



中国计算机学会会刊

ISSN 1001-3691

计算机应用研究

APPLICATION RESEARCH OF COMPUTERS

1997 • 5



美丽的蝴蝶

GBH多媒体查询系统制作工具

- 面向对象多媒体创作
- 制作方便快捷



ISSN 1001-3695



9 771001 369007



广西计算中心海蓝集团

地址: 广西南宁市星湖路32号 邮编: 530022

电话: (0771) 5851945 5860210 5851627

传真: 5852684



计算机应用研究

APPLICATION RESEARCH OF COMPUTERS

1997

GBH多媒体演讲辅助系统

● 交互式多媒体讲稿排版

● 关键屏任意跳转

● 三种自动演示方式



ISSN 1001-3695



09>

9 771001 369007



广西计算中心海蓝

地址：广西南宁市星湖路32号 邮编

电话：(0771) 5851945 5860210

传真：5852684

甘肃省计算中心在改革开放中阔步前进

甘肃省计算中心建立于一九七九年,是全省计算机应用、推广和研究的重要中心。全中心现有职工 68 人,其中专业技术人员 46 人,占全体职工的百分之六十七,其中,高级技术职称人员 9 人,中级技术人员 28 人。该中心设有七个专业研究室和部门:计算机应用研究室、计算机软件开发研究室、计算机控制研究室、CAD 研究室、运行部、维修部、培训部,该中心位于交通便利、位置显要的兰州市东方红广场。

在国家科委、省政府、省科委的关怀和领导下,在兄弟省市同行单位的大力支持下,甘肃省计算中心日益发展,不断成长,十几年来为国民经济发展做出了重要贡献,在科学计算、过程控制、办公室自动化、企业管理、计算机辅助设计、图象处理、网络系统集成等方面取得了一定成就,为省内外推广应用项目 130 多项,其中经省级鉴定的项目 40 多项,有 8 项获省科技进步奖。出版发行专业性著作 7 本,在国内外杂志上发表论文 60 余篇。

在八十年代,甘肃省计算中心紧密结合甘肃省国民经济发展的特点,积极开展计算机应用研究、推广和计算服务工作,为省内外科研、企事业单位提供了大量计算机机时,直接为科研、生产服务,服务范围遍及农业、水利、气象、电力、机械、建筑、地质、冰川、沙漠、航空航天、铁道、医学、教育、化工、光学仪器、核物理、轻重工业等各个领域,有力地支持了科研和工农业生产,促进了甘肃省国民经济的发展。与此同时,还承担或与有关单位合作完成了《三角点成果存储与检索系统》、《计算机终端通过市话线路的联机数据通讯》、《计算机藏文文字处理系统》(获省科技进步二等奖)、《卷烟生产微机监控管理系统》、《高层框架——剪力墙空间计算程序》、《电视机厂微机企业管理系统》、《大跨度反吊桥研究》、《被动式太阳房热工计算软件》、《甘肃省森林资源连续清查微机应用系统》、《核物理实验多参数采集系统数据识别和转换》等 42 项科研课题。《高层框架剪力墙空间计算程序》是当时国内首次编写的高层框架——剪力墙空间协同计算程序,经几十家建筑单位采用,取得了良好的经济效益。《甘肃省科技普查及其数据处理》项目 1988 年在国家科技普查评比会上荣获国家科技领导小组颁发的二等奖。

八十年代后期和进入九十年代之后,随着国家经济政策的调整,除了完成上级下达的工作任务外,甘肃省计算中心在注意搞好基础建设的同时,承揽了科研和生产中一些实用性科研与工程项目,其中有兰州亚欧商厦(微机网络及 POS 系统)工程项目,目前正在收尾阶段,它的完成将为兰州市利用计算机技术进行商业管理提供有益的尝试。96 年承担的省科委《甘肃省科技信息网络》项目,是利用先进的网络系统集成技术开发的,目前正在积极完成此重大科技任务。已完成的项目还有《啤酒发酵滤酒微机控制系统》(获省科技进步二等奖)、《商场微机应用电视监控系统》、《微机遥感图像处理》、《体视学分析与图像转换研究》、《铁路给排水计算机辅助设计》。正在实施的项目还有:《甘肃省机动车辆及驾驶员图文网络数据库管理系统》、《计算机辅助生产系统在建筑工程中的应用》、《室外给排水工程计算机辅助设计》等等。

在学术方面,出版的专业性著作有《舍入误差分析引论》、《科学技术统计》、《银行计算机》、《西北地区星火计划发展的若干数学模型研究》等七篇。在国内外省级以上学术刊物上发表论文 60 余篇。这些论文的发表,促进了专业技术人员业务水平的不断提高,稳定和壮大了技术队伍。

“雄关漫道真如铁,而今迈步从头越。”现甘肃省计算中心在研究员李天健主任主持下,正乘深入改革开放的东风,发挥自身优势,借助联合办刊的群体优势,以“团结奋进求实创新”的精神,沿着建设有中国特色的社会主义道路,阔步前进!(甘肃省计算中心联系电话:0931—8823594)

国防科技大学可靠性研究中心隆重推出

通用化工程化故障模式、影响及危害性分析软件系统—GeFMECA

“故障模式、影响及危害性分析(Failure Mode Effect & Criticality Analysis—FMECA)”是一种系统可靠性安全性维修性分析重要而有效的手段,它是分析产品中每一潜在的故障模式并确定其对产品所产生的影响,以及把每一个潜在故障模式按严重程度及其发生概率予以分类的技术。它由GB7826 及GJB1391 所定义,是装备研制与生产的可靠性通用大纲规定的主要工作项目。

国防科技大学可靠性研究中心一直致力于可靠性、安全性CAD 软件的研制,所开发的故障树分析软件系统FTAS 与FMECA 软件在第二届全国可靠性软件展示会上获得优秀软件奖,受到广大用户的欢迎。1997 年在原来FMECA 软件基础上,又开发推出了基于Windows95 平台的全新通用化工程化故障模式、影响及危害性分析软件系统GeFMECA。

系 统 特 点

- 实现了基于系统结构层次的FMECA 可视化
- 建立了各种装备类型系统结构层次的代码
- 建立了多种装备类型通用和专用的故障模式库
- 实现了系统各层次产品的FMEA 自动迭代
- 提供了与电子产品可靠性预计软件的接口
- 提供了与FTAS 软件的互联接口
- 建立了故障模式追加接口
- 提供了一个图形文本编辑、播放3D 和运行外挂的实用环境交互工具

欢迎查询、使用

联系人:周经伦 孙 权

联系电话:(0731)4555601 转7711

联系地址:湖南长沙国防科技大学七系

传 真:(0731)4510434

中国计算机用户协会信息系统分会 第九届信息技术交流大会征文启事

为促进我国计算机事业的发展,加强信息技术交流,中国计算机用户协会信息系统分会拟于1998 年8 月在浙江温州召开第九届信息系统分会信息交流大会。

为开好这次大会,现组织会议征文,有关事项如下:

一、征文范围:

1. MIS 的开发方法与工具。
2. 政府、城市企事业单位MIS。
3. 商业企业计算机管理信息系统(MIS)的开发与应用。
4. 数据库的应用和开发。
5. 计算机网络系统的开发与应用。
6. EDI/VAN 的建设及应用。
7. 有关Internet 网的开发与应用。
8. 多媒体技术在信息工程中的应用。
9. 人工智能、专家系统和决策支持系统。
10. 系统集成。
11. 软件工程和项目管理。
12. “三金工程”开发与应用技术。
13. 面向对象技术和CASE 工具。
14. 计算机安全管理、病毒防治的理论与实践经验。
15. 计算机在辅助设计、制造、测试、教学等方面的应用。

二、征文要求:

1. 论文全文不超过4000 字,全文包括论文题目、作者姓名、单位名称、邮码、摘要(不超过200 字)、关键词,主体包含图表(文中插图需一式两份)及参考文献。
2. 来稿必须字迹清楚或寄打印稿,文责自负。
3. 来稿务请挂号邮寄,征文恕不退稿。
4. 征文截止日期:1998 年3 月31 日,以邮戳日期为准。

三、征文联系人:

乐嘉敏 联系电话:(010)64010628 或(010)64018833 转302

地 址:北京市东黄城根北街16 号

科学出版社

邮编:100717

目次

综述评论

建设更安全的 Internet	龚波	郑若忠(1)
面向对象的三维图形软件包——Open Inventor	左旭	卫原平 阮雪榆(4)
WINDOWS95 的网络功能设置分析——新一代网络操作系统实践经验浅述	曹进	(6)
WWW 的几项关键技术浅析	陈琳	黄筌(8)

研究探讨

经典 Ramsey 数 $R(5,14)$ 的下界	罗海鹏	苏文龙	张正铀(10)
若干个经典 Ramsey 数 $R(5,q)$ 的新下界	苏文龙	罗海鹏	张正铀(11)
WINDOWS 消息驱动机制中的核心技术分析	杨亮	阮晓星	魏晋鹏(12)
EIS 需求分析的困难及方法	刘伟	杜昌勇	李朋(14)
汽车零件失效分析的知识模型及描述方法	杨开英	毕常青	张明友 陈兰英(18)
任意具有 $Z=F(X,Y)$ 形式空间曲面的生成	周术诚	王宜怀	梁忠(20)
用 JavaScript 改变 Internet 浏览器的界面			杨建华(22)
不精确推理模型和算法的描述及应用探讨			吴陈(24)
iRMX for Windows 与 Windows 的动态数据交换技术			王明春(27)
静探曲线数据处理中的计算机采样技术	徐斌		唐建东(29)
校园网计费管理的分析与研究	陈品德	龚正虎	汪钟鸣(32)
分布式集中供热微机监测系统	冯永祥	崔新民	贾颖力(35)
Sybase 数据库与 Web 的互连			邵立兵(38)
自由曲面视觉测量中一种有效的粗差别除方法	陈捷湘	罗飞路	朱震辉(40)

应用开发

PC 机与家电联网的集散系统	胡劲松	吴捷(42)
一种新的消息传递型多处理机系统中总线接口的设计	屈玉贵	赵静 张海勤(44)
非网络环境下的计算机通信		祝雪妹(48)
用单片机实现与 I ² C 总线器件的接口	孟维晓	徐玉滨 王斌(49)
ATM NIC(网络接口卡)的设计与开发	许冬	郑王利(51)
计算机在砷内缺超声探测中的应用	卢结成	吴龙标 王立敏 丁晓兵(53)
利用消息实现 Win32 进程间数据传递的方法		张丹宇(56)
图库——CAD 成图率上升的途径之一		刘希宏(59)
DBF 文件到电子表格的简捷转换	何其才	杜建丽(60)
语音交流电压表的设计		田辉(61)
SURFER 软件二进制绘图命令及应用	申时新	张志兵 谭昌勇 申时凯(63)
AIX 逻辑卷管理应用于建立 CC—MAIL 邮局		王蕾(66)
24LC32 串行 E ² PROM 在 96 单片机系统中的应用	王辉	马洪江(68)
便携式条形码识别存储器	许伟	王谦(71)
VESA 图象编程及其应用	魏庆国	黎鹰 吴建华(73)
虚拟视景仿真中的 LOD 及 mipmapping 技术		陈刚(75)
利用 Video Blaster SE100 实现连续图象捕捉与实时压缩	曹宁	吴琰(77)

维护维修

系统级故障诊断方法研究	杨金民	张大方(79)
如何防止公共机房中 CMOS 口令被篡改所造成的“死机”现象		李朔枫(82)
DEBUG 程序在微机故障诊断和排除中的应用	杨胜勇	李劲松 肖滨(83)
微型计算机开关电源疑难故障的检修	王维舟	张生金(87)
Word 打印的几个技巧		马军(89)

动态简讯

广告、简讯、启事	(3,5,10,26,31,34,62,90)
----------------	-------------------------

Soul Journal of Chinese Science & Technology Articles

APPLICATION RESEARCH OF COMPUTERS

VOL. 14 NO. 5(Total 79)

CONTENTS

SURVEY

- Building a More Secure Internet Gong Bo et al. (1)
Three-dimensional Figures Software Package of the Object-Oriented-Open Inventor Zuo Xu et al. (4)
Analysis of Installing a Network Function of WINDOWS 95 Cao Jin(6)
Simply Analysing a Few Items Key Technology of WWW Chen Lin et al. (8)

RESEARCH

- A Lower Bound of Classical Ramsey Number $R(5,14)$ Luo Haipeng et al. (10)
New Lower Bounds of Some Classical Ramsey Numbers $R(5,q)$ Su Wenlong et al. (11)
Analysis of Key Technology in WINDOWS Information Driving Mechanism Yang Liang et al. (12)
The Difficulties and Solutions of EIS Requirements Analysis Liu Wei et al. (14)
Intellectual Model of Analysing Spares Lose Efficacy of Automobile and Describe Method Yang Kaiying et al. (18)
Drawing a Surface with all Style of $Z=F(X,Y)$ Zhou Shucheng et al. (20)
Using Java Script to Change Web Brower's User Interface Yang Jianhua(22)
Approximately Reasoning Models Wu Chen(24)
A Communication Technology Between iRMX for Windows and Windows Wang Mingchun(27)
Computer Sampling Technology in Data Management for Static Cone Penetration Tests Curves Xu Bin et al. (29)
Analyses and Research on Accounting Management of Compus Network Chen Pinde et al. (32)
Monitoring System of Distributed Central Supplying Heat on Microcomputer Feng Yongxiang et al. (35)
Linking up Sybase Database and Web Shao Libing(38)
An Effective Method of Rejecting Rough Difference in Measuring Free Curve Visual Chen Dixiang et al. (40)

APPLICATION & DEVELOPMENT

- The Network of PC and Household Appliances Hu Jingsong et al. (42)
A New Kind of Design of Bus Interface in the Message-Passing Multiprocessor System Qu Yugui et al. (44)
Computer Communication under Environment of Non-Network Zhu Xuemei(48)
The Implementation of the Interface Between I²C Bus Devices and Single-chip Computer Meng Weixiao et al. (49)
Design and Implement of ATM NIC (Network Interface Card) Xu Dong et al. (51)
Computer Application in the Ultrasonic Test of Interior Flaw of Concrete Lu Jiecheng et al. (53)
The Method of Using Message for Data Transferring Between Win32 Process Zhang Danyu(56)
Image Storage—One of the Ways of Gking up Imagery Ratio of CAD Liu Xihong(59)
Forthright Changing DBF Files to Electronic Forms He Qicai et al. (60)
The Design of Speech A.C. Voltmeter Tian Hui(61)
Binary Drawing Command of SURFER Software and It's Application Shen Shixin et al. (63)
Applying AIX Logical Volume Management to Building CC-MAIL Post Office Wang Lei(66)
Application of Serial E²PROM24LC32 in 96 Single-chip Computer System Wang Hui et al. (68)
The Portable Memory of Identification Bar Code Xu Wei et al. (71)
Programming of VESA Images and it's Application Wei Qingguo et al. (73)
The Techniques LOD and Mipmapping in Virtual Visual View Chen Gang(75)
Realizing Continuous Catching Images and Real-time Compress by Video Blaster SE100 Cao Ning et al. (77)

MAINTENANCE

- Studying the Method of Diagnosing System Stage Faults Yang Jinmin et al. (79)
How Prevent Appearance "Dead Computer" as a Result of CMOS Command Distorted in Public Engine Room Li Shuofeng(82)
Application of DEBUG Program in Diagnosing and Removing Soft Fault on Microcomputers Yang Shengyong et al. (83)
Examining and Repairing Difficult and Complicated Faults of Switch Mains on Microcomputers Wang Weizhou et al. (87)
A Few Techniques of Printing Word Ma Jun(89)

INFORMATION

- Notice 7 Advertising (3,5,10,26,31,34,62,90)

建设更安全的 Internet

龚 波 郑若忠

(国防科技大学计算机学院 长沙 410073)

摘 要 Internet 上的安全机制很不完善,这种缺陷阻碍了商业活动在 Internet 上的发展。该文着眼于 IETF 在 Internet 加密安全方面的工作,分别在:IP 协议层、路由机制、DNS、网络管理和密钥管理等五个领域阐述面临的攻击,现在所采用技术和技术发展方向等。

关键词 IPv6 DNS SNMP 算法独立协议

1 引言

Internet 可以提供快捷的远程计算、资源共享和电子数据传输等服务,极大地推动社会的发展,但由于 Internet 安全机制不完善,商业和政府机构对于使用 Internet 进行重要活动犹豫不决。随着公司或企业单位致力于防火墙和企业网(Intranet)的建设,Internet 作为全球网的概念会逐渐衍变为到处是层层壁垒的私有子网部分互联的组合。在过去几年中,Internet 工程任务组(Internet Engineering Task Force)在 Internet 的基础结构中加入加密安全保护协议,为信息提供一定的保护手段,但安全漏洞依然存在。

本文在 Internet 基础结构的五个领域描述 IETF 在加密安全方面所作的努力。对于每个领域,分别阐述安全攻击、现时技术和可能的技术发展方向。当然,安全技术发展进程也受到非技术方面影响,如因商业利益的相悖导致新技术的流产或推迟使用(如第三方托管技术,数字签名标准 DSS 等)。可以相信,未来的 Internet 会更强大和安全,但现在不得不为抵抗越来越高明的攻击而开发新的安全技术。

2 IP 协议层

现在 IP 层主要使用 IPv4 协议,IPv4 没有安全设施,数据的保密性、完整性和系统的可用性经常受到威胁。IETF 现在正在开发 IPv6 协议,来替代 IPv4。IP 协议层不仅包含 IP,而且包含 ICMP(Internet Control Message Protocol)和 IGMP(Internet Group Membership Protocol)。ICMP 是差错与控制报文,IGMP 是网间网组管理协议,可以提供多目组的操作等信息。

图 1 为:协议图描述重要的 Internet 协议间的关系。

2.1 安全攻击

IP 协议易于在以下几方面受到攻击,比如,攻击者可以使用:

· ICMP 重定向信息来破坏路由机制和提高侦听

业务流的能力。

· ICMP 不可达信息来实现对点对点应用的拒绝服务。

· IGMP 加入/离开多目组信息实现对多用户的拒绝服务。

· IP 地址是软件可配置的,攻击者可以窃取、构造 IP 地址或 TCP 三握手协议的序列号,产生地址假冒和地址欺骗问题。

· 攻击者也可以在 IPv4 网上使用地址解析协议,协同代理 ARP 和逆向 ARP,来窃取业务流或者实现对单或多点的拒绝服务。

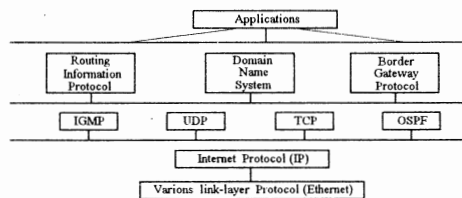


图 1

2.2 现在技术状况

加密安全在 IP 协议层应用有限,美国国防部 DOD 在 1980 年曾资助 IP 协议层加密协议 SP3D,但只限军事级网络加密设备。近年来,IETF 设计两种 IP 协议层加密安全机制。使用这两种机制,IP 报文减小了安全威胁,但增加了负载,减小了有限带宽和端——端吞吐率。第一种机制是 AH(Authentication Header)提供加密认证和完整性保护,但不保证保密性。由于 AH 不符合政府对保密性规定,可以被广泛应用。第二种机制是 ESP(Encapsulating Security Payload)是由 SP3D 发展而来。ESP 支持保密性和完整性。ESP 和其它加密技术协同使用,可以实现保密、验证和完整性。AH 和 ESP 是独立于算法的(见附录 A),可以和任何加密算法共同使用。两种机制不能保护 ARP,是源于两机制是基于 IP 报文,而 ARP 不使用 IP。但 ARP 不会给 IPv6 带来安全麻烦,因为 IPv6 的设计者采用 Neighbour Discovery,一种基于 ICMPv6 的协议来代

替 ARP。

2.3 未来方向

防火墙是用于内部网和全局 Internet 间的设备,对通过内部网边界的报文,实现分组过滤或地址过滤,以减少外部站点对内部站点的攻击。防火墙提供商已将 AH 和 ESP 结合进以 IP 为基础的产品中。未来大多数防火墙还会支持对 Internet 业务流的加密,也可能代表内部结点提供给外部站点密钥管理和安全服务的功能。加密防火墙甚至可以为未实现 AH 和 ESP 机制的系统提供加密保护。

在以后几年中,IP 协议层加密将被广泛使用。而且使用独立于算法的机制或协议如 AH 和 ESP,也会被广泛开发和应用。最后,现在大多数的 IP 安全产品使用软件实现加密,未来产品会趋向于硬件实现以提高性能。

3 路由机制

Internet 使用动态路由机制,每个路由器根据协议独立决定报文的转发方向。这种特性使系统容错性、抗恶劣环境能力大大增强。路由器间连接信息交换依据路由协议,域内路由协议包括 RIP(Routing Information Protocol)和 OSPF(Open Shortest Path First),域间路由协议有 BGP(Border Gateway Protocol)。

3.1 安全攻击

在过去几年中,对路由机制的攻击变得愈加频繁,因为攻击者认识到攻击路由机制比攻击单个系统更有利。可以产生侦听,拒绝服务等。如 Syn Flooding 攻击潮,就是攻击路由机制实现阻止合法用户和服务器的正常连接。大多数路由协议使用明文信息通讯,因为 BGP 通讯使用 TCP 协议实现路由器间的数据传输,则基于 TCP 的攻击很易影响 BGP 安全。

3.2 现在技术状况

IETF 最近推出一种适用于 RIP 和 OSPF 消息的独立于算法的加密验证方案。该方案使用安全散列函数 MD5 来验证路由报文的发送方,可以防止大多数的攻击,但对重播攻击(Replay attack)束手无策。

Cisco 路由器包含一种机制来实现对 TCP 会话的验证,减少了通过 BGP 域间路由信息攻击的可能性。

3.3 未来方向

IPv4 域内路由协议现在支持基于 MD5 的加密验证方案的使用,该方案是独立于算法的,一旦发现 MD5 的安全漏洞,则可轻易地使用更安全的算法来代替。

IPv6 实现 AH 和 ESP 机制,域内路由协议依赖 AH 和 ESP 实现安全保护。IDRP(Inter-Domain Routing Protocol)将取代 BGP,IDRP 中也包含加密验证功能。

使用 BGP 或 OSPF 需许多人工配置,配置密钥相对增加的工作量不算多。然而 RIPv2 不需很多人工

配置,所以密钥管理工作相对任务量较重。

4 域名系统

DNS 提供主机名到 IP 地址的映射和与远程系统互连的功能等。DNS 有一个层次化的 DNS 数据库服务器和一个授权系统。DNS 服务器设置是冗条的,以增强系统的容错性。

4.1 安全攻击

DNS 协议缺乏加密验证机制,易发生替代者攻击(man-in-the-middle-attack),即攻击者伪装成其它合法系统和本系统直接相连,而本系统不鉴别相连系统的真实性,通讯时会丢失机密信息。区域 DNS 信息传输可以帮助攻击者认清感兴趣的目标。比如,许多防火墙过滤外部访问标准依赖非验证 IP 地址,则如在 DNS 系统中加入错误的 DNS 信息后果是很严重的。Stevens 在 1989 年指出:DNS 和路由机制攻击联合起来,会产生灾难性的结果。

4.2 现在技术状况

Donald Eastlake 和 Charles Kaufman 最近开发一种扩展安全功能来增强 DNS 的安全性。扩展功能实现对公开密钥的证明机制,提供对 DNS 信息的验证功能。IETF 现在正对该功能进行标准化的工作。

Eastlake 和 Kaufman 的 DNS 扩展安全功能以地址、姓名、邮箱号等私有信息进行验证公开密钥的真实性。密钥管理协议也要求能验证密钥交换双方的真实性,存在 DNS 中的密钥证明有助于完成这项工作。

实施 DNS 信息的验证可以大大减少对 DNS 和使用 DNS 的系统的攻击。提供验证公开密钥的能力也有助于实现动态密钥管理,具有扩展安全功能的 DNS 在 2001 年会广泛使用。

5 网络管理

几年前,Internet 组织采用 SNMP(Simple Network Management Protocol)该协议有助于管理 Internet 和分析操作行为。SNMPv1 现已经开发商用产品,显著提高网络的免疫力。

SNMP 因为不只使用于 IPv4 之上,而且可以应用于如 NOVELL'S Internetwork Packet Exchange(IPX)和 ATM,所以应实现自己的安全机制,才能有较好的移植性。

5.1 安全攻击

SNMPv1 不使用验证或使用明文可重用口令(如组织名)来验证和认证信息,因此对 SNMP 业务流侦听,进而实现对 SNMP 数据的非法访问是较容易的。

一旦攻击者访问 SNMP 数据库,则可以实现多方面攻击。如安全和关信息、口令字等,如果攻击者对路由表有写权利,则可破坏路由协议,实现拒绝服务等。

5.2 现在技术状态及未来方向

现在许多 SNMPv1 实现不允许可以对 SNMP 数据库进行改变的操作,如写、添加等。该限制增强了

SNMP 安全性,但也限制了灵活性。

1993 年, IETF 第一次颁布了 SNMPv2, 最初说明包含对 SNMP 消息的 MD5 验证的强制实现, 这较 SNMPv1 是个进步。但也有人认为 SNMPv2 的安全模式太复杂, 1995 年将该机制从 SNMPv2 中移动, 现在, 一个讨论会正在分析 SNMP 并建议怎样改进来处理安全问题。IETF 在 1997 年很可能开始对 SNMPv2 加密安全的标准化工作。

6 密钥管理

密钥管理是加密体制中难解决的问题, 缺乏动态密钥管理的标准化协议是 Internet 大量应用加密机制的最大障碍。根据现代密码学, “一切秘密存于密钥之中”。密码体制的安全依赖于密钥的安全。OSI 安全结构中, 定义密钥管理的内容为密钥的产生、存储、分发、删除、获取和应用(ISO7498—2)。

6.1 现在技术

从根本上讲, Internet 现在主要有两种网络密钥管理技术: KDC(Key Distribution Center)和 Diffie—Hellman 方法, KDC 方法类似 Kerberos 验证系统使用的方法, 即可信第三方(Trusted third Party)。可信第三方验证通讯双方的真实性, 产生站点间会话密钥, 并通过数字签名等手段安全地分配密钥。Diffie—Hellman 不需 KDC, 通讯发起方产生站点间会话的秘密密钥材料, 并且通过数字签名或零知识证明等方式安全传递通讯密钥。

IETF 现在正开发 Okaley, 一种基于 Diffie—Hellman 技术的密钥交换协议。DNS 服务器存放公开密钥和公开密钥证明, Okaley 可以实现数字签名、有效防止对原型 Diffie—Hellman 的拦截者(man—in—the—middle)攻击等。

Okaley 还提供重要的安全特性: 密钥的独立性, 即使用该协议产生的密钥和以前产生的密钥是相互独立的, 所以攻击者不能因攻破主密钥就很易得到对话

密钥。IETF 正在开发一个补充的协议 Internet Security Association and key Management Protocol, 可以补充其它安全特性, 如密钥生存期、敏感级别、时戳等。

6.2 未来方向

Obkley 和 ISAKMP 的联合在以后会广泛使用, 联合协议不但可以管理和 Internet 协议层有关安全问题而且可以管理与 RIPv2、OSPFv2、传输层、以至更低层如 ATM 的有关安全问题。

多目密钥分配是需深入研究的领域, IETF 建议使用以下几种方案: 一种方案是在 ISAKMP 密钥管理框架中加入 KDC 的思想分配密钥管理安全问题; 另一种方案是类似的 Kerberos 的层次密钥管理中心 KDC。

7 结论

Internet 安全已引起人们的足够重视, 这种重视会促进技术的进步, 产生一个更加安全的 Internet, 适合于商业活动和政府组织的广泛使用。

附录 A:

算法独立的安全机制

IETF 正在开发的许多安全协议和安全扩展功能趋向于算法独立性思想是, 即使加密算法改变, 基本机制可以保持不变, 如 ESP 和 AH。算法独立性较易实现算法间转化, 一旦发现原来算法存在安全漏洞, 就可以使用更安全的算法来代替, 而机制修改工作较少。这也提供给用户更广泛的选择。现在, 许多安全机制依赖 DES 和 MD5 算法。这种状况正逐渐得到改善。

算法独立性思想在将来会广泛使用。

参考文献

- 1 [R J. A.] Randall J. Atkinson "Toward a More Secure Internet" IEEE Computer Magazine Jan. 1997
- 2 Computer Society: Inprocessings of the 1992 — 1995symposium on Security and Privacy

要想应用研究计算机 请订《计算机应用研究》

《计算机应用研究》系国内外公开发行的技术双月刊, 国家科委系统 14 省市计算中心等单位联合办刊, 中国科技论文统计源期刊, 中国计算机学会会刊, 1981 年元月创刊, 屡次评为优秀科技期刊。

刊物宗旨与内容: 普及计算机应用知识 促进计算机学术交流 传播计算机应用信息
拓展计算机应用领域 推广计算机应用成果 提高计算机应用效益

订阅代号: 62—68(在全国各地邮局订阅)

发行刊号: ISSN1001—3695(国际) CN51—1196/TP(国内)

订阅价格: 全年六期每份人民币 30.00 元(国内)

通讯地址: 成都市人民南路 4 段 11—1 号 邮政编号: 610041

电话号码: (028)5210177 5220666—2055 图文传真: (028)5221946

面向对象的三维图形软件包——Open Inventor

左 旭 卫原平 阮雪榆

(上海交通大学国家模具 CAD 工程研究中心 200030)

摘 要 本文介绍了面向对象的三维图形软件包——Open Inventor 的特点和主要结构。本文还详细讲述了 Open Inventor 和 Open GL 的区别与联系。最后,本文给出了用 Open Inventor 制作的两个实例。

关键词 Open Inventor Open GL 面向对象

1 Open Inventor 简介

Open Inventor 是由 SGI 公司推出的面向对象的三维图形软件包,它是一种相对独立的窗口系统,它可以通过支持不同的窗口系统移植到不同的硬件平台上。它采用“搭积木”的方式来构造复杂的三维场景。也就是说,Open Inventor 预先为用户提供了很多基本的三维图形的元素和对三维图形对象操作的方法,此外,Open Inventor 还为每一个图形窗口提供了统一的界面,用户无需关心界面制作,用 Open Inventor 创建的每一个三维图形对象都是可旋转、平移、放大/缩小或动画的,这种“搭积木”的方式可以使用户花费很少的时间就可构造复杂、优美的三维场景和最大限度地发挥硬件的效能。

Open Inventor 由于采用了面向对象思想,它创建的图形对象不同于用传统方法(如用 C 语言或 GL)产生的图形。用传统方法产生的图形和图形的操作之间是没有联系或联系很弱的,因此,用传统的方法产生复杂的三维图形并实现复杂的图形操作是十分繁琐的。然而,Open Inventor 把它创建的图形对象和对这些图形对象的操作“捆绑”在一起,从而使三维图形的创建与操作变得简单而有趣。Open Inventor 所创建的对象的所有信息,如:对象的位置、形状、大小、色彩、表现纹理、光源等都储存在 Open Inventor 的场景数据库中,用户可读取或显示这些信息。Open Inventor 把图形对象的信息和对图形对象的操作封装在一起。这样,用户很容易对创建的图形对象实现改变颜色、大小、纹理、移动位置、变换视角、动画、鼠标选取,高亮度显示,包围盒计算,搜索等一系列操作。

Open Inventor 不但是一个面向对象的高效的三维图形制作系统,它还提供了多种文件格式接口的方法。用户可以用这些方法读入或输出图形对象,用户甚至可用窗口系统中的剪贴板把 Open Inventor 中的图像剪切或粘贴。Open Inventor 既可采用 C++ 编程,也可采用 C 语言编程。

2 Open Inventor 和 Open GL 的区别与联系

Open GL 是在 SGI 公司的三维图形库 GL 的基础上建立的、支持不同的硬件平台(包括微机系统)的开放三维图形库,是一种事实上的工业标准。Open Inventor 通过它自身的场景数据库调用 Open GL 实现三维图形对象的显示。Open GL 直接把显示的对象送入到帧缓冲器中,而 Open Inventor 则把需要显示的对象先存入 Open Inventor 内部的数据库中,并把对象和操作封装在一起,当 Open Inventor 数据库的显示操作被激活时,Open Inventor 才把要显示的对象送入到帧缓冲器中进行显示;在 Open GL 中,对象的显示和对对象的操作是分开的,而 Open Inventor 把对象及其操作封装在一起,用 Open GL 显示对象,用户必须掌握很多编程细节,诸如对象的选取、旋转、平移等操作,用户必须用一系列的 GL 编程语句来实现,而 Open Inventor 预先提供了一系列的标准组件和操作,用户只需把这些组件组合起来即可,无需了解太多的细节。当然,Open Inventor 也保持了很多 Open GL 的灵活之处,使用户可以构造自己所需要的。Open Inventor 未提供的对象、组件及其操作。好比制造一台机器,用户既可以自己制造每一个零件(Open GL 的方式),也可先买一些标准件再自己制造一些特殊零件(Open Inventor 的方式)。

总之,本文认为,应该根据用户自身的实际需求来决定怎样组合使用 Open GL 与 Open Inventor,把 Open Inventor 提供的对象组件和 Open GL 提供的具体函数组合在一起使用,从而为用户节省大量的时间。

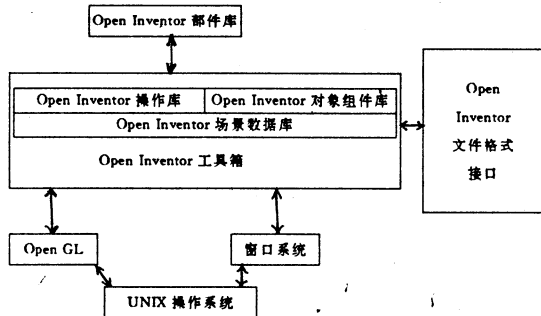


图 1 Open Inventor 的主要结构

3 Open Inventor 的主要结构

Open Inventor 建立在 UNIX 操作系统和 Open GL 图形库之上,它调用 Open GL 实现三维图形对象的显示并为 Open GL 提供了一系列的标准界面。Open Inventor 包括三大部分:Open Inventor 工具箱、Open Inventor 部件库和 Open Inventor 文件格式接口。而 Open Inventor 工具箱又包括对象组件库、操作库和场景数据库。Open Inventor 的主要结构如图 1 所示。

3.1 Open Inventor 工具箱

Open Inventor 工具箱是 Open Inventor 的核心,它不但为用户提供强大的编程应用接口,还管理 Open Inventor 创建的每个对象。它包括三部分:

a, 场景数据库

在场景数据库中,数据库元是最基本的一系列的对象,它们是用来创建三维场景数据库的最基本的“积木块”。每个数据库元包括一系列的重要信息如:形状描述,材料种类,几何变换和光源等。数据库元按照一定的规则组合起来就得到了复杂的三维场景图。场景图被储存在场景数据库中。场景数据库由 Open Inventor 管理。一旦用户生成了一幅场景图,用户就可以对场景图的对象进行操作,这些操作对象的管理也是由场景数据库来实现的。

在 Open Inventor 中,数据库元主要包括以下几种:“形状元”(如球、圆锥、圆柱、长方体、NURBS 曲面等)、“特性元”(如光照环境、光源种类、材料种类等)、“图形引擎”(用来联系对象和有关的场景并对场景产生动画)、“传感器”(用来对场景数据库中的数据改变作出响应,以调用相关的操作和实现时间进程管理)。

b, 对象组件库

对象组件库提供了许多 Open Inventor 预先规定的对象组建和搭配的机制。用户也可以根据自己的需要加入自己的对象搭配机制,加入时需要创建一个模板文件。Open Inventor 将根据这个模板文件来实现用户所规定的搭配机制。

c, 操作库

操作库主要响应用户的交互操作事件,这种响应是非常直观并且可以编辑的。最明显的例子是鼠标的点取和操作盒显示。用户可以点取 Open Inventor 三维场景中的任一对象,此对象将被高亮度显示,此对象的标志指针将被返回给用户的程序。用户还可以拖动对象的操作盒(一个类似于包围盒的、以线框方式显示的、高亮度的平行六面体),以实现对象的放大、缩小、改变位置等操作。用户使用操作库对图形对象操作就象用户使用鼠标操作窗口的大小与位置那样方便与直观。

3.2 Open Inventor 部件库

Open Inventor 通过 Open Inventor 部件库可以支

持多种窗口系统,尤其是 X Window 系统。Open Inventor 部件库接收窗口事件,然后把窗口事件转换为 Inventor 事件后交由 Open Inventor 的操作库处理具体的操作。Open Inventor 部件库为用户提供多种风格不同的标准界面,用户也可以编写自己的部件加入到 Open Inventor 部件库中。

3.3 Open Inventor 文件格式接口

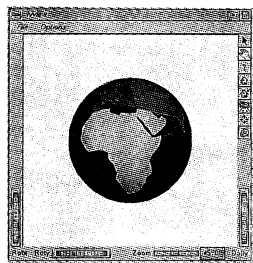


图 2 用 Open Inventor 制作的地球三维模型



图 3 用 Open Inventor 制作的材料编辑器

Open Inventor 文件格式接口包含了采用 Open Inventor 的文件格式进行读入、写出、剪切、粘贴的方法。通过这个接口,用户可以把自己的图形对象写入 Open Inventor 数据库或从 Open Inventor 场景中读出任一对象的信息。

4 结论

总之,我们在实践中感到:用 Open Inventor 实现三维图形的可视化是非常方便而又高效的。Open Inventor 提供统一、优美的界面(如图 2、3),无缝地集成了 Open GL 的显示,采用面向对象的方法封装了对象及对对象的操作。以上这些优点使得 Open Inventor 在 CAD、游戏制作、图像显示等多个领域得到了越来越广泛的应用。

参考文献

- 1 《SGI 用户通讯》,1995. 4, P15~20
- 2 《Open Inventor Mentor》, Addison-Wesley Publishing Company, First Printing, Dec. 1993, P1~144

重要更正

本刊 1997 年第 4 期第 6 页“Intranet—企业信息系统的的发展趋势”一文的作者及工作单位有误,更正后的作者及工作单位如下:

张裔智* 杨建军 陈卫东 鲁东明 潘云鹤

(浙江大学 CAD&CG 国家重点实验室 浙江大学人工智能研究所 杭州 310027)

(* 四川三峡学院计算机系 万县 634000)

特此更正

本刊编辑部

WINDOWS95 的网络功能设置分析

——新一代网络操作系统实践经验浅述

曹 进

(扬州大学商学院计算中心 225002)

摘 要 通过清晰明了的操作步骤,在 1. 网络的设置;2. 如何利用拨号网络;3. windows95 与 internet 这三个方面具体分析说明了视窗九五的网络功能,并提出了一些实际操作过程中需注意的几个问题。

关键词 windows95 internet 网上邻居 画像 桌面 控制面板 网络 拨号网络

1 如何在 windows95 中设置网络计算机

网络计算机的设置分为两步:第一步是硬件设置,第二步是软件设置。

必须要注意的是:计算机可能已被设置成网络计算机了。如果计算机在开始启动的时候提示用户输入一个网络号,或者如果“网上邻居”的画像(NETWORK NEIGHBORHOOD ICON)出现在 windows 的桌面(WINDOWS DESKTOP)上,计算机的网络功能已被设置好了,我们可以跳过这个选择。

在软件的设置之前,首先必须正确安装有关的硬件,这包括网络适配器(网卡或其它设备)和电缆的安装。

网络软件的安装:

1)点按开始(START)开关,再点按设置项(SETTINGS),再接着点按控制面板(CONTROL PANEL);

2)双击网络项(NETWORK);

3)点按增加(ADD)功能;

4)点按适配器(ADAPTER),然后再点按增加(ADD)功能。

此时如果还不知道用户的网络适配器是何种类,可以查看一下附带的说明书。

当设置好一个网络的适配器,windows 将自动地设置使用这个网络所必须的其它组成部件。

附注:

用户也可以通过双击控制面板(CONTROL PANEL)来增加或删除网络组件,并且可以改变从前安装的组件设置。

用户可以在家里或在去公司的路上通过“拨号网络(USING DIAL UP NETWORKING)联上办公室里的电脑,下面将对这部分内容做进一步的说明。

那么,重要的网络组件是些什么呢?

1)客户机(CLIENT)软件。

它可以用户与网络服务器连接起来,这样就能通过计算机网络分享诸如文件夹和图像之类的资源;

2)网络客户服务程序。

使用 netware 网络客户服务程序,用户可以与 novell 局域网的服务器相连接;使用微软的网络客户服务程序,用户可从服务器上分享运行着的 Microsoft Windows95 或 Windows For Workgroups、Windows-NTTM、Lan Manager,或者从其它与微软兼容的计算机上分享;

3)网络适配器(ADAPTER)。

这是一种扩充卡或其它把用户的计算机连接进网络的设备;

4)网络协议(PROTOCOL)。

所谓协议,是指一种语言,计算机通过网络互相通讯时使用的语言,同时使用几种语言是可以的,但两个计算机在相互通讯时必须使用相同的协议;

5)网络服务(SERVICE)软件。

包括分享文件和打印服务或自动备份到网络服务器等。

下一步要做的是确定用户的计算机在网络上的位置:

设置好网络软件后,用户需要提供一些信息给 windows 95 以便确定用户的计算机在网络中的位置。例如,用户的网络管理员可以在一个计算机工作群中管理用户的计算机,这些就象与用户想要使用的大多数资源一样。

在网络中确定用户的计算机位置的具体步骤:

①在网络对话框界面(DIALOG BOX),点按确定格(IDENTIFICATION);

②为用户的计算机输入一个名字,一个工作组中的名字,一个当其他人在网络上寻找用户的计算机时能够看到的描述用户的系统的名字。

另外,为了在对话框界面获得某种帮助,可以点按在右上方有提问标志(QUESTION MARK)的键,然后点按所需了解的问题

实践中需要注意的有关事项:

A)请不要将某个驱动器号与一个网络共享连接,而应该在用户的台式机上或“开始”菜单中为这些网络共享放置快捷方式(如:为\documents\user\dave 增加

快捷方式,而不是将这个共享资源重定向到驱动器 Z),这样网络共享只是在需要时才进行连接。

B)如果用户在某个组中工作,最好在公共服务器上保留文件夹的快捷方式,这样有利于网络定位,而组内其他感兴趣的用户也可以为该文件保留一个快捷方式以便快速存取。

2 利用拨号网络

无论用户在家还是在路上,都可以利用拨号网络(DIAL-UP NETWORKING)来连接办公室里的计算机。通过利用调制解调器和电话机联接计算机,用户可以使用计算机网上任何可被分享的计算机资源,例如一个文件夹或一个打印设备。

用这种方法联接两台计算机,这两台计算机都必须装有调制解调器。而且,在两台计算机中都要设置好拨号网络方式,特别的,欲联入的计算机还须设置成服务器。

如果在“我的计算机”的窗口,没有看到拨号网络的图标,用户可用下列方式进行安装:

- 1). 击开始(START)按键,点按设置(SETTINGS),然后再点击控制面板(CONTROL PANEL);

- 2). 双击安装或删除(ADD/REMOVE)程序;

- 3). 点击窗口设置键,选择通讯功能(COMMUNICATIONS),然后进行细部(DETAILS)设置;

- 4). 点击拨号网络(DIAL-UP NETWORKING)对话框,然后选择“OK”,再一次选择“OK”,下面就可以按屏幕要求来做了;

- 5). 双击“我的计算机”(MY COMPUTER),再双击拨号网络(DIAL-UP NETWORKING);

- 6). 再根据屏幕提示进行即可。

注意点:也可通过串口或并口用信号线联接不同的计算机,并且可通过查询帮助菜单中的“信号线直接联接(DIRECT CABLE CONNECTION)”获得更多有用的信息。

在计算机设置好拨号网络后,用户就可以接通电话,使用文件夹以前需确认是可分享的。

如果用户是第一次设置 WINDOWS,用户也可通过增加或删除(ADD/REMOVE)功能来安装其它未安装的窗口附件。

通过调制解调器使用 Windows 95 内置的通信功能,如超级终端或拨号网络,必须在“控制面板”的“调制解调属性”中配置调制解调器。这些配置并不影响那些为 MS-DOS 或 Windows 3.1 设计的程序中的调制解调器操作。

如果检测到的调制解调器为“标准调制解调器”,或者不能正确检测厂商和型号,可以通过“安装新调制解调器”向导下的“更改”按钮选用其它设备。如果选择了一个错误型号,则调制解调器不能与 Windows 95

通信功能正常工作。如果发现选错了型号,请双击“控制面板”中的“调制解调器”图标,先删除错误的调制解调器,然后添加一个标准的调制解调器型号。

配置为“标准调制解调器”后,Windows 95 将使用调制解调器出厂时的默认设置。调制解调器将自动优化以便加速与 Windows 95 通信程序的连接。但这时,用户不能调节象扬声器音量、多孔协议等调制解调器设置。

3 联接国际信息高速公路

Internet 丰富的联机信息几乎已覆盖用户能想象的所有领域,一旦与 Internet 网连接,用户就可以:

- 与世界各地的人们交换信息。

- 即时获得有关新闻、天气、体育和娱乐等最新信息。

- 可下载软件,如游戏、图片、程序等等。

- 加入到诸如电子公告牌、新闻、娱乐等讨论中。

因此,Internet 网络被人们比作国际信息高速公路而受到了空前的欢迎。

国际信息高速公路是一个非常富有的在线式信息源,从公告牌,到讨论组,到电子邮件和最近的新闻报道,Windows 95 提供了两种方法联上 Internet 网。

使用微软网络的一个新的在线服务,用户能加入 Internet 上的新闻组,下载软件、通过 Internet 发送和接收电子邮件。如要签约进入微软网络,可双击在桌面上的网络画像标志。

当使用拨号入网的方法时,用户可以通过 Internet 网上的一个提供者入网或者一个直接的服务器连接程序。然后,用户可用 FTP 和 TELNET(来自于视窗九五的一个实用工具编辑)来连接各种不同的 Internet 基点。或者,用户可使用其它的视窗九五基本程序来编辑世界范围网 WWW(WORLD WIDE WEB),这些程序可从各种 FTP 基点获得,或从软件销售商处购买。具体的与 Internet 连接步骤:

- 如果现在还没有 Internet 帐号,可双击“桌面”上的 The Microsoft Network(MSN)图标,然后按屏幕提示操作,以便与 Microsoft Network 签约。作为服务的一部分,Microsoft Network 可以访问 Internet。

- 如果已有一个联机服务(见下表)帐号或经常使用电子公告牌服务(BBS),只需再做以下两件事情之一即可:

- a)在双击“桌面”上的 The Microsoft Network 图标与 Microsoft Network 签约。

- b)从联机服务(见下表)中下载 Microsoft Internet 管理器文件(MSIE.EXE)。

- 如果已有一个 Internet access provider 帐号,可从 <http://www.microsoft.com> 中下载 Microsoft“浏览工具”和“Internet 管理器”。除了使用方便,Internet

管理器还能在用户的计算机与用户所喜欢的 Web 站之间建立一条捷径。

注意事项:

如果“桌面”上没有 The Microsoft Network 图标, 请按下列步骤执行来安装 MSN:

打开“控制面板”, 双击“添加/删除程序”图标, 然后单击“Windows 设置”。如果使用的是 Microsoft Plus! 那么用户已经有 Internet 管理器了。

可从如下地址得到 MSIE. EXE:

Internet 网:

ftp://ftp.microsoft.com/PerOpSys/Win-News/

全球网(World Wide Web)

http://www.microsoft.com/

Microsoft Network 网:

主菜单 Categories \ Computers 和 Software \ Software \ Microsoft \ Windows95

计算机服务网(CompuServe)

键入: Go WINNEWS

Prodigy

JUMP WINNEWS

America Online

用关键词: WINNEWS

GEnie

MOVE TO PAGE 95

更多的网络功能

当用户连上一个网络后, 用户还可以做下面这些事:

1) 为了更好的使用电子邮件, 获得更多的信息, 可在帮助索引中查阅“微软交换”(Microsoft Exchange)。

2) 建立一个网络资源的捷径, 按压控制面板上的捷径标志, 可以获得更多的关于捷径的信息。

参考文献

- 1 Dr. John Butler etc, Microsoft Windows 95, 1996.
- 2 Windows 95, 东箭工作室, 人民邮电出版社, 1996.
- 3 The Concise Guide to Windows 95, Michael Halvorson, 1995.

WWW 的几项关键技术浅析

陈 琳

(贵州省气候中心 贵阳 550002)

黄 答

(贵州省气象技术装备处 贵阳 550002)

摘 要 WWW 作为 INTERNET 资源和服务的重要组成部分, 为互联网上可视化和多媒体化的信息的传输和汲取发挥着重要作用, 本文就 WWW 的超文本技术、HTTP 协议、HTML、VRML 等几项关键技术作一粗浅论述。

关键词 WWW HTTP HTML VRML 三维技术

1 概述

WWW 即全球信息网(WORLD WIDE WEB), 它是一个跨越全球的超媒体网络系统, 由于它丰富的信息资源更由于其集文字、图形、影像及声音等于一体的信息传输、展示能力, 为网络用户传播和获取信息提供了一种革命的新方式, 从而吸引着无数的网络用户。WWW 业已成为 INTERNET 最主要的、发展最快的资源和服务组成部分。企业、研究机构、大学及至个人通过建立 WWW 网(服务器), 编制自己的网页(HOMEPAGE), 将已计算机化的广告、研究成果、论文等信息向网上发布, 众多 INTERNET 用户通过使用 WWW 客户机软件浏览、查询自己需要的信息。WWW 上可视化和多媒体化的信息的传输和汲取得力于其几项关键技术和工具, 本文就 WWW 的协议、

技术、工具等进行粗浅论述。

2 HTTP 及其特点

INTERNET 的基本协议是 TCP/IP 协议, 对应着不同的应用又有建立在 TCP/IP 之上的应用层协议, 如 FTP、ARCHIE、GOPHER 等, 与 WWW 对应的主要协议是 HTTP 协议, 即超文本传输协议。但 HTTP 协议支持的服务不限于 WWW, 还可以是 INTERNET 的其它服务, 因而 HTTP 允许用户在统一的界面下, 采用不同的协议访问不同的服务, 如 FTP、ARCHIE、SMTP 等。另外, HTTP 是为分布式超媒体信息系统设计的一个协议, 其主要特点可概括为:

2.1 支持客户/服务器模式

在 WWW 中, “客户”(CLIENT)与“服务器”(SERVER)是暂时的、相对的, 只存于一个特定的链接期间, 一般发出获取信息请求的最终用户称为客户, 而响应信息请求、提供信息的一方称为服务器。处于不同

链接过程的客户与服务器的地位与特定的链接相对应。基于HTTP的C/S模式信息交换过程的简化模型为:CLIENT通过申请套接字与SERVER建立连接→CLIENT向SERVER发送请求信息→SERVER发送响应信息→CLIENT与SERVER均可通过关闭套接字来结束TCP/IP对话。

2.2 简单快速

客户向服务器请求服务时,只需传送请求方法和路径。请求方法常用的有GET、HEAD、POST。每种方法规定了客户与服务器联接的类型不同。由于HTTP协议简单,使得HTTP服务器的程序规模小,因而通信速度很快。

2.3 灵活

HTTP允许传输任意类型的数据对象,正在传输的类型由CONTENT-TYPE加以标记。

2.4 无连接,无状态

无连接的含义即排它性连接,限定每次连接只处理一个请求。服务器处理完客户的请求,并收到客户的应答后,即断开连接。采用这种方式可有效减少传输时间。HTTP协议是无状态协议,即协议不记忆事务处理过程,后续处理若需前面的信息,则它必须重传。这一方面会导致每次连接传送的数据量大,另一方面,在服务器无需先前信息时它的应答较快。

3 超文本技术与HTML

传统的INTERNET应用基于文本并以简单的树形结构来组织信息,信息缺乏关联性,只单纯线性地用字符或数值式的格式化文本表示,用户只能看到单调的文本和菜单选择。所谓超文本技术就是以联想记忆的方式,采用非线性的网状结构组织块状结构信息,把文本按其内部固有的独立性和相关性划分为不同的信息块,然后,再按它们的自然关系链接成网络。超文本作为多媒体的一种基本应用,是实现信息的可视化和多媒体化的重要基础,为INTERNET带来了巨大变革。超文本技术在WWW中的应用是通过利用超文本标注语言HTML(HYPER TEXT MARKUP LANGUAGE)和以网状结构组织起来的信息实现的。

WWW的网页一般是通过HTML语言编制的。HTML是作为“标注语言”出现的,虽然它已逐渐向程序语言发展,但目前其多数属性仍与一般的排版语言更接近。我们在利用某种文字处理软件(如WPS)制作文档时,通过在软件中加入一些控制码,可让文件出现所需的输出效果,如改变字型、字体、加入修饰等。HTML文件的制作也必须加入“控制码”——标签(TAG),客户利用HTML与WWW服务器交互,取得HTML文本,客户端的浏览器通过识别、解释标签,并格式化文件,将设计者预期的信息组织的效果展示出来。客户浏览不同,对HTML命令的支持也不完全相同,所以同一份HTML文件在不同的浏览器上呈现的

效果是有差异的。一般认为NETSCAPE的NAVIGATOR4.0与MICROSOFT的INTERNET EXPLORER3.0的功能较强。HTML语言的主要特点是超文本、超链接、超媒体,可在HTML的文本里嵌入各种链接,而这些链接可以是另一个HTML文件或文本、语言、图形、图像等等,因此用HTML可多角度多媒体地展示信息。用户也可方便地在关联的链接间自由选择,较快地定位和获取自己感兴趣的内容。

4 三维动画技术与VRML

三维技术主要包括几何造型、交换、属性、光影和表面纹理。用HTML描述的页面是静态的,INTERNET上的GIF89A图形格式可将多幅图形组成动画,因此只要在HTML文件里嵌入这种格式的GIP图形就可显示动画,JAVA也可通过在HTML文本中嵌入APPLET的字节码,为WWW增添交互性和动态特性,但这些技术均是二维的而不是为三维的应用而设计的,只有VRML才真正地将三维动画带入INTERNET。VRML即VIRTUAL REALITY MODELING LANGUAGE(虚拟现实造型语言),正如HTML用于定义可与更多的信息相关联的二维世界的布局和内容那样,VRML是用于定义可与更多信息相关联的三维世界的布局和内容的一种文件格式,是一种将三维数据应用在WEB上的规范说明,是一种描述可以通过INTERNET相互联接和访问的虚拟世界的语言。其目的是通过INTERNET不仅描述静态的事物,同时制作具有复杂动态交互的仿真环境,给用户一种亲临现场的真实感和体验。VRML使得信息(包括与其它WEB内容相关联的信息)能够在一个交互的三维空间中很容易地被表达出来。它对大量的诸如3D的事物,或图像、声音、文本等客体的属性及其关联性进行描述,这些客体(称为节点)以层次结构组织起来,形成有序集合,在该集合中,任何一个在特定位置的节点状态的变化,都会影响所有跟随其后的节点。

与HTML文件的浏览原理类似,VRML文件的浏览是通过VRML[®]浏览器完成的。VRML浏览器将VRML文本信息解释成VRML空间中对象的几何形体的描述,与造型关联的光影、纹理等属性也随之确定。VRML空间被用户浏览器解释后,它将提供实时显示,这样客户端就将呈现出一个活动的场景。

5 结语

基于超文本技术的WWW信息服务方兴未艾,其涉及的技术也日新月异飞速发展,HTML、JAVA、QUICK TIME和GIF动画等二维技术与VRML技术的逐渐融合,促进着WEB从二维向三维转变。把VRML、HTML、JAVA、流媒体和数据库结合起来将形成一种新的三维超媒体的WEB,从而为WWW划时代的发展奠定基础。

经典 Ramsey 数 $R(5, 14)$ 的下界

罗海鹏

苏文龙

张正铀

(广西科学院 南宁 530031)

(广西梧州一中 543002)

(广西科委 南宁 530012)

摘 要 研究了素数阶循环图的一些性质,得到了一个 Ramsey 数新的下界: $R(5, 14) \geq 152$ 。

关键词 Ramsey 数 下界

1930 年,英国科学家 F. P. Ramsey 发现了一个定理^[1],揭示了“不可能有完全的无序”这深刻的自然规律而引起人们浓厚的兴趣,发展成为图论、组合数学和离散数学的核心内容——Ramsey 理论,在现代数学中产生了深远的影响,并在计算机科学、通讯工程学和决策科学等许多学科的实际问题中有着广泛的应用。

确定 Ramsey 数是 Ramsey 理论的核心问题,这是世界著名的数学难题。经过各国数学家六十年的努力探索,迄今为止算出来的 Ramsey 数仅有少数几个。1990 年澳大利亚的 B. D. McKay 和我国的张克民采用穷举法,借助于高速电子计算机连续运算了一个月,证明了 $R(3, 8) = 28$ ^[2],这是迄今已知的第 9 个不平凡的 Ramsey 数。随着 Ramsey 数的增大,采用这种方法的运算量将呈指数级的增长,所需要的运算时间是人们难以接受的。因此近年来各国学者主要用各种方法借助计算机对一些具体的 Ramsey 数给出估计^[3]。其中较常用的是沿用文献[4]于 30 年前倡导的方法研究一般的循环图,得到一些 Ramsey 数的下界^[5,6]。但这种方法在产生参数时存在一定的随机性,运算效率不高,很难取得太大的进展。有鉴于此,我们尝试一个新的方法^[9],研究了素数阶循环图的一些性质,利用平移、旋转和对称等手段有效地改进了产生参数的方法,提高了运算效率,得到了一个经典的 Ramsey 数 $R(5, 14)$ 的新的下界。

给定素数 p , 记 $Z_p = \{-p+1, \dots, -2, -1, 0, 1, 2, \dots, p-1\}$ 为模 p 的一个剩余系,参数的集合 $E \subseteq Z_p$ 。作

定义 1. 设图 G 的顶点集 $V_G = Z_p$, 边集为 E : 两个顶点 x 和 y 相邻当且仅当 $|x-y| \in E$ 。我们称图 G 为 p 阶循环图并记为 $G(E)$ 。

我们据此构造了一个 $p=151$ 阶循环图,其边集

$E = \{1, 3, 5, 6, 7, 8, 10, 14, 18, 19, 24, 26, 36, 37, 39, 40, 41, 48, 52, 56, 57, 59, 64, 69, 71\}$

我们在计算机上验证了:在如前定义的 $p=151$ 阶循环图 $G(E)$ 中既不含 5 点团 K_5 , 也不含 14 独立点集 \bar{K}_{14} 。由这个结论并据著名的 Ramsey 定理,我们就证

明了

定理 1. $R(5, 14) \geq 152$ 。

这个结论填补了文献[3]的空白。

参考文献

- 1 F. P. Ramsey, On a Problem of formal Logic, Proc. London Math. Soc 2nd Ser, 30(1930)264—286.
- 2 B. D. McKay and Zhang K. M., The value of the Ramsey number $R(3, 8)$, J. Graph Theory, 1992, 16(1): 99—105.
- 3 Stanislaw P. R., Small Ramsey numbers, RJT—TR, 1993, 1: 2—3.
- 4 J. E. Graver and J. Yeckel, Some graph theoretic results associated with Ramsey's theorem, J. Comb. Theory, 4 (1968): 125—175.
- 5 Geoffrey Exoo, A lower bound for $R(5, 5)$, J. Graph Theory, 1989, 13(1): 97—98.
- 6 王清贤,王攻本. Ramsey 数 $r(3, q)$ 的新下界, 北京大学学报(自然科学版), 1989, 25(1): 117—121.
- 7 宋恩民,董向锋,许如初. 求 Ramsey 数下界的循环巧妙图搜索算法研究, 应用数学, 1995, 8(4): 424—428.
- 8 谢继国,张志辅. 经典 Ramsey 数 $R(5, 9)$ 和 $R(5, 10)$ 的下界, 科学通报, 1996, 41(20): 1918—1919.
- 9 张正铀,苏文龙,罗海鹏,吴康. Ramsey 数 $R(3, 3, 3, 3, 3)$ 的下界, 计算机应用研究, 1997, 3, 30—31.

中国首创的特色科技期刊

中文信息

创刊十五年扩大发行双月刊

· 中国语文进入信息社会的通道

· 中华民族开拓高新技术的热线

◇“热点传真”提供海内外最新信息◇“语文信息技术”全国独有◇“软件开发”专为振兴信息产业服务◇“计算机教育”向全国中小学提供参考教材。

全国邮局订阅 邮发刊号: 62—56, 全年 30 元
直接订阅: 610065 四川联合大学(西)263 信箱

若干个经典 Ramsey 数 $R(5, q)$ 的新下界

苏文龙

罗海鹏

张正铀

(广西梧州一中 543002)

(广西科学院南宁 530031)

(广西科委南宁 530012)

摘 要 本文构造了 3 个新的素数阶循环图,从而得到了 3 个 Ramsey 数的新下界: $R(5, 19) \geq 312$, $R(5, 20) \geq 338$, $R(5, 21) \geq 374$ 。

关键词 Ramsey 数 下界

确定 Ramsey 数是组合数学和图论中著名的难题。根据 Ramsey 数研究权威 S. P. Radziszowski 的综述[1]的 1997 年 7 月 16 日的更新版本,当 $q=5, 6, \dots$, 15 时, $R(5, q)$ 的下界分别大于等于 43, 58, 80, 95, 114, 118, 140, 150, 158, 182, 198。其他的 $R(5, q)$ 的下界均是当前研究的空白。

寻找 Ramsey 数新的下界越来越困难,主要的原因是在这个寻找过程中,计算量的增大是目前的计算机难以承受的,有鉴于此,我们把数论、近世代数、组合学、图论等领域里的一些研究方法结合起来[2],用于 Ramsey 数下界的寻找中。

我们考查素数阶循环图,利用平移和旋转等性质改进了产生参数的方法,提高了运算效率,得到了一系列 Ramsey 数新的下界[2, 3, 4, 5, 6, 7, 8]。在我们已发表和待发表的文章中,还给出了 $R(5, 11)$ 、 $R(5, 12)$ 、 $R(5, 13)$ 、 $R(5, 14)$ 、 $R(5, 15)$ 的新的下界。在本文中,我们进一步构造了 3 个新的素数阶循环图,从而给出了 3 个 Ramsey 数的新的下界。

对于给定的素数 p , 记 $Z_p = \{0, 1, 2, \dots, p-1\}$, 选定一个参数集合 S 。设图 G 的顶点集 $V_G = Z_p$, 两个顶点 x 和 y 相邻当且仅当 $\min\{|x-y|, p-|x-y|\} \in S$ 。我们称图 G 为关于参数集合 S 的 p 阶循环图并记为 $G_p(S)$ 。

据此我们构造了 3 个素数阶循环图:

1) 给定素数 $p=311$ 与参数集合

$S = \{1, 4, 6, 8, 10, 16, 17, 20, 21, 22, 23, 24, 27, 34, 35, 36, 39, 46, 48, 49, 52, 60, 69, 77, 87, 95, 96, 98, 100, 101, 102, 103, 105, 107, 120, 126, 129, 132, 134, 138, 141, 144, 149, 151, 152\}$ 。

2) 给定素数 $p=337$ 与参数集合

$S = \{1, 6, 10, 15, 18, 19, 20, 24, 28, 32, 35, 37, 39, 43, 46, 51, 52, 54, 55, 58, 63, 68, 69, 70, 72, 73, 84, 85, 87, 92, 94, 95, 96, 99, 100, 102, 111, 112, 116, 117, 123, 125, 128, 129, 132, 134, 136, 139, 148, 155, 156, 159,$

$161, 165\}$ 。

3) 给定素数 $p=373$ 与参数集合

$S = \{1, 2, 4, 7, 17, 18, 21, 23, 27, 33, 39, 43, 45, 47, 48, 53, 54, 66, 69, 73, 74, 75, 78, 80, 83, 88, 89, 92, 94, 95, 97, 98, 104, 110, 114, 116, 120, 121, 123, 124, 128, 130, 132, 137, 138, 143, 145, 150, 154, 156, 159, 160, 165, 169, 171, 173, 176, 178, 182, 185\}$ 。

我们在计算机上验证了:如前定义的素数阶循环图 $G_{311}(S)$ 中既不含 5 点团 K_5 , 也不含 19 独立点集 \bar{K}_{19} 。循环图 $G_{337}(S)$ 中既不含 5 点团 K_5 , 也不含 20 独立点集 \bar{K}_{20} 。循环图 $G_{373}(S)$ 中既不含 5 点团 K_5 , 也不含 21 独立点集 \bar{K}_{21} 。由于这些结论并据 Ramsey 定理,我们就证明了

定理 1 $R(5, 19) \geq 312$, $R(5, 20) \geq 338$, $R(5, 21) \geq 374$ 。

上述 3 个结果填补了文献[1]中的有关 Ramsey 数下界的空白。

参考文献

- 1 S. P. Radziszowski. Small Ramsey Numbers. The Electronic Journal of Combinatorics 1(1994), DS1: 1-29.
- 2 Su Wenlong. The Estimation of Lower Bounds about Some Ramsey Numbers $R_n(3)$ and $R_n(4)$. 广西科学, 1996, 3(3): 4-7.
- 3 张正铀, 苏文龙, 罗海鹏, 吴康. Ramsey 数 $R(3, 3, 3; 2)$ 的下界. 计算机应用研究, 1997, 3: 30-31.
- 4 罗海鹏, 苏文龙, 张正铀, 吴康. 一个 Ramsey 数的新的下界. 广西大学学报, 1997, 2: 95-97.
- 5 罗海鹏, 苏文龙, 张正铀, 黄苏宁. Ramsey 数 $R_4(4)$ 的新下界. 桂林电子工业学院学报, 1997, 2: 49-52.
- 6 张正铀, 苏文龙, 罗海鹏. Ramsey 数 $R_5(4)$ 的新下界. 广西民族学院学报, 1997, 1: 1-4.
- 7 Su Wenlong, Wu kang, Luo Haipeng. The Normal Subgroup of Cyclic Group and the Estimation of Lower Bounds about Ramsey Numbers. 97' Sum-

mer School and International Conference on Combinatorics, 1997, 5.

8 张正袖, 苏文龙, 罗海鹏. 经典 Ramsey 数 $R(5, 11)$ 的下界. 广西科学, 1997, 4(2): 84, 92.

WINDOWS 消息驱动机制中的核心技术分析

杨 亮 阮晓星 魏晋鹏

(武汉大学计算机科学系 430072)

摘 要 本文主要基于 WINDOWS 的消息驱动机制, 通过对消息队列中几个未公开关键数据项的剖析, 揭示了 WINDOWS 在进行消息的接收/传递/发送过程中, 其内部工作机制的核心秘密。

关键词 消息驱动 消息队列 任务队列 系统队列

1 引言

消息驱动一直被认为是 WINDOWS 的核心部分, WINDOWS 正是通过它来实现应用程序之间的通信。我们知道, 在 WINDOWS 中无论是用户还是操作系统本身所做的一举一动都将被当做某个类型的事件来处理, 如键盘事件、鼠标事件等等, 并会被标识为某种类型的消息存放起来。这些保存在消息队列中的消息可以通过消息循环代码获取并发送到目标窗口进行处理。而 WINDOWS 内部的任务调度机制又是与消息驱动紧密相关的, 因为如果当前正在运行的任务因为所等待的消息没有出现或者是自身的所有消息都已处理完毕, 它将会暂时让出 CPU, 并唤醒任务队列中其它存在等待消息的任务, 让它们接管控制权以处理这些消息。

2 未公开的 WINDOWS 的消息类型标记

我们很想知道从 WINDOWS 内部的角度来看, 它到底是通过什么样的方法来动态地反映消息已经收到和送走呢? 它又怎么确定哪些消息允许接收, 哪些消息必须屏蔽呢? 从前面的叙述中我们可以看出消息队列(分为任务队列和系统队列)在这其中扮演着一个十分重要的角色, 它是保存消息以及与消息接收/传递/发送密切相关的信息的场所。从某种意义上来说, 通过跟踪 WINDOWS 对消息队列的数据项(尤其是几个消息类型标记)的读写过程, 我们就可以清楚地了解 WINDOWS 整个消息驱动过程的秘密。因为 WINDOWS 正是通过对它们的动态操作来实现消息在系统中的平滑流动的。而基于这个道理, 我们也可以仿照 WINDOWS 通过直接设置这几个消息类型标记来人为地在消息队列中创建各种消息。实践证明, 掌握了这个原则将十分有助于在 WINDOWS 环境下编程时采用灵活的手段来引进我们所希望出现的消息, 而不一定需要产生这个消息的条件成立。

下面我们就逐个阐明这几个未公开的消息类型标

记的作用, 并附上一个 WINDOWS 内部过程的反汇编代码作为实例说明。为了简洁起见, 我们分别把它们称作 WakeBits, ChangeBits 和 MaskBits。

WakeBits: 这个数据单元位于消息队列的偏移 44H 处, 它的作用是用来标识当前任务的消息队列中已经存在的消息的类型。一个任务每次接收到消息后, 都会由发送/传递消息的任务或捕获消息(如硬件消息)的任务来设置该任务消息队列中 WakeBits 的相应标记位, 以表示这个消息已经被保存在消息队列中。WakeBits 是比较可靠的, 当它的某位被清除或置位时, 确实表明有对应的消息加入到队列中或从队列中删除。WakeBits 在任务队列和系统队列中都存在。

ChangeBits: 这个数据单元位于消息队列的偏移 42H 处, 它的作用是用来标识消息队列中消息类型的动态变化过程。换句话说, 也就是自上一次调用消息获取函数 GetMessage()/PeekMessage()或消息验证函数 GetQueueStatus()以来, 新加入到消息队列且仍然保留在队列中的消息类型。因此每次调用这三个函数以后, 都会重新清除 ChangeBits 中本次被获取或验证的消息类型位。此外在调用 SendMessage()函数发送消息时, 如果 WakeBits 中其它类型的消息所对应的标记位没有被设置的话, 为了使得 ChangeBits 和 WakeBits 保持一致, ChangeBits 中的对应位也将被强行清除。而当队列中出现某一新类型的消息时, WakeBits 和 ChangeBits 又都会被置相同的位。因此 ChangeBits 和 WakeBits 的值并不总是相等的, 如果要判断消息队列中的消息类型时, 最好通过检测 WakeBits 的值来实现, 而不要使用 ChangeBits。ChangeBits 只存在于任务队列之中。

MaskBits: 这个数据单元位于消息队列的偏移 46H 处, 它的作用是用来标识禁止当前任务响应的消息类型的屏蔽位。一旦某一类型的消息屏蔽位被设置, 当该任务接收到此类型的消息后, 将不会使任务的事件计数器的值增加, 从而也就使任务失去了重新获得控制以处理接收到的消息的机会。设置 MaskBits 的目

的在于:一个任务在等待某个指定的消息出现时,可能会有其它一些不需要的消息加入到队列中。为了不受这些无关消息的干扰,避免任务在所期待的消息出现以前就被唤醒以处理不相干消息,我们就可以通过设置消息屏蔽位的方法来禁止任务响应无关消息,从而保证任务在被重新唤醒时,消息队列中一定已经存在着所需要的消息,但是有些消息是不能被屏蔽的(就象存在不可屏蔽中断一样),象 `Qs_SendMsg`。如果禁止响应它,将会造成整个系统因为互相等待发送的消息返回处理结果而死锁。更值得注意的是,在 `MaskBits` 中,某一位为 0 表示该位对应的消息类型被屏蔽;为 1,才表示允许响应,因此在设置消息屏蔽标记时,必须把 `Qs_SendMsg` 的值 0040H 加入到 `MaskBits` 中去。关于 `MaskBits` 的具体使用,请参见函数 `SetMsgFlags()` 的代码注释部分。`MaskBits` 也只存在于任务队列之中。在调用 `GetMessage()/PeekMessage()` 获取消息时,WINDOWS 还创建另外一个消息过滤标记 `FilterBits`,它的作用与 `MaskBits` 是截然不同的,前者表示禁止接收的消息的类型(被过滤的消息是不会被放入到消息队列之中去的),后者表示禁止响应某类型的消息,而且在队列中并不存在某个数据单元与 `FilterBits` 相对应。

消息类型标记 `WakeBits`, `ChangeBits`, `MaskBits` 的长度都是一个字节单元,它们各位的含义说明如下表所示,其值可以是多个标记位相或的结果。

表 1 WINDOWS 内部的消息类型表示

位(BIT)	标记	说明
0	QS_KEY	当前队列中存在键盘消息(01H)
1	QS_MOUSEMOVE	当前队列中存在鼠标移动消息(02H)
2	QS_MOUSEBUTTON	当前队列中存在鼠标按键消息(04H)
3	QS_POSTMSG	当前队列中存在普通消息(08H)
4	QS_TIMER	当前队列中存在定时消息(10H)
5	QS_PAINT	当前队列中存在屏幕重绘消息(20H)
6	QS_SENDMSG	当前队列中存在发送消息(40H)
14	*QS_REPLIED	在调用 <code>SendMessage()</code> 时使用
15	*QS_RESULT	在调用 <code>SendMessage()</code> 时使用

传递消息泛指通过调用函数 `PostMessage()/PostAppMessage()` 放入到消息队列中的消息,而发送消息则泛指调用函数 `SendMessage()` 放入到消息队列中的消息。在 WINDOWS 内部所有诸如 `WM_XXX`, `EM_XXX`, `BM_XXX` 之类的外部消息类型都被归结为传递消息或发送消息,只有当调用函数 `GetMessage()/PeekMessage()` 把消息从队列的消息缓冲区中取出并放入到 `MSG` 结构中时,这些外部消息类型才会被区别对待;打星号的表示该值并不是用来标识消息队列中有某个消息存在,而只是被 `SendMessage()` 函数用作某种过程的完成标记,它们分别表示接收消息的任务已经收到消息和处理完发送给它的消息。

3 消息类型标记在消息队列中的设置

我们前面提到了任务队列(`TaskQueue`)和系统队列(`SystemQueue`),前者是属于应用程序的消息队列,它是 WINDOWS 在创建一个任务时,由任务启动代码调用 `InitApp()` 函数创建的,用来保存与该任务有关的消息;而系统队列则是 WINDOWS 在启动时由 `KERNEL` 创建的,用来保存所有硬件输入消息(主要是键盘消息和鼠标消息)的。和任务队列不同,系统队列不属于某个具体的任务,也就是说它可以为所有的任务提供保存属于该任务的硬件消息的服务。但要注意,这种共享服务并不是同时进行的。在某个时刻,系统队列只为处在那个时刻的当前任务服务,这是因为在 WINDOWS 中,所有的硬件输入事件都被认为是与当前任务(也即输入焦点)有关的。

由于系统队列与任务队列都服务当前任务,所以当前任务在接收到一个硬件消息时,必须同时设置二者的 `WakeBits`;其中系统队列应该设置 `Qs_PostMsg` 位,任务队列则应设置 `Qs_Mouse` 或 `Qs_key` 位。而同样当系统队列中的硬件消息个数减少到 0 时,也必须分别清除系统队列和任务队列中 `WakeBits` 的上述标记位。

4 消息类型标记在 WINDOWS 中的使用实例

`SetMsgFlags()` 是 WINDOWS 的一个内部过程,其作用是设置指定任务消息队列中 `WakeBits` 和 `ChangeBits` 所对应的标记类型位,以表明该任务已接收到了此类型的消息,它通常由消息函数 `SendMessage()`, `ReplyMessage()`, `PostMessage()`/`PostAppMessage()` 调用。`SetMsgFlags()` 的程序代码是用 `Soft_Ice` 对 `User.Exe` 进行反汇编得到的,过程名系笔者所加。`SetMsgFlags()` 的入口参数是 `AX`(待设定的消息类型标记值)和 `ES`(指定任务的消息队列句柄—`hTaskQueue`),无出口参数。

0487:00004311 MOV CX,ES;CX←指定任务的消息队列句柄

0487:00004313 MOV BX,07AF;User 的一个固定数据段选择符

0487:00004316 MOV ES,BX;ES 指向 User 的一个固定数据段

0487:00004318 LEA BX,[00EE];BX←保存 `hTaskQueue` 的地址

0487:0000431C TEST AX,0006;判断产生的是否为鼠标消息(02+04/`Qs_MouseMove`+`Qs_MouseButton`)

0487:0000431F JNZ 432A;是,则转

0487:00004321 LEA BX,[00F0];BX←保存 `hTaskQueue` 的地址

0487:00004325 TEST AX,0001;判断产生的是否为键盘消息(01/`Qs_Key`)

0487:00004328 JZ 432D;不是,则转

431F→

注:如果产生的是硬件输入消息,为了使得以后在使用该消息时知道它属于那个任务所有,就必须先保存拥有此消息的任务队列句柄。鼠标消息和键盘消息将分别使用不同的数据单元项——00EEH 和 00F0H 来保存它。

0487:0000432A MOV ES:[BX],CX;在数据段中保存 hTaskQueue

4328→

0487:0000432D MOV ES,CX;ES 指向指定任务的消息队列

注:当系统中发生某种事件时,就通过设定指定队列中 ChangeBits 和 WakeBits 内的 Qs_xx 位,以表明该任务有这种类型的消息产生。

0487:0000432F OR ES:[0042],AX;设置指定任务中的 ChangeBits

0487:00004334 OR ES:[0044],AX;设置指定任务中的 WakeBits

注:一旦 Qs_xx 类型的消息被屏蔽后,拥有它的任务将暂时不会对该类型的消息进行处理。这种作法的目的在于保证那些交出控制权的任务。在其队列中出现其所期待的某种或多种消息之前,不致于被一些无关消息所唤醒。这也很好地说明了一旦存在着消息屏蔽,那么每一条消息的产生并不都对对应有一个属于该任务的事件(TDB;0006)的出现。

0487:00004339 TEST ES:[0046],AX;该类型的消息是否已被屏蔽

0487:0000433E JZ 4353;是,则转

注:在期待的消息出现后,应及时清除消息屏蔽标记。因为本次所屏蔽的消息在下次可能成为允许响应的消息,而本次希望产生的消息下次不一定还需要。

0487:00004340 MOV WORD PTR ES:[0046],0000;请 MaskBits (0 — MASK; 1 — UNMASK)

0487:00004347 PUSH ES

0487:00004348 PUSH WORD PTR ES:[0002];指定任务的任务数据库(TDB)

注:调用 PostEvent()后,该任务的等待事件数增 1,因此调度程序在进行下一次调度时就会把该任务唤醒,以处理已经产生的消息。

0487:0000434D CALL KERNEL!POSTEVENT;指定任务的事件计数器值加 1

0487:00004352 POP ES

435E→

0487:00004353 RET

4 结束语

从以上分析可以看出,Windows 内部的消息表示实际上都是围绕着这三个核心数据项来进行的,因此深入了解其工作原理对于我们掌握 Windows 的消息驱动机制将是十分有益的。

参考文献

- 1 Andrew Schulman 等著,金正等译. 未公开的 WINDOWS 核心技术. 清华大学出版社
- 2 雷军等著. 深入 WINDOWS 编程. 清华大学出版社

EIS 需求分析的困难及方法

刘 伟 杜昌勇 李 朋

(复旦大学管理学院 上海 200433)

摘 要 EIS(Executive Information System, 经理信息系统)主要面向企业经理及高层管理人员,而经理信息需求分析是 EIS 开发面临的主要问题。本文分析了经理信息需求的特点,讨论了当前几种经理信息需求分析的方法,然后提出了一种基于企业过程思想的经理信息需求分析的方法。

关键词 EIS 经理信息需求 企业过程

1 引言

当前市场竞争日益激烈,情况瞬息万变,企业面临着一个动态的环境。经理人员如何获得及时准确的信息,作出快速而有效的决策则显得十分重要。随着商业步伐的加速,过去可以等几星期的决策,如今必须在几

天、几小时甚至几分钟内作出。如果经理人员不能及时获得他所需要的信息,那么将导致失去市场机会,或不能及时解决问题。经理人员在面对价格、产品、资源分配、对竞争者的反应等情况时作决策的时间越来越少,他们需要计算机的支持及时获得需要的信息。经理信息系统(Executive Information System, 简称 EIS),将在这方面发挥重要作用。EIS 被定义为是一个帮助经

理人员方便并且及时地获得企业当前状态的高层次的战略信息的计算机信息系统。许多企业在尝试开发EIS,希望EIS能帮助经理人员方便快速地获得信息,提高经理人员的工作效率和决策水平,监控企业运转情况,提高通信能力,帮助获得竞争性信息等等。有些企业获得了成功并收到较好的效益,但更多的EIS开发则失败了,其中一个重要原因是开发的系统不能满足经理人员的需要。如何决定经理人员的信息需求是EIS开发成功的关键。本文从经理人员信息需求的特点入手,对当前几种需求分析方法进行分析评价,并提出了一种基于企业过程思想来分析经理人员信息需求的方法。

2 经理信息需求特点

经理人员在企业中处于高层的比较高的地位,他们最重要的任务就是制定决策。除了他们的智力和经验外,经理人员能利用的最重要的资源就是信息。经理人员的信息需求往往体现如下特点:

1)信息来源的广泛性。经理的信息既来自企业内部,又来自外部信息渠道。内部信息大多来自企业数据库,其数据内容由企业的事务处理系统及职能部门的管理信息系统产生;其它的内部信息渠道有职能部门、企业员工、各种小道消息、文件等。外部信息渠道有外部数据库、出版公布的数据、客户、供应商、新闻机构、证券市场、行业信息等等。

2)信息的弱相关性。经理人员经常被大量的信息包围着,而其中很多都是不相关的或弱相关的,甚至是没有用的信息;相反却常常得不到他们制定决策需要的信息。

3)信息需求的易变性。经理人员面对的是企业内外部不断变化的环境,情况的变化促使他们制定决策的内容和方式都随之变化,导致了他们对信息需求的不稳定性和易变性。

4)信息需求的不可知性。经理人员从事的是一些高度非结构化的工作,常常不知道他们需要什么信息。即使他们尽最大努力和EIS开发人员沟通,表达自己的想法,可一旦系统摆在他们面前,立即就会发现系统并没有提供他们所需要的全部信息或部分信息。

5)信息需求的独特性。经理人员常常面对的是许多非结构化或半结构化的问题,没有固定的解决方法,而经理人员的智力、知识背景、经验、思维方式等都有较大区别,导致他们对问题的处理方式及对信息的需求都有一定的差异,即使是处在相同地位的经理人员也是如此。

以上分析了经理信息需求的特点,从中可以看出经理信息需求分析是一项困难而又十分重要的工作,也对EIS开发人员提出了较高要求。那么在决定经理信息需求的过程中,应该采取什么方法呢?

3 经理信息需求分析方法评述

在研究如何有效地挖掘经理信息需求的过程中,一些机构或学者提出了自己的方法,其中有代表性的包括IBM公司的企业系统规划(BSP)方法。罗卡特ROCKART的关键成功因素法(CSF)和沃特森等人提出的原型法(Prototyping)。

1)BSP方法。这种方法为EIS分析人员提供一个简单而又有效的挖掘经理信息需求的调查框架。该框架从经理解决问题和经理决策两方面展开。

BSP $\left\{ \begin{array}{l} \text{问题} \longrightarrow \text{解决方案} \longrightarrow \text{信息} \quad (1) \\ \text{决策} \longrightarrow \text{信息} \quad (2) \end{array} \right.$

(1)问题的解决:

a 在您领导的部门(企业)里,完成目标遇到的主要问题是什么?

b 解决这些问题的最好方法是什么?

c 在这些解决方法中,信息是如何发挥作用的?

(2)决策:

a 在您的管理职责中,主要的决策有哪些?

b 为了更好地决策,您需要什么信息?

下面以企业订单处理系统为例进行说明。

问题的解决:

问题	解决方法	信息
库存经常短缺	更好的库存管理	缺货报表,平均信息
错误地将短缺商品发送给客户	让仓库部门了解不同客户的相对重要性及信用情况	客户信用信息,客户重要的等级
非重要的或存在信用问题的客户	让运输部门了解正在处理的订单的时间安排及运输计划的订单情况	划

制定决策:

决策	信息
需要拜访哪些客户,销售什么给他们?	客户订货历史记录,库存情况
向谁提供信用?多少?什么时候终止?	信用评定,付款历史,当前帐户情况
产品的去向?	客户地址
哪些产品可以一同发送以节省运输成本?	运输计划和客户地址
本?	订单优先级,客户的重要性,客户的信
什么时候可以更改订单?	用情况

2)关键成功因素法(CSF)其框架为:

CSF \longrightarrow 关键成功因素 \longrightarrow 信息

a 在您的部门(企业)里,关键成功因素是什么?

b 需要什么信息来确定关键成功因素在控制之中?

同样以订单处理系统为例进行说明:

关键成功因素	信息
足够的库存来完成客户订单	订单按时完成比例(全面的并且按客户和产品分类)
订单运输快捷	发运时间(全面的并且按客户分类)
客户按期付款的高比例	近期未付款报告
供应商及时供货	供应商没有及时供货的例外报表

BSP方法和CSF方法都比较简单实用,能够使系统分析人员和经理人员很快地熟悉掌握,并且这两种方法具有很强的信息处理理论背景,但是这两种方法只是提供了一个经理信息需求的概念框架,还需要将

这些概念性的想法和过程转化为具体的设计。

3) 原型法(Prototyping)

在事先难以确定信息需求时常通过建立原型,作为系统分析人员和经理人员沟通的工具。开发人员可以利用原型获得更为明确的信息,然后进一步修改原型,通过这样一个反复迭代的过程来接近目标。EIS 被认为是永久的原型,主要就在于经理人员的信息需求处在不断变化的过程中。原型法的缺陷在于原型的焦点在于反映经理人员的要求,而忽视了在企业建立一个良好基础为 EIS 提供支持(如信息的产生和提供);另外容易将一些经理个人的需要置于企业的需要之上,导致 EIS 很难成为企业有机的组成部分,而且一旦该经理人离开,带有较多个人痕迹的 EIS 将很可能被后来者抛弃。因此在建立原型时,必须努力寻求企业范围内的支持。

可以看出原型法可以和 BSP 方法或 CSF 方法紧密结合起来,由 BSP 或 CSF 方法获得经理人员概念性的想法和过程,然后结合原型法进行具体设计和实现。这样看来似乎很完美了,但是,一些根本性的问题仍没有解决。如:

①BSP 或 CSF 方法都过于简单,只是以框架形式列出几个问题来向经理人员寻求解答,不能给予开发人员以更多的指导和帮助,仿佛只要把问题列出来,让经理人员一填就可以了。

②缺乏系统的指导思想和明确的目标。经理人员往往倾向于从本部门的角度来考虑问题,而部门的目标是否和其它部门的目标有冲突,是否和企业目标一致?各部门如何保持紧密联系实现企业目标?对企业领导人来说,企业目标是否在各部门得到反映?开发人员基于什么指导思想来决定经理人员的信息寻求?

③对于部门经理人员来说,决策所需的信息大部分来自其它部门,经理人员是否能得到?这些信息其它部门是否能够提供,是否想到提供?企业领导人又是否能够从各部门那里获得他想要的信息?

所有这些表明在决定经理信息寻求过程中,没有树立明确的企业过程思想。往往是一个完整的企业过程被现有的一些部门分割了,使得这些部门间在客观上产生了很强的耦合关系,经理人员常常得不到足够的信息,这是问题的关键所在。其实在 BSP 和 CSF 方法所举的例子中已体现了企业过程思想,但没有明确提出来,也没有很好地运用到需求调查方法中去。

4 企业过程思想

在当前市场环境瞬息万变,时机稍纵即逝的情况下,许多企业感到它们的层次结构、部门分割和相应的信息系统已日益落后并威胁到它们的生存。

按功能或部门来设计组织导致各种管理层次的出现来协调、控制各部门及企业员工的工作,但是这并不能保证各部门或单位的目标和企业的目标保持一致。

在企业内部,信息是从下面一层一层地垂直向上流动,导致信息流动缓慢并被中间一些层次歪曲,企业上层得不到准确及时的信息导致决策失误并失去许多机会。各部门经理难以直接得到其它部门的信息,而通常是企业上层反馈过来,这也影响了信息的准确性和及时性。

另外,各职能部门自行建立其信息系统和数据库来满足其目标,缺乏跨部门/职能的信息系统或者建立跨部门系统时缺乏明确一致的指导思想,从而使得整个企业的系统缺乏集成,不能达到信息充分共享。

所有这些对经理人员的信息获得造成极大的困难,不能有效快速地制订计划和决策,响应市场的变化和客户的需求。造成问题的根本原因在于没有树立企业过程思想,那么什么是企业过程呢?

过程在 WEBSTAR 词典中的解释是产生某一个结果的一系列活动或操作,特别是指连续的操作或处理。在管理理论中,企业过程是指组织结构和人员,遵循管理原则,运用管理信息、技术和方法来实现企业目标的连续活动过程。可以看出,一个过程具有组织结构和人员、管理原则、管理技术、管理信息和方法等各个特性,而过程的主体是由活动组成,每一个特性的不同取值都将体现在不同过程的不同活动及其相互之间的关系之中。而活动是指一种变换或操作。企业过程的活动往往是接收某一种类型的输入,在某种规则的控制作用下,利用某种资源,经过变换或操作转换为输出。不同的活动其接受输入、处理规则、利用的资源、输出也不同。

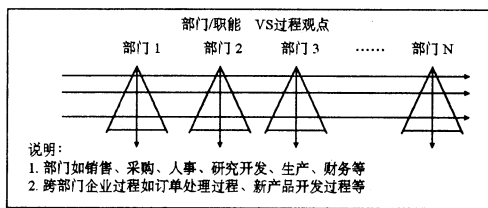


图 1

在企业系统中,和过程的意义容易混淆的概念是功能或职能。功能往往描述的是这个系统干什么,是个静态的概念,而过程强调的是为了完成目标任务,系统是如何进行的,是个动态的概念。用过程的观点来描述企业,即把企业看成是一系列企业过程的集合,如订单完成过程、新产品开发过程、维护商标过程等等。如图 1 所示。

因此,要解决经理人员如何及时准确获得信息这一问题,必须树立企业过程思想。一方面要求以企业过程为核心对企业组织进行改造,对经理人员的权利和责任进行调整;另一方面则要求 EIS 开发人员树立企业过程思想来决定经理的信息需求,从而建立基于过程的跨部门/职能的 EIS。至于如何以企业过程为核心

改造企业,对企业进行重组,建立扁平化组织、自我管理组、分散决策机制以及如何对企业过程进行优化和再设计,这在许多有关企业过程再工程(Business Process Reengineering,缩写为BPR)的文献中有详尽的阐述。本文将重点放在探讨如何基于企业过程思想来决定经理人员的信息需求。需要指出的是基于过程的企业改造为经理人员获得信息进行有效决策提供了良好的企业环境和组织管理保障,也是EIS获得成功的根本所在。

5 基于企业过程思想的需求分析方法

这种方法的核心是将经理人员的计划和决策纳入某一个或多个企业过程中来进行考虑,分析经理人员及其部门在相应企业过程中所处的位置,在这个过程中需要和哪些部门配合,获取什么信息。它要求EIS分析人员和经理人员事先就企业过程思想达成一致,作为双方交流的基础。

基于企业过程思想的经理信息需求分析方法分为以下六个步骤:

1)确立企业战略经营目标(Strategy Business Objective,简称SBO)。经理人员的计划和决策是处于企业环境中,它要求采用企业观点来看待信息需求,确认企业SBO和相关过程,并将它们和经理人员的职责联系起来。SBO是企业行动的方向,也是经理人员计划和决策的前提和依据。例如,SBO包括成为市场领导者、低成本、新产品上市时间快、零库存等等。

2)确立战略经营目标(SBO)的相关过程。一旦SBO确定,接下来就应确立相关的企业过程。例如目标是零库存,那么采购、制造、分销等过程对于SBO的实现则相当重要。这样有助于发现那些监控SBO的重要信息。

3)区别目标及过程的重要性。对于企业具有多个SBO,应确立这些目标及相关过程的优先级或重要性,使得EIS开发人员和经理人员能很好地区分各种信息的重要性,决定哪些信息是至关重要或必须首先获得的。

4)确定与SBO相关的过程之间的信息联系。这使得处在不同过程的经理人员就信息的共享达成共识,通力合作,实现企业目标。

5)确立每个过程的关键信息,如时间、成本、客户信息。对于一个订单处理过程,交货期限、订单所处状态、客户信用情况、库存信息等都是关键信息。

对于EIS的开发来说,正确信息的收集和表现是非常重要的。当前许多企业,一个过程往往涉及到多个部门,导致关于过程的关键信息分散给了不同的部门及经理人员,形成了这些经理人员制订计划和决策的强相关性。例如,在订单处理过程中,仓库部门的经理人员一般根据交货时间的先后给客户发货,但当产品

短缺时仍这样发货,就有可能让企业重要的具有战略伙伴关系的客户得不到所订购的产品而造成恶果,或者将紧缺商品发送给了一些信用低的客户。这就要求信用管理部门、客户关系部门或销售部门及时将信息传送给仓库部门,使得仓库部门经理能够在制订发货计划时灵活调整,进行有效决策。

由于过程内部决策的相关性,EIS开发人员和经理人员都应树立这种企业过程思想,加强合作交流,建立跨部门的信息系统。在这当中更高层次的企业管理人员应发挥积极作用。同时,在决定经理信息需求当中,开发人员不仅和单个的经理人员交流采访,而且应与过程相关的各部门经理人员召集起来,共同来决定需求。这不仅有利于挖掘单个经理人员潜在需求,而且各经理人员能对许多问题达成共识,有利于明确需求,并为今后的跨部门系统的开发打下良好基础。

6)在设计实现过程中不断挖掘经理人员的信息需求。EIS是个永恒的原型系统,它需要经理人员的不断参与,同时也要求开发人员不断重复以上步骤来确定经理信息需求及其变化。如此循环反复,没有止境。

6 结束语

EIS的开发是一项艰巨而又复杂的工作,而经理信息需求的分析则是EIS成败的关键,本文在如何决定经理信息需求方面做了一点探索。

该方法是从企业角度来分析经理人员的信息需求,往往导致建立企业级的EIS,然而即使是建立部门亦或个人的EIS,该方法也是可行的,它有助于确定EIS目标用户在整个企业信息处理中所处的位置,包括其输入输出,从而有效地确定目标用户的信息需求。

需要说明的是,经理信息需求分析不是EIS开发的全部,在后面的设计实现过程中,如何达到部门间的信息共享、各部门已有异构信息环境如何集成、信息如何组织、如何设计EIS用户界面等都是EIS开发人员、经理人员及其他相关人员面临的重大挑战,需要深入探索和研究。

参考文献

- 1 颜永洪等译. 决策支持系统基础. 福建科学技术出版社,1989年
- 2 黄丽华. 基于规则的企业过程结构优化的理论与方法. 复旦大学博士论文,1996.12
- 3 James C. Wethebe, "Executive Information Requirements: Getting it right", MIS Quarterly/ March 1991
- 4 Michael Hammer & James Champy, 《Reengineering the Corporation - A Manifesto for Business Revolution》, Harper Business, New York, 1993

汽车零部件失效分析的知识模型及描述方法

杨开英 毕常青 张明友 陈兰英

(武汉汽车工业大学电子系 430070)

摘要 通过分析汽车零部件失效分析知识的构成,给出了汽车零部件失效分析的知识模型。利用数据库系统对数据的简单组织方式实现复杂知识的表达,从而能以数据检索实现知识的推理。

关键词 失效分析 知识模型 数据库

1 引言

汽车工业是一个综合性的产业,汽车本身结构的复杂性,汽车用材的多样性,汽车零部件服役条件的苛刻性,决定了汽车零部件不可避免地会出现早期失效,也决定了汽车零部件的失效是一个动态的复杂系统。如何从众多的潜在因素中找出导致失效的根本原因,继而有针对性地采取改进措施,是一件难度很大的工作。因此,建立失效分析专家系统辅助决策是必要的。

专家系统的质量在很大程度上取决于所存储的知识的有效性和知识表达是否有利于推理的检索、匹配。因此首先必须研究汽车零部件失效分析知识的构成问题。研究知识的构成其实质是对知识进行抽象并能以合理的形式进行描述。

2 建立知识模型

知识模型是失效分析中各种知识的抽象形式,并且这种形式能以精确的表示来实现。我们从失效模式及严重程度分析的角度出发,找出具有典型意义的子集,以汽车轴类零件的断裂失效作为系统的原型,建立知识模型。

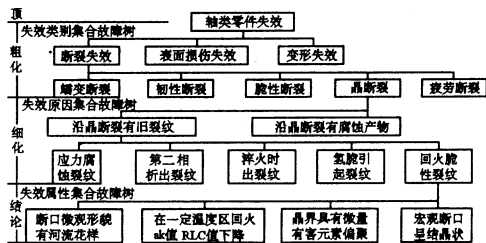


图1 轴类零件失效故障树

故障树分析法是失效分析中行之有效的经典方法。它将系统故障形成原因由总体至各个部分按树枝状逐渐细化,并以逻辑符号连接组成一棵或多棵能描述汽车系统性能故障的逻辑图。为了真实反映汽车零部件失效的信息,对复杂的汽车零部件的失效事件按不同的范畴(如失效类别集合、失效原因集合、失效属性集合、防止失效措施集合)建立若干棵故障树。以轴类零

件失效为例描述故障树如图1所示。

一个故障树就是一个层次结构,层次结构表现了知识的分类特性,是控制型知识。这类知识主要用于推理时引导搜索的方向,以缩小搜索的范围,提高搜索的效率。图中的一个节点则对应于一组个性知识,该组知识体现了失效分析领域中长期积累起来的原理、经验,及行之有效的措施等启发性知识。每条个性知识用产生式规则表示,规则的前提表示引发事件,规则的结论表示引发的结果,规则的结论既可是进入下层结点的入口,又可激活另一棵故障树,从而体现了针对同一失效事件,从不同侧面所建立的故障树之间的联系。位于上层树的叶节点虽然都能进入下层细化的树,但所对应的子树不同,每棵子树中的节点所对应的规则组不同。

例如,图1的根节点对应的一组规则是:

R1: IF (N > 2) THEN 断裂失效

R2: IF (N < 2) ∧ 真实尺寸 ≈ 原始尺寸 ∧ 重量减少
THEN 表面损伤失效

R3: IF (N < 2) ∧ 真实尺寸 > 原始尺寸 THEN 变形失效

其中, ∧ 表示前提事件之间是合取关系。

由此可知,知识模型的描述分为两部分:一是层次模型的描述,二是每个节点所对应的一组知识的描述。

3 知识模型描述

为了快速开发原型子集我们在成熟的关系数据库管理系统支持之下,用数据库的形式表现知识,从而能以简单的数据检索实现推理。

(1) 将层次结构用关系表示

关系模式 NODE_DB 用来描述节点间的层次联系,其属性和意义如下:

Node_code	Node_name	Sub_node
节点号	节点名	父节点号

node_code 是关键字。

(2) 节点对应的知识描述

节点对应的知识用一组规则表示。为了使规则的表达规范化,我们约定规则的条件和结论中所包含的断言用合取连接,而对于析取的断言,可用两条规则表

示,例如:

IF A V B THEN C → IF A THEN C, IF B THEN C
IF A THEN C V D → IF A THEN C, IF A THEN D

规则的条件和结论间的联系用规则关系模式
RULE_DB 描述,其属性和意义如下:

Rule_code	Fact_code	Type	Indicia
规则号	事实号	类型	标记

Typed 值为“0”时表示是条件,“1”表示是结论,
“2”表示该结论是本规则组终结;indicia 标识是否使用
及是否被匹配。同一条规则的条件和结论具有相同的
rule_code,同一个断言既可能是某规则的结论也可能是
另一条规则的条件,所以其关键字是(rule_code,
fact_code)。

每个节点对应的规则关系模式虽然结构相同但以
不同的节点名命名,仅当进入某节点时才打开对应的
规则库。

关系模式 RULE_CF_DB 描述每条规则的可信
度,其如属性和意义下:

Rule_code	Rule_CF
规则号	规则的可信度

规则的条件、结论及其所对应的事实用字典关系
模式 DICT_DB 描述,其属性和意义如下:

Fact_code	Fact	Feature	Trans	Prompt	Fact_CF
事实号	事实名	特征	内部表示	提示	事实的可信度

Fact_code 是关键字,Feature 的值和 Prompt 值
的联系如下:

Feature	Prompt
C	空或字符
F	函数或过程名
N	与其相反的事实号

当被证实的结论是某规则库的终结时,若是本次
推理的目标则将 DICT_DB 中 prompt 的值置为
END;否则是其下层节点的编号,以使推理能进入下
一层。当该事实是可观察事实时,Fact_CF 由用户交
互输入;当是可推演事实时,Fact_CF 为使用非精确
推理更新算法求出的计算结果。Fact_CF 的初值为 0,
当其值大于 0 时,表示该事实已被匹配时,以避免断言
的重复求证。

关系模式 WORK_DB(Node_code, Rule_code,
Fact_code)用来记录推理的路径。

4 示例

以轴类零件失效分析故障树为例进行说明。

层次结构库 NODE_DB,其节点采用从上到下,
从左到右的顺序编号。

Node_code	Node_name	Sub_node
000	轴类	
001	断裂	000

002	损伤	000
003	变形	000
004	蠕变	001
005	韧性	001
006	脆性	001
007	晶断裂	001
008	疲劳	001
...

根节点对应的规则库表示如下:

Rule_code	Fact_code	Type	Indicia
0001	0001	0	
0001	0002	2	
0002	0003	0	
0002	0004	0	
0002	0005	2	
0003	0003	0	
0003	0006	0	
0003	0007	0	
0003	0008	2	

上述断言表示如下:

Fact_code	Feature	Prompt	Fact_CF
0001	C	表面分裂数大于 2	
0002	C	001	
0003	N	0001	
0004	C	真实尺寸大于原始尺寸	
0005	C	002	
0006	C	真实尺寸近似原始尺寸	
0007	C	重量减少	
0008	C	003	
...	

5 验证

我们以正向推理及深度优先搜索法进行推理。推
理通过对 NODE_DE、及 NODE_DB 中节点对应规
则库的和 DICT_DB 等库中记录的检索而实现。由于
采用了多种索引,如规则库的主索引关键字是:Rule
_code+Type+Fact_code,所以相同规则的条件顺序
相邻而结论则紧跟其后,因此按此索引顺序可以快速
访问一条规则。其次,每当一个断言被匹配或被否定均
在规则库中进行了标注,因此使得推理的过程中不必
再对已标注的条件进行匹配。第三,DICT_DB 库中凡
是已证实的事实也进行了标注,使用过滤操作也可缩
小搜索的范围。综上可知,推理具有较高的效率。

笔者在开发《汽车轴类零件失效分析专家系统》的
原型系统时,用 Foxpro 作工具,容易调试,开发周期
短,不失为一种实用的方法。

参考文献

- 1 毕常青,杨开英,张明友.汽车零件失效分析专家系统的研究.计算机应用研究,1994.4
- 2 吴顺祥.基于数据库的专家系统的研究.计算机工程与应用,1996.5

任意具有 $Z=F(X,Y)$ 形式空间曲面的生成

周术诚 王宜怀 梁 忠

(福建林学院计算机科学系 南平 353001)

摘 要 本文探讨了具有 $Z=F(X,Y)$ 形式空间曲面生成的通用方法,即可通过直接输入 $F(X,Y)$ 及有关参数,便可以在屏幕上产生相应空间曲面。可广泛应用于特殊图形制作、空间曲面辅助教学、计算机图形学教学等方面。

关键词 空间曲面 消隐 投影 坐标变换

1 引言

图形绘制是应用软件封面设计的一个重要部分,利用 $Z=F(X,Y)$ 形式的空间曲面,产生比较复杂的、美观图形,存于磁盘,是应用软件封面产生的一种重要方法。但如何产生一幅满意的图形呢?不同的曲面方程或者同一个曲面方程选择不同的观察点,都将得到不同的投影图像,但是这些不同的图像利用计算机绘制成平面投影时,绘制的过程是完全相同的。通常在编程时,人们往往只针对于某一具体曲面方程和一个固定观察点,在程序中利用赋值语句直接写上曲面方程表达式和赋予参数变量,如果要改变曲面方程表达式或参数,就必须改变程序中相应赋值语句,这样,不利于那些对程序设计不熟悉的用户的使用,也不能满足连续运行的要求,更不利于同一曲面方程在不同观察点时得到的投影图像的比较。因此,如何使得具有 $Z=F(X,Y)$ 形式的曲面可以由键盘直接输入方程和观察点参数便可产生相应图形,就成为值得探讨的问题。这样,不同形式的曲面方程可以直接输入,反重执行;或者对一些特殊的曲面方程所生成的图形以文件的方式贮存起来,在需要时直接取出显示,以满足计算机辅助教学以及一些图形界面设计的需要。

2 图形投影算法

由于 $Z=F(X,Y)$ 是一个处于三维坐标系中的立体曲面方程,要把它所对应的图形在一个二维平面上显示出来,在程序设计时,一般要经过以下两个算法进行处理,才能在屏幕上得到一幅清晰的投影图:一是要通过坐标变换,即将三维坐标转换成二维坐标;二是要经过消隐处理,即将不见部分隐去不画。

2.1 坐标变换

计算机屏幕是一种二维显示设备,而空间曲面是处于三维坐标系中,因此对于三维图形的平面投影显示,就要把三维坐标转换为对应屏幕的坐标,同时考虑观察点和观察物与屏幕的距离、观察点的方向不同,都

将得到不同的投影图。因此,将观察点所处的位置用极坐标表示。它们的坐标关系如图 1 所示。观察物所处的坐标为标准坐标,观察点 P 的标准直角坐标为 (X,Y,Z) ,其在标准坐标系中的极坐标为 (ρ,θ,Φ) ,投影面与观察点距离为 D 。

ρ ——观察点与标准坐标原点 O 的距离

θ —— OP 在 XY 平面投影与 X 正方向间的夹角

Φ —— OP 与 Z 轴正方向间的夹角

观察点坐标系的坐标方向为: Z_e 轴指向 O 点, X_e 轴指向右方, Y_e 指向上方,由于观察点的位置与投影有直接的关系,因此,应先找出观察点坐标与标准坐标关系,再找出观察点坐标与屏幕坐标的关系。经过坐标的旋转变换,可得观察点坐标 (X_e,Y_e,Z_e) 与标准坐标 (X,Y,Z) 的关系为

$$X_e = -X \cdot \sin(\theta) + Y \cdot \cos(\theta)$$

$$X_e = -X \cdot \cos(\theta) \cdot \cos(\Phi) -$$

$$Y \cdot \sin(\theta) \cdot \cos(\Phi) + Z \cdot \sin(\Phi)$$

$$Z_e = -X \cdot \sin(\Phi) \cdot \cos(\theta) -$$

$$Y \cdot \sin(\theta) \cdot \sin(\Phi) - Z \cdot \cos(\Phi) + \rho$$

则观察物某点的投影屏幕坐标 (S_x,S_y) 为

$$S_x = D \cdot (X_e/Z_e)$$

$$S_y = D \cdot (Y_e/Z_e)$$

对于计算机屏幕,由于左上角坐标为 $(0,0)$,则可以根据显示器的分辨率修正 S_x,S_y 的值,以使图像位于屏幕中部及合适的高宽比,便于观察,即使投影屏幕坐标为

$$S_x = D \cdot (X_e/Z_e) + m1$$

$$S_y = -r \cdot D \cdot (X_e/Z_e) + n1$$

若显示器的分辨率为 $m \times n$,则以上两式中, $m1$ 可以取 $m/2$ 左右, $n1$ 可以取 $n/2$ 左右, r 取 0.5 左右。为了更具有通用性,这些参数可以从键盘输入。

因此,确定了观察点极坐标 (ρ,θ,Φ) 和投影面与观察点的距离 D 后,由 $Z=F(X,Y)$ 函数可以算出曲面上某点坐标 (X,Y,Z) ,即可以由以上式子算出对应的屏幕坐标 (S_x,S_y) 。

2.2 消隐算法

在二维平面上生成三维非透明物体图像时,即要决定每点或线、面在平面上的位置,而且要决定哪些是可见的,哪些是遮住的不可见的,可见的部分需要显示,不可见的部分则不必显示。因此,在程序设计时,必须作消隐处理,即找出并消除不可见的部分。消隐算法有许多种,如:物体空间算法、图像空间算法、列表优先算法等。本文采用的是极大极小值测试算法。

对应 $Z=F(X,Y)$ 形式的空间曲面上每点 (X,Y,Z) ,都对应屏幕上一一点 (S_x,S_y) ,但对应同一横轴坐标 S_x ,可以对应多个纵坐标 S_y ,因此必须记录下已经显示的各点中,对应同一个横坐标 S_x 值的纵坐标的最大值 Y_{max} 和最小值 Y_{min} ,每次画点时,都应该以当前的纵坐标值与同一个横坐标值所对应的最大值 Y_{max} 和最小值 Y_{min} 进行比较,若当前的纵坐标值介于 Y_{max} 和 Y_{min} 之间,则说明该点是被遮住的,不可显示,则不必画出,否则,该点是可见的,应该画出。

因此,对于分辨率为 $m \times n$ 的显示器,必须用 m 个变量存放最大值和最小值。为了做到这一点,可以定义两个数组,数组的下标变量对应屏幕上每点的横坐标,范围是 $m-1$,即 $Y_{max}[m-1]$ 和 $Y_{min}[m-1]$ 。

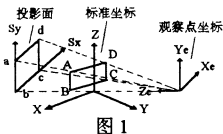


图1

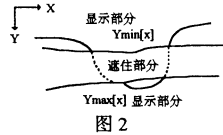


图2



图4

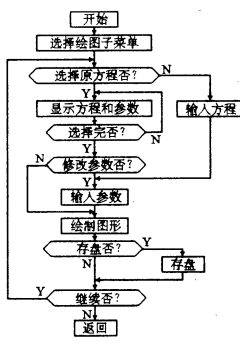


图3

由于每改变一个实际坐标 (X,Y,Z) ,都将得到一个新的屏幕坐标 (X_p,Y_p) ,因此,在实际消隐处理时,应该把当前点到上一次作图点之间的连线上各点的纵坐标值与对应的最大值和最小值进行比较,以决定是否显示。如图2所示。

3 程序实现

在具体程序设计时,如果仅绘制任意曲面投影,可以有多种方法实现,如:应用某种语言的自生成技术、数据库的链接技术等。从通用性和灵活性出发,本文采用了数据库链接技术,把输入的方程作为数据库的一

条记录,然后并入程序处理。能实现以下功能:1)记录原来输入的、已经绘制出满意图像的方程和参数;2)以菜单方式浏览原方程及选择或进入输入方式;3)保存满意的曲面图像,以便于直接调用。

主菜单包括浏览、绘图、和取图三个部分。在浏览时,可以查看已经输入过的方程、参数和本方程是否有保存图像以及原来保存的图像文件名。绘图子项完成方程和参数选择或输入工作,并且完成绘图、存图功能。取图子项功能为:取出某方程和固定参数对应图像,并在屏幕上显示出来。本文主要阐述图形的生成方法,因此,绘图子项程序框图如图3所示。

4. 程序运行实例

例如,要画

$Z=14 * \cos(0.15 * (X * X + Y * Y)) * \exp(-0.04 * (X * X + Y * Y))$ 的投影图,则在程序运行后,选择输入子项,提示如下(底下有画线的为输入部分):

请输入曲面方程:

$Z=F(X,Y)=14 * \cos(0.15 * (X * X + Y * Y)) * \exp(-0.04 * (X * X + Y * Y))$

请输入物体到屏幕的距离 $\rho=200$

请输入观察点到屏幕的距离 $D=2000$

请输入OP投影与X轴的夹角 $\theta=0.1$

请输入OP与Z轴的夹角 $\Phi=1.3$

以上方程和参数确定的曲面方程在屏幕上的投影如图4所示。

5 结束语

计算机图形系统至少应该具有如下功能:二维、三维图形的绘制及三维图形的投影图;建立图形数据库,对图形具有存取操作;对各种图形进行平移、旋转和缩放等变换;可以对图形进行增、删、改等操作;能对零件图标注尺寸、公差分析及对立体图形显示的动态模拟等。本文只叙述了具有 $Z=F(X,Y)$ 形式的曲面投影的作图及隐藏线的消除的方法,虽然,这只是图形系统中的一个子集,但是,我们能以此为出发点,根据实际需要,分析各种图形软件设计的关键技术,与硬件系统结合起来,开发出适合自己的图形系统。

参考文献

- 1 R. E 迈斯著, IBM-PC 微型计算机绘图. 海洋出版社出版
- 2 孟明辰等编著, 计算机辅助图形设计. 机械工业出版社出版
- 3 李兰友等编著, TURBO C 实用图形程序设计. 天津科技翻译出版公司出版

用 JavaScript 改变 Internet 浏览器的界面

杨建华

(清华大学自动化系 北京 100084)

摘 要 Internet 的使用越来越广泛,大部分用户面对的是 Web 浏览器,许多浏览器的界面过于复杂,并经常使大量的用户感到多余信息太多。为此,本文用 JavaScript 语言改进了浏览器的界面形式,在极端情况下可以只剩下一个没有菜单和工具栏的窗口,而其功能和原浏览器相同。

关键词 Internet JavaScript 界面

1 引言

Internet 已经渗透到教学、科研、生产、商业、医疗等各个领域,Internet 的发展甚至改变了人们的生活和生活方式。就 Internet 的使用来说,客户端的 Web 浏览器和用户之间的关系最为密切。Web 浏览器是用户从 Internet 获取信息的主要手段,目前最为人们熟悉的浏览器当属 Netscape 公司的 Netscape Navigator 系列(包括 Gold 系列)和 Microsoft 公司的 Internet Explore 系列。这些浏览器的功能越来越强大,界面也越来越丰富。但是在实际使用过程中,我们会注意到,浏览区域经常被压缩得很厉害,例如在 Internet Explore 3.0 中,如果选取全部工具栏中的选项,菜单和工具栏占的屏幕面积确实相当可观。因此问题是我们不能对它们进行修改,一个极端的情况是用户使用的浏览器没有任何菜单和工具栏,预先设置好初始页,对 Internet 进行访问。解决这个问题可以有几种方案:第一种方案是向提供浏览器的公司提出建议,由它们提供改进的浏览器可以修改菜单和工具栏,满足用户的需要,不过这一方面需要时间等待,另一方面这些公司未必能满足你的要求;第二种方案是自己开发,特别像 Microsoft 公司提供的 VisualC++ 4.2 版,具有全面支持 Internet 开发的功能,新增的许多类都是面向 Internet 的,但是要开发这样的浏览器十分复杂,而且局限于 Microsoft 所能提供的功能,事实上我们都看到过 Microsoft Internet Explore 浏览器存在的缺点,自己去开发显然困难多多。第三种方案是我们能不能用简单的方法对浏览器进行改造,特别是在许多时候我们甚至不需要原来菜单,或者只留下按钮栏,或者只留下 URL 地址栏,要实现这些功能,我们首先想到了 JavaScript。

2 JavaScript 开启新窗口

JavaScript 可以说是网络上的 BASIC 语言,但是 JavaScript 是面向对象的语言,每种对象都封装了各

种属性和许多方法。结合我们的问题,很容易想到的方法是 Open(),在当前窗口可以用 Open 方法实现打开一个新窗口,而新窗口的 Web 浏览功能是和原窗口相同的,所不同的是我们可以设置新窗口的各种属性了,这些属性包括 toolbar、location、directories、status、menubar、scrollbars、resizable、copyhistory、width、height。属性的说明如下:

●属性 toolbar=yes 用来创建标准的 Navigator 按钮栏,如“上一页(Back)”和“下一页(Forward)”。

●属性 location=yes 用来创建 URL 地址输入域。

●属性 directoried=yes 用来创建标准的 Navigator 目录按钮,如“What's New”和“What's Cool”。

●属性 status=yes 用来创建窗口底部的状态条。

●属性 menubar=yes 在窗口顶部创建菜单。

●属性 scrollbars=yes 创建水平和垂直滚动条,滚动条的出现取决于文本是否超出窗口范围。

●属性 resizable=yes 允许用户改变窗口大小。

●属性 copyhistory=yes 把当前窗口的历史记录拷贝到新窗口。

●属性 width=像素数 按像素数目规定窗口宽度。

●属性 height=像素数 按像素数目规定窗口高度。

在上面的属性中除了 width 和 height,其它的属性当等于 no 时就将在新窗口中屏蔽该属性对应的组件。下面是一个例子。

```
//filename, test. html
<html>
<script language="LiveScript">
function WinOpen(){
msg = open ("","DisplayWindow", "toolbar = no, directories = no,
menubar = no, status = yes, scrollbars = yes, resizable = no,");
msg. document. write("<html>");
msg. document. write("<HEAD><TITLE>这是我的主页</TITLE></HEAD>");
msg. document. write("<CENTER><h2>这是<B>JavaScript</B>
```

```

开的窗口! </h2></CENTER>");
msg.document.write("<body>");
msg.document.write("<hr>");
msg.document.write("<P><A HREF='http://www.cernet.
edu.cn/'>中国教学网</A></P>");
msg.document.write("</BODY>");
msg.document.write("</HTML>");
}
</SCRIPT>
<body onLoad="WinOpen()">
</body>
</html>

```

运行 test.html (运行前建立 html 文件与浏览器之间的联系, 或者运行浏览器后调入 test.html), 将建立一个新窗口, 充满整个屏幕, 除了状态条之外没有其它任何组件, 滚动条只是在需要的时候出现。点取“中国教学网”, 将进入 CERNET 的主页, 在窗口内的所有操作, 和原有浏览器并无区别, 但界面给人的感觉确实很新颖。

3 余留问题的解决

余下的问题是在这个新窗口后面还留有原先的浏览器窗口, 能不能把它自动关闭掉, 这样做也许更完善, 测试证明手工关闭该窗口并不影响新建窗口的工作。我们找到 JavaScript 的方法 close(), 先试一下在语句 <bodyonload="WinOpen()"> 后面加入

```

<script language="LiveScript">
self.close()
</script>

```

再运行 test.html, 问题出现了, 浏览器启动后又立即关闭了, 显然是在函数 WinOpen() 运行结束之前, 已经执行了 self.close()。为了解决这个问题, 可以考虑在执行 self.close() 时, 加入 WinOpen() 运行成功的条件, 但是有些复杂, 我们可以采取简单的方法, 即采用方法 setTimeout()。在语句 <bodyonload="WinOpen()"> 后面加入

```

<script language="LiveScript">
window.setTimeout('self.close()', 1000)
</script>

```

经过测试, 成功了。得到的窗口几乎分不清是 Netscape 公司的还是 Microsoft 公司的, 当然进入几个主页后会发现, 从不同浏览器继承的新窗口显示页面形式还是不同的。

4 综合测试

最后我们给出一个完整例子

```

//filename:mainpage.html
<html>
<script language="LiveScript">
function WinOpen(){
msg=open("", "DisplayWindow", "toolbar=yes, directories=no,
menubar=no, status=yes, scrollbar=yes, resizable=no, ");
msg.document.write("<html>");

```

```

msg.document.write("<HEAD><TITLE>这是我的主页</TITLE></HEAD>");
msg.document.write("<CENTER><h2>这是<B>JavaScript</B>
开的窗口! </h2></CENTER>");
msg.document.write("<body>");
msg.document.write("<applet>");
msg.document.write(" code=testspot.class");
msg.document.write(" id=testspot");
msg.document.write(" width=500");
msg.document.write(" height=100");
msg.document.write(" hspace=100");
msg.document.write("<param name='message' value='. 欢迎
访问 XXX 的主页.'>");
msg.document.write("<param name='font' value='黑体'>");
msg.document.write("<param name='font_size' value='
40'>");
msg.document.write("<param name='red' value=255>");
msg.document.write("<param name='green' value=0>");
msg.document.write("<param name='blue' value=0>");
msg.document.write("<param name='spot_red' value=
255>");
msg.document.write("<param name='spot_green' value=
255>");
msg.document.write("<param name='spot_blue' value=
255>");
msg.document.write("</applet>");
msg.document.write("<hr>");
msg.document.write("<<P><A HREF='http://www.cernet.
edu.cn/'>中国教学网</A></P>");
msg.document.write("</BODY>");
msg.document.write("</HTML>");
}
</SCRIPT>
<body onLoad="WinOpen()">
</body>
<script language="LiveScript">
window.setTimeout('self.close()', 1000)
</script>
</html>

```

在这个例子中, 我们附带测试了 Java Applet 在新窗口中的运行, 这个 Applet 的例子是许多 Java 参考书上的打光招牌小应用程序。结果在 Netscape Navigator3.0 以及 Gold3.0 上都工作正常, 但是在中文 Microsoft Internet Explore3.0 新创建的窗口上 Java Applet 居然运行不了, 原因不是它不支持 Java, 因为不用上面的例子, 直接调用 WinOpen() 生成的 HTML 文件是可以运行该 Applet 的。也许在许多主页上建议使用 Netscape 公司的浏览器不是没有道理。

另外应该注意的是方法 setTimeout() 中第二个参数的设置, 经过测试 1000 是比较冗余的, 在 Netscape Navigator3.0 中测试时, 设置成 1 也能正常工作, 但是在中文 Microsoft Internet Explore3.0 中设置成 500, 工作不稳定。我们测试的硬件平台是 Pentium120/

16M。在最后的例子中,我们把属性 toolbar 改成了 yes,主要是按钮“上一页(Back)”和“下一页(Forward)”在实际操作时十分方便。

5 小结

研究上述方法改变 Web 浏览器的界面的结果,我

们认为至少有二方面的价值:一是对精通 Web 浏览器的用户来说,扩大了实际浏览器的使用面积;二是如果我们要开发一个 Internet 应用,交给企业普通职工使用,界面的充分简单是十分重要的,它不仅可以减少误操作,而且可以缩短培训时间。

不精确推理模型和算法的描述及应用探讨

吴 陈

(华东船舶工业学院电子与信息系 镇江 212003)

摘 要 本文把不精确推理模型分为两种类型:评分和非评分不精确推理模型。通过引进特定的算子,将它们进行统一,并研究了这种统一形式中不确定性因子的传播算法,文中将不确定性因子分解成可信度与不可信度两种成份,是一种新的观点和提法。文中还把理论结果应用到了具体的问题求解中,进行了有益的应用探讨。

关键词 专家系统 不精确推理 传播算法

1 引言

随着不精确推理技术的研究和专家系统应用领域的拓展,所处理的信息越来越庞杂^[1,2]。大量不确定性、不精确性、不完全知道、模糊性、不完备性、不一致性甚至时变性的信息面临着处理方法的选择和更新。如何寻找更为合适的描述方法,进一步分析不精确推理模型,统一概括不精确推理模型及算法?这是一项十分有意义的工作。

在本文中,主要对基于规则的不精确推理模型及算法进行研究。由于引起证据的不确定性及人们具有的知识的不确定性是多方面的,因此处理的方法、推理的模式等亦都是千差万别的。就目前所使用的各种不精确推理模型来看,笔者将基于规则的不精确推理模型分为两类:一类是评分不精确推理模型,另一类是非评分不精确推理模型。主观 Bayes 方法、可能性理论、证据理论等均属于评分不精确推理模型,而纯逻辑判断推理模型则属于非评分不精确推理模型。

对于基于规则的评分推理模型来说,每个规则都有一个可信度因子。每个证据也有一个可信度因子。这些因子一般来讲都是属于某个线序空间,互相之间可以两两比较大小。但是对于非评分不精确推理模型来说,每个规则的可信度不一定是单纯的数值,而有时可能是某个集合或结构中的成份元素。因此两个因子之间不一定存在严格的可比较性,甚至根本上就没有严格的大小或次序关系。换句话说,因子间的关系不再是一种线序关系(或称全序关系),而可能只是一种偏序关系(或称半序关系),或者更一般的次序关系。对于这

种非评分推理模型,如何建立相应的推理算法或合成规则,则是过去在专家系统的不精确推理模型中未曾系统地研究过的问题。

在本文中,笔者首先建立推理模型的代数系统描述,将非评分不精确推理模型用算子变换为其可信度因子为可进行计算的形式(这既可以说是对不精确推理模型的深层次研究,又可以说是将各类不精确推理模型进行统一化的一个过程或步骤),然后对推理网络上的信度因子传播算法的算式进行了研究,并将其应用到一个具体的网络计算实例中。

2 定义和术语

定义 1 设 G 为证据和假设的集合, W 为某个序空间, $R \subseteq G^2 \times W$ 。则称三元组 (G, W, R) 为一个推理网络。其中 G 的元素(称为结点)既可能是证据,又可能是结论或中间结论。当为中间结论或结论时,又可分为与结点及或结点。对于 $(u_1, u_2, W) \in R$, 称如果 u_1 , 则 u_2 的可信度为 W 。

定义 2 对于 (G, W, R) , 定义 W 上两种抽象算子 \oplus 和 \odot , 使得 (W, \oplus, \odot) 构成满足下列运算的代数系统:

I) 封闭性: 对任意的 $x, y \in W$, $x \oplus y \in W$, $x \odot y \in W$;

II) 结合性: 对任意的 $x, y, z \in W$, 有 $x \oplus (y \oplus z) = (x \oplus y) \oplus z$, $x \odot (y \odot z) = (x \odot y) \odot z$;

III) 交换性: 对任意的 $x, y \in W$, 有 $x \oplus y = y \oplus x$, $x \odot y = y \odot x$;

IV) 分配性: 对任意的 $x, y, z \in W$, $(x \oplus y) \odot z = x \odot z \oplus y \odot z$, $x \odot (y \oplus z) = x \odot y \oplus x \odot z$;

V) 有两个特殊的元素分别叫做零元 θ 和 ϵ 元 e , 满足 $\theta \oplus x = x \oplus \theta = x$, $\theta \odot x = x \odot \theta = \theta$, $e \odot x = x \odot e = x$,

对任意的 x 都成立;

VI) 对于每个元素 x 都有相对应的负元, 记为 $-x$, 满足 $-x \oplus x = x \oplus -x = \theta$ 。通常地, 称 \oplus 为广义的加法, \odot 为广义的乘法。

定义3 设有规则 IF E THEN H (u, v), 其中 E 是证据, H 是假设或结论。u 和 v 都是规则的不确定性因子(或称规则强度)。称 u 为“如果证据 E 为真, 则假设 H 为真”的依赖程度。称 v 为“如果证据 E 为真, 则假设 H 为假”的依赖程度。

在定义3中, 规则的不确定性因子对 (u, v) 是由两个因子组成的。u 有时也称为支撑因子。v 称为干扰因子。笔者的这种提法与一般的基于规则的推理的不确定性因子的假定不同, 它更恰当地表示了现实世界中的生克关系。

定义4 对每一个规则: IF E THEN H (u, v), 称 MBS(H, E) 为因证据 E 的出现而(增加)对假设 H 为真的信任(增长)度; MDS(H, E) 表示因证据 E 的出现而(增加)对假设 H 为假的不信任(增长)度; $CF(H, E) = MBS(H, E) - MDS(H, E)$ 称为规则的(绝对)可信度; $NCF(H, E) = MDS(H, E) - MBS(H, E)$ 称为规则的(绝对)不信任度。

容易看出, MBS(H, E), MDS(H, E), $CF(H, E)$, $NCF(H, E) \in W$ 分别都是 W 中的元素。

在实际的推理中, 规则的绝对信任度应还与证据 E 的信任度有关, 因而是两者的运算合成结果, 分别可对应地用 MBS', MDS', CF' 和 NCF' 来表示。

在评分不精确推理模型中, W 通常进一步是一个全序空间。

3 非评分不精确推理模型

仍以三元组 (G, W, R) 来表示非评分不精确推理模型, 其中 $R \subseteq G^2 \times W^*$, 这里 W^* 表示 W 上任意个元素所构成的序列的集合。或者按照形式语言中的提法, W^* 就是以文字集 W 上的字母所构成的字符序列(或称单词)的集合。因而, 这种推理网络将自然地与形式语言中的模型相对应。

这样, 在非评分不精确推理模型中, 当一条推理规则 IF E THEN H (u, v)

给定后, 其不确定性因子偶对 (u, v), 依问题领域不同, 可信度 u 和不可信度 v 可能分别是 W 中的一个元素或元素序列。例如, 多个医生对同一诊断推理规则的肯定程度与不支持的程度就可能形成一个程度判断的非数值序列, 有待进一步首先将非数值值映射为全序空间中的元素如数值值, 然后再对多个结论进行综合, 亦即需要对 u 和 v 进行信息融合, 最后才能得到该规则的可信度和不可信度。这一过程可通过下面的定义进行形式生成。

定义5 对于一般的空间 W, 定义 W^* 上的信度函数 $m_1: W^* \rightarrow L$, 若 m_1 满足下列性质:

$$\begin{cases} m_1(x \oplus y) = m_1(x) \oplus m_1(y) \\ m_1(x \odot y) = m_1(x) \odot m_1(y) \end{cases}$$

其中 $x, y \in W^*$, 则称 m_1 是满足同态性质的一种映射。这里 $L = [0, 1], (0, 1)$ 或 $(-\infty, +\infty)$; \oplus 和 \odot 在 W^* 上的定义与在 L 中的定义一般是不同的。但它们分别均是二元运算, 并且 (W^*, \oplus, \odot) 与 (L, \oplus, \odot) 分别形成满足特定性质的代数系统。但为方便起见, 这里使用了相同的记号表示形式。

定义6 当 W 本身是一个序空间时, 在 W^* 中取 \oplus 和 \odot 分别为 \vee 和 \wedge , $+$ 与 \times 及 \max 与 \min , 则分别称该推理系统为逻辑推理系统、数值计算推理系统以及模糊推理系统。

从定义5可以看出, 实质上对非评分模型, 要投入实用, 必须首先进行变换或映射, 使之成为评分模型。但使用者应绝对把握两者之间的差异。

4 抽象网络推理的计算规则

由于笔者将推理规则中的可信度因子进一步分解为支撑因子与非支撑因子, 模仿了生克关系, 因此, 在此有必要给出其网络推理计算规则。又由于非评分模型在实用时应转换为评分推理模型, 即使不转换, 也是在广义加法和广义乘法支撑下的代数系统中进行演算。因此, 在下面的推理计算中, 使用的算符也是抽象给定的。

(1) 当证据为单个条件时, 由规则: IF E THEN H (u, v), 可得到规则的一个可信度 $CF(H, E)$ 和证据 E 的一个可信度 $CF(E, E')$ (E' 表示得到 E 之前的观察), 后者表示在所有观察 E' 的条件下, E 为真的可信度。但一般来讲, 整个规则的可信度是与这两者都有关的。在具体的推理计算时, 应从可信度与不可信度开始计算。其公式如下:

$$MBS'(H, E) = MBS(H, E) \odot CF(E, E')$$

$$MDS'(H, E) = MDS(H, E) \odot NCF(E, E')$$

$$CF'(H, E) = |MBS'(H, E) - MDS'(H, E)|$$

$$NCF'(H, E) = |MDS'(H, E) - MBS'(H, E)|$$

带有撇记号的 MBS', MDS', CF' 和 NCF' 是规则最终的绝对信度。

(2) 当证据是多个条件的组合情况时

a) 多个证据是由 AND 连接的;

IF E_1 AND E_2 AND \dots AND E_k THEN H (u, v)

若 E_1 AND E_2 AND \dots AND E_k 由所有的观察 E' 得到, 即有

$$MBS(E_1, E'), MBS(E_2, E'), \dots, MBS(E_k, E')$$

及 $MDS(E_1, E'), MDS(E_2, E'), \dots, MDS(E_k, E')$, 则 $CF'(H, E')$ 的计算由下列公式逐步计算得出:

$$MBS(E_1 \text{ AND } E_2 \text{ AND } \dots \text{ AND } E_k, E') = MBS(E_1, E')$$

$$\odot MBS(E_2, E') \odot \dots \odot MBS(E_k, E') = \bigodot_{i=1}^k MBS(E_i, E')$$

$$MDS(E_1 \text{ AND } E_2 \text{ AND } \dots \text{ AND } E_k, E') = MDS(E_1, E')$$

$$\oplus \text{MDS}(E_1, E') \oplus \dots \oplus \text{MDS}(E_k, E') = \bigoplus_{i=1}^k \text{MDS}(E_i, E')$$

$$\text{CF}(E_1 \text{ AND } E_2 \text{ AND } \dots \text{ AND } E_k, E') = \text{CF}(E_1, E') \odot \text{CF}$$

$$(E_2, E') \odot \dots \odot \text{CF}(E_k, E') = \bigodot_{i=1}^k \text{CF}(E_i, E')$$

$$\text{NCF}(E_1 \text{ AND } E_2 \text{ AND } \dots \text{ AND } E_k, E') = \bigodot_{i=1}^k \text{NCF}(E_i, E')$$

$$\text{MBS}'(H, E_1 \text{ AND } E_2 \text{ AND } \dots \text{ AND } E_k, E') = \text{MBS}(H, E_1 \text{ AND } E_2 \text{ AND } \dots \text{ AND } E_k) \odot \text{CF}(E_1 \text{ AND } E_2 \text{ AND } \dots \text{ AND } E_k, E_1)$$

$$\text{MDS}'(H, E_1 \text{ AND } E_2 \text{ AND } \dots \text{ AND } E_k, E') = \text{MDS}(H, E_1 \text{ AND } E_2 \text{ AND } \dots \text{ AND } E_k) \odot \text{NCF}(E_1 \text{ AND } E_2 \text{ AND } \dots \text{ AND } E_k, E')$$

$$\text{CF}'(H, E_1 \text{ AND } E_2 \text{ AND } \dots \text{ AND } E_k, E') = |\text{MBS}'(H, E_1 \text{ AND } E_2 \text{ AND } \dots \text{ AND } E_k, E') - \text{MDS}'(H, E_1 \text{ AND } E_2 \text{ AND } \dots \text{ AND } E_k, E')|$$

$$\text{NCF}'(H, E_1 \text{ AND } E_2 \text{ AND } \dots \text{ AND } E_k, E') = |\text{MDS}'(H, E_1 \text{ AND } E_2 \text{ AND } \dots \text{ AND } E_k, E') - \text{MBS}'(H, E_1 \text{ AND } E_2 \text{ AND } \dots \text{ AND } E_k, E')|$$

b) 当证据是由 OR 连接时:

$$\text{IF } E_1 \text{ OR } E_2 \text{ OR } \dots \text{ OR } E_k \text{ THEN } H(u, v),$$

则当 $E_1 \text{ OR } E_2 \text{ OR } \dots \text{ OR } E_k$ 在所有的观察 E' 下有:

$$\text{MBS}(E_1 \text{ OR } E_2 \text{ OR } \dots \text{ OR } E_k, E') = \bigoplus_{i=1}^k \text{MBS}(E_i, E')$$

$$\text{MDS}(E_1 \text{ OR } E_2 \text{ OR } \dots \text{ OR } E_k, E') = \bigodot_{i=1}^k \text{MDS}(E_i, E')$$

$$\text{CF}(E_1 \text{ OR } E_2 \text{ OR } \dots \text{ OR } E_k, E') = \bigoplus_{i=1}^k \text{CF}(E_i, E')$$

$$\text{NCF}(E_1 \text{ OR } E_2 \text{ OR } \dots \text{ OR } E_k, E') = \bigodot_{i=1}^k \text{NCF}(E_i, E')$$

同样可求得:

$$\text{MBS}'(H, E_1 \text{ OR } E_2 \text{ OR } \dots \text{ OR } E_k, E') = \text{MBS}(H, E_1 \text{ OR } E_2 \text{ OR } \dots \text{ OR } E_k) \odot \text{CF}(E_1 \text{ OR } E_2 \text{ OR } \dots \text{ OR } E_k, E')$$

$$\text{MDS}'(H, E_1 \text{ OR } E_2 \text{ OR } \dots \text{ OR } E_k, E') = \text{MDS}(H, E_1 \text{ OR } E_2 \text{ OR } \dots \text{ OR } E_k) \odot \text{NCF}(E_1 \text{ OR } E_2 \text{ OR } \dots \text{ OR } E_k, E')$$

$$\text{CF}'(H, E_1 \text{ OR } E_2 \text{ OR } \dots \text{ OR } E_k, E') = |\text{MBS}'(H, E_1 \text{ OR } E_2 \text{ OR } \dots \text{ OR } E_k, E') - \text{MDS}'(H, E_1 \text{ OR } E_2 \text{ OR } \dots \text{ OR } E_k, E')|$$

$$\text{NCF}'(H, E_1 \text{ OR } E_2 \text{ OR } \dots \text{ OR } E_k, E') = |\text{MDS}'(H, E_1 \text{ OR } E_2 \text{ OR } \dots \text{ OR } E_k, E') - \text{MBS}'(H, E_1 \text{ OR } E_2 \text{ OR } \dots \text{ OR } E_k, E')|$$

c) 由证据的可信度与规则的可信度可计算出结论的可信度:

$$\text{MBS}(H, E') = \text{MBS}(H, E) \odot \text{CF}(E, E')$$

$$\text{MDS}(H, E') = \text{MDS}(H, E) \odot \text{CF}(E, E')$$

$$\text{CF}(H, E') = \text{CF}(H, E) \odot \text{CF}(E, E')$$

$$\text{NCF}(H, E') = \text{NCF}(H, E) \odot \text{NCF}(E, E')$$

$$\text{因而有: } \text{MBS}'(H, E') = \text{MBS}'(H, E) \odot \text{CF}'(E, E')$$

$$\text{MDS}'(H, E') = \text{MDS}'(H, E) \odot \text{CF}'(E, E')$$

$$\text{CF}'(H, E') = \text{CF}'(H, E) \odot \text{CF}'(E, E')$$

$$\text{NCF}'(H, E') = \text{NCF}'(H, E) \odot \text{NCF}'(E, E')$$

(3) 当多条规则具有同一结论或假设时, 则按照规则的合成方式来计算。例如, 有两条规则:

$$\text{IF } E_1 \text{ THEN } H(u_1, v_1)$$

$$\text{IF } E_2 \text{ THEN } H(u_2, v_2)$$

$$\text{则: } \text{MBS}(H, E_1 \text{ Rule } E_2) = \text{MBS}(H, E_1) \oplus \text{MBS}(H, E_2)$$

$$\text{MDS}(H, E_1 \text{ Rule } E_2) = \text{MDS}(H, E_1) \odot \text{MDS}(H, E_2)$$

$$\text{CF}(H, E_1 \text{ Rule } E_2) = \text{CF}(H, E_1) \oplus \text{CF}(H, E_2)$$

$$\text{NCF}(H, E_1 \text{ Rule } E_2) = \text{NCF}(H, E_1) \odot \text{NCF}(H, E_2)$$

5 应用例子

下面以一个特定的例子来体现一下上节计算规则的作用。

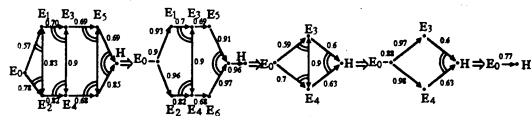


图1 推理网络的可信度计算过程

例 为简单起见, 在推理网络中仅设有可信度因子, 见图1。在图1中, 一条有向弧表示弧尾为证据, 弧头为假设, 弧上的权值表示可信度, 双向弧表示双向性规则。可信度 $W = [0, 1]$, 因而不须作进一步的变换。而且将 \oplus 和 \odot 两种运算分别定义为 $u_1 \oplus u_2 = u_1 + u_2, u_1 \odot u_2 = u_1 + u_2 - u_1 u_2$ 。

由图1可以看到, 推理网络经过逐步演算, 最终求出了从初始证据到最终假设之间的可信度因子。

6 结束语

本文首先对不精确推理网络进行了分类研究, 将其区分为评分不精确推理模型和非评分不精确推理模型。通过变换, 定义相应的映射, 本文将非评分不精确推理模型化成评分推理模型。另外, 本文中规则的可信度区分为支撑和非支撑两种, 模仿了生克关系, 并进一步给出了在不同情况下的网络可信度因子的推理计算规则。最后还给出例子, 所探讨的问题是网络推理中较深层次的问题, 所得到的计算方法对于网络推理具有一般的指导作用。

参考文献

- 1 何华灿. 人工智能导论. 西北工业大学出版社, 1988 年
- 2 N. J. Nilson(石纯一译), 人工智能原理, 科学出版社, 1983 年

本刊启事

为了更快地推动、促进我国计算机产业的迅猛发展, 及时地为计算机科研、开发、生产、管理等部门提供计算机行业的最新发展动向, 加速高新尖端技术、最新软硬件的开发、移植、引进, 及时地为广大读者及计算机爱好者奉献更多更新的计算机专业技术资料, 并使众多作者、译者脱颖而出, 《计算机应用研究》杂志社现正开展优惠出版各类计算机技术专著业务, 欢迎广大作者、译者踊跃投稿。具体出版业务欢迎来函与我刊张钢编辑联系(邮编: 610041 通讯地址: 四川省成都市人民南路四段 11 号附 1 号《计算机应用研究》杂志社)。

《计算机应用研究》杂志社启

iRMX for Windows 与 Windows 的动态数据交换技术

王明春

(东南大学动力系 南京 210096)

摘 要 本文介绍了 iRMX for Windows 实时操作系统与 Windows 操作平台之间利用动态数据通信(DDE)协议进行数据交换的基本思路和总体框架,具体阐述了通信环境的建立和配置。

关键词 操作系统 通信 动态数据交换

1 前言

随着计算机在工业控制系统中的广泛应用,人们越来越追求软件的实时性与美观的人机界面相结合。众所周知,Microsoft 公司推出的 Windows 图形界面操作环境,以其友好的人机界面、强大的图形功能使得操作、学习更加方便,但它的多任务运行是基于消息队列“先来先服务”的原则,没有任务优先级的控制,这就限制了其在实时系统中的响应能力,实时性较差;而 Intel 公司 92 年推出的 iRMX for windows 实时操作系统尽管具有强大的实时处理能力和响应能力,但它没有直接的图形开发功能,人机界面不友善。

能否将这两种系统有机地结合起来,使其兼有 Windows 图形开发和 iRMX 实时多任务的优点? iRMX for windows 提供的 iRMX DDE 库,支持其与 Windows 通过 DDE 进行通信,使 iRMX 强大实时性和 Windows 丰富图形界面的有机地结合在一起,即:把实时的处理、中断、运算、多任务调度等放在 iRMX 侧,而人机界面、实时数据显示和刷新放在 Windows 侧,度等放在 iRMX 侧,而人机界面、实时数据显示和刷新放在 Windows 侧,通过 DDE 完成其间数据通讯。因此,实现 iRMX for windows 与 Windows 间的 DDE 通信成为其中的关键问题。

2 iRMX 与 Windows DDE 通信环境的建立

通信环境的软、硬件配置:

硬件:386 主机(或以上),

≥8M 内存,

≥120M 硬盘,

支撑软件:iRMX for windows 2.0C 操作系统

统

Windows 3.1(标准模式启动)

中文之星

DOS 3.31 以上

编程语言:iRMX 侧选用 ic386

Windows 侧选用 Borlandc++3.1

iRMX 与 Windows 间的 DDE 通信可用于单机系统,也可用于网络环境的多机系统中,由基于 iRMX 侧的 DDE 库和 Windows 侧的 DDE 路由器应用程序(DDE Router)支持。iRMX DDE 库采用以标准网络接口建成的简单应用协议与 DDE 路由器通信,两侧的应用程序则分别与 DDE Router 和 iRMX DDE 库交互通信,从而完成 DDE 数据交换过程。当这种通信处于网络环境中时,需要在总线插槽中插入网卡并加载相应的网络驱动程序;而在单机内两系统间进行通信时,不需要网卡,但仍需加载支持链接的网络驱动程序。本文介绍了在单机系统内不带网卡的 Windows 应用程序与 iRMX 应用程序间的 DDE 通信,如图 1 所示。

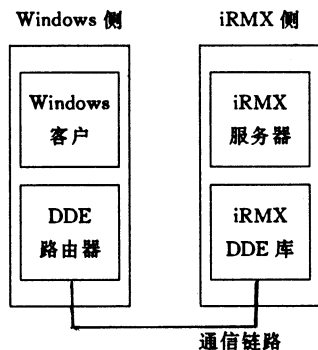


图1 单机内 Windows 客户与 iRMX 服务器间的通信

其中,通信链接是由 iRMX 侧加载的空数据链路作业(ntp4at.job)和网络重定向作业(netrrt.job)以及 DOS 侧加载的 pcnet 驻留程序共同支持的。

需要特别强调的是,进行通信前,在 iRMX 侧配置文件 loadinfo 中一定要加载标准模式 Windows 作业(smw.job),该作业允许 iRMX 操作系统和标准模式的 Windows 共存,并且将其作为 iRMX 操作系统下的一个任务来运行,而 Windows 侧需在 win.ini 初始化文件中加入:

[DDE Router]

Pcname=name

其中,DDE Router 和 Pcname 为关键字,name 是赋

予 PC 机的名字,用户可以任意取名,DDE 程序使用该名字来调用 DDE 路由器。在进入 Windows 运行 DDE 通信前,首先要启动 DDE Router 应用程序和 iRMX for Windows 在 Windows 下的仿真终端程序 wterm

3 iRMX 与 windowsDDE 通信的实现

众所周知,Windows 支持进程之间进行动态数据交换(DDE),两个 Windows 程序之间进行 DDE 对话时,这两个程序被称为“客户”和“服务器”。由于 iRMX DDE 库的支持,iRMX 与 Windows 之间也可以建立类似 Windows 内部的 DDE 通信机制,iRMX 与 Windows 之间的一个 DDE 对话可以用“机器名”、“服务名”和“主题名”来标识,其中,机器名就是在 win.ini 文件中[DDE Router]来定义的 Pcname,两系统之间可能建立多个 DDE 对话通道,但每个通道需要不同的标识。iRMX 与 Windows 之间的 DDE 对话有三种基本类型:

冷链接:在 DDE 对话建链后,只有当客户主动请求数据时(发送 WM_DDE_REQUEST 消息),服务器才发送数据给客户,在这一过程中服务器始终处于被动地位,对话结束后要发 WM_DDE_TERMINATE 拆链。

热链接:与冷链接相同的是要进行对话的建链与拆链,但中间数据交换过程不同,热链接在对话建链后还要对每一数据项建立链接,若被链接的数据项有变化,服务器会主动将变化了的数据发送给客户,对话结束时还要做数据项的拆链。

温链接:温链接结合了冷链接和热链接的特点,利用热链接客户对每一数据项建立链接,但这种链接标志说明客户只希望被通知数据发生了变化,但并不立即接收新数据,所以,当数据发生变化时,服务器只通知客户数据已变化,若客户希望获取这个数据项,与冷链接一样,使用 WM_DDE_REQUEST 消息来主动请求数据。

Windows 侧客户与 iRMX 侧服务器对话沿用了 Windows 内的 DDE 通信概念,但又有所不同。首先,在对话建立前,iRMX 侧服务器程序必须对 DDE 库初始化,并把自身登录为服务器,向 DDE 库传递必要的信息;其次,iRMX 与 Windows 并不处于同一消息系统内,因此,彼此之间的交谈有一个 Windows 与 iRMX 侧功能代码的相互转换过程,由 DDE Router 和 DDE 库来完成;Windows 侧客户机的编程与 Windows 内部进程 DDE 通信的客户机编程类似,iRMX 侧服务器应用程序则主要通过向在 DDE 库中登录的对话回调函数和数据回调函数来获取 DDE 通信的状态和数据。

下面就 DDE 热链接及在实时系统中的应用作一说明。DDE 热链接通信协议包括五个消息应答过程,即:

- ①对话建链(WM_DDE_INITIATE,WM_DDE_ACK)
- ②数据项热链接(WM_DDE_ADVISE,WM_DDE_ACK)
- ③数据发送(WM_DDE_DATA,WM_DDE_ACK)
- ④数据项拆除热链接(WM_DDE_UNADVISE,WM_DDE_ACK)
- ⑤对话拆链(WM_DDE_TERMINATE,WM_DDE_TERMINATE)

以上 DDE 热链接通信协议根据数据项的值变化与否来决定是否发送该数据,这样做虽然可以避免相同数据的发送以减缓信道的阻塞,但却一方面增加了建立和拆除布局项热链接的时间及其过程的复杂程度,使系统的可靠性降低;另一方面,服务器在发送某一数据时还要与上一次的数据值相比较,浪费了时间和空间。在实时控制系统中,考虑到实时数据始终处于不断变化之中,而对处于时刻变化中的数据,使用热链接不仅没有必要,反而会浪费宝贵的时间和空间,因此,在实时系统中可以对 DDE 热链接作如下简化:

①在 DDE 通信建立后,不做逐个数据项的热链接(ADVISE),在关闭通信时,也不做逐个数据项的拆链(UNADVISE),这样可以实现软件启停的快速性和可靠性,在大量测点情况下更是如此;

②服务器到客户的数据传输不再以“数据变化才传输”为准则,而是根据客户对服务器的请求做周期或随机性传递。客户用 WM_DDE_POKE 消息将请求的服务名和有关数据信息传送给服务器,它不同于冷链接的 WM_DDE_REQUEST。其实现框图如图 2、3。

这样,简化后的 DDE 协议忽略了 ADVISE、UNADVISE 环节,增加了客户侧向服务器侧的数据传递和服务请求(WM_DDE_POKE),更符合实时系统的需要。

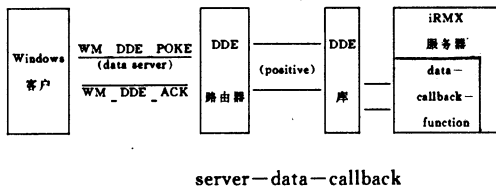


图 2 Windows 客户向 iRMX 服务器发送服务请求及数据

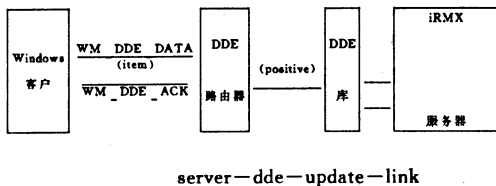


图 3 iRMX 服务器向 Windows 客户发送数据

4 结束语

利用 DDE 动态数据通信实现 iRMX 与 Windows 数据交换,将 iRMX 实时操作系统与 Windows 图形系统有机地结合起来,既满足了过程控制对实时性的要求,又兼顾了人机界面的友好性,是计算机实时控制系统的较为理想的方案。

参考文献

- 1 谷晓华,何志泉. Windows 3.1 程序设计技术与实例. 学苑出版社,1993.12
- 2 英特尔计算机技术有限公司. iRMX for Windows 实时操作系统(一)(二)(三). 海洋出版社,1992

静探曲线数据处理中的计算机采样技术

徐斌(同济大学地下建筑与工程系上海 200092)

唐建东(长江水利委员会第七工程勘测处)

摘 要 本文对静探曲线数据处理中所采用的一些计算机技术作了简要的讨论,并详细介绍了其中静探曲线采点程序,采用该程序采点,对南水北调某一实测静探曲线作了数据分析。

关键词 静探 采样 相关距离

1 前言

静探曲线的数据处理主要包括单孔土层和场地土层的均值、变异系数等统计参数的计算,相关距离的推求,概率模型(如正态分布、 β 分布)的拟合及 q_c 、 f_s 的相关分析等。

在用相关函数法或递推空间法求静探孔的相关距离时,需要将锥尖阻力 q_c 和侧壁摩阻力 f_s 的连续曲线处理为不同采样间距的离散化数据,当探头未配备数据处理微机或在处理老的静探曲线时,就必须从曲线上获得相应的点值。最简单也是最麻烦的方法是人工找点,这样做工作量太大,工效甚低,况且容易出错,对大规模的数据处理工作显然不行。我们使用的方法是用数字化仪首先将静探曲线读入微机,再在微机内通过编程找点。这样做虽然在数字化仪输入上增加了时间,但在曲线上找点与人工相比速度却大大加快了,精度也有了提高。

与图形转化为数据相比,计算过程中数据转化为图形的方法是比较丰富的,一般的高级语言都能胜任。除 FORTRAN 语言需要配备图形软件包外, BASIC、PASCAL、C 等高级语言都有专门处理图形的函数, AutoLISP 语言、C 的 ADS 环境以及 Windows 下的 C 编程且能方便地与 AutoCAD 建立通信,这方面的内容我们不再赘述。

2 数字化仪与 AUTOLISP 语言的使用

2.1 数字化仪

数字化仪通过鼠标的十字叉丝中心沿静探曲线的连续走动,在相应的 AutoCAD 绘图屏幕上用 Sketch 命令徒手勾勒出对应的曲线。Sketch 的取点间距应不大于原静探曲线的取点间距,也不能大于即将求得的

相关距离。

为区分双桥探头锥尖阻力曲线与侧壁摩阻力曲线,可在图中采用不同的颜色设置或线型设置,在笔者所编的 AutoLISP 程序中采用了颜色识别的方式。数字化仪的坐标标定最好选用以钻孔基准面为(0,0)点,以深度方向为 y 轴的坐标系,以利于程序取点以及与原静探曲线对照。

2.2 AutoLISP 语言

AutoLISP 语言是一种人工智能型的表语言,正如 C 程序的基本元素为函数一样, LISP 程序的基本元素为表,一个 LISP 程序由一系列表所组成。由于 AutoLISP 程序能直接调用 CAD 中的命令和函数,并以 CAD 为操作平台而运行,这就使得它能直接读出由数字化仪输入的静探曲线,且通过其 I/O 接口将所需的曲线点写入指定文件。

2.3 C 的 ADS 环境

ADS 最初的概念是在 AutoCAD 11.0 版本出现的,在 12.0 与 13.0 版本中得到了完善与发展。它的出现,使得熟练的 C 程序员可以不学习 AutoLISP 语言也能开发专业 CAD 软件,且能完全应用 C 的全部库函数,这是 AutoLISP 所望尘莫及的。在我们的程序编制中尚未用到 ADS 编程,但若将静探曲线数据处理的一系列过程开发为商用软件的话, ADS 的 Windows 编程不失为明智的选择。

3 编程原理及使用简介(源程序见附录)

相应于静探曲线统计指标的推求,我们编写了有关程序。这里我们主要讲解从静探曲线找点的 AutoLISP 程序,其他程序可参阅有关文献。

本程序为方便用户使用,设置了一些环境变量,分别为能包容曲线的窗口的最大 x、y 值及最小 x、y 值,加上 y 方向的取样间隔。如首次运行,并选择回车输

入,则此时缺省值为 $xmin=ymin=0, xmax=ymax=100, dy=0.1$ 。只要不超出本次 load 作用域,则首次运行后的缺省值为上次运行的设置。颜色输入值要求选择某一实体或一组实体,以该组实体中最后一个实体的颜色作为欲选点所在曲线的颜色。

程序从 $y=ymin$ 开始循环,每次循环增量即为取样间隔 ($y=y+dy$),判断选定曲线是否与由 ($xmin, y$) 及 ($xmax, y$) 为端点的线段相交;若相交(判断相交的函数为 inters),求出交点;若不相交,则转入一下循环(可参考程序框图)。

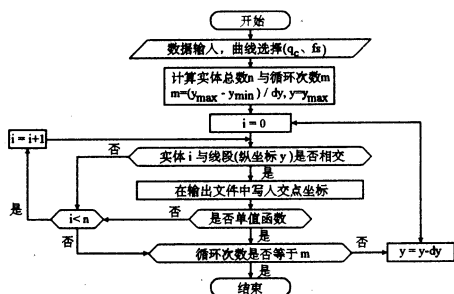


图1 静探曲线采样程序原理图

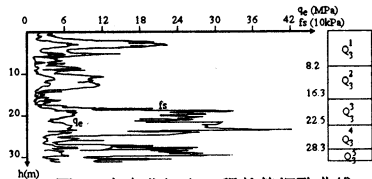


图2 南水北调山工程某静探孔曲线

这里要说明的是静探曲线是个单值函数,但由于数字化仪的人工扫描误差,难免会出现多值对应情形,此时应取其中某一合理值(一般来说,第一交点总是相对准确的)。程序中给出了单值函数与多值函数的选择,建议在读静探曲线时用缺省值 0,即选择单值函数。

程序运行中(0,0)点不予给出,可自行在结果文件中添加。结果文件中有可能给出 $x=0$ 时的 y 值,多是由于扫描误差而产生,建议用(0,0)替代。结果文件最后一个数据为点数,增加(0,0)点时还应加 1。

4 运行实例

南水北调山工程需穿越第三系黄土,覆盖层最深达近 40 米。在工程区范围内地质勘察中作了大量原位静力触探试验,由于记录仪器的局限性,测试成果只有静力触探曲线图,测试指标只能从曲线上查得,人为影响因素较大。为了弥补人为因素,充分反映地层的力学特性,我们根据静力触探原始曲线,运用上述程序进行了找点、指标统计等一系列计算。

下面选择某静探孔数据简要地介绍其过程。

为了更能说明问题,此处采点间距取为 0.025 米。由此我们得到静探数据的指标统计及概型拟合如表 1 示:

表1 静探数据指标统计及概型拟合

土层时代	Q_1^1		Q_2^2		Q_3^3		Q_4^4		Q_5^5	
指标分类	qc	fs	qc	fs	qc	fs	qc	fs	qc	fs
均值	2.701	8.314	2.238	6.654	4.545	18.226	4.438	16.881	4.465	19.041
标准差	1.867	5.889	0.988	3.463	2.223	10.205	1.735	10.301	1.090	6.992
变异系数	0.691	0.708	0.441	0.520	0.489	0.560	0.390	0.610	0.244	0.367
标准偏差系数	9.996	8.974	6.074	-0.709	1.337	3.212	3.010	2.041	1.758	-0.477
标准峰度系数	2.086	0.131	-2.569	-5.521	-4.377	-4.259	-3.804	-3.517	1.937	2.924
正态分布拟合	拒绝	拒绝	拒绝	拒绝	拒绝	接受	接受	接受	接受	接受
β 分布拟合	拒绝	拒绝	拒绝	拒绝	拒绝	接受	拒绝	拒绝	接受	接受

从以上结果看,数据比较不规则,变异系数较大,概型拟合也较差。这一方面是由于静力触探对土质的微弱改变较敏感,另一方面,该静探孔土层的划分也有值得商榷的地方。

除以上指标外,我们进一步探讨静探数据相关距离的计算机处理问题。它的理论及计算方法尚有待完善,具体可参阅[1]文。我们采用的相关函数形式为:

$$R(\tau) = Ae^{-B\tau} \cos \omega \tau$$

其中, A, B, ω 均为待求未知量。具体计算中,由于方程拟合的复杂性,笔者采用了三重迭代与曲线对比相结合的计算机人工交互形式。下面给出其相关距离计算结果(见表 2):

表2 静探数据相关距离计算

土层时代	Q_1^1		Q_2^2		Q_3^3		Q_4^4		Q_5^5	
指标分类	qc	fs	qc	fs	qc	fs	qc	fs	qc	fs
递推空回法	0.1900	2000	2000	1750	1940	1920	2030	1970	2980	300
相关函数法	0.5800	4180	2600	2570	6620	4420	5770	7380	5670	580

总体上看,两种方法求得的结果约相差 2 倍,其中的关系还有待研究。

5 结论

静探数据的处理是一个繁复的过程,应用计算机技术进行交互操作是今后的主攻方向。本文对此作了一些尝试,如何将所有的工作开发为软件包,以及对算法进一步完善,还需继续研究。

附源程序:

```
(defun c:/x y a i e d c l l c r c r l o b n f n f h p t1 p r2 p t3 p t4 p t m l t j k l t1 r p)
(setq fn (getstring "Enter output file name, "))
(setq fh (open fn "W"))
(if (= rp nil) (setq rp 0))
(princ "\nSingle or Multiple values function (0, Single, 1* Multiple) (*")
(princ rp)
(princ ")")
(setq n (getint))
(if (= n nil) (setq n rp))
(setq rp n)
(if (= xmin nil) (setq xmin 0))
(princ "\nEnter minimum x (*)")
(princ xmin)
(princ ")")
(setq x (getreal))
(if (= x nil) (setq x xmin))
(setq xmin x)
(if (= xmax nil) (setq xmax 100))
(princ "\nEnter maximum x (*)")
(princ xmax)
```

```

(princ " ");
(setq x (getreal))
(if (= x nil) (setq x xmax))
(setq xmax x)
(if (= ymin nil) (setq ymin 0))
(princ "\nEnter minium y(*)")
(princ ymin)
(princ " ");
(setq x (getreal))
(if (= x nil) (setq x ymin))
(setq ymin x)
(if (= ymax nil) (setq ymax 100))
(princ "\nEnter maxium y(*)")
(princ ymax)
(princ " ");
setq x (getreal))
(if (= x nil) (setq x yax))
(setq yax x)
(if (= dy nil) (setq dy 0.1))
(princ "\nEnter y increase value(*)")
(princ dy)
(princ " ");
(setq x (getreal))
(if (= x nil) (setq x dy))
(setq dy x)
(setq m (fix (/ (-ymax ymin) dy)))
(setq y ymax)
(setq i 0)
(setq j 0)
(princ "Please input line color by select a line:\n")
(setq a (ssget))
(setq n (sslength a))
(setq ed (entget (ssname a (1-n))))
(setq cl (assoc 62 ed))
(setq cr (cdr cl))
(setq ob (assoc "X"))
(setq n (sslength ob))
(princ "Write wanted points to file")
(princ fn)
(princ "... \n")
(setq k 0)
(repeat m
  (while ((in)
    (setq ed (entget (ssname ob i)))
    (setq cl (assoc 62 ed))
    (setq crl (cdr cl))
    (setq lt1 (assoc 0 ed))
    (setq lt (cdr lt1))
    (if (and (= cr crl) (= lt "LINE"))
      (progn
        (setq pt1 (cdr (assoc 10 ed)))
        (setq pt2 (cdr (assoc 11 ed)))
        (setq pt3 (list xmin y))
        (setq pt4 (list xmax y))
        (setq pr (inters pt1 pt2 pt3 pt4 T))
        (if (/= pr nil)
          (progn
            (princ (cadr pt) fh)
            (princ " * fh")
            (princ (car pt) fh)
            (princ " \n * fh")
            (setq j (+ j 1))
            (if (= rp 0) (setq i n))
          )
        )
      )
    )
  )
)
)

```

```

)
(setq i (- i 1))
)
(setq i 0)
(setq y (- y dy))
(setq k (+ k 1))
(if (= k (/ m 120))
  (progn
    (princ " * ")
    (setq k 0)
  )
)
)
(princ j fh)
(close fh)
)

```

参考文献

- 1 天津大学郭怀志等编. 提高水利工程可靠度分析方法的可信性. 1991
- 2 中国科学院计算中心概率组编. 概率统计计算. 科学出版社, 1979
- 3 方铁编著. AutoCADC 语言高级编程. 清华大学出版社, 1994
- 4 于乃江编著. 用C语言开发 AutoCAD12.0—ADS 程序设计指南. 国防工业出版社, 1995

征稿启事

一、征稿内容:

本刊目前特别欢迎作者撰写投寄以下内容的稿件:

1. 具有预见性、策略性、统计性、分析性、整合性、归纳性、技巧性、实用性、新颖性、爆炸性、趣味性、工具性、资料性的综述论稿;

2. 看了即懂、拿来可用、图文并茂、见解深刻、层次分明(含故障现象、故障分析排除等)、新颖实用的维护维修稿;

3. 目前计算机应用的新领域、新技术、难点、热点、空白点。

二、征稿要求:

稿件宜用激光照排正规打印, 标明字数、页码、图号、图位、文种、作者详细通讯地址、电话号码、邮码及英文标题、参考文献; 若无条件打印, 请正规誊抄于方格稿纸上, 图纸要正规描绘, 由于人手所限, 恕不退稿, 请作者自留底稿, 来稿要求新颖实用, 一稿一投。对优秀来稿, 经审核后, 本社将择优刊用, 稿酬从丰。

热忱欢迎广大作者踊跃赐稿!

《计算机应用研究》杂志社

校园网计费管理的分析与研究

陈品德 龚正虎 汪钟鸣

(湘潭矿业学院自动化系 411201)

摘 要 本文讨论了网络计费的环境和计费原则,对基于 IP 的计费策略进行了分析,同时还对计费信息的收集和处、用户接口的实现提出了可行的解决方案。

关键词 计费 校园网 TCP/IP SNMP MIB

0 引言

自从 1994 年 CERNET(中国教育科研网络)工程启动以来,已有 200 多所高校加入了 CERNET,并与 Internet 连通,校园内部的网络也已纷纷建成或正在设计和规划之中。作为现代的网络系统,网络管理系统已成为网络建设中必不可少的重要组成部分之一,而其中的计费管理问题又显得尤其紧迫和重要。目前,各高校在建设校园网的时候,一般在购买网络设备的同时也购置了相应的网络管理产品,如 SunNet Manager, IBM 公司的 NetView 等,这些产品支持 SNMP,它们既提供了一定的网管应用软件,也提供了一个网管开发平台,使得用户可以基于该平台上开发出自己的网管应用软件。但是,它们都没有提供现成的网络计费功能。因此,关于网络计费功能的实现需要用户自己去完成。

1 网络环境和计费原则

校园网内部一般由几个互联的子网组成,通过边界路由器与外界连接,家庭用户以拨号方式通过访问路由器上网,如图 1 所示:

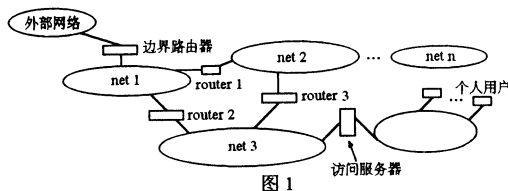


图 1

从计费的角度来看,校园网中有以下几点值得注意的地方:

1)、边界路由器,它们可能不只一个,可归结为几条出口信道,分别为 DDN, X. 25 或其他。它们的速率从 9600KBPS 到 64K 不等,相对于内部网络而言,速率很低,是当前校园网的瓶颈所在。对于这些信道,既要按月支付电信部门的通信费用,又要交纳 CERNET 收取的数据费用,是网络运转之后主要的维持费用之一,因此对出口信道的计费是当前我们主要所关心的,由于网络分组交换的特点,对出口信道的占用我们用

网络通讯的数据量来衡量。

2)、内部网络,我国高校大部分申请到 8~16 个 C 类地址,由于 TCP/IP 协议的特点,不管是何种结构的网络,子网的划分总得经过 IP 网关进行。因此,内部网络总可抽象为由 IP 网关互联起来的网间网,传输速度从 10M~155M 不等,就当前的应用状况看来是非常充足的,并且校园网络一般由学校一次(或几批)投资建成,作为学校的公共基础设施,可以免费使用。

3)、远程访问服务器,家庭用户以 PPP/SLIP 方式拨号入网,访问服务器的端口数目是有限的,当一个用户占据某端口时,其他用户就不能使用该端口,因此,端口的使用时间也是收费的依据之一,这些用户在计费管理当中应当作为一个特殊的用户群来看待。

4)、校内共用的主机资源,如各种服务器:电子邮件服务器、文件服务器、WWW 服务器等,自然这些资源也是有限的,应当考虑收费管理。

由此可知,网络计费可以归结为:

①对主机资源计费

②对信道资源计费,主要考虑对出口信道的占用和对远程访问服务器的端口的使用

2 网络的计费策略

2.1 主机资源

包括校园内部提供公共服务的各种主机资源,如文件服务器、邮件服务器、打印服务器、计算服务器,以及为用户提供的各种信息查询、信息服务等。这些资源类别种类繁多,无法一一列举,由该主机管理员自行解决收费问题,不是本文的讨论重点。

2.2 基于 IP 报文的信道资源

对于基于 TCP/IP 协议的分组交换网络,在网络中传递的是 IP 报文,因此,以 IP 报文的数目和长度来衡量信道的占用情况是比较合适的。从 IP 报文的头部中,可以提取原 IP 地址和目标 IP 地址,由此确定该报文的发起方和接收方。

但是,网络收费最终要对用户进行,虽然从 IP 报文的头部信息中能够得到源、目的地的 IP 地址,但 IP 地址并不一定代表着用户信息,那么如何确定用户呢?根据校园网的几种不同用户类型,我们分别作如下讨论:

1)、有固定 IP 地址的单用户主机系统

可以利用管理员在分配 IP 地址时所建立起来的表项:IP 地址、用户名,来确定用户。

2) 多用户主机

其 IP 地址是固定的,但由于有多个用户同时使用该主机,仅靠 IP 地址甚至 TCP/UDP 的端口号称仍然无法确定用户,在这种情况下,依靠计费系统直接向用户收费难以实现。但是,我们可以认为该主机存在一个管理代理人(Agent),即从网络管理员处得到该 IP 地址的注册用户,由他负责该 IP 地址的费用,至于他如何把这些费用分摊到各个实际用户头上,计费系统不干涉。其实,这也符合分级管理的原则。

3)、采用动态主机配置协议(DHCP)获得 IP 地址的主机

这时,主机本身并不固定地占有一个 IP 地址,而是在该主机在开机启动后,利用动态主机配置协议从 DHCP 服务器得到一个临时分配给它使用的 IP 地址,当它退出网络后,该 IP 地址又可以分配给别的用户使用,因此在这种情况下,也不能利用 IP 地址来确定用户。

因此,我们对 DHCP 用户收费时作如下考虑:

a、严格限制动态主机配置(DHCP)的使用范围,一般用于中心机房或某些特定的单位,网络计费时仍然针对 IP 地址进行,但属于动态主机配置 IP 地址范围内的费用向相应的单位收取,由该单位负责向其用户收取。

b、限制 DHCP 客户的使用权限,只许它们访问校园内部网络,或只能使用某种应用(如电子邮件,则我们可以按电子邮件进行收费)等。

由以上的分析可知,采取分级管理的原则,基于 IP 的计费管理是可行的。但是,需要注意的是,对于某些计算机,尤其是随着 PC 机用户的增多,用户可以自己安装 TCP/IP 软件,自己指定 IP 地址,因此有可能使得非授权用户进入网络,在这种情况下,仅靠 IP 地址,仍无法确定真实用户。其解决途径有二:

①通过在相应的路由器上设置基于包过滤的防火墙,只有经过授权的 IP 地址才能通过,但这种方法能限制非法地址的 IP 报文通过。

②结合 MAC 地址+IP 地址的方法来确定主机,MAC 地址在一个物理网络上唯一的,通常是和硬件设备相对应(如以太网地址在出厂时就已经做好在网卡上了,用户不能改变),因此 MAC 地址+IP 地址在一个物理子网上可以唯一确定一个节点位置。但是遗憾的是:MAC 地址在经过路由器(IP 网关)之后,数据的帧格式进行了转换,其原来的 MAC 不复存在了,因此,在跨子网的情况下,与从数据帧中得到原来的 MAC 地址已不可能。但是,我们注意到,任何一台计算机欲与外界通信,其所在物理子网的路由器中,在它

的 ARP 地址表中,一定保留着主机的 IP 地址和它的 MAC 地址的表项(ARP 协议使然)。因此,通过不定期地分析 ARP 表中的内容,我们可以确定是否有非法用户闯入网络。

2.3 采用 PPP/SLIP 拨号上网的用户

这是一个特殊的用户群,作为计费管理的一部分,我们要特别加以分析。

拨号用户是通过访问服务器上网的,自己没有固定的 IP 地址,他拨通哪个端口就占用哪个 IP 地址。同时,上网后,他除了和其他用户一样要占用信道资源外,还占据了一个拨号端口,这部分资源也是比较紧张的,因为当一个用户拨通一个端口后,其他用户就不能使用这个端口。因此,拨号上网的用户意见除了按数据量收取信道费用外,还要按时间长度收取端口占用费。这些计费信息从何处得到呢?可以从访问服务器处获得(如 3COM 的远程访问服务器 AccessBuilder,或 Cisco 公司的设备等。)(参见⑧125 页,142 页)

利用上述方法(跟具体设备有关),我们可以获得拨号用户占用拨号线路的时间和通信量的有关信息,但是除此以外,如何知道用户访问的是国内、国际信道呢?为了管理的方便,我们可以把访问服务的端口分成三组:校内组、国内组和国外组。对于拨号入校内组访问服务器端口上网的用户,只能访问校内资源,其它组情况类似。对于采用不同组端口上网的用户,其数据量分别采用不同的收费标准。

3 信息的收集和处理

主机资源及访问服务器端口使用情况的数据收集和处理,我们已经在本文中提及,这里不在赘述。

基于 IP 的数据流量信息的收集可以考虑以下三种方案:

方案一:由于我们关心的是出口信道的使用情况,边界路由器自然是最理想的数据收集点。我们可以利用 SNMP 协议直接从路由器获得所需数据。

例如,Cisco 公司的路由器支持 IP ACCOUNTING 的功能,它提供了与计费有关的 MIB:1.3.6.1.4.1.9.2.4.7.1.3 和 1.3.6.1.4.1.9.2.4.9.1.3,这两个 MIB 变量保留了对 IP 地址数据流量的一张表,周期性地利用 Get 操作可以从边界路由器处获得计费的有关数据。

这种方案的优点是只要边界路由器支持 IP ACCOUNTING 功能,则实现起来比较简单,缺点在于占用了存取资源和处理时间,影响了路由器的效率,并且由于计费信息的数据量非常大,这些数据如不及时取走,很快会发生溢出,造成计费数据的丢失。又由于网络通讯的不均匀性,难以确定合适的取数时间间隔,过于频繁,又使得对网络信道资源的开销增大。

方案二:假设存在这样的情形:边界路由器的一个端口(或几个端口)连至出口信道,另一端口连到一个

广播方式的局域网上(设为 A)。那么所有从边界路由器进出的报文一定也经过 A, 由于 A 是广播方式的网络, 我们可以在 A 上连接一个特殊的站点, 称为计费站点。它能监测到流经 A 上的所有报文, 必定包括从边界路由器上进出的报文, 因而可以把有关的计费信息提取出来。对于其它类型的网络, 可以按照图 3 所示进行改造, 所需费用也不大。

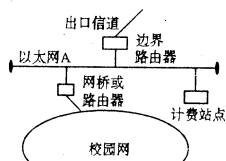


图 2

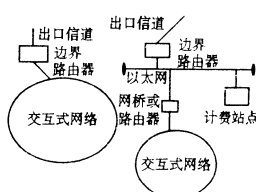


图 3

采用这种方案, 除了需要一台专用的计费站点外, 对原来的网络环境无任何影响, 也无须具有计费功能的路由器支持, 并且其功能的扩充相当灵活, 因而具有一定的实用价值, 缺点在于实现起来略显困难。

方案三: 考虑到 Internet 的几种主要应用如电子邮件、文件传输、远程登录都是基于 TCP 协议实现, TCP 提供的服务是一种有连接的服务, 在信息传输之前都有一个连接过程。因此, 可以在计费网关处设置一个监视进程, 监视通信的整个过程, 每当检测到连接请求报文通过时, 就暂时延缓这次连接, 在这个期间, 与发起该连接的主机进行联系, 一起确定该用户的合法权, 如果是合法用户, 则允许这次通讯过程继续进行下去, 并继续监视该次通讯的过程, 记录下这次连接的时间和通讯量。

显然, 这种方法实现起来更为困难, 需要对边界路由器的内部结构做较大改动, 而且要求所有上网主机都要安装一个认证程序, 影响了网络的吞吐能力, 增大了用户的响应时间。这种方案的优点在于: 它把网络的

安全管理功能结合进来, 而且能够直接面向用户进行收费管理(而不是 IP 地址); 其次, 在认证的过程中, 可以协商服务质量。

4 网络计费系统的实现

基于方案一的实现请参见⑥, 因为时间和条件的局限, 笔者对上文中提到的信息收集方案二, 在以太网和微机上的 UNIX SYSTEM V 平台上实现了数据的收集, 并基于 WEB 和 CGI 技术提供了用户和管理员的查询接口, 具体的实现限于篇幅另外撰文叙述。

5 结束语

网络收费是一个非常重要但又未能找到妥善解决方案的问题, 其合理性、可靠性及安全性是用户非常关心的问题, 有关这方面的问题的尚待进一步地思考和研究, 同时如何制定费率政策, 充分发挥信道的利用率本文也未提及。网络计费不仅仅是一个技术上的问题, 也涉及到人员、组织、政策及经济等各个方面的问题。说明: 本文是硕士学位毕业论文的一部分。

参考文献

1 RFC 文档

- ① RFC1155 truture and Identification of Management Information for TCP/IP-base Internet. May. 1990
- ② RFC1157 A Simple Network Management Protocol(SNMP). May. 1990.
- ③ RFC1213 Management Information Base for Network Management of TCP/IP-base Internets. MIB-II, Mar. 1991
- ④ RFC1303 A Convention for Describing SNMP-based Agents Protocol(SNMPv2)
- ⑤ RFC1351 SNMP Administrative Model.
- ⑥ RFC1272 Internet Accounting Background November 1991

2 周明天等. TCP/IP 网络原理与技术, 清华大学出版社, 1993.

3 石冰心主编. 中国教育和科研计算机网的研究与发展. 第一卷. 华中理工大学出版社, 1996 年 9 月

欢迎订阅《电路、信号与系统分析 For Windows》

由电子科技大学出版社出版的此书主要介绍在 Windows 支持下的电路、信号与系统的模拟分析与设计方法。第一章用一个简例, 介绍操作入门; 第二章介绍 Schematics, 即原理图的绘制、编辑及其分析、设置等内容; 第三章介绍模拟计算程序 Pspice 的使用方法; 第四章介绍图形后处理器 Probe; 第五章介绍信号源模型的编辑方法 Stmed; 第六章是元件模型的编辑 Parts; 第七章介绍电路、信号与系统分析及设计方法的应用实例。

本书所介绍的内容, 具有图文并茂, 通俗易懂, 可读性强等特点。作者在第七章中, 编入了有一定深度的应用实例, 书后还附有大量的元件清单, 这对于工程设计人员的设计是很方便的。

本书可作为大专院校的计算机辅助教学(CAI)教材或工程设计人员的参考书。

欲邮购此书的读者, 请将书款汇至我刊编辑部张钢编辑处。每本邮购价 18 元整。款到挂号寄书。

分布式集中供热微机监测系统

冯永祥 崔新民 贾颖力

(内蒙古工业大学电力学院呼和浩特 010080)

摘 要 本文介绍了呼和浩特市热力公司供热微机监测系统的基本功能及硬件配置,本系统采用星形结构的分布式微机监测系统,对各供热站的供热参数进行实时监测,可自动完成数据的记录、统计、分析、处理、贮存及打印工作,并能对各种故障进行实时分析、诊断及报警,系统运行稳定可靠。

关键词 分布式 集中供热 微机 监测系统

1 系统的基本构成

供热微机监测系统(以下简称“监测系统”)的监测对象是城市的集中供热系统。由于监测点十分分散,分布情况复杂,所以信号的传输是一大难点。为了简化系统的信号传输网络结构,充分利用现有系统的信号电缆和设备,降低成本,提高可靠性,本监测系统采用分布式计算机系统。所谓分布式计算机系统就是分布在不同地点而以协作方式互相配合进行工作的多计算机系统,一般在几个地方设置执行简单任务的低档计算机,而较复杂的任务则集中由高档计算机执行。执行局部独立功能的低档计算机称为下位机,而对各下级计算机起协调作用担任高级控制与管理职能的称为上位机。在这个系统中不仅各个独立的计算机在地理上是分散的,而且从硬件和软件资源来说也是分散的,但是它们是一个统一的整体,有明显的上下之分的层次性。这样的分布式计算机系统便于开发、维护和扩充,系统的可靠性高。各热力站由下位机负责参数的检测及管理,通过通讯网络将各下位机与上位机相连。这样恰当地解决了“统一”与“分散”的矛盾,既照顾了供热网布局分散,面大的特点,又满足了供热网需要整体分析的要求;每台下位机既可以单独工作,又可以统一指挥,个别下位机发生故障后,整个监测网可以照常运行,这样系统的可靠性大为提高,同时实现了系统的“资源共享”。我们经过对呼和浩特市供热网的分析研究,并在保持现有信号和设备的前提下,提出了分布式集中供热微机监测系统的设计方案,并已稳定运行三年,受到用户的好评。本系统采用星形结构的分布式计算机监测系统。系统的结构如图1所示。

在这个系统中,上位机采用易维护、扩充性、兼容性好的 AST486 作主机,它具有丰富的软硬件资源;上位机带有一台 STD 工业控制机作为主机的值班机,它的主要作用是由它实施与各子站(前端机)STD 工业控制机之间的通讯和联络,特别是由它完成日常的值班巡回读取数据,保持经常的通讯联络。而上位机不必每天 24 小时连续工作。值班机和上位机之间采用常用

的 RS-232C 串行通讯接口。这种结构的好处是避开了上位机连续长期工作的弱点,又能充分发挥其作用。在本系统中,系统的值班机可以保存 72 小时的数据,上位机可以完成其它工作,只要 24 小时时间内采集一次数据,即可保持数据的长期连续性。分布式计算机系统的值班机和下位机均采用 STD 工业控制机,

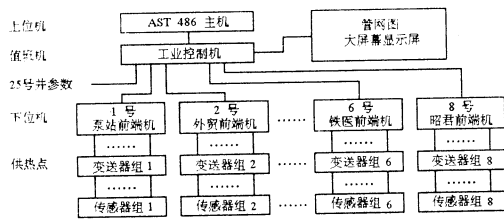


图 1 系统结构图

其特点是:系统便于维护,易于更新、扩充和升级,提高了系统的可靠性。通讯时,值班机处于主动地位,随时可根据程序或操作员的要求进入通讯状态,下位机处于被动地位,只有在值班机允许时,才能与值班机进行通讯。

从系统的结构图中我们可以看出,系统由以下几部分组成:

- ①传感器和变送器负责现场参数的转换;
- ②下位机负责各热力站供热参数的采集、存储、传送及实时故障诊断、分析;
- ③值班机负责从下位机读取数据及故障报警信息,并向上位机发回数据。同时控制模拟屏的状态及参数显示;
- ④管网图大屏幕显示屏,可通过软件定时转换显示不同的系统参数及状态,如温度、压力及流量参数。在模拟屏上设有供热管网图、供热站位置图及参数名称等,通过指示灯的亮灭来表明所选参数的性质;
- ⑤上位机主要负责系统数据的记录、统计、供热曲线的显示及建档、打印输出等工作。

下面简要介绍各部分的组成及其功能。

2 下位机

下位机主要负责各供热站的供热参数的检测及运

行状态的监测,并定时向值班机发回数据。

A 现场下位机的主要任务:

(1)下位机每隔数秒钟(可以人为设定采样时间)实时采集每一个供热点的供热参数,包括供水、回水的压力、温度及流量信号,并进行预处理,然后转换成相应的数字信号。

(2)下位机将所采集的数据,定时向值班机发送。为了保证传送可靠,并使上位机能监视下位机的工作状态,数据通讯采用应答方式。通讯为电流环串行通讯方式,通讯线路上为 20MA 的电流信号,通过控制电流的通断来传递 0,1 信号,达到传递信息的目的。在设计通信线路时,上位机及各台下位机联接都采用光电隔离器,这样即使通讯线路上偶然有高压干扰,也不会造成系统的瘫痪。为保证通讯的可靠性,在软件上采用了校验和的方法进行校验。即每发送一组数据,同时发送此组数据的和的尾数,收方收到数据后同样计算数据和的尾数,若尾数不等,则数据无效,请求重新发送。各下位机定时采集数据,并进行数据的初步处理与贮存。

(3)下位机可以用面板上的数码管实时显示当地的被测数据,供当地值班人员监视。

(4)为了提高传送数据的可靠性,保证数据的安全,下位机将不保存任何历史数据,而是将数据及时传送到值班机保存。

(5)为了提高系统的可靠性,下位机备有自检功能,将本身的各故障报告至值班机,以便及时处理。

(6)下位机在设计时已留有扩充余量,需要时可将该供热微机监测系统升级为供热微机监控系统。

B 下位机的组成结构:

(1)CPU 板:板上装有微处理器,128K(64K ROM+64K RAM)内存以及一个 IBM PC/XT 键盘接口。

(2)显示驱动板:具有字符和图形两种控制方式,在图形方式下,能把显示缓冲区中的每一字节对应于八个像素显示在屏幕上,高位在左低位在右,BIT=0 对应像素为暗,BIT=1 对应像素为亮。这样可以很方便地显示图表、曲线以及汉字等。为软件的设计带来了很大的方便。

(3)二级汉字库板:下位机的信息均以汉字方式显示,为方便起见系统配备硬汉字库。

(4)系统支持板:系统支持板是工业控制机中的通用模板,它具有如下的功能:

①由 MSM5832 芯片组成一个带日历的实时时钟电路,提供时间数据,此芯片配有电池,具有掉电保护功能,主机关闭电源后,时钟仍能正常运行;

②由 74LS123 组成的 Watchdog Timer(计时监视器,俗称看门狗电路),使系统具有受干扰后自行恢复运行的功能;

③由 8253 计数器芯片组成报警讯响电路,它用于驱动扬声器发出各种不同声调的报警信号;

④此板可以提供用于驱动“主机运行”和“主机停机”指示灯的信号;该板的看门狗电路的工作原理是:在主控程序正常运行时,每 0.25 秒向电路发出一个脉冲信号。这个脉冲信号将迫使电路中的可重触发 74LS123 的输出保持在高电平,这个高电平使“主机运行”绿色发光二极管发光,指示系统运行正常;一旦主机被干扰,程序发生“错乱”时,正常的控制功能便不能执行,同时每 0.25 秒发出的脉冲也随之消失,等到了 74LS123 预定的翻转时间后(约 2 秒钟),其输出变为低电平,在这之后该板将同时产生两个动作:

(一)熄灭绿色发光二极管,并使红色发光二极管发光,表明主机已经停机;

(二)向系统发出复位信号,使系统重新启动;

程序重新启动一旦成功,上述现象便可消失,红 LED 熄灭,绿 LED 发光,显示器屏幕熄灭后又重新显示,表明程序已经恢复正常运行。在该板上还有一个允许/禁止控制开关。这个开关只有打在“允许”位置上时,板上的复位功能才能发挥作用。若打在“禁止”位置上,虽然仍然能够监视主机是否正常运行控制程序,并将其状态显示在红绿发光二极管上,但当发生程序“错乱”时,无法使系统复位,这一功能是为了调试软件而设置的。开关打到下面是“禁止”位置,上面是“允许”位置。

(5)开关量输入板:负责输入面板上各按钮的状态。

(6)电流环串行通讯接口板:负责与值班机进行数据交换。

(7)脉冲量输入板:负责流量信号的输入。

(8)模拟量输入板:可将 16 路 4—20 毫安的电流信号转换为数字量输入计算机。负责各模拟量信号的输入。

下位机的结构示意图如图 2 所示。

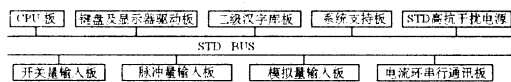


图 2 下位机结构示意图

C 下位机监控软件

下位机监控软件由以下几个主要模块构成:

(1) 初始化模块:完成系统参数的初始化。

(2) 命令菜单模块:显示并处理各菜单命令。

(3) 汉字显示模块:本系统均以汉字方式显示信息,此模块负责汉字点阵的读取及屏幕显示。

(4) 键盘输入分析模块:负责输入键盘信息,并进行各功能键的判断。

(5) 各种信息显示模块:负责显示参数及运行状

态等信息。

(6) 中断服务模块:每 0.25 秒系统中断一次,负责采集数据,故障检测,键盘状态和操作面板状态的读入以及串行通讯。

(7) 数据采集模块:负责各温度信号、压力信号、流量信号的采集,同时进行数字滤波。(8) 开关量输入模块:输入操作面板上各按钮的开关状态,以备主程序作出反应。

(9) 故障处理模块:负责检测各种故障情况,并将检查结果向值班机报告,同时下位机予以故障显示及声光报警。

(10) 串行通讯模块:负责与值班机联络,传送下位机参数及工作状态。

3 值班机

值班机用来辅助上位机工作,可连续 24 小时运行,负责各下位机数据的收集及向上位机的传递和模拟屏的控制。

A 值班机的主要任务:

(1) 25 号井供热参数的监测,25 号井为主网供热参数监测点。

(2) 供热管网图模拟屏的显示控制。

(3) 值班机定时收集各下位机发送来的各供热站的供热数据,进行真伪鉴别后,并存放在带掉电保护的存储器中。值班机将保存过去 72 小时的数据,并监视整个系统的工作状态。

(4) 为上位机提供过去 72 小时各站点的供热参数及工作状态。

(5) 可在显示器上显示各供热点的瞬时检测参数,任一供热点 72 小时供热参数曲线、网络工作图、各下位机的工作状态等。

(6) 值值班机可由上位机校对时钟,为了系统时间的统一,当时间出现误差时,可进行校对。

(7) 当值班机或某一下位机出现故障时,可及时发出报警信号,提示维修人员及时维修。

B 值班机的组成结构:

(1) 到(6)同下位机。

(7) 256K I/O 寻址存储板:此板是通过 I/O 口地址操作寻访地址的 256K 存储器板。并对 RAM 实施掉电保护。当对板上的存储器读写数据时,每读写一次,计数器自动将地址加 1,从而可以实现高速连续读写。用来存储检测参数及系统参数。

(8) 标准 RS-232C 串行通讯板:由 8251 芯片构成的标准 RS-232C 串行通讯板。本系统的软件是在 PC 机上开发的,这样可以充分地利用 PC 机的资源优势,方便地开发应用软件。我们利用串行通讯板将 PC 机中的程序传送到工业控制机中进行调试。使软件调试更加方便、快速、准确。当软件调试完成后,本系统利用此串行通讯板完成值班机与上位机的通讯。

(9) 模拟屏总线转换发送板:负责模拟屏显示驱动电路的控制。

值班机的结构示意图如图 3 所示:



图 3 值班机结构示意图

C 值班机监控软件:

值班机监控软件由以下各模块构成:

- (1) 初始化模块;
- (2) 命令菜单模块;
- (3) 汉字显示模块;
- (4) 键盘输入分析模块;
- (5) 各种信息显示模块;

以上各模块的功能与下位机中的各功能模块类似,在此不再赘述。

(6) 中断服务模块:负责值班机的故障检测,键盘状态和操作面板状态的读入以及与下位机和上位机的串行通讯;

(7) 开关信号输入模块:负责输入操作面板上各按钮的开关状态;

(8) 串行通讯模块:负责与上位机、下位机联络,以及各类参数的传送;

(9) 模拟屏信息显示模块。

4 上位机

上位机主要负责管理整个计算机网络,记录、统计、分析、处理、贮存、管理和打印测量数据以及对各种故障的分析、诊断及信息库的管理等工作。

上位机具有以下功能:

(1) 以流程图的方式显示供热总网运行图及各热力站供热情况图,以备实时状态监视。

(2) 定站巡点功能,根据输入的站号,在屏幕上显示出该站号相应的供热参数及运行状态。

(3) 巡站巡点功能,该命令的功能是:定时轮换显示各热力站相应的供热参数及状态。

(4) 定区巡站功能,我们把呼市供热网分为 8 个区,每一个区由一台下位机进行监测,如果要查寻某一区各供热站的供热参数,可以输入相应的区号,如果区号合法,屏幕上将显示相应区各热力站的供热参数及状态。

(5) 打印报表功能,因为经常需要统计各热力站的供热参数,为此我们设置了打印报表命令。包括参数打印及曲线打印。

(6) 读取数据功能,因为上位机所需要处理的历史参数存放在值班机中,为此每 24 小时需要利用串行通讯从值班机中读取,以备存档。

(7) 显示各热力站运行曲线,可将任意时间段的

各热力站运行参数曲线利用动态显示技术显示在屏幕上。便于操作人员随时掌握供热效果和管网的工作状态,作出相应的决策调整。

5 小结

集中供热微机监测系统的应用,必将为热力网的科学管理和安全运行提供可靠的保证,为提高供热效率,节约能源,减少环境污染提供科学的依据,必将带来良好的经济效益和社会效益。

Sybase 数据库与 web 的互连

邵立兵

(上海交通大学计算机科学工程系 200030)

摘 要 该文主要讲述 Sybase 数据库与 Web 的互连技术,着重讨论了 Web. works 中的 Internet&Intranet 中间件产品—web.sql,并用几个例子加以说明。

关键词 Web.sql HTS CGI NSAPI

1 引言

Internet 技术的不断发展,为 Internet 网络带来了大批的用户,使建立信息化社会成为可能。同时 Internet 技术的发展引起了计算机技术的更新和革命,各大计算机软硬件厂商纷纷进入这一市场,力图满足用户日益增长的 Internet 计算的需求。

Sybase 公司的 Internet 解决方案 Web. Works 是建立在 Sybase 开放的客户/服务器软件体系结构的基础上,是 Sybase 客户/服务器体系结构的扩展。通过 Sybase Web. works,可以实现在 Internet&Intranet 应用环境中,运用公司数据库、中间件和应用开发工具实现复杂的业务应用处理,同时 Sybase Web. works 提供了一个统一的、集成的和可扩展的应用开发环境,支持建立集成的客户/服务器应用和 Internet &Intranet 应用。

Sybase Web. works 主要由 Sybase 优秀的数据库产品—Sybase SQL Server for Internet、Internet &Intranet 中间件产品—web.sql 和一整套网络应用开发环境—Powersoft Internet &Intranet 应用开发系列工具所组成,在 Sybase Web. works 体系结构中实现 Web 服务器和数据库的透明集成、建立动态的数据驱动的 Home Page 及 Internet &Intranet 应用集成的开发环境。

Sybase Web.sql 是 Sybase Web. works Internet 解决方案中一个十分重要的软件,通过 Web.sql,使 Web 服务器和数据库服务器透明的集成在一起,WWW 浏览器可以灵活地浏览和处理存储在数据库中的数据,实现使用任何浏览器漫游任何数据源。

2 Web.sql 体系介绍

Web.sql 是实现 Sybase 客户/服务器体系结构的扩展,是一套 Internet &Intranet 中间件软件,它高效地建立起了 HTTP 服务器和数据库服务器透明连接

的桥梁。Web.sql 支持用户将 SQL 语句和 Perl 脚本语言直接嵌入到 HTML 中,实现对数据库的查询,更新,并且可以随心所欲地生成动态的、个性化的主页。

Web.sql 支持两种连接方式,用来实现 Web 服务器和 Sybase 数据库系统的连接:

1)通过标准的 CGI(Common Gateway Interface)的方式。这种方式适用面广,只要 HTTP 服务器支持 CGI1.1 协议,就可以实现与 Sybase 数据库系统的连接。如: Netscape Communication Server/Commerce Server, Open Market, Microsoft Internet Information Server, NCSA……。

2)通过 Web 服务器提供的专用应用编程接口。如: Netscape Communication Server/Commerce Server 提供的 NSAPI, Sybase Web.sql 针对 NSAPI 编写了专门的 Web.sql,以实现更优秀的数据库连接和访问性能。

Web.sql 中包含的相关术语有:

①HTS(Hyper Text Sybase)-是 Sybase 对 HTML 语言的扩充,在标准的 HTML 语法中增加了 Sybase 语句的标记,使用户可以直接在 HTML 制作的 Home Page 中添加访问 Sybase 数据库系统的语句,形成的文件以 HTS 的文件后缀命名。Sybase 语句标记可分为两种,一种是直接的 Sybase Transaction SQL 语句,另一种方式是嵌入 SybPerl Scripts 脚本语言,Perl Scripts 是 Web Home page 的一种编程语言,其输出的结果是 HTML 格式的数据流, Sybase 扩充了标准的 Perl Scripts,使用户可以在 Perl Scripts 中直接调用 Sybase 的数据库访问接口 Open Client 进行灵活的数据库处理。

②NSAPI(Netscape Application Programming Interface)-Netscape HTTP Server 的开放应用编程接口,与 CGI 相比较效率更高、速度更快。

③Open Client-Sybase 数据库服务器的应用访问接口和编程接口,是 Web.sql 的一个组成部件,提供

Web.sql 访问 Sybase 数据库的一个运行环境。

3 Web.sql 处理过程

在 CGI 连接方式下, Web.sql 实际上是一个遵循 CGI 标准的一个外部程序。当用户通过浏览器与 HTTP 提出请求, 通过 URL 或超文本的联接指定 Web.sql 处理相应的 HTS 的文件并传递相应的传输和环境设置; HTTP 服务器设定相应环境参数并调用 Web.sql 处理指定的 HTS 文件; Web.sql 处理 HTS 文件过程中就 HTML 描述的部分直接返回给 HTTP 服务器, Sybase 语句标记的部分则通过直接调用 Sybase Open Client 访问数据库或通过 Perl Scripts 运行环境和 Sybase Open Client 访问数据库, Sybase SQL Server 进行数据处理后将结果返回给 Web.sql, Web.sql 将数据库处理的结果格式化为 HTML 的格式返回给 HTTP 服务器, 或者通过 HTS 中 Perl Scripts 处理结果以 HTML 的形式返回给 HTTP 服务器; HTTP 服务器在接收到 Web.sql 处理的结果后直接返回给客户端 Web 浏览器; Web 浏览器便可以浏览到存放在 Sybase SQL Server 中的数据。

在 NSAPI 连接方式下, 其处理过程与 CGI 相似, 但 Web.sql 不再是一个被调用的外部可执行程序而是一个运行中的进程, 它可以同时维护多个 HTTP 服务器和数据库服务器之间的连接, 并具备数据高速缓存能力, 以实现高速的数据库访问。

4 举例

我首先在 Sybase 中建立一个 news_material 数据库, 再加入三张表。第一个表是总目录表, 它包括 total_catalogue_id 和 total_content 字段。第二个表是分目录表, 它包括 sub_catalogue_id, total_catalogue_id 和 sub_content 字段。第三个表是像带记录表, 它包括 recorder_id, time_code, sub_catalogue_id, tape_number 和 news_title 字段。

我要开发一个简单的数据库浏览程序。首先构造一个包含三个 Frame 的主页。命名为 Top、Left 和 Right。当用户进入浏览主页时, 在 Top 处显示所有的总目录。用户点击任一总目录, 则在 Left 处显示此总目录下的所有分目录。然后用户可点击任一分目录, 则在 Right 处显示此分目录下的所有记录。针对以上应用的 HTS 文件如下所示:

1) 浏览主页 - navigate.hts

```
(html)
<title>新闻资料浏览</title>
<frameset rows="10%,90%"
  (frame src="nav_t.hts" name="banner" marginwidth="1" marginheight="1")
  (frameset cols="20%,80%"
    (frame src="nav_frcont.hts" name="s_contents"
      marginwidth="1" marginheight="1")
    (frame src="nav_frmain.hts" name="s_main" marginwidth="1"
      marginheight="1")
  )
)
```

```
(/frameset)
</frameset>
</html>

2) 总目录浏览页面 - nav_t.hts
(html)
<title>新闻资料浏览----总目录</title>
<syb type=perl>
my $handle=ws_connect("pass");
my $sql_stmt=qq! select total_catalogue_id,total_content from
  news_material..total_catalogue;
my $rc=ct_sql($handle,$sql_stmt);
if($rc!=CS_SUCCEEDED){
  ws_error("ct_sql() call failed.");
  while(($rc=ct_results($handle,$result_type))--CS_SUCCEED)
  {
    RES_TYPE,{
      IF($result_type==CS_ROW_RESULT){
        my@row;
        while(@row=ct_fetch($handle))
        {
          $link=qq! nav_s.hts? s_catalogue_id=$row[0];
          printf"<a target='s_contents' href=%s>%s</a>",$link,
            $row[1];
        }
        last RES_TYPE;
      }
    }
  }
</syb>
</body>
</html>

3) 分目录浏览页面 - nav_s.hts
(html)
<title>新闻资料浏览----分目录</title>
<syb type=perl>
my $handle=ws_connect("pass");
my $catalogue_id=$s_form("s_catalogue_id");
my $sql_stmt=qq! select * from news_material..sub_catalogue where
  total_catalogue_id=$catalogue_id;
my $rc=ct_sql($handle,$sql_stmt);
if($rc!=CS_SUCCEEDED){
  ws_error("ct_sql() call failed.");
  while(($rc=ct_results($handle,$result_type))--CS_SUCCEED)
  {
    RES_TYPE,{
      if($result_type==CS_ROW_RESULT){
        my@row;
        while(@row=ct_fetch($handle))
        {
          $link=qq! nav_r.hts? s_sub_id=$row[0];
          printf"<a target='s_main' href=%s>%s</a><br>",$link,$row[2];
        }
        last RES_TYPE;
      }
    }
  }
</syb>
</body>
</html>

4) 记录浏览页面 - nav_r.hts
```

```

(html)
<title>新闻资料浏览----记录概况</title>
<syb type=perl>
    my $handle=ws_connect("pass");
    my $sub_id= $ws_form("s_sub_id");
    my $sql_stmt=qq! select news__material..record.record__id,news__
material..record.tape_number,\
        news__material..record.time_code,news__material..record.
news__title\
        from news__material..record \
        where news__material..record.sub_catalogue_id= $sub_id!;
    my $rc=ct_sql($handle,$sql_stmt);
    if($rc!=CS_SUCCEED){
        ws_error("ct_sql()call failed.");
    }
    while(($rc=ct_results($handle,$result_type))==CS_SUCCEED
    {
        RES__TYPE,{
            IF($result_type==CS_ROW_RESULT){
                my @row;
                print"table border width='100%'";
                print"<tr>";
                print"<td align=center width='50%'>新闻标题</td>";
                print"<td align=center width='20%'>序号</td>";
                print"<td align=center width='30%'>时间码</td>";
                print"</tr>";
            }
        }
    }

```

```

while(@row=ct_fetch($handle))
{
    print"<tr>";
    printf"<td align=center>%s</td>",$row[3];
    printf"<td align=center>%s</td>",$row[1];
    printf"<td align=center>%s</td>",$row[2];
}
print"</table>\n";
last RES__TYPE;
}
}
</syb>
</body>
<center>
    <A HREF="javascript:history.go(-1)">IMG SRC="images/prev.jpg"
WIDTH=49 HEIGHT=32ALIGN=center ALT="Previous"BORDER=0</A>
</center>
</html>

```

5 结束语

Web.sql 加上 Sybase 数据库提供了一个为企业内部 Intranet 发布数据的一个切实可行的方法。通过 Web.sql 将 Web 服务器和数据库服务器透明地集成在一起,企业可以迅速向 Internet/Intranet 平台迁移。

自由曲面视觉测量中 一种有效的粗差别除方法

陈棟湘 罗飞路 朱霞辉

(国防科技大学机械电子工程与仪器系长沙 410073)

摘 要 自由曲面的测量技术广泛应用于工业生产的各个部门。立体视觉测量是一种很有潜力的测量方法,但其中涉及到的粗差别除是一个难题,本文提出序列建模预报法来剔除三维测量数据中的粗差,该方法具有速度快、准确性高的优点。

关键词 自由曲面 立体视觉 粗差 建模预报

1 引言

在工业生产的许多领域中,例如汽车、造船、航空、内燃机和模具等制造部门,都经常涉及到自由曲面的测量问题。传统的测量方法是采用三坐标测量机,这种方法的测量精度较高,但是测量速度慢,并且容易对工件表面造成损伤。立体视觉测量方法利用计算机视觉的理论和从图象中恢复出物体的三维信息,由于其具有非接触测量、成本低、速度快、自动化程度高等优点,近年来在三维测量领域取得了非常迅速的发展。立体视觉测量的基本原理是:采用两台摄像机从不同的位置对物体摄像,计算机通过对两个位置图象的特征提取和匹配得出测量点在两幅图像平面上的坐标,再利用成像公式计算出测量点的空间坐标。采用这种方法进行三维测量,其相对精度可以达到万分之一以

上。但是由于图像信息具有很大的随机性,导致在图像匹配过程中可能会出现误搜索,从而使图像坐标不能准确获取,影响到立体视觉测量的精度。当图像坐标出现比较大的扰动时,所获得的物体的三维坐标相应地也会产生比较大的误差,这就是测量中经常会遇到的粗差。

数据处理中已经存在很多剔除粗差的方法,但现有的方法大多是以对同一物理量作重复测量所得到的数据作为研究对象,难以满足三维测量的实际需要。本文从分析图像坐标扰动对测量精度的影响入手,根据自由曲面视觉测量的实际情况,提出采用序列建模预报法来剔除三维测量数据中的粗差,在应用中取得了良好的效果。

2 图像坐标扰动对测量精度的影响

摄像机模型的直接线性变换(DLT)方程可用下式表示^[1]:

$$\begin{aligned} u &= \frac{k_1x + k_2y + k_3z + k_4}{k_9x + k_{10}y + k_{11}z + 1} \\ v &= \frac{k_5x + k_6y + k_7z + k_8}{k_9x + k_{10}y + k_{11}z + 1} \end{aligned} \quad (1)$$

式中 (x, y, z) 代表测量点的空间坐标, (u, v) 代表像点的坐标。 $k_1 \sim k_{11}$ 是摄像机的11个成像参数, 需要由标定得到。

设左右两个摄像机的成像参数分别为 $(l_1 \sim l_{11})$ 和 $(r_1 \sim r_{11})$, 测量点在两个图像平面上所成的像分别为 (u_1, v_1) 和 (u_r, v_r) , 则根据公式(1), 可以导出如下的矩阵表达式^[2]:

$$\begin{bmatrix} 1-u_1l_9 & l_2-u_1l_{10} & l_3-u_1l_{11} \\ l_5-v_1l_9 & l_6-v_1l_{10} & l_7-v_1l_{11} \\ r_1-r_9 & r_2-r_9r_{10} & r_3-r_9r_{11} \\ r_5-v_r r_9 & r_6-v_r r_{10} & r_7-v_r r_{11} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} u_1-l_4 \\ v_1-l_8 \\ u_r-r_4 \\ v_r-r_8 \end{bmatrix} \quad (2)$$

简记为 $AX=B$

当图像坐标出现扰动 $\Delta u_1, \Delta v_1, \Delta u_r, \Delta v_r$ 时, 设空间坐标的误差为 $\Delta x, \Delta y, \Delta z$, 则上述矩阵方程变为

$$(A + \Delta A)(X + \Delta X) = B + \Delta B \quad (3)$$

略去上式中的高阶小量 $\Delta A \Delta X$ 后, 可以解得

$$\Delta X = (A^T A)^{-1} A^T (\Delta B - \Delta A X) \quad (4)$$

由于 $\Delta B - \Delta A X$ 可以分解为

$$\Delta B - \Delta A X = D \cdot [\Delta u_1, \Delta v_1, \Delta u_r, \Delta v_r]^T \quad (5)$$

式中

$$D = \begin{bmatrix} a & 0 & 0 & 0 \\ 0 & a & 0 & 0 \\ 0 & 0 & b & 0 \\ 0 & 0 & 0 & b \end{bmatrix}$$

其中:

$$a = l_9x + l_{10}y + l_{11}z + l_4,$$

$$b = r_9x + r_{10}y + r_{11}z + l_8,$$

将式(5)代入式(4)中, 可以得到空间坐标误差与图像坐标扰动的关系表达式:

$$[\Delta x, \Delta y, \Delta z]^T = C \cdot [\Delta u_1, \Delta v_1, \Delta u_r, \Delta v_r]^T \quad (6)$$

式中 $C = (A^T A)^{-1} A^T D$, 是一个 3×4 阶的矩阵。可以看出, 对于某一具体的测量点, C 是一个常数矩阵。

由于矩阵 C 跟摄像机的成像参数和具体的测量位置有关, 所以 $\Delta x, \Delta y, \Delta z$ 同 $\Delta u_1, \Delta v_1, \Delta u_r, \Delta v_r$ 之间的关系很复杂, 但我们通过对实际测量数据的分析发现, 在一般情况下, 当图像坐标出现比较大的扰动时, 测量点空间坐标的三个分量都会产生比较大的误差, 也就是说, 该点将成为一个粗差点。

3 序列数据的三维混合模型

在自由曲面的立体视觉测量中, 可以利用激光扫描装置在工件表面形成高亮度的扫描线, 从扫描线上提取特征点进行三维坐标测量。因此, 同一扫描线上的测量点构成一个三维的空间序列, 可以利用动态数据统计分析的方法来对其进行处理。最自然的想法是建立一个三维自回归滑动平均模型, 通过模型参数的估计、模型的预报来识别粗差。但是多维模型的参数估计

是一项非常复杂的工作, 同时多维模型的阶数确定则至今也尚未得到很好的解决^[3]。另一方面, 由于测量点分布在自由曲面表面, 同一扫描线上的测量点实质上构成一条空间曲线, 因此用回归模型来对测量数据进行拟合也具有一定的物理意义。基于这两方面的考虑, 我们确定建立回归与自回归的混合模型。

对于测量数据的 x, y, z 三个分量, 可以分别建立如下三个混合模型^[4]:

$$x_i = \sum_{j=1}^{n_1} (a_{1j}x_{i-j} + b_{1j}y_{i-j} + c_{1j}z_{i-j}) + \epsilon_x \quad (7)$$

$$y_i = \sum_{j=1}^{n_2} (a_{2j}x_{i-j} + b_{2j}y_{i-j} + c_{2j}z_{i-j}) + \epsilon_y \quad (8)$$

$$z_i = \sum_{j=1}^{n_3} (a_{3j}x_{i-j} + b_{3j}y_{i-j} + c_{3j}z_{i-j}) + \epsilon_z \quad (9)$$

式中模型阶次 n_1, n_2, n_3 根据FPE准则进行确定。

式(7)到式(9)可以写成如下的统一形式:

$$\begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} = \sum_{j=1}^n \begin{bmatrix} a_{1j} & b_{1j} & c_{1j} \\ a_{2j} & b_{2j} & c_{2j} \\ a_{3j} & b_{3j} & c_{3j} \end{bmatrix} \cdot \begin{bmatrix} x_{i-j} \\ y_{i-j} \\ z_{i-j} \end{bmatrix} + \begin{bmatrix} \epsilon_x \\ \epsilon_y \\ \epsilon_z \end{bmatrix} \quad (10)$$

式中 $n = \text{Max}(n_1, n_2, n_3)$

这正是多维自回归模型的表示形式。可见式(7)到式(9)是向量模型的标量表示方法, 只是三个混合模型的阶次不一定相同。

4. 双向建模预报

式(7)到式(9)这三个混合回归模型的参数估计与预报方法与一般的自回归模型相似。根据序列预报理论, $AR(n)$ 模型的1步预报公式为

$$\hat{x}_i(l) = \sum_{j=1}^n \varphi_j x_{i-l-j} \quad (11)$$

$\hat{x}_i(l)$ 作为 x_{i-l} 的估计的95%概率的置信区间为

$$\hat{x}_i(l) \pm 1.96\sigma_x \sqrt{\sum_{j=1}^{l-1} P_j^2} \quad \hat{x}_i(l) \pm 1.96\sigma_x \sqrt{\sum_{j=1}^{l-1} P_j^2}$$

其中 σ_x 是模型参数估计的均方差, P_j 是所建模型的格林函数值, 当预报长度 $l=1$ 时, 可知预报值 $\hat{x}_i(l)$ 服从

均值为 $\hat{x}_i(l)$, 方差为 σ_x^2 的高斯分布。

从时间序列分析的理论与实践可以得知, 序列信息的充分利用对于提高模型参数估计的精度和预报的精度都是有好处的^[3]。因此可采取双向建模预报的措施来提高预报精度。

具体的做法是:

根据序列数据从正向和反向各建一个模型进行一步预报, 设 x 的正向预报值为 x_i , 反向预报值为 x_b , 则取 x 的双向预报值为

$$\hat{x} = 0.5 * (x_i + x_b) \quad (12)$$

设 x_i 服从高斯分布 $N(x_i, \sigma_i^2)$, x_b 服从高斯分布 $N(x_b, \sigma_b^2)$, 则 \hat{x} 服从高斯分布 $N[0.5 * (x_i + x_b), 0.25 * (\sigma_i^2 + \sigma_b^2)]$

$\sigma_x^2]$ 。当 $\sigma_x^2 = \sigma_y^2 = \sigma_z^2 = \sigma^2$ 时, $\frac{\Delta}{x}$ 的方差为 $0.5\sigma^2$, 比单向预报的方差减小了一半。

粗差的判断准则按以下方法确定: 对于测量数据 (x, y, z) , 若其双向预报值为 $(\frac{\Delta}{x}, \frac{\Delta}{y}, \frac{\Delta}{z})$, 则当

$$\sqrt{\frac{\frac{\Delta}{x}^2}{(x-x)^2 + (y-y)^2 + (z-z)^2} + \frac{\frac{\Delta}{y}^2}{(x-x)^2 + (y-y)^2 + (z-z)^2} + \frac{\frac{\Delta}{z}^2}{(x-x)^2 + (y-y)^2 + (z-z)^2}} > 1.96$$

$$\sqrt{0.25 \cdot (\sigma_{1x}^2 + \sigma_{1y}^2 + \sigma_{1z}^2 + \sigma_{2x}^2 + \sigma_{2y}^2 + \sigma_{2z}^2)}$$

时, 即可判定 (x, y, z) 为粗差点, 置信概率为 95%, 其中 $\sigma_{1x}^2 \sim \sigma_{2x}^2$ 分别是三个混合回归模型的正向和反向最终预报方差。

5 结论

在三坐标视觉测量中, 采用序列建模预报法剔除

自由曲面测量的粗差, 具有速度快、准确性高的优点。双向建模预报充分利用了序列数据中的信息, 使预报精度得到了提高。

参考文献

- 1 D. H. Ballard and C. M. Brown, Computer Vision, Prentice-Hall, Englewood Cliffs, New Jersey, 1982
- 2 罗飞路, 陈揆湘, 张屹, 周继伟. 自由曲面的立体视觉测量与加工一体化研究, 国防科技大学学报, Vol. 17, No. 2, June 1995, PP. 12-18
- 3 杨叔子, 吴雅. 时间序列分析的工程应用, 华中理工大学出版社, 1992
- 4 安鸿志, 顾岚. 统计模型与预报方法. 气象出版社, 1986

PC 机与家电联网的集散系统

胡劲松 吴捷

(华南理工大学电力学院研究生部 广州 510641)

摘要 文章提出利用通信电路和单片机接口电路, 将 PC 机与家电联网组成一个集散系统, 并给出了 2 个具体的例子。PC 机与家电联网将引发一场家电革命, 有无可限量的前景。

关键词 集散控制 通信接口 单片机

1 引言

如果你有位“管家”, 每天按时替你做饭菜, 洗净衣物, 你只要吩咐几句, “她”马上调好音响, 放你想听的曲子, 你不必费心去掌握复杂的操作方法, 一到特定的时间, “她”自动把电视机调到你原订的特别想看的节目, 你不必担心因忘记而错过, “她”还会二十四小时替你看家防盗、防火……。这位能干的“管家”就是也许不久将进入普通家庭的集散家电系统。

2 系统的原理

集散控制系统(DCS)本是指利用计算机技术对生产过程进行集中监视、操作, 管理和分散控制的一种新型技术。本文首次将这一概念引入对家电的统一管理和控制。现代家电已有很多采用单片机控制, 因此可以通过 PC 机对家电统一管理和控制, 使家电的功能极大的增强, 实现家务劳动和家庭管理的自动化。系统总图如图 1 所示。说明: PC 机经过接口电路和电缆与各种家电相接, 另一路是中断请求线, 作报警用。还有一路话筒用于输入口令, 一路喇叭输出提示。本系统的关键是 PC 机与各家电的通信问题。考虑到目前各种家电采用的单片机相差很远, 直接联成一个统一的通信系统基本上不可能(这正是难点所在)。要求所有厂家采用同一类单片机更不现实。本文在基本不改变各种家电的线路元件及程序的前提下, 及程序的前提下, 采

用增加一块 51 系列单片机和接口电路的方法解决这一问题。

通信原理及过程: 1) 当 PC 机遇到下列三种情况发生中断: (1) OXIC 软中断。中断服务的内容是一个相当于定时控制器的程序, 在设定的时间对指定的家电进行控制。(2) 话筒口令中断。话筒收到口令后, 由声音处理电路经 8259 向 PC 机发出中断。然后 PC 机对声音输入进行识别, 转译成控制信号发往家电执行。(3) 警报信号经报警线和中断请求线 IQR 向 PC 机发中断信号。2) PC 机为主机, 51 系列单片机为从机。PC 机用异步串行通信方式和从机联络。每台从机分配一个地址。PC 机发一个地址, 地址符合的那台从机发生中断响应, 与主机通信。其它从机不响应。细节参看文献[1]。3) 从机向主机汇报家电当前状态。4) 主机发控制命令。5) 从机接受命令并将命令翻译成控制信号, 经外围电路输入家电本身的微控制器(也是单片机), 再由微控制器控制家电。6) 从机向主机汇报执行完毕后退出中断。通信距离大于 1000M, 波特率 2400, 因采用光电耦合电流环接口, 系统的安全性、可靠性高。主机只有在前述 3 种中断发生时才转入家电服务程序, 绝大部分时间主机可作它用。家电可以由 PC 机控制, 也可以人工操纵。下面用两个实例阐明这一方法。

3 彩电的通信与控制

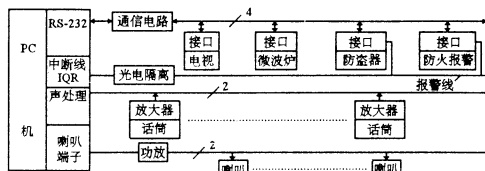


图1 系统总图

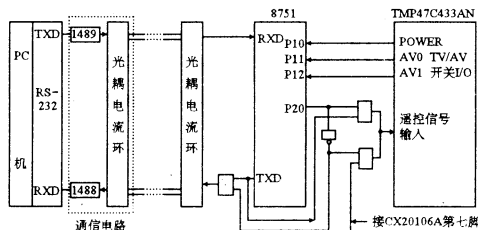


图2 彩电的控制

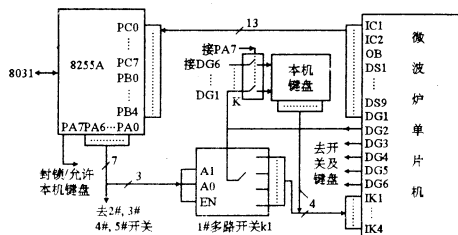


图3 微波炉的控制

增加的功能:除原机的所有功能外,主要增加了:

1)高级定时功能。在指定的日期和时间开机并切换到指定的频道。2)目前彩电的色彩、音量不能随频道的变化自动变化。经常有这样的情况:A台色彩逼真、音量适中,B台信号比A台弱,B台色彩偏淡,音量小,如果从A台换到B台同时要获得好的收视效果,就还要调节色彩和音量。这是很烦人的。本系统实行模式参数联动,频道、色彩、音量同步变化。3)与其它家电联动。放VCD、录像机时,自动开电视,自动切换到相应频道和AV端子。4)只要在PC机上增加程序,还可以实现更多的功能。8751单片机控制彩电原理:电路如图2,以常见的长虹C2161为例。8751经P₁₀~P₁₂从彩电微控制器输入电视当前状态汇报给PC机,PC机发控制命令,8751接收命令并把命令翻译成彩电遥控编码脉冲,在对串行口重新初始化后,以560BPS的波特率从TXD端输入彩电微控制器,由微控制器对彩电进行控制。因TXD还用于和主机通信,另外为避免来自红外遥控接收电路CX20106A的信号与8751的串行信号冲突,P₂₀作用于三个与门和一个非门,进行分时控制,整个过程就PC机经过接口,把参数输入电视机本身的微控制器,再由微控制器进行控制。录像机、音响、空调等带遥控功能的家电都可采用与此相似的方法控制。

4 微波炉的控制

电路如图3。增加的功能。1)、PC机存有自种食物

的多种烹调程序。2)你可以一次制订多天的食谱,到时候系统通过喇叭提醒你当天应准备的原料,之后自动按预定方式烹调。3)可以遥控甚至可以用电话或INTERNET网在异地命令系统对烹调方式和时间进行修改。例如你因故推迟回家。4)可以方便地输入模糊控制等先进算法。

4.1 控制原理

这是一台中低档的微波炉,接入系统后它具有高档的功能。IC₁是炉门信号,OB是蜂鸣器信号,作为烹调结束和确认键操作有效的音响信号。IC₂是确认探针插好信号,DS₁~DS₅和DG₁是显示运行阶段、温度、时间的输出脚,以上信号全部由8031的扩展I/O口输入,再反馈给PC机。8255A是扩展I/O片。IK₁~IK₄与DG₂~DG₆是炉子的键盘矩阵电路。8255A的PA口控制着5个4路单端输入模拟多路开关(图中只画了一个)。多路开关的A₁、A₀线选择通路,EN脚是开关本身的选中信号。总共是5×4=20种开关状态,相当于炉上的20个按键(实际上用18个)所以开关的闭合就相当于对按键的操作。PC机就可以通过8031、8255A、多路开关把控制参数输入微波炉本身的单片机,通过它间接控制炉子。PA₇脚控制多路开关K,用于防止PC机控制和本机键盘冲突。通过机上按键操作的家电,如洗衣机、电磁炉可采用与此相似的方法。

4.2 报警系统原理

单片机经报警线向PC机发中断请求,主机终止其它任务,用逐台查询方式与报警的那台从机联系,之后用喇叭报告警情和地点。

5 PC机的人机对话接口

1)、通过屏幕菜单选择。2)、通过遍布各房间的话筒输入声音。声音识别有一定难度,IBM已开发出英语识别,汉语识别尚在研制。所以本系统暂时只能用特定方式“编码”的击掌、口哨等声音控制。

6 结束语

该系统采用廉价的51系列单片机,结构简单可靠的光电耦合电流环,系统安全性高,运行稳定,其价格完全能被普通家庭接受。集散家电系统将引发一场家电革命,给人们的生活带来深刻的变化。国外尚无此类产品,国内各厂家应联手开发,制定统一的通信协议,为自己的产品加上统一的接口,用户只须把它们和PC机用电线联起来就可以方便的组成一个系统了。

参考文献

- 1 何立民. 单片机应用系统设计. 北京, 北京航空航天大学出版社, 1994年.
- 2 常宏. 长虹牌遥控彩电原理使用与维修指南. 北京, 电子工业出版社, 1994年, 107—155
- 3 刘景昌等. 家用电器维修与测试技术. 郑州, 河南科学技术出版社, 1993, 3: 542—549

一种新的消息传递型多处理机 系统中总线接口的设计

屈玉贵 赵 静 张海勤

(中国科技大学合肥 230027)

摘 要 多处理机系统分为共享存储器型和消息传递型。本文提出了一种新的消息传递型多处理机系统的总线接口的设计方法。区别于以往的总线接口的方法,引入了口的概念。优点是使得接收方也有权决定信息包接收与否,算法更灵活,可用广播传送方式,还可以动态地进行进程转移。

关键词 多处理机系统 总线接口

1 前言

多处理系统的通信问题非常重要,它直接影响整个系统的效率。多处理机系统的通信方式分为共享存储器型和消息传递型。共享存储器型,是多个处理机共享一个存储器模块,把共享存储器模块作为一个同步变量。这种方法会因访问共享存储器冲突而使系统传输效率下降。消息传递型,是各处理机把要进行交换的数据打包成包,写入发送消息总线接口,然后发往目的处理机。接收方的消息总线接口收到发来的信息包后,通知自己的处理机,告知信息包已到达,可以进行处理。在信息传递方式下,一次传递的信息包可达几十个字。而共享存储器方式一次访问为一个单位。消息传递方式可减少总线仲裁次数,提高传送效率。各处理机都有几乎相同的总线接口,互换性好,易扩充,且编译也简单。

以往的总线接口,是由发送方处理机来决定传送目的地。每个处理机都有一固定的 ID 号。传送的信息包中带有 ID 号,按 ID 号去找接收方处理机。这种方式的 ID 都是用硬件固定设计在各处理机中,程序是不能改变的。在编程时必须服从这种安排。这样,就使得通信算法复杂,且在进程转移时无法动态地决定接收处理机,为此我们引入了口的概念。

在口 ID 方式中,口 ID 是识别接收处理机的标号,而处理机是被独立管理的序号。每个处理机都有多个口。信息包中带有口 ID,只有与之对应的口打开的处理机,才接收这个信息包。各处理机中的口都可以独立控制其开、关状态。所以在引入口以后会带来以下好处:

1.1 接收方也有权决定是否接收总线上的某个信息包

以前的固定处理机 ID 方式,只能由发送方来决定信息包的接收地址。接收方除了在接收缓冲器满时拒绝接收外,都将无条件的接收发给自己的信息包,无权

决定收还是不收。这样就会在通信频繁时,增加从接收缓冲区取数进程的负担。如果接收方也有权决定接收某个信息包,就可以有选择的接收某些信息包,从而减少了多余的取数负担。

1.2 通信算法灵活

采用口 ID 方式可使通信算法更灵活。在共享资源的互斥管理中的信号量管理进程,如用固定处理机 ID 方式,必须要事先知道这个信号量在哪个处理机中,而在口 ID 方式中,不管信号量在哪个处理机中,一旦决定了对应于这个信号量的 ID,就可以向这个口中传递信息了。哪一个处理机中有这个信号量,自然就会把这个口打开。这样,就使得通信算法变得灵活了。

1.3 可以广播传送

固定处理机 ID 方式,完全是一对一的通信方式。而口 ID 方式中,口开着的所有处理机都可以接收该口的信息。可实现一对多的广播通信方式。

1.4 可以动态地实现进程转移

如果要想实现处理机间的进程转移,只要把转出处理机中该进程的口关闭,转去处理机的该进程的口打开就可以了。

2 消息传递总线接口方案

为充分发挥信息传递效率,而采取了各种措施,具体体现在以下几个方面。

2.1 在总线接口中设有发送缓冲区和接收缓冲区

在每个处理机总线接口中都设有发送缓冲区和接收缓冲区,每个缓冲区都可以容纳多个信息包,来平衡处理机的处理速度与总线传递速度之差。因为信息包传送的过程是:

(1)发送方处理机把信息包写入自己的发送缓冲区。

(2)发方总线接口等待总线空,一旦总线空,马上把信息包发往接收处理机的总线接口的接收缓冲区。

(3)总线通知接收处理机,告知信息包已到达。于是接收处理机中断原操作,从自己的接收缓冲区中取信息。

按照这样的顺序,如果缓冲器少,则会因处理机的处理速度跟不上总线传送速度而使缓冲器满,造成传送瓶颈,影响总线效率。为此必须设计多个缓冲器,以容纳多个信息包来平衡处理机处理速度和总线传送速度的差。在此把缓冲区大小设定为16个信息包的空间。

2.2 传送周期

如果采用时钟周期为8MHz的处理器,消息传送总线接口的时钟设为16MHz。在没有等待的情况下,CPU访存读需大约500ns,访存写大约需650ns。再加上读取指令时间,大约存取一个字的时间再1000ns以上。在消息传送总线接口中,2个时钟周期可传送一个字,所以需要125ns的传送时间。因此,与使用VME总线的CPU的直接访问共享存储器相比,可得到约10倍的传送效率。若能提高总线传送量,与连接的共享处理机台数成比例,则整体效率还会更高。与共享存储器方式,及局部存储器方式相比,达到总线饱和的台数可达数百台以上。

2.3 信息包的大小

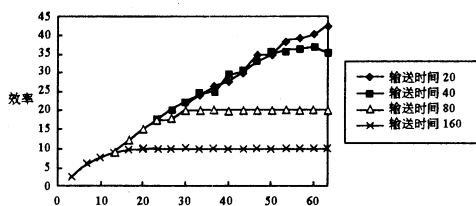


图1 信息包的长度,处理机台数与总线效率的关系图

从图1模拟的结果曲线可看出信息包的大小对总线效率的影响关系。传送时间为传送一个信息包到目的处理机的CPU周期数。如果一个信息包的传送时间为20个时钟周期,则连接60台处理机都没能达到总线饱和。所以信息包的长度可达16字乃至更长。

2.4 正确进行进程的态转移的控制

如前所述,口ID方式中,当在处理机之间进行进程转移时,只要关上原处理机中该进程的口,打开新处理机中该进程的口就可以了。但是,如果处理机上的开、关口有误,就会把进程传递到其他处理机上去了。也就是说,如果在进程转移时,口的开关没控制好,会在瞬间出现发、收双方的进程口同时开启或同时关闭的现象。这样就会影响进程转移的正确性。

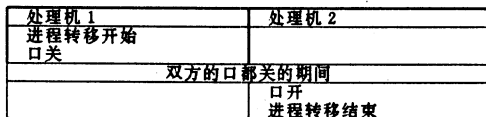


图2 进程转移中的失误

为解决这个问题,我们在消息传递总线接口上设计了根据消息包来决定口开或关的控制。而在消息传

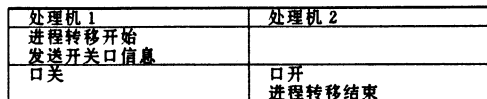


图3 进程转移成功

递总线上的消息包中也设置了特别的标志。通过这种标志信息,就可以由总线来改写各处理机口ID的开与关。由于在消息传递总线上的最小传送单位是一个信息包,如果用信息包中的某位标志来改写口,就不会存在图2所示的瞬间会出现收、发双方的口同时开或关的现象了。图3所示。

3 消息传递总线接口实现

3.1 消息传递总线接口框图

根据以上设计思想,得出的消息传递总线接口的总框图如图4所示。分为发送,接收,仲裁三大部分:

发送部分的功能是,当收到来自CPU的信息包写入命令时,开始启动,向仲裁器提出总线使用请求,得到仲裁许可后,便把信息包发往消息传递总线。

接收部分的功能是,发现总线上的信息包是发给自己的,则把该信息包收入接收缓冲器。同时通知自己的CPU,告知信息包已经到达。

仲裁部分的功能是,进行总线仲裁。当收到本发送器送来的发送请求后,等待总线使用权。得到使用权后,告知发送器可以使用总线。各处理机仲裁器采用循环标志法。



图4 总体框图

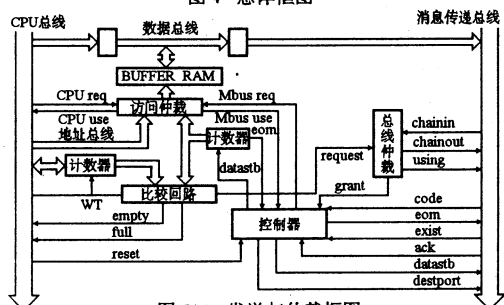


图5(a) 发送与仲裁框图

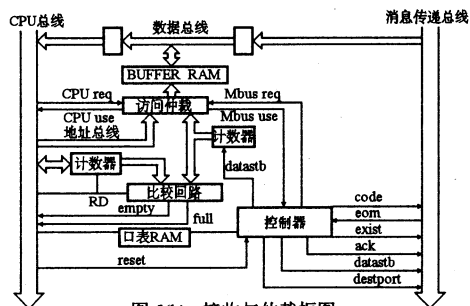


图5(b) 接收与仲裁框图

由图5(a)可知,发送部分由5个部分组成,分别是发送缓冲区,访问缓冲区冲突仲裁器,计数器,比较

器和控制器。图 5(b)可知,接收部分由 6 个部分组成,分别是接收缓冲区,访问缓冲区冲突仲裁器,计数器,比较器,控制器和口 ID 存储器。

3.1.1 发送缓冲区和接收缓冲区

发送缓冲区和接收缓冲区的结构基本相同。都是使用 256 字,访存周期为 25ns 的高速 RAM。分成 16 个信息包,一个信息包有 16 个字。还可以设计成其他组成方式。

发送缓冲区指针分为 CPU 指针和消息传递总线 MBUS 指针。MBUS 指针指示传送到消息传递总线的信息包的地址,是计数器的输出,也是信息包数据发送的选通信号。由 DATASTB 同步信号来触发计数器加 1。开始时可用 CPU 初始化为“0”。CPU 指针指示下一次要写入的信息包的缓冲区的地址,用写信号使其指针计数器加 1。总之,CPU 指针指示要写入缓冲器队列的地址,而 MBUS 指针指示要从缓冲队列中读出的地址。当 CPU 指针和 MBUS 指针一致时,说明有两种可能。一是因 CPU 指针的变动而造成不一致,则说明发送缓冲区已满;二是因 MBUS 指针的变动而发生不一致,则说明发送缓冲区空。缓冲区满时,CPU 不能再往里写数,缓冲区空时,不能再向总线发送信息包。两种情况都要进行相应的中断处理。接收缓冲区与发送缓冲区基本相同。只是 CPU 指针指示要从接收缓冲区取数的地址,而 MBUS 指针指示要从消息传递总线写入缓冲队列的地址。且 CPU 指针的变动发生一致时,说明接收缓冲区空,当 MBUS 指针的变动而发生一致时,说明接收缓冲器满。正好与发送缓冲器的情况相反。当两个指针不一致时,还要通知 CPU 取数。当接收缓冲区满时,要把满的状态信号返送给发送方。

3.1.2 缓冲区访问冲突仲裁模块

由于发送缓冲区和接收缓冲区都可以为 CPU 和消息传递总线双方服务,就有可能发生双方同时访问缓冲区的情况。用缓冲区访问仲裁来解决这个矛盾。发送和接收的原理相同。

在此设计成 CPU 访问缓冲区的时间单位是一个字,而消息传递总线访问缓冲区的时间单位是一个信息包。任何一方正在访问缓冲区时,另一方必须等对方完成这个访问时间单位后,才能访问。所以在设计时要充分考虑这个时间等待。

我们把使用情况分为 5 种状态,用 3 位二进制数表示为:双方都不访问缓冲队列 000;CPU 正在使用缓冲区 001;MBUS 正在使用缓冲区 011;MBUS 等待 010;CPU 等待 100。又设,MR 为 MBUS 的请求信号;CR 为 CPU 的请求信号;MU 为 BUS 的允许信号;CU 为 PU 的允许信号。都是“1”有效。则状态图如图 6 所示。由于这种逻辑关系比较复杂,且在发送,接收,及口存储器的仲裁管理中都用到了。所以可设计成一个专用 PAL 芯片。

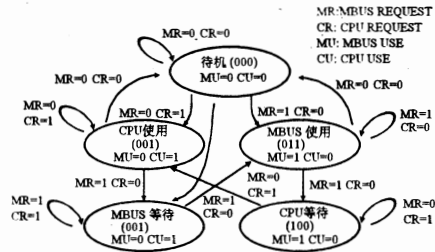


图 6 缓冲区访问仲裁状态转换图

3.1.3 控制部分

控制部分产生各种控制信号,控制整个接口的工作。控制信号定义如下:

MBUS 控制信号有:

数据线:三态,16 根

发送控制: DATASTB;三态。

DESTPORT;三态。

EOM; 三态。

接收状态: EXIST;线或。

ACK;线或。

CODE;线或。

仲裁控制: USING;线或。

CHAIN IN;

CHAIN OUT;

说明:

• DATASTB:时钟同步信号。

是发送和接收的同步信号,相对于 8MHz 的 CPU,有 120ns 的周期。连到所有处理机接口。

• DESTPORT:检测口信号。

指示口 ID 已放在数据线上,和信息包的第一个字同时有效。有效时可控制各处理机中的控制器去查口。当所有的 ACK 信号都收到后,说明接收方已完成口 ID 的接收工作,才使 DESTPORT 无效。连到所有处理机接口。

• EOM:指示信息包最后一个字的标志。

连到所有处理机。

• EXIST:接收方符合有效信号。

发送方把 DESTPORT 和口 ID 信号一起发往总线,接收方如果有对应的口是可着的,则向发送方发出此信号有效。此信号为线或,即只要有一个处理机接口的 EXIST 有效,总线上的 EXIST 就为有效。如果几个 EXIST 有效,则同时把信息传到所有有效的处理机接口。当 DESTPORT 无效时,则 EXIST 也无效。

• ACK:接收方处理完口 ID 标志信息。

接收方确认口 ID 信号,若为开,则进行处理,使口 ID 有效只有当所有的 ACK 信号都有效时,总线的 ACK 才有效。(非符合口 ID,也要有 ACK 回答?)随 DESTPORT 的无效,而使其无效。

• CODE:接收缓冲器满时,发向发送方的错误指示信号。

• USING:总线忙。

总线忙,该信号有效。

• CHAIN IN:总线预约权标志信号。

在仲裁器中,采用巡回标记方式。此信号有效,表示本处理机接口得到总线预约权。各处理机采用串行连接,标记巡回的办法。

• CHAIN OUT:相邻处理机交出总线使用权信号。

CPUBUS 控制信号有:

• 数据线:三态。

• 地址线:三态。

• RESET:复位信号。

• WE:写信号。

• RD:读信号。

• REC BUF REQ:发送缓冲区请求。

• SEND BUF REG:接收缓冲区请求。

• REC BUF USE:发送缓冲区允许。

• SEND BUF USE:接收缓冲区允许。

• PORT REQ:口存储器请求。

• PORT USE:口存储器允许。

• INT:中断申请,当发生缓冲器满或定时都要中断 CPU 运行程序,进行相应的中断处理。

3.1.4 口 ID 存储器。

口 ID 存储器用于存放口 ID 的开关状态。每个口 ID 对应于一个存储器地址,如果存储器的内容为“1”,表示该存储器地址对应的口为开,若为“0”,表示为关。整个口 ID 存储器采用关联存储器。系统复位时对其进行初始化,当信息包第一个字的最高位为“0”时,对所有口 ID 进行查找。当第一个字的最高位“1”时,要对口 ID 进行写入。写口时,可直接返回 EXIST 及 ACK 信号,在 DATASTB 时钟控制下,把信息包中的变更口 ID,写入对应的处理机接收部份的口存储器中。

3.2 信息包的组成

信息包分为一般信息包和变更口 ID 信息包

• 一般信息包的组成是:第一个字为口 ID。其中 0~9 位为 ID,10~14 位备用,第 15 位为“0”,指示此信息包是一般信息包。以后各字为包体。

• 变更口 ID 信息包的组成是:第一个字的 0~9 位为 ID。10~14 位对应于 0~4 号处理机的变更情况。“0”为不变,“1”为变更。第 15 位恒为“1”,指示此信息包是变更口 ID 信息包。第二字以后的各字,与第一字的意义相同。是对应于 5 号以后各处理机号。而第 15 位是没意义的。

3.3 信息包传送过程及时序

信息包的传送过程如图 7 所示,发方 CPU 首先向总线接口发出 REC BUF REQ 请求信号,经访问仲裁器裁决,得到了访问权后,向发送缓冲区写入一个信息包。信息包的第一个字是口 ID 信号。并且使发送缓冲区的 CPU 指针加“1”。要发送信息包时,由发送器来比较 CPU 指针和 MBUS 指针。如果不一致,则说明缓冲区中有数。则向总线仲裁器发出总线使用请求 REQUEST。总线仲裁器收到 REQUEST 信号后,等待使用总线预约权 CHAIN IN。权到之后,等其它处理机传送完一个信息包后,向发送器发出 GRANT 信号,告知可以使用总线。于是发送器把发送缓冲队列中的信息包的第一个字口 ID 送到 MBUS 上,同时使 DEST-PORT 有效。接收器则监视 MBUS 上的信息包的第一个字口 ID。如果自己的这个口 ID 是开状态,则向 MBUS 返回一个 EXIST 信号。

如果接收缓冲器队列满,则向发方返回一个 CODE 信号。不管口开与不开,满与不满,都要在接收准备工作完成后返回 ACK 信号。当发送器收到来自所有的处理机的 ACK,又确认 EXIST 及 CODE 后,才开始发送包体。信息包的最后一个字有 eom 标志。所以当收方收到 eom 标志后,进入接收完成处理。即接收缓冲器的 MBUS 指针和 CPU 指针相比较,如果不一致,则接收缓冲器队列中有信息,于是向 CPU 发出中断 INT,请求 CPU 来取信息。

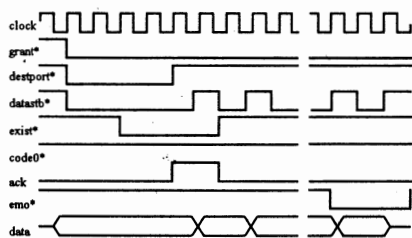


图 7 信息包接收时序 注: *为负有效

4 结束语

综上所述,我们设计的消息传递型处理机系统总线接口,从连接能力和传送效率都远优于共享存储器型。又由于我们引入了口的概念,使整个系统都能动态地决定信息包的传送地址。通信算法更灵活,并且当进程在各处理机之间动态转移时,也不需要复杂的过程,又能保证转移的可靠性。同时还可以实现广播传递方式。在发送和接收接口中,都设计有缓冲区,可以缓解处理机的处理速度,与总线传送速度不一致的矛盾,提高了总线利用率。该接口设计简单、连接方便、通用性好,适用于多种处理机的连接。

非网络环境下的计算机通信

祝雪妹

(南京动力高等专科学校自动化系 210042)

摘 要 本文简要介绍了非网络环境下的计算机通信方法,具有实用意义。

关键词 计算机通信 串行口 并行口

1 引言

在日常工作和工程实践中经常需要在计算机之间进行快速、简便的数据通信。在没有网络环境的情况下,一般采用将硬盘卸下来装入另一台计算机的方法。在此,本文将介绍几种在 DOS、Windows 下不用编程、简单实用的数据通信方法,具体有如下三种方法:

- 在 DOS 下利用串行口数据通信。
- 在 DOS 下利用并行口数据通信。
- 在 Windows 下利用调制解调器通过电话线路的数据通信。

以下结合使用的体会介绍这三种方法的使用和特点。

2 DOS 环境下的数据通信

2.1 利用串行口数据通信

在 DOS 环境下利用串行口在两台计算机之间进行文件传输,需要自制一根连接两台计算机串行口的连接线,具体方法如下:

材料: a. 两个 9 针或 25 针 D 型公插头。

b. 7 芯以上电缆若干米。

将两个 D 型插头按下图连接,未用引脚悬空。

串行口 1

串行口 2

9 针插头	25 针插头	7 芯电缆	9 针插头	25 针插头
5	7		7	5
3	2		3	2
7	4		5	8
6	6		20	4
2	3		2	3
8	5		4	7
4	20		6	6

图 1 串行口数据通信连接法

方法如下:

a. 在服务器上执行如下命令:

```
INTERSVR C:/COM2
```

若连接线接在服务器的串行口 1,则应将 COM2

改为 COM1。

b. 在客户机上,要在 CONFIG.SYS 文件中加入:

```
DEVICE = C:\DOS\INTERLNK.EXE/DRIVES:1/COM2
```

同样,若连接线接在客户机的串行口 1,则应将 COM2 改为 COM1。

重新启动客户机,以使上面命令生效。当要进行文件传输时在客户机上键入以下命令:

```
INTERLNK F=C
```

此命令将服务器的 C 盘映射成客户机的 F 盘,然后就可以在 F 盘和客户机上的其它盘间进行文件传输。这样相当于客户机多了一个 F 盘,对 F 盘文件存取实际上是对服务器 C 盘的文件存取。

这种方式下,服务器被独占不能进行其它工作,要退出服务器状态,按 Alt+F4 即可。两台计算机的连接随之终止。

2.2 利用并行口的数据通信

DOS 环境下利用并行口在两台计算机之间进行文件传输,同样需要自制一根连接两台计算机并行口(即打印机接口)的连接线,方法如下:

a. 两个 35 针 D 型公插头。

b. 11 芯以上电缆若干米。

将两个 D 型插头按下图连接,未用引脚悬空。

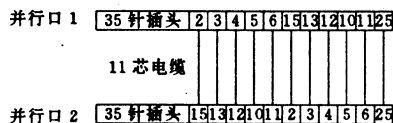


图 2 并行口的数据通信连接法

连接线做好后,将需要通信的两台计算机的并行口连接线连起来,仍以前述问题为例,服务器上的命令变为:

```
INTERSVR C:/LPT1
```

客户机上 CONFIG.SYS 文件中的命令行改成:

```
DEVICE = C:\DOS\INTERLNK.EXE/DRIVES:/LPT1
```

其它步骤与串行口通信完全相同。

3 Windows 环境下的数据通信

在 Windows 环境下可以进行远距离的文件传输,通信双方各需要一个调制解调器(Modem)或内置式传真卡(FAX),并有电话线连通。如图所示。具体步骤如下:

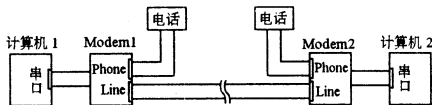


图3 远程通信连接法

a. 需要进行通信的两台计算机进入 Windows 3. x (西文、汉化均可),在附件(Accessories)组中有一个终端仿真程序(Terminal),执行该程序。

b. 选择设置(Settings)菜单中的通信(Communication)选项,设置一个最高波特率,根据我国实际情况,波特率不宜超过 4800,按确定按钮(OK)。

c. 进行通信的双方有一方称主叫方,另一方称被叫方。由主叫方拨号;主叫方在设置菜单(Settings)中选择拨号(phone Number),输入被叫方的电话号码,若是异地通信还应加上被叫方的长途区号。

d. 主叫方按电话(phone)菜单中的拨号(Dial)即开始拨号。

e. 被叫方听到电话振铃声后,会在窗口看到“RING”字样,被叫方键入“ATA”,作为应答。如果连接成功,双方均可在各自的窗口中看到“CONNECT 4800”字样,说明双方连接成功,接下来就可以进行文件传输了,既可由被叫方传往主叫方,也可由主叫方传往被叫方。

f. 假设被叫方有一文件需要传到主叫方,则被叫方需选择传输(Transfers)菜单中的发送二进制文件(Send Binary File),然后选取要传输的文件名,按确定(OK)按钮,即文件开始传送。

主叫方选择传输菜单(Transfers)中的接收二进制文件(Receive Binary File),然后键入接收“文件名”(无需与发送方同名)。

在文件传输期间,双方可通过信息窗口,了解文件传输进展情况。如果传输过程中发现了传输错误,则发生错误的数块,将被重新传输,在信息窗底部可以看到重试(Retries)次数,如果连续 6 次无法正确发送和接收某数据块,则文件传输终止。可以换条电话线或降低波特率重试。

在 Windows 下也可以直接利用串行口,不用调制解调器(Modem)进行短距离传输。方法类似,不再赘述。

4 结论

以上方法在工程实践中非常有用,方法简单易行,无需编程。前两种方法适合于近距离的环境,如同一办公室里的两台电脑,手提电脑和台式电脑之间。最后一种方法适用于远程通信。如果你编的软件在遥远的地方使用,一旦程序出现了一些小问题,你无需去现场修改,只需在你的办公室里修改好,把新的程序文本通过电话线传过去,既快又经济,省时省力。在实际的工作中,本人采用这样的办法解决了不少问题,特写此文推荐给大家。

参考文献

- 1 熊桂熹等译. Windows 3.1 使用指南. 清华大学出版社
- 2 夏德深等. PC 机使用大全. 东南大学出版社

用单片机实现与 I²C 总线器件的接口

孟维晓 徐玉滨 王 斌

(哈尔滨工业大学通信技术研究 150001)

摘 要 本文介绍一种新型的总线标准——I²C 总线的优点、结构、时序和数据格式,该总线标准适用于微处理电路及嵌入式控制,并以代表性器件 AT24C64 和 PCD3312C 为例,具体描述单片机与它的接口和通信过程。

关键词 I²C 总线 计算机通信

1. 引言

计算机通信方式广义上可以划分为并行通信和串行通信,并行通信速度快,实时性好,由于占用的口线多,不宜于小型化产品;而串行通信速率虽低,但在数据通信吞吐量不是很大的微处理电路中显得更加简单、方便、灵活。串行通信一般可以分为异步模式和同

步模式。RX232 接口标准就是采用串行异步通信模式。

I²C 总线标准是近年来在微电子通信控制领域被广泛采用的一种新型总线标准,它是同步通信的一种特殊形式,具有如下优点:

- 通信采用串行方式,只需要两条线,一条串行时钟,另一条为串行数据线;

- 通信的控制方式达到最简化,串行数据与应答

复用;

• 器件的封装形式可以做到很小,外围引脚十分简单;

• 在主从通信中,可以有多个 I²C 总线器件,同时挂到 I²C 总线之上。器件靠从地址(Slave address)来区分通信的对象。

• 通信的速率可以采用 100Kbps 以上最大到 400Kbps 的高速传输。

I²C 总线的接口原理如下图 1 所示:

SCL: 串行时钟

SDA: 串行数据

a, b...x: 地址选择译码输入

主器件一般为单片微控制器,每一个 I²C 总线上的从器件都有唯一的从地址,选择地址靠译码输入的逻辑电平,主 CPU 与它们之间分别通信的区别在于发送或接收有效数据之前,先下发从器件的地址,得到从器件的确认之后再发送或接收有效数据。不是本器件的地址,将不响应数据总线上的任何信息。

值得注意的是时钟和数据线必须上拉到高电平。

2. I²C 总线器件数据格式与时序

以通信应用中经常用到的两种 I²C 总线器件,EEPROM AT24C64 和双音多频发生器 PCD 3312C 为例。

2.1 AT24C64 的总线数据格式如下:

[S] [1] [0] [1] [0] [A] [2] [A] [1] [0] [R/W] [ACK] [X] [X] [AD] [3~AD] [4] [ACK] [AD] [7~AD] [0] [ACK] [D] [7~D] [0] [ACK] [P]

S: 表明起始条件

A₂~A₀: 表示 I²C 总线器件的从器件地址,软件地址的选择依赖器件外部连线电平。最大可以有 8 个同样的器件同时挂在同一 I²C 总线上。

ACK: 从器件对主 CPU 发给它的数据或地址的确认信息。

AD₁₃~AD₀: 某一从器件的内部数据地址,最高可达 8K 字节。

D₇~D₀: 串行数据。

P: 通信结束条件。

X: 表示对该位电平情况不做判别。

2.2 PCD3312C 的总线数据格式如下:

[S] [0] [1] [0] [R/W] [1] [0] [A] [0] [ACK] [X] [X] [D] [5] [D] [4] [D] [3] [D] [2] [D] [1] [D] [0] [ACK] [P]

其中,S、ACK、P、R/W、X 的意义与 AT24C64 一样。

A₀: 从器件的地址,同一 I²C 总线上同时只能挂两个这样的器件。

D₅~D₀: 串行数据,表明要发送的 DTMF 或其它音频的种类。

2.3 I²C 总线时序

I²C 总线上数据通信能否井然有序,主要决定于总线上的时序是否符合 I²C 总线的要求,I²C 总线对时序的要求十分严格,只要时序合理,传输误码率极低。

数据传输时序如图 2。

每个时钟脉冲传输一个数据比特,在时钟脉冲为高电平期间,数据线必须保持稳定,如果此时数据线上有变化将被理解为控制信号。

起始和停止条件如图 3 所示。

时钟与数据线均为高电平表明总线空闲,数据线从高电平到低电平的转变,同时伴随着时钟保持高电平被定义为起始条件 S;数据线从低电平到高电平转变,同时时钟线保持高电平被定义为停止条件。

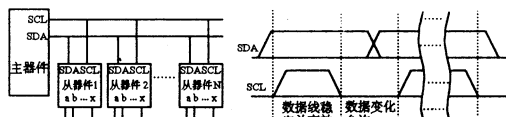


图1 I²C总线的接口原理

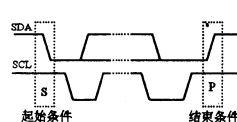


图3

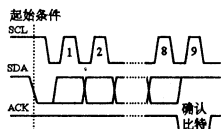


图4

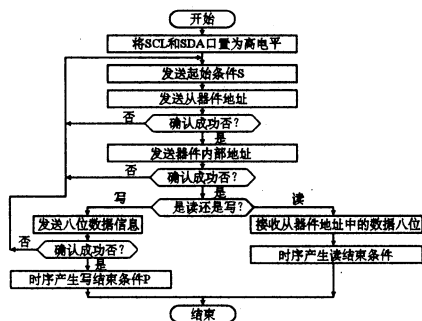


图5 I²C总线接口控制软件流程

接收确认

在起始和结束条件之间传输的数据字节数是没有限制的,每个八比特字节之后都跟随着一个确认比特。主发送器件将在外加接收确认脉冲将总线确认比特抬高,从接收器件在每接收一个字节后都将产生一个确认比特。器件产生的确认在确认脉冲期间会把数据线拉低。所以在与确认相关的时钟脉冲期间,数据线保持为低电平。时序如图 4。

3. 用单片机软件实现与 I²C 总线接口

任何一种单片机用软件均可以完成与 I²C 总线器件的接口,其中 SCL 为时钟输出,SDA 既为输出又为输入,只需要单片机的两个 I/O 口,参考流程如图 5。程序流程中的全部读写、发送、接收操作都要伴随着时钟脉冲条件,必须严格遵守 I²C 总线对时序的要求。

4. 结论

用三环开发器对控制软件进行仿真,结果稳定可靠。实践证明可以广泛应用于微处理电路及嵌入控制等通信与电子系统中。

参考文献

- 1 ATMEL Integrated Circuit Data Book Edition 3, 1994
- 2 HITACHI Semiconductor Devices for Telecommunications Sept. 1995
- 3 MDS-55 微处理器开发系统. 北京三环电子有限公司, 1993
- 4 张毅刚等, MCS-51 单片机应用设计. 哈尔滨工业大学, 1990. 8

ATM NIC(网络接口卡)的设计与开发

许冬 郑珏

(华北计算技术研究所太极计算机公司研究中心计算机通信室北京 100083)

摘 要 ATM NIC(网络接口卡)是计算机与 ATM 网络进行交互所必须的网络接口设备,本文就 ATM NIC 基本的硬件结构及功能实现进行了较为详细的论述,并根据笔者的开发实践介绍一种 ATM NIC 的设计思路。

关键词 ATM NIC(网络接口卡) SAR(分割与重组) CS-PDUs(会聚子层协议数据单元)

1 前言

目前,ATM(异步传输模式)技术作为一项全新的高速网络技术,已越来越为人们所关注。其传输的高效性、广泛的适用性及潜在的巨大优势,使之成为未来的通信网络技术发展的方向。

ATM 技术采用了面向连接的传输机制。在 ATM 网络中,站点间通信以建立在可靠的物理链路上的虚连接为传输途径,通过使用 53 字节长度的信元(cell)作为信息载体,携带用户信息进行传输。从 ATM 协议的层次结构来看,可将其划分为 PHY 层,ATM 层,AAL 层及高层协议,如图 1 所示。

高层协议		
CS(会聚子层)	AAL 层	
SAR(拆装子层)		
ATM 层		
TC(传输会聚子层)	PHY 层	
PM(物理媒体子层)		

图 1 ATM 协议的层次结构

ATM 技术的一个最大的优点在于 ATM 信元载体中可携带任何类型的信息(如语音,视频,数据),并支持多种类型的通信业务。而且,ATM 的传输速率可以不受限制,从理论上讲,ATM 能够支持 $n \cdot 51.84\text{Mbps}$ 的传输速率。同时,ATM 良好的可扩展性使它可以被广泛应用于工作组网至广域网的各类网络设计之中。

2 ATM 网络接口卡的设计目标及总体规划

ATM 网络接口卡(NIC)在 ATM 网络设备中占有重要的地位,是 ATM 网络中的关键性设备之一,网络中每台工作站、服务器及路由器都可能拥有多张 ATM NIC 以提供对多个网络段的访问。从实现的功

能来看,ATM NIC 主要负责数据的发送和接收,并保证用户数据传输的高效性和实时性。在 ATM NIC 的硬件设计方面,可以说既与其它类型的网络接口卡有着相似之处,又同时具有自己基本的设计要求。

1) 低价位。目前普遍使用的 Ethernet NIC 价格已相当便宜,ATM NIC 必须具有良好的价位才有可能与之相竞争。

2) 高性能。ATM NIC 必须能支持高效的带宽利用率及相应的高速率。此外,在处理多源发送及接收时,ATM NIC 还必须能够支持相当数量的虚连接。

3) 易开发性。由于 ATM 标准仍在进一步的发展之中,所以 ATM NIC 的设计方案中必须保留一定的余地以适应 ATM 标准的新变化。

4) 开发周期短,以适应市场的变化和需求。

下面,我们首先来讨论在 ATM NIC 产品设计之前,设计者应就哪些方面作一个总体性的规划,以使 ATM NIC 的设计方案具有最大的性价比。

首先,ATM NIC 所能支持的传输速率。目前有 25Mbps,100Mbps,155Mbps 等几种选择,但具体的实施必须依据对市场需求的分析和自身的技术条件。

其次,ATM NIC 所支持的传输介质。采用光缆还是双绞线,这同时也决定了传输速率的大小。

第三,ATM NIC 的设计针对性。对于不同类型的主机系统,对于诸如工作站、服务器、路由器等不同的使用环境和使用要求,对于采用何种总线结构,其 NIC 设计都需要在不同的方面加以取舍。

第四,如何在性能与成本之间加以权衡。这主要体现在 ATM NIC 设计中是否采用嵌入式微处理器及卡上的数据 RAM。

第五,应遵循 ATM Forum UNI 3.0 和 UNI3.1 标准,并保证易于升级以适应 ATM UNI 标准的新发展。

3 ATM NIC 的结构分析

下图(图 2)说明了目前在 ATM NIC 的设计方案中采用的基本硬件结构模型及功能模块。

其中, SAR 控制模块主要负责 SAR 子层和 ATM 层协议的实现。PHY 模块完成物理层协议的处理。NIC 上的 Local SRAM 中存放协议处理过程中所使用的数据结构。在这里, 微处理器为可选部件, 主要用于网卡初始化, 连接管理, 错误恢复和实现 OAM(操作与维护)功能的基本控制。

目前, 在 ATM 芯片开发厂商和 ATM NIC 设计者的产品的开发过程中一直存在着两种不同的实现方案, 其主要的不同点可简单地概括为一句话, 即 ATM NIC 从何处存取需要传输的 CS-PDUs。一种方案是利用 NIC 上的本地存储器, 如图 3 所示。

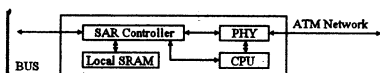


图 2 ATM NIC 的基本硬件结构及功能模块

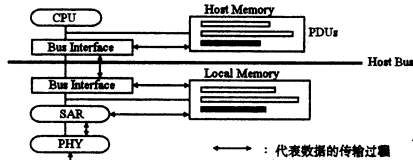


图 3 利用 ATM NIC 上的本地存储器的方案

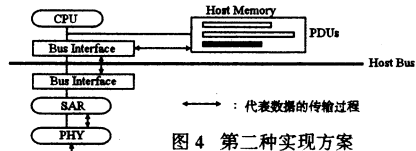


图 4 第二种实现方案

当开始数据传送时, 先由 SAR 控制器从主机存储器中拷贝完整的 CS-PDUs 到 NIC 上的本地存储器中, 然后对 CS-PDUs 进行分割和传送。接收时, 信元被放入 NIC 的本地存储器中进行重组, 然后由 SAR 控制器将重组好的 CS-PDUs 送到主机内存中。这个方案的优点之一是允许 NIC 以高于主机总线的速率进行传输。因此, 在主机总线有足够可利用带宽但延迟较大的情况下, 通过将整个 CS-PDUs 由主机内存拷贝到 NIC 本地存储器的方法可减少总线仲裁的开销。但是, 这种方案缺点也是显而易见的。首先, 由于在 NIC 上所使用的本地存储器空间毕竟有限, 所以 NIC 所能同时支持的 CS-PDUs 也不会太多, 同时, 这对于降低 NIC 产品的成本也是不利的。另外, 还有一个关键性的问题, 就是通过这种“接力”操作, 不但增加了数据传送的中间延迟时间, 而且在大块的 PDUs 从主机内存到 NIC 本地存储器的过程中会直接影响到主机系统中其它设备对总线的使用。

第二种方案(如图 4 所示)则回避了上述的这些问题, 在该方案中取消了 NIC 上存放 CS-PDUs 的本地存储器, 而直接在主机内存中进行 CS-PDUs 的分割

和重组。SAR 控制器通过 DMA 方式访问主机内存, 从而使用户数据可以直接经过物理接口和网络端进行交互。这样就避免了在 NIC 设计中花在本地存储器上的开销, 并保证了 NIC 能够同时支持大量的 PDUs, 同时也降低了传输延迟时间。其缺点则是 ATM 的突发性传输在传输速率上受到主机总线带宽的限制, 但鉴于目前业界已使用的一些高带宽总线如 PCI、S-BUS, 这一缺陷就显得不突出了。

4 ATM NIC 的工作原理

ATM NIC 是主机与网络间数据交换的接口, 下面分别就数据发送过程和数据接收过程对 ATM NIC 的工作原理加以剖析。

4.1 发送过程

(1) 首先, ATM NIC 驱动程序初始化与发送过程有关的寄存器 and 数据结构。

(2) 在数据发送之前, 驱动程序根据用户请求建立虚连接, 并在相应的数据结构中进行登记。

(3) 当需要传送的数据准备完毕, 驱动程序创建一个与之相关的的结构, 其中包含有数据块起始地址和长度指示、ATM 信元头参数及一些控制信息。

(4) NIC 上的 SAR 控制器在接到驱动程序的通知后, 根据所获得的数据块起始地址及长度信息从主机内存将数据块拷贝到 NIC 上的数据存储(缓冲)单元, 然后根据 CBR(恒定比特率), VBR(可变比特率), ABR(可利用比特率)的优先次序及相应算法, 按照信元的生成过程构成信元并传送给 PHY 部分。此后, 若 NIC SAR 控制器检测到数据块全部传送完毕, 可产生中断通知驱动程序进行下一步操作。

(5) PHY 部分中的 TC 处理模块接收到信元后, 生成信元头的 HEC 码并插入到 ATM 信元头中。然后, 将信元封装到物理层传输帧(STS-3C, STS-1 等)内, 由 PM 模块将其送到光缆或双绞线的物理介质上。

4.2 接收过程

(1) 首先, ATM NIC 驱动程序初始化与接收有关的寄存器和数据结构。

(2) 在接收数据之前, 驱动程序收到来自网络端的连接请求, 响应请求并调用相关的的结构。同时, 驱动程序为 NIC 提供可供使用的接收数据存储单元, 并将其首地址写入位于 NIC 本地存储器中的数据结构, 以供 NIC SAR 控制器使用。此时, NIC 进入数据接收状态。

(3) 物理传输帧由 PM 模块经过接收处理后送至 TC 处理模块, TC 模块从传输帧中取出信元进行差错检测后, 并经过 UTOPIA 接口到 NIC 上的数据存储(缓冲)单元。

(4) 这时, NIC 上的 SAR 控制器根据信元头中的 VPI/VCI 寻找相应的连接, 并读取预先分配的主机内

存的空闲存储单元首地址,然后,将信元中的用户数据通过主机总线传送到主机内存中。当一个完整的数据块传输完毕,SAR 控制器发出中断通知驱动程序进行处理。

(5)驱动程序对接收到的用户数据处理后提交给上层的应用程序,同时释放被该数据占用的资源,更新有关数据结构,以准备下一次的接收。

5 一种 ATM NIC 产品的介绍

笔者曾经作为主要设计人员参与了 155Mbps ATM NIC 的研制和开发工作,对 NIC 的硬件结构及功能实现均进行了一定深度的研究。我们所开发的 155Mbps ATM INC 主要作为工作站及高档微机的网络接口设备,可支持 AAL3/4,AAL5,AAL0 等业务类型,并同时支持 CBR,ABR,VBR 及 UBR 四种速率传输类型。该 NIC 基于 PCI 总线接口,在多模光纤上可进行 155Mbps 的全双工数据传输。其中主要的协议处理部件我们采用了 IDT 公司的 77201 SAR 控制器,以实现 SAR 子层和 ATM 层的协议功能。其结构原理如图 5 所示。

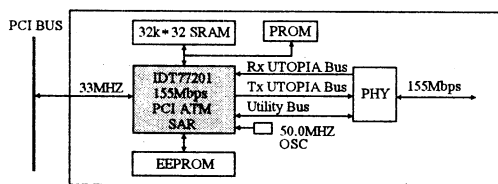


图 5 一种 ATM NIC 结构原理图

在本设计方案中,SAR 控制器部分和 PHY 部分是两个主要的协议处理单元。SAR 控制器主要完成 SAR 子层的协议处理功能,包括对 CS-PDU 的分割与重组,形成 AAL3/4,AAL5 等协议类型的信元,并保证 ATM NIC 与主机系统间数据的传输。另外,SAR 还实现 ATM 层的协议功能,如信元头的产生和提取、信元的复用和分路及一般的流量控制工作。PHY 芯片

则实现与 TC 子层和 PM 子层相关的一系列功能,其中,TC 子层主要完成传输帧的产生和恢复、HEC 码的产生及校验以及信元定界和信元速率匹配功能。PM 子层主要负责在物理媒体上正确发送和接收数据比特,完成线路编码、光电转换和比特定时等功能。

我们在进行 NIC 的设计过程中,根据预先对该产品设计指标及使用环境所作的规划,采用共享主机内存的方法,避免了在 NIC 上使用本地数据存储器,提高了数据传输速率,并降低了成本。同时我们将 NIC 上本地数据结构的存储器容量设为可选值,用户可根据实际的需要选用 32k*32 及 128k*32 两种存储器,从而保证了该 ATMNIC 最大可同时支持 16,000,000 条发送虚连接和 16,000 条接收虚连接。目前,从开发成果来看,本产品已具有推向市场的可能。

6 结束语

以上篇幅较详细地描述了 ATM NIC 的硬件结构和工作原理,并简单地介绍了笔者在 ATM NIC 开发过程中的一些实践经验。在 ATM 技术日渐成熟和完善的今天,许多知名的网络芯片及生产厂商已纷纷在 ATM 产品开发方面表现出了极大的投资热情,不少厂商如 Fore systems,Newbridge,TI,IDT,Brooktree 等公司都已分别有 ATM 芯片组,ATMNIC,ATM switch 的推出。面对 ATM 网络迅猛的发展势头,我国的网络业界人士将必须在平稳中求发展,在 ATM 技术的开发和应用中逐步积累经验,使我们能与国际的 ATM 技术发展保持同步。

参考文献

1. Tenning Chee, Delivering ATM to the Desktop: Practical Network Interface Card strategies, IIC'96 Conference Papers.
2. 国防科技大学学报, Vol. 18 Sup. 1996. 9
3. 程时端,刘斌译. 异步传输模式 宽带 ISDN 技术. 北京:人民邮电出版社,1995.

计算机在砼内缺超声探测中的应用

卢结成 吴龙标 王立敏 丁晓兵
(中国科学技术大学 合肥 230027)

摘要 根据超声波传播速度、首波幅度及接收波频率等参数与砼结构内部缺陷的相关关系,研究了一种微机检测系统。该系统包括数据采集、信号处理和缺陷判断三个主要部分,用于探测结构混凝土的内部缺陷。本文重点介绍了声时和首波幅度的快速算法、接收波形的频谱分析方法以及基于模糊数学原理和产生式规则的两种缺陷判断方法。

关键词 内部缺陷 隶属度 产生式规则

1. 引言

利用现代计算机应用技术,实现缺陷检测的智能化和定量化,已成为砼(混凝土)内部缺陷无损检测的重要内容之一。借助微型机的强有力处理功能,并配以相应的自动化、智能化分析软件,能够实现缺陷信息的自动提取、快速处理,最终获得检测结果。与人工无损检测方式相比,微机检测系统具有计算可靠、判断准确、处理的信息量大等明显优点。鉴于目前我国混凝土用量十分可观的情况,开展这一检测技术的研究,具有重要意义。

对于水泥构件或结构混凝土,由于其材料结构的复杂性,声波在其中传播时产生较大的散射和衰减,其内部缺陷的无损检测,通常采用超声脉冲法的对测方法,即将超声换能器的发射探头和接收探头分别置于被测物体的两个相对表面,利用接收波的声学参数变化特征来判别内部缺陷。当超声波通过内部缺陷传播时,接收波的传播速度降低,首波幅度减小,接收波频率变低等。本系统就是基于这一原理,在 486 微型机上开发的砼内缺陷检测系统,用于探测结构混凝土内部空洞和裂缝等缺陷。

2. 检测原理

砼内缺陷检测系统由硬件和软件两部分组成。硬件以 486 微型机为核心,包括数据采集和其它各种输入输出设备,并配以放大整形和延时触发电路以及超声脉冲源、超声换能器等必要的测试设备。软件采用模块化结构,由 windows 消息驱动核心控制下的多个子模块组成。这些子模块可分别完成以下功能:数据采集,信号处理,相关参数提取,缺陷的综合判断及检测结果的 CRT 显示与打印机输出等。本系统的工作原理框图如图 1 所示。

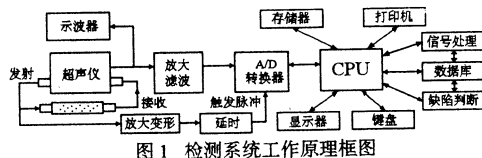


图1 检测系统工作原理框图

超声换能器的发射探头将电脉冲信号转换为超声波,此声波经被测介质传播之后由接收换能器接收下来,再经放大、滤波,送至 A/D 转换器的信号输入端。同时将发射脉冲整形延时,送至 A/D 板的同步触发信号输入端,以便立即采集到包括首波在内的前几个周期的超声接收信号。信号处理模块对当前采集的数据或存于磁盘文件的数据进行各种计算,包括采集数据的预处理,检测超声传播速度和接收波的首波幅度,计算信号的最大值、最小值及平均值,频谱分析,功率谱和相关分析等。缺陷判断模块采用了两种判断方法,即基于模糊数学原理的模糊判断和基于产生式规则的智能判断方法。必须指出,不论采用那种方法,都要分两步

进行,首先初步确定缺陷位置,然后在可疑的测试点附近加以细测,以准确测定缺陷的范围。

2.1 超声参数的检测方法

在本系统中,为了实时检测声速和首波幅度,设计了单向循环比较算法、双向循环比较算法、以及在屏幕上通过光标操作直接从采样波形上读取声速和首波幅度的方法。利用快速傅立叶变换技术,可以方便地获得频谱、功率谱、相关谱及接收波频率等参数。

2.1.1 声时和首波幅度的循环比较算法

这是一种自动计算声时、声速和首波幅度的快速检测方法,它分为单向循环比较算法和双向循环比较算法。单向循环法是从图 2(b)中的 A、B 点之间的任一点开始,向右循环比较采样码至 D 点,可先后检测到首波起点 B 和首波峰点 C,进而求得声时、声速和首波幅度。双向循环法是从图 2(b)中的 A、C 两点之间的任一点开始向右比较采样码,先求得首波峰点 C,然后从峰点向左循环比较,检测首波起点。循环比较法的工作流程图如图 3 所示,具体计算方法详见参考文献[7]。

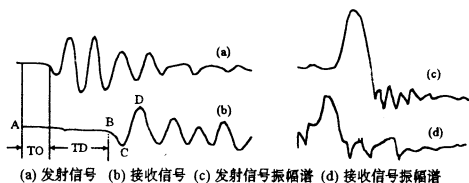


图2 发射、接收信号及其振幅谱

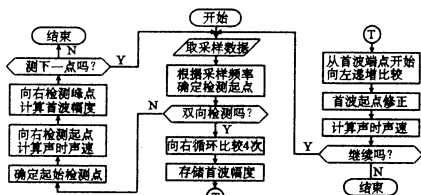


图3 声时(声速)及首波幅度检测流程图

2.1.2 移动光标读取声时、“首幅”和信号频率

软件中设置了两个光标 t_1 和 t_2 ,通过键盘操作,可在屏幕上使 t_1 、 t_2 沿着采样波形左右移动,并在波形上方设置了五个计数器,分别计数显示每个光标所在的采样点位置、对应的量化电平值以及两个光标的时间差。这样,只要将其中一个光标置于第一个采样点,另一个光标移到首波起点(或首波幅度)。如果将 t_1 、 t_2 分别置于波形上的两个波峰或波谷处,就可直接读取信号周期,从而按(1)式算出信号频率。

$$f = 1/r \times (t_2 - t_1) \times T. \quad (1)$$

式中, T 为采样周期, r 为比例常数,根据两个光标的位置选择相应的 r 值。如分别将 t_1 、 t_2 置于两个相邻的波峰或波谷,则 $r=1$;若 $t_2 - t_1$ 为信号的半个周期,则 $r=2$,等等。很明显,由于是两个时间相减,测试电路的延迟时间被抵消了,因而用(1)式能够准确地计

算信号频率。

2.1.3 谱分析及信号主频率的检测

为了全面考察超声波经不同介质传播时,接收波的频率变化及能量分布情况,我们设计了两种谱分析程序,一是频谱分析,求取振幅谱、相位谱和主频率;另一种是利用相关计算功率谱的估计值,得到相关函数、功率密度函数及信号主频率。其中频谱分析的步骤如下:

1)、以间隔 Δ 抽样得到有限长度离散信号示于(2)

式:

$$X(n\Delta), \quad n=0,1,\dots,N_0-1 \quad (2)$$

2)、从(2)式中截取 $N=2^k$ 项进行分析,记为

$$\bar{X}_n=X(n\Delta), n=0,1,\dots,N-1 \quad (3)$$

3)、对(3)式进行加窗处理得到

$$X_n=\bar{h}_n\bar{X}_n, \quad n=0,1,\dots,N-1 \quad (4)$$

4)、用FFT计算频谱

$$X_m=\sum_{n=0}^{N-1} X_n e^{-i2\pi mn/N} = U_m + iV_m \quad (5)$$

(0 < m < N-1)

5)、对振幅谱 $A_m = \sqrt{U_m^2 + V_m^2}$ 进行平滑处理

$$\bar{A}_m = P_m * A_m = \sum_{j=-2}^2 P_j A_{m-j} \quad (6)$$

6)、求振幅谱的主频率 f_M

$$\text{令 } f_M = mM/N\Delta \quad (0 \leq mM \leq N/2) \quad (7)$$

其中 mM 使

$$\bar{A}_{mM} = \max\{\bar{A}_m, m=0,1,\dots,N/2\} \quad (8)$$

由计算机很容易求出 mM ,因而可得到 f_M 。功率谱的计算方法与上述类似。最后可在屏幕和打印机上绘出谱分析的各种曲线,以此对超声接收波的各种频率成分进行分析。图2(a)、图2(b)对应的振幅谱分别示于图2(c)和图2(d)。

2.2 缺陷判断方法

分别研究了模糊综合判断方法和基于规则的专家系统判断方法。前者的突出优点是,综合考虑了各种检测参数与内部缺陷的相关关系,能够方便地把描述性的检测信息予以量化。后者更强调检测者的经验和专家的知识,达到了初步智能化的要求。

2.2.1 模糊综合判断方法

这种方法的基本原理是,根据模糊数学中隶属度的概念,利用声速 u_1 、首波幅度 u_2 、接收信号频率 u_3 及波形的相关系数 u_4 构成因子论域 U

$$U = \{u_1, u_2, u_3, u_4\} \quad (9)$$

在 U 上定义一个模糊子集 A ,其向量形式表示为

$$A = \{\mu_A(u_1), \mu_A(u_2), \mu_A(u_3), \mu_A(u_4)\} \quad (10)$$

式中 $\mu_A(u_i)$ 表示元素 u_i 对模糊子集 A 的隶属度,它反映了各个超声参数对于缺陷评价的相对贡献大小。

评定缺陷的结果只有两种:正常 V_1 与异常 V_2 ,由它们构成评语论域

$$V = \{v_1, v_2\} \quad (11)$$

在 V 上也可定义一个模糊子集 B

$$B = (\mu_B(V_1), \mu_B(V_2)) \quad (12)$$

式中 $\mu_B(V_1), \mu_B(V_2)$ 是评定的结果,分别表示正常与异常的隶属度,即分别对被测点正常与异常可能性大小的度量。然后由因子论域 U 与评语论域 V 构成一种模糊关系 R ,如(13)···式所示:

$$R = \begin{matrix} & \begin{matrix} v_1 & v_2 \end{matrix} \\ \begin{matrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{matrix} & \begin{bmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \\ r_{31} & r_{32} \\ r_{41} & r_{42} \end{bmatrix} \end{matrix} \quad (13)$$

式中矩阵的元素 r_{ij} 表示单独考虑因子 u_i 时,测点混凝土质量状况对 V_j 的隶属度,此即单参数评判的结果,而(12)式中的 $\mu_B(V_i)$ 则为多参数综合评判的结果。

由某测点的上述四种声学参数,每种参数的平均值及其均方差,就可求出矩阵 R 。对于 A ,我们取 $A = (0.45, 0.3, 0.15, 0.1)$ 。已知 A 和 R ,就可由(14)式计算 B

$$B = A \cdot R \quad (14)$$

用上述方法求出每个测点的评判结果,就可了解被测区域的混凝土内部是否有缺陷,以及缺陷涉及的范围与程度。具体计算过程参考了文献[5]。

2.2.2 基于规则的缺陷判断方法

规则判断系统的基本结构如图4所示。这是一种基于规则表示的专家系统方法,它是利用产生式规则对内部缺陷进行综合评判。在用规则表示的产生式系统中,各种检测信息和专家知识均用“IF...THEN...”的形式表示,并存放在规则库中,例如:

如果 被测点的接收波的声速低于其平均值与二倍标准差之差

则 该测点内部可能有缺陷

如果 沿某测试方向的首波幅度均小于其平均值与偏离值之差

则 沿该方向可能是一条裂缝

等等。

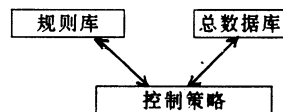


图4 规则判断系统的基本结构

总数据库有时也被称作上下文,或当前数据库,或暂时存储器。它主要用来存放每个产生式左边表示的条件(事实或断言)。控制策略的作用是说明如何使用规则。为了便于建立和使用规则,我们在程序设计中将规则分为两类,一类为信号处理规则,另一类是缺陷判断规则。建立信号处理规则是为了产生中间事实,即为缺陷判断提供必要的事实根据。缺陷判断规则用以直

接判断内部缺陷的位置,通过将规则的条件部分与相关参数进行匹配,得出被测点内部是否有缺陷的结论,肯定回答时,在被测试件的立体图形上的相应位置标以“+”号,否则用“-”号表示之。

3. 检测结果

用本系统对多种混凝土试块进行了测试研究,检测结果与普通超声仪测试的结果基本一致。现以边长 20 厘米的立方试块为例,分别测试密实试块和表面开口并向内部延伸的裂缝试块。换能器频率为 100KHz,用黄油为耦合剂,采样频率选择 5MHz。图 5 示为对密实试块用对测法检测的结果,图中 16 个测试点均为“-”号,说明该试块内部不存在缺陷;裂缝试块用模糊评判法检测的结果示于图 6,图中白影区域表示缺陷涉及的范围。图 7 表示用规则判断方法检测的结果。

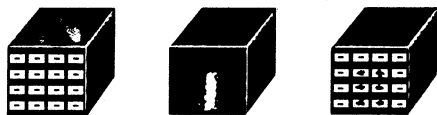


图5 密实试块 图6 模糊法检测结果 图7 规则法检测结果

应该指出,结构混凝土或水泥制品是由多种材料构成的具有弹一粘一塑性的复杂凝聚体,声波在其中传播时产生较大的散射与衰减,换能器的选择、耦合条件的好坏以及测试距离都会影响检测结果。因此使用本系统进行实际检测时应注意以下几个问题:

- (1)数据采集时,采集和存储的接收波数据应包括首波在内的前几个周期;

- (2)发射和接收超声波的两只换能器应与被测构件表面保持良好的耦合。各测点的耦合程度尽可能一致;

- (3)信号处理所截取的数据长度,至少应包含首波在内的前两个周期的接收波数据。

4. 结论

- (1)用循环比较采样码的算法和移动光标在采样波形上寻找首波起点的方法,都可快速准确地检测声速、声速和首波幅度。

- (2)利用声速、首波幅度、接收波频率及波形的相关系数,并结合模糊评判方法和规则判断方法,能够有效地综合判别砼结构内部缺陷。

- (3)要提高系统检测的准确性,首先必须保证微机采集到的各测点波形都能反映各点内部介质的真实情况。这主要取决于超声源和换能器的性能。

参考文献

- 1 David P. Roller, Joseph L. Rose and Michael J. Saparito, British Journal of NDT, Nov. (1984), 416—419
- 2 Chung, H. W. Concrete International, Designa Construction, 5—10(1983), 42—49
- 3 林维正. 无损检测, 6(1981), 29—31
- 4 程乾生. 信号数字处理的数学原理, PP. 167—173
- 5 卢明良. 无损检测, 11(1991), 301—303
- 6 傅京孙. 蔡自兴, 许光佑. 人工智能及其应用, PP. 277—292
- 7 卢结成. 戴英侠等. 中国科技大学学报, 3(1992), 40—43

利用消息实现 Win32 进程间数据传递的方法

张丹宇

(北京航天指挥控制中心软件室 100101)

摘要 本文说明了如何利用 Win32 的 WM_COPYDATA 消息在进程间传递数据,并提供了定义 MFC 消息映射宏的方法。

关键词 WM_COPYDATA SendMessage 消息映射宏

Windows 环境具有在进程间快速简便地共享数据和信息的能力,在 16 位 Windows 中,最常见的方法是调用 SendMessage 或 PostMessage,由于 16 位 Windows 中 SendMessage 和 PostMessage 只允许给另一个进程传递一个 16 位值和一个 32 位值,因此常分配一块全局存储(使用 GETMEMSHARE 标志)然后在调用 SendMessage 和 PostMessage 时传递该句柄(作为 wParam 或 lParam 参数),然后这条消息的接受者调用 GlobalLock 来取到该存储块的一个地址,并读

取数据,从而实现数据的传输。但在 Win32(32 位应用程序编程接口,Windows NT 和 Windows95 是两个支持 Win32 的操作系统)中行不通,因为每个进程有它自己的地址空间,一个进程不能够轻易地得到其它进程的地址空间中的数据。那 Win32 怎样使用窗口消息在进程间传递数据呢?

1. 实现方法

首先,我们考虑以下的调用:

```
SendMessage (FindWindow (NULL, "Clock"),
WM_SETTEXT, 0, (LPARAM)"Nw Title");
```

它确定了 Clock 窗口的窗口句柄,并打算将它的

标题变为 "New Title"。仔细分析一下,新标题串是在你的进程的地址空间中,所以你的地址空间中这个标题串的地址将作为 lParam 参数来传递,当 Clock 窗口的窗口进程收到这个消息后,它看一下 lParam 参数并试图将其作为一个字符串来操纵它,以便使它成为新的标题。但 lParam 中的地址指向了你的地址空间的一个字符串,而不是 Clock 地址空间中的一个字符串,因此存储违例肯定要发生,但它却能成功的运行!

原来 Win32 系统对在进程间传递的消息要进行特别的检查,如果其 wParam 和 lParam 参数代表了指向一个数据结构的指针,则将对这个消息进行特别的处理。对上面的调用,当 SendMessage 发现正在发送 WM_SETTEXT 消息时,将标题串从你的进程的地址空间复制到一块准备与其它进程共享的存储中,然后再发送该消息。当接受进程处理 WM_SETTEXT 消息时,它在自己的地址空间中确定共享存储块(含有新标题串的拷贝)的位置, lParam 参数被初始化为指向这个地址,且 WM_SETTEXT 消息被分配给相应的窗口过程,至此完成整个处理过程。

然而,当发送用户自定义的 WM_USER+x 消息(而不是系统知道的消息)时,系统并不知道你想分配一块共享存储并在发送消息时更新指针,此时可利用 Win32 新增的 WM_COPYDATA 消息,使用以下的调用:

```
SendMessage(hwndReceiver, WM_COPYDATA,
(WPARAM)hwndSender, (LPARAM)&cds);
cds 为 COPYDATASTRUCT 结构,其定义为:
typedef struct tagCOPYDATASTRUCT{
    DWORD dwData;
    DWORD cbData;
    PVOID lpData;
}COPYDATASTRUCT;
```

当发送数据给另一个进程的窗口时,首先要初始化 COPYDATASTRUCT 结构, dwData 是保留参数,可用它来标识发送数据的内容, cbData 指定发送数据的字节数, lpData 指向所发送的数据。

当 SendMessage 发现在发送 WM_COPYDATA 消息时,它分配一块大小为 cbData 字节的公共存储,并从你的地址空间拷贝数据,然后将此消息发送到目的进程的窗口。

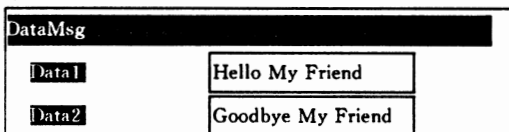
接受窗口过程处理此消息时,其 lParam 参数指向一个存在于接受进程地址空间中的 COPYDATASTRUCT 结构,这个结构的 lpData 指向公共存储块,并且地址也已改变成反映在接受进程地址空间中的公共存储的位置。

需注意的是发送 WM_COPYDATA 消息必须使用 SendMessage,而不能用 PostMessage,因为系统必须在接受窗口处理完消息后释放公共存储块,否则

系统不知道何时 WM_COPYDATA 消息被处理完,因而不能释放公共存储块。

2. 程序示例

这里利用 Visual C++ 4.0 的 MFC 类库编写了一个小程序 DATMSG.EXE,说明了如何利用 WM_COPYDATA 消息发送一块数据到另一个应用程序,程序在 Windows NT3.51 和 Windows95 上调试通过。它基于对话框,需至少运行两个 DATMSG 的实例,在其中的一个实例的编辑框中输入数据,点击相应的按钮将其发送到所有正在运行的 DATMSG 实例中去,每个实例更新自己的编辑框中的内容来反映新的数据。



这里要指出的是 MFC 使用消息映射机制来响应特定的消息,在消息映射中的每一个入口都将一个特定的消息对应到特定的成员函数,这就是所谓的消息映射宏。MFC 中包含了大量的预定义消息映射宏,也许由于 WM_COPYDATA 是 Win32 新增的消息,在 MFC 中没有它的消息映射宏,需要自己来定义,其方法如下:

```
#define ON_WM_COPYDATA
{WM_COPYDATA, 0, 0, 0, AxCfg_lwl, \
(AFX_PMSG)(AFX_PMSGW) (LRESULT)(AFX_ \
MSG_CALLCWnd::*) \
(WPARAM, LPARAM))OnCopyData},
ON_WM_COPYDATA()宏指定了 WM_ \
COPYDATA 消息对应的成员函数为 LRESULT On_ \
CopyData(wParam, lParam),接收数据的实例收到 \
WM_COPYDATA 消息时将调用 OnCopyData 函 \
数,其参数 wParam 保存了发送数据的窗口的句柄, \
lParam 则指向了 COPYDATASTRUCT 结构,从中可 \
以提取从另一进程传来的数据。
```

程序主要包括以下的处理:

- 初始化 COPYDATASTRUCT。如点击 Data1 按钮,则 COPYDATASTRUCT 的 dwData 置为 IDC_SENDBUTTON1(=1000),如点击 Data2 按钮,则 dwData 置为 IDC_SENDBUTTON2(=1001);根据数据的长度分配一块存储块,并使 COPYDATASTRUCT 的 lpData 指向它,然后将编辑框中的数据拷贝到该存储块中。

- 循环经过桌面上的所有窗口,向所有标题为 DataMsg 的窗口发送 WM_COPYDATA 消息。

- 释放 COPYDATASTRUCT 的 lpData 指向的存储块。

• 接收 WM_COPYDATA 消息的窗口调用 OnCopyData 函数,用接收到的数据更新自己编辑框中的内容。

以下是主程序清单:

```
#include<afxwin.h> //MFC core and standard components
#include<afxext.h> //MFC extensions
#ifdef _AFX_NO_AFXCMN_SUPPORT
#include<afxcmn.h> //MFC support for Windows 95 Common Controls
#endif // _AFX_NO_AFXCMN_SUPPORT

#define ON_WM_COPYDATA(\
    (wm_COPYDATA,0,0,0,AfxSig_lwl,\
    (AFX_PMSG)(AFX_PMSGW) (LRESULT (AFX_MSG_CALL-\
CWnd:: *
    (WPARAM,LPARAM))OnCopyData),
    class CDatmsgApp,public CWinApp
{
public:CDatmsgApp();
public:virtual BOOL InitInstance();
}
class CDatmsgDlg:public CDialog
{
public:
    CDatmsgDlg(CWnd* pParent=NULL);/*standard constructor
    enum (IDD=IDD_DATMSG_DIALOG);
protected:
    virtual void DoDataExchange(CDataExchange* pDX);/*DDX/DDV support
protected:
    HICON m_hIcon;
    //消息映射函数
    virtual BOOL OnInitDialog();
    afx_msg LRESULT OnCopyData(WPARAM,LPARAM);
    afx_msg void OnSendbutton2();
    afx_msg void OnSendbutton1();
    afx_msg void OnsysCommand(UINT nID,LPARAM lParam);
    afx_msg void OnPaint();
    DECLARE_MESSAGE_MAP()
};
//////////
CDatmsgApp theApp;
BOOL CDatmsgApp::InitInstance()
{
    CDatmsgDlg dlg;
    m_pMainWnd=&dlg;
    dlg.DoModal();
    return FALSE;
}
CDatmsgDlg::CDatmsgDlg(CWnd* pParent/*=NULL*/):
CDialog(CDatmsgDlg::IDD,pParent)
{
    m_hIcon=AfxGetApp()->LoadIcon(IDR_MAINFRAME);
}
void CDatmsgDlg::DoDataExchange(CDataExchange* pDX)
{
    CDialog::DoDataExchange(pDX);
}
//消息映射
BEGIN_MESSAGE_MAP(CDatmsgDlg,CDialog)
    ON_WM_SYSCOMMAND()
    ON_WM_PAINT()
```

```
ON_BN_CLICKED(IDC_SENDBUTTON2,OnSendbutton2)
ON_BN_CLICKED(IDC_SENDBUTTON1,OnSendbutton1)
ON_WM_COPYDATA()
END_MESSAGE_MAP()
LRESULT CDatmsgDlg::OnCopyData(WPARAMhWndsender,LPARAM cds)
{
    GetDlgItem(((PCOPYDATASTRUCT)cds->dwData->
    SetWindowText(LPSTR)(((PCOPYDATASTRUCT
    )cds->lpData));
    return TRUE;
}
void CDatmsgDlg::OnSendbutton1()
{
    CString m_data,m_caption;
    COPYDATASTRUCT cds;
    CWnd* pWndsibling;
    //初始化 COPYDATASTRUCT 结构
    GetDlgItem(IDC_SENDEDIT1->GetWindowText(m_data);
    cds.dwData=IDC_SENDEDIT1;
    cds.cbData=m_data.GetLength()+1;
    cds.lpData=HeapAlloc(GetProcessHeap(),HEAP_ZERO_MEMORY,cds.cb-
    Data);
    memcpy(cds.lpData,m_data.GetBuffer(m_data.GetLength()),cds.cbData);
    m_data.ReleaseBuffer();
    pWndsibling=GetWindow(GW_HWNDFIRST);/*定位第一个应用窗口
    GetWindowText(m_data);/*获取当前窗口的标题
    while(pWndsibling)
    {pWndsibling->GetWindowText(m_caption);
        if(m_data==m_caption) //窗口标题是否相同
            ::SendMessage(pWndsibling->m_hWnd,WM_COPYDATA,
                (WPARAM)(this->m_hWnd),(LPARAM)&cds);
        pWndsibling=pWndsibling->GetWindow(GW_HWNDNEXT);
    }
    HeapFree(GetProcessHeap(),0,cds.lpData);
}
void CDatmsgDlg::OnSendbutton2()
{
    CString m_data,m_caption;
    COPYDATASTRUCT cds;
    CWnd* pWndsibling;
    GetDlgItem(IDC_SENDEDIT2->GetWindowText(m_data);
    cds.dwData=IDC_SENDEDIT2;
    cds.cbData=m_data.GetLength()+1;
    cds.lpData=HeapAlloc(GetProcessHeap(),HEAP_ZERO_MEMORY,cds.cb-
    Data);
    m_data.ReleaseBuffer();
    pWndsibling=GetWindow(GW_HWNDFIRST);
    GetWindowText(m_data);
    while(pWndsibling)
    {
        pWndsibling->GetWindowText(m_caption);
        if(m_data==m_caption)
            ::SendMessage(pWndsibling->m_hWnd,WM_COPYDATA,
                (WPARAM)(this->m_hWnd),(LPARAM)&cds);
        pWndsibling=pWndsibling->GetWindow(GW_HWNDNEXT);
    }
    HeapFree(GetProcessHeap(),0,cds.lpData);
}
BOOL CDatmsgDlg::OnInitDialog()
{
    CHialog::OnInitDialog();
    SetIcon(m_hIcon,TRUE); //Set big icon
```

```

SetIcon(m_hIcon, FALSE); //Set small icon
//Initialize the Edit Control with Data
GetDlgItem(IDC_SENDEDIT1)->SetWindowText("Hello My Friend");
GetDlgItem(IDC_SENDEDIT2)->SetWindowText("Goodbye My Friend");
return TRUE;
}

void CDatmsgDlg::OnSysCommand(UINT, nID, lParam1Param)
{
    CDialog::OnSysCommand(nID, lParam1Param);
}

void CDatmsgDlg::OnPaint()
{
    if(IsIconic())
    {
        CpaintDC dc(this); //device context for painting
        SendMessage(WM_ICONERASEBKGD, (WPARAM)dc, GetSafeHdc(), 0);
        //Center icon in client rectangle
    }
}

```

```

int cxIcon=GetSystemMetrics(SM_CXICON);
int cyIcon=GetSystemMetrics(SM_CYICON);
CRect rect;
GetClientRect(&rect);
intx=(rect.Width()-cxIcon+1)/2;
inty=(rect.Height()-cyIcon+1)/2;
//Draw the icon
dc.DrawIcon(x,y,m_hIcon);
}
else
    CDialog::OnPaint();
}

```

参考文献

1. DAVID J. KRUGLINSKI · Visual C++ 技术内幕, 清华大学出版社, 1994. 11
2. Visual C++ 2.0 for Win32 大全, 清华大学出版社, 1996

图库——CAD 成图率上升的途径之一

刘希宏

(甘肃兰州铁道部第一勘测设计院桥隧处 730000)

目前, CAD 技术在设计领域中得到了广泛应用, 设计单位也从中受益非浅。主要是减轻了手工绘图的工作量, 提高了工效。而且设计图纸质量高, 设计周期短, 能较好地适应激烈的设计市场。

但是, 在 CAD 技术应用的后期, CAD 成图率上升很慢甚至徘徊不前。究其原因, 是因为相当一部分设计图纸无法用较快的图纸生成程序完成, 还得借助人机交互。这样便大大延长了设计时间。针对这种情况, 考虑到图纸的特殊性, 我们提出了“图库”这个概念, 根据设计人员的项目不固定, 有时间性, 单位有多个设计实体的情况, 我们制定了“收集旧图, 便于交换, 集中管理, 便于查询”的原则。依据这个原则, 我们开发了“图纸数据库管理系统”, 并在生产中加以应用。

了解国内工程的人知道, 国内某单位有一套庞大的工程图纸系统, 带有扫描消蓝等功能。但对普通单位来说, 投资巨大。因此我们根据目前管理水平不算太高、交流不便的情况, 建议使用“图库”的概念。使用“图库”我们制定了配套的规定, 如〈图形数据入库规定〉、〈图库使用规定〉, 只有遵循这些规定, 它的作用才能逐步发挥出来。

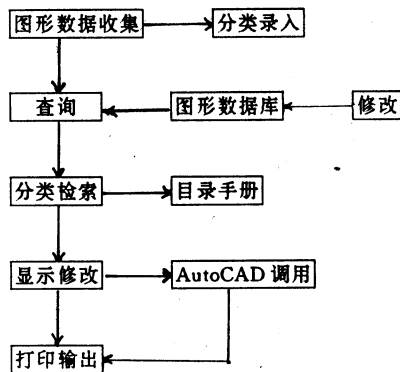
图库软件的开发要以本单位的实际情况为基础。我们选定的硬软件环境为:

硬件: 80486 以上微机, 4M 以上内存, 500M 硬盘或有光驱动器。

软件: MSDOS 3.0 以上, UC DOS 3.0, AutoCAD 12.0 FoxPro 2.5

这里要说明的是光驱动器主要是做为图形备份所

用, 详细的结构流程图如下:



软件开发是密集性的工作, 对于设计单位, 应紧密结合生产实际, 目标要定得高, 不断完善, 才能适应生产需要, 不致造成生存期太短的弊端。基于这种考虑, 除可完成设计要求外, 力求为用户提供一个友好的界面, 设置了汉化下拉式菜单, 热键帮助, 为用户提供了一个友好的环境。程序全部采用模块化, 面向对象编程, 采用按钮与鼠标移动, 大量采用弹式菜单, 多重菜单, 根据库内容动态查询, 使用户只需根据专业情况就可使用, 整个人机对话只需输入少量汉字就可工作。为具有专业的设计人员提供了有力的帮助, 并使徘徊不前的 CAD 成图率有了明显提高。

考虑到 .DWG 的直接显示和修改, 本程序预留了接口, 只要在当前环境屏蔽 ACAD 即可。

收稿日期: 1997 年 4 月 8 日

DBF 文件到电子表格的简捷转换

何其才 杜建丽

(解放军第二十九医院 天水 741020)

摘 要 本文根据 DBF 数据库的结构,将其记录的数据信息,按用户要求生成表格文件,在文本编辑软件支持下方便输出。

关键词 DBF 数据库 电子表格 转换

1 引言

以 DBF 数据文件的格式存储数据库数据的计算机管理信息系统,在多个领域中建立并运行,不断完善,现已成为进行各项工作必不可少的有力助手。系统中的打印输出模块,都是事先设计好的固定格式,在实际工作中,所需的随机性报表,不能打印输出。如我院应用的《军队医疗信息管理系统》中的 DBF 数据库文件记录着住院病人的基本医疗情况,系统设有打印输出统计指标、质量指标、住院消耗费用等功能,1996 年 11 月上级卫生部门要求汇总上报 1995 年 10 月至 1996 年 10 月住院费用在 5000 元以上的军队伤病员情况表,数据库中有所需数据,但因打印模块中没有设计这种格式,而不能自动输出。

能否设计一个按要求从数据库中生成任一结构表格的方法。CCED 等电子表格软件有从数据库输出报表的功能,但格式说明复杂,掌握较困难。本文介绍的这种方法,可简单、方便、直观的从 DBF 数据库中输出任意结构的表格文件,在文本编辑软件的支持下打印输出。

2 构思与方法

2.1 DBF 数据库文件由三部分组成:库结构说明(见表 1),字段说明(见表 2)和记录内容。

文件结构说明 表 1

位置	长度	内容
0	1 字节	最低两位为版本号
1—3	3 字节	建库年月日
4—7	4 字节	记录个数
8—9	2 字节	结构说明和字段说明, 共占的字节数+1
10—11	2 字节	记录字节个数
12—31	20 字节	保留

字段说明 表 2

位置	长度	内容
0—10	11 字节	字段名 ASCII 码
11	1 字节	字段类型
12—15	4 字节	字段在记录中的位置
16	1 字节	字段长度的 2 进制数
17	1 字节	小数点后位数的 2 进制
18—19	2 字节	保留
20	1 字节	工作区标志 ID
21—31	11 字节	保留

获得?我们把结构说明和字段说明建立成与它们对应的两个结构类型数据 dbs 和 record。根据结构类型数据可计算出任意条记录 num 的位置: $dbs_stru_byte_num + (num - 1) * dbs_record_byte_num$ 。用户以菜单方式选择要转换的字段项,修改字段名,将其输入的数据存入结构数组 orecord(定义同 record)。从记录中提取字段项的内容,主要解决两问题:一是字段名与字段内容长度不一致时,大者首选;二是字段名或内容的长度为奇数时,长度加 1。其次对要插入的空字段项,可在 orecord.field__type 处设标识符进行处理。

2.2 转换流程如图 1 所示。

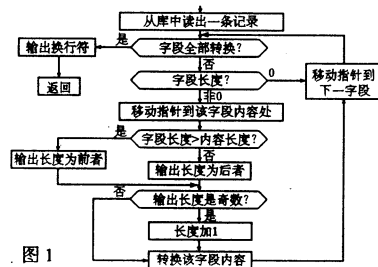


图 1

转换的信息是数据库中的记录内容,怎样准确的

本文用 C 语言实现了这一功能。源程序在 IBM-286、兼容-386、GW-486 等机器上用 Turbo C V2.0 编译、运行通过、功能源程序如下:

/* 两结构类型与文件结构说明、字段说明对应 */

```

typedef struct dbfl
{
    char ver;
    char date[3];
    unsigned long record_num;
    unsigned int stru_byte_num;
    unsigned int record_byte_num;
}DBF1;
typedef struct dbf2
{
    unsigned char field_name[11];
    char field_type;
    unsigned long unsel;
    char field_len;
    char decimal;
    char unse2[2];
    char work_area;
    char unse3[11];
}DBF2;
/* 分页转换数据库全部记录函数 */
void chg()
{int i,j,n=0;
    char tj[11],*p1,*p2;
    for(i=0;i<((dbs.stru_byte_num-33)/32);i++)
        if(odbre[i].field_len!=0+n)
            if(n==0)return;
        goto _xy(13,33);
    printf("数据库共有%d条记录",dbs.record_num);
    goto _xy(15,33);
    printf("请输入报表每页行数:");
    scanf("%d",&n);
    fseek(fp,dbs.stru_byte_num,SEEK_SET);
    getchar();
    prtfh();
    for(i=0,j=1,i<dbs.record_num;i++,j++)
        {prthx(i);plr(i);if(j>=n){j=0;prtwx();fputc('\n',op);prtfh(i);}}
    prtwx();/* prtfh()输出表头;prtwx()输出表格线;
    prtwx()输出表尾线 */
}
/* 转换一条记录的函数 */
void plr(long num)
{int i,j,n,m;
    char *t,*tt;
    t=malloc(dbs.record_byte_num);
    tt=t;
    fseek(fp,num*dbs.record_byte_num+dbs.stru_byte_num,0);
    fread(t,dbs.record_byte_num,1,fp);
    fputs("|",op);
    for(j=0;j<((dbs.stru_byte_num-33)/32);j++)
        if(odbre[j].field_len!=0)
            if(odbre[j].field_type!='-')
                {
                    t=tt+odbre[j].unsel;
                    n=odbre[j].field_len;
                    m=strlen(odbre[j].field_name);
                    if(n>=m){fwrite(t,n,1,op);if(n%2!=0)fputc(' ',op);}
                }
            else
                {fwrite(t,n,1,op);
                    for(i=0;i<(m-n);i++)fputc(' ',op);
                    if(m%2!=0)fputc(' ',op);
                }
            fputs("|",op);
        }
    else
        {
            n=odbre[j].field_len;
            m=strlen(odbre[j].field_name);
            if(n>=m){if(n%2!=0)+n;for(i=0;i<n;i++)fputc(' ',op);}
            else{if(m%2!=0)+m;for(i=0;i<m;i++)fputc(' ',op);}
            fputs("|",op);
        }
    fseek(op,-2L,SEEK_CUR);
    fputs("|",op);
    fputc('\n',op);
    free(tt);
}

```

语音交流电压表的设计

田 辉

(郑州轻工业学院计算机系 450002)

摘 要 介绍一种以单片机为核心的智能语音报数交流电压表的硬件电路和软件设计。

关键词 单片机 语音 电压测量

1 引言

在供电和用电部门,必须保证正常的工作电压,因此需要对供电电压进行动态监测和记录。我们以单片机为核心,研制了智能语音报数交流电压表,以适应工农业生产中的一些特殊环境条件的使用。

2 硬件电路

交流电压表电路框图如图1所示。



图1 语音报数电压表框图

被监测交流电压经过有效值/直流转换电路,转换为直流电压,再通过A/D转换为数字信号送入单片机,单片机控制语音报告当前电压值。若此表用于监测供电电压使之保持在某一电压值,当出现欠压或超

此届展示研讨会是由中国 HP 公司、北京建银电脑公司主办、中国建行四川省分行、成都分行协办的。福建实达电脑公司、河北聚星终端公司、湖南计算机厂、西安星海电子科技公司、电子部 52 所等单位也在会上展示了相关产品,深受客户青睐。

SURFER 软件二进制绘图命令及应用

申时新 张志兵 谭昌勇

(成都理工学院信息工程与地球物理系 610059)

申时凯 (昭通师范专科学校)

摘要 本文介绍 SURFER 软件的二进制绘图命令, 给出利用 C 语言输出绘图命令组的实用子函数和应用实例。

1 引言

SURFER 软件是美国 GOLDEN 公司的二维、三维高级绘图软件包, 不但可直接绘制各种二维、三维图形, 还可以该软件包为基础开发二次绘图软件。开发二次绘图软件的基本方法即是利用 SURFER 的绘图命令。SURFER 本身和不少文章都介绍了它的文本绘图命令, 但文本方式命令所占的磁盘空间大、执行效率低。我们知道 SURFER 软件输出的图形文件. PLT 除文本方式外还可有二进制方式, 而二进制方式所占用的磁盘空间小, 因此若能知道 SURFER 的二进制绘图命令, 则对减少命令占用空间、提高显示和图形打印效率具有很实用的意义。本文作者对此作了研究, 得到了 SURFER 软件的二进制绘图命令, 并用 C 语言编制了一些绘图实用子函数, 我们把它们介绍出来, 以供有兴趣的作者应用, 作者还给出了一个应用实例。

2 SURFER 的二进制绘图命令

为能研究出 SURFER 的二进制绘图命令, 作者启动 TOPO, 将一网格化文件先形成一文本图形文件, 然后仅改变输出文件格式(不改变参数设置)以另一图形文件存盘, 将这一文件用 DEBUG 逐一查看 ASCII 码, 与文本图形文件进行对比即得到了二进制绘图命令形式, 作为比较, 我们将两种绘图命令一并列入表一。

表一: SURFER 的绘图命令

功能	文本命令	二进制命令 ASCII 码
绝对移笔	MA	2
相对移笔	PR	9
绝对落笔	PA	1
相对落笔	MR	8
选笔	SP	7
写字符串	SS	6

3 生成 SURFER 二进制绘图命令组的一些子函数

下面给出用 Turbo C 输出 SURFER 绘图命令组的常用子函数。

```
(1)选笔函数 setcolor(pen)
#define SP 7
void setcolor(int k)
{
    fprintf(ff1, "%c%c", SP, k);
}
```

```
(2)绝对移笔函数 moveto(x,y)
#define MA 2
void moveto(float x, float y)
{
    x=x/2.54;
    y=y/2.54;
    fprintf(ff1, "%c", MA);
    fwrite(&x, sizeof(float), 1, ff1);
    fwrite(&y, sizeof(float), 1, ff1);
}
```

```
(3)绝对落笔函数 lineto(x,y)
#define PA 1
void lineto(float x, float y)
{
    x=x/2.54;
    y=y/2.54;
    fprintf(ff1, "%c", PA);
    fwrite(&x, sizeof(float), 1, ff1);
    fwrite(&y, sizeof(float), 1, ff1);
}
```

```
(4)画线函数 line(x0,y0,x1,y1)
void line(float x0, float y0, float x1, float y1)
{
    moveto(x0, y0);
    lineto(x1, y1);
}
```

4 绘图实例

以下是作者利用以上绘图函数编制的绘制地震剖面的程序, 利用该程序我们绘制了某区的一条地震剖面(见图1), 其输出绘图命令组文件长度仅 4,934,687 字节, 比文本方式绘图输出文件的 12,170,364 字节少得多, 几乎仅为文本方式的 1/3。同时, 绘图显式和输出之执行速度也明显比文本方式快。

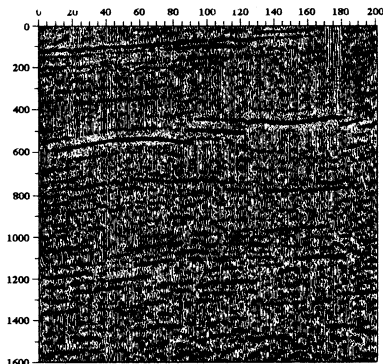


图1 某测线地震剖面图

```
#include<stdio.h>
#define MA2
#define PA1
#define PS4
#define SS6
#define SP7
#define MR8
#define PR9
#include "ptlibb.c"
#include<conio.h>
#include<math.h>
#include<stdio.h>
main()
{
FILE *in, *in1, *in2, *out, *out1; static float ht=0.2, an=0;
long int s, s1, s2, ss1, ss2, ss, ss4, head, sd, tin, pin;
float x10, cmax, ampu[255], bmax[1000], dltl, dltr, dltu, dltid, yoo,
yb, ye, x01, ssc;
float a, ch, ch2, chp, xor, yor, cmin=0.0, dit1, dit2, x11, y11, x12,
y12;
char key, xs[61], ys[61], *argv[3]={"seismicsfile", "dte2.par",
"ssx.plt"};
char *ssss;
int i=0, is, il, mmm, symbol[256], fast;
int color[3]; float ii, iil, x1, y1; int iix;
float dltx, dity, scale1, scale2, scale3, xmin, xmax, ymin, ymax, y, y01,
sjm0;
int ldx, ldy, ldx1, ldy1, lxmin, lymin, lxmax, lymin, lxb, lyb, x1xb,
y1xb, d1xb, d1yb;
textattr(YELLOW+(BLUE<<4));
clrscr();
window(15,30,21,60);
gotoxy(3,5);
printf("Enter parameter filename");
scanf("%s", argv[1]);
fflush(stdin);
if((in=fopen(argv[1], "r"))==NULL){
printf("cannot open source file %s\n", argv[1]);
exit(1);
}
textattr(YELLOW+(BLUE<<4));
clrscr();
window(15,30,21,60);
gotoxy(3,5);
printf("Enter fast,");
scanf("%d", &fast);
fflush(stdin);
clrscr();
window(10,30,21,60);
gotoxy(3,5);
printf("Enter seismics filename,");
scanf("%s", argv[0]);
fflush(stdin);
if((in2=fopen(argv[0], "rb"))==NULL){
printf("cannot open source file %s\n", argv[0]);
exit(1);
}
clrscr();
window(10,30,21,60);
gotoxy(3,5);
printf("Enter plot filename,");
scanf("%s", argv[2]);
init(argv[2]);
```

```
fflush(stdin);
clrscr();
window(10,30,21,60);
gotoxy(3,5);
printf("Trace title,");
gets(xs);
fflush(stdin);
clrscr();
window(10,30,21,60);
gotoxy(3,5);
printf("Point title,");
gets(ys);
clrscr();
fflush(stdin);
window(10,30,21,60);
gotoxy(3,5);
printf("Enter key(Y or N);");
key=getch();
for(il=0; il<3; il++){
clrscr();
window(10,30,21,60);
gotoxy(3,5);
printf("color[%d],", il);
scanf("%d", &color[il]);
scanf(in, "Minimum trace, %ld", &s1);
scanf(in, "\nMaximum trace, %ld", &s2);
scanf(in, "\nTrace to Trace distance, %f", &dltx);
scanf(in, "\nTrace increment, %ld", &tin);
scanf(in, "\nMinimum point, %ld", &ss1);
scanf(in, "\nMaximum point, %ld", &ss2);
scanf(in, "\nPoint to point distance, %f", &dity);
scanf(in, "\nPoint increment, %ld", &pin);
scanf(in, "\nTotal point of per trance, %ld", &ss4);
scanf(in, "\nTrace—head lenth (Number of byte), %ld",
&head);
scanf(in, "\nScale of trace, %f", &scale1);
scanf(in, "\nScale of point, %f", &scale2);
scanf(in, "\nScale of amplitude, %f", &scale3);
scanf(in, "\nStarting position (X and Y), %f%f", &xor,
&yor);
scanf(in, "\nPlotting trace of starting, %f", &xmin);
scanf(in, "\nPlotting trace of ending, %f", &xmax);
scanf(in, "\nPlotting point of starting, %f", &ymin);
scanf(in, "\nPlotting point of ending, %f", &ymax);
scanf(in, "\nStarting amplitude, %f", &chp);
scanf(in, "\nFirst trace you want to tic, %d", &lxmin);
scanf(in, "\nLast trace you want to tic, %d", &lxmax);
scanf(in, "\nTrace increment to be ticed, %d", &ldx);
scanf(in, "\nFirst point you want to tic, %d", &lymin);
scanf(in, "\nLast point you want to tic, %d", &lymax);
scanf(in, "\nPoint increment to be ticed, %d", &ldy);
scanf(in, "\nFirst trace you want to lable, %d", &x1xb);
scanf(in, "\nTrace increment to be lable, %d", &ldx1);
scanf(in, "\nStarting lable for trace, %d", &Lxb);
scanf(in, "\nLable increment per trace, %d", &d1xb);
scanf(in, "\nFirst point you want to lable, %d", &y1xb);
scanf(in, "\nPoint increment to be labled, %d", &ldy1);
scanf(in, "\nStarting lable for point, %d", &lyb);
scanf(in, "\nLable increment per point, %d", &d1yb);
sjm0=0.01;
symb("set10.sym");
xor=xor;
yor=yor;
```

```

x11=(xmax-xmin+2*tin)*dltx/scale1+xor;
y11=(ymax-ymin)*dltly/scale2+yor;
line(xor,yor,x11,yor);
line(x11,yor,x11,y11);
line(xor,yor,xor,y11);
line(xor,y11,x11,y11);
for(i=lxmin;i<=lxmax;i+=ldx){
    ii=(i-lxmin+1)*dltx/scale1+xor;
    line(ii,yor,ii,yor+0.2);
}
for(i=lxlb;i<=lxmax;i+=ldxl){
    ii=(i-xmin+1)*dltx/scale1+xor;
    iix=lxlb+(i-lxlb)*dltx;
    line(ii,yor,ii,yor+0.3);
}
sprintf(ssss,"%d",iix);
is=0;
while(ssss[is]!=NULL)is++;
outtextxy(ii-(float)is/2.0*ht,yor+0.35*ht,an,ssss);
}
for(i=lymin;i<=lymax;i+=ldy){
    ii=(i-ymin)*dltly/scale2+yor;
    line(xor,ii,xor-0.2,ii);
}
for(i=ylxb;i<=lymax;i+=ldyl){
    ii=(1-ymin)*dltly/scale2+yor;
    iix=lyb+(i-ylxb)*dltly;
    line(xor,ii,xor-0.3,ii);
}
sprintf(ssss,"%d",iix);
outtextxy(xor-0.35*ht,ii-ht/2.0*ht,an,ssss);
}
outtextxy(xor+(xmax-xmin+1)*dltx/scale1/2,yor+1.5*ht*2,
an,ss);
outtextxy(xor-2.0,yor-(ymax-ymin)*dltly/scale2/2,ht*1.5,an+
90,ss);

setcolor(color[1]);
cmax=0;
il=0;
for(s=s1;s<=s2+1;s+=tin){
    bmax[s]=0.0;
    for(sd=s1;sd<=s2+1;sd+=pin){
        ss=(s-1)*(ss4*4+head)+head+(sd-1)*4;
        fseek(in2,ss,SEEK-SET);
        fread(&ch,4,1,in2);
        ch=fabs(ch-chp);
        if(bmax[s]<ch)bmax[s]=ch;
        if(cmax<ch)cmax=ch;
    }
    if(bmax[s]==0.0){
        bmax[s]=dltx/scale1*tin/bmax[s];
    }
}

if(cmax==0.0){
    cmin=0.0;
}
else

```

```

{
    cmin=dltx/scale1*tin/cmax;
}

for(s=s1;s<=s2+1;s+=tin){
    x10=(s-xmin+1)*dltx/scale1+xor;
    for(sd=s1;sd<=s2+1;sd+=pin){
        ss=(s-1)*(ss4*4+head)+head+(sd-1)*4;
        fseek(in2,ss,SEEK-SET);
        fread(&ch,4,1,in2);
        if(key=='y'||key=='Y'){
            x11=x10+(ch-chp)/scale3*cmin;
        }
        else{
            x11=x10+(ch-chp/scale3*bmax[s]);
            y11=(sd-ymin)/scale2*dltly+yor;
            ch=ch-chp;
            if(sd==s1){
                ch2=ch;
                x12=x11;
                y12=y11
            }
            else{
                if(fact==1){
                    if(ch*ch2<-0.000001&x11==x12){
                        y01=y12+(x10-
x12)*(y11-y12)/(x11-x12);
                        setcolor(color[1]);
                        if(ch>0.0){yb=y01,ye=y11;line(x12,y12,x10,yb);}
                        else{yb=y12,ye=y01;line(x10,ye,x11,y11);}
                    }
                    if((ch)=0.0)&&(ch2)=0.0)&&(ch*ch2>0.000001)){yb=y12;
ye=y11;}
                    if((x11==x12&&ch*ch2<-0.000001)||((ch)=0.0&&ch2)=0.0)
&&(ch*ch2>0.000001)){
                        setcolor(color[2]);
                        ssc=(x11-x12)/(y11-y12);
                        for(y=yb,ye=y11;y=y+ssjm0){
                            x01=x12+(y-y12)*ssc;
                            line(x10,y,x01,y);
                        }
                        else{setcolor(color[1]);line(x12,y12,x11,y11);}
                    }
                }
                else{
                    setcolor(color[1]);
                    line(x12,y12,x11,y11);
                }
                ch2=ch;
                x12=x11;
                y12=y11
            }
        }
    }
}

end()
}

```

参考文献

- 1 王常明. 利用.PLT文件和PLOT绘制工程图形, 计算机应用研究, 1994. 6

AIX 逻辑卷管理应用于建立 CC—MAIL 邮局

王 蕾

(浦东新区计算机信息中心 上海 200120)

摘 要 本文分析了在 UNIX 家族中独具特色的 AIX 逻辑卷管理器的优点,并应用其特点结合实际在 RS/6000 系列小型机 380 上建立 CC—MAIL 邮局,描述了建立邮局的主要过程。

关键词 AIX 操作系统 日志文件系统 逻辑卷管理 TCP/IP CC—MAIL 邮局

1 引言

随着开放式系统的广泛应用,UNIX 的不断发展,IBM 公司在九十年代推出了 RS/6000 系列小型机。目前 RS/6000 系统上运行的 AIX (Advanced Interactive Execution) 4.1 操作系统是一个高度可靠性、可用性和服务性的 UNIX 平台。在文件管理、安全性、多语言支持和网络管理等各方面独具特色,尤其是逻辑卷管理 (LVM) 及集成在化的系统管理工具 SMIT (System Management Interface Tool),使其成为 UNIX 领域中的佼佼者。如何将其优势与实际应用结合起来,是拥有 RS/6000 系列小型机的用户所最为关心的问题。

而 CC—MAIL 是在 E—MAIL 领域中应用较广且技术成熟的电子邮件产品,它具有独立于系统和网络环境的特点,已被选为国内政务系统推广的首选电子邮件软件。为了使昂贵的 RS/6000 系列的 380 小型机发挥更多的效益,最近我们结合建立 CC—MAIL 邮局的项目,应用 AIX 逻辑卷管理原理,并克服异种机互联之难点,在 380 小型机上安装了邮局。

2 AIX 的文件系统和逻辑卷管理

AIX 操作系统中的存储管理采用的日志文件系统 (Journaled File System) + 逻辑卷管理器 (Logical Volume Manager) 模式。这种模式将复杂的文件系统日志管理与数据透明镜像集成在一起,能有效地利用存储资源,提高数据的访存速度,并且支持超长的文件系统 (10^{12} bytes)。

2.1 日志文件系统

AIX 日志文件系统的结构与其它 UNIX 的文件系统相似,是一个倒挂的树形结构,由目录、子目录和一些文件所组成。它的特点是可以在这棵树上“挂接”多个文件系统,只有当文件系统被“挂接”上时,其空间对用户才是可用的。

2.2 逻辑卷管理器

逻辑卷管理器是 AIX 特有的,它是处在应用程序和管理存储设备的硬件之间的一个软件层。逻辑卷管

理器包含有卷组、物理卷、物理分区、逻辑卷和逻辑分区的概念。

(1) 物理卷 (Physical Volume (PV)) 和物理分区 (Physical Partition (PP))

物理卷是指硬盘。卷组 (Volume Group (VG)) 是指一个处理器上相关的物理卷的集合,但它不是其它卷组的成员。有了这个概念,系统管理员只要使用 Varyonvg/varyoffvg 命令就能使一个卷组联机和脱机。每个系统最多可有 255 个卷组。若要在已存在的卷组上增加一个物理卷 (即硬盘),无需关机,只须执行 mkdisk 命令就可动态完成增加硬盘的操作。

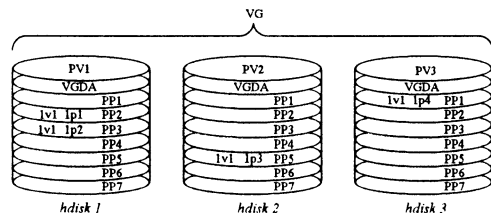
物理分区是卷组中盘空间的最小单位,是一个物理卷上固定的连续字节组。在整个卷组中,物理分区必须是同样大小。

(2) 逻辑卷 (Logical Volume (LV)) 和逻辑分区 (Logical Partition (LP))

逻辑卷是单个文件系统留驻所在,系统允许其可动态地扩展。逻辑分区是一个逻辑卷所对应的 1, 2 或 3 个物理分区,在没有镜像时,与物理分区一一对应。

从下图中可见逻辑卷能跨越多个物理卷,但必须包含在一个卷组之中。每个个卷组至多有 256 个逻辑卷,每个逻辑卷可大于 2GB。

由于文件系统位于一个逻辑卷之上,逻辑卷能动态地改变大小,使得文件系统也能动态地扩展,逻辑卷能跨越多个物理卷,使得文件系统可以处在不连续的空间中。这样不仅增加了文件系统存储的灵活性,而且大大提高了文件的读写速度 (因为处于不同空间,读写机构相对增多,速度得以提高)。



2.3 AIX 文件系统的特点及逻辑卷管理的优越性

根据前面对文件系统和逻辑卷的剖析,可以归纳出 AIX 在文件系统管理方面的一些优势:

(1)可挂接多个文件系统,有利于控制敏感数据,有利于备份或恢复单个文件系统而不是整个系统。

(2)可动态地给系统添加硬盘。

(3)可动态地改变逻辑卷的大小,使得系统安装前无须选择好每个分区的大小。

(4)文件系统可处在不连续的空间中(即一个文件系统可跨多个硬盘),有利于随时扩展文件系统的大小。

3 在 RS/6000 系列的 380 小型机上建立 CC-MAIL 邮局

CC-MAIL 的邮局是网络文件服务器上的一组数据库文件,是 CC-MAIL 信件的中心集散地。所有本地用户和远程用户的邮箱都在邮局中。

现假设网络环境为:以 380 小型机作为 CC-MAIL 的网络文件服务器,10 台微机作为 CC-MAIL 的客户机,采用 TCP/IP 协议构成以太局域网。建立 CC-MAIL 邮局的主要过程如下:

3.1 创建一个逻辑卷

由于逻辑卷是文件系统的驻留地,因此首先要在创建文件系统前创建逻辑卷。在系统已有的 rootvg 卷组上创建一个逻辑卷 ccmallv。根据 CC-MAIL 软件要求,邮局含 3 个子目录,ccadmin 至少 400KB,ccmail 至少 2KB,ccdata 根据用户数而定,一般每个用户需要 2 至 4MB,由此可见,邮局的空间大小是由用户数决定的,过大的文件系统会导致文件搜寻时间加长,系统开销增大,效率降低。建立邮局时无法为今后用户的规模开设空间大小,而 AIX 可动态扩展逻辑卷的功能,使得设置逻辑卷和文件系统的大小只需根据当前用户数来定。

现假定按 10 个用户规模的 CC-MAIL 用户开设 40MB 大小的逻辑卷,并设置 1 个镜像。由于每个物理分区的大小为 4MB,故需开 10 个逻辑分区。执行如下命令:

```
# mklv -y' ccmallv' -c' 2' -L'/home/mailroot rootvg 10
```

这时,在 rootvg 中创建了大小为 81920 块(每块 512 字节),名为 ccmallv,标识为/home/mailroot,有 10 个逻辑分区,20 个物理分区的逻辑卷。

3.2 创建一个文件系统

CC-MAIL 邮局所在目录在 AIX 角度来看就是一个文件系统。现在来设置一个大小为 81920 块的文件系统。

```
# crfs -v jfs -d' ccmallv' -m' /home/mailroot' -A' no' -p' rw' -t' no' -a frag='4096' -a nbpi='4096' -a compress='no'
```

此时,系统根据逻辑卷的大小产生了大小为

81920 块的文件系统/dev/ccmailv,并有可读/写的访问权限其挂接点为/home/mailroot。若修改-A 参数为 'yes',则系统将在启动引导时自动挂接该文件系统。

3.3 挂接已创建的文件系统

文件系统只有挂接后才能存取。执行如下命令进行挂接:

```
# mount -v' jfs' /dev/ccmailv/home/mailroot
```

至此,安放邮局的目录/home/mailroot 的物理空间已创建好并可使用。

3.4 扩展逻辑卷和文件系统的大小

假定上述建立的逻辑卷需扩展 8MB,即需增加 2 个逻辑分区,系统自动向该逻辑卷增加 4 个物理分区,执行命令:

```
# extendlv ccmallv2
```

这时,逻辑分区增加到 12 个,系统自动将物理分区增加到 24 个,逻辑卷大小扩展到 98304 块。再执行命令:

```
# chfs -a size='98304' /home/mailroot
```

此时,文件系统的大小扩展到 98304 块。

3.5 权限设置

根据 CC-MAIL 软件的要求,子目录 ccadmin 只对管理员有读/写/执行的权限,子目录 ccmall 只对用户有读/执行的权限,ccdata 对用户有读/写/执行的权限。

(1)用 mkgroup 命令创建三个组,组名分别为 CCADMIN,CCMAIL,CCDATA。如: # mkgroup -A' ccadm

(2)增加 CC-MAIL 管理员用户 POSTADM,程序员用户 POSTMAIL 和一般用户 POSTUSR。将 POSTADM 设置在 CCADMIN 组,并与 CCMail、CCDATA 同组,这样既是管理员又是一般用户,执行命令:

```
# mkuser pgrp='ccadmin' groups='ccadmin,ccmail,ccdata' home='/home/mailroot/ccadmin' postadm
```

这时,即增加了用户,又创建了目录/home/mailroot/ccadmin。将 POSTMAIL 设置在 CCMail 组,创建目录/home/mailroot/ccmail;将 POSTUSR 设置在 CCDATA 组,并与 CCMail 同组,这样 POSTUSR 为一般用户,创建目录/home/mailroot/ccdata。执行类似的 #mkuser 命令即可完成。

(3)根据 CC-MAIL 软件的要求,对子目录 ccadmin,ccmail,ccdata 设置访问权限。按如下命令设置:

```
# chmod 755 /home/mailroot
# chmod 700 /home/mailroot/ccadmin
# chmod 750 /home/mailroot/ccmail
# chmod 770 /home/mailroot/ccdata
```

3.6 利用 NFS 输出和安装文件系统

NFS(Network File System)是不同机器和操作系统之间交换数据的一个标准。NFS 提供对整个局域网内共享文件的存取。

由于 RS/6000 支持 TCP/IP 协议,故在用户端的 PC 机上安装 PCTCP 软件,使 PC 机与 380 小型机实现异种机互联形成以太网域,同时,NFS 嵌入 TCP/IP 模型的应用层。因此,在该局域网上将服务器的目录输出到客户机,执行命令:

```
# /usr/sbin/mknfsexp -d' /home/mailroot' -t
'rw' '-b'
```

即 380 上文件系统/home/mailroot 已输出。在客户机上,利用 PCTCP 软件,可安装远程文件系统/home/mailroot 到本地,由此可实现透明访问。

以上 3.1 到 3.6 的部分步骤可采用 AIX4.1 提供的 SMIT 菜单来完成。

3.7 安装 CC-MAIL 软件

(1) 安装 CC-MAIL 服务器端软件

①在与 380 联网的一台微机上,运行 PCTCP 软件,以 POSTADM 登录,联通 380 上的目录/home/mailroot,并将其设置为微机上虚拟盘 G。这时,可访问 G 盘的三个目录:ccadmin,ccmail,ccdata。

②在 DOS 平台上安装邮局:在安装过程中,根据

提示将邮局管理程序安装在 g:\ccadmin;用户程序安装在 g:\ccmail;邮局数据子目录安装在 g:\ccdata。

(2) 安装 CC-MAIL 用户端软件

以 POSTUSR 登录,联通 380 上的/home/mailroot,并将其设置为微机上虚拟盘 H,此时,只含两个子目录 ccmil 和 ccdata。在安装时,按提示输入 MAIL.EXE 所在的路径 h:\ccmail 和邮局数据路径 h:\ccdata

4 结束语

随着计算机通信在全球的高速发展,CC-MAIL 的用户将有很大发展,这使邮局能否动态扩展成为一个至关重要的问题。而前面所提及的系统(除 UNIX 家族中 AIX)外,均难实现。若网络系统已有 380 小型机,不妨将邮局设在其上,充分利用逻辑卷管理器的优势,以提高 CC-MAIL 邮局管理效率。

本项目得到联想系统集成公司方皓先生的帮助,在此深表谢意。

参考文献

- 1 RS/6000 简介,联想集成系统有限公司,1996
- 2 蔡传俊主编,UNIX/TCP/IP/NFS 网络编程与应用开发,海洋出版社,1993
- 3 Dave Gibbons 著,E-MAIL 电子邮件使用大全,学苑出版社,1994

24LC32 串行 E²PROM 在 96 单片机系统中的应用

王 辉 马洪江

(湖南大学 长沙 410082)

摘 要 本文介绍了串行 E²PROM24LC32 的基本情况及其 I²C 总线规程,介绍了 24LC32 在 96 单片机系统中应用时的接口方式,同时也提供了接口程序。

关键词 串行 E²PROM 96 单片机 接口

Microchip 公司生产的 24LC 系列串行 E²PROM 具有与 I²C 兼容的总线规程,其价格低,只须 5V 的编程电压,电可改写,具有简单的二线接口,使之成为用作辅助存储空间理想器件。本文介绍了 24LC32 串行 E²PROM 与 96 单片机的接口方式。

24LC32 为 4KB 的串行 E²PROM,可在 2.5V 至 6V 的电压范围内工作,具有 100KHz 和 400KHz 两种模式,具有 8 个 8 字节的页面,典型的写时间为 2ms,最多可在 I²C 总线上挂 8 片 24LC32,得到 32KB 的容量,有 DIP 和 SOIC 两种封装,管脚排列如图 1。各管脚定义如下:

A0、A1、A2 为片地址输入端,用来完成两线总线标准下的多器件操作,SDA 为串行地址/数据输入输出端,SCL 为串行时钟输入端,SDA、SCL 需要各接一个上拉电阻到 VCC,100KHz 时为 10K,400KHz 时为 1K,VCC 接 2.5V 至 6V 的电源,VSS 接地,NC 无内部连接。

8 片 24LC32 挂在 I²C 总线上,与 96 单片机的连接方式如图 2。由于 96 单片机无 I²C 接口,因此用 96 单片机的 P2.5 和 P2.6 模拟 I²C 时序,P2.5 输出串行时钟,P2.6 输入输出数据。

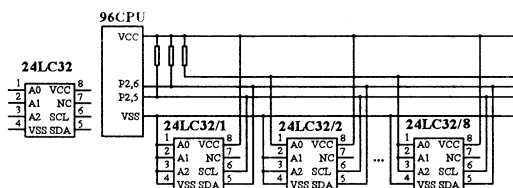


图 1

图 2

I²C 的总线规程定义如下:

- (1) 总线不忙: 数据线和时钟线均为高;
- (2) 开始数据传送: 时钟线为高, 数据线由高到低的变化为开始条件;
- (3) 停止数据传送: 时钟线为高, 数据线由低到高的变化为停止条件;
- (4) 数据有效: 开始条件后, 时钟线高电平期间, 当数据线稳定时, 数据线的状态表示数据有效;
- (5) 确认: 在接收到每一字节后. 当寻址时, 接收器件必须产生一个确认信号, 当内部编程周期正在进行时, 24LC32 不产生确认信号;
- (6) 器件寻址和操作:

控制字节跟随在开始条件后, 其前四位为控制码 1010, 表示对串行 E²PROM 进行读或写操作, 接着是器件选择位 A2、A1、A0, 用此三位选择对 8 片 24LC32 中的哪一片进行读或写, 最低一位是读写控制位, 为 1 时表示读 E²PROM, 为 0 时表示写 E²PROM. 接收到的下两个字节定义了第一个数据字节的地址, 因为仅使用 A0—A11, 所以最高的四位地址码必须为零. 串行数据先发送最高有效位。

对 E²PROM 的操作有读当前地址内容、读随意地址内容、读顺序地址内容、字节写、页面写 5 种方式, 本文提供了一个接口程序, 入口方式是: 往 30H 存储单元写入对 24LC32 进行读写操作的首地址, 它为 16 位地址, A15 无效, A14—A12 为 8 片 24LC32 时的逻辑地址, 由软件分离到控制字中, A11—A0 为片内地址; 往 32H 存储单元写入单片机的数据缓冲区的首地址; 往 34H 存储单元写入本次读写操作的字节数, 其值小于等于 255, 即可直接调用接口程序, 程序中对用到的寄存器均有入栈保护. 本接口程序已用 AEDK5196 仿真器调试通过, 在开发的产品中用于写定值。

对 E²PROM 进行页面写有其特殊的地方. E²PROM 写入时, 需要一定的写入时间, 无法连续地写入多个数据字节, 因此设置有页面写功能, 即在 E²PROM 中设有一定容量的数据寄存器, 用户一次写入 E²PROM 的数据字节可用较快的速度写入到数据寄存器, 遇到停止条件时, 启动内部写周期. 页面写时, 地址的低位部分(页面地址)溢出时不会向高位部分进位, 造成写入数据在地址空间出现“翻卷”现象, 因此在软件中作了相应的处理. 在读 E²PROM 时无翻卷现

象, 且可一次读出整个存储器的内容。

24LC16B 型的 E²PROM 与 24LC32 相似, 但它具有硬件写保护, 只能在 I²C 总线上挂 1 片, 为 2KB 容量, 只有 11 位地址, 高 3 位在控制字中, 低 8 位作为一个字节跟在控制字后发送, 只有一个 16 字节的页面. 本文提供的接口程序作适当修改也适用于 24LC16B。

接口程序清单。

```

ORG 2080H
ANDB 16H, #0FEH; 设置 P2.5 为输出口
; . . . . . ; 主程序其他部分
LD 30H, #0F24H; 写调用程序段
LD 32H, #0A700H;
LDB 34H, #0FFH;
LCALL WRN ; 将 A700H 开始的 255 个字节写入 EEPROM 中
0F24H 开始的单元
; . . . . . ; 主程序其他部分
LD 30H, #0C93H ; 读调用程序段
LD 32H, #0A700H ;
LDB 34H, #0F3H ;
LCALL RDN ; 将 EEPROM 中 0C93H 开始的 243 个字节读
入 A700H 开始的单元
; . . . . . ; 主程序其他部分
WRN: PUSHF ; 子程序部分
PUSH 30H
PUSH 32H
PUSH 34H
PUSH 36H
PUSH 38H
PUSH 3AH
ORB 34H, 34H ; 连续写 1—255 个字节数据
JE WRLOP4 ; NUM=0, 返回主程序
ANDB 36H, 31H, #70H
SHRB 36H, #03H
ORB 36H, #0A0H ; 地址 A14—A12 转到控制字中
ANDB 31H, #0FH ; 高 4 位地址必须为 0
LDB 3BH, 30H
ANDB 3BH, #3FH ; 取低 6 位地址
WRLOP1: CMPB 34H, #40H ;
JH WRLOP2 ; 还需写多于一页的数据, 转 WR-
LOP2
ADDB 37H, 3BH, #34H ;
CMPB 37H, #40H ;
JNH WRLOP3 ; 余下的数据可一次写完, 转 WR-
LOP3
WRLOP2: LDB 3AH, #40H ;
SUBB 3AH, 3BH ; 本次页面写的数据个数
LCALL WRPAGE
SUBB 34H, 3AH ; 余下待写的数据个数
LDB 3BH, #00H
ADD 30H, 3AH ; EEPROM 地址调整
SJMP WRLOP1
WRLOP3: LDB 3AH, 34H
LCALL WRPAGE ; 一次写完
WRLOP4: POP 3AH
POP 38H

```



```

POP 36H
POP 34H
POP 32H
POP 30H
POPF
RET
WRPAGE, LDB 37H, 3AH ;模拟 IIC 总线往指定地址开始写 n
           个字节数据, #01<=n<=#40H
LCALL STAR
LDB 38H, 36H ;发送写命令
LCALL WRBYT
LCALL CACK ;检查应答位
JBS 39H, 0, WRPAGE;非正常应答时重新开始
LDB 38H, 31H ;发送写入地址高位, 低位
LCALL WRBYT
LCALL CACK
JBS 39H, 0, WRPAGE
LDB 38H, 30H
LCALL WRBYT
LCALL CACK
JBS 39H, 0, WRPAGE
WRDAT, LDB 38H, [32H]
INC 32H
LCALL WRBYT
LCALL CACK
JBS 39H, 0, WRPAGE
DJNZ 37H, WRDAT;页面写是否完成, 未完转 WRDAT
LCALL STOP
RET
WRBYT, LDB 35H, #08H ;向 SDA 线发送一个数据字节, 数据
           长度为 8 位
WRLP0, SHLB 38H, #01H ;待发送数据左移, 使待发送位进入
           C
JC WR1
SJMP WR0
WRLP1, DJNZ 35H, WRLP0 ;8 位是否发送完, 未完转 WRLP0
RET
WR0, ANDB 10H, #0BFH ;发送 0 程序段
ORB 10H, #20H
NOP
ANDB 10H, #0DFH
SJMP WRLP1
WR1, ORB 10H, #40H ;发送 1 程序段
ORB 10H, #20H
NOP
ANDB 10H, #0DFH
ANDB 10H, #0BFH
SJMP WRLP1
RDN, PUSHF
PUSH 30H
PUSH 32H
PUSH 34H
PUSH 36H
PUSH 38H
PUSH 3AH
ORB 34H, 34H
JE RDL0P2 ;NUM=0, 返回主程序
ANDB 36H, 31H, #70H
SHRB 36H, #03H
ORB 36H, #0A0H ;地址 A14-A12 转到控制字中
ANDB 31H, #0FH ;高 4 位地址必须为 0
RDLOP1, LDB 37H, 34H ;模拟 IIC 总线读任意地址开始的 1-
           255 个字节数据
LCALL STAR
LDB 38H, 36H ;发送写命令
LCALL WRBYT
LCALL CACK ;检查应答位
JBS 39H, 0, RDLOP1;非正常应答时重新开始
LDB 38H, 31H ;发送写入地址高位, 低位
LCALL WRBYT
LCALL CACK
JBS 39H, 0, RDLOP1
LDB 38H, 30H
LCALL WRBYT
LCALL CACK
JBS 39H, 0, RDLOP1
LCALL STAR ;重新启动
INCB 36H
LDB 38H, 36H ;发送读命令
LCALL WRBYT
DECB 36H
LCALL CACK
JBS 39H, 0, RDLOP1
RDDAT, LCALL RDBYT ;读一个字节到 36H
STB 36H, [32H]
INC 32H
DJNZ 37H, ACKRD ;n 个字节读完否, 未完转 ACK
LCALL NACK ;n 个字节读完发送非应答位
LCALL STOP
RDLOP2, POP 3AH
POP 38H
POP 36H
POP 34H
POP 32H
POP 30H
POPF
RET
ACKRD, LCALL ACK ;发送应答位
SJMP RDDAT ;转读入下一个字节数据
RDBYT, LDB 35H, #08H ;从 SDA 线读取一个数据字节, 数据
           长度为 8 位
LDB 36H, #00H ;置读出结果初值为 0
RDLP0, ORB 10H, #40H
ORB 10H, #20H
LDB 38H, 10H
JBC 38H, 6, RD0
SJMP RD1
RDLP1, DJNZ 35H, RDLP0 ;8 位读完否, 未完转 RDLP0
RET
RD0, SHLB 36H, #01H ;读入 0 程序段, 拼装入 36H 中, 左移 1
           位
ANDB 10H, #0DFH

```

SJMP RDLPI
 RD1: SHLB 36H, #01H ;读入1程序段,拼装入36H中,左移一位
 ORB 36H, #01 ;36H最低位置1
 ANDB 10H, #0DFH
 SJMP RDLPI
 STAR: ORB 10H, #40H ;开始条件
 ORB 10H, #20H
 NOP
 ANDB 10H, #0BFH
 NOP
 ANDB 10H, #0DFH
 RET
 STOP: ANDB 10H, #0BFH ;停止条件
 ORB 10H, #20H
 NOP
 ORB 10H, #40H
 NOP
 ANDB 10H, #0DFH
 RET
 ACK: ANDB 10H, #0BFH ;发送应答位
 ORB 10H, #20H

NOP
 ANDB 10H, #0DFH
 ORB 10H, #40H
 RET
 NACK: ORB 10H, #40H ;发送非应答位
 ORB 10H, #20H
 NOP
 ANDB 10H, #0DFH
 ANDB 10H, #0BFH
 RET
 CACK: ORB 10H, #40H ;应答位检查
 ORB 10H, #20H
 CLRB 39H
 LDB 38H, 10H
 JBC 38H, 6, CEND
 ORB 39H, #0FFH ;无正常应答, 39.0=1
 CEND: ANDB 10H, #0DFH
 NOP
 RET
 END

参考文献

- 1 Microchip 公司 1994 年数据手册

便携式条码识别存储器

许伟 王谦

(西南交通大学 成都 610031)

摘要 本文提供了一种用单片机组成的便携式条码识别、存储装置。其特征在于设置了内藏式微型光电阅读头,不需其它外部设备便可识别和处理国际上通用的六种条形码,并可作计算器使用。

关键词 条形码 光电阅读头 比较器

1 引言

目前在条码扫描识读系统中,都需要条码阅读器 and 专用的接口系统。这些条码阅读器的功能单一,只能完成将条码的光电转换,再经整形处理为矩形脉冲,经放大、比较、译码从显示器输出为数字代码,或作为计算机的输入信号进行信息处理。条码从显示器上输出,需要人为的抄录,当数据量很大时,抄录人员工作量大、效率低;条码阅读器与计算机连接时,需要接口系统,由于国际上流行的阅读器与计算机种类较多,难于完全匹配与兼容,致使这种专用的接口系统成为条码系统中不可缺少的一部分,在需要野外使用的场合极为不便。

本文的目的是提供一种便携式条码识别存储器,它能有效地识别和存储国际上通用的六种条码。

2 基本结构

该装置内部有一个 PLLC 封装的 AT89C51 单片机,片内有 4K 字节可编程闪速存储器;非易失 8K×8 数据存储;16×1 字符点阵式液晶显示器;0-9 数字

键, +、-、×、÷、.、= 符号键和其他功能键组成的键盘;以及 RS232 串行异步通信口组成的单片机系统。另外装置内有一个内藏式微型光电阅读头,其输出信号经放大器、比较器和边缘检测组成的条码输入电路处理,信号输出分别送 89C51 的 T1 和 INT0,如图 1 所示。

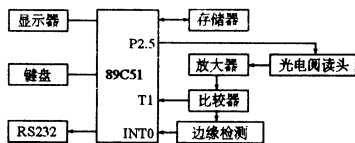


图1 条码识别存储器结构框图

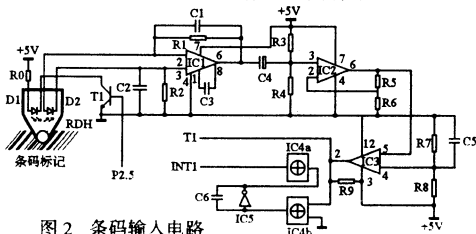


图2 条码输入电路

3 工作过程

开机后,通过执行片内的程序,在液晶显示器上显示提示符“A”,通过键盘键入相应的键以后有以下三种情况:

(一)当在键盘上按下 RD(阅读条码)功能键时,单片机转入执行阅读条码子程序,显示器上显示即将阅读的条码类型,在键盘上连续按下 BC(条码选择)功能键,就可以循环选择“1”(EAN-13 码)、“2”(EAN-8 码)、“3”(UPC-A 码)、“4”(UPC-E 码)、“5”(DUN-14 码)和“6”(DUN-16 码)。这时单片机的 P2.5 引脚输出高电平,使内藏式微型光电阅读头工作,阅读头内的 LED 发光二极管点亮,发射出一束光在条码标记上聚成一个光点。光点直径小于条码标记中最窄的标记。从条码标记返回的光为光检测器所接收。光检测器产生的信号经条码输入电路处理,信号输出分别送单片机的 T1 和 INT0 输入端,经过译码,得到一数字代码送到液晶显示器,如没有数字显示或者显示的数字不正确,则重新对准条码再进行一次扫描,或者直接在键盘的数字键上键入这个条码的数字。如按下键盘上的 CT(继续)功能键,则将这一代码存储在非易失数据存储器中,再继续下一个条码的阅读;如按下键盘 3 上的 ED(结束)功能键,则退出阅读条码子程序,返回主程序。

(二)当在键盘上按下 TR(发送条码)功能键时,单片机转入执行发送条码子程序,显示器上显示与计算机通讯的提示符“C”。这时本装置通过串行异步通讯口与计算机的 RS232 相连接,可以由计算机将存储于数据存储器中的数据代码读出送到计算机内。数据读出完毕以后,计算机使单片机退出发送条码子程序,返回主程序。

(三)当在键盘上按下 CU(计算器)功能键时,单片机转入执行计算器子程序,显示器上显示数字“0”提示符。这时本装置相当于一部普通的计算器,通过键盘上数字和符号键,能进行包括小数点在内共 16 位数字的加、减、乘、除运算。如按下键盘上的 ED(结束)功能键,则单片机退出计算器子程序,返回主程序。

4 条码输入电路

读条码电路如图 2 所示,它由 4 个部分构成,分别为光电阅读头,放大部分,比较电路,边缘检测电路。光电阅读头由高亮度的光电二极管 D1 和平面硅 PIN 光电二极管 D2 组成。当单片机的 P2.5 输出高电平时,三极管 T1 导通,D1 发出的光通过阅读头尖上的宝石(或碧蓝色玻璃)圆球聚焦在条码标记上。条码标记的光返回后经头尖上的小球进入阅读头中为 D2 所接收。D2 输出的电信号由接成标准差放的 FET 运放 IC1 进行放大。运放周围所接电容是为了减小来自交

流光的纹波干扰和控制响应时间用的。IC1 输出的信号由构成标准同相放大器的运放 IC2 作进一步放大。放大器增益约为 200,输出信号幅度在 0 和 2.5V 之间。此电压信号送入电压比较器 IC3,产生高低电平宽度不同的脉冲信号,此信号传入单片机的 T1 端和 IC4、IC5 组成的边缘检测电路,此时 T0 作为定时器使用,T1 端利用其第一功能作为高低电平的记录,边缘检测电路输出接入单片机的 INT0 端。

采用人工扫描方式,扫描速度要求保持在 75—1300mm/s 范围内,扫描速度低于 75mm/s,常常会发生抖动,不易保持匀速扫描;扫描速度高于 1300mm/s,超过人工手动正常工作速度,使人感到不适。在放大倍率为 1 的情况下,一般一个模块的最小宽度为 0.33mm 左右,条码的最大模块数为 4 个,空白区是 9—10 个模块,以 75mm/s 的速度扫描 4 个模块需要 17600us,10 个空白区需要 44000us,本单片机采用 12MHz 的振荡器,定时器 T0 的初值可定为略大于 ABE0H,可保证在以这样的速度扫描不会发生溢出,一旦发生则可认为检测到结束符。

边缘检测电路第一次检测到 IC3 输出的电压跳变则产生一个负脉冲,同时通过 INT0 端产生中断,触发中断服务程序,启动定时器 T0 开始计数;当边缘检测电路检测到 IC3 输出的下一个电压跳变,又产生一个负脉冲,再次触发中断服务程序,把定时器 T0 此时的值送入指定的片内 RAM 单元,定时器 T0 的初值减去这个值得到的即为此条码的宽度,同时把 T1 端检测到的“1”或“0”电平记录下来,送入指定的片内 RAM 单元,定时器 T0 恢复初值,重新开始计数运行。这个过程重复进行,直至检测到超过一定数目的空白区,完成条码扫描到此完成。

5 结束语

影响条码识别效果的因素有很多,主要是系统误差和随机误差。光学扫描系统输出的是模拟信号,虽经比较器转化为数字信号,但不能很准确地反映条码的宽度。人工扫描速度不均匀是随机误差的主要因素。所以,只有尽可能的考虑所有这些因素的影响,才能得到较高的识别率。

参考文献

- 1 李华主编. MCS-51 系列单片机实用接口技术. 北京航空航天大学出版社, 1993 年 8 月
- 2 马晓峰, 程承恩编著. 条码技术手册. 中国计量出版社, 1992 年 8 月
- 3 黄以群, 董湘陵编著. 条形码技术. 国防工业出版社, 1991 年 3 月
- 4 孙涵芳, 徐爱卿编著. 单片机原理及应用. 北京航空航天大学出版社, 1988 年 2 月

VESA 图象编程及其应用

魏庆国 黎 鹰 吴建华

(南昌大学电子工程系 330029)

摘 要 本文介绍了 VESA 视频接口标准及其图象编程技术,并给出了在卫星云图显示中的一个应用实例。

关键词 VESA 标准 图象编程

1 引言

视频显示适配器经历了 MDA、CGA、EGA 和 VGA 的发展历程,现在向更高分辨率、更多颜色数目的方向发展。当初,IBM 推出 VGA(Video Graphics

Array)显示技术的时候,为了使应用程序的视频接口达到标准化,制定了一套标准的 VGA 显示模式。后来,随着计算机应用领域的不断扩展,计算机的图形功能越来越引人注目。在个人计算机上,一个友好的交互式用户图形界面可以大大提高使用效率。在数字图象处理领域,以前必须依赖于图象显示接口外接一台监视器作为图象显示设备,而现在由于计算机显示器分辨率的提高和颜色数目的增多,在计算机显示器上可以直接进行图象图形显示。

但是,VGA 以后,虽然高分辨率显示适配器相继出现(这些产品统称为 Super VGA,SVGA),但不同厂商在如何进行技术上的改进及其扩充 VGA 的显示模式方面缺乏统一的标准,使得众多的软件开发者在这些增强特性时遇到很大困难。有的软件如 Windows 要求厂家提供驱动软件,已部分地解决了这一问题。但用户如要在自己的软件中使用高级的 Super VGA 的特性,特别是效率、速度成为软件的重要指标的时候,直接对视频接口编程仍然是最佳选择。制定统一的视频标准还是必要的。

为此,视频电子标准协会(VESA)承担了这一任务,制定了 VESA 标准,以规范视频显示器的编程。近年来的 Super VGA 产品均支持 VESA 标准。本文在 VESA 图象编程方面进行了一些探讨和实践,并将其应用于卫星云图显示,是一次 VESA 图象编程的成功的尝试。

2 VESA 标准

(1)VESA 显示模式

VESA 规定了一组高分辨率模式下的标准模式集。为了不与现有的模式号发生冲突,VESA 将模式号从 7 位扩充到 15 位。VESA 标准模式见表 1。

(2)VESA 附加的 BIOS 功能

与标准 VGA 的 BIOS 功能一样,所有 VESA 扩充

模式号	分辨率	颜色数
100h	640×400	256
101	640×480	256
102	800×600	16
103	800×600	256
104	1024×768	16
105	1024×768	256

的 BIOS 功能都是通过中断 10h 来实现的。已指定的 SVGA 的扩充功能号 4Fh,所有的扩充 BIOS 功能具有相似的格式:

AH=4Fh

AL=VESA 功能代码

每一个功能的返回状态信息存于寄存器 AX 中,其意义为:

AL=4Fh 如果支持此功能

AH=0 如果调用成功,否则失败

VESA 扩充 BIOS 的功能号为 0~5,它们的功能依次为返回 Super VGA 信息,返回 SVGA 模式信息,返回当前显示模式,保存和恢复 Super VGA 状态以及显示存储器窗口控制。

详细了解这些功能的调用方法可参见文献[1]。在此,仅对显示器分页技术及编程作一说明。

Super VGA 使用 256k,512k 或 1024k 的显示存储器结构,为了使处理器可通过一个 64k 窗口来存取这样大的显示存储器,大多数 SVGA 采用存储器分页机制,将显示存储器的 64k 映射到 CPU 的地址空间中。这种分页技术类似于扩充存储器(EMS)的分页方法,将显示存储器划分为若干页,使得在同一时间只允许选择 1 个存储器页。VESA 的页尺寸为 64k,页的多少由视频存储器容量和页尺寸决定。

VESA 通过视频 BIOS 功能 5 来实现页的选择,C 语言源程序为:

```
Void selectpage(page)
unsigned char page;
{ unsigned addr;
RW_Page=page;
addr=64
addr/=vesa->Win_Granularity;
addr*=RW_Page;
asm mov dx,addr
```

```

--asm mov ax,4f05h
--asm mov bx,0000h
--asm int 10h
return;
}

```

其中 RW—Page 是外部变量,以使其他程序通过访问 RW—Page 判断视频显示器已选择了哪页。vesa 是存储 SVGA 信息的结构指针,通过 BIOS 功能 1 可以返回关于某个模式的信息,Win—Granularity 是显示存储器的粒度,即存储器页起始地址的最小增量(单位为 k 字节)。上述 C 语言源程序用到了内部嵌入汇编,编译程序必须为 Borland C++3.1 以上或 Microsoft C6.0 以上版本。

3 图象显示子程序库

根据 VESA 标准和图形技术,在 256 色模式下我们对有关图象显示的程序编制了不同的功能模块,可以方便地进行调用。这些程序模块均针对 256 色模式,包括模式 100h,101h,103h 和 105h。下面对主要程序模块作原理性的说明。

(1) 象素读写程序 putpixel(int x,int y, char color) 以及 unsigned char getpixel(int x,int y)

由前面所述的显示存储器的分页机制,256 色模式下每个象素占显示存储器的 1 个字节,要将 1 个象素的 x,y 坐标位置转换成 64k 页面的显示存储器的页(page)和页内偏移地址(offset)(图 1),可使用下列关系式:

$$\begin{cases} \text{pagei} = (y * X_r + x) / 10000h \\ \text{offset} = (y * X_r + x) \bmod 10000h \end{cases} \quad (1)$$

其中 X_r 为所选模式的水平分辨率,除法运算完成后必须取整,mod 表示求余。(1)式的运算可以非常高的效率实现,以长整数存储 $(y * X_r + x)$ 的结果,则 page 为这个长整数的高 16 位,offset 为这个长整数的低 16 位。而每个存储器的段地址都是 A000H。

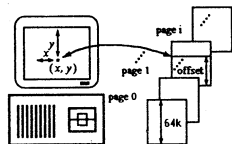


图1 象素坐标和显示页、偏移对应关系

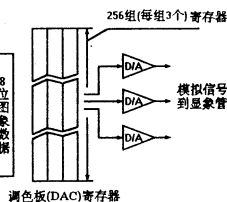


图2 图象数据译码显示原理

(2) 读写扫描行。

读写扫描行可以通过多次调用读写象素来实现,但这样效率太低,因为计算机存储器页和偏移都有重复工作。由于一个扫描行的数据在一个存储页里是连

续存放的,因此只需计算第一个象素的页和偏移,然后用数据块操作指令则可实现高效读写扫描行。但对于一扫描行跨越两个存储页的情况应做不同处理,可以通过调用读写象素实现,也可分两次进行数据块操作,由于这样的行数比起显示器的水平行数少得多,所以分别处理不会明显影响读写速度。

(3) 画一直线、圆、填充实矩形

画圆的程序通过计算圆上的坐标点描出。画线程序采用了标准的贝森汉画线算法,但要增加页边界的检测。填充实矩形调用写扫描行模块来实现。

(4) 置调色板

在 256 色模式下,通过改变视频调色板寄存器来决定具体的象素值对应哪一种颜色。SVGA 图象数据译码显示原理如图 2⁽²⁾。

8 位图象数据在 DAC 寄存器中查找相应的那一组寄存器,取出红、绿、蓝三个彩色分量,送到 D/A 转换器后输出模拟信号以决定象素的显示颜色。因此,调色板(DAC 寄存器)起着查找表的作用。由于每个寄存器是 6 位的,故共有 $2^6 \cdot 2^6 \cdot 2^6 = 2^{18}$ 共 256k 种颜色可供选择。如果是显示黑白图象,由于要求三个彩色分量相等,故只能显示出 $2^6 = 64$ 种灰度出来。

调色板寄存器的设置通过标准 VGA 的 BIOS 调用(功能号 10h)实现,调用前存储调色板的数据指针的段地址和偏移量分别放入 AX 和 DX 寄存器。

4 应用实例

利用 VESA 图象编程模块,我们在 SVGA 上进行了卫星云图显示的程序研制,实现了基本上实时的图象显示,其中近日内数十幅的云图连续显示达到了很好的动画效果,形象、逼真地演示了天气的变化过程。

红外云图是 8bit 图象,可见光云图是 6bit 图象,为了增大动态范围,可见光云图的数据左移 2 位。调色板寄存器的值采用中国科学院图象所提供的数据库^[3],依温度从低到高按蓝、白、红缓慢过渡,以使不同温度的区域有明显的对比。有必要时利用光标定位,精确显示卫星云图上某点的温度值。

卫星云图显示包括改变调色板,放大缩小图象,装入指定云图,红外、可见光云图对照以及动画演示等功能,都是利用前述的基本 VESA 模块编写成的,卫星云图的显示效果略。

参考文献

1. 朱文占. Super VGA 高级编程指南,北京科海培训中心,1991 年 5 月。
2. 周孝宽. 实用微机图象处理,北京航空航天大学出版社,1994 年 10 月。
3. 中国科学院图象所, GMS 卫星云图接收设备说明书。

虚拟视景仿真中的 LOD 及 mipmapping 技术

陈 刚

(郑州解放军测绘学院制图系计算机制图教研室 450052)

摘 要 本文主要介绍了在虚拟视景的生成过程中,为节约视景生成的时间,提高实时仿真的效果,所采用的 LOD 及 mipmapping 技术,包括 LOD 及 mipmapping 的主导思想,特点以及实现的方法和过程。

关键词 虚拟现实 实时视景仿真 LOD 数据模型 mipmapping 纹理

虚拟现实是九十年代刚刚兴起的计算机应用领域的前沿技术,其最大的特点是可进入性和实时性。所谓可进入性指的是通过新型计算机交互设备(如视听头盔,数据手套,立体眼镜等)达到进入计算机所绘制的场景(又称虚拟视景)之目的;而所谓实时性则指所进入的虚拟视景可以实时地进行改变。前者的实现需要相应的辅助设备,而后的实现则归功于性能优越的计算机硬件平台和高效的编程技巧。

有关新型的交互设备的介绍已经有了相应的文章,这里所要谈到的是虚拟现实为实现视景的实时变化所采用的编程技巧,这些技巧对于那些刚刚从事虚拟现实视景仿真或进行其他动画编程制作的程序员都有很大的参考价值。

我们知道,要让计算机所绘制的场景达到实时变化取决于两点:一是所采用的计算机平台具有很高运算速度,比如说带有专业图形卡的高档微机,或者是专业的图形工作站,如 SGI, SUN 工作站等;二是所绘制的几何目标要尽量简单,通过减小计算量的方法来增加实时性能。众所周知,所有的计算机图形制作系统的制图能力都是有限的,包括功能强大的计算机图形工作站在内,如高档的 SGI 工作站,这种限制表现在一定的帧频下,每帧所能绘制的原始几何模型(三角形)数是有限的,而如果通过减少模型的数量或详细程度来节约运算时间,又会带来降低视景的逼真程度的不利后果。在虚拟现实的视景仿真中,为提高视景生成的效率,达到视景充分实时变化之目的,采用了 LOD 技术及 mipmapping 技术。以下分别予以介绍。

1 LOD 技术。

LOD,英文全名为"Level of detail",即细节的详细程度。先来看看 LOD 主导思想。我们知道,同一个物体,把它放到远近不同的位置,人的眼睛所能看到的该物体细节的详细程度是不一样的。比如说一辆轿车,放在近处时,我们可以看的见它的门柄,它的牌照,它的后视镜,甚至还可以看见驾驶舱内的驾驶员。而如果汽车开远后,或是在有雾有烟的环境下,象门柄,牌照

这些细节会变得模糊起来,甚至于因太小而予以忽略,这时识别汽车的主要依据只剩下它的车身了,而就是车身,随着离视点越来越远,其形状也变越来越简单。因此,对于近在眼前和远在千米外的同一辆轿车,如果采用同一个数据模型来参与视景的生成是很不科学的。LOD 技术正是根据这一视觉特点,为同一个物体建造了一组详细程度有别的几何模型。计算机在生成视景时,根据该物体所在位置离视点距离的大小,分别调入详细程度不同的模型参与视景的生成。这样,避免了不必要的计算,既能节约运算时间,又不会降低视景的逼真程度,使计算的效率大大提高。

我们再来看看 LOD 是怎么实现的。LOD 的实现要分为两步:1)为物体建造一组详细程度不同的模型。2)建立好约定,即怎样根据物体离视点的距离来调用相应的模型。

1.1 为物体建造一组详细程度不同的模型

这一步的关键实际上是数据模型的简化。有关数据模型简化的方法有很多种,用户可以根据不同应用领域的特点来选择不同的简化方法。因数据模型的简化不是本文的讨论重点,故对简化不再作进一步描述,暂且假设已经有了一种模型简化的方法。通过利用一定和简化方法对相应的目标进行简化分级,形成一组详细程度有别的 LOD 数据模型。将这一组 LOD 模型根据细节的详细程度从多到少进行排序,并用序列号(1, 2, ..., N)给以标识,以便计算机进行选择,同时,简化的对象可以不单指一个目标,也可以对该目标下的各个子目标也建立相应的 LOD 模型组。比如,不仅可以对车身建立一组 LOD 模型,同样也可以对车轮,后视镜等建立各自的 LOD 模型组。有关分级数的大小和每一级数据模型的细节详细程度,可根据需要来确定。图 1 显示了模型结构的划分。

1.2 建立模型与视距间的关系约定

通过计算视点与目标中心点间的距离,我们可以得到目标的视距。为每一个目标建立一个有关视距的阈值,用阈值把视距划分为不同的视距段。在选择 LOD 模型参与视景生成计算时,首先判断目标的当前视距处于哪个视距段,再找到该视距段所对应的该目

标 LOD 数据模型的标识号,调用标识号所指向的 LOD 模型来代表参与该目标与视景生成。还可以在最近和最远处增设两个视距段,当视距小于最近视距段或大于最远的视距段时,认为该目标处于不可见位置,不将该目标的数据模型参与视景的生成。同时,为减小两个 LOD 模型间视觉上的跳跃现象,还可采用透明技术在视景生成时将相邻的两个 LOD 模型进行加权融合,即当视距处于视距段的交接域时,同时采用两个 LOD 模型参与计算,使用透明的效果来生成一个过渡模型,以减小在切换相邻的两个 LOD 模型时视觉上的跳跃现象。当然这会增加计算量。图 2 表示了 LOD 模型与视距间的对应关系。

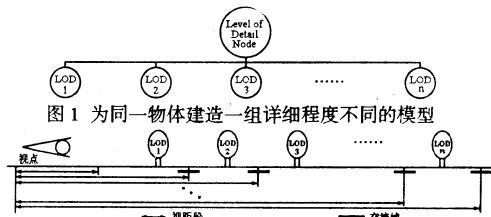


图 2 构造模型与视距的对应关系

以下图 3 所示的是在 SGI 工作站上用 Performer 生成的一组 LOD 模型的视景效果。以轿车为例,给出了五个详细程度不同的 LOD 模型。虽然作为数据模型来讲,LOD 模型的空间大小是一致的,但在实际的虚拟视景中,因远近不一样,它们不应该是等大的,这里仅仅是为了观察方便。

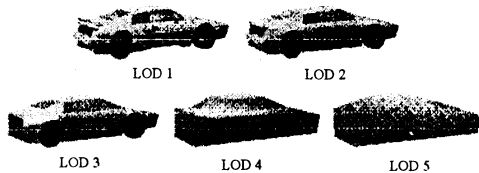


图 3 一组详细程度不同的轿车 LOD 模型

2 mipmapping 技术

mipmapping Mip 是拉丁文“multum in parvo”的缩写。英文译成“many things in a small place”,我们不妨把它翻译成“纹理的压缩”。这是因为 mipmapping 技术是一种高效的纹理映射技术。

要让生成的虚拟视景具有很强的真实性,真正做到以假乱真的效果,那是绝对离不开纹理映射的,并且纹理映射还可以简化模型。比如说虚拟一栋房子,我们可以把它的数据模型构造的非常简单(例如用一个长方体来代替),然后把每个面的照片作为纹理贴上去,最后得到的效果要比不用纹理,而模型建造得相对更加复杂所得到的效果要好的多。而且前者可以减少多边形的数量,节约生成视景所耗用的时间,增加实时效果。

纹理映射虽然可以增加虚拟视景的逼真程度,掩盖简单数据模型带来的不真实性,但纹理映射是点与点之间的对应过程,这一映射过程需要耗用大量的计算时间。那些具有纹理映射功能的工作平台往往都配备有专门的纹理加速卡来减轻 CUP 的负担。因此,如果能够减少点与点之间的纹理映射量,对提高虚拟视景的生成速度是会有很大帮助的。同 LOD 一样,纹理也有一个远近详细程度不同的问题,比如当房子离视点越来越远时,如果还是用相同大小的图片来做纹理贴面,且不说该做法的低效性,最后的结果还会出现闪烁的现象。在虚拟视景的生成过程中,用 mipmapping 技术来解决这一问题。mipmapping 技术的思想是:为将要贴纹理的几何面准备好一组大小各不相同的纹理图像,根据几何面离视点的距离来调用相应大小的纹理图像参与纹理的映射。

mipmapping 技术的实现实际上同 LOD 技术的实现是一样的。

首先,需要准备好一组大小与详细程度不同的纹理图片(Mipmaps)。原始的纹理图片(指详细程度最高的图片)可以通过扫描照片来获取,利用一定的图像处理工具软件(如 PhotoShop, PhotoStyler 等)对原始的纹理图片进行不同的比例的压缩可进一步得到其它的较小的纹理图片。

然后,建立好不同的纹理图片与视距间的对应关系。当物体离视点越来越远而变的越来越小时,通过对应关系,根据物体的大小变化来选用相应大小与详细程度的图片来做纹理贴面。这样一来,当需要贴纹理的几何面离视点较远时,用较小的纹理图片来作纹理映射,使点与点之间的纹理映射量大大减少,从而提高了映射效率,达到增强了虚拟视景实时性的目的。图 4 以房屋为例,构造了一组用于 mipmapping 技术的纹理图片(Mipmaps)。



图 4 用于 mipmapping 技术的一组房屋纹理图片(Mipmaps)

LOD 技术和 mipmapping 技术的程序实现是很容易的,虚拟现实所提供的编程语言如 OpenGL, Open Inventor, Performer 等都有相应的功能函数支持 LOD 和 mipmapping 的实现。作为技术本身,LOD 和 mipmapping 并不复杂,但这两种技术的运用,却大大改善了虚拟视景仿真的实时性。这些原理和思想对于广大从事动画软件编制的程序员来讲,都具有很大的借鉴和参考价值。

参考文献

1. Jed Hartman, Patricia Creek. 《IRIS Performer Programming Guid》Silicon Graphics, Inc. 1994.

2. Jackie Neider, Tom Davis, Mason Woo. 《OpenGL Programming Guid》Silicon Graphics, Inc. 1993.

利用 Video Blaster SE100 实现连续图象捕捉与实时压缩

曹 宁 吴 瑛

(河海大学常州分校计信学院 213022)

摘 要 本文针对 Video Blaster SE100 用户, 提出如何利用 Windows MCI 快速简便的实现连续图象捕捉与实时压缩, 旨在提高用户开发此类软件的效率。

关键词 图象 图象压缩 控制

1 引言

图象处理作为多媒体技术的一个重要组成部分, 其应用无处不在, 除了通信领域的信息存取与交换之外, 目前还广泛应用于科研、军事、冶金、煤炭、石油化工等工业控制方面。由于各自的侧重点不同, 通用图象软件难以胜任。用户迫切希望能够开发出专用的图象处理软件。本文基于目前广泛使用的 Creative 公司的 Video Blaster SE100(下称 VB SE100)图象卡及大多数用户的需求, 提出快速有效地实现图象的连续捕捉与实时压缩, 旨在帮助用户提高开发此类软件的效率。

2 VB SE100 图象卡及其应用程序接口

VB SE100 是 Creative 公司 1994 年推出的高效图象叠加、采集卡。可令用户在 PC 机显示器上获得来自各种视频源(包括 CCD、录像机、摄像机、影碟机)的实时动态画面, 并可捕捉动态画面中的任一帧图象。该卡还具有图形、图象叠加功能。视频卡软件套件包含了几个 Windows、DOS 通用软件, 用以显示视频及控制视频卡的硬件部分。

视频卡的应用程序接口为用户提供了 DOS、Windows 环境下编程范例, 便于用户进行二次开发。

3 图象卡用户软件设计

对于大多数应用而言, 尤其是许多控制场合, 用户希望能够连续地采集一些画面。但图象信息量是非常巨大的, 例如一幅 CIF 标准(352 像素×288 线)的 24 位色的彩色图象, 其数据量为 304,128 字节, 若每秒捕捉 4 帧, 则数据量为 1,216,512 字节/秒。如此之大的数据量不便于图象信息的传输与存储, 迫使我们必须采用当今多媒体信息处理的核心技术—压缩。对于普通用户而言, 若不具备相当的图象专业知识, 要想开发出这样具有实时压缩功能的连续图象捕捉软件, 其难度可想而知。在这里, 我们提出充分利用 Windows

MCI 实现这一功能。建议用户在 Windows 环境下进行软件的开发, 这是由于 windows 在内存管理、对显示器、鼠标及多任务的支持方面均优于 DOS, 用户无需在底层进行开发, 因而软件开发效率较高。

3.1 Windows MCI

Windows MCI(Media Control Interface)是控制多媒体设备的高层命令接口, 提供了与设备无关(Device Independent)的控制多媒体设备的方法, MCI 包含在 Windows 系统的 MMSYSTEM.DLL 动态连接库中, 用以协调多媒体事件和 MCI 设备驱动程序之间的通信。Windows 系统文件 system.ini 中的[MCI]段用以描述已安装好的 MCI 设备。当用户成功地安装了 VBSE100 之后, 在[MCI]段中将出现: overlay = se100mci.drv.overlay 为视频叠加设备, 可处理活动视频, 用户不必进行 DLL(动态连接库)级的编程, 只需利用视频叠加命令集进行相应的操作。

3.2 MCI 命令字符串接口

Microsoft 提供的 mmsystem.h 文件中定义了调用 MCI 功能的数据类型与函数原型, 在使用 MCI 功能的任何源模块中都应包含该文件。MCI 定义了两种与 MCI 通信的接口方式, 即命令消息方式与命令字符串方式, 相应的函数为命令消息函数与字符串函数, 这里我们使用后者。MCI 命令字符串方式使用 ASCII 字符串来发送驱动 MC 设备的命令其接口函数有 DWORD WINAPI mciSendString(LPCSTR lpstrCommand, LPSTR lpstrReturnString, UINT uReturnLength, HWND hwndCallback); BOOL WINAPI mciGetErrorString(DWORD wError, LPSTR lpstrBuffer, UINT uLength); mciSendString 用于向 MCI 设备发送命令字符串, 第一个参数是远程指针 lpstrCommand, 指向一个以 NULL 结尾的 MCI 命令字符串, 该字符串和格式为: command device—name argument。关于 overlay 命令字符串集我们将作详细的讨论。

该函数的第 2 个参数 lpstrReturnString 指向一个用于存储该命令执行后返回字符串信息的缓冲区, 第三个参数指定该缓冲区的大小, 若 lpstrReturnString 设为 NULL, 则将忽略返回信息。其第四个参数 hwndCallback 指定一个接受 MM-MCINOTIFY 的窗口句柄, 若 MCI 命令中不包含 NOTIFY 标志, 则该参数可忽略。

若该函数返回非 0 值, 则调用失败, 可用 mciGetErrorString 取得包含错误信息的字符串。

3.3 overlay 命令字符串集在用户软件中使用

overlay 命令字符串集中的命令字(command)包括: capability、close、copy、cut、delete、freeze、get、info、load、open、paste、put、save、set、status、unfreeze、where、window 等。这里我们根据自己的编程经历介绍用户软件设计中所需的四个最基本、最重要的命令: open、capability、set、save。由于篇幅所限只能提供相关的源代码(采用 Windows 3.1 下的 Borlandc++4.0 编程环境)。

open 用于初始化视频叠加器件。定义为: open device __id[parameters][notify][wait]。其中有关参数的含义如下:

alias device __alias 用于为视频叠加器件定义一个别名, 在以后的命令字符串中可将其作为器件的标识号 device-id。

Style Style Value 用于定义缺省窗口的风格如 overlapped、popup、child 等。

parent hwnd 当缺省窗口风格为 child 时, 为其定义父窗口。

[notify]和[wait]作为关键字使得命令执行完之后再返回应用程序。

我们在程序中使用命令 open 的相关源代码为:

```
.....
char szMciString[MAXFILENAME];
//初始化命令字符串。
lstrcpy (szMciString, "open overlay alias vb
style child parent"); //构造命令字符串, vb 作为器
件别名。
lstrcat (szMciString, ltoa (hwnd, temp, 10));
//将窗口句柄赋给字符串。
mciSendString ( szMciString, NULL, NULL,
NULL)); //发送命令字符串。
.....
capability device __id parameter[notify][wait] 用
于查询视频叠加器件的功能, 如能否进行视频冻结、解
冻, 是否支持图象的存盘等等。其中扩展参数 card
type 用于返回视频卡的类型号, 对于 VB SE100 而
言其类型号为 3。使用 capability 的相关源代码为:
.....
```

```
char temp[32]; //字符数组, 存放视频卡类型号。
mciSendString ("capability vb card type", temp,
16, NULL); //发送命令字符串。
```

.....

set device __id parameters 用于设置音频通道及视频特性如对比度、饱和度、亮度等等。扩展参数 card type to number 用于设置视频卡的模式, 程序中我们将 number 设为命令 capability 返回的类型号。相关代码为:

.....

```
int nCardType;
//初始化整型的视频卡类型号。
nCardType = atoi(temp);
//将 capability 命令返回的类型号转为整型。
if (nCardType == 3)
mciSendString ("set vb card type to 3", NULL,
NULL, NULL); //将视频卡设为 VB SE100 模式。
```

.....

save device __id filename[parameters][notify][wait] 用于将视频缓冲区的内容存为磁盘文件。filename 为用户定义的文件名。有关参数的定义如下: at rectangle 用以指定目标矩形区域, rectangle 的格式为: X、Y、W、H、X、Y 定义了矩形的左顶点, W、H 分别定义矩形的宽与高。

type file __type 用于定义存盘图象的类型。file __type 的取值为整数值 0—25, 用户可根据需要将文件存为 BMP、TGA、TIF、MMP、PCX、GIF 和 JPG 格式。为了获得实时压缩效果, 我们可以用 JPG 格式来保存文件。由于 JPG 格式的图象文件采用了 JPEG (Joint Picture Expert Group) 标准, 所以可以在很高的压缩比下取得较好的图象质量。关于 JPG 文件的扩展参数定义如下:

jpegopt rectangle 以矩形的定义法定义 4 个整数 Y C S x。Y 表示亮度值, 取值范围 (0—100, 缺省值 0); C 为色差值, 取值范围 (0—100, 缺省值为 0); S 为下采样率取值范围 (0—2, 缺省值为 0)。0 对应于 4:4:4; 1 对应于 4:2:2; 2 对应于 4:1:1。x 表示该项无效。

3.4 实现图象的连续捕捉与实时压缩

在这里, 我们需要用到定时的方法, 以使系统能够周期性地捕捉图象, 压缩后进行存盘。采用多媒体计时器函数 timeGetTime 可用较小的系统开销, 直接返回系统时间。应用程序的部分代码如下所示:

.....

```
char * filename; //初始化文件名。
DWORD time0, time, tdelay; //初始化系统
时间及时延变量。
time0 = timeGetTime(); //获取系统时间。
for (i = 0; i <= 99; i++)
```

```

{
do{time=timeGetTime();
tdelay=time-time0;
}while(tdelay<250); //系统定时。
itoa(i,filename,10); //构造文件名
strcat(filename,"jpg"); //字符串。
time0=timeGetTime(); //获取系统时间。
wsprintf(szMciString,"save vb %s type 24 at 0 0
176 144//构造字符串,文件类型为JPG(24),图象
jpegopt10 15 10", (LPSTR)filename); //尺寸为176象素
×144线,亮度值取10,色差值取15,下采样率为4:2:2(1)。
mciSendString(szMciString, NULL, NULL, NULL); //发送
命令字符串。
}
.....

```

4 试验与结论

我们在 Windows 3.1 下利用 Borlandc++4.0 编程环境调试并通过该图象处理软件。试验中我们在 25 秒左右的时间里捕捉了 100 幅 QCIF 格式(176 象素×

144 线)的图象,图象源格式为 RGB24bit 彩色图象,信息量为 76,032 字节。经过处理后每幅图象 JPG 文件的大小仅为 2,400 字节左右,压缩率为 31 倍左右,图象的主观质量较好,实现了图象的连续捕捉与实时压缩,便于图象信息的存储与传输,适合于各种工业控制方面的应用,也适用于通信系统的信息交换与存储。

由此可见,VB SE100 的用户完全可以利用 WindowsMCI,在较高的层次上快捷地开发出自己的图象处理软件。对于其他图象卡的用户而言,只需了解本身的一些具体参数和性能,一样可以利用 WindowsMCI 进行应用程序的开发。

参考文献

- 1 陈向群等译.多媒体开发指南.北京:清华大学出版社,1995.7
- 2 周长发编著.多媒体计算机技术开发与应用.北京:电子工业出版社,1995.7
- 3 Video Blaster SE100 用户指南;1995.9

系统级故障诊断方法研究

杨金民 张大方

(湖南大学计算机系 长沙 410082)

摘 要 本文全面研究了系统级故障诊断中的集中式诊断方法,概率诊断方法和分布式诊断方法,并归纳了系统级故障诊断方法的性能评价指标。

关键词 系统级故障诊断 集中式诊断 概率诊断 分布式诊断 性能评价指标

1 引言

随着多机系统的日益发展,尤其是大规模多机系统和网络系统的发展,对这些系统的可靠性要求在不断提高。研究快速识别多机系统中故障处理机的实用技术已成为计算机科学中的一个重要分支。自 1967 年 preparata^[1]等提出系统级故障诊断的思想以来,系统级诊断方法已被广泛研究,并已取得了一些令人瞩目的成果。但系统级诊断是一门复杂的科学,仍有许多问题未解决,还有待研究。本文将对系统级诊断领域中已取得的成果作出总结、分类和对比。本文的第 2 部分归纳了系统级诊断方法的性能评价指标,第 3.4.5 部分分别研究了集中式诊断方法,概括诊断方法和分布式诊断方法。

下面给出本文要用到的一些符号和定义。一个多机系统由 n 个结点(处理机单元)组成。测试用有向图 $G(V, E)$ 表示, $V = \{u_1, u_2, \dots, u_n\}$ 表示处理机集, $E = \{(u_i, u_j) | 1 \leq i, j \leq n\}$, 边 (u_i, u_j) 表示 u_i 测试 u_j 。测试结果用 a_{ij} 表示, $a_{ij} = 0$ 表示 u_i 认为 u_j 为好机, $a_{ij} = 1$ 表示 u_i

认为 u_j 为坏机。系统的所有测试结果的集合叫做系统的一个症候 S 。系统中每个处理机的状态用 $\{0, 1\}$ 表示, 0 表示为好机, 1 表示为故障机。系统中测试结点 u_i 的结点集用 $\Gamma^{-1}(u_i)$ 表示, $\Gamma^{-1}(u_i) = \Gamma_0^{-1}(u_i) \cup \Gamma_1^{-1}(u_i)$, $\Gamma_1^{-1}(u_i)$ 为测试结点 u_i 其测试结果为 1 的结点集。 $d(u_i) = |\Gamma^{-1}(u_i)|$, 即结点的入度, 对系统进行诊断, 如果没有正常结点被错诊为故障结点, 则认为该诊断是正确的。如果诊断中能把所有的故障结点都诊断出来, 则认为该诊断是完全的。一个诊断方法的精度是指它所执行的诊断中, 完全正确诊断所占的百分比。

2 系统级诊断方法的性能评价指标

评价一个系统级诊断方法的性能, 可以从下面 7 个方面考虑:

1) 能诊断故障种类和能诊断的故障数目。故障一般分永久性故障、间歇性故障和线路故障。

2) 所要求的测试数目。即进行一次完整的诊断要求进行的测试次数。测试次数越多, 系统开销就越大。

3) 对测试图的结构要求。每一种诊断方法都要求测试图要满足一定的条件。要求越多、越严, 诊断方法的适应性就越差。

4) 诊断算法的时间复杂性。计算量越大, 占用机器的时间就越长, 灵敏性也就越低。

5) 诊断信息传输量。系统要收到测试信息后,才能作出诊断,诊断信息传输量越多,系统的开销也就越大。

6) 诊断算法的精度。即执行的诊断中,完全正确的诊断所占的百分比。

7) 诊断延迟。诊断延迟是指从测试到故障发生到故障被诊断出来的这段时间大小。诊断延迟越长,系统在故障态下工作的时间也就加长,系统的可靠性就会降低。诊断延迟是一个综合指标,它与许多因素有关,比如:诊断信息传输量,算法的时间复杂性,测试策略等。

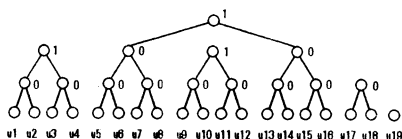


图 1 自适应测试过程图

3 集中式诊断方法

1967年 preparata 等提出了系统级诊断的图论模型。在 PMC 模型中,假设正常机所作的测试,其结果是可靠的,反映了被测机的状态,而故障机所作的测试,其结果呈随意性,不可靠。preparata 还提出了 t -故障一步诊断的算法,并推证了测试图满足 t -故障一步诊断的充分必要条件:系统中每个处理机至少要被 t 个别的处理机测试,而且任意两个处理机之间还不能相互测试。 t -故障一步诊断所要求的测试数 $\geq nt$ 。算法的时间复杂性为 $O(nt)$ 。这种算法要求要有一个无故障的中心机来收集各处理机的测试信息并执行诊断,而且还限定系统中故障机数不能超过 t 。针对对测试图的要求太严这一问题,后来又提出了 t -故障顺序诊断策略,即要求每一次至少诊断出一个故障机,对诊断出来的故障机更换后,再进行下一轮诊断,直到所有故障机都被诊断出来。对 t -故障顺序诊断,测试图的构造方法已被深入研究,并已有一些成果。 t -故障顺序诊断要求系统中配有备用机。

针对系统中故障机数目 $\leq t$ 这个限定, Vaidya 和 pradhan 于 1994 年提出了 t -故障定位, u -故障探测的诊断算法^[2], 当系统中故障机数 $\leq t$ 时, 这种算法能对故障机定位, 当系统中故障机数为 $t+1$ 到 $n-t-1$ 之间时, 它能探测到。他们给出了测试图满足这种算法的充要条件。

为了减少诊断中的测试次数, Hakimi 等人于 1984 年提出了自适应性诊断策略^[3]。其思想是分步进行测试和诊断, 后一步的测试要根据前一步的测试和诊断结果来确定, 采用这种策略, 平均测试次数要比一步测试诊断所要求的测试次数少。1995 年, 蔡兵等提出了

一种较好的自适应诊断算法^[4]。其自适应测试过程可用一簇二叉树表示,例如一个含 19 台处理机的系统,其自适应测试过程之一为图 1 所示。

这种策略采用 Chwa-Hakimi 比较测试模型。图中叶结点为处理机。叶结点的上层结点在它的两棵子树中分别任选一个处理机进行比较测试所得的测试结果。逐层往上发展,当测试结果为 1 时,便停止。最后得到 2 类二叉树簇,第一类是根为 0 的二叉树簇,如图中的 (u17,u18),(u19),第二类是根为 1 的二叉树簇,如图中的 (u1,u2,u3,u4),(u5,u6,u7,u8,u13,u14,u15,u16),(u9,u10,u11,u12)。依据这种测试策略,总的测试次数 $\leq n+t-2$ 。采用并行测试策略时,所需的测试回合次数 $\leq \log_2 n$ 。文献[4]还给了基于这种测试策略的诊断算法。

上面介绍的这些方法也被称作确定性诊断方法，它是从症候 S 中唯一地确定出故障集来。

4 概率诊断方法

确定性诊断方法对测试图要求很严,而实际中许多系统很难满足这种要求。为了克服这一问题,后来人们提出了概率诊断方法⁽⁵⁾。概率诊断方法不象确定性诊断方法那样从症候中唯一地识别出故障集来,而是只试图正确地诊断出高概率的故障结点,不要求限定性假设测试图的结构。概率诊断方法中,用概率参数来描述故障结点和正常结点的特性,和概率参数如表 1 所示。

表 1 概率参数

f_{si}	f_{sj}	a_{ij}	$P(a_{ij} f_{si}, f_{sj})$
0	0	0	1
0	0	1	0
0	1	1	P_{ij}
0	1	0	$1 - P_{ij}$
1	0	0	Y_{ij}
1	0	1	$1 - Y_{ij}$
1	1	1	S_{ij}
1	1	0	$1 - S_{ij}$

表中 $f_{i,j}$ 表示 u_i 的状态, $f_{i,j}=0$ 表示 u_i 为好机, $f_{i,j}=1$ 表示 u_i 为故障机, 事先认为 u_i 为故障机的概率为 p_i , $p(a_{ij} | f_{i,j}, f_{j,j})$ 是已知 $f_{i,j}, f_{j,j}$ 的条件下, a_{ij} 发生的概率。 p_{ij} 为好机测故障机, 测试结果为 1 的概率。 p_{ji} 描述了间歇性故障的特征。间歇性故障占真实系统中的故障的很大一部分, 其特点是处理机呈现出时好时坏的情形。当 $p_{ji}=1$ 时表示故障为永久性故障, 好机对它的任何一次测试, 故障都能测试出来。前面介绍的确定性诊断方法都作了 $p_{ji}=1$ 的假设。对间歇性故障, 可以通过增加测试次数来提高好机对故障机的发现率, 例如, 进行 n 次测试, 至少出现一次“1”的测试结果的概率为:

$$P = 1 - (1 - P_{ij})^n$$

可解得： $n = \log(1-P) / \log(1-P_{ij})$

当 $P_{ij}=0.5$ 时,连续测 7 次,故障发现率能从 0.5 提高到 0.992。 r_{ij}, s_{ij} 分别表示故障机能正确测出好机和故障机的概率。概率诊断方法对测试图没有严格的限定

要求。概率诊断方法依据在诊断故障结点时所用到的测试的信息量可分为三类。A类为利用整个系统测试信息,这类方法有 Blount 于 1977 年提出的一个算法^[5]。这类算法的效率不高,因为计算的时间复杂性很高。B类为利用本地测试信息($|\Gamma^{-1}(u_i)|$, $|\Gamma_0^{-1}(u_i)|$ 和前面步骤已诊断出的故障结点)来诊断结点 u_i 的状态。这类方法采用分步诊断的策略,每次只诊断一个故障结点。C类为只利用本地信息($|\Gamma^{-1}(u_i)|$, $|\Gamma_0^{-1}(u_i)|$)来确定每一个处理机 u_i 的状态 $\{0,1\}$, Blough 于 1988 年提出的算法属于这一类。

就 B 类和 C 类算法, Lee 和 Shin 在 1993 年提出了一种局部最优的诊断算法^[6]。每一种概率诊断算法必须保证在保持测试图结构不变的条件下,当系统的结点数增加到无限大时,算法的诊断精度趋近于 1。

概率诊断方法也可以采用顺序诊断的策略。Blough 和 Pelc 在 1993 年提出了一种顺序诊断的算法^[7],其策略是分二步进行诊断。第一步是让系统中各处理机分链后进行测试,得到一个核心正常集,第二步是利用核心正常集依次对其它机进行测试、诊断,当诊断为好机时,把它纳入核心正常集,当为故障机时则进行更换,直到系统中处理机全部为好机。

评价概率诊断方法的一个重要指标是诊断精度。一个诊断算法的诊断精度达到最高级,该诊断算法便认为是最优的。诊断精度可用 $P_c(\text{Algorithm})$ 表示, P 表示概率, G 表示测试图, Algorithm 表示算法。它与算法和测试图有关。获得最优诊断是一个 NP-难度问题。

上面提到的各种概率诊断方法的性能指标如表 2 所示。

表 2 各种概率诊断方法性能对照表

诊断方法	计算复杂性	测试次数	优化	诊断概率参数个数
Blount(1977)	$O(2^N - 1)E$	$O(N \log N)$	是	全部
Bough(1988)	$O(E)$	$O(N \log N)$	否	2
Lee & Shin(1993)	$O(N^2)$	$O(N \log N)$	局部	4
Blough & Pelc(1993)	$O(N \log N)$	$O(N \log N)$	否	2

概率诊断方法比确定性诊断方法朝实际应用又靠近了一步。现在多处理机系统正在朝着大规模的方向发展,系统中处理机越来越多。而概率诊断方法的诊断精度随系统中处理机数目增多而提高,因此概率诊断对大规模系统更适应。概率诊断方法虽对测试图结构无限制,但概率诊断方法通常要求用概率参数来描述正常结点和故障结点的特性以及用它来估计所作诊断的质量。在实际系统中获取这些概率参数是一个尚未解决的问题。

5 分布式诊断方法

上面所概述的诊断方法都假定了测试结果能可靠地传输到一个正常的中心机上,由中心机对系统中各处理作出诊断。这种假设跟实际情况不完全相符,测试信息全部可靠地传输给中心机,在实际中较难实现。因为测试信息传给中心机,中间要经过别的处理机的转

发,当中间转发机为故障机时,测试信息在传输中就会失真。另外,系统中要在一个不出故障的中心机,这一点也很难办到。针对这一问题,八十年代初, Kuhl 和 Reddy 提出了分布式自诊断思想^[8]。其思想是在分布式系统中,每个处理机分别独立地对系统中的处理机作出诊断。不象集中式诊断,它不需要一个中心机。典型的分布式算法有 NEW-SELF^[9], EVENT-SELF^[10], DSD^[11] 算法, NEW-SELF 是一个基于 PMC 模型的非自适应的 t -故障一步诊断算法,它采用固定的测试图来完成测试和诊断。它要求每一个结点至少被 $t+1$ 个别的结点测试,以保证系统中的好机能联成一条链,以及任一故障机都能被好机测试到。系统中好机作的诊断是一致正确的。EVENT-SELF 算法是 NEW-SELF 的改进型。DSD 算法是一个基于 PMC 模型的自适应 t -故障一步诊断算法。它采用自适应方法进行测试。系统中的处理机按序编号。每一处理机对它后面的处理机依次进行测试,直至它测试到一个它认为是正常的处理机时,它便停止往后测试。这种测试策略使得系统中的好机联成一个环,整个系统的测试信息都通过这个环传到每一个正常机上,最后由每个正常机作出一致的诊断。

NEW-SELF 算法和 DSD 算法的性能指标如表 3 所示。

表 3 分布式诊断方法性能对照表

算法	测试数	算法时间复杂性	传输信息量	能诊断的故障数
NEW-SELF	$N(t+1)$	$O(N(t+1))$	$N^2(t+1)^2$	$\leq t$
DSD	N	$O(N)$	$\Delta \cdot N$	$\leq N-2$

DSD 算法的诊断延迟比 NEW-SELF 小,就测试数目和信息传输量以及算法时间复杂性方面, DSD 算法是一种较优的算法。DSD 算法还是该领域中首先应用于实际系统诊断的算法。DSD 算法和 NEW-SELF 算法都能诊断间歇性故障, NEW-SELF 在多总线的条件下能诊断线路故障, DSD 算法不能诊断线路故障。

从系统级诊断方法的发展来看,面向实际系统提出性能更好的算法仍然是系统级诊断研究的重要方面,自适应分布式概率诊断有着广阔的前景。优良的诊断算法,不仅要能诊断永久性故障、间歇性故障,还应该把线路故障也考虑进去。

6 结束语

系统级故障诊断方法经过 20 年的发展,已取得了大量的研究成果,但仍有许多问题有待进一步研究。我们认为主要研究的问题可归纳为:

- 1) 综合现有各种算法,提出性能更优的新算法。
- 2) 结合某一具体算法,模拟实际系统的诊断过程,探讨在整个系统中如何来组织测试、诊断,使得系统运行在最优状态。这是一项很有意义的工作。
- 3) 系统级概率诊断中的概率参数的统计和分析。
- 4) 计算机网络系统正在迅猛发展,结合网络的特

点,把系统级诊断的成果应用到网络诊断中,对网络的可靠运行将会起到保证作用。

5)构造软件仿真系统,对各种诊断方法进行比较研究和实验研究。

参考文献

- 1 F. P. preparata G. Metze R. T. Chien, On The Connection Assignment Problem Of Diagnosable Systems, IEEE Trans. Electronic Computer, 1967, 16, 848~854
- 2 N. H. Vaidya D. K. Pradhan, Safe System Level Dignosis, IEEE Trans. Comput, 1994, 43(3), 367~370
- 3 S. L. Hakimi k. Nakajima, On Adaptive System Diagnosis, IEEE Trans. Comput. 1984, 3(3), 234~240
- 4 蔡兵, 杨晓帆, 陈廷槐. 对称比较模型下的系统级故障诊断, 计算机学报, 1995, 18(11), 858~865
- 5 S. Lee K. G. Shin, The Probabilistic Diagnosis Of Multiprocessor System, ACM Computing Surveys, 1994, 26(1), 121~139
- 6 S. Lee K. G. Shin, Optimal And Efficient Probabilistic Distributed Diagnosis Schemes, IEEE

Trans. Comput. 1993, 42(7), 882~886

- 7 D. M. Blough A. Pelc, Diagnosis And Repair In Multiprocessor System, IEEE Trans. Comput, 1993, 42(2), 205~217
- 8 J. G. Kuhl S. M. Reddy, Distributed Fault-Tolerance For Large Multiprocessor System, In Proc. 7th Annu. Symp. comput. Architecture, IEEE, 1980, 23~30
- 9 S. H. Hoseini J. G. Kuhl & S. M. Reddy, A Diagnosis Algorithm For Distributed Computing System With Dynamic Failure And Repair, IEEE Trans. Comput, 1984, 33, 223~233
- 10 R. P. Bianchini K. Goodwin & D. S. Nydick, Practical Application And Implementation Of Distributed System-Level Diagnosis Theory, In Proc. Twentieth Int. Symp. Fault. Tolerance Comput, IEEE, 1990, 332~339
- 11 R. P. Bianchini R. W. Buskens, Implementation Of On-Line Distributed System-Level Diagnosis Theory, IEEE Trans. Comput, 1992, 41(3), 616~625

如何防止公共机房中 CMOS 口令被篡改所造成的“死机”现象

李翔枫

(西南财经大学信息系 成都 610071)

在公共机房,尤其是高校的机房,一台微机通常被许多学生轮流使用,每天一台机器平均要轮流使用 5 人次以上。学生启动机器时常常遇到屏幕询问用户口令的情况。使用者往往不知具体的口令,于是机器不得不终止启动,造成“死机”而无法使用。那么为什么会出现这种情况?又该如何有效防止它的产生呢?

首先要弄清出现询问口令的原因。在 286AT 档次以上的机器中都装有一块集成片 CMOS,其用途是记录系统日期和时间,内存的型号容量,硬盘的类型容量,软盘的规格等等。包括设置系统口令。在整个微机运行过程中 CMOS 提供了许多重要的参数。它们一旦被错误地设置或丢失,则会导致机器不能正常运行乃至“死机”,前面所述的“死机”现象就是其中较为典型的例子。在 CMOS 中有一项关于机器口令设置的参数。它有三种选择:

. Disable 不执行检查口令程序

. Setup 只要求进入 CMOS 设置程序时才检查口令程序。

. Always 开机启动及进入 CMOS 设置程序时均执行检查口令程序

第一种选择与口令无关(往往机房中的机器处于这种状态或根本不设置口令)。

第二种选择,出现询问口令时如果不知道口令,采用的解决方法是打开机箱对 CMOS 芯片进行放电处理,以消除包括口令在内的所有 CMOS 信息,然后重启机器设置 CMOS 各项参数。另一种方法对于熟悉使用计算机的用户是轻而易举之事。即先不进入设置 CMOS 程序,直接启动 DOS,然后通过 70H, 71H 端口的读操作,读出加密后的密码。解密后获得口令。

第三种选择最让人头疼,因为机器一启动就在屏幕上出现询问口令的信息。使用者不知口令或遗忘口令的话就无法继续使机器运行,更谈不上使用计算机的软、硬件资源,最后不得不采用 CMOS 放电的方法来解决。

在实际教学管理机房中,通常口令参数为 Disable 或根本不设口令。然而有些使用机器技巧较高且又喜欢恶作剧的学生,他们或是直接设置口令,或是先读出当前口令,然后重新启动机器进入设置 CMOS 的状

态,再取消当前口令,重新设置新口令。不管是哪种方式设置口令后他们都将口令参数改为 Always。一旦其他学生用机,刚开机就遇到要他回答口令(Passdword)的屏幕显示,回答不出口令,则根本无法使用这台机器。对此现象教师也奈何不了。最后由机房管理员进行 CMOS 放电处理。

如何有效防止这类原因造成的“死机”,保护 CMOS 中的口令及其它参数信息不被篡改呢?目前较为常见的做法是在公共机房微机的 BIOS 扩展槽中插入一块保护卡(硬卡)。实质上它是一块存有固化了 CMOS 正确信息和相关程序的 EPROM 芯片。其主要功能是每次启动之前先将 EPROM 中的 CMOS 的正确参数向 CMOS 中写一次,用来消除计算机病毒或人为因素造成的 CMOS 信息的破坏。不过这种硬卡方式有一个很大的缺陷。因为虽然 BIOS 提供了用于执行扩展槽中硬卡上程序的接口,但接口是在 BIOS 的 CMOS 设置程序之后。当口令参数为 Always 方式时,使用者只有正确回答口令后,BIOS 才会执行所接硬卡上的功能,否则启动机器的过程就到此为止。哪里还能继续执行硬卡上保护 CMOS 的程序,其结果“死机”现象仍无法避免。笔者在带学生上机实习中曾遇到过此现象。显然这种方法是不可靠的,最后还得要作 CMOS 放电处理。

因此有必要对硬卡方案进行改进或者采用别的措施。改进硬卡方案的设想是:鉴于硬卡方案的缺陷主要是在于执行硬卡功能滞后于系统询问口令,若解决了这个“瓶颈”问题。就完全避免了因口令问题所造成的“死机”现象。那么,必须先改变硬件线路,待机器启动时,CS:IP 不指向 FFFF:0000 而指向硬卡的入口,待执行完硬卡中的程序后,再跳到 FFFF:0000 处检查

内存,执行 CMOS 设置与检查程序。只要经过这样的改进,相信是可避免出现询问口令的情况,从而达到防止产生“死机”的现象。

不难发现这其实是在修改 BIOS,不过这种改进硬卡的方案也并非尽善尽美。它一方面可防止用户篡改 CMOS 中的口令及其它信息避免因口令问题所造成的“死机”,另一方面却也失去了设置 CMOS 信息的方便性。因为所有的 CMOS 信息被固化在 EPROM 中。若个别用户想修改其中某些参数时,势必对 EPROM 进行重写,这项工作也不是一件轻松的事。

不过针对高校公用机房中因学生篡改口令而造成“死机”无法使用机器的情况,选择安装改进硬卡方案显然比 CMOS 作放电处理安全方便实用得多。

由于 CMOS 设计者在口令设置方面考虑不周到造成普通用户有权篡改口令的结果。使得对公共机房的机器管理带来很大麻烦,也大大妨碍了 CMOS 发挥其正常作用。要想进一步解决这个问题不妨同时从软、硬件着手改变 BIOS 的流程。

硬件方面:在机器内安装一块存有该机器识别码的 ROM 芯片。仅限于机房管理员知道。并保证该识别码的存储和识别由硬件完成。

软件方面:设置检查口令程序。若使用者想修改口令,则系统先要检查当前口令,回答正确后再由硬件检查机器识别码是否正确。由于普通用户根本不知其识别码,即使知道口令也无法修改口令。

这样,从软、硬件两方面同时把关,就杜绝了用户(上机学生)篡改口令的可能性,因为他们无法同时知道正确的口令和机器识别码。这种方案不失为防止口令被篡改而造成“死机”的一种较完善的设想。

DEBUG 程序在微机软故障 诊断和排除中的应用

杨胜勇 李劲松

(四川省经济管理干部学院计算中心 成都 610041)

肖 滨

(中国工程物理研究院久远辐射技术总公司 绵阳 621000)

摘 要 计算机软故障的急剧上升,使其成为计算机故障的主流。本文讨论了如何使用 DEBUG 程序进行微机常见软故障的诊断与排除,着重探讨了对系统配置类、磁盘文件类及计算机病毒等软故障的诊断、保护和排除措施。

关键词 DEBUG 程序 微机软故障 诊断 排除

微型计算机系统故障可分为两大类：一类是硬件故障，它是由计算机系统硬件的机械或电子系统产生的故障。另一类是软件故障，它是由于计算机系统的软件被破坏，或者配置参数被修改等因素，导致系统不能正常工作而产生的故障。

随着半导体工业的发展，计算机的硬件质量不断提高，集成度也越来越高。微型计算机系统硬件的平均无故障时间也大大增长了，硬件故障在微型计算机系统故障中所占的比例大幅度下降，实践表明，大部份微型计算机的硬件平均无故障时间可达一年以上，即使硬件出现故障，更换相应的部件，其费用也不是很高，加上计算机厂商完善的售后服务网点，因此硬件故障的维修对用户来说已经不是很大的负担。

软件故障是微机用户经常遇到的一种常见系统故障，尽管计算机硬部件是完好的，但由于软件故障，仍然使用户不能正常使用计算机，从而影响正常工作。据统计，这类软件故障在整个微机故障现象中所占的比例已达 70% 以上，而且仍在继续上升，因此，如何诊断和排除微机软件故障已经显得越来越重要。

诊断和排除微机系统软件故障的方法较多，可以采用通用的工具如 PC-TOOLS、NORTON 等，也可以采用随机的诊断软件（在原装机中居多），这些软件具有功能强大，使用方便等特点，但这些工具软件，或者系统本身很大，其推广受到限制，或者由于是随机软件，其通用性受到影响。而 DEBUG.COM（以下简称 DEBUG）是 DOS 系统提供的一个实用程序，在普通 DOS 系统的微机上均能找到，DEBUG 的功能很强，即可以对目标程序进行动态和静态调试、分析，也可以对系统进行软件维护和软故障的修复。因此，DEBUG 以其短小、简捷、直接、方便的特点，在开发、管理和维护计算机系统（包括操作系统、磁盘文件系统）方面，仍是一个常用且有效的工具。本文将讨论微机常见软故障如何用 DEBUG 来进行检测和排除。

DEBUG 的传统使用方法人们已经熟悉，按照传统的使用方法，首先在 DOS 提示符运行 DEBUG 程序，然后一条一条地执行 DEBUG 命令。这种方法当然可行，但是其重复性差，历史命令记忆量大，结果分析困难，不便于一般用户使用。为了避免这种缺陷，以充分发挥 DEBUG 程序在微机软件故障诊断和排除中的作用，笔者利用了 DOS 的重定向功能，解决了这一问题。其具体使用方法如下：首先使用一文本编辑器将需要执行的 DEBUG 命令，严格按照正确的顺序一行一行地录入，然后在 DOS 提示符下运行 DEBUG 程序，其输入和输出重定向到文件。例如执行 c:\>debug<test>test.dat，其中 test 为 DEBUG 命令的文本文件，test.dat 为命令的执行结果。这种方法在我们的研究过程中行之有效，为使用 DEBUG 作为工具，研究微机软件故障的诊断和排除提供了极大的方便。下面我们对

于较长的 DEBUG 命令序列均给出的是这种文本文件形式，在实际应用中，读者只需使用一文本编辑器，如 DOS 中的 EDIT、EDLIN 等，将其录入计算机，形成一文本文件，然后在 DOS 提示符下，执行 DEBUG 命令，并以该文本文件作为其输入文件，其输出也可重定向到一个文件中。

1 微机系统参数设置故障的处理

1.1 CMOS 数据的备份与恢复方法

CMOS 参数是微机系统的配置参数，当 CMOS 参数与机器的配置参数不匹配时，机器不能正常启动。尽管利用 BIOS 中的 CMOS 设置程序，可恢复已破坏了的 CMOS 参数，但其前提条件是用户必须知道正确的系统配置参数，当用户购买机器时间太长，配置参数可能忘记，另一方面，即使对于刚购置的机器，由于配置参数的复杂性，一时也难以掌握，一旦造成 CMOS 数据丢失，将很难完全恢复正常使用，因此，有必要在购买机器验收后，对该机器的配置值及时备份，以备系统恢复之用。

众所周知，CMOS 分为标准 CMOS 和扩充 CMOS，它们均不占用内存地址。可通过 I/O 地址的 70H 和 71H 口来访问 CMOS 的各个单元，70H 为索引地址端口，71H 为数据端口，标准 CMOS 有 64 个字节，编号为 00-3FH，扩充 CMOS 也有 64 个字节，编号为 40-7FH，一般 286、386 机器的 BIOS 仅有 64 个字节，而 486、586 机器一般为 128 个字节。

下面是使用 DEBUG 程序进行 CMOS 数据的备份与恢复的方法：

(1) 将 CMOS 数据以文件 CMOS.DAT 存于 A:

```

盘上
A100
MOV BX,300
MOV CL,00
MOV AL,CL
OUT 70,AL
IN AL,71
MOV [BX],AL
INC CL
INC BX
CMP CL,80
JNZ 105
INT 20

```

(空一行)

```

G=100
N A:\CMOS.DAT
R BX
0
R CX
80
W 300

```

Q

(2)将 CMOS 的备份文件 A:\CMOS.DAT 恢复

N A:\CMOS.DAT

L 300

A 100

MOV BX,300

MOV CL,00

MOV AL,CL

OUT 70,AL

MOV AL,[BX]

OUT 71,AL

INC CL

INC BX

CMP CL,80

JNZ 105

INT 20

(空一行)

G=100

Q

1.2 CMOS 口令的解除

当用户设置了 CMOS 口令,但忘记了口令,或某些病毒的作用,使 CMOS 具有口令,此时,如果 CMOS 的参数与系统本身的配置不一致,而系统又不能启动,则此时可短接主板上的 CMOS CLEAR 跳线或取下电池的一端,让 CMOS 放电,如果能从 A:盘或 C:盘启动,则只需用 DEBUG 即可处理,其 DEBUG 命令的文本文件内容如下:

O 70 10

O 71 00

Q

运行完 DEBUG 命令之后,重新启动机器,进入 CMOS 设置程序,此时口令清除,即可重新设置。

1.3 硬盘物理参数的测试

CMOS 中的硬盘参数是一类很重要的参数,用户在系统中更换、安装新的硬盘或者 CMOS 中硬盘参数的丢失,必须在 CMOS 中设置其相应的参数。

我们知道,硬盘参数有逻辑参数和物理参数之分,CMOS 中所设置的参数是硬盘的逻辑参数,当硬盘驱动程序在访问硬盘时,向硬盘控制器提供欲访问的硬盘逻辑地址,硬盘控制器将硬盘逻辑地址转换成物理地址,并完成相应的操作,因此在安装一个新的 IDE 硬盘时,即使选择了错误的 BIOS 参数,容量只要在物理容量的范围之内,该硬盘同样可以使用,所以,对于一个硬盘,了解其物理的柱面数、磁头数和扇区数以及物理容量显得很重要。

为此,下面我们使用 DEBUG 程序通过读取硬盘的 BIOS 中的参数,来获取该硬盘的物理参数:

A 100

MOV DX,1f6

MOV AL,A0

OUT DX,AL

MOV DX,1F7

MOV AL,EC

OUT DX,AL

IN AL,DX

CMP AL,58

JNZ 10C

CLD

MOV BX,300

MOV CL,0

MOV DX,1f0

IN AX,DX

MOV [BX],AX

INC BX

INC BX

INC CL

CMP CL,80

JNZ 11A

INT 20

(空一行)

G=100

D 300 L 160

在 0302,0303 存放的是该硬盘的物理柱面数(高位在前,低位在后,下同),在 0306,0307 存放的是该硬盘的物理磁头数,在 030c,030d 存放的是该硬盘的每磁道扇区数,硬盘的物理容量可以由这三个参数计算出来,计算公式为:硬盘的物理容量=物理柱面数×物理磁头数×每磁道扇区数×512B,另外我们也可以得到该硬盘的产品序列号和产品名称。

下面我们给出一个例子,从下面的例子我们可以看出,该硬盘的柱面数为 0383H,即 899 个柱面,磁头数为 0FH,即 15 个,每磁道扇区数为 003EH,即 62,该硬盘的物理容量为 428,067,840B,该硬盘的产品序列号为 RA684131,产品名称为 ST3491A。

```
1A14:0300 5A 04 83 03 00 00 0F 00-90 8D 48 02 3E 00
00 00 Z.....H>...
```

```
1A14:0310 00 00 00 00 20 20 20 20-20 20 20 20 20 20
30 30 .... 00
```

```
1A14:0320 41 52 38 36 31 34 31 33-03 00 F0 00 10 00
37 30 AR861413.....70
```

```
1A14:0330 30 2E 2E 37 31 30 54 53-34 33 31 39 20 41
20 20 0.. 710TS4319A
```

```
1A14:0340 20 20 20 20 20 20 20 20-20 20 20 20 20 20
20
```

.....

2 磁盘文件系统故障的处理

目前计算机信息的存储媒体大多是磁盘,包括硬盘和软盘。由于磁盘可任意次进行读写,磁盘文件系统的故障就不可避免,例如:由于操作的失误、软件的故障或病毒的侵蚀,都会使系统不能正常运行,如何保护好磁盘系统,掌握诊断和排除磁盘文件系统的故障显得非常重要。

2.1 硬盘主引导记录的备份与恢复

无论是硬盘或是软盘其 0 面 0 道第一扇区都是一个举足轻重的扇区,软盘中的该区是 DOS 引导扇区,硬盘的该区是主引导记录和分区表信息。

若硬盘的主引导扇区被破坏,那么该硬盘一般不能正常启动或不能正常运行程序,因此对该扇区的备份就显得很重要。下面我们给出用 DEBUG 进行备份和恢复的方法。

(1)将硬盘的主引导扇区备份至 A:\MBOOT.DAT,

```
A 100
MOV AX,0201
MOV BX,300
MOV CX,0001
MOV DX,0080
INT 13
INT 20
```

(空一行)

```
G=100
R BX
0
R CX
200
N A:\MBOOT.DAT
W 300
Q
```

(2)将硬盘的主引导扇区的软拷贝 MBOOT.DAT 恢复:

```
N A:\MBOOT.DAT
L 300
A.100
MOV AX,0301
MOV BX,300
MOV CX,0001
MOV DX,0080
INT 13
INT 20
```

(空一行)

```
G=100
Q
```

2.2 磁盘引导扇区的备份与恢复

磁盘引导扇区指的是 DOS 引导扇区,可直接由 DEBUG 的简单命令即可实现备份与恢复,勿需调用中断。下面是用 DEBUG 命令将 C 盘的 DOS 引导扇区备份至 A 盘的 BOOT.DAT 文件:

```
c:\>debug
-1 100 2 0 1
-n A:BOOT.DAT
-r cx
CX 0000
,200
-W
```

将 C 盘的 DOS 引导记录的软拷贝 A:BOOT.DAT 恢复的 DEBUG 过程如下:

```
c:\>debug
-n A:BOOT.DAT
-1 100
-w 100 2 0 1
-q
```

如果我们发现软盘或硬盘的 DOS 引导扇区损坏,但又没有其备份,那么如何进行引导扇区的恢复呢?我们以硬盘的 DOS 引导扇区被损坏为例,可以将同版本的 DOS 格式化的软盘的 DOS 引导扇区的内容读出,稍加修改,即可作为硬盘的 DOS 引导扇区,具体方法如下(其它情况类似):

```
c:\debug
-1 100 0 0 1  插入同版本的 DOS 格式化的软盘到 A 驱,读
                BOOT 扇区内容至 100H 处
-1 300 2 0 1  读 C 盘的 BOOT 扇区内容至 300H 处
-m 303 33d 103  移 C 盘的 BPB 至 103,即将 A 盘的 BPB 换成 C
                盘的 BPB
-w 300 2 0 1  写回 C 盘的 DOS 引导扇区
```

3 微机病毒的判断与排除

病毒是近几年来出现的一种计算机社会现象,人们对病毒的看法,最初是恐惧,然后由于大量的清除病毒软件的出现,人们的恐惧感渐渐消失。但由于新病毒的不断涌现,人们又不得不随时提高警惕,因为病毒的侵蚀,不仅使你不能正常使用系统,而且可能造成大量重要数据的丢失。

微机的病毒的侵入,引起计算机的不能正常工作,这是一种特殊的计算机软件故障,这种故障最简单,最有效的方法是直接使用通用杀毒软件,如 KILL、KV200、CPAV 等,但这些软件只能清除已知病毒,对于病毒的变种和新的病毒就无能为力了,另外,当手上没有这些消毒软件时,又该如何处理呢?

为此,笔者从病毒的本质研究着手,使用 DEBUG 程序解决了病毒带来的许多困扰。

3.1 如何发现病毒

(1)检查内存

大部份病毒都要占据内存,因此检查内存是否完整,是发现病毒的一种简单的方法。

```
C:\>debug
```

```
-d 0,0413
```

413 和 414 两个单元的内容表示常规内存的容量,若小于 0280H(即 640K),则一般有病毒。有些病毒不驻留内存,此时只能通过其它方法发现病毒。

(2)检查系统引导扇区

如果有引导扇区型病毒,则系统引导扇区会被改变,可用 DEBUG 检查系统引导扇区。

```
a 100
mov ax,0201
mov bx,300
```

```
mov cx,0001
mov dx,0080
int 13
int 20
```

(空一行)

```
g=100
d 300180
```

●正常的引导扇区前两个字节为FA33H,若前两个字节的内容为EBH和1CH,则可判断主引导记录区已经感染“Genp”或者“Ripper”病毒。若为E9H和95H,则可判断已经感染“Touch”病毒。

●正常的引导扇区在00EB-00ECH单元的字节内容为0000,若其内容为0519H,则可断定为主引导区上已经感染了“新世纪”病毒。

●正常的引导扇区在018B开始的位置,有“Invalid partition table. Error loading operating system. Missing operating system”标志,如果没有,则肯定有引导区型病毒。

(3)检查系统文件

文件型病毒一般会感染可执行文件(EXE、COM),所以MS-DOS的系统文件COMMAND.COM也会被感染,因此只需将硬盘上的COMMAND.COM文件和没有病毒的DOS系统盘上的文件进行比较,如果不相同,则肯定有病毒。

(4)文件测试法

文件型病毒也可用文件测试法来发现,用DEBUG编一个小程序TV.COM

```
c:\>debug
-a 100
```

```
xxxx:0100 int 20
xxxx:0102
-n tv.com
-r cx
2
-w
-q
```

然后运行tv.com几次,用DIR查看tv.com的文件长度是否变大(正常的文件长度是2),如果变大,则肯定有文件型病毒。

3.2 病毒的清除

(1)系统引导区型

消除系统引导区型病毒,只需用DEBUG把干净的系统引导区写到硬盘上即可清除病毒。具体的操作过程参加前面引导扇区(包括主引导扇区和DOS引导扇区)的备份和恢复方法。

(2)文件型和交叉型病毒

文件型和交叉型病毒十分复杂,使用DEBUG程序难度较大,这里限于篇幅不作进一步讨论。不过由于大量的通用清病毒软件的出现以及它们不断的更新升级,采用通用杀毒软件,或许是最好的解决办法。

参考文献

1. 唐余余. 实用DOS技术, 四川大学出版社, 1989.
2. 张怀莲. IBM PC(INTEL 8086/8088)宏汇编语言程序设计, 天津大学出版社, 1988.
3. 李向宇, 郭薇. 计算机反病毒实用指南, 北京, 国防工业出版社 1992. 11
4. 谢深泉. 微机常见故障诊断与排除的软件技术, 清华大学出版社, 1995.

微型计算机开关电源疑难故障的检修

王维舟

张生金

(西安理工大学)

西安 710048)

摘 要 本文对微型计算机开关电源无输出及负载能力降低等典型故障进行了分析, 给出了解决此类故障的方法。

关键词 计算机 开关电源 脉冲 输出电压

1 微机电源的工作原理

微型计算机电源均为无工频变压器的脉宽调制变换型稳压电源。这种电源采用直接整流、高频变换和脉宽调制技术, 具有体积小、重量轻、效率高、输出电压保持时间长、输出不易过电压、性能稳定可靠等特点。它

的工作原理如图1所示。

由图中我们可知, PC类微机电源实际上是一个变换器, 它将市电整流成直流后通过变换型振荡器变成较高的矩型或近似正弦波电压, 再经过高频整流滤波环节变成低压直流电压供给电压负载。它通过控制电路对振荡器输出电压进行调制, 从而实现稳定直流输

出电压的目的。图中变换振荡器是整个电路的核心部分。从维修角度分析,这一部分亦是故障的集中表现的部分。

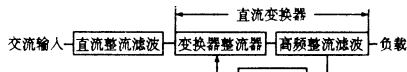


图1 微机开关电源原理图

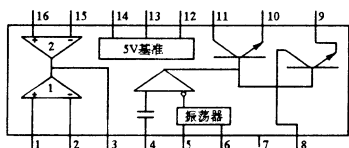


图2 PC-494内部结构框图

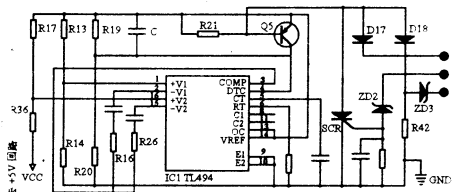


图3 控制部分原理图

2 几种疑难故障的电路分析

2.1 电源无电压输出

(1) 保护电路故障

打开电源外壳后,测的输入部分有 300V 的直流电压,自电源部分输出为 14.4V,控制器 MPC-494 的 4 脚电压为 1.5V,该控制器处于关断电源的状态。

在这类电源中,控制部分是由集成开关电源控制器 PC-494 及相关元件组成,PC-494 的内部框图如图 2 所示,它是由一个线性放大电路与数字脉冲电路合成在一起的。它的工作原理如下:定时元件接在 5、6 两脚使之产生一个三角波,这个三角波与放大器 1、放大器 2 送来的信号进行比较,并得到一个一定宽度的脉冲。该脉冲与死区电压产生的脉冲相配合,并经过数字电路控制后分别送到两个控制三极管的基极,以得到相互相差 180 度的宽度可调的脉冲,并以此来控制主变回路中振荡管的导通时间,进而调节输出电压的大小。表一列出在正常工作状态下的 PC-494 各脚的直流电位值。

表一 PC-494 各脚的直流电位值

脚号	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
电压值	2.5	2.5	3.6	0.25	6.0	2.5	6.0	0.25	0.25	15.4	15.4	15.7	14.5	5.5	3.6	3.6

控制部分原理如图 3 所示。

PC-494 的 1 脚、16 脚接 +5V 输出电压的反馈信号,以完成采样放大功能;2 脚接电阻 R13 和 R14,分压后的 2.5V 作为基准电压;4 脚接由 R19 的 R20 分压得到的 0.25V 死区时间电压;3 脚作为消振反馈电路。

在正常工作时 Q5 不导通,所以 4 脚的电压为 0.25V,一旦保护回路动作,则将导致 Q5 导通,这时 4 脚

的电压就不是 0.25V。图中 A 点接 12V 回路中的过流信号,即 12V 有过流时 D17 导通,这就使得 Q5 导通。B 点接 +5V 回路中的过压信号,当 +5V 有过压时 ED2 工作,则 SCR1 可控硅导通,使得 Q5 导通。

根据上述分析,此故障是保护电路工作,致使是电源无电压输出。重点检查保护电路中的二极管、稳压管,更换损坏元件,故障排除。

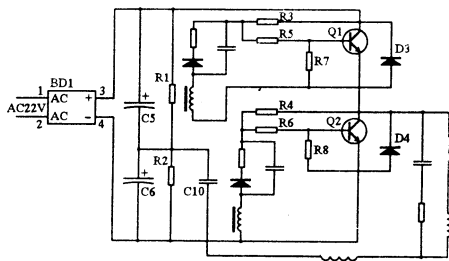


图4 电源开关管及其控制电路

(2) 电源开关管同时截止

电源开关管及其控制电路如图 4 所示。主变换器的 Q1、Q2 两开关管交替导通,它的驱动电路由于 IC494、功放管 Q3、Q4、驱动变压器 T1、T2 以及分压电阻 R3、R4 等相关元件组成。当 Q1 的基极得到驱动电流时 Q1 导通, C_s 经 Q1、T₂ 的初级绕组 NPC₁₀ 放电,使次级得到一个正向脉冲。Q1 截止后 Q2 的基极得到驱动电流,使 Q2 导通。图中 Q1、Q2 的外围是一对称的平衡电路, R1、R2 阻值相等 (180~270K)。此电路在设计中选择一对参数稍有差别的功率管 Q1、Q2,以打破电路的平衡状态,从而使 Q1、Q2 交替导通。但由于电源在使用一段后, Q1 和 Q2 的参数变化使其近似或相等,使电路再处于平衡状态。致使 Q1 和 Q2 同时截止,使得电源无输出。适当调换 R3 或 R4 的阻值,用 Q1 和 Q2 外围电路的不平衡性来改变电路的平衡状态。使 Q1、Q2 交替导通,开关电源恢复正常。

2.2 电源负载能力下降 (参见图 4)

电源负载能力下降的现象表现为,主机自检、屏显正常,但软、硬盘不能启动或电源轴流风扇 (接在 +12V 输出端) 工作正常,但主机不能启动。测量电源空载输出电压 +5V、+12V 均正常,而加载后 +5V、+12V 均有所下降。

加载后四组输出电压均下降,说明是变压器 T2 初级绕组提供的能量不足,故障在于主变换回路。检查各级电压时 Q1 的集电极为 210V, Q2 的集电极为 80V,整流后 C5 的正端为 210V, C2 的正端为 80V。由此表明 C5、C6 的容量减小,而且 C5、C6 的容量相差较大,使直流输入电压降低,导致 Q1、Q2 的输出电压幅度下降,次级得到的幅度也随之下降,故导致四组输出电压在加载后普遍下降。更换 C5、C6 后故障排除。

Word 打印的几个技巧

马军

(山东财政学院信息系

250000)

Word 打印功能是十分丰富的,除了 Word 书中介绍的以外,本文根据笔者的经验,总结出一般 Word 书中没有或不被人们注意的几个技巧。

1) 当现用机器没有安装打印机或需要更换打印机时,可在 Windows 3.x 的“程序管理器”中运行“控制面板”,然后双击“打印机”,或在 Windows 95 中调出“资源管理器”,运行其中的“控制面板”,再双击“打印机”来实现打印机的安装或更新;

2) “文件”菜单中的“打印”命令与“常用”工具栏中的“打印”按钮的不同之处在于,如果对打印范围等选项已设置好或不需要改变当前的打印设置,可单击后者,这样系统就会不再出现“打印”对话框,而是立刻驱动打印机打印文档;

3) 打印文件前,最好先“保存”一下当前文件,以免因打印过程中出错而造成修改信息的丢失;若打印的文档太长,打印过程中还需要计算机做别的工作,可在出现“打印”对话框后单击“选项”按钮,在“选项”对话框中单击“后台打印”按钮,进行了这样的设置就可以在打印文档的同时,前如进行其他事务的处理;

4) 选择打印的内容时,可分别选择打印“文档”、“摘要信息”、“批注”、样式、“自动图文集词条”、“键值”等,根据需要单独打印文件中某方面内容,也可以在指定打印文档时附带打印“摘要信息”、“域代码”、“批注”、隐藏文字、“图形对象”等,具体操作是:选择“文件”菜单中的“打印”命令,出现“打印”对话框后,在“打印内容”栏中选“文档”,然后单击“选项”按钮,出现“选项”对话框,在“打印文档的附加信息”一栏中,根据所需再单击“摘要信息”、“域代码”、“批注”、隐藏文字、“图形对象”之一,按“确定”按钮即可;

5) 不需要打印全部文档时,可对打印的部分先做块标记,然后就可以在“打印”对话框中选择“选定范围”单选按钮(没有做块标记时,“选定范围”单选按钮不能用);若只打印某几页,应先选择“页码范围”单选按钮,再类似地输入“3-9”表示打印文档中的第三页到第九页,输入“3,5-9,10,22”表示打印文档中的第三页、第五页到第九页、第十页、第二十二页;

6) 要实现文档的双面打印,可在“打印”对话框中的“打印”栏中先选择“奇数页”打印完所有的奇数页后,将打印纸整理好,再选择“偶数页”打印在已用打印

纸的另一面打印;

7) 若要同时打印多个文件,就需要利用“文件”菜单中的“查找文件”命令;运行“查找文件”命令,在随后出现的“搜索”对话框中设置搜索条件,单击“确定”按钮,经过一段时间的查找后,在随后出现的“文件列表”中列出了所要找的文件,选择其中要打印的那几个文件(选择方法请参考 Windows 3.x 中“文件管理器”选择连续或不连续文件的方法),接着单击“命令”按钮,选择其中的“打印”命令,此时出现我们见过的“打印”对话框,照 4、5、6 的方法加以选择,就可以连续地打印出这多个文档(也可附带地打印他们的“摘要信息”、“域代码”、“批注”、隐藏文字、“图形对象”等);

8) 按键盘上的“Print Screen”键可将当前屏幕打印到粘贴板上然后可利用粘贴板的功能对此进行操作;

9) 在“打印预览”工具栏中关闭“放大镜”按钮可对预览内容直接编辑,单击“打印预览”工具栏中的“标尺”按钮,在预览内容窗口中出现水平标尺和垂直标尺,用鼠标拖动标尺上的按钮可调整左右边距,在拖动鼠标的同时按下 Alt 键,在标尺上就会看到文本区大小及页边距的大小数值,在预览中若发现最后一页中仅有几行文字,可单击“打印预览”工具栏中的“放入同页”按钮便可将这几行文字追加到上一页的尾部,从而减少了文档的页码,但要注意:这上一页中的字体大小和行间距会缩小,另外追加的这几行文字太多时就不能实现“放入同页”;

10) 在分栏排版打印时,选择“打印”对话框中的“纵向”按钮在宽行纸上打印,往往会出现右边的内容打不在纸上的情况,但若改选“横向”打印,并将宽行纸旋转 90 度竖起装纸,就不会出现此类情况。实际上,以下两组操作程序的效果是一样的:(1) 选择“纵向”打印,自定义宽行纸的“宽”大于“高”,将宽行纸(宽边在前)装入打印机,按“打印”对话框的“确定”按钮后将在宽行纸上如“打印预览”中看到的一样打印;(2) 选择“横向”打印,自定义宽行纸的“宽”小于“高”,将宽行纸旋转 90 度竖起(窄边在前)装入打印机,按“打印”对话框的“确定”按钮后打印内容也将旋转 90 度在宽行纸上打印。

敬告作者和读者

有读者和作者来函来电询问本刊刊物级别和是否为核心期刊等问题,因人手所限,恕难一一作复,集中函复如下:

1.《计算机应用研究》杂志创办于1981年,系国内创办最早的计算机刊物之一,是具有国际连续出版物刊号和国内统一刊号的国内外公开发行的科技刊物,现为中国计算机学会(国家一级学会)会刊,国家科委系统省级计算中心联合办刊。

2.《计算机应用研究》从1990年起,即被中国科技论文引文库确定为中国科技论文统计源核心期刊,中国科技论文引文库是国家科委资助的、迄今为止国内最大、最具权威的期刊引文数据库,并与国际上著名的《SCI》、《ISTP》、《EI》等大型检索数据库保持有数据交

换、信息共享关系和密切的业务联系。凡在本刊上刊出的学术论文,均被上述引文库统计收录,建库和存档。

3.《计算机应用研究》从1996年开始正式入编《中国学术期刊(光盘版)》电子出版物,本刊是首批入编的学术刊物之一。

总之,《计算机应用研究》杂志系国内外公开发行的国家一级学会核心期刊,特此集中作复,并衷心感谢多年来与本刊风雨同舟、休戚与共、热情关心支持本刊的广大作者与读者,向这些作者与读者致以崇高的敬意,祝其事业繁荣兴旺,蒸蒸日上,欣欣向荣!

《计算机应用研究》编辑部

一九九七年七月一日

《计算机应用研究》杂志联(协)办单位

中国计算机学会
四川省计算中心
贵州省计算中心
安徽省计算中心
吉林省计算中心
福建省计算中心

内蒙古电子计算中心
云南省电子计算中心
新疆电子计算中心
甘肃省计算中心
广西计算中心
新潮计算机产业集团公司

山东省计算中心
河南省计算中心
青海省测试计算中心
福建林学院计算机科学系
武汉汽车工业大学电子系
81890部队自动化站

《计算机应用研究》杂志社董事会

董事长:周赛渝
董事:乔中南 张国栋 徐雅君 王升亮
徐世坤 叶大卫 管小宝 明德
冯德成 朱华 安承熙

《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:罗海鹏 周荣春 龚宇清 张湘金
倪凯民 王义章 范德元 李文华
李天健 毕常青 崔振远 赵桂林
宁正元 刘铁军 贾洪钧 蒋立建
李建华 黄安南 刘启茂

1997年第5期(总第79期)

出版日期:1997年9月30日

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:国际标准刊号:ISSN 1001-3695
国内统一刊号:CN51-1196/TP

邮发代号:BM4408(国外)
62-68(国内)

主 编:张执谦 副主编:李泽民

主管单位:国家科委、四川省科委

主办单位:四川省计算中心

暨国家科委13省计算中心

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 图文传真:(028)5221946

电 话:(028)5210177,5220666 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