

电脑与法律

不要一稿多投 本刊评论员 (2)

电脑应用

会计电算化问题探讨..... 黄志慧 (3)

电子数据交换(EDI)面面观 (1) ... 李凤仪 (5)

商业管理信息系统与收款机 管 强 (9)

综述

计算机技术和产品的现状与展望 ... 钱承德 (11)

软件纵横

ORACLE RDBMS V5.1 系统分析及应用研

究(上) 沈欣等 (15)

网络与通信

高级 UNIX 连网技术讲座 第三讲 NFC

和 RPC 冯家宁 (19)

使用与维修

打印机维修二例 陈宁裕 (24)

Star-AR2463 打印机维修一例 温 波 (24)

也谈使九针打印机字迹变更清晰 ... 向勇飞 (25)

四通 MS-2401 打字机断针的预防及断针的更换

..... 蔡长安 (26)

NEW

数据保护神 无忧卡 (27)

IDEA

合理搭配 CCBios 2.13H 和 SPOOS 5.0 的批处

理程序 杭箴良 (28)

多操作系统共字硬盘技巧 王兴昌等 (29)

编续往来

一篇文章的疏忽 黄泳进 (29)

天堑变通途 苏 睿 (30)

ABC

一种快速删除无用文件的方法 张少丹 (30)

巧用 FOXBASE 中的宏命令 宋 捷 (31)

DRIVE.SYS 设备命令的几种用法

..... 曹松林 (32)

游戏乐园

和电脑玩“24 点” 程宇清 (33)

让“半仙”变“全仙” 任绥海 (34)

侠影记的修改 麦永佳 (35)

浅谈 RPG 游戏的发展 赵礼海 (36)

病毒防治

MS-DOS6.0 的防病毒及清病毒功能

..... 顾子明译 (39)

一种新的 1990 病毒 胡向东 (40)

单片机与单板机

单片机与模糊控制讲座 第七讲 MC6805 的指

令系统和程序设计(上) 余永权 (42)

用 Z-80 组成的数字自动增益控制

..... 丁永林 (44)

小辞典

..... (48)

电脑用户

在程序中直接读取 CMOS 信息

..... 罗辉等 (50)

PC 机驱动器清洗工具 Driver Clean

..... 刘永强 (51)

用 Tango 和 AutoCAD 联合进行电原理图设计

..... 蒋肖南 (54)

2130A 矢量字库的压缩与恢复

..... 虞培发 (55)

广告索引

1. 广州白云山电源设备厂
2. 广州易通计算机应用工程开发公司
3. 广州电子设备公司
4. 特强(广州)电子有限公司
5. 赛宝星河
6. 广州方正公司
7. 广州科教电脑设备厂
8. 电子工业出版社广州公司
9. 南方计算机网络有限公司
10. 电脑杂志社科技开发经营部
11. 华力科技开发公司
12. 珠海征达科技工贸广州公司天河商行
13. 星辰企业发展总公司
14. 广州市天河科立特技术发展公司
15. 广州海谊电子仪器实业公司
16. 京粤收款机系统
17. 香港现代电子出版社
18. 清华大学科技馆
19. 电脑杂志社
20. 华粤电脑工程公司
21. 华芝电子新技术公司
22. 广州无线电研究所

不要一稿多投

·本刊评论员·

刊上期刊登了读者裴钰华的来信,他指出本刊近几期中,有几篇文章是在其它刊物上登载过的,这是对读者不够负责的表现。就他提出的具体情况来看,批评是正确的。

稿件重复刊登有种种原因:

一、一篇文章,在某个刊物上刊登之后,被另外的刊物认为有价值而予以转载(全文或摘要转载);

二、多个刊物,都认为有必要刊登同一文章(这一类多为新闻的、官方的、权威性的文章),而造成同一时期内(不一定同一天)一篇文章出现在多个刊物上;

三、一篇已发表的稿件,经过长时间之后,认为有必要重新发表;

四、作者把作品投出后,没有得到收稿方的答复(包括未来得及答复或因刊物未收到该稿),认为该稿未被采用转而投,但结果在两个刊物上同时被采用;

五、同一作者有意一稿多投;

六、发表之后,经作者改头换面(没有新的内容)重新投出而被不了解情况的刊物采用;

七、抄袭作品而被发表;

当然,还会有其它的情况,如稿件被人盗用,原作者和盗用者投出后分别被不同刊物发表,等等。总的来说,稿件在刊物中重复出现,原因是多方面的。

我国著作权法第三十二条规定“著作权人向报社、杂志社投稿的,自

稿件发出之日起十五日内未收到报刊通知决定刊登的,或者自稿件发出之后三十日内未收到杂志社决定刊登的,可以将同一作品向其它报社、杂志投稿。双方另有约定的除外。

作品刊登后,除著作权人声明不得转载、摘编的外,其他报刊可以转载或者作为文摘、资料刊登,但应当按照规定向著作权人支付报酬。”

根据著作权法的规定,上述前三种情况,只要在转载文章时作出必要的说明和照规定付酬,原作者和读者是不会有意见的。上述第四种情况比较复杂。按著作权法规定向报刊投稿的,决定刊登答复的限期报纸是15日;杂志是30日。但这个期限都较短,很多编辑部难以做到。所以,著作权法有“双方另有约定的除外”一句。什么是“双方另有约定”呢?编辑部和作者之间如果真已签定协议,当然属于双方有了协定。但事实上,不可能在每个作者投稿之前,都去和编辑部专门签订一个协定。所以,著作权法指的这个协定,还包括指编辑一方根据自己的情况,提出“征稿启事”,“投稿须知”(以下简称“须知”)等编辑方面的要求;由于投稿是自愿的,作者向某个刊物投稿,实际上是以承认这个“须知”作为前题的,这就等于双方有了一个共同的协定。因此,双方都必须依照“须知”办事。以本刊为例,多次(不是每一期)登在本刊上的“须知”是任何向本刊投稿的作者所必须知道的。该“须知”规定之一,便是“为了维护读者的利益,请不

要一稿多投。来稿恕不退回,请作者自留底稿。投稿两个月后未收到用稿通知,作者可自行处理。”可见,这里规定是用稿通知应在投稿之后的两个月内发出,而不是著作权法规定的一个月;同时还规定了“不要一稿两投”。这两点就属于刊物和作者的协定之一,双方都要遵守。有些作者对这些规定没有足够的注意,稿件投出后不到规定期限,便以为本刊不予采用,转而投,容易造成文章重复被录用。有些刊物这个期限长达三、四个月,投稿者也应当遵守。等到过了这个限期没有接到用稿通知,才可以把原来投出的稿件另行处理。

至于有意一稿两投,抄袭或改头换面重投,这都是存心欺骗读者,这种错误做法是十分不可取的。而严重的就是违反著作权法的行为。

防止一稿多投,编辑部有相应的责任。在措施上最有效的是要建立一个与本杂志内容相接近刊物的文章索引数据库,以便检索核对。但是,这不是一件轻易的事情,以计算机为例,我国当前这方面的杂志有数十种之多,信息量非常大。要完全把这些文章都收齐很难于一朝一夕办到。本刊第一步,将对国内著名的计算机刊物建立当年的文章索引的数据库,防止本刊稿件和这几个著名刊物上的文章重复,然后再逐步扩展。但毕竟编辑部的人力有限,所能查对的范围也很有限,要彻底消除一稿多投的现象,最根本的还是要提高作者对读者的责任感和守法意识。

396

会计电算化问题探讨

天河会计师事务所 黄志慧

会计电算化,就是用电子计算机代替人工的记帐、算帐、报帐工作,部分替代由人脑完成的对会计信息的分析和判断。完成这些功能的是会计核算软件。会计核算软件处理的是会计核算的基础数据,这些数据的记录和计算是否准确可靠,关系到国家会计制度的贯彻执行,在手工条件下,记帐程序、规则由统一的会计制度规定,各级财政、财务、审计等部门通过检查凭证、帐簿、报表来监督和验证会计工作是否正确贯彻了国家的法规和制度。会计电算化后,记帐程序、规则也必须遵循国家统一的会计制度的规定。而要做到这一点,首先要使会计核算软件的技术处理符合要求。为此,财政部根据现行的会计制度和会计电算化的特殊要求,发布了《会计核算软件管理的几项规定(试行)》和《会计核算软件评审问题的补充规定(试行)》,要求对会计核算软件进行评审,根据这两个文件的精神,广州市财政局制订了《广州市会计电算化应用系统评分标准(试行)》,要求会计核算软件在合法性、安全保密性、功能性三方面达到该标准规定的要求。评审工作由财政部门安排会计师事务所负责。单位申请、会计师事务所主持评审、财政部门审批会计核算系统工作程序如后表。我们在评审中发现比较典型的问题主要有以下几方面。

1、受不良传统做法的影响,手工操作不规范。例如:不是逐笔录入银行日记帐,而是十几张甚至更多的银行收支业务混在一起录入,不方便与银行对帐;有关债权债务的科目没有按单位、个人分设明细帐户;还有的在同一张凭证上涉及多项会计事项以致出现不合常规的多借多贷会计分录等等。

2、会计核算软件模仿手工操作条件下的传统做法,以致数据重复存储和数据处理流程迂回曲折,既浪费存储空间,又影响运行速度。例如手工会计核算中由总帐来控制各种明细帐,制约各类帐册之间的正确性。在会计核算系统中,总帐与明细帐的数据均取自凭证,只要软件编制正确,以明细帐、日记帐余额为基础汇总而编制的“总分分类科目余额表”是可以作为总帐运用的,

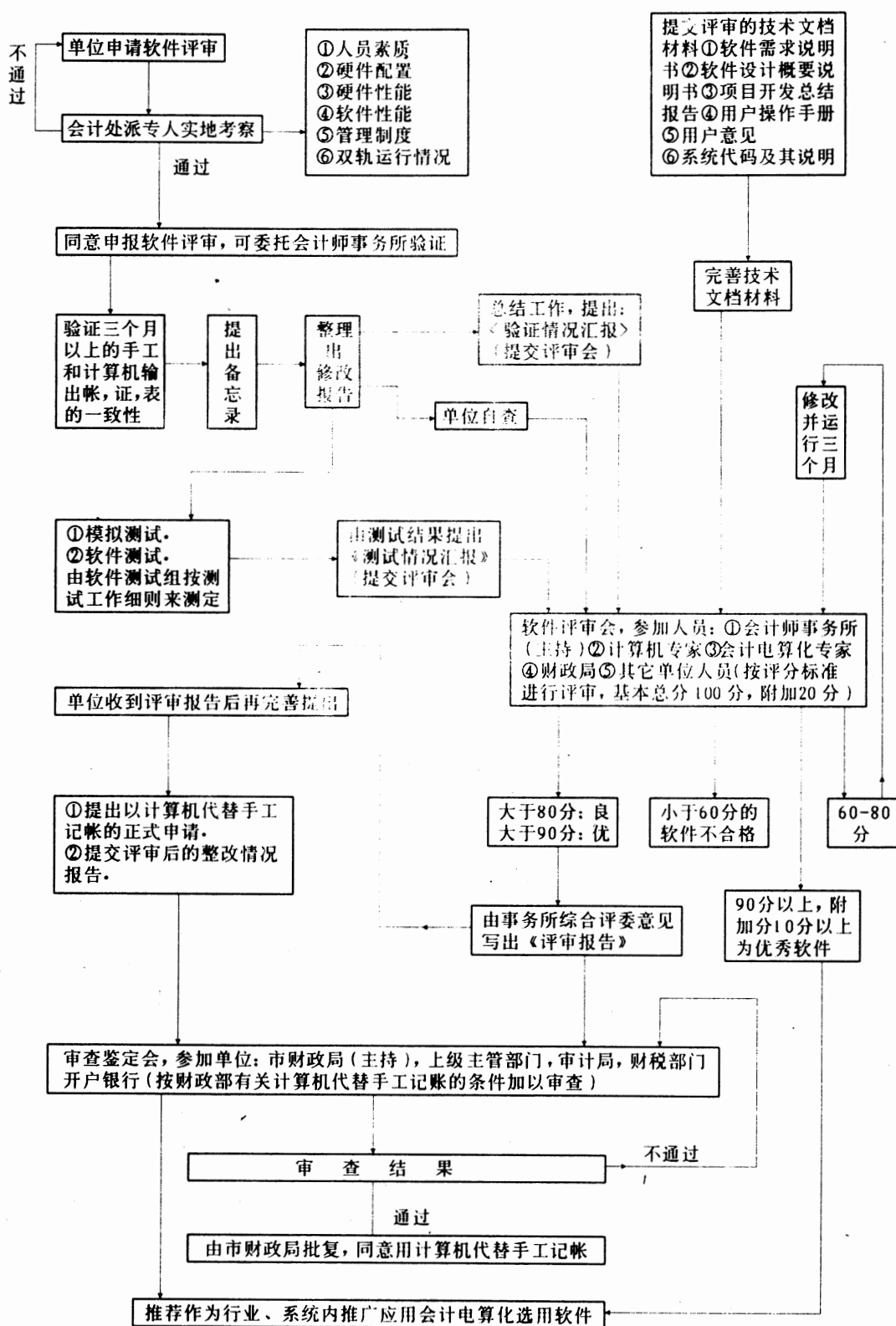
在机内不需要另外再设总帐。

3、开发人员弄不清会计报表间以及报表各指标间的勾稽关系,报表取数公式设置错误,从而导致会计报表上的数字不正确。

4、忽视了辅助功能的重要性。由于会计核算软件所处理的数据结构复杂、量大,直接关系到各方面的经济利益,因此需要建立权限控制、时序控制、校验机制、备份机制、防毒机制、运行日志等一系列辅助功能,以提高会计核算软件的安全保密性和可靠性,防止利用计算机犯罪事件的出现。权限控制是指进入系统或进入记帐、复核、登帐、编报表等重要模块应加密码或权限控制措施;各操作员和财务主管使用各自独立的密码或独立的工作权限,其密码自行设置,自行更改;系统开发人员不得参与数据录入,出纳人员只能查询和打印现金、银行存款日记帐及进行对帐。时序控制要求工作日期不应超越系统日期,凭证编号与凭证日期满足同时递增的规律。校验机制包括对会计科目对应关系的校验,对借贷方金额的平衡校验,对责任人的逻辑校验。备份机制包括提示备份和强制备份,以便在非正常情况下,能利用现有的备份数据恢复到最近状态。防毒机制是指预防病毒的措施,包括安装防病毒卡及加强管理制度。运行日志对每次使用时间、人员、模块进行记录,并能记录企图进入系统但未遂的人员所使用的密码及作案时间。

5、缺少财务分析功能。在市场经济条件下,要搞好企业经营管理,就必须对经营过程实行全方位的控制。因此,利用会计核算系统的数据作计划执行、增减变动、结构比重、成本、利润等分析,从管理的角度提高财务工作的自动化、科学化显得尤其重要。企业要搞好会计电算化工作,还应注意培养会计电算化人才,特别是从财会专业人员中培养会计电算化应用的复合人才。根据规定,开展会计电算化的单位的操作人员必须取得市财政局颁发的《会计电算化操作员合格证》或《会计电算化初级程序员合格证》。

广州市会计电算化软件申请使用、评审、审批流程



电子数据交换 EDI 面面观(1)

中国海关管理干部学院 李凤仪

第一章 EDI 技术综述

一、EDI 定义

近年来,EDI(电子数据交换)已成为一种全球性的、具有战略意义和巨大商业价值的贸易手段,各国海关亦逐步部署 EDI 系统。EDI 是 Electronic Data Interchange 的缩略,在我国译为“电子数据交换”,港台及海外华人地区则译成“电子资料联通”或“电子文件(资料)交换”,由于使用 EDI 可以减少甚至消除贸易过程中的纸面单证,因而 EDI 也被通俗地称为“无纸贸易”。

EDI 定义是按照协议,对具有一定结构特征的标准经济信息,经过电子数据通讯网络,在商业贸易伙伴的电子计算机系统之间进行文件交换和自动处理。

上述定义表明:

(1)EDI 是面向经济信息的,如订单、发票、船运单、报关单、进出口许可证等。并且这些信息都是具有固定格式的。信件等非格式化文件不在 EDI 的处理范畴之内。

(2)EDI 报文要依循国际标准。

EDI 的标准包括数据元素标准、数据段标准和单证格式标准等。(3)信息传递的路径是从计算机到数据通讯网络,再到商业伙伴的计算机,中间不需要人工干预。

(4)EDI 信息的发送者和接收者都是计算机的应用软件系统,能自动处理传递来的数据。这种传输模式是机——机、应用——应用的。

联合国标准化组织将 EDI 描述成“将商业或行政事务处理(Transation)按照一个公认的标准,形成结构化的事务处理或报文(Message)数据格式,从计算机到计算机的电子传输方法。

EDI 按联合国使用的定义为“用约定的标准编排有关的数据,通过计算机向计算机传送业务往来信息”。

二、EDI 产生背景与 EDI 应用效益

EDI 起源于 60 年代末的西欧和北美。EDI 这一概念是一家由英国政府资助的服务机构 Simpler Trade Procedures Board (SITPRO) 于 70 年代提出的。

信息技术的发展使计算机及通讯网络不断更新换代,从而影响贸易发展的地理距离和信息成本等因素的作用在逐渐减少,为 EDI 的出现提供了技术基础。

跨国公司的出现促进了全球化市场的形成。跨国公司如雨后春笋般涌现,由全球性跨国公司交易的世界总产值份额越来越大,市场竞争也出现了新的特征,价格因素在

竞争中所占的比重逐渐减少,而服务性因素所占比重逐渐增大,全球范围的商业后勤组织和管理成为产品成本的主要组成部分,提高商业文件的传递速度、处理速度、空间跨度及准确无误,追求商贸的“无纸化”,便成为所有贸易伙伴共同的目标,这是 EDI 出现的最主要的原因。全球贸易额的上升带来了各种贸易单证、文件数量的激增。据美国专家统计,平均每做成一笔生意需要 30 份纸面单证。全世界每年因贸易活动而产生的纸面文件要以“亿”为单位计算。据许多国家统计,单证费用占产品贸易额的 76% 左右,我国外贸每年的单证费用约 70 亿美元。美国生产纸表格的主要生产商 Moore 公司估计纸张表格占美国商业每年生产的 2.5 万亿张纸页的三分之一,各公司每年要花 1000 亿美元来处理表格。贸易和运输单证编制、传递的质量和速度直接影响贸易的进程和成本,纸面文件成本高、传递慢、经常重复处理、处理费时费力又易出差错,纸面贸易文件成了阻碍贸易发展的一个比较突出的因素。例如,在我国经由深圳出关运香港的货物,就曾出现过因单证未能及时运到而不得不压车的事情;出口到香港、澳门等地的湖南鲜活畜产因压关、压港而死亡,造成严重经济损失。1991 年出口欧洲的纺织品由于压关、压港等原因造成经济损失达几千万美元。美国海关规定,远洋船到港前 7 天必须报关,否则作滞后处理,一切责任由运输部门负责。资料分析得知,一艘装载 2700 箱的远洋轮,如压港一天,便要损失 5—6 万美元,由此延误交货日期所付的违约损失,可高达 50—60 万美元。EDI 使从原料到生产、销售的整个过程的各个环节能更紧密地结合,降低了生产成本。

由于 EDI 采用标准化格式和规范语言,消除了国家和民族间语言、文化的障碍,有利于合同签订,适应了经济区域化的趋势,并为其提供强有力的联系手段。

EDI 一经出现,便显示出强大的生命力,迅速地在广大地区得到广泛应用,历经萌芽期、发展期而步入成熟期。愈来愈多的国家均已认定 EDI 是经商的唯一途径。据统计 1989 年世界 EDI 用户约 2 万,1992 年达 10—15 万,平均年增长率为 86%。美国排名前一百个大企业中有 97% 使用了 EDI。

据统计,使用 EDI 可提高文件传送速度 81%,可降低邮递成本 44%,可降低文件处理成本 38%,还可减少错漏造成的商业损失减少 40%,可提高竞争力 34% 和减少顺

客流失量等,效益巨大,是企业获得最佳经济效益的有力手段,故一经出现,就发展迅猛,形成全球性 EDI 热潮。

例如,美国通用电气公司(GE)用直接 EDI 环节处理其业务,该系统包括 2500 家贸易伙伴,每年仅仅在 GE 这一端就取消了几千 400 万张业务纸表格,估计可节约数千万元美元。

又如,据行家估计,新加坡全国贸易网 TRADENET 每年可为新加坡节省 10 亿新元(约合 6 亿美元)的文件处理费用。新加坡在实现 EDI 前报关一般要 3 天,每次 12—15 新元;在实现 EDI 后,报关只要 15 分钟,每次 1—2 新元。

美国五角大楼已有七年历史的 CALS(计算机辅助采购和后勤支援),据估计可使武器开发时间缩短 40%,EDI 还将帮助军方每天采购五万种价值在 25000 美元以下的商品。

美国 DEC 公司使用 EDI 后,存货周期从 35 天缩减为 3 天,每笔订单费用从 125 美元降至 32 美元。

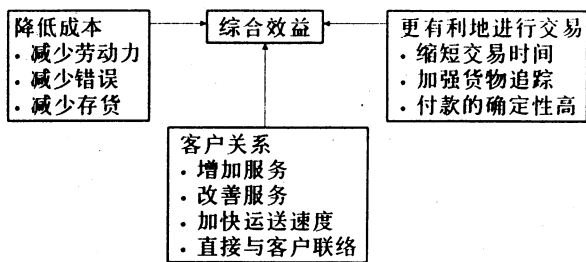
美国通用汽车公司推行 EDI,每生产一辆汽车可节约成本 250 美元,按年产 500 万辆汽车计算,一年便可产生 12.5 亿美元效益。

美国通用电器公司近五年来统计,应用 EDI 后,产品零售额上升了 60%,库存由 30 天降至 6 天,每年仅连锁店文件处理费一项就节约 60 万美元。

英国煤炭公司应用 EDI 于货物进口,每年可节省 300 万英镑。

日本东芝公司在使用 EDI 之后,每一笔交易的文件处理费用仅为原来的四分之一。

EDI 的商业效益可图示为:



据估计,使用 EDI 的间接效益能达到其全部贸易价值的 3%—5%。间接效益主要来自于将原来分散的业务密切综合统一而取得的规模经济效益。

三、EDI 技术特点

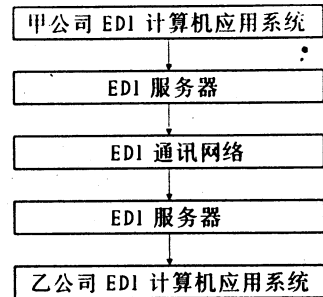
按照 EDI 的定义,EDI 系统模型可图示如下:

系统模型显示了 EDI 的三个方面的。

1. EDI 是一种数据的电子传输方法,所以 EDI 系统必包括通讯网络。

2. EDI 信息可被自动处理,这要求通讯网络要与商业

伙伴的计算机相连接,同时要有一定的软件系统来支持。



3. EDI 信息要为不同的计算机系统所识别,这就要求有一套大家公认的标准格式,同时还要有一个装置来完成用户文件格式与标准文件格式之间的相互转换。

这三个方面可概括为以下三句话:

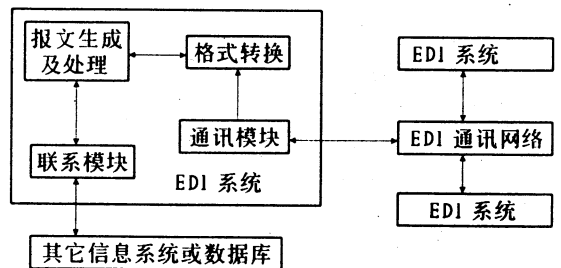
数据通讯是 EDI 的基础。

计算机应用是 EDI 的条件。

数据标准化是 EDI 的特征。

一个完整的 EDI 系统应包括四个功能模块,即联系模块、报文生成及处理模块、格式转换模块和通讯模块。

以下为 EDI 功能框图。



• 联系模块是 EDI 系统与内部信息管理系统和数据库的接口,同时也是和用户的接口。

• 报文生成及处理模块用来生成 EDI 报文和接收外部 EDI 报文并处理。

• 格式转换模块将产生的报文转换成符合通讯标准的格式,同时将接到的报文转换成系统可读懂的格式。

• 通讯模块是 EDI 系统与通讯网络的接口,执行呼叫、自动重发等功能。

EDI 系统工作过程如下:

1. 当一份 EDI 报文经网络传输过来时,通讯模块首先将其接收下来,检查无误后,发确认报文,通知对方已收到正确的 EDI 报文。如果报文有误,则通知对方重发。

2. 将收到的 EDI 报文交给格式转换模块,进行非结构化格式转换,形成一份计算机可读懂的报文格式。

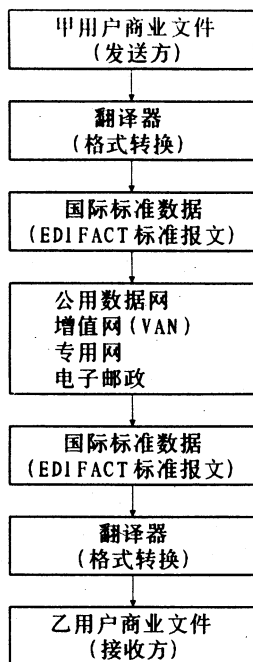
3. 将经格式转换后的报文送到报文生成及处理模块,

按照不同类型的要求进行处理。

4. 系统将该报文传递给联系模块, 将必要的信息通知用户和内部信息管理系统, 由系统决定是否激发其它处理过程, 如查询库存、发出零部件订单等。

系统产生、发出 EDI 报文的过程与接收 EDI 报文的过程正相反, 不再赘述。

系统工作流程亦可图示如下:



EDI 标准范围包括:

- EDI 行业应用标准
- EDI 联系标准
- EDI 处理标准
- EDI 语义语法标准
- EDI 网络通讯标准

作为 EDI 标准, 应达到以下目的:

- 提供一种任何贸易伙伴都可使用的语句, 这种语句是无歧义的, 是可以使得使用者明白其含义的。
- 这种标准是不受计算机机型影响的, 是适用于计算机间的数据交换, 同时又独立于计算机之外的标准。

EDI 网络通信标准目的为解决 EDI 通讯网络应该建立在何种通信网络协议之上, 以保证各类 EDI 用户系统的互连。

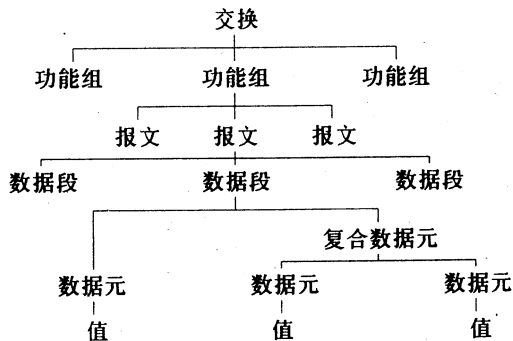
目前国际上提出采用 MHS(文电作业系统)作为 EDI 通信网络结构, 为此要研究 EDI 与 MHS 的接口标准, 以解决 EDI 的支撑环境。

EDI 处理标准为不同地域、不同行业的各种 EDI 报文相互共有的“公共元素报文”的处理标准。它们与数据库、管理信息系统(如 MRP II——制造资源计划)等接口有

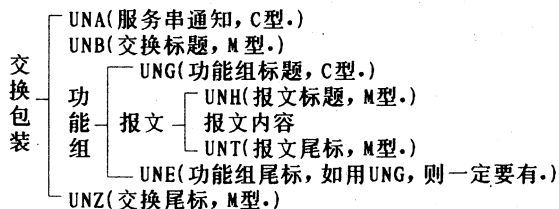
关。

EDI 语义语法标准主要解决各种报文类型格式、数据元编码、字符集和语法规则等。

为了电子传输, 必须按标准规则和格式把报文逐步细分, 每个电子数据交换及联合国标准报文(UNSM)的结构如下图



不同地区、部门或不同性质的报文组成不同的功能组, 一个功能组可包含几个同部门或同性质的报文。报文格式的更细致描述如下图所示:



UNA 后跟报文分隔符, 例如 + : ' ? 等, 规定报文所使用分隔符。

UNB 给出发送者和接收者的有关信息。

UNZ 给出在一个交换中的功能组数或报文数, 或给出与 UNB 所规定相应的交换参考号。

UNG 定义功能组。

UNE 规定在一个功能组内的报文数。

UNH 规定报文类型、子集类型和报文版本。

UNT 规定在一个报文中段总数。

图中各项分强制型(M型)和条件型(C型)两种, 前者是一定要有的, 后者可按情况选用。

ISO7372 规定了数据元标准。

ISO9735 明确了 EDI 的语法要求, 主要用于格式转换。

EDI 联系标准主要解决 EDI 用户所属的其他信息管理系统或数据库(参见 EDI 功能框图)与 EDI 系统之接口。

行业应用标准如:

CIDX 化学工业数据交换

EIDX 电子工业数据交换
 ODETTE 用于汽车工业自动化
 UCS 用于杂货
 VICS 用于零售
 WINS 用于货栈
 ISO2382 数据处理词汇
 ISO3166 国家名称代码
 GB2659-86 世界各国和地区名称代码
 ISO4217 货币及基金代码
 ISO6346 运输集装箱代码
 GB4290-84 集装箱运输状态代码
 GB6582-86 运输方式代码
 GB7407-87 中国及世界主要海运贸易港口代码
 AWB 国际航空货运单(国际航空运输协会)
 CIM 铁路托运单(国际铁路运输公约)
 CIMP 货物交换程序手册(国际航空运输协会货物进出口手册)
 MAR 国际海事组织标准格式和国际海运局标准提单
 SAD 欧洲经济共同体单一管理单证
 ISO646 信息交换用七单位编码字符集
 GB9648-88 国际单位制代码
 ISO216 书写纸张和某些类型印刷物品
 联合国推荐的 EDIFACT 标准由 UN/ECE 印制为“联合国贸易数据交换指南”(UNTDID), 它包括十个部分:

(1) EDIFACT 语法规则 (ISO 9735, 1987 年 9 月认定)。

- 适用范围
- 相关标准
- 名词定义·字符集级别
- 标准字符集
- 标准报文结构
- 单证到报文的转换
- 段重复
- 段嵌套
- 数字型数据元使用
- 附录 A 名词术语定义
- 附录 B 服务段描述
- 附录 C 段顺序

(2) 报文设计指南(1989 年 12 月 UN/ECE 认可)

- 说明介绍
- 设计总体规则及报文类型
- 设计步骤——数据元分析, 段结构设计, 报文结构设计。

设计。

- 报文格式修改及如何得到最新国际报文格式。

(3) 语法应用指南

- 总体介绍
- 交换协议
- 专用名词术语
- 交换字符集的定义
- 对数据元、段和报文的要求
- 对报文标准版本的规定
- 基本语法规则
- 与其他标准相互转换, 标准的支持与维护。

(4) EDED (EDIFACT 数据元目录)

数据元号、名称、定义、数据类型、数据长度。近 300 个。是联合国贸易数据元目录(UNTDID)的一个子集。

(5) EDCL (EDIFACT 代码目录)

数据元号、代码。共 103 个。

(6) EDCD (EDIFACT 复合数据元手册)

复合数据元号、组成数据元号及类型(M—必写、C—条件)、用途。共 60 多个。

(7) EDS (EDIFACT 数据段手册)

段名、段标识(段名英文缩写、三个字母)、组成数据元及类型。共 70 多个。

(8) EDMD (EDIFACT 报文手册)

使用范围、专有名词定义、报文结构、段功能、段表、分支表。

报文分三级:

0 级—草案级

1 级—试用推荐草案级

2 级—推荐报文标准级

一、二级报文现共有 100 个, 二级报文有 18 个。

1991 年 9 月联合国推荐的海关报文有:

CUSDEC 海关申报

CUSRES 海关复文

CUSCAR 海关货物舱单

CUSREP 海关运输(工具)舱单

正在设计的有:

CUSEXP 海关快速申报

PAXLST 旅客名册

INSTAT 内部团体 (INTRACOMMUNITY) 统计申报 (CUSDEC 子集)

还将研制:

VATDEC 增值税 (VAT) 申报

(9) UNCID (贸易数据交换格式构成总览)

标准产生背景、目的、对用户要求。

(10) 适当的说明解释。

一、前言

商业领域是最早采用计算机进行信息处理,并且是计算机技术应用最广泛、最完善的领域之一。据

有关资料统计:在发达的工业国家,商业计算机应用系统在整个计算机应用系统所占的比例是: 美国: 60% 日本: 55% 新加坡: 65%

由此可见,商业计算机的信息管理在计算机信息管理中的地位和重要性。我国商业计算机信息管理的起步较晚,应用落后。据商业部有关统计资料,商业管理信息系统在整个管理信息系统中所占比例不到1%,究其原因,一方面是原来我国经济体制不是市场经济体制,商品的流通与管理以计划为主,商业部门不作为一个产业部门予以承认,企业的盈亏由国家负责,与企业本身无太大关系,这样从客观上防碍了计算机在商业领域的发展。另一方面,由于计算机设备,尤其是收款机方面,国内无成套产品供应,加上认识问题,也阻碍了商业管理信息系统的应用和发展。改革开放以来,有些商业企业,也自主开发了一些有关商业的管理信息系统,但大部分是基于后台管理(如财会、人事管理)开发的,并没有采用收款机联网工作,导致商业中最重要的销售、经营信息得不到完整、及时的处理,结果是系统不完善,达不到预期的目标,成效甚微,投资也难以收回。

发展商业信息管理系统,提高商业企业现代化管理水平,以促进和加速发展我国社会主义市场经济,已成为国家商业领导部门和每个商业企业迫切的任务和急切的需求。为此,国家已将发展和促进计算机在商业领域的应用做为八五科技规划中的重要内容。作为商业信息管理系统中的重要组成部分——收款机,国家电子产业部门,也列入到重点发展的产业计划。

本文就商业管理信息系统和收款机在其中的作用,谈些看法,供读者参考。

二、关于商业信息管理

商业管理信息

与

收款机

管强

金额等等。

(5)商品的库存信息。如商品在各个仓库或位置的存货、缺货数量,库存资金、盈亏数量、保本期、保鲜期等等。

(6)财会信息。如各种会计凭证,会计明细帐等等。

(7)其它管理信息。如职工劳动人事信息,职工社会保险信息,办公室自动化信息等等。

这些原始信息经 BMIS 系统处理后,可按商业管理上的要求,产生出与经营相关的信息,供经营者使用。

3、商业管理信息系统的功能

按照 BMIS 处理商业经营过程中原始信息的内容,在一定的信息管理原则和方法下,建立这些数据之间的关系和加工逻辑,便实现了商业信息管理系统的功能。

BMIS 系统的功能实现,应围绕如下几个目标进行:

(1)对经营过程中的商品进、销、存信息进行全面管理,有效地控制商品经营成本,扩大商品的销售数量,缩短商品资金的周转周期。

(2)建立完善的经营管理体系和制度,提高经营管理水平。

(3)实现经营管理的计划性,作到有根据的预测和科学地决策。在这些目标要求下,BMIS 应具有如下功能:

①商品特征信息的管理功能。

BMIS 的基础是管理商品,而描述一个商品则用商品的特征信息。商品最重要的特征信息是商品编码,它是商品的唯一标识符号。商品编码由不定长数字码组成,其内容可以是条形码、自定义“店内码”。

采用条形码可以与销售收款机相连系,通过收款机自动完成商品编码的扫描输入和一系列处理,但条形码用做分类统计较为困难。

自定义“店内码”实际上是分类统计码,如采用这种编码方法,也可将其转化、印刷成条形码,供销售商品时使用。

1、商业管理信息系统

商业信息管理指的是根据商业企业在运作过程中所产生的信息,加以统计、分析,指导和实施对商业企业的经营管理。如果借助计算机系统来实现采集、处理、统计、分析这些信息,以及时、准确的数字结果反映给企业经营者,辅助企业经营管理,这样的计算机系统就称为商业管理信息系统(The Business Manage of Information System)。我们简称为 BMIS。

2、商业管理信息系统的内容

商业管理信息系统(BMIS)主要内容是:利用现代计算机技术,采集商业企业经营过程中所产生的各种原始信息,根据经营管理的要求,对这些信息分类、汇总、分析,及时、准确的得出数据结果,供经营者随时了解企业经营状况,做出正确的经营决策。

商业管理的核心是管理商品。小的商店管理数千种商品,大型商场则管理数万种商品,所以,BMIS 的核心是能够随时提供反映每一种商品状态的信息。

做为一个完整的 BMIS 系统,它们能够采集、处理的原始信息种类有如下内容:

(1)商品特征信息。如商品编码、分类码、名称、进货价、不同销售价等等。

(2)商品采购信息。如商品进价、变价、采购数量、包装、验收、破损数量,商品供应商资料和可供商品信息等等。

(3)商品流转信息。如商品的调拨、周转等等。

(4)商品销售信息。如商品销售的单价、数量、币种、付款方式、

良好的 BMIS. 商品描述应同时采用这两种编码方法, 当然, 这会增加管理上的难度。除了编码外, 还应有商品进价、销价、供应商等方面的特征描述处理。

②商品价格管理

价格管理包括商品变价, 变价计划与策略, 商品变价后的市场反映, 甚至广告反映及变价后的商品销售利润分析等内容。

③商品进货管理

商品进货管理包括商品缺货、进货策略、进货时间控制、进货价格、商品供应商情况, 商品的运输与验收管理等内容。在先进的工业国家, BMIS 可以根据库存情况和供货周期自动产生商品定货单, 通过 EDI 网络直接由计算机向商品供应商签属订货合同, 并安排好运输、验收计划。

④商品调拨管理

商品调拨管理包括商店内部多仓库之间的商品的相互调拨。对于连锁店的 BMIS, 还可实现根据不同商店的库存和销售情况由配货中心自动配货和安排商品运输计划。

⑤商品销售管理

在 BMIS 系统中, 商品销售管理借助收款机实现, 包括: 销售明细记录, 按收款机、收款员、营业员销售统计, 销售时区分析, 销售汇总、分析等内容。

⑥商品库存管理

商品库存管理是 BMIS 的重要部份, 包括多仓库的动态库存数量, 商品的货位、货架、保本期、保鲜期、警戒库存数、仓库盘点盈亏数管理及商品库存数量、资金分析等内容。

⑦财会管理

包括记帐凭证、会计明细帐、总帐、商品帐, 会计报表及涉及到金额的固定资产、工资等内容。

⑧其它管理

在 BMIS 中, 还可以加入例如人事档案、职工保险、办公自动化等方面的管理内容。一个好的 BMIS 系统, 不仅仅是要功能齐备, 更重要的是这些功能实现合理, 功能之间互连性操作方便, 用户界面直观易学。

三、收款机在商业管理信息系统中的作用

1、收款机的特点与发展

收款机的发明和首先应用在美国, 当时的作用是计算商品的销售金额, 最先出现的是手摇式机械收款机, 相当于一个手摇计算器。随着电子技术的发展, 出现了电子收款机 (ECR), 其作用仍然是销售计算和统计。当出现了商品条形码技术和政府将收款机作为计税工具之后, 收款机有了大的发展和普及。现代高级收款机已经发展到销售点处理终端 (Point of Sale, 简称 POS), 并开始向销售工作站 (Workstation for sales) 发展, 成为具有极高性能的销售专用计算机系统。

现代流行和广泛使用的 POS 系统具有如下主要特点:

(1) 通过各类计算机网络与主计算机系统相连, 在 BMIS 中作为销售数据采集、预处理网络工作站, 既可以连网操作, 也可以单机操作。

(2) 具有商品编码 (PLU) 的存储、处理能力 (几千条至几万条不等), 可以外接条形码阅读器、信用卡阅读器、打印机等设备。

(3) 不但能对商品金额处理, 也可以对商品名称 (也可处理中文)、数量、库存处理, 并对所销售的商品进行分类查询、汇总和分析, 做出销售报表。

(4) 通过软件可增加、修改、完善收款机的功能, 适应各种销售环境和不断发展的功能要求。

2、收款机与商业管理信息系统的关系

前面已介绍了商业管理信息系统和收款机。不难看出, 两者之间有密切的关系。为了进一步说明这些关系, 可以概括为下面几点:

(1) 收款机是 BMIS 的销售数据采集工作站

收款机面对顾客, 将顾客的购物信息输入到收款机, 通过网络连接, 再送到主计算机, 由 BMIS 处理, 从而实现了销售数据的实时采集和处理。至于所采集的销售数据详细程

度, 则要看收款机的能力和档次了。

(2) 收款机是 BMIS 销售数据的预处理工作站。

BMIS 要求有每一商品的详细数据。这样就要求每一商品都具有本身唯一标识符, 通常用编码表示, 也称为 PLU。如果在商品销售过程中靠收款员输入这些商品编码, 则商品销售速度太慢, 而且难于记忆, 易出差错。针对这一问题, 发展了商品条形码, 即将商品在出厂时就编好码, 将其转化为一组粗细不等的条纹, 印在商品包装面上。阅读时通过条形码阅读器, 再转换回一组数字编码。这样, 既保证了商品编码的唯一性, 又免除人的记忆, 使得商品编码的阅读快速、可靠。收款机就要求能通过条形码阅读器读取商品条形码, 将其转化为数字形式的商品编码, 经处理后形成销售记录, 送回到主计算机中。

除条形码外, 收款机还能够对信用卡、支票等非格式化数据经过预处理, 转化成 BMIS 可辨认的格式化销售数据, 送 BMIS 进一步处理。

(3) 收款机是 BMIS 销售管理的工具

收款机在 BMIS 中的逻辑地位可以是一个柜组、一个部门, 甚至是一个分店。它除了可以快速、准确地完成顾客的商品销售过程处理外, 还可以通过收款机记录的销售记录, 由 BMIS 对每一分店、部门、柜组, 甚至每一营业员商品销售情况、业绩进行统计、分析与管理, 也可以对每一收款员销售货款进行准确的掌握与核对。当然, 也可以利用这些销售数据作销售汇总、分析与预测。

四、结束语

商业管理信息系统是商业大、中型企业的管理方向, 也是必由之路。而收款机则是商业管理信息系统必不可少的组成部分, 收款机的功能强弱, 直接影响到 BMIS 的功能。所以, 在考虑一个 BMIS 的时候, 应首先考虑采用什么样的收款机。

本文希望能对打算建立一个 BMIS 系统的人士, 提供一些帮助。

尽

管当前西方发达国家经济不景气,但是各类计算机系统的竞争激烈程度都是相当高的。计算机厂商面对增长不是很大的市场,都必须在技

术和工艺方面进行角逐,才能保证自己目标所能占稳的市场份额,免遭亏损和淘汰。这当中占份额最大的个人计算机的生产厂家竞相接连不断地降价销售,造成市场大滑坡,利润率下跌,给计算机界蒙上阴影。

鉴于上述形势,计算机公司纷纷大量裁员,改组机构,调整产品结构。无论哪一家公司,即使象 IBM 这样的蓝色巨人,也难于应付在开发新技术、新产品并将其推向市场过程中所承受的人力、财力、物力和时间上的巨大支出。许多公司捐弃前嫌联手结盟,以保证计算机产品的更新换代和占领市场。这里只举几个典型例子。HP 公司看准要进军工作站市场,而且以开放式系统为其赌注,开发 RISC(精简指令集)技术。她看好 Apple 公司占领工作站市场,因此购并了 Apple 公司。AT&T 公司原有自己的计算机公司,购并了 NCR 公司。而 IBM 公司与 Apple 公司、Motonola 公司联手,由 IBM 公司提出系统结构和系统,Apple 公司从事软件开发,而 Motonola 公司从事芯片加工,等等。

新一轮微处理器芯片的角逐

微处理器的单片集成度从 i4004 的 2300 个晶体管发展到 Super SPARC 的 310 万个晶体管,其集成度提高了 1300 多倍,平均每两年翻一番。生产加工工艺从 LSI 到 VLSI 再到 ULSI,微细加工线宽进入亚微米级,0.8 μm 生产已成为成熟工艺,现已进入 0.5 μm 工艺和 0.35 μm 工艺。单个微处理器运算速度已突破 100MIPS(即每秒 1 亿次运算)桌上亿次机已成现实。单个微处理器运算速度达到 1000MIPS 也不十分遥远了。

微处理器运算速度的迅速提高,主要归功于 RISC 技术和半导体生产工艺。到 90 年代初期 RISC 技术已进入的第三代。第三代 RISC 微处理器主要采用原来属于大型机和巨型机的三项技术:超长指令字、超级流水线和超标量。同时增加了更加复杂的控制和调度逻辑。这就保证了微处理器的性能不断提高。

表 1 列出了第三代 RISC 微处理器芯片各种性能指标,其中只有 Intel 公司的 Pentium 芯片(原来代号为 80586)是 CISC 技术与 RISC 技术结合的产品。

第三代 RISC 微处理器的时钟频率和芯片集成度都有极大的提高。Super SPARC 集成度达到 310 万个晶体管,Alpha 芯片的时钟频率达到 150/200MHz。助于软件设计与开发简单化。其中已有多种 RISC 微处理器同一型号有几个版本,这几个版本分别采用不同的时钟频率或系统配置,以满足不同的需要。例如 MIPS 公司的 R4000 就有三个版本:R4000PC, R4000SC 和 R4000MC。它们分别用于台式系统,高性能服务器和多处理器环境。预计从今年开始将有用 RISC 微处理器搭建的台式个人计算机或笔记本计算机问世;会采用 Unix 环境或 Windows NT 环境,这推动个人计算机的发展。

高速微处理器芯片在跨入 90 年代以后是个高速成长时期。

RS/6000(POWER)微处理器芯片是 IBM 公司 1990 年代推出的第二代 RISC 产品,

采用了当时本公司 0.5 μm CMOS 最新工艺,其主时钟频率为 41MHz。采用超标量技术,能同时并行最多执行 5 条指令。并强调了浮点部件的优化,使之浮点部件与整数部件具有相同的运算速度。该浮点部件是同时执行乘法/加

计算机技术和产品的现状与展望

钱承德

法操作的乘加法融合部件。RS/6000 采用多芯片 MPU 结构,共有 9 块芯片组成。目前,IBM 公司与 Motonola 公司,Apple 公司正进一步开发 POWER 芯片。

各著名厂商的产品详细参看表 1, 该表的数据表明许多芯片已采取超标量和超级流水线结构。

计算机产品更新换代加快

计算机的新秀——工作站和服务器,在各类计算机产品中销售增长最快。工作站是事务处理和计算机辅助设计与制造的得力工具,又是软件开发的得力工具,因此倍受企业和政府部门的重视,是目前科技和商业界用的桌上计算机的主要候选机种。服务器则是采用微处理器芯片,开放式系统体系结构,结合原来的小型计算机的优点而发展起来的。它具有很强的事务处理能力,特别是联机事务处理能力(OLTP),近年来又采用多处理器技术,不断增强其性能,业已成为政府和企事业单位从事各种类型的管理信息系统的主要手段。工作站和服务器均具有很强的联网能力。

个人计算机目前仍占计算机市场的很大份额。不过已有种种迹象表明,原有的桌上型个人计算机将会在 90 年代中后期逐渐会被工作站和各种便携式个人计算机取代。在便携式个人计算机中以笔记本型个人计算机发展最为迅速。笔记本型个人计算机是在 80 年代末期才兴起,而在短短几年已经历了 8088,80286,80386,80486,性能成倍提高,联网能力增强,特别是强化了电源管理系统,使其能工作较长时间。今后一、二年很多计算机厂家会进一步改善笔记本型个人计算机,并降低价格,使之成为适用各行各业使用的个人计算机。

多媒体(Multimedia)技术也是在 80 年代末期兴起的,是 90 年代计算机技术发展方向之一。目前多媒体系统已成功地进入了办公、商业、培训、教学、文娱等领域,随着图象压缩和声音压缩技术的发展,多媒体系统和板卡会在各行各业内广泛使用。并将使计算机与传统的电子设备(如电话、传真机、录音机、录像机、电视机、激光唱盘等)有机地融为一体。

总之,当前计算机产品朝着缩小化(Downsize)方向发展,具有很强的扩展性,根据不同用户的需要,实现完整的系统集成解决方案。

90 年代是计算机网络飞速发展的年代。提供良好的分布式处理环境,以便使用户获得逻辑上集中,物理上分布的计算机系统、数据库系统和磁盘系统等。

目前所采用的局域网基本上是两大类,一类是采用无屏蔽双绞线或同轴电缆的,传输速度为 10Mbps 的总线型拓扑结构的 Novell 网,LAN Manager 网,10BASE-5,10BASE-T 以及 ARCnet 网等,采用环形拓扑结构的主要有 IBM Token Ring 等;另一类是采用光缆作为使传输介质的 FDDI,其传输速度可以达到 100Mbps 以上。FDDI 将在今后几年有较大的发展。

客户服务器(Client-Server)体系结构是近几年发展起来的。计算机网络本身发展经历了集中式处理,即计算机主机采用分时结构,信息从主系统被传送到各个非智能设备;信息共享方式,即在计算机网络系统中,个人计算机或工作站可以作为独立的或相连接的处理单元,可以在同一个系统中完成处理和输出工作;和客户——服务器方式等三个阶段,在第三阶段它要求在一个单位内进行分布式处理,用户可以在网络中的许多节点上对数据进行存储和处理,并能够在分布式处理器中对资源进行存储、处理、管理和动态分配。

客户—服务器体系结构的广泛采用,反过来又促进了局域网的发展,特别是 FDDI 的发展。

目前广域网有以下几种:私有线路的专用网、租用线路的专用网、线路交换的公用数据网(CSDN)、分组交换公用数据网(PSDN)、综合数字网(IDN)和综合业务数字网(ISDN)。

ISDN 是一种由电话综合数字网络演变而成的网络,它可以提供各种服务所需的终端功能,用户可以通过有限的标准多用户界面,获得存取作业的支援。很明显,ISDN 是一种可以利用现有电信基础设施的标准通信环境,基于完全对等和可以互相交换的模式,在数据和语音之间进行呼叫的通信作业。

在 90 年代主要发展窄带 ISDN,到 21 世纪初将发展宽带 ISDN。而 UNIX 系统大量使用,其中包括 UNIX 通信连接功能,可以把一些局域网连接成更大的网络。这种软件产品的核心就是 TCP/IP。(传输控制协议/网际协议),它已成为目前网络之间进行通信的基准方法。

PCP/IP 是用于计算机网络之间互连及在众多异型计算机之间安排传输的协议集合。

TCP/IP 提供了可靠的传输,提供了各种网络实用服务的基础,其中包括电子邮件、文件传输和远程登录等。电子邮件是用来传输较短的正文文件,文件传输是用来传输较大的程序和数据文件,并且还提供安全检查以控制文件

的传输;远程登录则允许用户在远程机上登录并进行交互会晤。TCP/IP 所提供的互连与传输;主要是保证可靠性、可交互操作性、安全性和灵活性。

软件和应用系统日益完善丰富

开放式系统 (Open System) 和计算机辅助软件工具 (CASE) 第四代语言以及面向对象技术 (OOT) 是今后几年软件开发项目。

开放式系统将是今后数年计算机技术发展主要技术之一,是 90 年代的明星产品。开放式系统概念是从 UNIX 操作系统的发展而建立起来的,其主要特性在于基本上摆脱了具体硬件环境,实现软件的可移植性、可适应和可互操作性。开放式系统的概念和具体规定,特别是标准对所有人都是公开的,而且欢迎不同的厂商公司和个人参与其概念的形成,标准的制定。开放式系统相对于传统的、互不兼容的封闭式系统是一种新的体系概念,其本质特点是为应用软件提供一种统一的公共运行环境。

从当前来看,采用 UNIX 操作系统的开放式系统是一个非常成功的例子。它是以 UNIX 操作系统为核心的一整套支持软件和网络环境和工具,构成供应用软件和开发用的平台。这样一个软件开发与运行平台,基本上摆脱了具体的硬件设备,可以为应用软件提供一种统一的公共运行环境,极大地提高软件的可移植性和可复用性,保护了最终用户在软件和硬件上的投资利益,大大减轻了移植软件的负担,目前就有三个大的集团在发展 UNIX 操作系统,它们分别是 UNIX 国际公司 (UI)、开放式软件基金会 (OSF) 和先进计算机环境 (ACE),它们分别推出的 UNIX 最新版本是 UNIX System V4.0 版,OSF/1 和 ACE/SCO OPEN DESKTOP。

在 UNIX 系统基础上,相应地建立了一整套支持软件和网络软件、协议、它们有建立在 MITX 窗口系统基础上的 OSF/MOTIF 和 OPEN LOOK 图形用户界面;图形库软件 Open GL;局域网及其标准网络协议 TCP/IP;分布式处理客户—服务器结构;计算机辅助软件工程工具,第四代语言、面向对象的程序设计语言 (OOP);网络环境软件 (NES);数据库管理系统,如 SQL, Informax, Ingres, Unify, Sybase, Oracle 等关系型数据库管理系统,以及面向对象的数据库管理系统 (OODBMS) 和分布式数据库;图形软件,如 GKS, PHIGS 等;开放式网络系统,包括网络窗口系统,网络联接系统,开放式网络计算机 (ONC) 等。

程序设计语言正在从面向过程语言向面向对象语言的过渡。当然目前使用最多的还是第三代语言,例如 C 语言、FORTRAN 语言和 Pascal 语言等。第四代语言不仅仅是一种程序设计语言,而且还是一个较好的开发工具,以往各代程序设计语言都是面向过程的,仅适用于程序设计工作者,而第四代语言则面向应用对象,可使更多的用户直接使用第四代语言来开发自己的应用软件。

面向对象技术 (OOT) 在近几年发展比较迅速,特别是有一个专门组织——对象管理集团 (OMG) 专门从事面向对象技术的研究与发展,并制定相应的标准,以促进该项技术的发展与应用。目前面向对象技术主要从事面向对象程序设计语言 (OOP) 和面向对象数据库管理系统 (OODBMS) 的研究与开发,已有多种产品问世。例如, OOP 产品有 C++, Simula, Smalltalk, Common Lisp 等,面向对象数据库有 Intermedia 等。面向对象技术的好处在于开发者不限于程序设计工作者,还可以是广大计算机用户,以提高软件生产率,同时达到所开发的软件具有较高的可复用性。

计算机辅助软件工程 (CASE) 最初定义是指以个人计算机和工作站等工作平台,重点支持软件开发周期前端的软件开发工具,对用户需求分析和设计规格说明提供自动工具支持,而现在的 CASE 工具则必须支持整个软件开发周期,用软件设计规则说明直接自动生成软件,达到软件自动开。CASE 工具也将推广到其它机种,使该工具能支持自动代码生成和数据驱动技术,能提供项目管理功能和规划、评估、构造和工序控制等功能。以美国 IDE 公司 CASE 工具 Software thorough Pictures 为例,说明 CASE 工具在软件开发中的重要性,该软件包从 1984 年至 1991 年已开发出 7 个版本,从前到后依次是 Client/Server 结构版本,共享存储库的集成工具环境, X Windows 版本,异构网络浮动版本,目标管理 SQL 模式生成, OOSD 软件设计注解,多产品集成 CASE 环境。在软件开发中采用 CASE 工具可以改善软件开发方法和产品设计质量,同时,开发组织也将软件工程看作是改善其开发活动的可预测性和增强软件项目的合作者之间的交流。CASE 工具提供了对软件工程方法和自动化的支持,能够在质量、生产率、可预测性和交流方法带来益处。

CASE 工具, 也向图形和图象的开发方面发展, 例如 SGI 公司的 CASE Vision, 就有一组集成软件工具, 它们分别用于代码直觉化, 通过一个动态的直觉的视觉界面组合提供强有力功能方式, 进行静态和动态分析; 进行进程控制和组态管理等。

人机界面好, 今后会更有利于人们使用计算机, 朝着简单、容易和方便方向发展。除了简化用户命令, 加强交互功能外, 还要大力发展窗口技术、图形用户界面、光学文字识别、声音识别、语音合成和手写体文字识别等功能, 以创造非常友好的人机界面。

计算机要应用好, 除了上面各种因素之外就是应用系统了。应用系统是计算机应用成熟度的具体体现。主要应用系统包括文字处理系统、电子数据报表、财务会计系统、统计系统、办公自动化系统、情报检索系统、商情数据

库系统、从计算机辅助设计与制造系统和柔性生产线结合计算机集成制造系统 (CIMS)、电子设计自动化系统 (EDA)、集成化管理信息系统 (IMIS)、电子数据交换系统 (EDI)、实现无纸贸易、激光照排系统、图形与图象处理系统、地理信息系统 (GIS)、分布式计算机系统、联机事务处理系统、商业处理系统、金融处理系统、交通管制系统、航空订座系统、计算机辅助教学系统、城市规划系统、农业规划系统等各行各业用的应用系统, 以及采用人工智能技术实现的各种专家系统, 如面向医疗、勘探、计算机配置、旅游、战术模拟、运动生理模拟等。

正因为各种应用系统在各行各业的成功应用, 使得计算机系统在科技工程计算、数据处理、文字声音图形与图象处理、仿真技术和控制领域日益发挥巨大的作用。

表 1 高速微处理器芯片性能一览表

型号	厂家	字长	时钟频率 (Hz)	峰值速度 (MIPS)	工艺	晶体管数 (万个)	年份	备注
RS/6000 (power)	IBM	32	50	200	CMOS 0.5 μ m	120	1990	9 块芯片 700 万晶体管
80860xp	Intel	32	50	150	CHMOS 0.8 μ m	255	1991	
R4000	SGI-MIPS	64	100(内) 50(外)	100	CMOS 0.8 μ m	120	1991	
88110	Motonola	32	50	150 (63.7SPECmark)	ECL	130	1991	
32SF640	NS	32	33/40/50	100		110	1991	
C400	Intergraph	32	50				1991	
T9000 Transputer	SGS-Thomson	32	50	70 SPECmark		320	1991	
i960A	Intel	32	40	30SPECmark		57.5	1991	
PA-RISC 7100	HP	32	66	132	0.8 μ m		1992	
R4400	SGI-MIPS	64	150(内) 75(外)	120 80SPECmark	CMOS 0.8 μ m	120	1992	
Alpha	DEC	64	200	400 100 SPECmark	CMOS 0.75 μ m	168	1992	
Hyper SPARC	Sum/Cypress	32	55.5/80	52.SPECmark 85'SPECmark		110	1992	
Super SPARC	Sun/TI	32	50	100	BICMOS	310	1992	
Pentium	Intel	32	66	100	CHMOS 0.8 μ m	300	1993	CISCIR

ORACLE RDBMS V5.1 系统分析及应用研究

(上)

上海邮电管理局计算中心 沈欣 杨峰

上海科技大学计算机技术系 陈霞萍

【摘要】本文分四部分。第一部分详细分析了 ORACLE RDBMS 的软件结构体系。第二部分系统介绍 ORACLE 分区和 DB 文件的组织方式,着重讨论表空间的分配和管理技术,并深入研究了用于提高 ORACLE 系统性能的两大关键技术:索引和聚簇的应用技术。第三部分概要介绍 ORACLE 为实现查询优化的策略和措施。第四部分细致探讨了 ORACLE 为保证数据一致性的内部封锁机制,并提出查询优化的策略和措施。第四部分细致探讨了 ORACLE 为保证数据一致性的内部封锁机制,并提出了合理使用不同类型、不同方式 ORACLE 锁的原则和有效方法。

引言

ORACLE 是目前世界上最流行的关系型数据库管理系统,它采用 ANSI 批准的 SQL 数据语言,用 C 语言编写开发而成,具有广泛的兼容性和可移植性。由于 ORACLE 运行在不同计算机的不同操作系统下具有相同的用户界面,使用户在更换或升级系统时都不会影响已开发的软件。它还为用户提供了 SQL * FORMS、SQL * MENU、SQL * REPORT、SQL * CALC 等一批第四代语言工具在内的支持环境。另外,还具有很强的数据词典和决策支持系统等功能。ORACLE RDBMS V5.1 又是第一个开放型的具有分布式处理功能的数据库系统,它能满足硬件、操作系统和网络的独立性,使得不同计算机、不同操作系统、不同网络、甚至不同的 DBMS 都能集中在一个统一的计算机处理及信息系统之中。

1. ORACLE RDBMS 软件结构分析

ORACLE 软件系统是一个共享可重入代码系统。它是以共享映象的方式实现的,当启动映象时,共享映象被映射到用户地址空间。由于 ORACLE

代码存在于已安装的共享映象之中,所以不论何时,不论有多少用户在运行 ORACLE,内存中都只有一份 ORACLE 代码,即所有 ORACLE 用户共享同样的物理目标代码,多个用户可以同时存取系统中相同的表,甚至是行。

ORACLE 采用集中式的缓冲池,使得用户、数据库文件和前映象文件间的通讯都在 SGA 中进行。ORACLE 在内存中主要包括两大部分,即系统全局区 SGA 和 ORACLE 内核。而所有的 ORACLE 实用程序及工具软件,如用于启停和初始化 ORACLE 的 IOR: 用于数据库恢复的后映象日志 AJJ: 用于监视用户和 ORACLE 进程的显示系统 ODS 等等,都是驻留在外存上的,只有在调用时才会装入内存。在多用户系统中,还可有四个后台进程:前映象写 BIW、缓冲池写 BWR、异频提前读 ARH 和清理 CLN。它们构成了系统内部的恢复机制。下面详细分析 ORACLE RDBMS 中的重要数据结构和与这些存储结构相关的后台进程控制。

1.1 系统全局区 SGA

SGA 是位于主存或虚存中的一个共享存储区域,ORACLE 的四个分离的后台进程与用户进程一起共享着该全局区,并且同时映射到一个磁盘文件上,以便保存它不在内存中的页。SGA 是 ORACLE 运行期间数据活动的中心,它包含了控制 ORACLE 进程所需的各种信息,是一个跟踪 ORACLE 用户的公用资源的共享空间。这些资源包括 ORACLE 的可写缓冲区以及并发和字典控制。SGA 中的各部分大小由 INIT. ORA 文件中的参数设定,应该注意的是这些参数的取值因操作系统而异,实用程序 SGI 为 DBA 提供了用以检验由不同的参数文件所产生的 SGA 尺寸的合理性。

1.2 ORACLE 内核

ORACLE 内核负责解释并执行用户所要求的 SQL 操作,由它完成对 DB 文件和 BI 文件的读写,以维护 SGA 中的数据,协调多个并行用户之间的活动。

1.3 系统初始化文件

INIT. ORA 文件包含若干系统参数,当由 IOR 启动 ORACLE 时,要

读取和使用这些参数。这些参数值的设置将直接影响到 ORACLE 系统性能的资源。最好,把操作系统中控制虚存工作区的参数设定得与 SGA 的才相符。

1.4 数据库文件

ORACLE 数据库是由一个或多个 DB 文件组成的,它们包含了 ORACLE RDBMS 所存贮的或所产生的全部用户数据表和数据字典表。DB 文件由 DBA 用 CCF 实用程序创建,且每个系统中至少要有有一个 DB 文件存在。最大个数视不同操作系统而定。

1.5 前映象文件

BI 文件包含数据被改变之前的映象,即用来存储在更新操作(Insert、Update、Delete)中被修改的块的拷贝。另外,IB 文件还支持多个用户同时存取数据。BI 文件也用 CCF 创建,且一个 ORACLE 系统只能有一个 BI 文件,它所需要的绝对最大空间可能比 DB 文件的规模还要大,但是一般情况下不超过 DB 文件的 50%,这是因为几乎不可能一次修改整个数据库。BI 文件最小规模 4000 块。

BI 文件是一个“环形文件”,即当它的最后一个文件块被分配完后,ORACLE 重新又从第一块开始使用名块。这时称 BI 文件被重卷了一次。但这种重复使用是有限制的;除非此时在该块之前的所有块都已释放,否则仅释放而已。都达不到分配的目的。

1.6 后映象文件

AJJ 文件用业记录已经提交的事务,所有对 DB 文件作的变更都将存贮在 AJJ 文件中。当发生故障时,把它与数据库的务份拷贝结合起来使用,可使数据库恢复到最后一的正确拷贝。即实现“向前恢复”。(Roll forward)面使用 AJJ 的前提是定期把处于一致状态的数据库作备份,备份

中包含了所有的 DB 文件和 BI 文件的拷贝。

ORACLE 的 AJJ 文件由一组磁盘文件购成,它们可以是一些预先安排好的文件,也可以是一系列由 ORACLE 动态分配的特定尺寸的文件增长就很快,当日志写满一个文件后,再打开下一个文件继续写。在每个 AJJ 文件,但同一时刻只写入其中某一个文件。若数据库的更新负担繁重,则 AJJ 文件的首部都记有一个日志序号,在重启 ORACLE 之后用独立的 AJJ 实用程序按照日志序号的顺序,依次读出日志中的每个文件,并且用那些成为已提交事务的组成部分的已修改块,对该数据库进行更新。当完成日志的应用并把 BI 文件消除后,则可重启 ORACLE 使数据库重新活动,因系统故障而导致夭折的事务则被忽略。

1.7 数据字典

数据字典是 ORACLE RDBMS 最重要的成份之一,它是由 ORACLE 自动建立并及时列出新的一组表和视图。它们记录着用户标识。描述表和视图以及有关用户特权和数据存储的信息。这些信息不仅是数据库重果的可贵资源,为 DBA 用来监视 RDBMS 的使用并协助用户完成其应用,而且 RDBMS 又正是利用数据字典来记录修改以及控制要进行的工作。

1.8 前映象写进程 BIW

BIW 是唯一负责向 BI 文件写数据的进程,它把 SGA 中的前映象高速缓冲区中的数据块拷贝到 BI 文件中。当进行事务回退、CPU 故障或要求提供一致性的数据查询时,都将使用到 BI 文件中的拷贝块。

1.9 缓冲区写进程 BWR

BWR 是唯一负责向 DB 文件和 AJJ 文件写数据的进程,当需要为存

放新的数据块而不得不腾出缓冲区空间时,BWR 把修改了的块从 SGA 缓冲区中取出,写到 DB 文件中。同时也写入到 AJJ 文件中记录所提交的事务,此时对数据的改变才永久地保存起来。

1.10 提前异步读进程 ARH

当执行象 select 这样的命令做扫描查询时,就需要通过 ARH 从 DB 文件中把数据块读到 SGA 中,以便减少查询所需要的时间。值得一提的是:在 ORACLE 中 ARH 读块操作与所检索数据的那个程序的执行(用户进程)是并行的。

1.11 清理进程 CLN

CLN 周期性地扫描 SGA 进程信息,一旦发现某个进程异常终止,则临时借用该进程的 PID 行把它未完成的事务全部回退,然后注销该进程使与 ORACLE 断开。

2. ORACLE 分区和 DB 文件的组织

2.1 SYSTEM 分区

分区是 ORACLE 空间管理的一个逻辑单位,每个数据库系统在逻辑上是由若干个分区组成的,而且至少必须有一个。在 ORACLE 初始安装时,只有一个 SYSTEM 分区被创建,以后可以根据需要增加其他分区。每个分区必须对应至少一个真正存放 ORACLE 数据的物理文件,即 DB 文件,初始时 SYSTEM 分区只对应一个物理文件 ORA. DBS. 以后也可以增加新的文件到该分区中,这些组成分区的文件可以存放在不同磁盘的任何位置。

一个分区中可存放多个数据库表,一个表可以分跨在同一个分区中的不同 DB 文件上,但是不能跨分区存贮。用户在创建表时若不显示给出分区说明。隐含存贮在 SYSTEM 分区中的。通常表的索引应当与基表放在同一个分中,这样做的好处是当表

间作连接需要使用索引时或者通过索引检索数据时,将是方便迅速的。

在 SYSTEM 分区中存贮着: 数
据字典表、临时表、联机 Help 表、用户
表。

2.2 创建分区和增加文件

DBA 可以根据需要用 create
partition 命令创建一个新分区, 然后
再通过命令 alter partition 向该分
区中增加文件以获得空间。增加的文
件必须事先已经用 CCF 实用程序创
建, 且同一文件不能增加两次, CCF
命令中的 size 参数指的应是分配给该
文件的操作系统块数, 而不是 ORA-
CLE 块数。分区一旦创建就永久存在
于现有的 ROACLE 系统中, 不能删
掉。

由于在建表或向表中增加新记录
时, 都要在该分区中寻找一组其大小
能满足分配需要的初始数据和初始索
引区间或新的数据和索引区间的连续
自由块, 所以在自由空间中应尽量避
免存在一大堆较小的块组, 解决数据
库“破碎”的方法有三种:

(1) 用 CCF 创建新的 DB 文件,
然后把它加入到指定的分区中。

(2) 先利用 Export 把分区中的数
据卸出, 再用 Import 把数据重装入,
即使数据库重组。

(3) 把某些不常使用的表用 Ex-
port 卸出保存, 然后把这些表删除出
数据库, 当以后需要使用时再用 Im-
port 重装入。

DBA 用户可以通过 ORACLE 提
供的数据字典 SPACEMAP、PARTI-
TIONS、或建立用户视图方式来了解
当前数据库系统的分区、DB 文件和自
由空间块数的使用情况。

2.3 表空间的分配和管理

2.3.1 分区、表、段和区间关系

一个分区逻辑上是由若士个表组
成的, 而每一个表逻辑上又是由一些
段组成, 每个段物理上由一个或几个

区间组成, 区间又分为存放数据的数
据区间和存放索引的索引区间。

表是 ORACLE 数据加存贮数据
的基本单位, 当创建一个新表时, OR-
ACLE RDBMS 将自动为该表分配
一个初始数据区间和初始索引区间。

2.3.2 非聚簇数据块

非聚簇数据块的逻辑格式如图 6
所示。每个数据块中包含指向向前趋和
后继块的指针以及该块最近一次被写
入时的时间标记 (Timestamp)。再加
上 4 字节块头信息, 即物理数据块首
部共需 76 字节, 它们与另外 72 字节
构成了逻辑数据块首部。ORACLE 逻
辑块的大小加操作系统而异。大多为
2K 或 4K。

2.3.3 OARCLE 的表开销

每个表需要 2 个数据块及 1 个索
引块开销。第一个数据块是区间块, 其
中包含当前为建表占用的每个区间的
起始地址、最后块号、该区间中的块数
等信息。第二个数据块是数据字典块,
它含有从数据字典中取来的表定义信
息, 索引块记录该表的所有索引, 以及
当前正使用的区间的有关信息。

2.3.4 OARCLE 表空间的分配管理

每个分区都有一个分区自由块表
PFB。

用于记录该分区当前可用的数据
块。当使用 alter partition 命令为分
区增加一个 DB 文件时, 就将该 DB 文
件所包含的块全部登记在 PFB 表中,
以后当一个表申请新的区间时, 就将
若干块从 PFB 表中转到该表的段自
由块表 SFB。

当创建新表时, ORACLE 立即为
其分配初始数据段和初始索引段。而
当向该表插入记录行时, 初始数据区
间将填入到数据段的块中; 类似地当
对该表建立索引时, 初始索引区间也
将填入到索引段的块中, 三种 DML
操作即 insert、update 和 delete 将真
正地影响到数据块中的数据。

2.3.5 ORACLE 空间分配方式

“空间定义”的作用在于控制对一
个表分配数据空间的方式 (包括初值、
增量、最大值), 以及把这个表建立在
某个指定的分区中。每个表在建立时
都要引用一个空间定义, 以便控制该
表潜在的空间分配需要。空间定义的
建立方式既可采用由 ORA-
CLERDBMS 给定的缺省定义, 且一
经定义即可为所有用户引用, 供任意
多个表所使用。

2.3.5.1 default 空间定义

为了把一个表与一个空间定义联
系起来, 在 Createtable 命令建表时用
space 限定词指明。若建表时不指定
space, 则意味采用 default 空间定义,
因此该表将被建立在 SYSTEM 分区
中, 在建表后马上为该表的数据及索
引各分配 5 个 ORACLE 块, 如果该表
中的数据已填满初始的 5 块, 在插入
新的行时, 则要求为该表再增分一个
区间, 该增量区间的尺寸为 25 个 OR-
ACLE 块。

2.3.5.2 temptable 空间定义

temptable 空间定义是用来创建
临时表使用的。按缺省方式, 临时表也
被建立在 SYSTEM 分区中, 初始数
据或索引分配块数是 20 个, 增量区间
尺寸为 100 块等。该空间定义可为
DBA 改变, 但不可把该空间定义删
除。

空间定义仅仅是用来控制空间的
分配原则, 任何具有 RESOURE 特权
的用户都可以使用任何用户创建的空
间定义, 也可以改变甚至删除空间的
定义, 因此用户在使用一个空间定义
时最好从字典 SPACES 中了解当前
可用的空间定义分配情况, 也要避免
不同用户定义的空间名重名。

在用户建立空间定义时并不真正
分配任何空间, 而是确定分配的规则,
仅当用 Create 命令建表时, ORACLE
才根据其选定的空间分配方式为该表

第一次分配空间,以后随着对数据作 insert 或 Update 将按照该空间定义获得附加的空间。

ORACLE 虽然为用户提供了 Alter space 命令用于改变已有空间定义的分配方式,但只能影响以后的建表空间分配,至于原先的表仍旧按照原先的空间定义进行分配。除 temptable 外,用户能使用 Drop space 命令删去任何空间定义。

为合理使用空间定义,建议宁可很少几个大的区间而不要用一堆小区间,最理想的情况是让一个表的所有数据都装在它的第一个区间中;若原表中含有较多 NULL 值,则以后的更新会导致行加宽,因此 PCTFREE 应取大一些;在建表,增加数据或索引时,一定要保证有足够逻辑上连续的块用于数据及索引区间的初始分配;除非必须,一般不要指定 maxextents 值,而采用操作系统的缺省值 9999。

ORACLE 为用户提供了若干与空间定义有关的数据字典,下面用程序说明如何从这些字典中获得重要信息。程序五:查询 SPACES 列出当前存在的空间定义名及其定义的空间分配方式;程序六:通过查询 TABQUOTAS 给出所有用户表定义所引用的空间定义使用情况;程序七:查询 EXTENTS 反映当前用户表使用区间的情况,而 DBA 用户可以查询 SYSEXTENTS 看数据库中全部区间的使用情况;程序八:通过 STORAGE 反映 EXTENTS 中信息的一些总和信息;程序九:通过 TABALLOC 反映分区的总计信息,DBA 用户则可以从 SYSTABALLOC 中看到全部表的分配总计信息。

2.3.5.3 空间估算

合理地估算空间定义中的数据段和索引段空间,确定定义中的参数取值将直接影响到系统的性能构成。

2.3.5.3.1 数据段空间计算

估算数据段所需空间大小需小涉及诸多因素。

例如:每个表的行数;变长字段数据的平均长度;每个字段空值的百分比;会扩大行长度的修改量等。

$$\text{每行平均字节数 BR} = \sum_{i=1}^{n \text{列}} [(1 - n_i) \times (C_i + K + I)] + (L + Q), \text{ 每块平均行数 RB} = \text{INT} \left[\frac{(1 - f) \times (S - H)}{\text{BR}} \right]. \text{ 表所}$$

需的初始区间逻辑块数 $B = \frac{r}{\text{RB}}$ 。

其中: C_i 是第 i 列的平均长度; n_i 是第 i 列空值所占的比例; r 是表的总行数; f 是每个块中预留空间所占百分比; K 是列长度开销 (VAX 为 1Byte); I 是列标志开销 (VAX 为 1Byte); L 是行长度开销 (VAX 为 2Byte); Q 是行序号开销 (VAX 为 2Byte); S 是 ORACLE 逻辑块尺寸 (VAX 为 2K); H 是 ORACLE 块头信息开销 (VAX 为 76Bytes)。

2.3.5.3.2 索引段空间计算

估算索引段空间比较困难,因为它依赖的因素很多。例如:多少字段被索引;索引字段的长度;NULL 的百分数;建立的索引是压缩还是不压缩;表的数据是装入前还是装入后建立索引等。注意一个表的所有索引是存储在同一索引段中,且聚簇键也存在该索引段中,粗略估计,一个表的空间要求中约 20%—30% 为索引所要求。

初始索引区间块数

$$B = \frac{r \times 1.10}{(S - H) \times 0.75 / \text{KL}} \text{ 其中:}$$

r 是表中行数; S 是块尺寸; H 是块头开销; KL 是索引字段长度。若是压缩索引。 $\text{KL} = 10 + \text{索引字段数} + \text{索引字段长度之和}$; 比便因子 0.75 是已被使用的索引块的近似百分数,它假定 PCTFREE = 25。若为一个已建立的表作

估算,该因子取 0.80 即 PCTFREE = 20。

由于 PCTFREE 参数对于聚簇是无意义的,所以供聚簇使用的索引段空间分配应包含簇中所有表的全索引和聚簇键。应该尽量为每一个唯一聚簇链分配一个逻辑聚簇长。这是因为如果有更多的数据与该链有联系,则多余的数据就可能存放到链接块中,这显然背离了当初建立聚簇的目的。

2.3.6 行标识 ROWID

数据库表中的每一行都有一个逻辑字段 ROWID 值与之对应,它是该行的“地址”。ROWID 有三部分组成:分区标识,该分区中的块号、该块中的行序号,其中:逻辑分区块号表示此行在该分区中的逻辑块,它的前 5 位表示该分区中的物理块序号,后 3 位表示此物理块中的逻辑块序号。

ROWID 中的行序号部分与作为每行开销而存储的行序号是相同的,所不同的是 ROWID 值并不实际存储在数据库中,这样仅就不能对它作 Insert、Delete、Update,并且日常的记录更新或卸出/重装都会改变一行在表中的物理位置,从而改变 ROWID 值,但是,仍可以使用普通列名那样在 SQL 语句中使用 ROWID。

利用 ROWID 值中明显的地址意义,不仅能快速直接存取一特定的行,而且还可获得一个行级锁,以便删除两个完全相同的行。另外,通过 ROWID 还可获得有关表、块、记录间的重要信息。(待续)

401

高级 UNIX 连网技术讲座

广州昂立自动化工程公司 冯家宁

第三讲 NFS 和 RPC

3.7 管理

NFS 环境的管理从正确的安装设置和 NFS 与所有其它 RPC 服务依赖的平台的维护开始。大多数情况下,这是指 LAN 被物理上正确地安装, DARPA 网络软件已正确地设置,适当的文件已作编辑。

NFS 作为服务器在一具体机器上的安装可能涉及下面一个或更多的步骤。若机器作为 NFS 服务器运行,它必须被标识。服务器的磁盘驱动器必须定义,共享文件系统的位置必须在文件 `/etc/exports` 里指出。`/etc/exports` 文件也用于指定服务器提供或与客户共享的文件系统的存取权限。如果本地机用作网上另一台机的客户,本地的 `/etc/fstab` 必须编辑。此文件让本地机安装远程服务器的文件系统。

管理员可能需要在本地机上为文件系统建立安装点。安装点是指在 UNIX 文件系统层次内的目录。安装点可以是空的或满的目录。如果是满的, NFS 安装过程将有效地隐藏本地目录的内容。在某些情况下,这可以是保证即使在失去网络连接时本地系统也可以继续工作的办法。

一旦 NFS 在运行,本地机的运行方式可能需要改变。新的系统可能要使用本地机的数据而作为本地机的客户加到网上。新用户可能要用新系统而加到本地机。也许组织的或本地用户的安全级别需要改变。多数情况下,这些改变要求编辑本地的文件而在某些情况下要改变 NFS 服务器的运行方式。

一般地,作为一个服务器,本地主机必须移出一些网上其它系统能用的文件系统。这种移出是通过 `/etc/exports` 文件和 `exportfs` 命令控制的。本地主机配置为服务器是由文件 `/etc/exports` 的内容指定的。移出由 `exportfs` 命令来完成。这里的文件和命令在稍后作描述。

如果本地主机要服务于不同结构的机器,系统管理员要在适当文件系统的 `/usr` 目录里安装远程系统需要的程序的二进制拷贝。

不同的活动要求把本地机配置为一个 NFS 客户。首先要在本地机上建立适当的安装点。如果你用 `mkdir` 命令,系统管理员将在 UNIX 文件系统上建立目录作为安装点。无盘机器要有一组与本地机不同的安装点,但过程是类似的。安装点建立后,本地文件 `/etc/fstab` 要编辑以指定要安装的远程文件系统的位置。再次强调,无盘计算机系统的 `/etc/fstab` 文件与有盘的是不同的,但处理是相似的。`/etc/fstab` 文件的内容后面会作描述。

安装一个远程文件系统的过程是用命令 `mount` 来完成

的。大多数情况下,在 `/etc/fstab` 指定的远程文件系统的安装是自举时间通过 shell 文本完成的。最终用户永远不需要使用 `mount` 命令除非他们想改变或增加文件系统。

下面各节描述 NFS 和其它 RPC 服务在执行任务,配置时用到的命令,文件,驻留程序。注意厂家在为不同的计算机系统提供 NFS 时可能有更为方便的 shell 文本以简化安装和管理过程。这里假设没有这些文本提供。

为一个 NFS 服务器增加客户的过程有赖于客户是无盘的还是有盘的。只有最近版本的 NFS 才支持无盘客户,你可向 NFS 的供应商了解。增加一个有盘的客户是简单的。有必要在服务器上有一个超级用户。作为超级用户,你必须以根目录注册或用 `su` 命令把你的用户名改为 `root`。在成为服务器的超级用户后,你必须把客户的 Internet 地址加到 `/etc/hosts` 文件里而客户的 Ethernet 地址加到 `/etc/ethers` 文件里。在文件 `/etc/exports` 文件里决定适当的存取权是否授予客户。如没有,可以编辑文件。所提的三个文件的内容和结构的详细信息在下一节。

3.7.1 系统文件

下一步将讨论在 UNIX 环境下建立 NFS 使用环境所需的文件的使用和管理。十个系统文件对 NFS 运行是必须的或有影响。文件及其主要用途如下:

- `/etc/fstab` 描述要被安装的文件。
 - `/etc/mntab` 当前已安装的文件系统表。
 - `/etc/exports` 可供远程系统安装的文件系统表。
 - `/etc/xtab` 当前本地机已移出的文件系统表。
 - `/etc/hosts` 网上已知的主机表。
 - `/etc/netgroups` 网络范围的分组,用于远程安装的许可检查时作速记。
 - `/etc/inetd.conf` 当 Internet `inetd` 驻留程序收到有效请求时调用的驻留。程序表。
 - `/etc/protocols` 网络使用的有效的协议表。
 - `/etc/services` 网络服务表。
- `/etc/fstab`

在 NFS 术语里, `/etc/fstab` 文件用于描述哪一文件系统可被远程安装。`/etc/fstab` 文件正常情况下在自举时使用户可能用上的文件系统有效。`rc.boot` 文件通常也包含 `mount` 命令加一 `-a` 选项。这实操作系统试图安装 `/etc/fstab` 文件上的所有设备和文件系统。可以用文件方式隐藏一些不让最终用户看的网络管理文件,因为用户也许不会用到 `mount` 命令。除了

mount 命令, 此文件会被几个系统工具所用, 以决定它们执行操作的文件系统。

/etc/fstab 文件包含一组形式如下的条目:

```
filesystem directoryname type option freq pass
```

filesystem 是要安装的远程文件系统名。这里的格式是 remote hostname: remotepathname。directoryname 是要用做远程文件系统的目录路径名。对一个目录可使用两个路径名。如远程的目录 /usr/harry 在本地用时可用 /usr/remote/tom。

type 是文件系统的类型。这里我们只关心 NFS 文件系统, 所以此条目是 NFS。其它有效条目的例子是 4.2 和 swap。

option 是一个用逗号隔开的安装选项表。NFS 接受的选项与实现有关。不过通常几个选项在所有实现里都提供。如果一个 mount 请求第一次失败, 你可以让它重试。重试的过程可以在前台也可以在后台, 这可通过 bg 或 fg 来指定。另外重试的次数可用 retry=N 选项来指定。

另一组选项用于控制 NFS 对客户不能保持与服务器联系时的响应。retry=N 选项指定在服务器给出 "Server not responding" 出错提示前的重试次数。timeo=N 选项给出 NFS 在一个请求超时前的十分之一的的时间。soft/hard 选项告诉 NFS 在超过最大重发次数时做些什么。如果你想让 NFS 继续请求直到服务器回答, 把选项设为 hard。这有停止应用程序直到服务器回答的效果。而 soft 将让 NFS 返回一个出错信息给应用程序。通常这会让应用程序停止, 但它能让用户做其它的事。两种选择之间的折中是与 hard 一起设 intr 选项。这一服务器当用户在本地机上不活动时暂时无效。通过 hard, intr 选项用户回来时无需再启动他们的应用程序。

最后一组选项控制用户向一文件系统作写操作的能力。rw 选项告诉 NFS 以读写模式安装文件系统, 让用户有最大的权力使用文件, 无论是什么用户名分组名。ro 选项让 NFS 以只读模式安装文件系统。这使用户只能读或执行远程文件系统上的文件, 但不能对文件系统内的文件作写, 删除, 移动。

freq 指定倾斜的天数。对于 NFS 安装的文件系统通常设为 0, 它假定远程计算机负责维护备份文件。pass 告诉文件系统的检查工具 fsck, 它传达检查文件系统。通常 pass 设为零因为远程计算机系统也负责保证文件系统的统一性。

注意 /etc/fstab 中条目的次序决定 mount, umount, fsck 如何处理文件系统。每一工具都按 /etc/fstab 中的次序处理文件系统。例如, 安装一个文件系统, 然后安装另一个文件系统作为父亲将使第一个安装无效。以下是一个 /etc/fstab 的例子:

```
/dev/sd0a / 4.2 rw 1 1
/dev/sd0g /usr 4.2 rw 1 2
viper:/usr/luigi /sur/liugi nfs bg, rw, hard 0 0
viper:/usr/ct /usr/ct nfs bf, rw, hard 0 0
chacha:/usr/ops/bin /usr/ops nfs bf, rw, soft 0 0
server:/usr/games /usr/games nfs bf, ro, soft 0 0
/etc/mtab
```

/etc/matb 文件包含当前本地计算机系统安装的文件系统表。此表对管理员和想知道哪些文件系统是有效的用户是很有价值的。可以通过 getmntent() 调用在程序上使用这一文件。

/etc/exports

此文件包含本地计算机通过 NFS 提供给远程计算机的文件系统表。让一文件系统给远程使用的过程有时叫做移出。/etc/exports 文件决定本地计算机系统是否为一 NFS 服务器。通过 exportf 命令告诉 NFS 驻留程序哪一个文件系统可以接受

mount 请求。通常这在本地计算机自举时完成。exportfs 常常包含在 rc.local 的 shell 文本里。

/etc/exports 文件组织如下:

```
directoryname option, option ...
```

directoryname 是本地要移出的目录或文件的路径名。通常支持几种选择, 但它们与所用的实现有关。选项用于决定受权使用的类型, 有权使用具体文件系统的远程计算机系统, 对远程机上根用户和未知用户请求方式的处理。

ro 选项对所有远程计算机移出只读文件系统。rw=hostname 选项移出的文件系统让多数远程机只读而一些可读可写。hostname 可取 hostname:hostname:hostname... 的形式。安装的使用通过选项 access=client 规定。client 可以是一主机名也可以是一网络组名。网络组名用于称呼一组主机。每一 client 首先在 /etc/netgroup 里检查, 然后在 /etc/hosts 文件里检查。只有在 /etc/netgroup 或 /etc/hosts 里列出的客户才允许安装指定的文件系统。其他客户或者被排斥或者给一个默认的标识有限制地使用文件。

未知用户的请求可通过 anon=uid 选项来控制。如果存取文件的请求来自未知的远程用户, 本地系统将用 uid 作为存取标识。把 uid 设为 -1 就不让任何未知用户的使用。具有根特权的远程用户使用本地文件系统时也可得到控制。root=hostnames 选项给出可以使用根的根用户。根用户总被 NFS 认为是匿名的除非有 root= 的选项。

/etc/xtab

/etc/xtab 文件含有当前本地机移出的文件系统的表。本文件让程序决定哪一个当前有效的文件系统可被远程客户安装。当 exportfs 命令运行时, 这些信息就会放到 /etc/xtab 文件上。它会被安装驻留程序 rpc.mountd 用来决定哪一个远程系统可以接受 mount 请求。文件所含条目表的形式与 /etc/exports 一样。/etc/xtab 文件可以被程序通过 getexportent() 调用来使用。

/etc/hosts

此文件含有本地及大范围 DARPA Internet 所有已知的主机表。文件里每一主机一行, 包括主机在 Internet 的号码, 主机名, 主机的别名。文件被几个实用程序使用。exportfs 命令检查 /etc/hosts 或 /etc/netgroup 文件以决定在 /etc/exports 文件里命名的机器的 Internet 号码。exportfs 依次为 NFS 驻留程序, nfsd 提供 Internet 号码。

/etc/ethers

此文件有网上所有主机的 48 位 Ethernet 地址表。每一主机占一行, 形式是 x:x:x:x:x:x 和主机名。每一 x 是 0 到 FF 间的十六进制数, 表示网络地址的一个字节。主机名应与 /etc/hosts 文件所列一致。/etc/ethers 的目的是为程序提供机器的 Ethernet 实际地址。

/etc/netgroups

此文件是网络范围的分组表。这些分组在远程安装的许可检查时有用。文件条目有如下形式:

```
groupname member member ...
```

每一 member 包含三个字段——hostname, username, domainname。如其中有一字段是空的, 则任何东西都可与其配对。如其中有字段含有字母, 数字, 下划线以外的东西, 则没有东西能与其配对。特别是横线就表示如此。当一主机分在某些网络组里, 但此主机没有用户时, 这些表示是有用的。

/etc/inetd.conf

这是 Internet 服务器驻留程序 inetd 的配置文件。当客

户请求时, inetd 启动本地主机上其它服务的驻留程序。配置文件的格式是 inetd 通过插座收到 Internet 请求后要调用的驻留程序的清单。每行里有如下字段:

```
servicetype - of - socketprotocolwaitmode uidname - of -
daemon arguments
```

service 是被调用的驻留程序名。它必须列在文件 /etc/services 里。基于 RPC 的服务以相同的方式列出,但服务的版本号要用斜杠附加在名字之后(如, /1 表示版本 1)。type - of - socket 可以是 stream 代表 stream, 也可以是 dgram 代表数据报, raw 代表原始的, rdm 代表可靠传送, seqpacket 是序列包插座。protocol 是所用的协议。协议必须是列在 /etc/protocols 上的。一个 RPC 服务必须列为: rpc/协议类型。

waitmode 可以是无等待的服务,它释放插座直到超时。也可以是等待的,它占有插座。RPC 服务通常是无等待的。uid 是运行服务的用户名,或者是 root 或者是用户名。name - of - daemon 是驻留程序被调用的路径名,arguments 是传递给驻留程序的参量。

/etc/protocols

此文件含有所有用于 DARPA Internet 的有效协议。所有在 /etc/inetd.conf 里用到的协议必须在此列有。文件里列有协议名,正式的协议号,可能有别名。在 UNIX 里此文件将被一名字服务器所代替,如 Berkeley Internet Domain 服务(BIND)或 ISO X.500。

/etc/services

文件是一表,它列出在 Internet 或用户自己的网上有效的驻留程序。表里有服务,端口,服务所用的协议,服务的别名。象 /etc/protocols 那样, /etc/services 也会成为 UNIX 的古迹而被一名字服务所代替。

3.7.2 Sun Yellow Pages 名字服务

Yellow Pages 是 Sun Microsystems 的 SunOS 提供的一个分布式名字服务,用以简化计算机网络的管理。名字服务的功能类似于数据库。通常名字服务就是把名字转换为网络地址。

Yellow Pages 数据库也叫地图。地图代替了每台机上的本地文件,这样就不需要对每台机的文件进行更新。地图可以被 Yellow Pages 范围内的任何机器使用。这里的范围是一组 Yellow Pages 服务器。

Yellow Pages 是分布式服务,在一定的区域内可以有几个 Yellow Pages 服务器。其中一个作主服务器,其余的作从服务器。当主服务器的地图更新时,它会通过一个叫推出的过程把改变的信息传给区域内其余的从服务器。命令 yppush 用于把一个 Yellow Pages 服务器的地图让其它服务器使用。特别是,如把 yppush 命令放到文件 cron 里就能做周期性的更新。当然,网络管理员可以在任何时候强迫一次推出。

Yellow Pages 用地图代替了许多象 /etc/hosts, /etc/group, /etc/passwd 那样的本地文件,简化了 NFS 环境的管理。这样,如果网络的配置有变化,管理员只需在 Yellow Pages 范围内改动一次。Yellow Pages 地图的格式与它所代替的文件一致。只是为了更快地存取它用 dbm3 文件而不是 ASCII 文件。每一服务器的 YellowPages 地图都放在子目录 /var/yp 里。

Yellow Pages 的管理超出了本书的范围。但是, Yellow Pages 在网上的存在,改变了 NFS 管理员使用文件的方式。主要的区别在于,程序通常(也有一些例外)访问 Yellow Pages 地

图而不访问本地文件。Yellow Pages 使用时,某些 NFS 文件会有冲突。

/etc/hosts 只在启动时被访问。在 Yellow Pages 主机地图被使用后, /etc/ethers, /etc/netgroups 不再被使用。/etc/protocols 不再使用,因为 Yellow Pages 提供了同样的数据。/etc/services 文件也不再使用。

因 Yellow Pages 而被取代的有趣的文件是 /etc/passwd。程序将首先访问本地的 /etc/passwd。如果在文件中有 + 或 - 的条目,那么 Yellow Pages 地图将被访问。

在本章末尾讨论自动安装功能时, Yellow Pages 服务可以被自动安装服务使用以提供网上文件服务的位置。这样网上所有可共享资源都能被所有自动安装的用户透明地使用。

如果你对学习更多 Yellow Pages 服务有兴趣,可以参考 Sun Microsystems 的 Network Programming。

3.7.3 驻留程序

/usr/etc/nfsd。此命令用于启动几个 NFS 服务器上的 UNIX 驻留程序以处理客户的文件系统的使用请求。命令形式为:

/usr/etc/nfsd [服务器数]

服务器数是一台服务器打算处理的负荷量的函数。首要的规则是先四个驻留程序开始并在必要时向上调整服务器数。nfsd 驻留程序是用户应用程序进入基于核的 NFS 服务器的入口点。对多驻留程序的支持提供了并行处理并改善了整体响应时间。

/usr/etc/biod。此命令用于启动几个在一客户上的异步块 I/O 驻留程序。biod 驻留程序让客户向远程文件系统做提前读提前写的请求。biod 在客户上缓冲这些请求以改善 NFS 的整体性能。它的命令格式是:

/usr/etc/biod [驻留程序数]

参数调整的规则是以四个驻留程序开始,在必要时向上调。

/usr/etc/rpc.mountd。此驻留程序响应远程计算机的请求以安装一个本地系统。mountd 是基于 RPC 并执行三项服务的。首先, mountd 检查 /etc/xtab 文件以确定哪一个文件系统是可安装的。如果文件系统可被请求的计算机安装, mount 处理请求并让远程计算机提供所安装的文件系统的初始文件句柄来请求使用 NFS。最后, rpc.mountd 通过 showmount 命令提供哪一个文件系统被远程计算机系统所安装信息。mountd 驻留程序通常由本地机启动时的 rc 文本装入运行。

/usr/etc/inetd。inetd 驻留程序是一服务程序,它将在得知一个具体的服务请求时调用网络服务驻留程序。inetd 在开始时就读配置文件 inetd.conf。然后等待来自配置文件指定的服务的连接请求。这是通过标识服务与其通信的 Internet 端口来做的。无论连接请求出现在什么端口上, inetd 都会调用相应的服务。在服务启动时,配置文件会指定必要的传递参数。当服务完成工作后, inetd 继续等待请求。通过减少任一时间等待通信请求的进程数,此服务降低了任何机器上的负荷。服务在运行时也不断地考虑各种可选的方法。首要的规则是,如果一服务是被频繁地或定期地请求的,它就应在任何时候都保持有效。其它用得频繁的服务可通过 inetd 启动。另外,为使效率达到最大值,多 NFS 进程 (nfsd) 也保持有效。

3.7.4 命令

在 NFS 环境的管理和操作中,有五个主要的命令和四个基本的驻留程序。下面详细介绍这些命令。命令和驻留程序简要地列为:

* 驻留程序

— nfsd 用于启动多个处理文件的 UNIX 驻留程序。
— biod 用于启动一个客户系统的块 I/O 驻留程序。

— rpc.mountd 用于服务器系统上的安装(mount)驻留程序。

— inetd 用于其动 Internet 的 inetd 驻留程序。

* 命令

— exportfs 使一文件系统可被安装；它通常在启动时运行。

— mount 告诉一个远程的安装驻留程序安装一个文件系统。

— showmount 用于确定哪一些远程系统使用本地的文件系统。

— rpcinfo 用指定的 RPC 子程序检索网上的服务信息。

/usr/etc/exportfs. exportfs 命令用于移出或使一组本地文件系统能用作安装。此命令通常在自举时由 rc.local 文本来运行，它读文件 /etc/exports 上的信息并把一张当前移出的文件系统表放在文件 /etc/xtab 上。exportfs 可在任何时候由超级用户运行以改变移出表或移出目录的属性。命令有如下形式

/usr/etc/exportfs [选项] 目录名

简单地键入命令 /usr/etc/exportfs 将打印一张当前移出的文件系统。

选项 -a 移出列于文件 /etc/exports 的所有目录，-u 所有的目录都不移出，-v 用于冗长的操作，-i 忽略 /etc/exports 文件选项而用命令行的文件选项。exportfs 支持的选项与 /etc/exports 相同。

/usr/etc/mount. mount 命令把一个远程文件系统连接在本地目录结构的一个节点上。mount 命令的一般形式是

/usr/etc/mount [开关] [-t 类型] [-o 选项] [文件系统] [目录 W]

对于 UNIX 操作系统，mount 命令用于安装本地磁盘以及 NFS 文件系统。如果命令以 主机名:路径名 形式包括一个文件系统，那么命令 mount 将假定它是一个 NFS 文件系统。目录参数是远程文件系统连接(安装)在本地目录节点上的路径名。所有安装的文件系统都保存在文件 /etc/mtab 里。不带参数的 mount 命令将显示已安装的文件系统表。mount 有几个开关。最有用的一个是 -a。它表示安装全部。mount 将试图安装列于 /etc/fstab 上的所有文件系统。另一个有用的开关是 -p。表示打印已安装的文件系统表 /etc/fstab。此命令在用户安装了 /etc/fstab 中不含的文件系统时是有用的。已安装的文件系统表可以代替 fstab 文件。-v 表示冗长，-r 表示安装只读文件系统。-r 可代表 -o ro。

选项指定文件系统如何被安装(例如，类型，bg, fg)。选项的使用与 /etc/fstab 的方式相同。参阅 /etc/fstab 一节，其中有何使用这些选项的更多信息。

showmount. /usr/etc/showmount 命令用于确定哪一台计算机已经安装了一个文件系统到主机上。此命令对于系统管理员在关闭主机前想知道哪些远程系统正在使用主机来说非常有用。而且信息可以辅助调试。最后，信息可以以记帐的形式记录下来。

命令有形式

/usr/etc/showmount [-abc] [主机]

主机的默认值是本地的主机名。-a 开关告诉 showmount 用 主机名:目录 的形式显示所有远程的安装。-d 开关只打印远程安装的目录。-e 开关打印所有移出的文件系统。

showmount 通过远程过程调用与指定主机的 mountd 驻留程序通信以决定所需的信息。mountd 将维护这些信息并在文件 /etc/rmtab 损坏时保存这些信息。

rpcinfo. 此命令用于获得基于 RPC 的网络服务，如 NFS 的信息。每一 RPC 网络服务都有一个唯一的程序号和版本号。用三种方法之一，rpcinfo 命令可以与网上的任何服务通信，确定服务是否运行。另外，rpcinfo 可以确定服务是否在正确的主机上运行。

rpcinfo 用的第一个方法是用 RPC 通过 UDP 或 TCP 调用指定主机上指定程序的过程 0。任何 RPC 程序的过程 0 是一个空请求过程，它只返回 RPC 服务是否在运行。这样用户或管理员可以知道所问的主机是否提供给定的服务。这一方法的语法是

rpcinfo -u/-t 主机名 程序号 [版本]

这里 -u 代表 UPC，-t 代表 TPC。

有时候用户对知道一台主机上的所有服务有兴趣。这可以用 rpcinfo 查询主机的端口图。每一 RPC 服务在出现时总要在端口图上注册。这样，端口图就会返回一张主机上所有 RPC 服务的表。此方法的语法是

rpcinfo -p 主机名

最后，用户也许想知道所有计算机在提供一个给定服务时的能力。你可以让 rpcinfo 在 UDP 传输层发送一段 RPC 广播来获得这些信息。所有与程序和版本号配对的 RPC 服务都会答复广播的请求。多数情况下，系统管理员用此命令确定所有的机器是否在运行一项特定的服务。RPC 广播的语法是

rpcinfo -b 程序号 版本号

如不指定版本号，rpcinfo 将寻找指定 RPC 服务的所有版本。

记得 NFS 事实上是一项 RPC 服务并将回答 rpcinfo 对信息的请求。

nfsstat. 系统管理员和其它用户用 nfsstat 命令获得一些本地机上 NFS 和 RPC 的 UNIX 接口正在使用的统计信息。命令显示服务器或客户的统计信息。此命令虽然对所有的用户都可用，但实际上是为系统管理员而设计的，因为所显示的信息给出了本地机在服务请求时的效率。大量请求的失效说明低层通信机制的失效(网络接口，网络软件等)或是客户方面有问题。

nfsstat 显示 RPC 调用和接收的次数及失效的 RPC 调用和接收的次数。RPC 的超时信息。RPC 等待与远程服务通信的频度。这两组信息在确定网络及本地系统的性能时是有用的。与其它管理命令一起，nfsstat 可以作为网络管理员的非常有用的工具。

3.7.6 性能和可靠性

安装问题 mount 过程至少包括 13 个不同的步骤。对这些步骤已经作过介绍。由于 mount 过程是关键的并且在大多数情况下应与用户隔离。此过程出现的任何问题都会导致混乱。多数 mount 的问题是由于不适当的安装和配置所致。只有少数情况下要设置和修改安装的基本配置。在安装时，通常设置是正确的。问题往往出在要修改配置的时候。

多数出错是 /etc/fstab 的配置不正确。或是要安装的文件系统列的不正确或是指定的选项不正确。其它的出错可能

是指定的远端机上没有文件系统或作为安装点的目录在本地机上不存在。

在网络正常运行时也可能出错。可能是有规律的也可能是无规律的。如果服务器死了或因其它原因而不可用,会有“server not available”的信息。这就有助于故障的排除。但是远程主机上的安装驻留程序或 NFS 驻留程序的失效必须经过从新起动才能使用。

程序挂起 记得 NFS 文件系统可以是硬安装也可以是软安装。区别在于应用服务器的在出错时对应应用程序的影响。由于 NFS 是无状态的,服务器的死机然后从新起动对应用程序没有不利的影响。当然响应时间是例外。如果一个 NFS 文件系统是硬安装的,在服务器死机时,对文件进行操作的应用程序会挂起直到服务器恢复。这对某些用户来说可能是个问题,因为一些重要的应用程序在他们的主机上挂起了。文件系统的软安装使应用程序有等待超时然后退出的功能。这使用户能执行其它任务而无需等候停机的服务器。

如果在文件存取时挂起应用程序,可以检查服务器文件系统有否重新安装上。如果服务器正常而应用程序仍然挂起,有可能是远程 NFS 驻留程序的问题。系统管理员可试撤消 NFS 驻留程序并从新起动它们看能否改善应用程序的运行。撤消 NFS 驻留程序并不会给文件系统的用户带来不良影响,因而这是一个安全的选择。如果情况没有改善,就有可能网络的问题使客户无法使用服务器。下一步可作网络的故障排除。

性能 许多因素影响 NFS 的性能。服务器磁盘子系统的容量和性能是最重要的。服务器磁盘控制器的存取速度越快, NFS 就越快。减轻服务器的负荷也是有用的。保持服务器上的应用程序负荷在合理的程度上使 NFS 有足够的 CPU 时间。影响性能的另一个因素是网络及客户上的块 I/O 驻留程序。如果网络是 Ethernet,在性能显得有点慢时,运行 netstat 实用程序是有用的。netstat -i 命令将显示本地主机向电缆的重发次数。重发率大于 5% 说明本地接口或以太网接头有毛病。高重发率也可能由网络的过分拥挤引起的,但正常情况下,重发率高于 5% 是由接口引起。

永远要避免网络的超负荷。一般来说有几条可用的规律, Ethernet 段的节点数越少越好。通常 50 台主机就足以使网络慢下来。网络的分段应优化。块 I/O 驻留程序 (biod) 用于处理文件的快速缓冲。在性能下降时,检查本地 biod 驻留程序是否挂起。撤消驻留程序并从新起动它们可以解决问题。

NFS 和其它类型网络服务的交互作用并不明显。从理论上说,几个应用程序会增加网络的负荷并影响 NFS 的性能。网络窗口系统如 X Windows 是设计在分布式环境下支持交互图形应用程序的。特别是 X 系统让应用程序位于服务器上而屏幕绘制和与用户交互作用部分位于远程设备如工作站上。这种设置丰富了功能,但加重了网络的负荷。这时所给的节点数应取得更为保守。

安全性 NFS 环境的安全性是双倍的。首先, NFS 用 UNIX 证书保持和加强了网上的兼容性。对此我们在前面讨论安装和 NFS 文件使用请求的处理时作过介绍。实际上 NFS 服务器要求每一 RPC 报文都传递 UNIX 证书。UNIX 证书包括有用户 ID、用户组 ID 及用户所属的组表。NFS 服务器保存它自己的证书,在远程用户要使用文件或文件系统时比较他们的证书。这样就保持了 UNIX 的安全性。非 UNIX 机器的客户在使用 UNIX 服务器时也必须建立 UNIX 证书。这可以是提交一个文件的内容或接受默认的用户 ID nobody。

正常使用传递 UNIX 证书的方法时,远程用户不能使用

本地主机根目录的口令。否则这将是用 su 用户名命令冒充另一用户的很容易的事。这命令把有效的用户 ID 改为用户名,让远程用户逃过 UNIX 认证。NFS 的最新版本 BFSSRC 4.0 有一项加强网络安全的新功能,它用新的 RPC 安全机制传递请求者真实的用户 ID 而不是有效的用户 ID。此方案还进一步认证 NFS 客户。通过这种认证服务器得以确保所传递的真实用户 ID 来自可靠的地方。这就填补了 su 用户名命令安全上的漏洞。关于 NFSSRC RPCSRC 安全机制的详细讨论可参阅 Sun Microsystems 的文件 SecurityFeatures Guide, part Number 800-1735-10。

3.8 未来的方向

3.8.1 对无盘工作站的支持

使远程文件系统好像是在本地的 NFS 机制具有比只是远程数据存取更广的应用。它可使无盘工作站也有根和对换分区。无盘工作站的配置是 CPU 主板、网络卡、实存 RAM、还有在 ROM 里的少量自举代码。自举代码主要用于寻找服务器,从中加载一份操作系统和连网代码的拷贝。不同的机器有不同的 ROM。在某些情况下可以把某一版本的操作系统的内核写在 ROM 上。

但在 UNIX 的情况下,设计是让操作系统从磁盘调入主存。这种调入也叫自举,因为机器必须解决在没有操作系统的情况下如何把操作系统装入内存。在 NFS SRC4.0 里 Sun Microsystems 增加了 NETdisk 的机制,它用于起动无盘工作站。一旦操作系统载入客户与服务器的分页对换都通过 NFS 来完成。客户的机器可以与从中自举的服务器机器不同。例如,一台 UNIX 的无盘工作站可以从运行 VMS 的 VAX 上起动。这样做的优点是重新使用现有机器来支持新的机器。而且系统管理人员只需要熟悉一种无盘工作站的使用。

NETdisk 机制依赖于一个叫 bootparams 的 RPC 服务。此服务由 inetd 或用户调用,并用配置文件 /etc/bootparams 上的 Internet 地址响应主机的自举请求。bootparams 服务为客户系统提供其主机名以及有关根分区,倾泻分区和对换分区定位的足够信息。这些分区可定位于网上的任何地方并可由无盘工作站通过 NFS 存取。无盘工作站由通常的自举机制载入内核。然后内核通过 bootparams 服务对话找到其根,对换,倾泻分区。

这一 NFS 的新应用有有趣的支持和管理上的分支。而且许可证的签发也变得麻烦,因为大多数操作系统的软件使用许可证只对一种具体类型的计算机系统。把操作系统存在不同的机型上可能违反了许可证协定。可以预料此问题要花一定的时间才能解决。

3.8.2 NFS 3.0 版协议

NFS Version 3.0 是计划对当前 NFS 协议的修改。由于 Sun Microsystems 是 NFS 的所有者。只要它愿意,它就可以修改协议。但是由于 NFS 已有广泛的使用环境,只有在公众都会接受的情况下,它才会修改协议。

如果你想学习更多有关对 NFS 的修改建议,可与 Sun Microsystems 联系:

NFS Administrator
2550 Garcia Avenue
Mountain View, CA 94043

修改建议大多是厂商和用户的要求。修改保持 NFS 的原有目标:一种简单的,透明的文件使用方法。客户现在可以为自己增加特殊的功能。这并不意味着这些实现会是非标准的,它们只提供超出目前默认的功能。

例 1:

故障现象: M2024 打印机, 开机后, 除底座风扇转动外, 无其它动作, 面板上所有的指示灯不亮。

故障分析与排除: 底座风扇能转动, 证明交流供电正常, 变压器里的热熔断丝完好。打开机壳, 发现 +36V 电源部分的保险管 F3 (4A) 已烧断。把接口板从 J11 插座取下, 拔掉控制板上的其它插头, 保留 P10 电源插座, 用万用电表测 +36V 和 0V 之间阻值, 出现短路。从控制板电路可知, +36V 电源主要是驱动打印针, 直流电机和步进马达。由于保险管 F1 (2.5A) 完好, 电

机驱动电路正常。为确定短路点出自哪部分, 拔去保险管 F1。断开电机驱动电路, 再拔去打印头供电插头, 仅剩下电源部分, 在 F3 处换上好的保险管, 加电保险管仍烧断, 证明短路出自电源部分。从电源部分分析, 因击穿能直接导致 +36V 短路的元件有: 整流管 DB2、可控硅 SCR2 和滤波电容 C47 等, 逐个检查, 结果发现, 在打印头供电插座 J8 旁边的电容 C70 (50V, 4.7μ) 被击穿, C70 与电容 C47 并

联, 共同对 +36V 电压提供滤波。更换 C70, 打印机恢复正常。

例 2:

故障现象: EPSON FX-85 打印机, 开机后, 打印头复位, 缺纸时, 测纸器报警, 面板显示正常。几十秒后, 变压器发出响声, 出现过载现象, 且有烧焦味。

故障分析及排除: 变压器过载, 主板上元件工作异常, 打开机壳, 在变压器初级线圈一侧的保险管完好。拆下主板, 检查各个元件, 发现三极管 Q11 (D1392) 接脚处被烧焦。取下三极管 Q11, 用万用表检查, Q11 已损坏。EPSON FX-85 型为早期的 9 针打印机, 从主电路可以看出, 9 根打印针由 9 片 D1392 控制, 编号为 Q6-Q14。Q6、Q9、Q12、Q7、Q10 控制奇数针, Q14、Q11、Q8、Q13 控制偶数针, 检查其余的 D1392, 均正常。D1392 为大功率、高耐压的开关硅管, 市面上难以购买, 故使用功率和耐压相近的三极管 C2373 替换, 打印机恢复正常。

403

故障现象: 一台 AR2463 打印机开机无

反应无复位动作 (打印头字车不动) 但电源指示灯亮。

处理过程: 因为电源指示灯亮估计电源没问题。经检查, 各组电源电压正常。

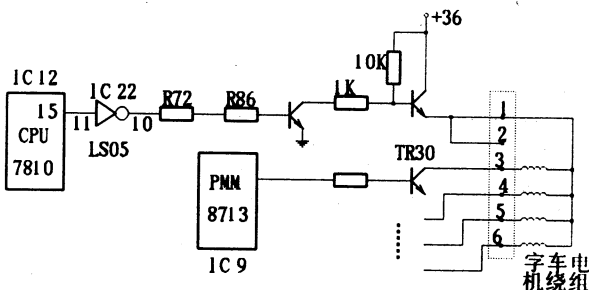
根据字车不动这一现象, 首先看 +36V 电源是否加到字车步进电机上。字车驱动电路简图见附图。用数字表测量 TR35 集电极对地电位接近 0V, 又测发射极电位为 -36V 正常, 说明 TR35 截止, 遂查前级 TR34 集电极电位为 +36V, 说明 TR34 也截止, 而正常情况下应导通, 这样才可使 TR35 导通, 字车电机得到 36V 电压而运转, 现在 TR34 截止, 于是怀疑 74LS05 反相器 10 脚输出, 测量发现 11 脚输入为高电平时, 输出 10 脚为低电平, 但是 11 脚

Star-AR2463 打印机维修一例

唐山钢铁公司职工大学 温波

为低电平时, 10 脚仍然是低电平, 说明 11 脚

和 10 脚之间的反相器已损坏, 更换 74LS05 后, 故障排除。另外 AR2463 与 AR3240 不一样, AR2463 有上盖检测开关, 不盖上盖, 打印机不动作。



404

也使九针打印机字变更清晰

广东清远市华美洁具有限公司 向勇飞

九针打印机因其价格便宜,与中华学习机、PC机都兼容,配合适当的软件可打印中西文,为此一般家庭电脑用户都喜欢配置一台九针打印机。目前市面上出售的九针打印机牌子、型号很多,其各项技术性能也有差别。本文主要介绍更换 KY-980 打印机打印头,使 KY-980 打印机打印的汉字更加清晰,还可使用双面色带芯。

KY-980 打印机的打印针设计成片状针,使用日久后,打印字的笔划就会变粗,特别是打印小号中文字时,字迹便不清晰,为此一些采取打磨打印针头使字迹清晰的文章也就见诸报端,但只是权宜之计。另又因原装的色带芯为单色带,市面很少有售,给日后更换带来麻烦。须然可用双面色带芯(如 EPSON LQ-1600K 色带芯)代换,但由于使用双面色带芯后,色带的油墨很容易沾污打印针片而造成打印针出针不畅,使打印的字符残缺不全或漏点等故障。为此,笔者认为要彻底解决上述问题就要更换打印头,以下介绍用 EPSON LX-850 九针打印头替代 KY-980 打印头的方法:

EPSON LX-850 九针打印机的打印头取用直径为 0.20 的钢丝做打印针,用来替代 KY-980 打印头效果很好。

首先到电脑商行去购一个 EPSON LX-850 九针打印头。按(图 1)用金属或硬塑料做一个打印头座。将 KY-980 打印机的打印头拆下,然后把自制的打印头座装到原 KY-980 打印头的位置上,旋紧下面两颗螺丝。即可将 LX-850 九针打印头安装在其上面。

由于 LX-850 打印头接口与 KY-980 打印机打印头电缆接口不兼容,在安装前须对 LX-850 打印头接口进行改造:

两种打印头接口功能对照见(表一),改造时可用锋利的小刀将 LX-850 打印头的接口与各针驱动线圈引出线间的印刷线路(见图 2 打×处)切断。参照(表一),用 $\Phi 0.31$ 的漆包线将 LX-850 打印头各针驱动线圈的引出线与接口脚相连接,焊接好后,即可进行安装。先把 KY-980 打印机打印头电缆插到改造后的 LX-850 打印头接口上,由于该接口为 12 脚,而电缆为 11 脚,插时切勿插错,可参见(图 2)。最后将 LX-850 打印头装到自制的打印头座上,旋紧安装螺

丝,即可试机打印,如打印的字符浅色或深色,可将打印头作前后调整。使打印的字符清晰即可。

KY-980 打印机经过更换打印头后,使得用小号字打印笔画比较多的汉字也很清晰了。

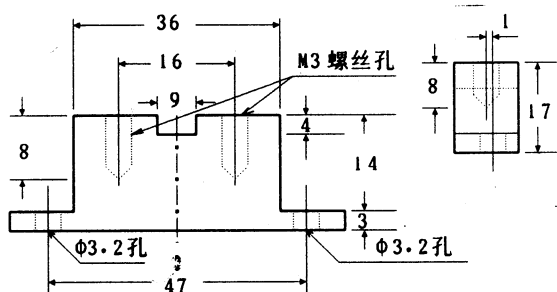


图 1

KY- 980 打印头接口插座脚号	1	2	3	4	5	6	7	8	9	10	11	(1 脚与 11 脚为公共脚)	
LX- 850 打印头接口插座脚号	6	7	2	5	3	1	12	10	9	11	4	6 7 8	(6、7、8 脚为公共脚)
打印针编号			2	8	6	4	3	5	7	9	1		最上面的针为 1 号针，最下面的针为 9 号针)

表 1

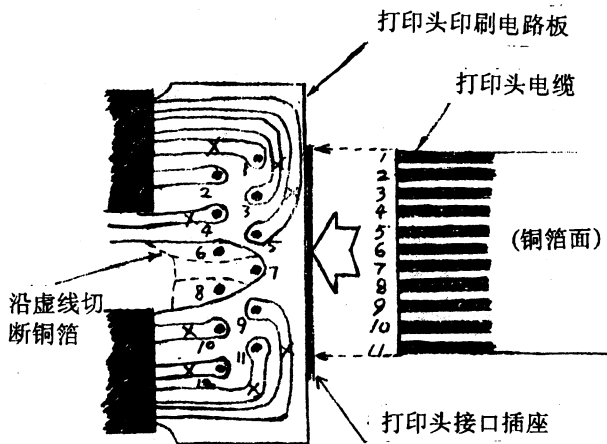


图 2

四通 MS—2401 打字机断针的预防

* 蔡长安 *

及断针的更换

四通 MS 2401 打字机是现代办公室较常的设备之一。我们在使用一段时间后,发现打字机的打印头如使用不当,就会经常发生断针,断针后则影响打印质量,重则整机无法使用。因此,必须采取适当的措施预防打字机断针。本人经过使用实践,总结出几种预防打字机断针的方法。

1、经常清洗打印头,保证打印针击打的灵活性。MS 2401 打字机在打印文件时是按最短距离进行双向打印的,即打印头向左和向右移动的同时都可以进行字符打印,然而色带当向右移动,而打印头向左移动并进行字符打印时,打印针很容易挂住色带,因此时打印头与色带的相对运动速度较快,打印针打在色带上时还未来得及退回,而色带已经开始了运动,如果打印头上污垢(污垢大多是由细小纤维丝和色带上的油墨及蜡纸上的脏物组成。这些污垢随着打印针的不断动作卡入打印针出口孔中去。这些附有粘性的污垢就是影响打印针正常动作的因素,气温越低,它的粘性就越大)太多时,打印针的击打速度慢,就更容易挂住色带,又由于打印字符时,出针率较高,故极易挂坏色带,甚至造成断针。清洗打印头的方法是:将打印头从托架上拆下,先将打印头前端的油污用酒精擦拭干净,再将打印针的导杆拆下,露出打印针上半部,然后用镊子夹一团棉球对打印针进行清洗。清洗时应将打印头的前端朝下清洗,着重清洗针尖附近,特别是针孔部分,将打印针逐根擦拭,但不要上下扳动打印针,以免使针尖弯曲。反复清洗几次,直到清洗干净为止。注意:不要让清洗用的酒精流入打印头内部。

2、要根据打印纸的厚度调整打印头与打印字辊的间距。四通 MS 2401 打字机的左内侧有一个纸厚调节扳手,共分四档位置,用它来可调整打印头与滚筒之间的距离。间距过大,字迹浅、噪声大容易断针;间距过小,字迹模糊、纸面易脏也影响色带寿命。1 档位置纸张最薄,4 档位置纸张最厚。运用普通纸时应将调整杆位置设定为设定为“2”档,2 页纸时为刻度“3”,纸越多档次也要适当加大。但当超过规定纸的限度时,不仅打印速度减慢,而且会缩短打印头的寿命。

3、经常擦拭色带盒内被动轮与驱动轮,以便色带运转灵活。当两轮之间油墨较多时,就会使摩擦力减小,甚至产生打滑,而带不动色带,这样就容易挂坏色带。

4、要使用质量好的色带。在选择色带时要重点看看接头处是否平滑,不平滑的

色带最容易在打字时别针,挂住色带,造成打印头断针,同时由于打印头位置发生了变化,还会使得被打印的文件整个报废。对于打印中色带出现皱褶的现象,可从盒中取出色带,放置桌面摊开,用烧热的电熨斗烫平,再置入盒中,就可用了。

5、运行中不能扯纸。由于打字机一般都是高速双向打印,如在运行中扯纸,就会造成针在接触纸的瞬间,针回不到原位,被强行拉断,如需扯纸可按“取消命令”键,使之暂停打印,然后再进行扯纸操作。

6、不要使用折皱和撕裂的纸。打印纸的优劣,关系到走纸的好坏。如走纸不好,就会在打字头处重叠,影响打印。上纸要帖紧,可用手拨动走纸轮走一走看纸上得是否合适。如走纸不好,也会造成打印针挂住色带,严重时还会造成断针。

7、当色带使用到一定程度后,在色带表面就会起毛,这时就不宜再使用了,应及时更换新色带,这样做既可减少打印针的磨损,又可避免发生断针现象,还能提高打印的质量和效率。

总之,采取适当的方法,可以避免打字机断针的经常发生。此种预防打字机断针的方法同样适用于各种打印机。

虽然我们在打印中力求避免发生断针,但随着打印次数的增多,还是可能发生断针的。由于该种打字机的打印头结构较其它打印机的打印头简单,拆装也比较容易,我们完全可以自己拆换打印针,现把本人摸索的一套更换打印针的方法介绍给广大读者。

四通 MS 2401 打字机的打印头主要由两部分组成,一是后盖,另一部分是打印针套部分。打印针套部分有 24 根打印针(每根由针杆和针把焊成近似直角形状,其直径为 0.2mm。24 钢针呈锯齿排列,纵横间距为 1/180 英寸)、打印针固定座(一块圆形金属圈上有 24 个凸进去小孔,供放打印针针把)、电磁驱动线圈部分、三块塑料制成的打印针导向板(注:第一和第二块导向板有正反面之分)。第一块导向板有 24 个小圆孔,并组成一圆形,第二块导向板(处于中间部位)也有 24 个小圆孔,并组成一椭圆形,第三块导向板是打印针最前面的一块塑料板,有两排平衡的小圆孔,每排有 12 个小圆孔。第一、二块导向板是活动的,所有打印针被拆出后这两块导向板可取出。一般的方法是取下断针后再插入新针,但据笔者的经验,发现此法插入较难,容易与其它针绞在一起,可按下列方法更换打印针。

1、用力扳开打印头上的橡皮外套,即可露出整个打印头部分,用十字起子拧开前面的前盖上的四颗螺钉,再拆开后盖的三个固定螺钉(有的是二个螺钉),即可拆下前后盖。这时可看到中间有几个中心圆垫片及透明橡胶圈,取下这几个中心圆垫片及橡胶圈。在边上有一个塑料环形的薄膜垫片固定在针座上,拆下这个薄膜垫片,可看到下面是固定针座,固定针座通过两根螺钉与下面的部分相固定,24 根针被镶嵌在针座的凹入部分,并呈圆周形状排列。

2、从打印针套的第三块向导板前部可以判断断针所在的排以及在排中所处的位置,确定其位置后,可以用镊子把从断针至开口处的几根针全部取出。

3、把完好的针从断口处依次沿原针位置的三个导向孔穿回,穿进时可在开口处观察针头插入情况,防止错位或者与其它针绞合。当针嘴穿到第三个向导孔时,由于它不象前两个孔一一对应的,故可用一根大头针沿第三块向导板的后部针排方向并靠近该排打印针左或右稍托一下,打印针可穿进第三个导向孔。换针的工作是一项十分细致的工作,要有耐心。对于初换者往往由于开始换得不好而失去耐心,乱捅硬压,从而使新换的针还未装进去就再次弄断,这是换针时一定要注意避免的。

4、以上步骤完成后检查打印针是否正确,按拆下的步骤的相反次序装好胶圈、垫片和后盖,再套好橡皮外套。

5、在换针时还要注意以下几点,一个是切忌把拆盖后的打印头头朝上放置,以免其它针掉下来,若要头朝上时,一定要用拇指把所有针把按住;再一个就是换针时若其它的针已经掉下来,则可先将掉下来的针恢复,因为这样的针再插上去比取下来的针插上去容易些;另外刚换的新针,一般比旧针长。如直接装上新的打印针进行打印,那么凸出的打印针就会刮烂色带,很容易造成打印针再次折断。如长得比较多,如用油石或细砂打磨,则易引起折断。对此,宜在穿针前用手术剪轻轻剪掉一点,余下的部分待穿好后再用油石或细砂打磨校正。同时在用油石或细砂打磨打印头时,开头用力要很轻。宜将细砂布或油石平放于玻璃板上,打印头一定要和细砂布或油石面垂直,切勿歪斜。用力要逐渐加大,不能用力过猛,当全部打印针与定位片在同一平面上时,这个打印头就算装好了。

电脑是人类社会智能化产品的杰出代表作,已成为现代人工作和生活中不可缺少的工具。

然而,电脑的发展远未达全臻的境地,面对来自电网的各种干扰,每个电脑操作员都不无后顾之忧,突然的掉电、电压的波动,突发性的脉冲乃至不小心碰了碰电源线都有可能将正在进行的操作全部毁掉。一切都必须重新开始,尤其是遇到大工作量、长时间的操作和运行、操作员大都提心吊胆,生怕出现稍稍的不慎。有些时候,工作不可以中止而人却要休息,就只好打疲劳战了。

怎样使电脑具有自我保护能力,为计算机创造一个无忧环境是一个富有挑战性的课题。无忧电脑是电脑必然的发展趋势,所谓无忧电脑是一种集断电保护、数据加密、防病毒于一体的全新计算机。北京超世纪电子新技术研究所研制开发了新一代的电脑数据保护神——无忧卡,从而揭开了无忧电脑跨入电脑世界的序幕,将开辟电脑发展的新纪元。

无忧卡由微机插卡、充电电池和驱动软件组成,全部置于微机内,正常状态时自动监测电源,对电网波动起稳压作用;遇到断电,自动储存当前运行环境,等供电恢复,就自动恢复到原来状态。这就使计算机无人值守和分时进行大程序运行操作成为可能,另外,无忧卡还具有热键储存、恢复功能,可以储存一些必要的环境,在需要时调到当前状态。

无忧卡集稳压和数据保护于一身,将以其高智能性成为电脑家族不可缺少的一员干将,而无忧电脑也会在不久的将来成为电脑世界的主宰。

一、设计构想:

无忧卡是针对微型机系统设计的,微机运行应用软件时,从硬盘或软盘将程序调入内存中运行。所以,只要把内存中的数据及 CPU 数据做一备份,就可以把系统以及执行状态完整地保存起来。

有些硬件芯片初值设定及输出到 I/O 接口的数值无法由软件读出,无忧卡设计了逻辑线路,可以记录送到 I/O 接口的信号,系统恢复过程中,就可将原来状态完全恢复。

无忧卡的控制软件开机时常驻内存(约占 20K),随时检测有无断电,若有断电信号,立即中止应用软件的运行,存储当前状态,然后关闭电源(无忧卡提供直流电源),一旦恢复供电,自动恢复成断电前状态,继续执行未完成之操作,使微机操作具有不受断电影响的功能。

二、软件设计

无忧卡软件分非断电存储,断电存储,复电恢复及系统安装四大部分。

为了避免断电存储时,因硬盘空间不足而造成存储失败,安装时自动在硬盘划出一个空间,以备断电存储,其容量为:

基本内存+扩展内存+扩充内存(EMS)+显示卡内存+

64K

对于一台 4M 内存的 386,无忧卡安装后约占硬盘 5M 的容量。

从方便用户的角度考虑,无忧卡提供非断电存储功能,当用户需在同一环境下反复运行多次时,用户可以把此环境先存储起来,然后在需要时恢复成此环境。这些操作均可通过菜单或热键执行。例如,用户调试一程序时,在某处进行存储,然后可以多次从此开始运行,提高工作效率。

另外,若用户不需断电存储,可用热键或菜单将无忧卡关闭。

三、对电源不稳定的处理

电网波动或电力干扰造成的脉冲,大都不超过一秒,所以无忧卡设计思想是断电时立即检测断电能否在一秒内复电,如果断电不超过一秒,系统不存储任何数据,由无忧卡电源供电继续操作,达到稳压功能,若断电超过一秒,则先储存数据,然后再检测是否复电,若供电恢复,则将存储在硬盘上的数据恢复;若没有复电,则关闭无忧卡,停止供电。等复电时,系统自动将硬盘数据恢复,继续应用软件的运行。若在存储数据期间复电,系统要继续完成储存,随即恢复所存数据,继续执行断电前的文件运行,就象未曾断电一样。

四、中文系统

现在中文系统规格各异,无一统一标准,大致可分为使用软字库和使用硬字库(汉卡)的两类中文系统。

使用软字库的中文系统与显示卡密切相关,无忧卡支持 Hercules, GGA, EGA, MC-GA, VGA 等标准显示卡,还可以支持 TVGA, VGA PLUS, SUPER EGA, SUPER VGA, CVGA 等非标显示卡。

对于非标显示卡有各自不同的驱动方式,无忧卡将不同的驱动程序归类,提供驱动程序。

对于使用硬字库的中文系统,无忧卡亦有丰富的驱动程序支持各种硬字库(汉卡)。

五、鼠标种类

目前的鼠标可分成两类:一种是与 Microsoft 兼容的 mouse;一种是智能性 mouse。其处理方式同汉卡一样,前者利用无忧卡上提供的逻辑线路即可存储或恢复,后者必须先判定 mouse 种类并提供驱动软件,然后才可处理存储和恢复。

六、其他

断电前若正在进行软盘的读写操作,则在复电后系统会自动辨认软驱中的软盘是否断电前的,若不是,则系统会提示用户。待确认后方能继续断电前的操作,并可将软盘读写头定位至原先位置,以防写错盘片而损伤宝贵资料。

无忧卡适用于配置了硬盘并且有一个空插槽的 XT、AT、286、386、486 等各种 PC 机、也可用于 Novell 等三种网络环境。



合理搭配 CCBIOS 2.13H 和 SPDOS 5.10

的批处理程序

青岛市市南区莱芜二路5号 杭箴良

CCBIOS 2.13H 和 SPDOS 5.10 是目前我们常用的软汉字系统。2.13 系列推出早, 功能较多, 尤其其它的打印功能出众, 因此, 有不少中文应用软件以它为运行环境。可惜, 2.13H 的屏幕显示有“黑线”和色彩不正的缺陷; SPDOS 5.10 的屏幕显示出色, 汉字输入法种类多, 使用方便灵活, 与西文软件的兼容较好, 其 WPS 文字处理系统的使用者也为数甚多。从 SPDOS 5.10 用户手册的介绍来看, 它的打印功能也很出色, 但一般因没有其打印软字库, 如字型选择等部分功能在 SPDOS 5.10 中无法使用; 而且, 即便有了 SPDOS 5.10 的打印软字库, 原来在 CCBIOS 2.13H 下运行的中文应用软件改在 SPDOS 5.10 下运行时, 也存在修改打印模块的问题。

我们知道, 打印输出是应用软件, 尤其 MIS 不可缺少的组成部分。在编写应用软件的过程中, 本人使用如下批处理程序来综合 CCBIOS 2.13H 和 SPDOS 5.10 两种汉字系统的优势, 使“鱼与熊掌, 二者可以兼得”:

```
@ECHO OFF
C:
REM 进入 CCBIOS 2.13H 的子目录
REM 调用 CCBIOS 2.13H 中的打印部分
CD \213
REM 调入显示字库
IF NOT EXIST f:HZK16 COPY HZK16 f:
REM 调入读虚拟盘字库模块
FILE3 f2
REM 若虚拟盘的盘符不是 F 则改以上二行中的 f 为实际虚拟盘的盘符
REM 调入 2.13H 的打印机驱动程序
PRTA
REM 调入读显示字库模块
FILE16B
REM 调入读 24 点阵字库模块, 常用字库区可存字 50 个
FILE24A ISFHK
REM 调入读 40 点阵字库模块, 常用字库区可存字 50 个
FILE40A ISFHK
REM 调入 24 点阵 3# 字符库
ZF24 3

REM 进入 SPDOS 5.10 的子目录
REM 调用 SPDOS 5.10 的显示部分
CD \WPS
REM 装载 SPDOS 5.10 字库
SPLIB
REM 装载 SPDOS 5.10 显示模块
```

```
SPDOS CIZU
CD \
@ECHO ON
```

本批处理程序搭配使用 CCBIOS 2.13H 和 SPDOS 5.10 启动批处理程序中的命令, 先安装 CCBIOS 2.13H 的打印部分, 然后安装 SPDOS 5.10, 但是不装 SPDOS 5.10 的打印驱动程序。这时, 中文应用软件便可享受 SPDOS 5.10 的显示和汉字输入法以及 CCBIOS 2.13H 的打印功能了。如使用 WPS, 只要在 WPS 中打印时正确设置打印参数, 本批处理程序是不会影响 WPS 的打印功能的。

CCBIOS 2.13H 的打印部分和 SPDOS 5.10 的显示部分二者装入顺序可不分先后, 但后装 SPDOS 5.10, 可利用其“辅助功能”菜单中的“移去 CCDOS”功能退出中文 DOS。当然, 退回西文后, 未卸下的 2.13H 打印模块和驻留内存的 SPLIB.EXE 仍占用约 100k 内存。

本批处理程序对 MS-DOS 3.3 和 5.0 版本都适用。为给中文应用软件多留些内存空间, 当然应选用 MS-DOS 5.0 版本。顺便提醒注意, 由于 MS-DOS 5.0 中做虚拟盘的设备驱动程序 RAMDRIVE.SYS 与 MS-DOS 3.3 中 VDISK.SYS 的差异, CCBIOS 2.13H 的 FILE3.COM 程序中偏移 027AH 处的内容在 MS-DOS 3.3 中运行时为 10H, 而在 MS-DOS 5.0 中运行时应为 11H; 由于和 MS-DOS 3.3 相比, MS-DOS 5.0 对 DOS 驱动器参数块 (DDPB) 的格式作了改动 (偏移 0FH 处的每个 FAT 表占用扇区数由 3.3 版的一个字节改成 5.0 版的一个字), 故针对 DOS 的两种版本, SPLIB.EXE 也须变动两个字节 (用 PCTOOLS 等工具软件分别查找并修改 807F16F8H 和 8B4710H 为 807F17F8H 和 8B4711H)。此外, 为加快 SPDOS 读取字库的速度, 建议使用磁盘高速缓存 [尽管 MS-DOS 5.0 及 MS-WINDOWS 3.1 中都附带有 SMARTDRV 来做磁盘高速缓存, 但因为 Central Point Software 公司所出 PCTOOLS 软件包中的 PC-CACHE.COM 程序可根据实际需要, 在 DOS 命令行上装载和卸下磁盘高速缓存, 所以还是用 PC-CACHE 方便些。

多操作系统共享硬盘技巧

王兴昌 赵家亮

我单位使用的高档微机有浪潮 386、华南 386、新世纪 386 等,这些微机地硬盘容量较大,均在 80 兆以上,如果只安装一套操作系统,那资源浪费太大,另一方面我行使用的各种应用软件又是在不同的操作系统下开发的,于是提出了多操作系统共享硬盘的问题,我们在实际工作中进行了摸索和尝试,积累了一些经验和技巧,现介绍如下,供同行参考:

一、Xenix 和 DOS 的安装

1、在硬盘上安装 DOS 系统。先用随机盘中的诊断程序将硬盘初始化,这样硬盘就成空的了,其次用 FDISK 给 DOS 分配适当的柱面数,建立 DOS 分区,再用 FORMAT C: /S 格式化硬盘,最后将西文 DOS 拷入硬盘,就完成了 DOS 系统的安装。

2、在硬盘上建立 Xenix 分区。安装完 DOS 系统后,就可以安装 Xenix 系统了(安装方法请参考 Xenix 系统安装手册)当安装进行到调用 FDISK 菜单时,选择建立 Xenix 分区,系统将请你输入 Xenix 分区的磁道数,你可以用硬盘总磁道数减去 DOS 分区所占磁道数来确定 Xenix 分区应占磁道数。Xenix 不一定要占第一分区,但分区不允许重叠。

3、Xenix 与 DOS 之间的切换。实现 Xenix 与 DOS 操作系统之间的切换方法很多,但最好的方法是将 Xenix 分区设置为活动分区,当系统启动到 BOOT 时直接回车进入 Xenix 系统,键入 DOS 后回车则进入 DOS 操作系统。

二、DOS 分区中多种中文操作系统的安装

1、将要用到的中文操作系统拷入硬盘,并把拷入的操作系统的自动批处理命令更名,待用户启动机器后再根据使用要求选择进入需要的系统,更名时应尽量使用能代表系统特征的名称,如 2.13F 系统可更为 213F.BAT。

2、安装系统应注意的问题。大多数中文操作系统要求在根目录下使用(一些特殊系统也能在子目录下使用,如西山 DOS5.1)。其主文件存在根目录下,对于多套系统中一些同名的文件(如字库),需将这些文件更名使用,这些文件更名后,相应的调用它们的地方要修改,对于用编译程序调用的可用 Pctool 等工具软件修改调用文件的地方。

3、因 DOS 在同一级目录下能管理的最大文件数是

512(针对硬盘),装多套系统后,同一级目录下的文件可能大于 512,这时用户的各种应用程序最好安装在用户自建的子目录下,以确保应用程序的安全,装好后用户可根据情况建立一些调用应用程序的批处理命令。

409

一篇文章的疏忽

黄泳进

下面是黄泳进读者来信的片段:

前不久我阅读了贵刊 1993 年第 4 期第 61 页的一篇文章,是安徽财贸学院计算中心的鄂大伟同志所著的《动画图形设计中的位图像操作与掩模技术》,文章末附了一个用 Turbo C 实现的 mask 动画演示程序,我用 286 机在 Turbo C 2.0 版上调试后,发现该程序有几点疏忽:

①变量 CX 未予赋值,这样程序在编译时会给警告(warning)。同时也就没给出卫星运动的初始位置(即 cx 的值)。改进方法是:在 main() 这函数的首句定义 cx 的值,即首句改为: int gmode=VGAHI. gdriver=VGA. cx=0. midx. i. cy;

这样卫星就可从屏幕的最左边开始往右运动了。

②卫星移动后,在卫星的初始位置会留下一个空白的矩形区域,其原因是在 for 循环语句的第一次循环时所画的背景是一个空背景。

改进方法:在 main() 主函数中的 Pustar(): 语句和 for(i=0;i<600;i+=STEP) 语句中插入一句: getimage(cx, cy, cx+SIZE, cy+SIZE, covered): 即可。

以上两点是我在调试此程序中发现的一些问题,经这样改进后,才可显示出文章所说的优美的动画画面。

.....

非常感谢黄泳进读者的指正,我们已把来信的复印件寄给鄂大伟作者,我们也正十分欢迎,并热情地希望广大读者多来信交流、切磋,以后有这类来信我们将第一时间刊登。

410

天堑变通途

苏睿

我是一名电脑培训班的教师,汉字输入是电脑教学中一个很重要的部分,许多学员对此有浓厚的兴趣,但由于课时辅导教师的数量有限,许多学员在学习过程中的疑问得不到及时解答,影响了教学质量。特别是五笔字型汉字输入法,由于规则太多、要记忆的字根繁杂,大多数学员感到学习吃力,视此为畏途。作为教师,除了选择的好的教材,不断总结教学经验外,最希望的是能找到一套好的辅助教学软件,藉此减轻教学强度和提高教学质量。我本人搜集了不少的这一类软件,但关于汉字输入方面的却比较少,有几套这样的软件(如星汉、王码等),虽然勉强可用,但其质量和效果却远远无法让人满意,有些在数据产生上相当死板,有些在适应机型和显示器上有严格的要求,联机求助信息也很单一,不适于教学使用。

前不久,在同行手中得一套《CT教学软件》,经使用顿觉耳目一新,有“踏破铁鞋无处寻,得来全不费功夫”之感,我安排了部分学员作对比试验,经过一周时间的使用,教学效果之好,超出了我的意料,使用CT软件的学员,一周之后,已能熟练使用五笔方法输入大篇文章,输入速度远远快于没有使用CT软件的学员。平时学员均感入门关难过,输入速度难以提高,CT软件轻而易举地解决了这些问题,各种练习模块设置合理,尤其是丰富的屏幕指导信息,学员们颇为欣赏,他们说:“有了这套软件,再也不用翻书了。”有的甚至开玩笑说,有了这套软件,老师就有失业的危险了。另外,我也找了一些以前毕业的学员来进行试验,他们的回答是:“如果当初就使用这套软件,学习就不会那么吃力了。”

我一直想同作者取得联系,后来得知《电脑》杂志社代理这一软件,并看到了刊登在93.1、93.5和93.6上的文章,贵刊为电脑打字的普及工作做了一件好事,在此我代表广大的电脑教员和学员对贵刊和作者赖国全同志表示衷心的感谢,同时寄予如下厚望:

一、对推广使用CT软件的电脑培训班,是否可以出售不加密的CT软件。

二、价格对不同的购买对象是否可以优惠一些。

三、广泛搜集教师和学员的意见,把软件界面做得更漂亮一些。

四、建议作者和贵刊能与广大教师充分合作,大力推广和宣传,让更多的教师和学员有机会使用CT软件。

五、编写更多更好的电脑教学软件(如编辑、排版方面),推进电脑教学自动化进程。

另,建议《CT教学软件》更名为“五笔通”汉字输入教学软件,使人更能理解这套教学软件的优越性。

一种快速删除无用文件的方法

张少丹

在计算机教学实验中,常常需要在第一批学生上机操作结束后,马上删除某些文件,再让第二批学生上机操作,特别是在上机考试时更是这样,以防作弊。如果用DEL命令个文件删除,当然可以满足我们的需要。但这样速度太慢,尤其是在分组考试期间,要在短短的几分钟内,分别在几十台微机上做这个工作,更显得困难。下面以在AutoCAD软件教学实验为例,说明如何快速删除无用的(或不应该存在的)文件。

我们首先用文本编辑程序建立一个批处理文件DCAD.BAT,内容如下:

```
C>TYPE DCAD.BAT
DEL *.DWG
DEL *.SCR
DEL *.SHX
DEL *.SHX
DEL *.PAT
DEL *.LIN
DEL *.SLD
DEL *.DXF
DEL *.MNU
DEL *.MNX
DEL *.OLD
DEL *.LSP
DEL *.BAK
DEL *.? -?
DEL *.
COPY \CCAD\*.*
```

DCAD.BAT文件中的前几条命令用于删除当前目录中无用的文件,但这样做可能把AutoCAD软件中的某些必要文件删除掉。因此,在DCAD.BAT文件中最后一条命令的作用是;恢复刚才被删除掉的AutoCAD软件中必要的文件。故此,我们必须建立一个子目录(假设子目录名为CCAD),用COPY命令拷贝AutoCAD软件中的必要文件。即在CCAD目录下有下列文件:

```
ACAD.PAT
ACAD.NMX
ACAD.DWG
ACAD.LIN
TXT.SHX
MONOTXT.SHX
COMPLEX.SHX
SIMPLEX.SHX
ITALIC.SHX
VERTICAL.SHX
```

这样,当第一批学生操作结束后,只要敲入DCAD即可快速删除无用的文件,而又不影响第二批学生的操作。

这种方法的特点是:简单、方便、实用、快速。

巧用 FOXBASE 中的宏命令

宋捷

我们在编程时通常把重点放在“有良好的人机对话界面”上，而忽略了实现机与机对话的程序设计技巧，实际上，实现机与机对话的程序设计才是编程成功与否的关键，现将 FOXBASE 中实现机与机对话的程序设计技巧介绍如下：

大家都知道，LOTUS-123 的最大优点之一是具有“宏命令”，FOXBASE 也具有“宏命令”功能，而实现机与机对话的关键就是使用宏命令，所谓“宏命令”就是能够替代手工操作键盘自动执行运行过程的命令语句，FOXBASE “宏命令”的主要优点是：

1. 可以节省时间

在菜单中执行各项操作时，不需要操作者干预，如打印报表时，操作者不必在打印机旁等待上一张表打印完后再键入数据以选择打印下一张表，“宏命令”可使打印所有报表的操作一次完成；

2. 可以节省精力

通过把一系列的操作步骤编制在一个“宏命令”里，使操作者可以不必记住各操作步骤的顺序便可进行操作；

3. 确保输入正确

对于初学者，可以通过设计“宏命令”把数据开始输入处定在 BROWSE、APPEND、EDIT、MODI COMM 等全屏幕编辑命令的一个确定的行和列处；

实现机内对话的“宏命令”主要有两个：

1. 命令格式：KEYBOARD “填入缓冲区的数据”，填键盘缓冲区命令 KEYBOARD 允许键盘缓冲区被任意的字符装满，然后数据保留在缓冲区，直到 FOXBASE 寻求输入为止，那时仿佛是从键盘缓冲区进入的数据一样；

2. 命令格式：SET FUNC n TO “填入缓冲区的数据”，其中 n=1-10，分别表示设置十个功能键的命令，设置功能键可实现 FOXBASE 的这种“批处理操作”功能，即在操作时仅按一键相当于按了多次数字键、字母键、Ctrl 键及回车键；

以上两个命令中的“填入缓冲区的数据”可以是需操作者键入的任何 ASCII 字符，但应注意第 1 种命令方式中回车用 CHR (13)，而第

2 种命令方式中回车用“；”代替，现举二个实例来对“宏命令”加以说明：

例一：只要在所编应用程序的主菜单中加一句 SET FUNC 8 TO “^F^X^A^E”，则在以后运行到由 BROWSE 命令控制的语句时，按一个 F8 键即可完成 <

CTRL>+F、<CTRL>+X、<CTRL>+A、<CTRL>+E 四个键的功能，从而执行“→”、“↓”、“←”、“↑”四个操作，光标自动循环一圈后回到原位，应用这种方法可控制光标到 BROWSE 屏幕的任一指定位置，实现了由机内程序自动控制机内操作的机与机对话，注意在定义功能键 F8 的语句中有控制字符，因此这行语句必须在 EDLIN 中用 1 命令来添加，在 WS 等字处理软件或 FOXBASE 的编辑器中无法添加此行宏命令。

例二：示例见程序 HML. PRG，程序 HML. PRG 运行后，在主菜单中按数字键 4 即开始执行“4. 报表批处理”，报表批处理运行 KEYB 命令后的所有键值，其过程如下：首先将回车符 CHR (13) 置于变量 C 中，然后执行“宏命令”，即向缓冲区填数，第一个变量 C 前的“231”分别表示程序运行时先自动执行主菜单中的“2. 计算报表数据”，以便运行 JS. PRG (此程序省略) 计算报表，待计算完后，执行“3. 打印报表数据”，进入打印报表子菜单，在打印报表子菜单中，选择打印子菜单中的“1——资金平衡表”，填入第一个变量 C (回车) 确认选择后，执行 DY1. PRG 来打印表一 (打印表一至表十的程序 DY1. PRG 至 DY10. PRG 在此处略去)，以此类推，“32”+C 至“39”+C 分别表示自动选择打印表 2 至表 9，最后“3110”表示打印完了表 10 后自动执行主菜单中的“0. 返回”而直接退到 DOS。总之，原来每二、三分钟操作者就要干预一次的操作，变成了操作者仅按一个键即完成了二十三分钟工作量的全部操作，实现了这二十三分钟内的“机内”对话，另外，程序中用“*”开始的两行表示“宏命令”的第二种命令方式在程序中的使用，其运行效果与上一行的 KEYB 语句相同，读者不妨一试。

以上程序在 SUN486、HP386、GW386 主机 AR3240 打印机上实现，操作系统为 DOS3.3，汉字系统为 CCDOS2.13H。

C>TYPE HML. PRG

```
set talk off
clear
do while .t.
@2,25 say "***** 财务报表 *****"
@4,25 prompt "1. 输入报表数据" mess "共输入十张财务报表"
@5,25 prompt "2. 计算报表数据" mess "计算三分钟"
@6,25 prompt "3. 打印报表数据" mess "每张表打印两分钟"
@7,25 prompt "4. 报表批处理" mess "共需二十三分钟"
@8,25 prompt "0. 返回" mess "退到操作系统"
set mess to 24
menu to xz
do case
case xz=1
do sr
case xz=2
do js
case xz=3
clear
```

```

clear
xz1=' '$8.——do while .t.
@1.5 say '*** 打印报表 ***'
@2.5 say '1——资金平衡表'
@2.40 say '2——应上交及弥补款项情况表'
@3.5 say '3——利润表'
@3.40 say '4——销售利润明细表'
@4.5 say '5——成本表'
@4.40 say '6——电力成本构成明细表'
@5.5 say '7——生产费用表'
@5.40 say '8——技术经济指标表'
@6.5 say '9——专项工程支出明细表'
@6.40 say '10——企业管理费用明细表'
@8.20 say '请选择(0退出)' get xz1
read
if xz1='0'
exit
endi

```

```

if val(xz1) >= 0. and. val(xz1) <= 10
do dy8. xz1
exit
endi
endd
case xz=4
c=CHR(13)
note 1. KEYBOARD 命令实现“宏命令”功能
keyb "231"+c+"32"+c+"33"+c+"34"+c+"35"+c+"36"+
c+"37"+c+"38"+c+"39"+c+"3100"
note 2. SET FUNC N TO 命令实现“宏命令”功能
*SET FUNC 2 TO ":231:32:33:34:35:36:37:38:39:3100"
*ACCEPT "请按功能键 F2" to F2
case xz=5
quit
endcase
enddo

```

413

DRIVE.SYS 设备命令的几种用法

曹松林

在 DOS3.2 以上版本中,增加了一条 DRIVESYS 设备命令。允许利用引用一个逻辑驱动器字母使你访问和使用磁盘设备。该命令放在 CONFIG.SYS 文件中,在系统启动时对 DOS 进行配置,使我们能对正在使用的磁盘机加上一个或多个磁盘机的逻辑名称。其句法格式如下:

DEVICE=DRIVER.SYS /D:ddd

/D:ddd 指定物理驱动器号。物理驱动器号具有 0—255 之间的值。0 值指定第一个物理软盘驱动器并且在 DOS 命令中以驱动器 A 被引用。值 1 指出第二个物理软盘驱动器。值 2 指出第三个物理软盘驱动器(它必须是外部的)。为了指定第一个物理硬盘驱动器,应使用值 128。值 129 指定第二个物理硬盘驱动器。

/T:ttt 指定每一面的磁道数目(1~999)。约定值是每一面 80 磁道。

/S:ss 指定每一磁道的扇区数目(1~99)。其约定值是每一磁道 9 扇区。

/H:hh 是最大的磁头数目(1~99)。其约定数目是 2 个磁头。

/C:指定所要求的变更线(changeline)支持软件。这只有对支持软盘变更线的机器才有意义,例如 IBM-PC/AT 机。

/N 用于指定的物理设备是不可拿走的块设备。硬磁盘是不可拿走的块设备的例子。

/F 用于指定设备类型(形式因子)。160/180/320/360KB 的软盘机的值是 0。1.2MB 软盘机的值是 1。720KB 软盘机的值是 2。

本人将使用该命令的经验进行了总结,归纳为以下几种用法:

一、将一个软盘驱动器建立为另一外部软盘驱动器的逻辑驱

动器。例如,为了在 IBM-PC/XT (一个内部软盘驱动器和硬盘)机上建立 720KB 外部软盘驱动器的逻辑驱动器(D),使用下述命令:

DEVICE=DRIVER.SYS /D:2

二、从 720KB 外部软盘驱动器拷贝到同一个驱动器上。为了达到这一目的,则在 CONFIG.SYS 文件中把同一条命令放置两次,这就是(对 IBM-PC/XT)给驱动器指定逻辑驱动器字母 D 和 E。

DEVICE=DRIVER.SYS /D:2

DEVICE=DRIVER.SYS /D:2

三、在同一台磁盘机实现文件拷贝。这是最常见的用法,例如:我们使用两个分别为 A 和 B 的软磁盘机,没有硬盘机,就可**通过 DRIVER.SYS 为 A 磁盘机或 B 磁盘机多一个逻辑名称 C。这样一来同一个磁盘机就有两个逻辑名称 A 和 C 或 B 和 C。**而在命令中,是把 A 和 C 或 B 和 C 当成是两个磁盘机,轮流使用同一个实际的磁盘机,而这样以来我们可以用同一个磁盘机把文件从一个拷贝至另一个磁盘。否则,用户若要在同一个磁盘机上拷贝文件,系统就会提示“file cannot be copied onto itself”信息。

四、可增加软磁盘的容量。在 DRIVER.SYS 命令的句中,我们已经知道参数 /T:ttt 是指定一面磁盘所含的磁道数。若是 360KB 的磁盘只有 40 条磁道。现在我们把参数填为 43 道。系统重新启动后,这样该逻辑磁盘机就被定义为 43 道的磁盘机。然后用户可把将要格式化的软磁盘放入该逻辑磁盘机中格式化就可获得 43 道的磁盘,即把原来的盘片容量增加了 $3 \times 2 \times 512 \times 9$ 个字节(27648 个字节)。

五、可对磁盘或磁盘中的文件加密。我们只要在第四种用途中加一引伸,就可以对磁盘或磁盘中的文件加密。一般情况下,在具有多于 40 道的逻辑磁盘机中格式化的磁盘是不能用 DISKCOPY 命令在 40 道磁盘机中复制。但可以用 COPY 命令复制。因此用户只要把要加密的密码存入 40 道以后的那些磁道中的任意扇区中,并把那些磁道中的扇区置为费区,当用户启动系统或使用某个文件时,让系统或文件到你存入密码的那些扇区去取密码。这样即使非法用户在 360KB 软盘中用 COPY 命令拷贝了你的系统和全部文件,由于你的密码没有被复制,所以非法用户还是不能使用你的磁盘或文件,而对于用户磁盘没有因为存入密码而损失磁盘空间(磁盘仍旧具有 360KB 空间)。

414

K



和
电
脑

“24点”

程宇清



“24点”是一种很好的益智扑克游戏，大家也许都玩过，游戏的规则很简单，任意抽取四张牌，运用加、减、乘、除各种运算，看谁先算出24来。

运行以下程序，选择“Work VS the computer”，共有难、中、易三项选择，电脑随机抽取四张牌，你如果算出来就按一个键，电脑就会说“You are very clever!”如果你在一定时间里回答不出，电脑就会给出结果。当然，电脑只不过是让你的，如果你想知道它算得到有多快，就选择“Test the computer”，输入四张牌，不出三秒钟，电脑就会给出计算方法，如果这四张牌无法得到24，它就会说“Sorry, these four cards can not be 24”

本程序在 Turbo Pascal 4.0 以上均可运行。

```

program Points0124;
uses crt;
var l:array[1..4] of integer;
    p:char;
    i:integer;
procedure FirstScreen:
var a:char;
begin
    clrscr;
    gotoxy(24,11);
    writeln('The twenty-four points poker game');
    gotoxy(30,16);
    writeln('Written by Cheng Yuqing');
    gotoxy(20,20);
    writeln('LH International Software Co.Ltd--1993.6');
    repeat until keypressed;
    clrscr;
end;
procedure Secondscreen:
begin
    clrscr;
    gotoxy(30,8);writeln('Work VS the computer');
    gotoxy(30,12);writeln('Test the computer');
    gotoxy(30,16);write('Please Select W or T');

```

```

P:=readkey;
if not (p in ['W','w','T','t']) then secondscreen:
end;
procedure Wait(t:integer):
var i,j:integer;
begin
    for i:=1 to t do
    for j:=1 to t do
    if deypress then begin writeln('You're clever!');halt:end;
end;
procedure Thirdscreen:begin
randomize;
clrscr;
gotoxy(30,6);writeln('Easy');
gotoxy(30,8);writeln('Middle');
gotoxy(30,10);writeln('Difficult');
gotoxy(30,16);write('Select E M or D');
p:=readkey;
clrscr;
for i:=1 to 4 do l[i]:=random(12)+1;
writeln('The four cards is',l[1]:3,l[2]:3,l[3]:3,l[4]:3);
if p in ['e','E'] then wait(1200);
if p in ['m','M'] then wait(600);
if p in ['d','D'] then wait(300);
end;
procedure input:
begin
    writeln;
    write('Enter four cards (A--1, J--11, Q--12, K--13):');
    readln(l[1],l[2],l[3],l[4]);
    if not (l[1],l[2],l[3],l[4])<=[1..13]) then input:
end;
procedure Smallwork(a,b:integer;var c:real):
begin
    if ((c=0) and (a=5)) then c:=-9999
    else
    case a of
        1:c:=c+b;
        2:c:=c-b;
        3:c:=c*b;
        4:c:=c/b;
        5:c:=b/c;
    end;
end;
procedure Print(a,b,c,d,e,f,g:integer):
var ch:array[1..5] of char;
begin
    ch[1]:='+';ch[2]:='-';ch[3]:='*';ch[4]:='/'
    ch[5]:='%';
    writeln(l[a]:3,ch[b]:3,l[c]:3,ch[d]:3,l[e]:3,ch[f]:3,l[g]:3);
    halt;
end;
procedure Thirdscreen:begin
randomize;
clrscr;
gotoxy(30,6);writeln('Easy');
gotoxy(30,8);writeln('Middle');
gotoxy(30,10);writeln('Difficult');
gotoxy(30,16);write('Select E M or D');
p:=readkey;
clrscr;
for i:=1 to 4 do l[i]:=random(12)+1;
writeln('The four cards is',l[1]:3,l[2]:3,l[3]:3,l[4]:3);
if p in ['e','E'] then wait(1200);

```

```

if p in ['m', 'M'] then wait (600);
if p in ['D', 'd'] then wait (300);
end;
procedure input;
begin
  writeln;
  write (' Enter four cards (A—1, J—11, Q—12, K—13): ');
  readln (l[1], l[2], l[3], l[4]);
  if not (l[1], l[2], l[3], l[4]) <= [1..13]) then input;
end;
procedure Smallwork (a, b: integer; var c: real);
begin
  if ((c=0) and (a=5)) then c:=-9999;
  else
    case a of
      1: c:=c+b;
      2: c:=c-b;
      3: c:=c*b;
      4: c:=c/b;
      5: c:=b/c;
    end;
  end;
  procedure Print (a, b, c, d, e, f, g: integer);
  var ch: array[1..5] of char;
  begin
    ch[1]:='+'; ch[2]:='-'; ch[3]:='*'; ch[4]:='/'; ch[5]:='%';
    writeln (l[a]:3, ch[b]:3, l[c]:3, ch[d]:3, l[e]:3, ch[f]:3, l[g]:3);
    halt;
  end;
end;

```

```

procedure Work;
var a, b, c, d, e, f, g: integer;
t: real;
begin
  for a:=1 to 4 do
    for b:=1 to 5 do
      for c:=1 to 4 do
        for d:=1 to 5 do
          for e:=1 to 4 do
            for f:=1 to 5 do
              for g:=1 to 4 do
                if (a+c+e+g=10) and (a*c*e*g=24) then begin
                  t:=1/a;
                  Smallwork (b, l[c], t);
                  Smallwork (d, l[e], t);
                  smallwork (f, l[g], t);
                  if t=24 then Print (a, b, c, d, e, f, g);
                end;
              end;
            end;
          end;
        end;
      end;
    end;
    writeln (' Sorry. The four cards can't be 24! ');
  end;
begin
  Firstscreen;
  Secondscreen;
  if p in ['T', 't'] then
    begin
      input;
      work;
      writeln;
    end
  else
    begin
      thirdscreen;

```

115

让“半仙”变金仙

甘肃省庆阳一中 任绥海

《电脑》今年第五期“游戏乐园”上发表的赵礼海先生的文章，将《决战中国海》改成了“不死金刚的半仙版”。之所以叫“半仙”，是因为虽说飞机“死”不了，但顾虑仍然存在，就是文中所说的，“如果关尾决战时，子弹用完了，那可是活受罪了。”作者也因此提醒我们：“注意节约子弹”。

确实，这架飞机的装载能力有限，最多时也只有 99 枚导弹，99 发子弹，而最后关头上的那个庞然大物，它一个就得 60 多枚导弹才能打发。

可不可以提高它的装载量呢？当然可以。

原程序中所赋初值是 50，十六进制表示为 32H。赋值指令及十六进制代码如下（共有三处）：XXXX—C7847C133200—MOV—WORD—PTR，3200—我们把 32H 改成一个大一些的数值如 7530H，就可以把初值提高到三万！本人玩的时候，自始至终不

开一枪，全放导弹，也没有再有意去补充给养，到结束时导弹数量还在两万以上！这个数量已经大得有点过分了，你可千万别再贪心把它改得更大，以防物极必反。因为当寄存器高位为 1 时，低 15 位（或 7 位）所表示的将是一个负数！例如 8000H=-32767！这时候，你不但发不出导弹，还倒欠了计算机的账，而且怎么还都还不清！

本来，这样一改应该是高枕无忧了，但我们还不能闭着眼睛去冲锋陷阵。

游戏中常能看到一只彩蝶，打它一下，又会变成一朵五出梅花，吃掉梅花后，你的三项能源数据都将变成 99。如果是原先，这当然是求之不得的好事，但我们前面已经将初值改成了 30000，让它再改成 99，那怎么行！看来这里也得修改。

程序中相应指令及代码如下：

XXXX— C7847C136300— MOV— WORD— PTR[S1+137C], 0063

现在我们将 MOV 指令改成 ADD（追加），即

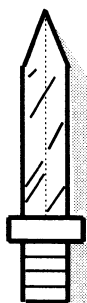
ADD WORD PTR[S1+137C], 63，其代码为

83 84 7C 13 63 90

该指令共出现五次，要全部修改。

这样一来，我敢打赌，你即使闭着眼睛随便打，也能很快过关的。

116



侠

影

记

的修改

侠

影记是一个相当好玩的游戏，但对于初接触这个游戏的人来说有一定的难度。主人公傅剑寒初出江湖，武功低微，可使用之物只有银两 1000，对付魔教一个无名小辈也要费尽心思。就算玩家高手有时也会因找不到某件物品而玩不下去，为此，笔者花了两天时间对侠影记进行分析，终于发现侠影记的初始化数据及存局数据均存放在文件 DATA2.GRP 中。对 DATA2.GRP 进行修改即获得所需的物品和武功。

具体方法如下：

使用 PCTOOLS 查找 FF 81 0B 80 四个字节，查找过程中共发现六处地方。其中第一处地方不用管它，第二处地方即为初始化数据，第三处地方为档案 1 数据，其余分别为档案 2,3,4 数据。每处数据结构相同，如下所示：

偏移：

```
0000 FF 81 0B 80 B3 C5 BC 43 B4 48 00 00 0D 80 00 00
0010 C8 00 C8 00 01 00 20 00 01 00 01 00 01 00 05 00
0020 05 00 05 00 01 00 00 00 00 00 00 00 00 00 00
0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0040 00 00 00 00 00 00 00 00 00 00 00 00 00 00 60 00
0050 E8 03 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0060 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0070 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0080 00 00 00 00 00 00 00 00 00 00 00 00 00 00 1E 00
```

方框内（从偏移 004E 起）为物品记录数据，每一件物品由四个字节组成。以第一项 60 00 E8 03 为例，60 是银两的代号，00 起分隔作用，E8 03 即十六进制 03E8H。也就是十进制 1000，代表银两 1000 两。以下是各物品代号：

60 银	两	6F 紫	金	鼎
61 树	枝	70 辟	火	丹
62 卷	线	71 红	尘	簪
63 竹	子	72 多	情	谱
64 弓	箭	73 铁		箱
65 钝	斧	74 玉		佩

66 灰	锁	钥	75 截脉散	解药
67 红	锁	钥	76 石	块
68 绿	锁	钥	77 青虹追命针	
69 血	龙	丹	78 人	参
6A 人	参	粉	79 长白参	王
6B 暴雨梨花针			7A 雪	莲
6C 真	钢	剑	7B 何	首
6D 终南山路图			7C 大	铁
6E 菩提软玉			7D 魔教铁牌	

你可拥有 16 件物品，需增添某件物品，只要在一未用的项目（数据为 00 00 00 00）填上物品代号和数量即可，当然你也可修改某件物品的数量或减少某些物品甚至删除某件无用物品以容纳更有用的物品。

偏移 008E 起 4 个字节是武功状况记录，该记录使用一位（bit）表示一门武功，该位为 1 即表示学会了该门计功。

每位代表的武功如下所示：

偏移	008E	偏移	008F	偏移	0090
bit	武功	bit	武功	bit	武功
0	恶煞拳	0	阎王指	0	恶煞拳 4
1	五行拳	1	莲花挡	1	恶煞拳 5
2	小麒麟掌	2	大力金刚指	2	恶煞拳 6
3	青龙指	3	残影金龙斩	3	小翻浪手
4	侠影步	4	亢龙有悔	4	鹰爪功
5	火龙掌	5	龙战于野	5	魔功 1
6	金钟罩	6	恶煞拳 2	6	魔功 2
7	太极拳	7	恶煞拳 3	7	魔功 3

偏移 0091 只用了第 0 位，表示魔功 4。武功级别为恶煞拳 1 最低；魔功 4 最高。每当游戏重新开始时，傅剑寒只会五行拳、小麒麟掌、青龙指和侠影步四门武功，因此此时该处数据为 1E 00 00 00。

有了足够的物品，再加上魔功第四层，再玩这个游戏，你简直是所向无敌，以后的就不用我多说了。

浅谈 RPG 游戏的发展

湖北黄石日报社

赵礼海

笔者近期有幸玩了几个 RPG 类游戏，它们出品年代不一，游戏内容各异。从总体水平来说各有千秋，按出品年代可划分为《TIMES OF LORE 时间传说（又名国王传奇）》ORIGIN 公司 1988 年出品，载体容量 2DD；《神州八剑》台湾软件世界发行，智冠科技有限公司 1991 年出品，载体容量 2HD；《侠客英雄传》台湾大宇资讯有限公司 1991 年出品，载体容量 3HD；《中国之心》DYNAMIX 公司 1991 年出品，载体容量 7HD。

说游戏就先谈谈游戏的特色，你说对吗？

《TIMES OF LORE 时间传说》说的是发生在未知年代里的艾贝雷斯帝国的故事。这个国家由于外来入侵者的战争使国土沦陷、军阀割据，国内一片混乱、邪恶的牧师与盗匪时刻威胁人民的生命安全。你奉国王密旨去全国各地帮助人民战胜邪恶、恢复和平。

甲：时间传说可以说是一个缩小了的经典名著《创世纪》。这里有山川、河流、海洋、沙漠、城堡、地下洞穴等，完全是一个刚刚开化的地区。你要控制主人公在其中战斗、生活。喝着酒与群众交谈；拿着手斧与敌人搏斗。多么紧张！什么抓间谍、救人质、盗宝物。啊！任务可多的无法胜数，尤其完成任务后受国王嘉奖。你说多么刺激、多么光荣！谁能想到这一切全是一张 360K 磁盘上创造出的奇迹。

乙：在时间传说里作战你打不赢的话，可以脚底抹油，溜之大吉。可是面对这个游戏的英文提示，你这位英文水平不甚高的朋友可要知难而退了。除非你有完全攻略说明与战略地图（一张

360K 说明盘），否则只有抱着大本英文字典埋头苦干了。这个滋味可不足以向外人道哦！

《神州八剑》一个由中国人编写的中文 RPG。讲的也是一个未知的年代，靖南王爷篡位不成，劫持公主逃入地狱变这幅古画中去。千年后你这个现代人在参观古画时被吸入古画，处身于这个奇幻空间（古画地狱变中）里，你要指挥自己的抗暴军与凶恶的靖南王爷作战，在广阔的国土上寻宝，以达到救出公主逃出古画的目的。

（甲：在游戏中你作为一名抗暴军的指挥官，除了按月领取军晌，还可以雇佣军队甚至怪物与王爷的军队作战，攻城略地好不威风！本游戏还有战术倾向，你要利用相克的两种动物作战以达到最好的效果。另外随着经验值的增加你的等级也在相应增加，不过这也要取决于你的战功情况与找寻宝物的多少。并且这个游戏的音效不错，在 PC 机的小喇叭上也能奏出近似立体声的音乐。

乙：这个游戏最差的要算是存储功能了，这好像也是软件世界发行的几个游戏的通病。你看，好好的搭了几座桥过河，到了河中间，刚一存盘，再一读出就发现你落水了（桥没了），回又回不去，过又过不来，你说急不急死人！

甲：在这里我要感谢兰州铁路分局的闫益民先生的无私支持。如果朋友们对上述软件感兴趣，请与我或闫先生联系。我处地址：湖北省黄石日报
赵礼海 邮政编码：435002

乙：是啊！祝天下“有情人”终成“眷属”，同道皆为知己。哈！哈！

甲：还有就是神州八剑的 GAMEDATA.AY

文件,其第零与第一个字节记载的是你的等级数,第二与第三个字节记载的是经验值,第四与第五个字节记载的是月薪值,第六与第七个字节记载的是现金数,第八与第九个字节是领导能力值,第十到十一个字节记载的是法术值,善改上述数据,有助于减小游戏难度,但同时也失去游戏的乐趣,所以改时望慎之。)

相比起来《侠客英雄传》可是正统的武侠小说RPG,这段故事发生在大明末年的中国,主人公不满当时武林以大欺小、恃强凌弱的现实,更不满魔教勾结东瀛倭寇横行华夏、残害民众,毅然揭竿而起,从广州出发,战四川、打武昌,一路杀向魔教的老窝(洞庭湖上的小岛),使出一招少林达摩祖师创造的一苇神功横渡大江,足以令天下人为之倾目。

(甲:玩侠客英雄传,可是免费的全国旅行(当然全靠脚走),再与各门各派的武士决斗,学习各派武功。可辛苦了,不过要知道吃得苦中苦,方为人上人,要不然你以后如何当武林盟主啊?打累了,你可以就近到各大城市休整疗伤,并武装自己【哇!大还丹与青铜甲好贵,打这么多仗赚的战利品还不够买一件,逼的只有再造杀孽】。而且本游戏最大的特点就是有数不清的奇遇乃至艳遇向你袭来,你可要端正态度,不要犯作风问题哦!这真是大明末年、天下大乱、群雄辈出、小混混也能拳打武当老掌门、脚踹少林贼和尚,好痛快!

乙:我说这个游戏的特色就是烦,你辛辛苦苦从广州跑到武昌,眼看就要过关了,可缺一样过关法宝,还指名道姓要你去广州找,没法又只得跑回广州去,如此颠来倒去,真要命!不过这个RPG也继承了中国传统文化的精粹,连八卦阵都设计的象模象样,什么生门、休门、死门、景门等八个门进去了准保吓你个半死!不信,就请放马一试。)

好了,最后谈谈《中国之心》这套由DYNAMIX公司制作的RPG名著。这是一个外国人眼里的旧中国,在三十年代,成都的军阀邓利抢走了外籍教会护士凯蒂,准备将其纳为后宫之妃,而凯蒂的父亲是一个百万富翁,命令你这个负债累累的飞行员纳克从香港起程去救凯蒂。你这一路上要结识许许多多的人物(包括中国忍者,土匪等)请他们协助你完成这艰巨的任务。

(甲:好一幅西洋的英雄救美图。我首先声明:

中国人象邓利那么坏的很少,而象中国忍者那样乐于助人的人很多。希望老外不要误会!这个游戏最值得一提的倒是其VGA超豪华电影动画,他率先在软件设计中采用电影分镜手法,制作出的效果直逼电影专业化水准,而且所有的角色均由不同种族肤色的专业演员扮演拍摄而成,玩游戏简直是在看电影、演电影。在声效上也无可挑剔,举个例子“连臭虫叫都听得一清二楚(哈!哈!)”。世界性的旅行从香港到成都再到加德满都又到伊斯坦布尔,最后甚至到了香舍丽榭的巴黎,而且外景全是在实地拍摄的。这在PC游戏史上还是首见。

三位主角可以轮流扮演(飞行员、中国忍者、凯蒂),有时还要同时扮演三人以达到目的,这在PC游戏史上也属少见。RPG游戏中穿插坦克大对战与东方快车斗剑决两个模拟类与动作类游戏,虽小但十分逼真,这也开创了PC游戏史上的先例。

还有值得一提的是剧中根据你的所作所为提供了多种不同的结局。有最完美的结局:双双来到了巴黎,一举成名。也有老死于狱中;被杀死在邓利的庄园里,最妙的就是飞行员纳克被伊斯坦布尔的公主看中,强抢民男入宫,公然践踏国际法,一阵皮鞭、你不得不屈从,作公主的嫔妃,这回真正的吃“软”饭了。好糗哎!另外还有不少结局连笔者都没有看到……

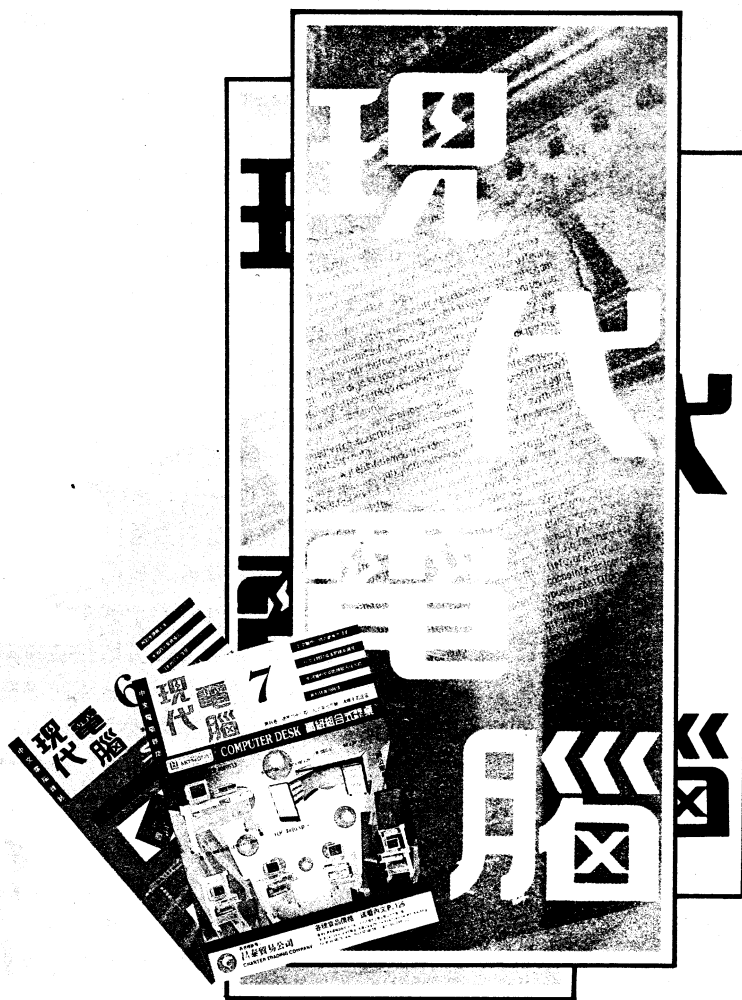
乙:不要再吹了,你不就有7M的载体容量吗!这是该游戏的优点也是其缺点所在,你看他玩不了一下就读盘,真烦人。另外英文提示的问题也在不同程度上困扰了使用中国之心的朋友,他们只有猜着玩,多没意思。)

从上述四则游戏来看,今后游戏的记忆载体容量将会越来越大,不会再有低于1M的软件出现。游戏的色彩图形也会趋向于VGA版本,其它EGA、CGA等将慢慢不再支持。故事情节将会多样化。从远古到未来;从地球到宇宙都是取材范围。而且因为广大讲中文的计算机应用者精通英文的不多,这个潜在的市场将会促使中文类软件高速发展,不光台湾、香港的公司出中文类游戏,连一些外国公司也试着生产中文软件,如日本光荣公司的三国志III。所以可以断言,今后将是个中文化的世界,喜欢游戏软件而又英文水平不高的朋友将不再烦恼。(418)

全球第一本帶領閣下

叩開中國訊息市場大門

的全中文電腦資訊月刊！



欲

有任何查詢 請電：5514304

廣州聯絡處：廣州市德政北路393號

郵編：510055

現代電子

微型電腦 專刊

電子技術

工業自動化

兼

售

MS—DOS6.0 的防病毒功能

广州汽车配件公司 顾子明

MS—DOS6.0 系统有一套防病毒的软件,它们的使用翻译如下:

一、VSAFE

动态监视计算机中的病毒,当发现病毒时,发出报警,vsafe 是一个使用 22k 内存的常驻内存程序。

在运行 WINDOWS 不能使用 vsafe 命令。

语法: vsafe [/option[+m-]...] [/ne] [/nx] [/axm /cx] [/n] [/d] [/u]

参数: option

规定 vsafe 如何监视病毒,在数字后使用加或减号置选择开或关,以下是供选择的数字。

- 1、完全删除硬盘数据的格式化警告,缺省值为“开”。
- 2、某程序企图驻留内存警告,缺省值为“关”。
- 3、防止程序写磁盘,缺省值为“关”。
- 4、检查 ms-dos 打开的可执行文件,缺省值为“开”。
- 5、检查所有磁盘的自举扇区病毒,缺省值为“开”。
- 6、对企图写硬盘的自举扇区和分区表发出警告,缺省值为“开”。
- 7、对企图写软盘自举扇区发出警告,缺省值为“关”。
- 8、对企图更改可执行文件发出警告,缺省值为“关”。

/ne 防止 vsafe 装入扩展内存。

/nx 防止 vsafe 装入扩展内存。

/ax 置热键为(alt)+(x) (x 为任意键)

/cx 置热键为(ctrl)+(x) (x 为任意键)

/n 允许 vsafe 监视网络驱动器上可能出现的病毒。

/d 关闭检查总结。

/u 把 vsafe 移出内存。

注意事项:

在安装 WINDOWS 前,关闭 vsafe。如果 vsafe 正在运行,不可能完成 windows 安装。如果要在 WINDOWS 上使用 vsafe,则要进行 mwavtsr.exe 常驻内存程序,并要把该命令按以下方法加到 win.ini 文件中:

load=mwavtsr.exe

mwavtsr.exe 使 vsafe 的信息能在 WINDOWS 中显示。

例:规定 vsafe 不检查删除硬盘全部数据的格式化,对企图写软盘自举扇区发出警告,并用(alt)+(t)键作为显示 vsafe 热键,则可使用以下命令。

vsafe /1- /7+ /at

二、msav

扫描计算机中已知病毒。

语法: msav[drive:] [/sm /c] [/r] [/am /l] [/n] [/p] [/f] [/video]

参数:

drive:

规定 msav 扫描病毒的驱动器,如果有作规定 msav 则扫描当前

驱动器。

/s 扫描规定的驱动器,但不清除发现的病毒。

/c 扫描规定的驱动器,并清除 msav 发现的病毒。

/r 建立一个 msav.rpt 文件,该文件列出 msav 查到病毒的文件数目,发现病毒的数目,以及清除病毒的数目,如果缺省,msav 不建立该文件,msav.rpt 文件,将建立在根目录上。

/a 除 a 和 b 驱动器外扫描所有驱动器。

/l 除网络驱动器外扫描所有本地驱动器。

/n 如果 msav.txt 文件存在、显示其包含的内容,该文件在含有 msav.exe 文件的目录中,然后 msav 扫描当前驱动器和规定驱动器。msav 不使用图形界面。如果 msav 查出病毒,则返回 86 码代替在屏幕上显示信息。

/p 以命令行界面代替图形界面显示。

/f 不显示已被扫描的文件名。该参数只能与 /n 或

/p 参数一块使用。

/video 显示一系列影响 msav 如何显示的参数,这一系列参数如下所列。

/25 置屏幕显示为 25 行,为缺省值设置。

/28 置屏幕显示为 28 行,该参数只能用于 vga 显示器。

/43 置屏幕显示为 43 行,该参数只能用于 ega 和 vga 显示器。

/50 置屏幕显示为 50 行,该参数只能用于 vga 显示器。

/60 置屏幕显示为 60 行,该参数只能用于 video7 显示器。

/in 使用彩色方案运行 msav。即使不知是否是彩色显示器。

/bm 使用黑白方案运行 msav。

/mono 使用单色方案运行 msav。

/lcd 使用 lcd 方案运行 msav。

/ff 在使用 cga 显示器的计算机上进行最快速屏幕更新。该参数会降低显示质量。

/bf 利用计算机 BIOS 显示。

/nf 允许使用变体字。

/bt 允许在 WINDOWS 中使用图形鼠标。

/ngm 使用缺省鼠标符代替图形鼠标符运行 msav。

/le 变更左右鼠标按钮。

/psz 如果鼠标光标消失或锁死重置鼠标。

注意事项:

chklist.ms 文件

在扫描病毒时 MICROSOFT 反病毒程序将在每个目录上建立一个 chklist.ms 文件,MICROSOFT 反病毒程序将每个程序的检查结果记录在 chklist.ms 文件中,在随后的扫描期间,则用新的检查结果与原来的检查结果相比较以确定程序是否已被更改。如果检查结果值有差异,则表明可能发生病毒感染。

真正的病毒误报警迹象

易受侵害的程序,一些常住内存的程序,以及硬件间的冲突可能会引起计算机好象产生受病毒感染的动作,但实际上并非如此。如果怀疑计算机受到感染,仔细考虑引起所见迹象的原因,不是计算机病毒,而是硬件或软件困难可能会引起错误。

检查所有文件选择项

当检查所有文件选择被选择时,MO-DOS 反病毒只检查带下列扩展名的文件:.386,.app,.bin,.cmd,.dom,.dll,.drv,.exe,.fon,.ico,.ov*,.pgm,.pif,.prg,.sys。

msav.ini 及 msdosdata 环境变量。

在启动 MS-DOS 反病毒程序时,将使用一个名为 msav.ini 的初始化文件配置该程序。为了确定 msav.ini 文件的位置,ms-dos 反病毒程序首先要寻找 msdosdata 环境变量,该变量规定了初始化文件的路径。如果没有定义该环境变量,则在启动 MS-DOS 反病毒程序的目录中查找 msav.ini 文件,如果反病毒程序没找到该文件,该程序就会在该目录中使用缺省值建立一个。

如果与其他用户共享 msav.exe 文件 msdosdata 环境变量是有用的,除非使用自己为 MS-DOS 反病毒程序的配置。例如:假设 msav.exe 文件在驱动器 p 的根目录上(一个只读网络服务器),你想在使用 MS-DOS 反病毒程序时使用自己的配置。规定使用在自己 c 盘 dos 目录上的 msav.ini 文件代替使用在网络上 p 盘的初始化文件,则需在自己的 autoexec.bat 文件中加上下列命令:set msdosdata = c:\dos

在启动 MS-DOS 反病毒程序时,键入下列命令:

p:\msav

把 MS-DOS 的反病毒程序的配置更改存贮在 c 盘 dos 目录上的 msav.ini 文件中。则使用 /n 参数。

如果规定了 /n 参数,如果 msav.txt 文件存在且在 msav.exe 文件所在的目录中,msav 会显示 msav.txt 文件的内容,msav 然后扫描当前驱动器和规定驱动器,msav 不使用图形界面,如果 msav 发现一个病毒,则返回存在码 86 代替在屏幕上显示一条信息。

例:用黑白方案使用 msav 程序,并规定除 a 和 b 驱动器外检查所有驱动器,可使用下列命令:

msav /bw /a

写一简单名为 virus 的批处理程序支持 msav 存在码和 /s 参数扫描当前驱动器,可使用 MS-DOS editor 编辑器输入下列命令:echo off

```
rem smith's msav command
msav /s /n
if errorlevel 86 goto virus
if not errorlevel 86 goto none
:virus
echo msav has detected a virus on your
current drive!
goto exit
:none
echo msav found no viruses on your
current drive
goto exit
:exit
```

419

最近,在广州又出现了一种新的系统型病毒,因该病毒的特征字为 1990,故笔者取其名为 1990 病毒。

该病毒隐藏在硬盘的主引导扇区或软盘的引导扇区中,通过带毒盘的启动而常驻内存。

该病毒如同其它一般的系统型病毒一样,更改 0:0413 的值,使内存减少 1K,故易用 PCTOOLS 等工具发现。

该病毒修改原磁盘中断 int 13H 或修改软盘中断 int40H,使其指向病毒中断的入口 XXXX:0110。当病毒驻留内存时,观察中断向量表的 0:4CH 或 0:100H,会发现其中的变化。

病毒感染软盘时,把软盘的原引导扇区的内容搬到 0 道 1 头 3 区中,故清除软盘中的 1990 病毒,只须把 0 道 1 头 3 区的原引导程序调入内存,再覆盖到软盘的 0 道 0 头 1 区,或把磁盘密度相同的、相同 DOS 版本的、不带病毒的软盘的引导扇区调入内存,再覆盖到带毒软盘中即可。

病毒感染硬盘时,保留原硬盘主引导分区参数而抛弃原硬盘的主引导程序。由于该病毒在设计时已考虑到硬盘启动问题,故正常情况下,带毒的硬盘是可以启动的,只是速度会减慢而已。

清除硬盘的 1990 病毒,可反其道而行之,即可取任一不带病毒的硬盘主引导程序去替代带毒硬盘主引导扇区中的病毒程序,但一定要保留偏移量 01BE 至 01FF 的主引导扇区中的分区参数

一种新的 1990 病毒

胡向东

表。这一清毒方法可对任一系统型病毒有效,笔者也多次介绍过。

1990 病毒的首语句是 EB 34 JMP 0136。这与 MSDOS3.3 的首语句相同,故用 DEBUG 把带毒软盘的引导扇区调入内存,若不细心观察,很容易受蒙骗。

该病毒在偏移量为 00A4 和 014FH 处,都有病毒的特征字:1990,这可作为识别这种病毒的一个判据。

该病毒在程序设计中,既没有任何标新立异令人怀疑的屏幕显示,也没有象一些“杀手”病毒那样对磁盘中的数据恶意删除,而只是侧重于进行无声无息的传染。

而正是这种用现有的 SCAN 清病毒软件查不出来而又令人容易麻痹大意的病毒,由于其驻留内存后经常进行写盘感染的操作容易造成硬盘的主引导扇区的分区参数丢失或 FAT 表的破坏。

综观 1990 病毒,在病毒设计方面完全没有有什么新花招,只不过是过去的一些系统型病毒程序改头换面、东拼西凑之作而已。

根据笔者判断,1990 病毒的作者,只是一个令人谴责的、毫无职业道德的初学者。

笔者已经编写了 1990 病毒的清毒程序,需要咨询数据修复及病毒清除问题,欢迎与本人联系。地址:广州龙口西路 1-38 号,邮编:510630,电话:7570626,7570627,5515961-5931,BB 机:3322322-4364。单位:正大电脑应用与数据修复有限公司。

420

单片机与模糊控制讲座

第七讲 MC6805 的指令系统和程序设计(上)

广东工学院 余永权

MC6805 单片机的指令系统是 MC6801 的指令系统的子集。这个指令系统有 10 种寻址方式,可以访问 64K 存储空间。在采用数字单片机组成模糊控制器时,模糊控制的实现是通过软件算法来完成的;为编制优良的模糊控制程序,必须熟悉单片机的指令系统。单片机 MC6805 适用于对家用电器,工业生产过程实行模糊控制,故此,在这里介绍它的指令系统。

一、寻址方式

MC6805 有 10 种寻址方式,有利于用户采用最佳的寻址方法进行程序编制,从而可以产生优化的模糊控制程序。这 10 种寻址方式是:

1. 立即寻址 IMM 指令中操作码后的操作量就是操作数。例如,“LDA #50”,意义是 50→A。

2. 直接寻址 DIR 操作码后边的是操作数地址。直接寻址指令为双字节指令,可访问存储器地址 \$0000~\$00FF,即 256 个字节。例如,“LDA \$40”指令是把 \$40 地址的内容送到累加器 A,即(\$40)→A。

3. 扩展寻址 EXT 和直接寻址类似,只是操作数地址为二个字节,故可访问 64K 存储器空间。这种寻址方式指令为三字节指令。例如,LDX \$1234。

4. 相对寻址 REL 相对寻址方式只用于转移和转子指令,转移地址等于 PC 加上操作码后边的 8 位有符号整数。相对寻址的地址为本指令地址为基地址的-126 到+129 字节的地址。例如,BCC LABEL。

5. 无偏址变址 IX 变址寄存器 X 的内容是操作数的地址。这种寻址方式可对 \$0000~\$00FF 这 256 个地址进行寻址。例如,LDA X。

6. 带 8 位偏移量变址 IX1 变址寄存器 X 的内容加上操作码后的一字节无符号数形成操作数地址。这种寻址方式的最高寻址地址是 \$1FE。这种寻址方式用于进行表格元素查找十分方便。例如,STA \$40.X。

7. 带 16 位偏移量变址 IX2 变址寄存器 X 的内容加上操作码后的二个字节无符号数形成操作数地址。这种寻址方式和带 8 位偏移量变址类同,只是偏移量为 16 位整数而已。它可以访问 64K 存储器空间中的任何表格。例如,ADD \$1200.X。

8. 对位置“1”清零 BSC

操作码后边的一个字节指明被操作的数的地址,即相当于直接寻址;被操作数的位则由操作码指明。在存储器的 \$0000 到 \$00FF 地址中,任何位都可以用这种方式进行置“1”或“清零”,例如,BCLR 3.\$70。

9. 位测试转移 BTB 这种寻址方式把直接和相对寻址方式结合。被测试的位和测试条件包含在操作码中;被测试的字节地址在操作码后边的一个字节中,而带符号的 8 位偏移量位于第三字节。测试成立时,则偏移量加到 PC 上。故这种寻址方式对 \$0000~\$00FF 的 256 个地址内容进行测试;而转移范围则是本指令地址为基地的-125 到+130 个字节之间。例如,BRSET 5.\$6A.LAB2。

10. 隐含寻址 1NH

操作数地址和目的地址都包含在操作码中。例如,“TAX”指令的意义是把累加器 A 的内容传到变址寄存器 X 中。

二、指令系统

MC6805 单片机的指令系统及其操作码的值如表 1 中所示。在表中,还给出了各种指令的变址方式。在表中的每个注明指令汇编符的格中,左上角的数字是 HMOS 工艺的单片机的周期数,右上角的数字是 CMOS 工艺的单片机的周期数,左下角是指令字节数,右下角是寻址方式。

在表 1 中,可以在汇编符的第一行的第三个位置上找到指令 BRA。从左下角信息可知它是二字节的相对寻址指令。故它的格式必定为 BRA LAB1。其中 BRA 是操作码,其代码为 \$10,而 LAB1 是相对转移量或标号。如

表 1 指令系统及操作码表

Hi Low	位操作		转移	读/修改/rj				控制		寄存器/存储器						Hi	Low
	BTB	BSC	REL	DIR	A	X	IX1	IX	INH	INH	IMM	DIR	EXT	IX ₂	IX ₁		
0	0000	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
0000	0000	0010	0001	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111	
1	BRSET0	BSET0	BRA	NEG	NEG	NEG	NEG	NEG	RTI		SUB	SUB	SUB	SUB	SUB	SUB	0
0010	BRCLR	BCLR0	BRN	DIR	A	X	IX1	IX	INH		IMM	DIR	EXT	IX2	IX1	IX	0000
2	BRSET1	BSET1	BHI						RTS		CMP	CMP	CMP	CMP	CMP	CMP	1
0001	BRSET1	BSET1	REL						3	INH	SBC	SBC	SBC	SBC	SBC	SBC	0001
3	BRCLR1	BCLR1	BLS	COM	COM	COM	COM	COM	SWI		IMM	DIR	EXT	IX2	IX1	IX	2
0011	BRCLR1	BCLR1	REL	DIR	A	X	IX1	IX	INH		CPX	CPX	CPX	CPX	CPX	CPX	0011
4	BRSET2	BSET2	BCC	LSR	LSR	LSR	LSR	LSR			AND	AND	AND	AND	AND	AND	4
0100	BRSET2	BSET2	REL	DIR	A	X	IX1	IX			IMM	DIR	EXT	IX2	IX1	IX	0100
5	BRCLR2	BCLR2	BBS								BIT	BIT	BIT	BIT	BIT	BIT	5
0101	BRCLR2	BCLR2	REL								IMM	DIR	EXT	IX2	IX1	IX	0101
6	BRSET3	BSET3	BNE	ROR	ROR	ROR	ROR	ROR			LDA	LDA	LDA	LDA	LDA	LDA	6
0110	BRSET3	BSET3	REL	DIR	A	X	IX1	IX			IMM	DIR	EXT	IX2	IX1	IX	0110
7	BRCLR3	BCLR3	BEQ	ASR	ASR	ASR	ASR	ASR			TAX	STA	STA	STA	STA	STA	7
0111	BRCLR3	BCLR3	REL	DIR	A	X	IX1	IX			INH	STA	STA	STA	STA	STA	0111
8	BRSET4	BSET4	BHCC	LSL	LSL	LSL	LSL	LSL			CLC	EOR	EOR	EOR	EOR	EOR	8
1000	BRSET4	BSET4	REL	DIR	A	X	IX1	IX			INH	EOR	EOR	EOR	EOR	EOR	1000
9	BRCLR4	BCLR4	BHCS	ROL	ROL	ROL	ROL	ROL			SEC	ADC	ADC	ADC	ADC	ADC	9
1001	BRCLR4	BCLR4	REL	DIR	A	X	IX1	IX			INH	ADC	ADC	ADC	ADC	ADC	1001
A	BRSET5	BSET5	BPL	DEC	DEC	DEC	DEC	DEC			CLI	ORA	ORA	ORA	ORA	ORA	A
1010	BRSET5	BSET5	REL	DIR	A	X	IX1	IX			INH	ORA	ORA	ORA	ORA	ORA	1010
B	BRCLR5	BCLR5	BMI								SEI	ADD	ADD	ADD	ADD	ADD	B
1011	BRCLR5	BCLR5	REL								INH	ADD	ADD	ADD	ADD	ADD	1011
C	BRSET6	BSET6	BMC	INC	INC	INC	INC	INC			RSP	JMP	JMP	JMP	JMP	JMP	C
1100	BRSET6	BSET6	REL	DIR	A	X	IX1	IX			INH	JMP	JMP	JMP	JMP	JMP	1100
D	BRCLR6	BCLR6	BMS	TST	TST	TST	TST	TST			NOP	JSR	JSR	JSR	JSR	JSR	D
1101	BRCLR6	BCLR6	REL	DIR	A	X	IX1	IX			INH	JSR	JSR	JSR	JSR	JSR	1101
E	BRSET7	BSET7	BIL						STOP			LDX	LDX	LDX	LDX	LDX	E
1110	BRSET7	BSET7	REL						INH			LDX	LDX	LDX	LDX	LDX	1110
F	BRCLR7	BCLR7	BIH	CLR	CLR	CLR	CLR	CLR	WAIT			STX	STX	STX	STX	STX	F
1111	BRCLR7	BCLR7	REL	DIR	A	X	IX1	IX			INH	STX	STX	STX	STX	STX	1111

LAB1 是相对转移量, 则它是连符号在内共为一字节的数; 如果 LAB1 是标号, 则这个标号的地址必须在 BRA 指令地址的 -126 到 +129 字节范围内。

在表 1 中给出 10 种寻址方式, 结合表 1 给出的有关信息, 对各种寻址方式举例说明如下:

```
SUB #data : A ← data → A
SUB addr8 : (A) ← (addr8) → A : DIR 方式
SUB addr16 : (A) ← (addr16) → A : EXT 方式
SUB .X : (A) ← ((X)) → A : IX 方式
SUB addr8.X : (A) ← ((X) ← addr8) → A : IX1 方式
SUB addr16.X : (A) ← ((X)) ← addr16 → A : IX2 方式
BCS LABEL : C=1. 转向 LABEL : REL 方式
BRSET bit.addr8.REL : (addr8).bit → C : BTB 方式
; 如果 (addr8).bit=1:
; 则 (PC)+3+REL → PC:
BSET bit.addr8 : 1 → (addr8).bit : BSC 方式
RTS : 返回; 1NH 方式
```

在表 1 中, 还注明有一种寻址方式, 即是 A 寻址方式, 这种寻址方式是对 A 累加器的内容执行操作, 本质上是属于隐含寻址方式的, 故在寻址方式中没有把它单独看成一种方式。

三、指令说明

MC6805 的指令系统有基本指令 184 条, 下面分别说明这些指令的功能和操作意义:

1. ADC (A) + 操作量 + (C) → A
2. ADD (A) + 操作量 → A
3. AND (A) ∧ 操作量 → A
4. ASL 算术左移一位
5. ASR 算术右移一位
6. BCC C=0. 则转移
7. BCLR 对指定地址的指定位清 0
8. BCS C=1. 则转移
9. BEQ 相等, 则转移
10. BHCC H=0. 则转移
11. BHCS H=1. 则转移
12. BHI (C ∧ Z)=0. 即大于, 则转移
13. BHS C=0. 大于等于, 则转移
14. BIH INT=1. 则转移
15. BIL INT=0. 则转移
16. BIT (A) ∧ 操作量
17. BLO C=1. 则转移
18. BLS (C ∧ Z)=1. 小于等于, 则转移
19. BMC I=0. 则转移
20. BMI N=1. 小于, 则转移
21. BMS I=1. 则转移

22. BNE Z=0. 即不等, 则转移
23. BPL N=0. 即大于等于, 则转移
24. BRA 按偏移量转移
25. BRCLR 指定的位为 0, 则按偏移量转移
26. BRN 双字节空操作指令
27. BRSET 指定的位为 1, 则按偏移量转移
28. BSET 对指定地址指定位置“1”
29. BSR PC 入栈, 按偏移量转移
30. CLC 0 → C
31. CLI 0 → I. 即开中断
32. CLR 对操作地址清 0
33. CMP (A) 和操作量比较
34. COM 对操作量取反
35. CPX (IX) 和操作量比较
36. DEC 操作量减 1
37. EOR (A) ⊕ 操作量 → A
38. INC 操作量加 1
39. JMP 无条件转移
40. JSR 调用子程序
41. LDA 取数到 A
42. LDX 取数到 IX
43. LSL 逻辑左移一位
44. LSR 逻辑右移一位
45. NEG 对操作量取负
46. NOP 空操作
47. ORA (A) ∨ 操作量 → A
48. ROL 循环左移
49. ROR 循环右移
50. RSP 堆栈指针恢复指令, \$07F → SP
51. RTI 从中断返回
52. RTS 返回
53. SBC (A) - 操作量 - (C) → A
54. SEC 1 → C
55. SEI 1 → I
56. STA 存 A 到存储器
57. STOP 停止
58. STX 存 IX 到存储器
59. SUB (A) - 操作量 → A
60. SWI 软件中断
61. TAX A → IX
62. TST 操作量 - 0, 即零测试
63. TXA IX → A
64. WAIT 等待中断

用 80 组成的数字自动增益控制

镇江市高等专科学校教务处 丁永林

【提要】本文在数字自增益控制中，合理地选用了增益衰减和提升的参考标志，从而简化了运算，使自动控制能在以 Z-80CPU 组成的简单实现，大大降低了成本。

一、语言信号的动态范围

语言信号的动态范围较宽，一个人的语言强弱变化约有 30dB。且因人而异，又因讲话人所处环境与情绪而异，综合考虑这些因素，人的语言信号动态范围有可能达 (60—70)dB。由此在通讯或音响设备中对语言信号进行量化编码时遇到的困难是：为了适应较大输入信号的要求，要将量化阶距选得足够大避免过冲；又要将量化阶距选得足够小不致使小幅度信号淹没。

数字自动增益控制就能较好地解决上述困难，当输入信号大时自动将放大器增益减少，以防止过冲。反之则自动将放大器增益增加，使输入信号提升，以达到满意的量化分辨率。

数字自动增益控制具有控制范围大，精度高，设计灵活等优点。其组成框图如图 1。

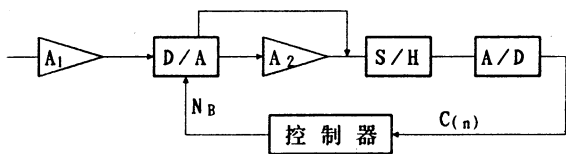


图 1 数字自动增益控制组成框图

图中 A_1 为前置放大器；D/A 与 A_2 组成数控衰减器；S/H 为采样保持电路；A/D 为模数转换电路；控制器将 A/D 输出的码化字 $C(n)$ 经过运算和比较后，输出一个适当的控制字给 A/D 输入端，以改变数控衰减器的增益。

其中关键部分是数控衰减器与控制器，现就它们的组成与设计叙述如下。

二、数控衰减器电路

数控衰减器电路有多种，这儿采用一片 AD7520 作 D/A 转换，用一片 F007 作输出放大器，其具体连接如图

2：

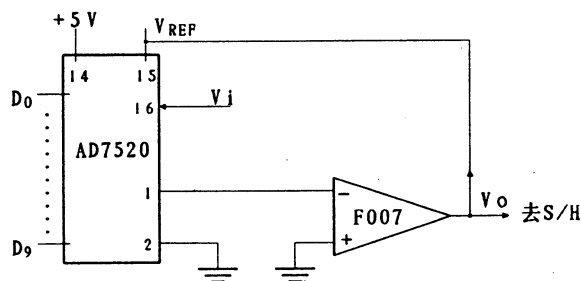


图 2 数控衰减器电路

将 F007 输出端 V_o 接入 AD7520 的 V_{REF} 端，语言输入信号 V_i 由 AD7520 反馈电阻端 (第 16 脚) 输入，运放的输入电阻为 R 。根据运放的虚地概念，可得 D/A 输入与输出之间的表达式：

$$\therefore I_{OUT1} = \frac{V_o}{R} (a_1 \cdot 2^{-1} + a_2 \cdot 2^{-2} \dots + a_{10} \cdot 2^{-10})$$

$$- I_{OUT1} = \frac{V_i}{R}$$

$$\therefore V_o = - \frac{V_i}{(a_1 \cdot 2^{-1} + a_2 \cdot 2^{-2} \dots + a_{10} \cdot 2^{-10})}$$

放大器闭环增益：

$$A = \frac{V_o}{V_i} = - \frac{1}{(a_1 \cdot 2^{-1} + a_2 \cdot 2^{-2} \dots + a_{10} \cdot 2^{-10})} \\ = - \frac{2^{10}}{D_9 \cdot 2^9 + D_8 \cdot 2^8 \dots + D_0 \cdot 2^0}$$

$$\text{令：} A_0 = 2^{10} \quad N_B = D_9 \cdot 2^9 + D_8 \cdot 2^8 \dots + D_0 \cdot 2^0$$

$$\text{有 } A = - \frac{A_0}{N_B}$$

根据上式可求出衰减器每帧作 3dB 衰减时，D/A 输入端所需数字量的控制字。列入表 1。

表1 衰减器输入控制字

序号	A	衰减量	控制字
1	-1024	0dB	001H
2	-512	-6dB	002H
3	-341	-9dB	003H
4	-256	-12dB	004H
5	-171	-15dB	006H
6	-128	-18dB	008H
7	-93	-21dB	00BH
8	-64	-24dB	010H
9	-46	-27dB	016H
10	-32	-30dB	020H
11	-23	-33dB	02DH
12	-16	-36dB	040H
13	-11	-39dB	05AH
14	-8	-42dB	080H
15	-6	-45dB	0B6H
16	-4	-48dB	100H
17	-3	-51dB	16AH
18	-2	-54dB	200H
19	-1.4	-57dB	2D3H
20	-1	-60dB	3FFH

D/A 输入端控制字不允许为 000H。否则入放大器将工作于开环状态, F007 工作于饱和区, 电路工作失去正常。

三、控制器电路

控制器的任务是将 A/D 转换后的数字量通过计算, 依输入数字量的大小, 输出适当的控制字给数字衰减器, 以取得合适的放大器增益; 并能按要求协调各部份工作。

1. 通常的计算方法:

通常控制器计算方法是计算语言信号在短时直角窗

内的方差 $\sigma_{(n)}^2 = \sum_{m=1}^n C_{(m)}^2$, $C_{(m)}$ 为 A/D 输出的码化字。

已知衰减器增益 $A = -\frac{A_0}{N_B}$ 。使 N_B 正比于语言信号

标准差 σ 。可得 $A = -\frac{A_0}{\sigma}$, 即可完成控制。但这样需要电

路完成平方、求和、开方等复杂运算, 对专用语言处理芯片完成这些运算并不困难, 但成本较高。价廉而使用广泛的 Z-80 CPU 难以完成这些运算, 为此我们需寻找另外的办法。

2. 控制器设计

电路采用 AD7574 作 A/D 转换器, 数字输出为 8 位, 采用偏移二进制编码, 参考电压取 $V_{REF} = \pm 10V$ 。并要求在小信号输入时, 其输出编码最小分辨率为 4 位。

(一) 工作区域的划分:

设置数字衰减器的起始放大倍数 $A_N = 32$ (-30dB) 要求控制器输出的起始控制字为 020H。将工作区域划分为三段:

(1) 正常工作区: 当输入语言信号动态范围小于 (0-0.3)V, 大于 (0-19)mv 时, 经数控衰减器放大 32 倍, A/D 转换器输入端可得一个动态范围小于 (0-10)V 大于 (0-0.63)V 的输入信号。

(2) 衰减区: 当输入语言信号动态范围大于 (0-0.3)V 时, 经放大 A/D 输入端信号将可能大于 10V, 其输出必产生过冲, 为此要求控制器快速增大其输出控制字的数值, 使数控衰减器进行每点 3dB 衰减。最大衰减量为 -60dB。即放大器放大倍数为 1, 控制器输出控制字为 3FFH。这时允许输入语言信号幅度的动态范围可大到 (0-10)V。

(3) 提升区: 当输入语言信号的动态范围小于 (0-19)mv, 经数控衰减器放大 32 倍后, A/D 输入端信号仍小于 (0-0.63)V, 达不到 A/D 转换器 ± 8 个量化阶距, 不能满足给定的最小量化分辨率要求, 需控制器减少其输出控制字数值, 数控衰减器放大增益作每帧 3dB 提升。数控衰减器最大增益为 0dB。此时放大倍数为 1024, 允许输入语言信号幅度的动态范围可小到 (0-10)mv 控制器输出控制字应为 000H。

(二) 比较标准的选取

(1) 衰减区比较标准是控制器判断 A/D 输出的每一个码化字是否出现满度值 FFH。若出现则作快速衰减, 不出现则保持原有的控制字。

(2) 提升区的比较标准选择是一个困难问题, 因为语言信号的随机性, 我们无法预知语言信号在不同情况下的数值, 并由此给出一个比较标准。

已知在均匀量化系统中, 量化后的信噪比用 SNR (dB) $\approx 6.02B - 7.2$ 式计算 (B 为量化位数)。当电路中 A/D 芯片选定后, 电路所能保证的最大 Bmax 确定

(SNR_{max} 确定),但在 A/D 转换过程中实际的 B 是与语言段中信号幅度 X_{\max} 有关的量。只有当 X_{\max} 等于 A/D 的 V_{REF} 时才能达到 B_{\max} 。当语言段中信号幅度 X_{\max} 较小时,实际 B 值将很小,例如本设计中 A/D 输入端信号幅度 $X_{\max} < 140\text{mv}$ 时, $B=1$, $\text{SNR} < 0$ 。信号将被背景噪声淹没。为此在设计中必须规定要求每个语言段信号量化必须达到的最小 B 值,即经 A/D 转换输出的最小分辨率。本设计中选 $B=4$ 。当然随应用场合的不同也可选 $B > 4$ 。

选 $B=4$ 时,在偏移二进制编码中 A/D 输出应占有 ± 8 个量化阶距,如只考虑正值,8 个量化阶距的码字为 88H。以此作提升区的比较标准。当选 $B > 4$ 时,只需相应修改程序中比较环节的比較字。

(三) 计算窗口宽度与计算方法

根据语言信号处理知识,语言信号具有短时平稳性,一般认为在 20ms 时间段内的语言信号具有相对平稳性,并知汉语最短音节长度为 7ms。我们只需取一段时间内码化字中的最大值与 88H 比较即可,本设计为了使增益得到及时补偿,取计算窗口宽度为 5ms。在采样频率为 8KHz 情况下,每窗有 40 点,其中计算 32 点用 8 点时间作为输出控制时间。

计算方法是在 Z-80 CPU 中设置二个比较环节,将前一点输入的码化字与后一点输入的码化字进行比较,存其大者,舍去小者,比较 32 点后,得 5 ms 时间内最大值再与 88H 比较,若小于 88H 则作 3dB 提升,一窗结束后再进入第二窗计算。

3. 控制器的组成

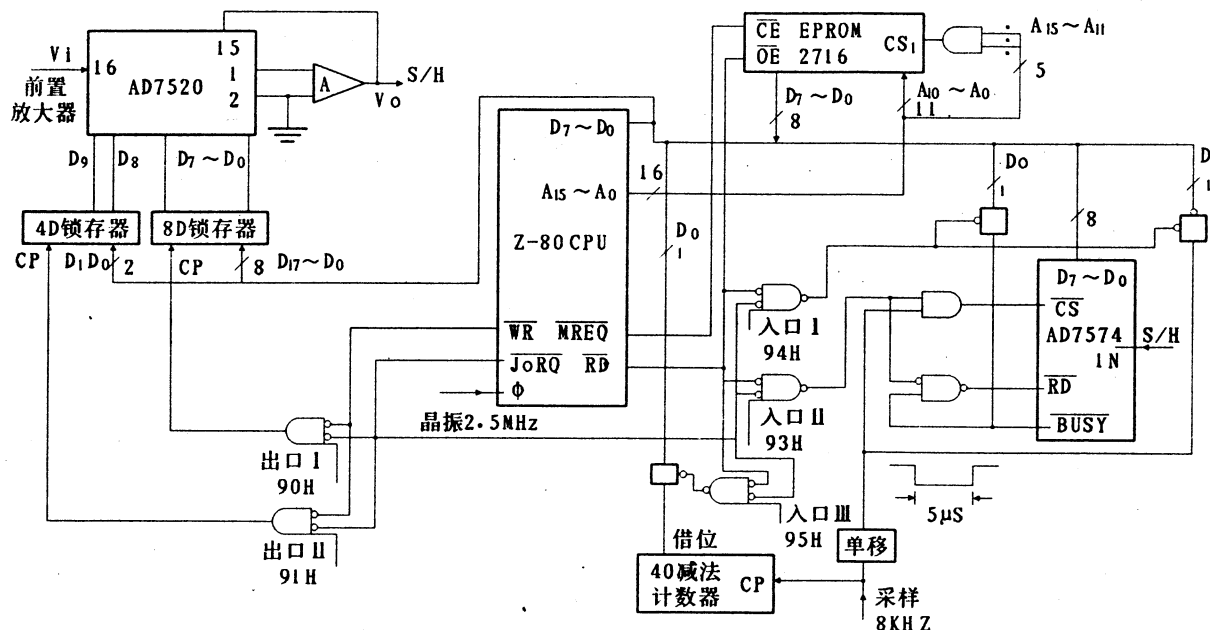


图3 控制器原理图

控制器由 Z-80CPU, EPROM2716, 时钟振荡器, 采样振荡器, 定时器, 及一些接口电路组成的最小微机系统如图 3 所示,图中各器件与 CPU 的连接均已详细绘出。其它均为一般脉冲数字电路,所以从简。

4. 控制器程序流程

控制器程序可借用 Z-80 单板机 TP-801A 进行调试,调试中可利用 PIO A 口(或利用 Z-80 CTC)进行中断请求以节省硬件,但因 PIO 中断一次只能进行一次数据传送,所以数据的输入与输出均不经过 PIO,而是将输入、输出数据的程序均汇编在中断服务程序中。

(一) 出、入口地址选定

(1) 确定四个入口地址

A/D 输出码化字入口地址 93H

A/D 状态(BUSY)入口地址 94H

5ms 定时器状态入口地址 95H

PIO 中断请求入口地址

(2) 确定二个出口地址

控制字高二位出口地址 91H

控制字低八位出口地址 90H

(3) 中断服务程序入口 2800H

(二) 寄存器分配

计数器一个 B 用以记录计算点数。

输入数据寄存器(8 位)H

大数寄存器(8 位)L

控制字地址码寄存器(16 位)DE。

(三) 控制字存储空间 38 个, 2C00H-2C27H

将表 1 所列 20 个控制字自 001H-3FFH 依次存入

2C00H—2C37H, 每个控制字为 10 位需占 2 个空间。其示意图见图 4。

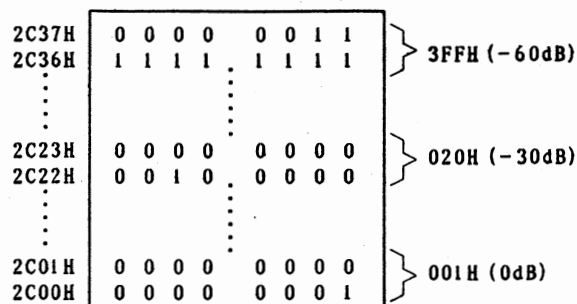


图 4 控制字存储空间示意

(四) 在程序调试中, 中断请求信号可在语言信号输入

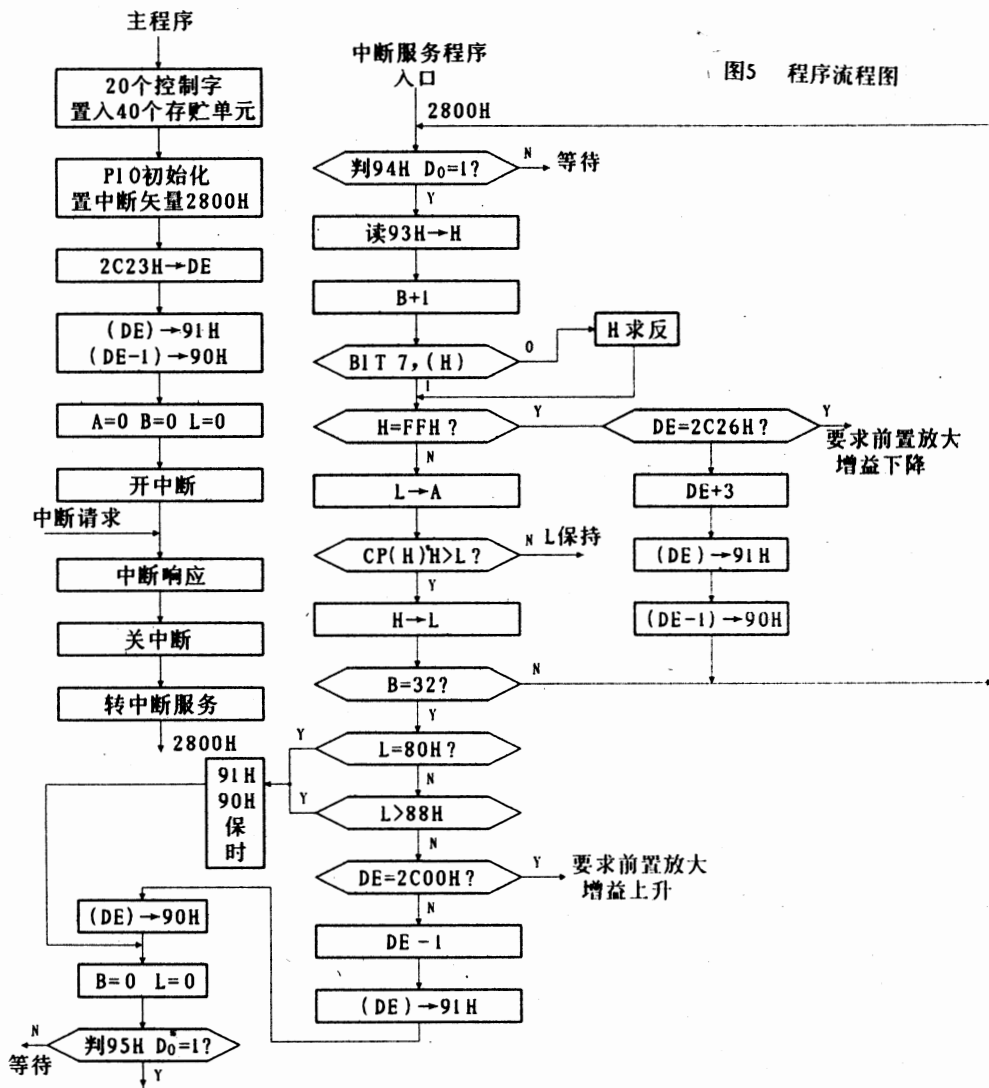
前, 人为地在 PIO A 口, $\overline{\text{ASTB}}$ 端给一负脉冲信号。当 CPU 响应中断并转入中断服务程序后, 除去收到复位信号 $\overline{\text{RESET}}$ 外, 即不再返回主程序, 这样既节省硬件又简化程序, 但是在语言的间隙期, 系统增益会增至最大值。为解决这一问题, 在程序中增加一个判断 5ms 后, L 最大值 是否为 80H? 如果是, 则是语言间歇期间立即转入读下一窗, 不作增益提升。

(五) 在调试程序中, 不应出现有随机写入存储器的数据, 这样在调试结束后可根据 EPROM 的要求, 将控制程序作适当修改和预置数据一并写入一片 2716。

(六) 调试程序流程见图 5, 具体程序不再一一列出。

三、结束语

以上设计说明数学自动增益控制, 完全可以在用 Z-80 CPU 组成的最小微机系统中实现, 从而大大降低成本, 并完全体现数字自动增益控制的控制范围宽, 输出信



号平稳和设计灵活的特点。

当需要将系统分辨率提高时,例如用 12 位 A/D 片 AD574 代换 8 位 A/D AD7574。本设计只需将 A/D 输出数据中低四位自动丢失, CPU 只读入 A/D 高八位数据,并适应 AD574 要求将接线作相应改变,其它无需改动即可适应工作。

参考文献:

- 〔1〕语言信号数字处理 科学出版社 〔美〕L. R. 拉宾纳 R. W. 谢弗著
〔2〕集成 A/D 和 D/A 转换器应用技术 国防工业出版社 于继洲编著
〔3〕微型计算机硬件软件及其应用 清华大学出版社 周明德编著

422

小 辞 典

关于电脑技术的英语单词或词组,如果在许多的字典中都找不到其中文解释,那么不妨查一下下面的小辞典(续上期)。

A

accounting file 计算文件
accounting management 帐务管理
account management 帐目管理
account resource management (ARM) 帐目资源管理
accounts payable module 应付帐模块
accounts receivable module 应收帐模块
ACE (Advanced Computing Element) 先进计算单元
ACE (Advanced Computing Environment) 先进计算环境
ACF (Access Control Field) 存取控制域[字段]
ACIA (Asynchronous Communication Interface Adapter) 异步通信接口适配器
ACID (Atomicity, Consistency, Isolation, Durability) 原子性、一致性、隔离性与持久性
ACK (acknowledgment) 肯定,应答,确认,接收
acknowledgment (ACK) 肯定,应答,确认,接收
ACL (Access Control List) 访问控制表
ACMS (Application Control and Management Services) 应用程序控制和管理服务
ACMS (Application Control and Management System) 应用程序控制和管理系统
acoustic transducer 声传感器
ACP (Architectural Cost Program) 建筑成本估价软件包

acquaintance 熟知
acquisition time limit 学会时间
ACSE (Associated Control Service Element) 相联控制服务单元
ActionMedia II development toolkit AM II 开发工具箱
action widget 动作界面对象
active database 主动数据库
active hub 有源集线器
active matrix 活动矩阵,有源矩阵
active monitor 活动监督器[站]
active object 主动对象
active pull down 有源下拉
active state 活动状态
active window 活动窗口
activity 开发活动,活动,活动性,能动性;功率,效率,占空系数
activity monitor 活动监督程序
act-link 激活链路
actor/thread 操作者/线程
acttrace 激活跟踪
ACU (Automatic Calling Unit) 自动呼叫部件,自动呼叫单元
ACVC (Ada Compiler Validation Capability) Ada 编译程序验证能力
AD (Audio Disc) 音频盘
Ada compiler validation capability (ACVC) Ada 编译程序验证能力
Ada intermediate machine (AIM) Ada 中介机器
adaline 自适应线性元件
Ada programming support environment (APSE) Ada 程序设计支持环境
Ada program support environment (APSE) Ada 程序支持环境
adapter 适配器,转接器
adapter description file (ADF) 适配器描述文件

423

BROTHER 打印机及传真机

发布会暨产品展示会

日本兄弟工业株式会社和总代理三井物产(香港)有限公司主办,并由广州新力科技发展有限公司,广州华昆电子印刷办公设备公司和深圳托福科技有限公司协办, brother 打印机及传真机发布会暨产品展示会于十一月五日在广州举行。

在这次发布/展示会上,兄弟公司展示了他的新产品——高负荷高速汉字打印机,除了拥有西文 450 字/秒和汉字 200 字/秒的超高打印速度外,更同时内置宋体和黑体两种汉字库,使用户有更多的选择。

这次向中国市场推出全新的普及型传真机型号 FAX-600。基本功能包括自动加插文件封面、自动切纸、自动识别电话/传真讯号,更具备独特之防传真纸卷曲功能。FAX-600 还具文字平滑处理功能及多种解像传送,令文字及图象传真清晰过人。

424

美国容错电脑有限公司

广州办事处开业典礼及新产品

发布会

Stratus 美国容错电脑有限公司广州办事处开业典礼及新产品发布会于一九九三年十一月十八日假座广州市花园酒店三楼紫荆厅举行。参加的嘉宾来自金融业、邮电通讯、交通运输等行业共有一百多人。

美国容错电脑有限公司成立于 1980 年,总部位于美国麻省万宝路

市。该公司是世界第一家采用硬件容错技术,制造出新一代零故障的容错计算机,其独特的硬件容错体系结构设计,为用户提供分秒不停连续操作。世界上采用容错计算机的用户多属要求高可靠性的行业,如银行业、通信业、交通运输管理、国防警报系统,及生产企业管理系统等。该公司 1992 年营业额为四亿八千六百万美元,比 1991 年增长百份之八。1993 年至第三季度止,营业额为三亿六千五百五十万美元,比去年同期增长百份之四。亚洲及远东地区增长率则为百份之十一。

美国容错电脑公司的销售网除遍布美国本土外,在加拿大、英国、法国、意大利、西班牙、德国、比利时、荷兰、瑞典、瑞士、香港、新加坡、日本和澳大利亚等国家及地区成立了 50 多家分公司。自 1990 年始,分别在北京及上海设有代表办事处。

随着中国的开放政策的实施,中国对计算机技术的引进迅速增长,美国容错电脑有限公司作为一家拥有最先进行容错技术公司,主动将其先进的技术介绍给中国用户。过去数年来,在中国部分用户包括:中国银行、中国工商银行、中国建设银行、交通银行、哈合滨财政证券公司、中国国际贸易促进会、北京无线通信局、中国国家旅游局信息中心、上海航空公司、昆明金龙饭店、南京饭店、上海银河宾馆、蛇口集装码头有限公司、广东省邮电储汇局、广东省移动通讯总公司、126 广东省寻呼联网、上海通讯开发总公司、深圳市邮政局、天津信托投资公司……等。

美国容错电脑有限公司继 1993 年 6 月在中国推出 XA/R 精简指令集容错计算机为平台,运行容错 UNIX 操作系统 FTX。产品具有更高的性能价格比。将通过代理商销售。

XA/R 服务器系列包括下列产品:(型号的一S 代表服务器)

初级系统:XA/R5-S. XA/R10-S. XA/R15-S

中档系统:XA/R25-S. XA/R35-S. XA/R45-S. XA/R55-S

索取资料请与美国容错电脑有限公司广州办事处联系。

425

新一代彩色笔记本电脑上市

一种高性能、高显示质量的新一代笔记本电脑——VERIDATA XP486/33V。已由北京友邦电子技术有限公司和台湾英加电子工业股份有限公司联合推出,已在国内上市。

台湾英加电子工业股份有限公司是台湾专业生产笔记本电脑的厂商,年产量 20 万台,其产品主要行销欧美市场。

VERIDATA XP486/33V 笔记本电脑, CPU 采用 33MHz 486 芯片, VIDEO RAM 容量为 512K。可有三种分辨率,分别是 640×480 256 色、800×600 16 色、1024×768 256 色。采用国际上最先进的 Dual Scan 显示方式,它是 LCD 最新一代产品,显示效果远远优于目前市场上流行的第二代产品 STM 彩色 LCD。由于采用双扫描方式,灯管由 1 个增加到 2 个,故 LCD 的亮度和对比度明显增强。特别是视角由 ±20° 提高到 ±60°,极大地扩大了用户的可视范围。LCD 屏幕由 9 英寸扩大到 10 英寸。该笔记本电脑内存容量为 4M 字节,软盘 1.44M 字节,硬盘 120M 字节。另提供内置 F/M。跟踪球和独创的 F/M 跟踪球选项。

(钱承德)

426

在程序中直接读取CMOS信息

湖南双峰工商银行 罗 辉 陈又来

目前在 286 以上的档次较高的微机，都配备有一个用于保存系统的硬件配置参数和系统实时时钟信息的 CMOS 电路。它由内部电池供电，保证在关机后系统信息不会丢失。一般情况，CMOS 上的信息通过正确设置后就不需要再去管它。因而用户在系统没有出现故障 CMOS 的故障时，一般没去感觉它的存在。但 CMOS 上的信息对系统来说却是相当重要的，没有它系统就不能正常运行或根本不能启动。充分掌握这部分信息，对我们进行系统维护或特殊的程序设计有一定的帮助。譬如：可从这部分信息掌握具体机器的基本配置情况，控制你的程序不同的流向；或者为你的软件在此设置安装或版权标志。据称美国曾有几种病毒，就将它的感染标志或特征码设置在 CMOS 上，为一般检测程序所不及。

CMOS 上的信息实际保存在其内部芯片 MC146818 上。其有两个端口：地址口 70H，数据口 71H。信息共有 64 字节长。各信息的地址号及其意义如下：

地址号	内容
00	秒
01	秒报警
02	分
03	分报警
04	时
05	时报警
06	星期
07	日
08	月
09	年
10~13	状态寄存器 A~D
14	诊断状态，没有故障为 0
15	系统电源状态，有市电时为 0
16	A~B 软驱类型。高四位对应 A，低四位对应 B 其值：1—360 KB；2—1.2 MB；6—1.44 MB；7—720 KB
17	保留
18	C~D 硬驱的类型。高四位对应 C，低四位对应 D 其值结合 19 号地址口的值决定硬盘的类型
19	硬驱类型辅助字节。
20	设备装备状态
21~22	基本内存大小的低、高八位。对 640K 内存其值为 80H、02H（即十进制 640）。内存大小以 K 为单位
23~24	扩展内存大小的低、高八位。对 1M 扩展内存其值为 00H、04H（即十进制 1024）。
25	C 驱类型号，如安装硬盘类型号为 22，其值为 16H
26	D 驱类型号

27~45	保留
46~47	CMOS 中部分重要信息字节的校验和。如果编程单独修改了某些重要字节内容，而没有相应改变该校验和，下次系统启动时会出现校验和错。但有些字节不参与该校验和运算，如日期、时间、报警等信息字节。因而可单独修改这些字节而不出现校验和错。
48~49	同 23~24 字节，扩展内存大小
50	世纪值。如 19 世纪其值为 19H
51	信息标志
52	CMOS 密码设置标志，为 0 表没有密码；否则有密码
53~63	保留。设置的 CMOS 密码常保存在此。但不同的生产厂家保存密码的方式不同，如 AST 机器仅保存一字节密码校验和及密码长度；而 LAND 机则逐字节保存密码的加密密文

其中，上面注明“保留”的字节，用于其他系统高级特性参数的保存。不同的机器这部分字节保存的信息意义不同。

在程序中读写 CMOS 信息相当简单：在读写这些信息前，先向地址口发送地址号，以指名下一步在数据口读写的信息是哪个地址号上的数据。

下面用 TURBO PASCAL 编制一个显示 CMOS 信息、并设置系统报警时间的示例程序，以供参考。

```

program readcmos;
uses printer;
var
    count: byte;
    value: byte;
begin
    port($70) := 1; {设置系统报警时间为每天十点半}
    port($71) := 0;
    port($70) := 3;
    port($71) := 30;
    port($70) := 5;
    port($71) := 10;
    count := 0;
    writeln('ADDRESS VALUE: ', '3', 'ADDRESS VALUE');
    while count < 64 do {读 64 字节信息，十进制显示}
    begin
        port($70) := count;
        value := port($71);
        write(count:7, value:11, '3');
        count := count + 1;
        port($70) := count;
        value := port($71);
        writeln(count:7, value:11);
        count := count + 1;
    end;
end.

```

PC 机驱动器清洗工具 DriverClean

湖南省人民警察学校微机室 刘永强

《电脑》92年第6期上介绍了一个苹果机驱动器清洗辅助程序,其原理颇有可取之处。PC机驱动器一般可以“插盘自转”,把清洗盘插入后,关好小门,驱动器即会自动旋转几秒钟,不过时间均不长,而且磁头不作纵向移动,造成清洗盘在0道位置上形成明显的脏痕,影响了清洗盘的寿命。为此,经反复改进,本文提供一个较为成熟的磁头清洗辅助程序 DriverClean v1.2。该程序用汇编语言实现。

一、原理

笔者经试验,DOS的磁盘服务中断13H中,读、写等操作均不能移动磁头寻道(插入清洗盘),故采用13H中断中的5号格式化功能,交替格式化0道及最大道,从而达到快速移动磁头的目的,充分利用清洗盘的各有效部位,延长清洗盘的使用寿命。

注:本程序不能作格式化程序使用。

1.2版程序能自动识别40道的360K驱动器以及80道的1.2M和1.44M驱动器,从而调整最大道号,避免了道号过大导致驱动器发出嘎嘎噪音的现象。这一功能设在程序的40句至50句,它利用了13H中断中的一个功能,用法如下:

入口: AH=08H 读驱动器参数功能
DL=驱动器号(0=A,1=B)
INT 13H

出口: BL=驱动器类型
CX=最大道号及最大扇区数
DH=最大磁头号
ES:DI=11字节参数表的地址
如出错则进位标志设立

注:

驱动器类型为:

01=360K 02=1.2M
03=720K 04=1.44M

此功能仅由XT、AT及PS/2机提供。如果此功能调用出错的话,说明该机不支持高密驱动器。

1.2版程序还能够识别插入指定驱动器的是否是清洗盘,如果不是则会报警,以免把有用的无写保护的软盘格式化,造成不可挽回的损失。不过如果软盘有写保护的话,用户可强制进行。本功能是通过磁头0道进

行读操作来实现的,如果能正常读写,那么肯定不是清洗盘。

程序中104句到105句是为了避免下面的情形:

如果在1.2M驱动器中插入了360K软盘,在调用本程序进行清洗时,DOS系统自动设置驱动器参数,驱动器的最大道只能是第39道了,即使换用清洗盘也不会改变,这时进行清洗就会出现嘎嘎噪音。

上述问题的发生,实际上是DOS设置了在1.2M驱动器上进行360K格式化,我们只须重新设置为1.2M格式化即可。该功能用法如下:

入口: AH=17H (为格式化设置DASD类型)
AL=DASD类型
DL=驱动器号
INT 13H

出口: AH=状态
如出错则进位标志设立

注:

DASD = Direct Access Storage Device
DASD类型有:

01=360K 驱动器格式化 320K / 360K 软盘
02=1.2M 驱动器格式化 360K 软盘
03=1.2M 驱动器格式化 1.2M 软盘
04=720K 驱动器格式化 720K 软盘
其它值非法。

二、使用

读者可以先用文本编辑软件建立本程序的汇编语言源文件DRCLEAN.ASM,再如下操作:

C>MASM DRCLEAN;用宏汇编进行编译
C>LINK DRCLEAN;用连接程序连接
C>EXE2BIN DRCLEAN DRCLEAN.COM 转换为COM

C>DEL DRCLEAN.EXE 删除原EXE文件
C>DRCLEAN 盘符使用DriverClean

盘符必须给出,可以为A或B,分别指定A、B驱动器,大小写不限,不过命令与盘符之间只能有一个字符(空格或后斜杠)。

程序启动后,用户可敲入以下键选择:
回车键——自动操作 (自动移动磁头)

空格键——手动操作 (手工移动磁头)

Esc 键——退出

选择手动操作时,磁头初始位置在最大道,用户在清洗过程中可以使用数字小键盘上的+、-两键移动磁头。

如果程序检测出驱动器中插入的盘不是清洗盘时,用户可敲入以下键选择:

回车键——更换清洗盘再进行清洗

空格键——强制清洗

Esc 键——退出

清洗开始后,用户须按回车键才可中止清洗并退出程序。

本程序在 PC 机、XT 机、AT 机上均能正常运行, DOS 要求 2.0 以上即可。

```
;DriverClean V1.3 source file
;serial # 9307A01C
;Liu YongQiang (Strong)
;
K_ENTER equ 13
K_SPACE equ 32
K_ESC equ 27
DISP macro string
call STRdisp
dw offset string
endm
code segment 'code'
assume cs:code,ds:code
org 100h
start: xor ax,ax
int 13h
mov ax,cs
mov ds,ax
mov es,ax
DISP TITmes
mov si,80h
lodsb
or al,al
jnz prmt
jmp errExit
prmt: inc si
and al,0dfh ; upper char
sub al,'A'
mov DRnum,al
or al,al
jz keyin
cmp al,1
jz keyin
jmp errExit
keyin: mov dl,DRnum ; driver #
mov al,'A'
add al,dl
mov DRV,al
mov DRV2,al
```

```
mov ah,8 ; Read driver
int 13h ; parameters
jc HDno
mov MAXtrk,ch
mov byte ptr HDdrv,1 ;assume drv is HD
cmp bl,1 ; BL=drv type
ja HDyes
HDno: mov ax,2700h
mov MAXtrk,ah
mov HDdrv,al ;HDdr V=0: not a HD
HDyes: DISP STRmes
keyin1: xor ah,ah ; read char
int 16h
mov Aop,al
cmp al,K_ENTER
jz clean
cmp al,K_SPACE
jz clean
cmp al,K_ESC
jnz keyin1
jmp Exit
clean: mov cx,3 ; 3 times retry
mov bx,800h ; buffer ES:BX
mov dl,DRnum ; driver #
xor dh,dh ; head 0
VRF: push cx
xor ch,ch ; track 0
mov cl,1 ; sector start
mov ax,0206h ; Read sect to
int 13h ; check disk
jnc noit
xor ax,ax
int 13h
pop cx
loop VRF
jmp doit
noit: DISP WANmes
keyin2: xor ah,ah
int 16h ; read char
cmp al,K_ENTER
jz clean
cmp al,K_SPACE
jz doit
cmp al,K_ESC
jnz keyin2
jmp Exit
doit: cmp Aop,K_SPACE ;IS manual operation?
jnz doit2 ;no
DISP MNLmes
doit2: cmp byte ptr HDdrv,0 ;If HD driver?
jz doit3 ; no
mov ax,1703h ; Set DASD type 3 for
int 13h ;format (1.2M in 1.2M)
doit3: mov dl,DRnum ; driver #
xor dh,dh ;head 0
mov ch,MAXtrk
mov cl,1 ; sector start
```

```

dr_cl:  push dx
        push cx
        call trkNUM
        DISP STPmes
        pop cx
        pop dx
        mov ax,0600h ;format disk
        int 13h
        mov ah,1      ;If key press?
        int 16h      ;
        js TRKchg     ;no
        xor ah,ah
        int 16h      ; read char
        cmp al,K_ESC
        js ExitThis   ;yes
        cmp Aop,K_ENTER ; Is AUTO ?
        js TRKchg     ; yes
        cmp al,'+'
        jnz TRKdec
        inc ch
        jmp trkjp
TRKdec: cmp al,'-'
        jnz TRKchg
        dec ch
trkjp:  cmp ch,MAXtrk ; out of range?
        jbe TRKchg    ; no
        mov ch,MAXtrk
TRKchg: cmp Aop,K_ENTER ; Is AUTO ?
        jnz dr_cl     ; no
        xor ch,MAXtrk
        jmp dr_cl
ExitThis: DISP BLANK
        DIPS ANOMes
ETJ00:  xor ah,ah
        int 16h
        mov Aop,al
        DISP BLANK
        mov al,Aop
        and al,0dfh ;upper char
        cmp al,'Y'
        jnz Exit
        mov al,1
        xor DRnum,al ;chang driver
        jmp keyin;clean again
Exit:   DISP THKmes
        mov ah,4ch
        int 21h
errExit: DISP ERRmes
        jmp Exit
STRdisp: pop bx
        mov dx,[bx] ; string offset
        inc bx
        inc bx
        mov ah,9      ; display string
        int 21h
        jmp bx        ; return
trkNUM: mov al,ch
        xor ah,ah
        mov bl,10
        div bl
        add ax,3030h
        mov trk,ax
        ret
;
DRnum  db ?
MAXtrk db ?
HDdrv  db 1
Aop     db ? 1ah
TITmes db 13,10
        db 'LYQ-Strong DriverClean version 1.3,13,10'
        db '(C)Copyright Strong Organization ,1993'
        db 13,10,10,24h
WANmes db 7
        db '-----Warning-----',13,10
        db 'The diskette in driver '
DRV2   db 'X: is not a',13,10
        db 'Disk-cleaner .Do not insert the useful'
        db 13,10
        db 'disk into the specified driver ,because'
        db 13,10
        db 'it will be formatted , and the data will'
        db 13,10
        db 'lost ! ! ! ',13,10
        db '-----',13,10,10
        db 'Change disk and press ENTER to continue'
        db 13,10
        db 'Press SPACE to continue without change'
        db 13,10
        db 'Press ESC to quit',13,10,10,24h
STRmes db 'Insert Disk-Cleaner in driver '
DRV    db 'X:',13,10
        db 'Press ENTER to Auto-operation'
        db 13,10
        db 'Press SPACE to Manual-operation'
        db 13,10
        db 'Press ESC to quit' 13,10,10,24h
STPmes db 'cleaning track '
trk     dw ?
        db ', press ESC to stop',13,24h
MNLmes db 'Press +&- to change track#'
        db 13,10,10,24h
BLANK  db 13,60,dup (32),24h
ANOMes db 13,clean another driver (Y/N)? ',24h
THKmes db 13,'Thank you for using Drvierclean'
        db 13,10,10,24h
ERRmes db 7,'Not sepcify driver# or had command'
        db 13,10,10,24h
code ends
        end start

```

用 Tango 和 AutoCAD 联合进行电原理图设计

深圳康佳电子(集团)股份有限公司 蒋肖南

在电原理图设计方面,目前电子行业使用较多的是 AutoCAD 以及 Tango、OrCAD 等软件,它们在使用时各有利弊。Tango 是电子电路设计的专用工具软件,它们在存贮位置坐标时一般采用整数,因而在显示时明显地要快得多。但它出图的质量也存在着一定的缺陷:如绘出的圆不太圆滑、写的字符字型较难看等等。与之相反,AutoCAD 作为计算机辅助绘图的通用软件包,在制图时具有很强的功能和很大的灵活性,且出图质量好。但其存贮位置坐标采用的是浮点数,大量的浮点数的运算使得显示速度很慢,尤其当图形较为复杂时,其重显或重新生成的速度更相当缓慢,这样就极大地降低了其使用效率。此外,其功能方面也很难做到 Tango 一样方便。

那么,能否利用两者的长处,扬长避短,把两者结合起来进行电原理图联合设计呢?基于这一想法,我们把 Tango 和 AutoCAD 结合起来进行电原理图的设计。先由 Tango 输入电原理图的元件、连线、连接点和自由字符串等,接着通过一个转换程序把其转换成 AutoCAD 的格式,最后在 AutoCAD 环境下,使用绘图仪进行出图。

设计这样的一个系统,主要要进行以下几部分工作:

1、必须了解 Tango 存贮电路图的格式和 AutoCAD 输入图形的命令文件的格式

在 Tango 中,电原理图是用以.S01、.S02...等为后缀的二进制方式文件存贮的,其详细的存贮格式读者可参阅胡万海编《新编多层自动布线印制版的设计与实现:TANGO V1.13-3.12 版》的第八章。

Tango 存贮电原理图时,是以节(16 字节)为单位的,一条线段、一个连接点各占一节,每个字符串占两节,每个元件占若干节,视乎元件的管脚数目和点阵大小,其中前四节为元件的总体信息。

线段记录存有起点、终点的位置和线型等;

连接点记录存有其位置等内容;

字符串记录存有其起始点位置、字符尺寸、放置方向、字符串内容等;

元件记录存有其插入点位置、放置方向、点阵内容以及元件位号和元件值的位置、尺寸方向和内容等。

2、建立相互对应的 Tango 和 AutoCAD 各自的元件库

由于 Tango 和 AutoCAD 分别是两种不同的软件,前者的元件库是点阵库,而后者的元件库是矢量库。为了能产生所需的电原理图,需要分别建立其各自的元件库。

为了便于图形的转换,在建立元件库时,确立如下元件的对应关系:

(1) 元件具有相同的插入点。Tango 中元件插入点在其点阵的左上角,沿 X 轴镜像后在其点阵的右上角。建立 AutoCAD 的元件库时,选择相同的点作为插入点。

(2) Tango 中一个元件对应 AutoCAD 中两个元件,分别表示元件本身及其沿 X 轴的镜像图形。这主要是因为 AutoCAD 中元件的镜像会引起其位号等随之镜像而反转,与所需不符。

3、编制 Tango 向 AutoCAD 的转换程序

在 AutoCAD 接收转换的电路图时,我们采用其命令文件的格式。把 Tango 的电原理图转换成 AutoCAD 的以.SCR 为后缀的命令文件,再用 SCRIPT 命令把它读入 AutoCAD,形成所需的电路图。

根据以上的叙述,就可以编制一个程序,来实现两者的转换。

线段根据其线型不同,分别用 AutoCAD 中不同线型的 LINE 或一定宽度的 PLINE 画出;

连接点通过插入特定的块构成;

字符串用 TEXT 命令形成;

元件则调用相应的 AutoCAD 的元件库。把元件中的点阵和 Tango 元件库中的点阵进行比较,如果相同,则确定的调用 AutoCAD 的元件库的文件名。元件位号和元件值则分别送入相应 AutoCAD 元件的属性中。

完成转换后,可在 AutoCAD 环境下,使用绘图仪进行出图。如果需要修改电原理图,可以在 Tango 中进行,之后同样转换到 AutoCAD 中。

利用本系统解决了采用 Tango 或 AutoCAD 进行电原理图设计时的效率和出图质量之间的矛盾,因而具有很强的实用性。

2130A 矢量字库的压缩与恢复

上海化工进出口公司

虞培发

随着计算机技术的飞速发展,汉字输出技术不断提高,与之配合,汉字存贮也由原来的点阵存贮向矢量化发展,美观漂亮的矢量汉字成为大多数人的追求,并成为一门新兴技术。由于矢量汉字库所占存贮空间较大,我们实际见到的都是已压缩过的字库,这给我们自己使用矢量字库增加了障碍。本文以 2130A 系统为例,介绍一下该字库的压缩存贮与恢复。

一、矢量字库的存贮、显示

通常,我们接触的中文操作系统,其汉字是以固定数目字节的点阵存贮的,一个 16×16 点阵的汉字为 32 字节, 24×24 点阵汉字为 72 字节。而一个矢量汉字则是以数目变化为 (x, y) 矢量集作为汉字显示的依据(其中 (x, y) 为计算机屏幕的坐标)。以图(1)为例,字“”将以 $(x_1, y_1), (x_2, y_2), \dots, (x_9, y_9)$ 矢量为显示的“字模”,每个 (x, y) 矢量由二个字节组成。

因此,一个矢量汉字字库,其结构将如图(2)所示。其中,区一为索引区,区二为矢量区。区一中,每三个字 (w_1, w_2, w_3) 代表一个汉字,其中第一个字 w_1 和第二个字 w_2 作为地址(高字,低字)指向区 2 中该字的矢量集,第三个字则为该字在区二中的矢量个数,如图 1 中,第三个字应为 $9 \times 2, 18$ 。

由此,一个矢量汉字的显示过程大致为:

1. 将汉字内码转换为 16 进制表示的区码和位码。
2. 将 $(\text{区码} \times 94 + \text{位码}) \times 6$, 定位该汉字在区一中的位置,并读 3 个字。
3. 根据 2 中得到的 3 个字,在区二中找到该字,并读该汉字的矢量集。
4. 将 3 得到的矢量集进行复原、放大、填充等操作。
5. 显示该汉字。

二、2130A 矢量汉字库的压缩存贮

如上文所说,矢量汉字库是以压缩方式进行存贮的,其优点不言而喻:1. 节约了存贮空间,2. 一定程度上起到了加密作用。然而,其缺点则增加了操作,也使我

们调用其字库增加了困难。笔者通过跟踪 2130A 系统,试解其压缩字库如下:

(一) 字库名称

HZKSLSTJ 汉字库矢量宋体简

HZKSLFSJ 汉字库矢量仿宋简

HZKSLHTJ 汉字库矢量黑体简

HZKSLKTJ 汉字库矢量楷体简

(二)、矢量字库压缩、加密原理:

通过索引区,取得某汉字矢量集后,各字节含义如下:

(1) 字节高二位为 11, 意为笔划首字节,将该字节低 6 位,及下一字节一个高位共七位作为坐标 x_0 , 将下一个字节低 7 位作为坐标 y_0 , (x_0, y_0) 一起作为笔划起点(其中 C 为符号位, 0 为 E, 1 为负)

(2) 字节高二位为 00, 则该字节低 6 位存有计数值, 设为 cx , 其后有 cx 个字节, 每个字节高 4 位为 x' , 低 4 位为 y' , x', y' 为 x_0, y_0 的偏移量, (x_0+x', y_0+y') 为笔划坐标。

(3) 字节高二位为 01, 则该字节低四位为计数值, 记为 cx , 其后 cx 个字节, 每个字节高 4 位为 x' , 低 4 位为 y' , x', y' 为 x_0, y_0 偏移量, 并根据

BP=00 $+x', +y'$ 坐标 (x_0+x', y_0+y')

01 $+x', +y'$ 坐标 (x_0+x', y_0+y')

10 $-x', -y'$ 坐标 (x_0-x', y_0-y')

11 $-x', -y'$ 坐标 (x_0-x', y_0-y')

得到其矢量坐标。

(4) 字节高二位为 10, 则随后二位设为 BP。

当 BP=00, 则以该字节低 4 位为 x' , 以下一字节为 y' , 其坐标为 (x_0+x', y_0+y') ;

= 01, 则以该字节低 4 位为 y' , 以下一字节为 x' , 其坐标为 (x_0+x', y_0+y') ;

= 1x, 则以随后二字节分别作为 x', y' , 其坐标为 (x_0+x', y_0+y') 。

根据以上原理,我编制程序,试解 2130A 压缩字库,完全成功,特作文字以记之,与同行商讨。

