \\$27FFH: 第二背景码区所对应的 64 个色块;

\$3F00H \sim \\$3FFFH: 颜色发生器。

1.2 字模设计

图像是字模码和背景码结合起来形成的。在设计游戏节目的背景、前景时必须根据所需的图像信息编制字模, 游戏机的字模点阵有两种: 8×8 点阵和 16×16 点阵, 8×8 点阵占用 16 个字节空间(如图 1), 16×16 点阵占 32 个字节空间。下面以 8×8 点阵为例说明字模的编制方法(以“1”为例):

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

图 1 8×8 点阵字模的存储格式

第一步、绘出图像的 8×8 点阵, 标出每一行的十六进制数(用 1 表示该点显示, 0 表示不显示);

																18H
																38H
																18H
																18H
																18H
																18H
																18H
																3CH

图 2 “1”的点阵图

第二步、根据图像写出字模;

每个字模有如下表一所示的七种表示方式:

	前 8 个字节	后 8 个字节
1	00H	字模
2	FFH	字模
3	字模补码	字模
4	字模	字模
5	字模	00H
6	字模	FFH
7	字模	字模补码

由于各种方式不同, 字模的前景、背景的显示颜色也不同, 图 3 所示的是采用表一中第一种方式表示“1”的字模:

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00	00	00	00	00	00	00	00	18	38	18	18	18	18	18	3C

图 3

第三步、将编写的字模送往字模高区 \$1000H \sim \\$1FFFH(或字模低区 \$0000H \sim \\$0FFFH);

第四步、图像码字的取法;

在提取背景码、前景码时, 取字模所在字模低区 \$0000H \sim \\$0FFFH(或字模高区 \$1000H \sim 1FFFH)地址的中间两位, 如上述“1”的字模存放在字模低区 \$0300H \sim \\$030FH, 则其码字是 30H。

1.3 背景码的设计

电视屏幕的分辨率为 256×240, 由于每个图像信息字模是 8×8 点阵, 所以把整个屏幕以 8×8 点阵为单元分成 32×30=960 个块, 要在某块位置上显示一

8×8 的图像信息,只需把相应的码字送到该块位置即可。屏幕块位置与前述背景码区 1、2 一一对应,如下图所示:

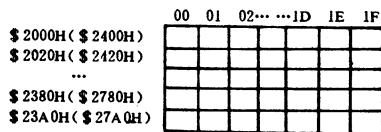


图 4 码区与屏幕映射图

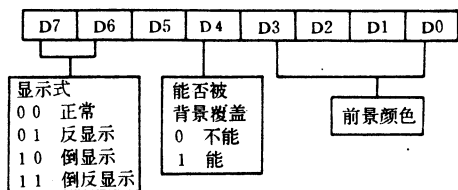
1.4 前景的设计

前景的格式为:

行坐标	码字	色码	列坐标
-----	----	----	-----

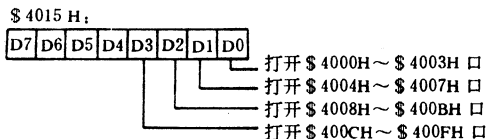
其中,行列坐标以点计算,码字是所要显示的前景图像信息,色码字位的含义如下:

前景码区在 PRAM 的 \$0200H~\$07FFH 之间的任何一页中,也可送往 \$2004H 口,具体在哪一页,由送往 \$4014H 中的控制字(02H~07H)决定,例如:(\$4014H)=07H,则前景码在 \$0700~\$07FF 之中,且每个前景码显示格式的四个字节放在 \$N0N1H~\$N0N1+3 中,其中 N0=07H(\$4014H 中的内容),N1=00H,04H,08H,0CH,10H,……,F8H,FCH 另外还必须向 \$2003H 口送 00H,以控制前景显示。



2 音响处理

声音的形成受中央处理器 CPU 6527 控制,其控制口是 \$4015H,具体的音质、音频、音量、音速以及长音、短音等由 \$4000H~\$400FH 的 16 个声音口确定,也就是说,往 \$4000H~\$400FH 中送入不同的数,将输出不同的声音。口地址具体控制方式如下:



游戏中的声音是通过以下控制字来实现的:

\$4000H:控制发音形式;

D7=1:发长音;D7=0:发短音

\$400AH:频率控制;

00H→FFH:高→低

\$400BH:音质和频率范围的控制;

\$400CH:发音形式和音量的控制;

(1)、D5D4 控制发音形式:

1 X:发连音; 1 1:发长音; 0 X:发顿音。

(2)、D3D2D1D0 控制音量和速度:

D4=0 D3D2D1D0=0→F:速度由快到慢

D4=1 D3D2D1D0=0→F:音量由小到大

\$400EH:控制音色和频率;

\$400FH:控制音质。

根据以上的控制字,再结合丰富的音乐知识即可设计出优美动听的旋律。

3 键盘(手柄)处理

手柄分为主手柄和副手柄,它们对应的端口分别是 \$4016H 和 \$4017H,手柄中的键值是以串行方式送到数据总线的 D0 位上,CPU 访问 8 次 \$4016H 或 \$4017H 后组成一字节,其各位含义如下(Di=1 表示该键被按下)

D7	D6	D5	D4	D3	D2	D1	D0
A	B	SELECT	START	↑	↓	←	→

编程时,先要根据需要选择主副手柄,方法是向 \$4016H 送 01H 时选中主手柄,送 00H 时选中副手柄,具体编程如下:

```

LDA    # $01
STA    $4016    ;选中主手柄
LDA    # $00
STA    $4017    ;选中副手柄
LDX    # 08      ;访问手柄次数
LOOP:
LDA    $4016    ;访问主手柄
NOP                    ;延时
OR     $4017    ;将主副手柄的值合在一起
LSR                    ;将访问的结果移到进位标志 C 中
ASL    $16      ;将进位标志 C 左移到 $16 单元中
DEX                    ;访问次数减一
BNE    LOOP     ;所有的键访问完了吗? 如果没完,则转 LOOP
RTS                    ;否则结束

```

然后根据 \$16H 中的内容转到相应的处理程序。

4 中断处理

中断入口以 \$FFFAH 中的内容为低八位,以 \$FFFBH 中的内容为高八位。一般情况下,将 PRAM 中从 \$0200H~\$07FFH 的任一单元的地址存放到 \$FFFAH(低八位), \$FFFBH(高八位)中,在该单元处存放一条三字节的 JMP \$XXXX 指令转到相应的

用DDD盘检测和调整软盘驱动器

吉林省计算机技术研究所(长春 130012) 梁学高 吉林省建筑工程学院 孙吉权

软盘驱动器是各种计算机最重要的外部设备之一,由于它结构精密、使用频繁因而故障率相对来说很高。一旦软盘驱动器出现故障,整台计算机就会瘫痪。

大量实践表明软盘驱动器的故障百分之九十以上为机械故障,特别是因为磁头定位不准而造成读写错误,也就是说绝大部分故障可通过磁头定位调整来修复。以前该项调整是通过AAD法(猫眼法)来实现的,但此方法有严重的缺欠:

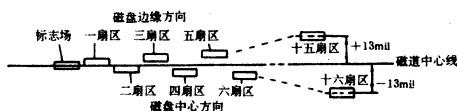
1). 对使用的仪器要求较高,比如中长余辉示波器等等。

2). 要求维修人员有较高的水平。

3). 对复杂调整所需时间太长。

近年来,美国DYSAN公司推出一种DDD盘(DIGITAL DIAGNOSTIC DISKETTE),这种盘含有五种类型的信息,分布在24条磁道上,并可以自己写一些信号到盘上,帮助做一些辅助检测。这种盘精度高(经向误差正负一密尔,方位角误差正负3分)且以数字显示,调整时不但可以指明磁头偏移方向,而且可以指明偏移的大小,因此大大提高了维修人员检测、调整软盘驱动器的效率。

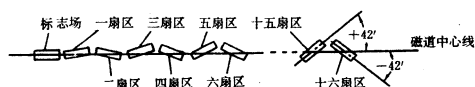
用DDD盘的经向调整是使用盘片上的递增磁道。这种磁道除标志场(ID)在磁道上外,所有16个扇区都偏离磁道:第一扇区往盘片的边缘方向偏6密尔(+6mil),第二扇区往中心方向偏6密尔(-6mil),第三扇区又往边缘方向偏7密尔,第四扇区往中心方向偏7密尔……以此类推,直到第十五扇区往边缘偏13密尔,第十六扇区往中心偏13密尔,参见下面示意图。



这样,当磁头由盘片上相对滑过时,在读出标志场以后,就开始依次读出扇区的信息,根据无错误读出的扇区数,就可以知道当前磁头偏移的方向和大小。

例如:当磁头最终可正确读出+13和-7时,则表明磁头已往磁盘边缘的方向偏了,大小为这两个数绝对值相减后除以2,即 $1/2(|+13| - |-7|) = 3$ (密尔),这时我们只要把磁头往中心方向调整3密尔就可以了(注意对于不同的驱动器调整部位不一样)。

对于方位角偏斜的检测、调整,该盘片有交替磁道可用。同递增磁道不同的是在这种磁道上标志场和所有十六个扇区的信息全都记录在磁道上,但是这些扇区的信息都记录得和磁道有一个夹角,夹角的大小是从+21分(-21分)开始,以+3分(-3分)为增量直到+42分(-42分)。如下示意图所示。



同递增磁道一样,当磁头相对盘片滑动时,在读出标志场后就开始依次读出扇区的信息,相应地显示出读出的信息(转换成偏斜角度以后),如果磁头偏斜,显示出的两个数字的绝对值就不会相等,就要对磁头的偏角进行调整。例如:这时显示出+39, -27的话,则这两个数绝对值的差的一半即 $1/2(|+39| - |-27|) = 6$ (分)就是此时磁头的实际偏斜量。调整驱动器的相关部位,使正负二个数的绝对值相等就可以了。

以上是利用该盘片调整软盘驱动器的主要功能,此外还可以用它来调整主轴转速、中心位置、零磁道开关等等。86年我们利用该盘片研制了QDJ-1型驱动器检测仪,性能指标目前仍是国内最高水平,87年中试生产后现已推广到全国二十余省、市、自治区,取得了很好的经济效益和社会效益。

收稿日期:1993年12月11日

中断服务子程序。

为避免屏幕抖动,将游戏中的显示及屏幕移动的处理放在中断服务程序中,在中断服务程序中通过改变显示行列坐标,使背景发生上下、左右移动,通过改变前景的行列坐标,使前景发生上下、左右移动,从而达到动画效果。但是,中断程序必须在0.02秒内完成,且返回调用者,否则会发生混乱而死机。因此,当中断

服务子程序较长时,要化整为零,将其分成几部分,每次中断,根据相应条件处理一部分,以避免死机现象。

参考文献

- 1 王让定,王小牛. 一种高性能游戏控制器. 计算机应用研究. 1993. 3, P. 57~58
- 2 王让定等. PC系列微机总线与任天堂系列游戏机卡槽总线的接口. 计算机应用研究. 1993. 2, P. 37~39

Foxpro 下拉菜单系统的设计

徐州矿务局物资供应管理处(221006) 侍永新

摘要 本文讨论了 Foxpro 下拉菜单系统的概念及设计方法,并设计了一个菜单系统的所有数据均与程序相对独立的菜单程序。

关键词 Foxpro 下拉菜单系统 条形菜单 菜单点 弹出式菜单 菜单项 激活

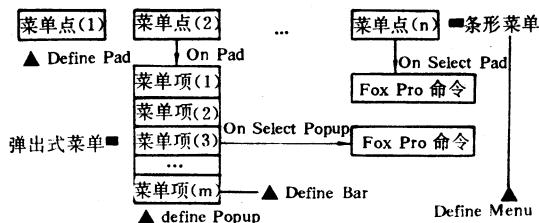
相对 Foxbase 而言, Foxpro 下拉菜单设计更加灵活、方便。Foxpro 下拉菜单命令共有三组,分别为定义性命令、关联性命令和激活性命令。定义性命令的作用是定义下拉菜单的名字和内容;关联性命令作用是解释菜单各部分间的关系;激活性命令作用是激活下拉菜单。为讨论方便,首先列出下拉菜单设计所需的命令及其作用,如下表所示:

分类	序号	命 令	作 用
定义性命令	1	Define Menu <Menu Name> Message<ExpC>	定义一条形菜单
	2	Define Pad<Pad Name> of <MenuName> Prompt<ExpC1>[at<Row,col>] [Message<ExpC2>]	为一条形菜单定义菜单点
	3	Define Popup<Popup name>From <Row1,Col1>To<Row2,Col2> Message<ExpC>[...]	定义一弹出式菜单
	4	Define Bar<ExpN>of<Popup Name> > Prompt<ExpC1> Message<ExpC2>	为一弹出式菜单定义菜单项
关联性命令	5	On Pad <Pad Name> of <Menu Name> [Activate Popup<Popup Name>]	将一菜单点与一弹出式菜单联系起来
	6	On Select Pad <Pad Name> of <Menu Name> [<Command>]	将一菜单点与一命令联系起来
	7	On Select Popup <Popup Name> All <Command>	将一弹出式的菜单项间接地与某一具体操作(过程)联系起来
激活性命令	8	Activate Popup<Popup Name>	激活一弹出式菜单
	9	Activate Menu<Menu Name>[Pad<PName>]	激活一条形菜单(下拉菜单系统)

注:命令格式仅列出了常用的项,其它见参考手册。

我们可以将整个下拉菜单系统形象地表示为下

图,其中,方框内为菜单的组成部分,箭头连线表示关联,▲表示定义,□为名称。



整个下拉菜单系统是由上层的条形菜单、下层的弹出式菜单及 Foxpro 命令组成。条形菜单与 Foxplus 的亮条菜单有些相似,由菜单点(Pad)构成,而每个菜单点又可通过命令语句与 Foxpro 命令或弹出式菜单相联系。条形菜单由命令语句 Define Menu 定义,它仅定义一个菜单名字,其内容即菜单点由命令语句 Define Pad 定义。每个菜单点既可通过 On Select Pad 与一 Foxpro 命令相联系,又可通过 On Pad 与一弹出式菜单相联系。On Pad 使得一旦该菜单点成为条形菜单的当前项,则相应的弹出式菜单自动地显示(当然,也可自动消隐),而 On Select Pad 则需要用空格键或回车键选中该菜单点才能执行相应的命令。

弹出式菜单与条形菜单相似,仅定义方式与表现形式不一样。它由 Define Popup 定义菜单名,由 Define Bar 定义其内容即菜单项。每个菜单项由 On Select Popup 与各自的 Foxpro 命令相联系,其联系方式与 On Select Pad 相似。一般地,当选中一菜单项时,将函数 Pad()和 Bar()作为参量传递给一过程,由该过程分析处理下一步的操作。在下拉菜单系统中,弹出式菜单是由 On Pad 语句隐式激活的,当然,它也可在程序中单独使用,并用 Activate Popup 激活。

最后,用 Activate Menu 语句激活整个下拉菜单系统。在使用时,用空格键或回车键选中。

以上介绍了下拉菜单的组成、概念及使用方法。但是,要想使用得好,得结合自己的需要,编制有自己特色的菜单程序。没有绝对通用的程序,“好”的程序也不是生成出来的。下面给读者提供一下下拉菜单程序,供读

者参考,相信读者在此基础上能编出更好的,更能满足自己需要的下拉菜单程序。

该下拉菜单程序的特点是:构成下拉菜单的数据(菜单点、菜单条提示符等)与程序相对独立,并且所有数据均在一个数据库文件中。这种紧凑型的风格使得无论是程序调试还是重新构造系统都很方便,笔者在实际中使用,感到非常满意。

保持数据与程序相对独立的基本思路是:将下拉菜单的所有数据放到数据库文件中去,并且为了紧凑和使用的方便与灵活,把所有数据均放到同一个数据库文件中。程序基本分为三部分,首先初始化应用环境,然后在数据库文件中提取菜单系统数据,最后定义并激活下拉菜单系统。该程序运行于 Foxpro 2.0 环境下,并要求有能智能识别西文制表符的字符型汉字系统如超想汉字系统的支持。关于程序的其它说明见程序注释。

Note. Program: Menu. Prg New Virsion

* Author: ShiYongXin(侍永新)

* Date: 1993. 6. 26

* Copyright(c)1993. 6 S. Y. X

* MENU.DBF 字段的意义说明:

* Name: 菜单点或菜单项的名称

* D1: D1<0 表示 Name 为条形菜单的菜单点,且 Abs(D1) 为该菜单的属性,如若其它记录的 D1 等于该属性值,则为该菜单点所关联的弹出式菜单的菜单项。

* D1=0 表示本记录 Message 字段为应用系统标题;

* D1>0 表示 Name 为弹出式菜单的菜单项,其关联的条形菜单为 D1 值等于该项值负值的那条记录。

* D2 同组菜单中以此值顺序排列。

* Prg: 基菜点或菜单项对应的程序名或过程名,可以为空。

* Message: 菜单条或菜单项的描述,若 D1=0 则为应用系统标题。

Set Talk Off

Clear All

* Note. 此处可以做以下工作:

* 1. 设置 Set 开关项;

* 2. 恢复应用系统的内存变量;

* 3. 打开数据库文件等。

* 可以用过程文件或函数方式调用。

Select 0

Use menu

* Note: 应用系统文字描述或存在则写到 0 行

local for d1=0

if found()

Sys- title=alltrim(message)

else

Sys- title=" "

endif

Set Color To W+/B

@0,0 Clear To 0,79

@0,0 Say Padc(Sys-title,80)

Set Color Of Scheme2 To N+/W,N+/W,N+/W,,N/W,
N/G,R/W,N/N

* Note: 设置系统屏幕色彩

Set Color To B/W

@1.0 Clear To 1.79

Set Color To W* /N

@2,0 Clear

* Note: 此处可调用系统注册程序,防止非法用户闯入,如:

* if. not. Login()

* Quit

* endif

* 当注册成功时函数返回真值。

sele menu

index on d2 to temmenu

set filter to d1<0

calculate CNT(), max(len(alltrim(name))) to topnum,
Padlen

dimension top(topnum) && 定义主菜单数组

dimension tz(topnum) && 定义主菜单条特征数组

dimension prgname(topnum,18) && 定义各菜单对应的过程名

Note: Get the Main Pad—name from database

* * * * * 从数据库文件中取菜单点名称

i=1

go top

Scan

tem=int((padlen-len(alltrim(name)))/2)

top(i)=padc(alltrim(name),Padlen)

tz(i)=abs(d1)

Prgname(i,1)=alltrim(prg)

i=i+1

EndScan

Note: Define the Main Menu Pad

* * * * * 定义条形菜单及其菜单点

spacelen=int((80-topnum*padlen)/(topnum+1))

define menu main

i=1

do while i<=topnum

tem=alltrim(str(i))

define Pad menuPad&tem of main prompt top(i);

at 1,spacelen+(spacelen+Padlen)*(i-1)message"

i=i+1

enddo

Note: Define The Popup Menu And Relations

* * * * * 定义弹出式菜单及关联

i=0

do while i<topnum

i=i+1

tem=alltrim(str(i))

set filt to d1=tz(i)

calculate cnt() to tolrec

do case

实现矢量汉字的一种方法

合肥电子工程学院 303 单 洪

安徽省计算中心(合肥 230031) 李长喜

摘要 本文主要介绍了实现矢量汉字输出的方法,包括汉字库的组织 and 具体实现方法两部分内容。

关键词 汉 化 矢 量 汉 字 汉 字 标 准 坐 标 变 换

在实现系统的汉化改造时,使用矢量汉字输出可以很方便地实现汉字的缩放、任意角度的旋转等,尤其是在大屏幕显示器上显示地图、图形、图像等信息时,矢量汉字的显示输出就显得更为重要。另外矢量汉字还可以很方便地在绘图仪上写出来,这就为汉字系统中图形的输出提供了另一个重要的手段。下面就矢量汉字的具体实现方法作一简要介绍。

1 矢量汉字库的组织

矢量汉字库不同于计算机系统点阵汉字库,后者主要是为计算机屏幕正常显示、打印和存储用的,而矢量汉字库是矢量数据库,可直接用绘图仪写出汉

字,也可用于屏幕显示。在汉化系统时,所使用的矢量汉字库应包括常用汉字,以及英文、俄文、希腊文字母等共 6000 多个,基本上要包括 GB2312-80 中所列出的汉字及符号。这些字一般可经数字化仪输入到机器中的,将这些数据组织成一个完整的字库数据文件。

矢量汉字在字库中是以坐标的形式表示的。即一个矢量汉字的信息为坐标值的组合。而各坐标之间可能要画线,也可能不要画线,必须通过一定的标志将此特征表示出来才能正确地输出汉字。其中的一个问题是坐标参考点如何选取。例如选取的方式:汉字的初始大小定为 100×100 的范围,并假定汉字为一正常不作旋转的汉字,其坐标值在此基础上根据具体的汉字来确定。例如“大”字的坐标值可能为:dot1(7,5),dot2(14,11),dot3(24,19),dot4(34,31),dot5(41,41),

收稿日期:1994 年元月 9 日

```
case tolrec=0. and. len(prgname(i,1))=0
on select pad menupad&tem of main=1
loop
Case tolrec=0. and. len(prgname(i,1))<>0
on select Pad menupad&tem of main do&Prgname(i,
1)
loop
otherwise
define popup popmenu&tem from 2, spacelen+(space-
len+padlen)*(i-1)shadow
end case
j=1
go top
Scan
define bar j of popmenu&tem prompt alltrim(name)
message alltrim(message)
prgname(i,j)=alltrim(prg)
j=j+1
EndScan
on pad menupad&tem of main activate popup
popmenu&tem
enddo
on select popup all do pro DO with pad(),bar()
```

Activate Menu main&& 激活主菜单,可将该命令嵌在一循环中。

```
Clear All
Close All
Clear
Quit
Procedure ProDO
Para mpad,mbar
if mbar=0
return
endif
i=1
do while i<=topnum
tem="MENUPAD"+alltrim(str(i))
if tem=mpad
exit
endif
i=i+1
enddo
if len(alltrim(prgname(i,mbar)))<>0
do &prgname(i,mbar)
endif
Return
```

dot6(45, 55), dot7(48, 65), dot8(50, 97), dot9(50, 62), dot10(55, 49), dot11(62, 38), dot12(70, 26), dot13(82, 15), dot14(94, 7), dot15(7, 68), dot16(92, 68), 其中坐标点 dot8, dot9 之间、dot14, dot15 之间不要画线, 其它坐标点间要画线, 具体作图情况见图 1。

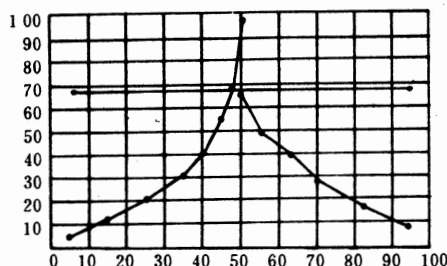


图1 大字的矢量坐标

矢量汉字库即可按上述方法将 GB2312-80 中大部分的汉字和符号组织成一个矢量汉字库数据文件。

由上可知, 要显示一个矢量汉字, 首先应知道此汉字要在屏幕上显示的位置, 然后再根据矢量信息进行坐标变换才能得到各坐标点的绝对坐标, 从而实现矢量汉字的显示, 同样的道理也可在绘图仪这样的设备上输出。

2 矢量汉字的索引

矢量汉字数据采用信息块方式组织, 每个汉字的矢量信息(坐标值)构成一个独立的信息块, 全部汉字的信息块有序地汇集成为汉字库, 即一个随机数据文件。由于每个汉字的笔划不同, 其信息块的大小也就不同, 为了要索引其中的矢量汉字就必须知道其在字库中的位置即偏移量, 也就是说, 我们必须根据此偏移量来索引汉字的记录信息。但是, 此偏移量由于汉字信息块大小不同而是非线性连续的, 这样国标码就不象点阵汉字那样与偏移量具有一个直接的对应关系, 而一般的汉字系统都要求用国标码来索引汉字。为了要达到此目的, 在汉化的系统中应建立了一个国标码与偏移量的映射文件, 即 table 文件, 用户可以通过此文件将国标码转换成汉字在矢量字库中的偏移量从而得到汉字的矢量信息, 达到在屏幕上显示或在绘图仪上写出汉字的目的。此文件可以与字库文件放在同一目录下。

3 显示汉字的实现

由上可知, 要将一个用国标码表示的汉字在屏幕上显示出来, 首先要通过索引文件 table 将国标码转换成此汉字在字库中的偏移量信息, 然后再根据偏移量从字库中得到汉字的矢量信息, 最后才能根据显示要求如大小、旋转角度等在屏幕上显示出矢量汉字。

例如用 C 语言实现在图形显示器上以矢量形式显示一个汉字的函数为 mgicgvc, 其语法形式为:

```
int x, y, hx, hy;
float nx, ny;
char *cch;
mgicgvc(x, y, hx, hy, nx, ny, cch)
```

其中的变量 x, y 指出的是显示汉字的中心点坐标; hx, hy 指出的是显示汉字的大小, hx 是汉字的宽度, hy 是汉字的高度; nx, ny 是显示汉字的旋转角度, nx 是沿 X 轴的旋转角度, ny 是沿 Y 轴旋转的角度。例如, nx 和 ny 都为 0.0 时汉字就不作旋转; 若 nx, ny 都为 180.0 则显示的是一个倒着的汉字; 而当 nx 为 0.0, ny 为 90.0 时, 则汉字就变成了一条横线。变量 cch 是一个指向汉字国标代码的指针。

具体步骤见下图 2。

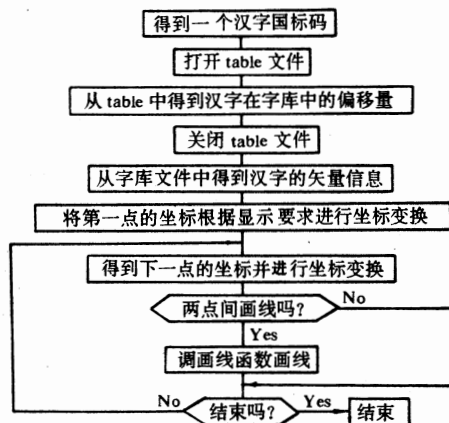


图2 显示矢量汉字程序框图

上述函数用 C 语言写成, 用 C 语言编写显示汉字和图形程序时, 可以直接调用。若要用绘图仪画图时, 在函数中加入绘图仪的驱动程序, 即将在屏幕上画线的语句改成绘图仪命令即可。

4 小结

以上简要介绍了一种在计算机上实现矢量汉字的方法, 其中的矢量汉字为线体字, 字型不是太美观。但是利用这种方法可以很方便地实现比较复杂矢量汉字的输出。

参考文献

- 1 郭平欣, 张淞芝编. 汉字信息处理. 国防工业出版社. 1984
- 2 王世明, 单洪. MC6700 微型机图形子系统汉化技术报告. MC6700 机汉化技术鉴定会. 南京, 1993

主从分布式监控系统的设计

西南交通大学计算机系(成都 610031) 余立建 王长林

摘要 作者应用局部网络技术,采用 8044 单片机智能仪器和 IBM PC(兼容机)构成主从分布式监控系统,由主机系统进行协调、调度、向下位机发布操作命令,从而达到实时控制、集中管理之目的。同时,该系统还将下位机回传的各种参数及有关操作自动存入数据库,由数据库管理系统完成检索,输出报表及变化曲线。

关键词 局部网络 专家系统 实时监控 双向 RAM SDLC 规约

1 概述

IBM PC 机(兼容机)与单片机测控仪器按照一定的方式及协议构成小型集散控制系统,是单片机应用发展的趋势。它正在仓库监测,气象、水文测报,炉群控制,电力、化工、供水,供气(汽)等系统中得到广泛的应用。本文就一个工厂的供水、供气(汽)监控系统进行分析和设计,系统的下位机采用智能测流流量测控仪表,通过开关设置,可测量液体、气体、蒸汽;上位机采用 AST 微机作为主机。上位机完成数据采集,传输、存储与处理全过程的自动化和电算化,将各下位机回传的参数(瞬时流量,累计流量等)自动存入数据库,由数据库管理系统检索,统计出各分厂,车间、部门用水、用气(汽)情况,进行考核,作为产品成本核算的重要依据,进而对信息进行分析,对系统进行模拟,提供短期预测和支持决策的建议。其结构原理如图(1),系统以主从结构联网,符合 SDLC 通信规约,SDLC 是基于在公共通讯链路上,主站和单个或多个从站之间进行数据交换的规约。主站负有控制整个网络的责任,它向从站发送各种控制命令,而从站只能按照来自主站的命令动作,并作出相应的回答。主站可以选择任意一个从站交换数据,构成通讯的双方,从站一旦被选中,既可以接

收来自主站的信息,也可以向主站发送。通信接口采用标准 RS-422A 接口,该接口允许在传送线上连接多个接收器,适当降低传输速率,最大传输距离可达 1200M。

2 系统的硬件设计

2.1 主站与主机的硬件设计

主站的 8044 单片机与主机系统总线(PC62 总线)硬件接口如图(2)所示,通过扩展的并行口,完成主站与主机数据的并行交换。主站板插在 62 总线扩展槽内,数据并行交换在主站板上的双向 RAM 内进行。在发送数据过程中,主机先把数据写入双向 RAM,并向主站发送标志,主站 8044 单片机响应后,把数据读入片内 RAM 中,并以 SDLC 规约将这些数据发送到指定的从站。在接收数据过程中,从站将 SDLC 信息帧发送到主站,主站 8044 接收到数据后,将数据放入缓冲区,并把数据组合处理后送到双向 RAM 中,向主机发送标志,主机响应后,读取双向 RAM 中的数据,存入相应的数据库中。

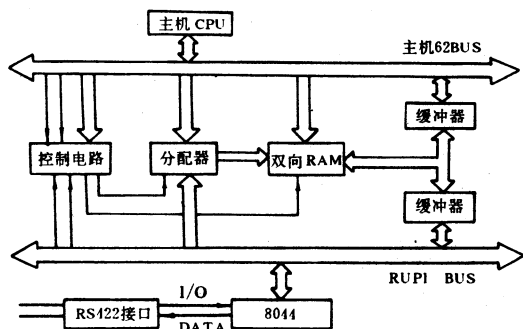


图2 主站与主机接口原理图

2.2 从站的硬件设计

如图(3)为从站的硬件原理图,可用键盘设置各种参数和调用系统功能,同时测量两路流量的流速(频率)、压力、温度,瞬时流量,累计流量,适用流体介质为:液体,气体,蒸汽,压力P,温度T,频率F由变送器转换成0°~5°直流信号,经过A/D转换送入8044缓

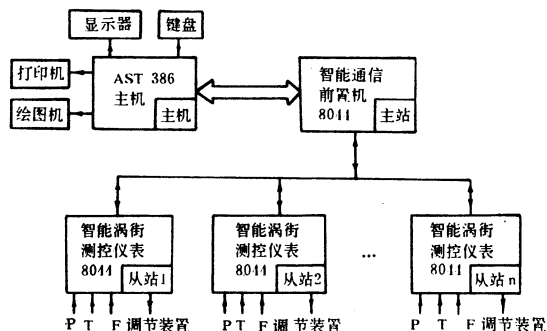


图1 系统结构原理图

冲区内,不变参数及有关测试结果存入 E²PROM 中,根据主站的命令及有关参数,通过 D/A 给出信号控制调节装置

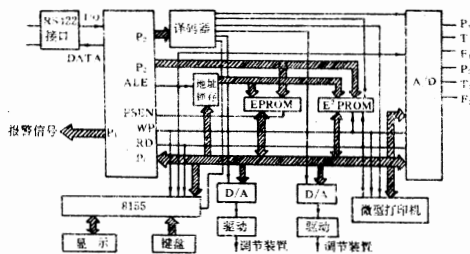
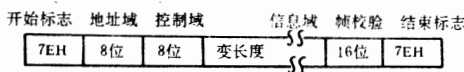


图3 从站硬件原理图

3. 系统的软件设计

3.1 SDLC 的帧结构

在 SDLC 规约中,帧是信息传输的基本单元。SDLC 的帧,可以用于通讯线路的控制,也可以用于数据传递。整帧信息由开始标志,地址域,控制域,信息域,帧校验序列,结束标志组成:



标志序列 7EH 作为开始标志及结束标志,在接收过程中,又作为同步字符使用;地址域用来选择从站,无论是主站发出的命令帧,还是从站发出的响应帧,其地址域都是从站的地址。控制域标明 SDLC 规定的三种不同类型的帧:信息帧用来传递数据,管理帧用于控制,非序号帧用于启动和控制从站。信息域的长度是任选的,它包含着所要传送的数据,当用于启动和控制从站的非序号帧,帧内并不需要信息域,即该域长度为 0;利用每一帧所带的 16 位帧校验序列(循环冗余码),对随机或突发差错造成的帧破坏,有很好的校验效果。

3.2 主要程序框图

图(4)为主机主控程序框图,平时执行实时监控功能,监测各个从站(车间,分厂)的用水,用汽情况,发布各种命令。其它功能完成核算指标输入,进行成本核算,输出报表及有关曲线。图(5)是主机,主站的程序框图,系统采用循环监控方式。图(6)是从站程序流程图,图(7)为 SIU 中断程序流程图。

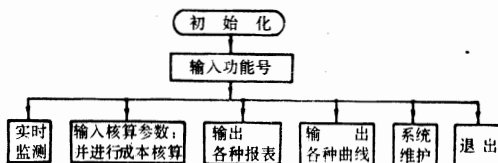


图4 主机管理系统框图

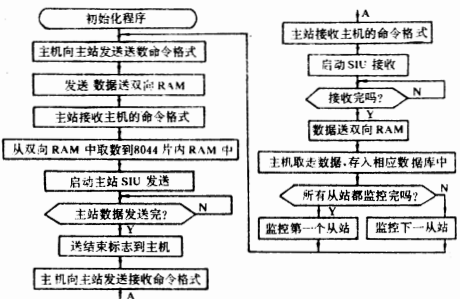


图5 主机、主站通讯框图

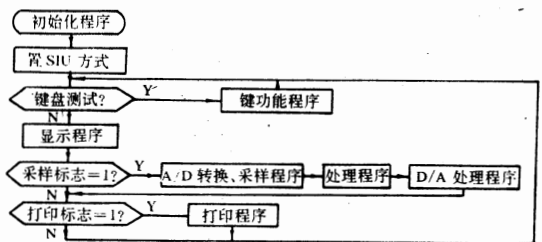


图6 从站程序框图

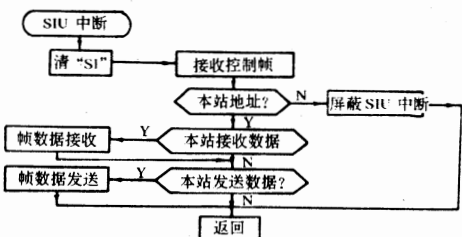


图7 SIU中断程序框图

4 结束语

本系统应用局部网络技术,采用 IBM PC 机(兼容机)作为主机,主站和主机采用并行通讯方式,并且主机和主站在双向 RAM 中交换数据,克服了传统数据通讯中占用主机大量操作时间的弊病,使主机从繁忙的通讯事务中解脱出来。另一方面,现场监测数据自动存入数据库,形成历史文件,通过归纳,整理再加上专家知识,形成专家系统。当专家系统建立后,就可自动通过主站向从站发布各种操作命令,监控从站,从而达到自动调度,实时控制,集中管理之目的。

参考文献

- 1 何立民主编. 单片机应用文集. 北京航空航天大学出版社
- 2 余立建等. 智能多路流量仪表的研制. 计算机应用研究, 92年4期
- 3 余立建. 单片机智能仪器与 PC 机之间的通讯. 电子与仪表, 92年4期

只用二张 360K 的低密盘也可以使用 WPS

辽宁财政专科学校(丹东 118001)朱广忠

1989 年 6 月

WPS 是目前十分流行的一种文字处理系统软件。它的新颖友好的用户界面,操作简便的编辑手段,多种汉字输入方式,灵活的排版制表方法,强大的打印功能,倍受广大用户的欢迎。已经成为各个部门办公自动化主要工具,但是早期购买的低档 PC 机没有配置硬盘,而西山 DOS5.0、5.10 因显示方式及字库大无法在 PC 机上使用,许多用户渴望在无硬盘的机器上也能使用 WPS 文字处理系统。为此推荐 CCDOS4.03 版本可以只用二张低密盘在无硬盘的任何机型上使用 WPS 的方法。下面介绍如何实现的。

1 CCDOS4.03 及 WPS 的主要文件分析

1). CCDOS4.03 主要模块功能:

CCLIBJ.DOT 简体显示软字库模块

CHLIB.COM 西山 CCDOS 字库管理模块

VDKEY.COM 西山 CCDOS 屏幕、键盘管理和汉字国标区位码输入模块

PY.COM 西山 CCDOS 多功能汉语拼音汉字输入模块

WBX.COM 五笔字型输入法模块

2). WPS 主要模块功能:

WPS.EXE WPS 文字处理系统工作环境主程序

WPS1.OVL WPS 文字处理系统屏幕、文件服务、帮助等辅助程序

WPS2.OVL WPS 文字处理系统字型平滑变换辅助程序

WPS.CFG 系统配置文件(由系统自己生成)

2 CCDOS4.03 系统启动过程的分析

1). 用 DOS3.3 启动机器

2). 启动 CHLIB.COM

CHLIB 检查当前目录下是否有简体字软字库 CCLIBJ.DOT,如果没有则检查当前目录下是否存在繁体字软字库 CCLIBF.DOT,若找不到,则报告“CCLIB NOT FOUND”否则,根据其字库大小和用户需要分配内存,若有扩展内存,则将字库读到扩展内存里,然后将字库常驻部分读内存,并将字库读写的程序驻留内存。

3). 执行 VDKEY.COM:此模块识别显示器类型,设置键盘中断入口地址,将显示器设置成图型方式,后屏幕显示如下信息:

西山 CCDOS 版本 4.03

香港金山公司金山电脑有限公司

半角 英文数字:

11:03:12

4). 启动 PY.COM 进入多功能拼音输入方式(双拼双音,全拼双音)。

5). 启动 WBX.COM 进入五笔字型输入方式。

6). 启动 WPS 后屏幕下面显示:

警告:Super 汉卡没有安装,按任意键将继续

敲任一键,则进入主菜单,到此 WPS 启动成功可以进入编辑。

3 制做二张 360K 软盘启动 WPS 系统盘

从 CCDOS4.03 启动过程分析出本系统主要文件是:CCLIBJ.DOT、CHLIB.COM、VDKEY.COM、PY.COM、WBX.COM。WPS 的主要文件是:WPS.EXE、WPS1.OVL、WPS2.OVL、WPS.CFG,这些文件是不可缺少的,只要有了这些文件,就可以进入 WPS 正常使用拼音及五笔字型进行编辑,经反复试验,发现这些主要文件的字节空间不足两个低密盘,因此只需用两张低密盘(其中一张可以带系统)就可装下这些文件,其具体做法如下:

1)用 DOS3.3 格式化一张带系统的 360K 软盘称 1#盘,再格式化一张不带系统的 360K 软盘称 2#盘。

2)然后将主要文件分别拷入 1#和 2#盘中,内容如下:

1#盘:

Volume in drive A has no label

Directory of A:\

COMMAND COM 25308 2--02-88 12:00a

CCLIBJ DOT 243648 4--09-88 3:32p

CHLIB COM 20681 11--18-89

AUTOEXEC BAT 14 11--08-93 12:22p

VDKEY COM 15465 11--18-89

5File(s) 1024 bytes free

其中 AUTOEXEC.BAT 的内容如下:

ECHO OFF

CHLIB

VDKEY

ECHO ON

2#盘:

Volume in drive A has no label

Directory of A:\

PY COM 63040 11--18-89

WPS EXE 56653 11--18-89

收稿日期:1994 年元月 15 日

给硬盘“贴上”写保护

南京工程兵工程学院计算机教研室(210007) 戴水贵

为了防病毒,人们首先想到的就是在软盘上贴上写保护!能否给硬盘也贴上写保护呢?回答是肯定的。但硬盘的写保护无法采用物理操作方法(除非把硬盘的写盘信号线挂起),而只能采用软件方法——把写硬盘的软中断口接管!

笔者鉴于接管写硬盘软中断口这一思想,成功的用软件方法给硬盘“贴上”了写保护(程序清单略)!当该程序运行后,硬盘就被“贴上”了写保护。此后,所有对硬盘的写盘操作都无效,即无法把文件拷入硬盘,也无法把硬盘上的文件删除!但对软盘的写盘操作仍正常。

更为可喜的是,如果有病毒企图侵入硬盘,则病毒必然要对硬盘进行写盘操作,此时,计算机会自动发出报警声!

由于硬盘处于写保护状态,当计算机发出报警声时,用户即可知道有病毒在活动(但病毒无法感染到硬盘或硬盘的文件上,因为硬盘已处于写保护)。

为了使硬盘上的写保护既能“贴上”又能“撕开”,程序特地设置了“贴上”和“撕开”的操作密码!

“贴上”写保护的操作方法为:按下 CTRL+F9(不管当前硬盘是否已处于写保护状态,只要按下 CTRL+F9,即把硬盘置为写保护状态)。

“撕开”写保护的操作方法为:按下 CTRL+F9后,再键入密码 QWERT(密码可由用户自己改动)。

当把硬盘的写保护“撕开”后,一切对硬盘的操作都恢复正常。

1 程序使用方法

把汇编源程序 WRPRO.ASM 编译、连接,生成

WRPRO.COM 即可进行安装运行。

步骤为:

MASM WRPRO;回车

LINK WRPRO;回车

EXE2BIN WRPRO.EXE WRPRO.COM 回车

DEL WRPRO.EXE 回车

WRPRO 回车

安装运行后,硬盘即处于写保护状态(最好把这一步放在 AUTOEXEC.BAT 中运行)。

2 如何利用 WRPRO.COM 程序防病毒

病毒侵入硬盘一般有两种途径:即引导型病毒和感染型病毒。

引导型病毒是通过用带病毒的系统盘启动而感染的,而感染型病毒则是由于向硬盘拷入带病毒的文件而交叉感染的。

WRPRO.COM 可以有效的防止感染型病毒(硬盘处于写保护后,做实验的学生可以使用硬盘上的所有信息,但学生无法把软盘上的文件拷入硬盘,也无法把硬盘上的任何文件删掉),但无法阻止引导型病毒感染硬盘(因为,当用带病毒的软盘引导时,WRPRO.COM 还未运行)。为了解决这个问题,笔者又开发了一个与 WRPRO.COM 相配套的程序 WPW.COM。

WPW.COM 程序的功能是:如果用户用软盘启动,则无法进入硬盘,只有从硬盘启动时才能进入硬盘(当从硬盘启动时,由于 WRPRO.COM 放在 AUTOEXEC.BAT 中,因而启动后,硬盘就处于写保护状态了)!

另外,为了防止病毒感染 WRPRO.COM 自身,最好把 WRPRO.COM 文件改为仅读属性。

程序的设计思路请读者阅读程序清单(篇幅所限,清单从略,需要者,可寄 5 元工本费向本刊邮购)。

收稿日期:1993 年 12 月 20 日

```
WPS1 OVL 37000 11-18-89
WPS2 OVL 33223 11-18-89
WPS CFG 1016 1-01-80 12:02a
WBX COM 47566 11-18-89
WP BAT 14 1-01-80 12:04a
7File(s) 77824 bytes free
其中 WP.BAT 的内容如下:
ECHO
CLS
PY
WBX
WPS
ECHO ON
说明:批处理的文件名不能用 WPS.BAT
```

4 用二张 360K 软盘启动步骤如下:

1)将 1# 盘插入 A 驱动器,开机启动 DOS3.3 后由自动批处理进入西山 DOS 系统显示信息如下:

```
西山 CCDOS 版本 4.03
香港金山公司金山电脑有限公司
1989 年 6 月
半角 英文数字: 11:03:12
```

2)取出 1# 盘在 A 驱上插入 2# 盘,键入 WP.便进入 WPS 主菜单,这时便可以进行了,建议把 1#, 2# 盘防写口封好,防止文件丢失和病毒感染,把要编辑的文件存入 B 盘。

(此方法在 PC 机、神州 286、神州 386、LEO386、AST286 机上通过。)

双积分型 A/D 片与单片机接口大全

辽宁师范大学物理系(大连 116022)孙庆梁 大连海运学院 孙建国

摘要 本文叙述双积分型 A/D 转换器与 51 系列单片机的常规接法及三种简单实用的接口方法和程序框图。

关键词 A/D 转换 接口

双积分型 A/D 转换器具有转换精度高、抗干扰性好和价格低,因而广泛应用于数字电压表、数字面板表及采样速度不高的计算机系统。

双积分 A/D 片的工作过程,以 $4\frac{1}{2}$ 位 BCD 码输出的 ICL7135 为例,可分为 A/D 转换和数据输出二个阶段(图 1)。A/D 转换是一个二次积分过程(信号积分 t_1 和消除积分 t_2), t_1 对于同一型号片子是定值, t_2 与输入信号大小成正比, A/D 转换结束进入数据输出阶段 t_0 , 由 D5~D1 口依次送出万、千、百、十、个位扫描信号, 由 $\overline{\text{STB}}$ 端输出 5 个位数字选通信号, 在每个位扫描信号或位选通信号期间, 该位的 BCD 码数据从脚 B₈、B₄、B₂ 和 B₁ 并行输出。 t_1 、 t_2 和 t_0 的大小常用各自所包含的时钟周期数 n_1 、 n_2 和 n_0 表示。

端口 R/H 是启动转换和停止转换控制端, 当 R/H 接 +5V 时, A/D 片处于连续转换状态, 此时 A/D 片的转换周期 t 为:

$$t = t_1 + t_2 + t_0 = (n_1 + n_2 + n_0) / f_{\text{CLK}} \quad (1)$$

当 A/D 片输入满量程时, $n_2 = n_{2\text{max}}$, $t = t_{\text{max}}$, 因此允许的最小采样周期 T_{min} 是:

$$T_{\text{min}} = t_{\text{max}} = (n_1 + n_{2\text{max}} + n_0) / f_{\text{CLK}} \quad (2)$$

已知 ICL7135 的 $n_1 = 10000$, $n_{2\text{max}} = 20000$, $n_0 = 1000$, 所以 7135 的 $T_{\text{min}} = 31000 / f_{\text{CLK}}$

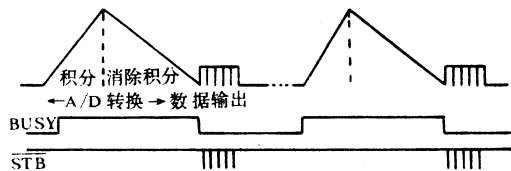


图1 A/D片工作过程

通常情况下, A/D 片与单片机接口需占用较多的单片机 I/O 口, 例如 ICL7135A/D 片与 8031 单片机相连, 仅数据线就占 9 个端口, 其中 4 个 BCD 码, 5 个位号口。在较大的应用系统中, 往往要在二者间扩展一片 8255 之类接口芯片以解决 8031 I/O 口数不足的矛盾, 这不仅使线路复杂而且增加了额外的开销。

1 方法一(图 2)

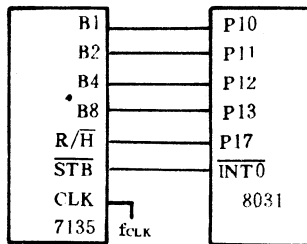


图2 接线图

不使用 D5~D1 位号线, 而用端口 R/H 和 $\overline{\text{STB}}$ 控制 5 位 BCD 码数据向单片机传送。

当 8031 的 P₁₇ 口控制 R/H 端使处于高电平时, 开始 A/D 转换, 转换完成后从 $\overline{\text{STB}}$ 端输出 5 个从万位到个位的位数字选通信号, 使 8031 产生 5 次 $\overline{\text{INT0}}$ 中断, 每次中断从端口 P₁₃~P₁₀ 读入相应位的 BCD 码数据, 在第 5 次中断接收到个位 BCD 码数据后, P₁₇ 输出低电平停止 A/D 转换, 直至 P₁₇ 由程序再次输出高电平开始一个新的采样周期。采样周期受 P₁₇ 口控制, 可由程序任意设置但不能小于 T_{min} 。

2 方法二(图 3)

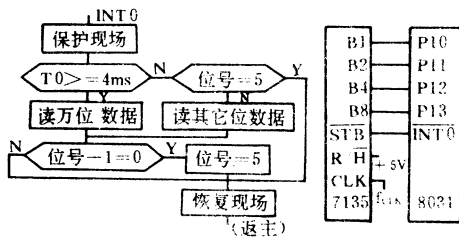


图3 接线及程序框图

将 R/H 端接 +5V 使 A/D 片处在连续转换状态。从图 1 可看出, 每组 $\overline{\text{STB}}$ 信号的第 1 个负脉冲与前后相邻负脉冲之间在时间上存在明显差异, 据此可由程序准确判断出每组 $\overline{\text{STB}}$ 信号中的第 1 个负脉冲的出现时刻, 进而相继读入 5 位 BCD 码数据。

例如当 $f_{\text{CLK}} = 100^{\text{K}} 100\text{KHz}$ 时, 组内负脉冲的间隔为 200 个时钟周期即 $2^{\mu\text{s}}$, 此时可取定时值 $T_0 = 4^{\mu\text{s}}$ 。在 $\overline{\text{STB}}$ 的作用下, 8031 每次产生中断时就判别当前负脉冲与上次负脉冲之间的间隔, 若大于 $4^{\mu\text{s}}$ 则为第 1 个负脉冲, 否则是第 2 至第 5 个负脉冲。

(上接 34 页)

```

if((filein=fopen(namein,"rt"))==NULL){
    cprintf("\nNO source file or source file error%s\n",
namein);
    exit(1);
}
gotoxy(10,13);
cprintf("please input target file name:");
gets(nameout);
if((fileout=fopen(nameout,"wb"))==NULL){
    cprintf("\ntarget file error: %s\n", nameout);
    exit(1);
}
gotoxy(10,15);
cprintf("pleasr waiting... \n");
textcolor(YELLOW);
gotoxy(10,9);
getdate(&today);
cprintf("Today's date is %02d-%02d-%02d", today.
da-mon, today. da-day, today. da-year);
gettime(&now);
cprintf("The time is %02d:%02d:%02d:%02d\n",
now. ti_hour, now. ti_min, now. ti_sec, now. ti_hund);
for(k=0; k<8; k++){
    chl=getc(filein);
}
while(! feof(filein)){

```

```

for(k=0; k<2; k++){
    chl=NULL; bitnum[k]=0;
    chl=getc(filein); bitnum[k]=atoi(&chl);
    if(chl=='A') bitnum[k]=10;
    if(chl=='B') bitnum[k]=11;
    if(chl=='C') bitnum[k]=12;
    if(chl=='D') bitnum[k]=13;
    if(chl=='E') bitnum[k]=14;
    if(chl=='F') bitnum[k]=15;
    kk=bitnum[0]*16+bitnum[1];
    fputc(kk, fileout);
}
close(filein);
close(fileout);
gotoxy(10,19);
getdate(&today);
printf("Today's date is %02d-%02d-%02d", today. da-
mon, today. da-day, today. da-year);
gettime(&now);
printf("The time is %02d:%02d:%02d:%02d\n", now.
ti-hour, now. ti-min, now. ti-sec, now. ti-hund);
textbackground(DARKGRAY);
textcolor(RED);
gotoxy(30,22);
cprintf("___ The end ___\n\n");
}

```

这种接口方法的采样周期 $T=t \cdot N$ 。其中 N 为正整数,可由程序设定。

3 方法三(图 4)

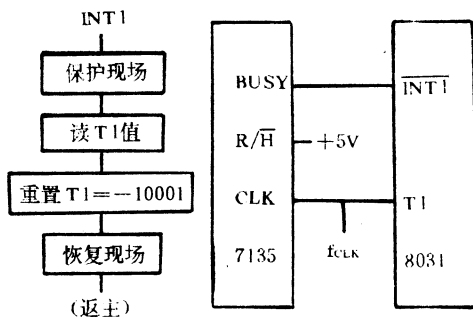


图 4 接线及程序框图

此法最简单,仅用 2 根连线,并且 A/D 片的数据输出不是从常规的输出端(B8~B1 和 D5~D1)输出,

因此本接口方法与双积分型 A/D 片的数据输出形式(BCD 码或二进制数)无关,适用于所有双积分型 A/D 片。

A/D 片有一个输出端 BUSY,它是表示 A/D 正在转换还是已转换完毕的信号(见图 1)。如果在 BUSY 信号处于高电平期间,用计数器对时钟脉冲计数,在计数结束后减去积分过程所含的时钟数就能获得消除积分期间的时钟数即 A/D 转换值。

实际应用时,考虑到 A/D 转换结束时的 1 个过零时钟,对于 ICL7135A/D 片其计数器的扣除值应为 10001。程序设置 8031 的 T1 为 16 位计数器,置 T1 的初值为 -10001,这样在 T1 终止计数时其值就自动少计 10001 个数,而不必再让 T1 计数值减去 10001。

当 BUSY 信号由低变高时 T1 开始对时钟计数,在 BUSY 从高变低时停止计数,同时利用 BUSY 信号的这个下降沿使 8031 产生 $\overline{\text{INT1}}$ 中断将 T1 中的 A/D 值读入。采样周期 $T=t \cdot N$, N 可由程序设定。在要求采样定时且能由程序任意设定的场合,可把 R/H 端接向 8031 的 1 个 I/O 口,但定时不能小于 T_{\min} 。

如何在应用程序中使用鼠标器

石家庄陆军参谋学院基础部(050084)黄建新 于长海

摘要 本文通过对比 Microsoft 公司提供的三种使用鼠标器的方法,认为使用中断的手段较适于应用程序的需要。并在本文中给出了使用 Turbo C 编制的两个使用这种方法时所必备的函数,同时本文还提醒了程序员在编程时应注意的几个问题。

关键词 鼠标器驱动程序 中断 中断向量地址 函数 文本方式 图形方式

鼠标器是一个向计算机传送信号的电子设备。自 Microsoft 公司 1983 年 6 月生产了第一个鼠标器发展至今,其共经历了三代的发展历史。目前在应用软件设计中使用鼠标器越来越多,这是因为它操作简单,灵活方便,易于控制,价格低廉,尤其在图形制作和菜单设计中有键盘无可比拟的优越性能。由于鼠标器具有如此之多的优点,使得许多程序员对它都十分青睐。但国内流行的各种语言版本都没有支持鼠标器工作的功能,为此给程序员的开发工作带来了诸多不便,为了解决这一问题,下面笔者给出一种解决方法。

鼠标器的使用是通过驱动程序的调用来完成。鼠标器的驱动程序解释鼠标器传送的数据并给操作系统提供统一的接口。Microsoft 鼠标器驱动程序为此提供了 35 个功能调用,使应用程序可以了解鼠标器当前的状态,并执行特定动作。例如,检查鼠标器按钮的状态,鼠标器光标位置,光标大小,色彩等。所以,我们在程序中对鼠标器进行控制,其实就是对其驱动程序加以访问。为此,Microsoft 公司提供了三种与鼠标器驱动程序通信的方法:鼠标器菜单,鼠标器函数库和直接调用 MS-DOS 的中断 33H。

鼠标器菜单方法主要适用于对初始设计时并未考虑到使用鼠标器的现有应用程序。

函数库的方法是使用软件包中提供的 MOUSE. LIB 函数库为 Microsoft 高级语言服务。这些库函数可以把鼠标器功能变为 PASCAL 语言中的过程, QUICK BASIC 和 FORTRAN 语言中的子程序,或者 C 和 QUICK C 中的函数。库函数所使用的参数是与所使用的语言一致的,对此,若要使用函数库,在应用程序设计语言中必须遵循 Microsoft 库函数的约定。如果某种语言支持这种约定,就可以使用函数库,否则,无法使用函数库来完成对鼠标器的控制。

使用中断 33H 的方法是当 MS-DOS 启动时,使用 AUTOEXEC. BAT 或 CONFIG. SYS 文件装入鼠标器驱动程序。该程序把自己作为操作系统的一部分,

并把入口地址作为中断向量设置到中断 33H 上,这样程序员就能用软中断 33H 访问鼠标器驱动程序。当软件执行到这一中断时, DOS 会在中断向量表中查到入口地址,使程序进入鼠标器驱动程序并执行相应操作。

通过分析这三种与鼠标器的通信方法,笔者认为若程序员在自己的程序中使用鼠标器,选择第三种方法,即中断 33H 较适宜。一则它执行速度快,因为直接调用中断会省去函数调用过程中的操作,二则是当无法和函数库相连接时,仍然能通过寄存器调用鼠标器功能。下面,给出使用 Turbo C V2.0 编制的调用中断 33H 的关键性两个函数: MOUSE() 和 CHECK—MOUSE()。

MOUSE() 主要进行 33H 中断各功能的调用, CHECK—MOUSE() 用于若调用成功,返回调用中断向量地址。

具有这两个函数后,程序员可在这两个函数基础之上,利用 33H 中断功能表(略),根据自己的情况编制一些功能函数,即可实现对鼠标器的操作。

两个函数清单如下:

函数 MOUSE() 清单:

```
void mouse(int * m1, int * m2, int * m3, int * m4)
{
    union REGS inregs, outregs;
    inregs.x.ax = * m1; inregs.x.bx = * m2;
    inregs.x.cx = * m3; inregs.x.dx = * m4;
    /* 将功能号 * m1 传递给 inregs.x.ax, 参数 *
    m2, * m3, * m4 分别传递给 BX, CX, DX, 以备 INT86
    () 使用 */
    int86(0x33, &inregs, &outregs);
    /* 使用 INREGS 的内容调用 DOS 的 33H 中断
    */
    * m1 = outregs.x.ax; * m2 = outregs.x.bx;
    * m3 = outregs.x.cx; * m4 = outregs.x.dx;
    /* 返回调用中断后的结果, 返回的 AX, BX, CX,
```


DX 的内容保留在 *m1, *m2, *m3, *m4 中 */

}

函数 CHECK_MOUSE()清单:

```
int check_mouse(int need_mouse)
```

/* 如果鼠标器驱动程序未正常安装

且 need_mouse 参数为 1, 返回 0, 并给出错误信息, 否则返回 1 */

{

void far *address;

/* 取鼠标器驱动程序中断向量地址 */

address=getvect(0x33);

/* 检查是否为空地址或 IRET 结构 */

if((address=NULL)||(* (unsigned char *)

address=0xcf))

{

if(need_mouse){

printf("Mouse driver NOT installed\n");

exit(1);

}

else return 0;

}

return 1;

}

给出了 MOUSE()和 CHECK_MOUSE()这两个关键性函数后, 程序员怎样使用呢? 这对用户来说是十分关心的。下面我介绍一下使用方法:

1) 对鼠标器进行初始化, 首先使用 CHECK_MOUSE()函数取出中断地址, 判断鼠标器驱动程序是否正常安装。若成功, 根据自己的设备情况(主要是 CRT 使用方式)判断应使用哪种方式初始化, 我们以 HERC 模式为例, 分析一下如何进行初始化过程。

①选择 CRT 显示方式

HERCULES 图形卡通过在文本方式下仿真 IBM 单色显示适配器工作。但它并不被认为是标准的 IBM 模式, 对于这种特殊方式, 仍有解决方法。BIOS 使用位于 0X0040; 0X0049 的一块特殊内存区域为它的显示例行程序存储各种各样的参数。该区域 (0X0040; 0X0049) 开始处的那一个用于存储当前显示状态, 当 HERCULES 处于文本状态时, 这里存放的为 7, 但不幸的是, 若你转到 HERCULES 图形方式时, 由于 IBM 并不认为 HERCULES 为其标准显示方式, 此时 0X0040; 0X0049 位仍为 7, 这样 BIOS 不会进入图形状态。为解决这一矛盾, 程序员可将此位调整为 6, 实现方法可在程序中设一状态标志, 以便使用时用程序修改, 从而保证 HERCULES 图形卡正常在图形方式下工作。

②初始化鼠标器光标

在检查完 CRT 方式并设定参数后, 下一步就是将鼠标器的光标进行初始化工作。对于图形方式, 图形窗口的左上角位置为 (0, 0), 这也是鼠标器驱动程序默认位置, 而对于文本方式, 其文本屏幕左上角起始位置为 (1, 1), 这样在选择使用方式后, 应给定确切的鼠标器位置参数。

③打开鼠标器光标

在方式及位置参数选定后, 即可打开鼠标器光标, 打开函数中可使用调用功能 1 来完成:

{

...

M1=1;

MOUSE(&m1, &m2, &m3, &m4);

...

}

2) 编制一些功能函数, 仅有本文给出的两个关键性函数并不能完全适应控制鼠标器的工作。程序员可利用 MOUSE()再编制几个实用函数, 诸如: 隐藏光标(调用功能 2), 取按钮压下状态(调用功能 5), 取按钮放开状态(调用功能 6)等。这些函数的实现方法十分简单, 程序员可自行编制完成。

3) 完成上述两项工作后, 程序员即可利用这些函数进行开发工作。为方便起见, 程序员可建立一个尽量多地包含鼠标器控制函数的工具库, 以便今后广泛使用。笔者提醒用户注意: 在使用工具库时, 需和调用该库的源程序一同编译, 否则无效。上面给出了在应用程序中程序员自行开发鼠标器的方法。在此, 笔者仍需提醒程序员在具体编程时注意三点:

①正确处理文本方式和图形方式下的鼠标器光标位置。

鼠标器驱动程序总是按像素方式返回坐标并始于 (0, 0) 位置, 而 Turbo C 从 (1, 1) 处开始其文本坐标, 这样在文本方式下使用鼠标器时, 对文本坐标位置就需作一调整, 以标准 80×25 模式来说, 8 个图形像素点对应一个文本单位, 这样鼠标器在文本方式下每移动一次, 坐标都应乘 8 再加 1。而对于图形方式鼠标器所在位置即为当前实际位置。

②鼠标器光标移动时应保持屏幕的完整性。

为防止鼠标器光标的移动影响原屏幕, 在每次光标移动时, 光标下面的文本或图形都应被存储, 并在光标离开该位置时恢复。使用这种方法来保持屏幕的完整性, 在绝大多数的情况下是正常的, 但用户仍需考虑到, 若屏幕正好是在光标位置上写东西时, 那么使用上述方法所保存的图像中仅为未写以前的信息, 这样待鼠标器光标移开后, 该位置将是写以前的内容, 从而影响了屏幕的完整性。为解决这种特殊情况, 建议最好是先将光标隐藏起来, 待写完后再将光标

提高静态存储器掉电保护可靠性的方法

哈尔滨工业大学计算中心 (150001) 宋颖慧 马 平 谢 生

摘要 工业控制计算机在电源启、停瞬间,数据总线、控制总线上的不稳定电位,可能破坏掉电保护 RAM 区中的数据。本文介绍一种可有效地消除这类现象,适用于工业控制机及单板机系统的方法,以提高数据保护的可靠性。

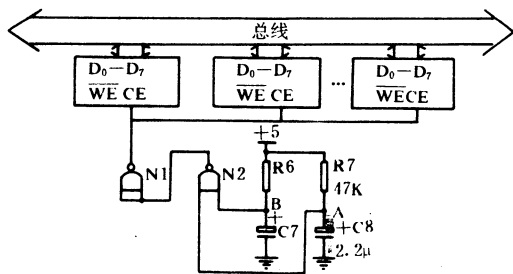
关键词 掉电保护 静态存储器 可靠性

1 问题的提出

目前在计算机监控或智能仪表系统中,大都设有静态存储器掉电保护电路,即在加电、断电瞬间,禁止存储器进行读写操作。使用中发现,一般存储器的掉电保护效果并不理想,反复加电、断电进行试验,则有可能改变一些单元中的内容。究其原因是在电源消失和恢复时间段内,总线电位处于不稳定状态,造成了对存储器的非正常写入,从而破坏随机选中单元中的内容,这在实用中有可能造成严重后果。

2 电路分析

为避免加电和掉电过程中 CPU 对数据存储器的非正常写入,简单而直接的办法就是在此时间段内,封锁数据存储器的写入允许端 \overline{WE} 或片选端 CE,图中给出在控制机或单板机系统中实现此功能的一个实例。



在电源加电过程中,输入端 A 点的电位遵循指数规律由 0v 上升,经 $t_A = 3\tau$ 时间达到 4.5v (其中 $\tau = RC$

为时间常数),合理选取图中 RC 值,如使 τ 为 0.1 秒,故在 t 时刻前,史密特触发器 N_2 的输出为高,再经过一个具有史密特触发输入端的反相器 N_1 ,输出一个低电平信号控制存储芯片的片选 CE,片选端 CE 是高电平有效信号端,所以在 t 时刻前存储芯片被禁止选中,这就消除了加电瞬间总线电位不稳定时对存储器的误操作,待经过 τ 时刻后,总线上电位稳定后,再对存储芯片操作。注意到 B 端的 t_B 稍小于 A 端的 t_A ,所以当 A 端达到 4.5v 时,B 端已达到这个值。此时 A 端起主导作用。

断电(或掉电)过程中,与加电过程原理类似,但此时是 B 端起作用,当电源电压跌落到 TTL 典型工作电压最小值 4.5v 以下时,B 点的电位几乎是瞬间由 4.5v 降到 0v (通过 C_7 、 R_6 回路放电),这里只要将 C_7 、 R_6 值取得很小,即让时间常数在 ns (纳秒)级,不超过存储器的一个读写周期,同时断电后由电池为史密特触发器供电,以使它能继续工作,经过 A 端“与上”B 端,史密特触发器经过两级输出一个低电平,使 CE 端不生效,那么在掉电过程中写操作对存储器就不起作用了。所以避免了掉电过程中的非正常写入。

3 结束语

本文所介绍的静态存储器掉电保护电路,意图在于给出一种方法,在实际应用中可以根据存储器芯片的型号不同,来调整 RC 回路中的 RC 值,以使时间常数满足一定条件,这样就能起到掉电保护的作用了。这个电路线路简单,元器件少,实用可靠,所以具有一定的推广应用价值。

收稿日期:1994 年元月 20 日

显示出来。这样就不会出现上述情况。

③程序中使用鼠标器时,必须确保在使用之前鼠标器的驱动程序已在 CONFIG.SYS 或 AUTOEXEC.BAT 中正常安装。以上仅给对使用鼠标器感兴趣的程序员提供了一种在自己的程序中使用鼠标器的方法,

并未给出所有使用函数内容。至于其它部分,相信各位程序员能顺利完成。

参考文献

- 1 陈辉东编译. Turbo C 高级程序员编程指南
- 2 许志平编译. Microsoft 鼠标器程序员参考手册
- 3 Microsoft 鼠标器技术手册

舍位平衡算法

广西计算中心(南宁 530022) 魏 鹰

摘要 本文介绍了报表处理中的舍位平衡问题,提出了解决舍位平衡的条件和算法,并讨论了平衡中出现的一些情况和问题,最后通过一个例子来说明平衡的过程和方法。

关键词 舍位平衡 平衡条件 算法

1 前言

在报表处理中,有时需要对数据进行舍位平衡。舍位,是将表格中的数据抹去尾数的一位或几位数,例如:将以元为单位的表格,改为(产生)以千元为单位的数据表格,就是舍去表格中数据的后三位。舍位平衡,是指在不破坏原有的数据平衡关系、不改变表格总数的前提下,对表格中的数据基本按照四舍五入的原则,进行舍位,调整数据。

对报表舍位后的平衡依据应该是反映报表关系的数据公式,根据这些关系来进行调整,使之平衡。

2 问题的提出

有一组数据:

a11	a12.....a1n
a21	a22.....a2n
.....
aml	am2.....amn

(数据式 1.1)

其中: a_{ij} 为实数($i=1, \dots, m, j=1, \dots, n$)

$$a_{1k} = \sum_{i=2}^m a_{ik} \quad (k=1, \dots, n)$$

$$a_{1l} = \sum_{j=2}^n a_{lj} \quad (l=1, \dots, m)$$

通过调整除 a_{11} 以外的其它任意数据项 a_{ij} (i, j 不同时为 1), 使得:

$$[a_{1k}] = \sum_{i=2}^m [a_{ik}] \quad (k=1, \dots, n)$$

$$[a_{1l}] = \sum_{j=2}^n [a_{lj}] \quad (l=1, \dots, m)$$

我们称这一问题为舍位平衡问题。

这里,调整的意思即为 ± 1 , $[a_{ij}]$ 表示数据 a_{ij} 四舍五入后的取整数。

这样的一组数据和关系就构成了报表中的平衡单元,我们称数据式 1.1 为一个平衡单元。一般地,在一张报表中,包含了许许多多这样的平衡单元,我们先来讨论一个平衡单元的平衡问题。

3 几个定义

①和向列:向列 $a_{11} a_{12} \dots a_{1n}, a_{11}, a_{21} \dots a_{m1}$ 称为和向列。

②子向列:向列 $a_{i1} a_{i2} \dots a_{in} \quad (i=2, \dots, m)$

$a_{1j} a_{2j} \dots a_{mj} \quad (j=2, \dots, n)$ 称为子向列。

③平衡差:

$$P_k = [a_{1k}] - \sum_{i=2}^m [a_{ik}] \quad (k=1, \dots, n)$$

$$P_l = [a_{1l}] - \sum_{j=2}^n [a_{lj}] \quad (l=1, \dots, m)$$

我们称 P_k, P_l 为平衡差,其中 P_k 为纵向平衡差, P_l 为横向平衡差。

4 平衡条件

能使任意两个不平衡的向列(平衡差均不为 0), 调整为平衡的充分条件有:

- ①两个不同方向的子向列的平衡差符号相同;
- ②两个相同方向的子向列的平衡差符号相异;
- ③和向列与不同方向的子向列的平衡差符号相异;
- ④和向列与相同方向的子向列的平衡差符号相同;
- ⑤两个和向列的平衡差符号相同。

我们称以上条件为平衡条件。

由此可见,要想使一个平衡单元上的数据平衡,就要找出符合平衡条件的两个不平衡向列来进行调整。通常,在一个平衡单元内,会发生同时满足多个条件的情况,这就要逐个条件的分别进行调整。

猜想:一般地,在一个平衡单元中,只要有一个向列发生不平衡(平衡差不为 0),那么,必然存在有另一个向列也不平衡,而且这两个向列满足平衡条件。

要证明这一猜想的正确性(形成定理)是十分困难的,这就有待于读者来完成了。

5 平衡算法

我们只要针对以上 5 个平衡条件,分别进行相应的调整,即可形成平衡算法。

先来定义调整值:

$$D = \frac{\text{平衡差}}{|\text{平衡差}|} \quad (\text{即} \pm 1)$$

对应于 5 个平衡条件的平衡算法如下:

I. 在两个子向列的交叉项上,加上 D ;

II. 取其中一子向列(任意的)的 D 值,在这两个子向列的非合计项上,并行查找不为 0 的两个数,且这两

个数所在的子向列(另一方向)上的平衡差为 0, 对其中一个数(取 D 值的子向列上的那个)加 D, 对另一个数减 D;

Ⅲ. 在两个向列的交叉项上, 加上和向列的 D 值;

Ⅳ. 在这两个向列的非合计项上, 并行查找不为 0 的两个数, 且这两个数所在的子向列(另一方向)上的平衡差为 0, 然后对这两个数分别加上 D;

V. 找到一个不在和向列上的任意不为 0 的数, 且这个数对应交叉位置的两个子向列上的平衡差为 0, 然后对这个数以及对应交叉的两个子向列上的合计项(两个), 分别加上 D。

在以上的平衡算法中, 每执行一次某个算法作业后, 其两个向列仍然满足相应的平衡条件(平衡差仍均不为 0), 这时, 应再执行一次同样的算法作业, 一直到这两个向列不满足相应的平衡条件为止。

平衡算法的优先级为:

I → II → III → IV → V

6 平衡的几个问题讨论

6.1 双向、单向平衡

上述所讨论的是有两个和向列的平衡问题, 即双向平衡。还有另一种情况, 就是只有一个和向列的平衡问题, 即单向平衡, 其平衡算法是较为简单的, 因为在作单向平衡时, 不用考虑对另一方向上的向列的影响。

对单向平衡的平衡算法为:

第一步: 在向列上的非合计项中, 找到一个不为 0 的数;

第二步: 将这个数减去 D, 向列上的平衡差减一次 D;

第三步: 如果平衡差仍不为 0, 继续找下一个不为 0 的数, 返回第二步, 否则结束。

6.2 双向平衡中的单向平衡处理

有时会出现这样的情况: 数据式 1.1 中, 有个别子向列不属于某一向的和关系式中, 同时也不属于其他任何平衡单元的同方向的和关系式, 但这些向列本身又满足另一方向的和关系式, 即这些向列对一向是独立的, 而对另一向则是非独立的。

对这类向列的平衡调整, 实际上已经演变为单向平衡问题。

通常在进行双向平衡时, 所有涉及到的向列(报表中即行、栏), 我们都给一个标志, 那些没有给过标志的, 便是这类单向平衡处理的向列。

6.3 平衡优先层级

上面所述的是一个平衡单元的平衡问题, 对于多个平衡单元, 而且具有嵌套的层级关系的平衡问题, 又应当怎样处理呢?

我们知道, 在对一个单元平衡时, 有一个基本原则, 就是总合计数(即数据式 1.1 中的 all)是不能动

的, 但可以调整其他数。如果是由内层往外层作平衡, 那么, 内层平衡了, 而内层的和向列又要作为外层的子向列, 在对外层平衡时, 可能会调整到这些子向列(内层中的和向列), 这样又破坏了内层原来的平衡。

因此, 平衡的优先层级为: 由外层往内层。

6.4 关系式中的减运算处理

上面我们讨论了和运算(+)的平衡问题, 即和向列等于子向列之和, 至于关系式中出现减运算的情况, 我们可以将减运算化为加运算, 办法是将被减的向列乘上一个因子 -1, 待平衡处理后再还原回来(再乘因子 -1)。

7 对一个单元的平衡过程举例

我们可以通过以下的一个平衡例子, 来说明怎样对一个单元的平衡过程。

有一个数据单元:

行栏 →	(1)	(2)	(3)	(4)
↓				
(1)	217	65	97	55
(2)	142	26	74	42
(3)	75	39	23	13

其中: 行 1 = 行 2 + 行 3

栏 1 = 栏 2 + 栏 3 + 栏 4

现在要将这组数据舍 1 位, 化整后结果如下:

平衡差 →		+1	+1	
↓				
-1	22	7	10	6
	14	3	7	4
+1	8	4	2	1

平衡步骤:

①用平衡算法 I, 将第 3 行、第 3 列交叉上的数 2 调整(加 1)为 3;

②用平衡算法 III, 将第 1 行、第 4 列交叉上的数 6 调整(减 1)为 5。

这样, 平衡后的结果为:

22	7	10	5
14	3	7	4
8	4	3	1

微机硬磁盘自举功能恢复一例

重庆工业管理学院管理工程系(630050) 周 宏

1 故障机型

DEC 386

2 故障现象

系统启动,进行到 A 驱动器指示灯亮时即死机,指示灯一直不灭。

3 故障检查

- 1). 从软盘启动机器并进入 C 盘;
- 2). 用查病毒软件查病毒,未发现;
- 3). 用 PCTOOLS 查两个系统隐含文件 IBMBIO.COM 和 IBMDOS.COM,以及命令文件 COMMAND.COM,均存在,且字节数正常;三个文件名依次排列在其它文件之前,文件名的位置正常;

4). 查 CONFIG.SYS 和 AUTOEXEC.BAT 的内容,均正确。

至此,并未查明故障原因,但是故障又确实存在。

4 试排除故障

1). 估计硬盘的引导信息有错误。把事先保存的该硬盘的引导信息拷贝回硬盘引导区,重新启动机器,故障如初;

2). 估计系统隐含文件 IBMBIO.COM 和 IBMDOS.COM 或命令文件 COMMAND.COM 有错误。删去以上三个文件,重新传送正常的系统隐含文件、拷贝命令文件,再次启动机器,故障仍旧不变。

5 再查故障

用 PCTOOLS 的 E 命令查系统隐含文件 IBMBIO.COM 在硬磁盘上的存储位置,发现磁盘数据区的第一扇区并不是被 IBMBIO.COM 所占据,从该扇区开始存有另一文件,这种存储位置安排是系统所不允许的。

6 故障排除

1). 把从磁盘数据区第一扇区开始存储的文件拷贝走,然后删除根目录中的该文件,给 IBMBIO.COM 留出空间;

2). 用 PCTOOLS 把磁盘中原来存储位置不对的系统隐含文件 IBMBIO.COM 和 IBMDOS.COM 删除;

3). 向硬盘重新传送系统文件;

4). 再启动机器,发现启动完全正常,故障已排除。

此例中,系统隐含文件 IBMBIO.COM 和 IBMDOS.COM 存储位置错位的原因不清,但说明当硬磁盘不能自举时,操作者需要考虑到这种情况。

硬磁盘不能自举是微机使用者很容易遇到的,由较多的流动人员共同使用的微机,例如学校的教学用机,硬磁盘不能自举更是常有的事。

硬磁盘不能自举的原因很多,例如计算机病毒、磁盘数据错误、磁盘文件错误等;有时候也可能是由于磁盘的物理损坏,但是这种可能性很小。当硬磁盘不能自举时,只要方法得当,一般都可以方便地恢复磁盘的自举功能。

收稿日期:1993年10月6日

开创影像通讯时代 实现“天涯咫尺”梦想

远隔重洋的有情人也能天天见面,这在不久以前还仅仅是梦想而已,而今宏基电脑公司开发成功的视讯电脑使这一梦想成为现实。

以制造个人电脑而闻名全球的台湾宏基(Acer)集团,积极向尖端科技迈进,将当前最热门的电脑科技——视讯会议技术成功的应用在个人电脑上,让身处异地的人可以透过电脑荧幕,面对面交谈或举行国际会议。同时宏基的产品并将以平价走入寻常百姓家。

宏基产品兼具影像电话、电话答录机、传真机和视讯会议的多元功能,是目前 PC 视讯产品中功能最齐全的,而价格却与目前未具备如此多功能的产品不相上下,同时极具市场竞争力,市场前景极为乐观。设想,不久的将来,人类必将进入“影像通讯时代”,世界将掀起电讯事业的改革。

华基电脑公司是宏基公司在大陆的总供货商,是宏基产品在大陆有力的推广者,为宏基产品在大陆的推广普及做出了重要的贡献,如今又将成为宏基的视讯会议技术——“多媒体视通信系统”走入中国大陆市场的先驱者!

(北京市华基电脑公司总经理 温金流)

未来 CAD 展望

四川泸州长江液压件厂情报室(646006) 胡德森

当今的设计人员往往面临着许多相互矛盾的要求。他们要在更短的时间内完成更多工作,要在规定的时间内生产出 100%合格的产品。而消费者的要求越来越苛刻,产品成本必须降低。此外,设计人员仍然想要愉快地度周末!

每当面临这诸多的考虑时,许多设计人员深深感到,提高个人工作效率,不仅对公司竞争生存有利,而且对他们自己也十分重要。

尽管“工作”一词的概念需要重新定义,但设计人员的工作范围已大大扩大,现在已包括了理论设计、应力分析、详尽的技术文件编制等。

新一代 CAD 技术也不是灵丹妙药。现在 CAD 技术可更有效地对付上述提及的工作。

今天,在大功率 RISC 基工作站上工作的 CAD 软件,如 HP700 系列可帮助处理和管理设计数据。

新一代 CAD 应是什么样的呢?

1 CAD 的关键:柔性

从理论上讲,新型软件和硬件工具必须具备极大的柔性,以帮助设计人员根据愿望校正、修改设计,在不再输入数据、重新了解软件应用包或进行常规的系统执行及文件编制的情况下在更多方面帮助设计工作。

最新的 CAD 软、硬件生产商意识到这些需要,并已作出了相应的反映。设计软件使整个设计过程自动化,帮助管理设计数据,支持参数设计方法。

应用这些先进的软件,加拿大一家航空元件生产商从基本的制造方法迈向了新的设计理论,用软件进行应力分析并设计一个产品系。这些工作都可用先进的集成式软件工具实现。

然而,高技术软件不能独立存在。较大功率的工作站可提供更快的 CPU 处理速度,计算功率及更好的图形性能,满足这些智能的、具有特色的设计软件包的需要。

任何 CAD 技术是否被认可都依赖于整个系统的响应时间。设计的部分工作是同时进行的。如果在输入每种指令后,系统停顿,它就会破坏其固有协调性,不管它有多么复杂,或技术有多先进。一种响应时间太慢

而不能满足日益增长需要的软、硬件是不会被商业界广泛接受的。现在的 CAD 工作站 90 年代的计算机速度要快 10—15 倍。

2 强有力的参数化设计工具

更大柔性、更快的修改周期推动了参数化设计工具的开发步伐。参数化设计工具是指可利用以前设计中积累的参数或法则的智能软件。这些参数化软件应用贮存的法则帮助设计人员进行修改工作。据称,应用参数化工程的 ProEngineer 软件包设计全系列工业动力产品时,一家电子生产厂的生产周期更短,再设计工作量更小。

这种软件的一大问题是:智能设计软件与更大功率工作站嫁接在一起的本质是什么?

其部分作用是增添设计人员的能力,进一步使现行的绘图程序自动化。计算机集成制造(CIM)系统可用于参数收集、计算,甚至设计制造程序本身。

用于创造生产,新型网络允许世界各地的设计人员同时对一项目攻关。应用 HP 共用型 X 窗口,各地的设计人员可在计算机上展示各自的设计。当他们认为有必要作出修改时,可用电话进行讨论。这一程序完全象两个人在图板前一样,仅仅是更为舒服而已!

在不久的将来,设计人员将在计算机设计中加进声音信号。当接收者启动计算机时,计算机便会讲道:“喂,某某人,请检查一下,特别应注意某某部分。”

计算机也能在口语指令的基础上应用声音识别技术进行绘图。比如,只需说出“请画一下油缸,其缸径 2 英寸,长 4 英寸,垂直于 X 轴……”,计算机就会绘出图形。

此外,设计人员可环视设计的结构或“进入”他设计的发动机。虽然现存的软件具有这一功能,而实际的微处理机速度太慢,使得这一工作又不够真实。只有到了 1995 年,当 BIP(每秒亿万指令)的微处理机问世后,CAD 才会真正成为强有力的设计工具。

未来的 CAD 软件将在设计过程的更多方面提供帮助。应用它,设计人员可大大改进产品质量,提高生产率,降低成本,又可轻松愉快地度周末!

从 Pascal 到 C 的自动转换

四川大学计算机科学系 (成都 610041) 周 韬 魏志毅

Pascal 语言作为一种结构化的程序设计语言因为其结构严谨、功能较强曾经流行一时,它被用来在许多领域编写了大量的程序,但随着 C 语言的出现和发展, Pascal 的地位受到了极大的冲击。C 语言是一种结构化、模块化、可编译的通用程序设计语言,它的强大功能、高质量的代码灵活性、和良好的可移植性,使得 C 语言被广泛地应用于系统程序和应用程序的开发。因此把 Pascal 书写的大量源程序这一宝贵资源自动地转换为 C 源程序是一项十分有意义的工作。本文提供了 Pas_To_C 这一工具。它能自动地完成从 Pascal 到 C 的移植的 70% 的工作量。程序已在 TC、BC 环境下在 386、486 上调试通过,运行正常,并在国家自然科学基金资助项目 HBase 的移植中应用,经受住了考验。

移植的要点在于:

- 1) 充分利用两个语言的相似性。
- 2) 对其不同之处进行自动转换。

当然,被移植的 Pascal 源程序首先应该是正确的。

程序的主过程是 Do_PasToC。在其中 235 行开始分别对需要转换的不同情况进行处理。其中有对注释、串、指针、赋值号、等号、不等号的处理,在 default 中还有对函数和过程的处理,其中比较重要的部分是:

1 保留字的转换

从程序的 57 行开始是保留字的转换过程。在其中的保留字表中列出了 Pascal 的所有保留字以及与之相对应的 C 的保留字。为了便于二分查找,我们对保留字进行了按字母排序。当程序处理到一个词时,就在保留字表中搜索,找到后就转换为相对应的 C 的保留字,否则照原样输出。

2 处理变量说明

从程序的 108 行开始是变量说明的转换过程 Do_

Declare。Pascal 中的变量说明的形式是变量在前,说明在后,例如: a:integer; b:real; c:char; 而 C 语言的变量说明却正好相反,如: int a; float b; char c; 因此在转换过程中,我们设置了一个栈,把读到的变量压栈,等到读出变量类型并进行转换后,再将变量弹出。

函数及过程中的参数说明的转换是类似的。是由 Do_Procedure 及 Do_Function 完成的。

3 函数返回值的类型说明的处理

在 Do_Function 函数中,我们把除了类型说明的整个函数头作为一个变量名,并调用 VarChg 来处理函数的类型说明前置问题,然后在处理函数内部的参数说明。从程序 170 行开始的函数对此作了处理。

4 函数及过程嵌套的处理

Pascal 中允许在函数和过程中再定义嵌套的函数和过程,而 C 却不允许,这就需要将其中的函数和过程抽出,可这样会很麻烦,因此为了不使程序过大,我们只是在有函数和过程嵌套的地方标出了嵌套的层数,以便于以后的修改。

此外,因为 Pascal 中不区分大小写,而 C 中却区分,所以在程序中的 GetWord 函数中在取出单词时统一改成小写。

本程序可以完成转换的 70% 以上(在未做转换的地方做出了标记)。由于 C 语言毕竟比 Pascal 技高一筹的语言,有些移植工作涉及到语义,因此只有利用知识库技术,才能完成 90% 以上的自动移植,这已超出了本文的范围。

由于对篇幅的限制,程序略,对次要地方(如交互、容错、界面等)作了一些压缩和精简,但同时留下了扩展的线索和基础,例如读者可在函数 OpenFile 中加入交互及容错代码,等等,使之成为一个更好的实用程序。需程序清单者,可同笔者联系。

在本文的撰写过程中得到了唐常杰教授的指点,特以致谢。

收稿日期:1993 年 11 月 3 日

好消息

* 本刊 93 年精装合订本现已正式装帧成套,封面烫金,适宜图书室、资料室、档案室上架存档,亦适宜个人珍藏备用,每套 25 元。

* 《计算机应用研究》1993 年专辑第 2 号(即“第五届全国计算机应用研究学术交流会议论文集”)现已出版问世,全书共 118 篇新颖实用的计算机学术论文,250 页,16 开本,彩色封面,70 克胶版纸印刷精致,欢迎订阅,每册 18 元,需要以上资料者,可直接寄款至本刊编辑部。

《计算机应用研究》编辑部

APPLICATION RESEARCH OF COMPUTERS

Vol. 11 No. 3(Total 59)

CONTENTS

SURVEY

A New Computer Application Architecture Client/server	Li Wen hua et al. (1)
Data Communication on Microcomputers	Fang You liang(4)
Look Forward to the Future CAD	Hu De sen (58)

RESEARCH

A Study on Computer Recognizable Code Marked by Hand	Wu Qing xiang(6)
Handwritten Numeral Recognition by Fuzzy Associative Memory Neural Networks	He Fen dao(8)
The Characteristics of Matrix Algorithm about IC Minimal Complete Test Set and Realizing Its Program Language	Zheng Xiao wei(11)
Technology of Fast Displaying on Image Rotating	Li Wei qing(13)
Simply Analyzing Principle of Drive Program on DOS	Zhou Yong li et al. (17)
Maximum Model of Calculating Joint Profit on Markets of Microcomputers	Zhang Hong ming et al. (21)

APPLICATION

Numbers of Equality of Goldbach's Conjecture—Programming Exercise on Software Standard for Examination	Luo Hai peng(27)
How to Get the Space Place of Collagen Fibers in Aorta of Rat from 2—D Biomedical Image	Gao Yun qin(19)
A Intellectual Control Card for Stepping Motor	Sun Chuan zheng(24)
A Bi—terminal Display Controller for Different Visions	Zhou Li et al. (14)
GAL Device Application In Design of the intelligent Instrument	Zhou Ming zheng(28)
An Advertisement Screen with Light Emitting Diodes	Wang Shui qing(30)
Display of VGA 256 Colours	Li Long shu (31)
The Methods and Implement to Protect MIS from Exit Accidently	Cai Qing hua et al. (32)
Solidifying Program on 51 Single Chip Computer By EPROM Card	Wang Jing yuan (34)
The Text Print Routine of Laser Printer	Chen De ming(35)
How to Replace Communication and Print Card with Multifunctional Super Card	Ye Fu le (36)
Special Technique of EGA/VGA/TVGA Screen	Chi Kun(37)
Design Gist of Player Routine	Wang Rang ding et al. (39)
The Design of A System for Foxpro Pull Down Menu	Shi Yong xin(42)
A Method of Realizing Vector Chinese Character	Shan Hong et al. (44)
The Design of Master —slave Distributed Monitoring System	Yu Li jian Wang Chang lin(46)
To Use WPS Only by Two Low Density Disk of 360K	Zhu Guang zhong(48)
All Interface of Dual Integration ADC to Single Chip Computer	Sun Qing liang Sun Jian guo (50)
How to Use Mouse Mark Device in Application Program	Huang Jian xin et al. (52)
Original Technology of Cartoon—Colour Flowing Method	Fu Yun qing(7)
Truncation Balance Algorithm	Wei Ying (55)
Automatically Changing from Pascal to C	Zhou Tao et al. (59)

MAINTENANCE

An Example about Maintenancing Abnormal Fault on CTX—3 Monochromatic Display	Wang Jian zhong et al. (10)
Analyses on Two Typical Problems of Printers	Rao Zeng ren(38)
Detecting and Debugging Floppy Disk Driver by DDD Disk	Liang Xue yu et al. (41)
“Stick on ”Writing Protection for Hard Disk	Dai Shui gui(49)
Method of Raising Reliability of Break down Electricity Protection on Static Memory	Song Ying hui et al. (54)
An Example about Restoring Bootstrap Function of Hard Disk on Microcomputer	Zhou Hong (57)

INFORMATION

News in Brief & Advertising	(Cover2,Cover3,57)
Notice of Our Magazing	(26,36)
News in Brief	(10,59)
Message of New Technology and New Products	

新技术、新产品荟萃

1. 博大(BD)系列电脑电视字幕机——用于电视录像及广告制作

BD 系列电脑电视字幕机由微计算机、标准 PAL 内外同步器、彩底发生器、PAL 彩色合成器、视频合成器、高精密中西文字库及字幕、图形编辑、创作软件组成。主要功能有：文字字幕特辑、动画/图形创作、数字图像处理特技、外设备同步、特辑文件存储等功能。主要型号如下：

● BD-1：同屏 16 色 11 矢量字库 286 主机 高分彩显

● BD-2：同屏 256 色 11 矢量字库 三维动画 386 主机(带协处理器) 高分彩显

● BD-3：同屏 32768 色 11 种矢量字库 386 主机 高分彩显 视频质量达广播级标准

● BD-4：同屏 32768 色 11 种矢量字库 数字特技 三维动画有渐变色 486 主机 高分彩显 视频质量达广播级标准

本系列产品荣获 1988 年度四川省科技进步二等奖。产品遍布全国，已推广千余台。

2. SMD-6420 三维制鞋辅助测量辅助设计系统——用于各种鞋类设计、样片制造及生产管理

本系统是制鞋业计算机辅助制鞋面测量和辅助制鞋设计而研制开发的系统。系统提供的辅助测量及辅助设计功能覆盖了制鞋设计全过程各个阶段，能满足皮鞋、布鞋、胶鞋、旅游鞋等各类男、女、童鞋设计需要，使用该系统设计准确度高、快捷高效、简便易学，是目前国内功能最完整、最实用、性能价格比最优的一套系统。该系统已在数百家制鞋企业中推广应用。

硬件：主机：名牌 386 微型计算机 外设：三维制鞋面坐标测量仪、1280×1024 高分辨率图形显示器、TMS34X 系列图形加速卡 A3 幅面平面数字化仪、A3 或更大幅面 XY 绘图仪。

软件：全套软件包括：辅助制鞋面测量、全鞋面展平、平面、立体帮、底样设计及帮、底样片扩缩、单位优化排料、面积测量计算、样片切割、制鞋档案资料管理和设计数据管理等功能。

该项目已获 90 年度四川省科技进步二等奖，92 年起被列为国家重点推广项目。

3. MD 系列机器设备状态监测与故障诊断系统

MD 系列状态监测与故障诊断系统以国内外这一领域最新理论、技术及方法为基础，应用当代先进的 ASIC 技术、网络技术的优势，率先在全国推出最高水平、最优性价比的 MD 系列产品。该系列产品其性价比优于进口产品，系替代进口的最佳选择。主要用于工业生产工艺流程中的各关键设备(机组)的运行监测、故障预报、故障分析诊断和指导维修，进行单机及多机组同时在线监测。该系统是保证生产安全运行，减少维修费用，提高经济效益的不可缺少的手段。本系统已在航空燃气轮机研究中心及四川化工厂等重要现场中运行。

92 年 12 月，本系统获得四川省重大科技成果三等奖。93 年列为四川省重点科技推广项目。

● MD 3905 系列机器设备状态监测与故障诊断网络系统——用于工业生产流程中各关键设备(机组)的运

行监测、故障预报、分析诊断和指导维修，实行多机组同时在线监测。

● MD3902 系列机器设备状态监测与故障诊断系统——用途同于 MD3905、3906 系统，区别在于适用于单个机组的在线监测，如企业自备电厂汽轮机系统。

● MD8500 系列便携式数据采集/分析仪——现场数据采集及预处理。适用于设备的定期巡检、重点监测及现场事故处理使用。

● MD9500 系列便携式旋转机械故障诊断仪——用于化工、机械、冶金、电力、轻工等工业现场对旋转机械进行现场监测和现场分析。

4. BMP-C 系列彩色脑地形图处理系统——用于诊断脑部疾病

BMP-C 型彩色脑地形图处理系统是诊断脑部疾病和研究大脑神经活动的重要仪器。它利用人体自发产生的脑电波，经频谱分析处理为直观的地形图，反映出病灶在脑部的范围和位置。我们在总结分析几十万临床病例的基础上，率先在国内开发出 BMP-C 型脑地形图处理系统和性能优异的 STAR9316 16 道脑电信号采集器，并可配置成为一套完整的一体化系统。BMP-C 型对于脑肿瘤、癫痫、脑炎、脑外伤、脑血管、脑萎缩等颅脑疾病的准确率达 70% 至 92%。尤其是独特的癫痫分析模式的研制成功，标志着 BEAM 技术进入新的里程。同时 BMP-C 型带有性能优良的十六道脑电信号放大器和大容量记录仪，独家推出脑电全息处理模式，成为当今最为理想的 EEG/BEAM 系统。本系统现已在全国推广近 500 套。

5. CBM 系列心电体表电位标测系统——无创性心电检查，早期诊断心脏疾病。

体表电位标测是一种新型的无创性心电检查方法，它通过安放在人体躯干上的 64 个电极获得心动周期中各瞬间身体表面心电电位的空间分布，经计算机处理，以体表等电位图等多种形式表达心脏各部分兴奋与恢复的过程，它反映的是心脏电生理活动的空间变化规律，提供的信息比传统的心电图更多、更全面，因而更能客观准确地反映心脏的电生理活动。

该系统图象清晰、失真小、重复性好、操作简便，在诊断心肌梗塞、预激综合症、束支阻滞、心室肥大等病方面比常规心电图灵敏、准确性高，为心脏疾病的早期发现、定位诊断提供了先进设备。本产品已于 92 年通过省级技术鉴定，并以最优的性能价格比(价格仅为国外同类产品的二十分之一)，在国内数家著名大医院获得推广应用。

6. KRL 矿热炉微机控制系统——用于黄磷、铁合金、电石行业的矿热炉控制

矿热炉是广泛应用于化工、冶金、有色金属行业的重要冶炼设备，也是能耗大户。KRL-1 矿热炉微机控制系统就是为降低设备的能耗，提高经济效益而研制的高科技节能产品，该系统用于黄磷、铁合金、电石冶炼和工业硅过程控制，投资少，见效快，其可靠性大大超过同类产品。绵竹黄磷厂统计，使用该系统后可增产黄磷 6.3%，节电 7.86%，电极消耗下降 6.1%，焦炭消耗下降 9.98%，年直接经济效益在 25 万元以上，使用后三个月至半年即可收回投资。现在该系统已经在数台台矿热炉上成功地运行。92 年

已被列入四川省重点推广项目。

7. WCY 型电网调度微机远动系统——用于地方电网的监测、调度指挥

WCY 微机远动系统由调度机与厂站端机构成。以 1:N 的循环调度模式,实现对各分散厂站端的遥测、通信、遥控三遥功能。整个系统具有统一的时钟,组合数台电视形成大屏幕代替老式模拟屏,用高精度电压、电流可变量取代传统的变送器,系统可靠性高、运营简便,保证地方电网有效合理地分配利用能源。本产品适用于地县级电网调度自动化及大厂矿电力管理。

8. DZJ-5G 电能综合计量仪——电压的监测、电能分时计量

专门用于电能的分时计量、电压缺相检测、电压合格率统计计算。它是电能分时段计量、监测电能质量的专用智能仪表。经省级技术鉴定和中国测试技术研究院测试,各项指标均属国内先进水平。可广泛应用于发电厂、供电局变电站、电力用户、小水(火)电、电力公司,是电力现代化管理最理想的仪器,已在全国推广 300 余台。

9. 电子防盗、电视监控报警保安系统

本系统由闭路电视监视、电子防盗报警、内部有线对

讲、进出门管理四个子系统组成。该综合系统将报警、监视、指挥、录像功能融为一体,根据各种对象,不同规模和独特的系统要求。可以选择单个子系统独立使用,或者组成综合系统。使用可靠,功能丰富,性能价格比优。本系统广泛应用于银行、宾馆、车站、机场、大型综合商厦、交通枢纽、博物馆、展览馆、停车场、重要库房、军事设施、各企事业单位财务室、档案室以及别墅、普通家庭等。

10. DNX 型电脑成像机——用于旅游点电脑画像

DNX 电脑成像机通过了四川省省级鉴定,获四川省科技进步三等奖。此机广泛适用于旅游点、宾馆、游乐场、科技馆、照像馆等为人画像。

该机由电脑主机、摄像头、快速高密度打印机、监视器组成,经电脑处理,可在 35 秒左右打印出像片。这种像片既有一般照片的逼真性,又有素描画像的艺术性,别具风格,可永久保存。若备专用色带和调温熨斗,还可将像片转印到混纺织物旅游纪念品上,深受旅游部门和群众的欢迎。

电脑成像机是在一般白纸上印象,取材就近便利,且结构简单,操作方便直观,一般人经过半小时学习可熟练掌握。

对上述系列产品及项目感兴趣的单位及个人,欢迎来人来函洽谈,欢迎实地考察,具体洽谈事宜请同本刊编辑部张钢、哲明同志联系。(联系地址:成都人民南路四段 11 号附 1 号,邮码 610041)

《计算机应用研究》杂志办刊单位

中国计算机学会
四川省电子计算机应用研究中心
贵州省科委计算中心
安徽省计算中心
吉林省计算中心
内蒙古电子计算中心
云南省电子计算中心

《计算机应用研究》杂志社董事会

董事长:周赛渝

董事:管小宝 张国栋 闫长荣 王升尧
李天健 郑国基 叶大卫 乔中南
冯德成 朱 华 王俊杰

新疆电子计算中心
甘肃省计算中心
广西计算中心
山东省计算中心
河南省计算中心
青海省测试计算中心
武汉工学院

《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 罗海鹏
张湘金 倪凯民 范德元 李文华
王义章 崔振远 刘铁军 毕常青
刘启茂

1994 年 3 期(总第 59 期) 出版日期:1994 年 5 月 责任编辑:唐大利

计算机应用研究(双月刊)

JI SUAN JI YING YONG YAN JIU

(公开发行)

刊号: CN51-1196/TP (国内)
ISSN 1001-3695 (国际)

邮发代号: 62-68 (国内)
BM4408 (国外)

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

电 话: (028) —5582666 转 2055

印 刷: 成都市新都华兴印务有限公司

订 阅 处: 全国各地邮政局

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 050 号

每册定价: 2.00 元