

# 微小型计算机 开发与应用



4

1987

微小型计算机开发与应用编辑部

# 欢 迎 订 阅

## 《微小型计算机开发与应用》

《微小型计算机开发与应用》是1981年经电子部计算机工业管理局批准编辑出版、向全国公开发行的中级技术刊物。

重点报导微型、小型计算机开发及应用中的经验与成果，及时介绍国内外微、小型计算机领域的最新产品、技术和发展动态。有30余名计算机专家经常为本刊撰写各类文章。

《微小型计算机开发与应用》辟有“计算机系统”，“计算机软件”，“微小型计算机应用实例”，“网络设计”，“典型程序介绍”，“经验点滴”，“市场分析”和“动态与信息”等栏目。为满足不同层次读者的需要，还将陆续开辟“专题连载”，“初学者之友”等新栏目。

《微小型计算机开发与应用》适于计算机界广大科技人员，企管人员，教学人员，大专院校学生及各行业的计算机用户阅读。

本刊为双月刊，逢单月末出版，16开本，48页，每期定价0.60元。由天津市邮局向全国发行，全国各地邮局均可订阅。代号：6—87

本刊编辑部地址：天津市河西区友谊路爱民道5号

---

**编辑：**《微小型计算机开发与应用》编辑部  
**出版：**天津市电子计算机研究所  
**地址：**天津市河西区友谊路爱民道5号

**发行：**天津市邮局  
**印刷：**天津市静一胶印厂  
**定购处：**全国各地邮局  
**定价：**0.60

# 目 录

1987年(双月刊)

第4期(总第30期)

捷克斯洛伐克的微处理机.....刘家松(1)

## 计算机系统

Sun-3工程工作站的系统结构(续1).....(6)

## 微小型计算机应用实例

港口外贸费收管理系统的分析与设计.....(12)

产品结构优化系统.....刘辉 吴振庆 刘於勋等(17)

计算机在收购粮食工作中的应用.....钱维勤 赵敦玉 钟淑敏(21)

计算机在能源审计中的应用.....张更明(24)

## 计算机软件

DBASE I与高级语言关系的探讨.....林 坚 于京平(28)

微机应用中被删除文件的恢复.....崔秀起(35)

## 计算机网络

汉字AST-PCNET局域网.....刘京松(37)

## 主机与部件

可编程控制器S5-115U的CPU硬件结构.....金广业(41)

征订通知.....(封四)

动态与信息.....(46)

新书介绍.....(47)

## 征稿启事

.....(27)



(上接第36页)

在图2右端项目中可以查到被删文件CUI02. ASM被改成了eUI02 ASM,在这行的左端列出了被删标记e(ASCII码是E5)所在的内存地址是08FF:0BC0。

—E 08FF 0BC0 43; 把有删除标记 E 5  
的地址内容修改为  
43(C的ASCII码)

—W 100 0 0 8 ; 把修改过的内容存  
盘

—Q ; 退出DEBUG, 返  
回DOS。

A> copy CUI02.ASM B; ; 将恢复的文件拷  
贝到B盘。

这时B盘上就得到了已恢复的被删文件。

## 四、几点说明

1. 以上实例是在PC兼容机 ABACUS 上做的,这种方法在DOS操作系统下在所有IBM-PC微机及其兼容机都适用。

2. 虽然CROMEMCO微型机的CDOS对磁盘盘区格式化不同,盘区分配不同,扇区是交错排序,目录项各字节功能安排也有差异,但每个目录项同样为32个字节,同样首字节为E5H表示文件被删除。CDOS通过位图对文件进行管理,以位图的每一位代表盘空间一个簇的使用情况,用“1”表示该簇已被占用,用“0”表示此簇空闲。被删除文件分配的簇改用“0”标注,这一点和PC机原理是一样的。因此本方法完全适用于CROMEMCO微型机。

3. 必须注意,本方法只能在被删文件的名字和内容未被新文件覆盖以前有效,一旦被新文件覆盖,那末删除的文件不能再恢复。

### 参考资料:

1. 周明德,《微型计算机IBM-PC(0520)系统原理及应用》,清华大学出版社
2. 杨长生,“IBM-PC软盘文件的恢复方法”,《小型微型计算机系统》86.1
3. 《CROMEMCO微型计算机软件资料汇编》,清华大学出版社。

# 捷克斯洛伐克的微处理机

刘家松

(天津大学计算机系)

**摘要** 本文主要介绍捷克斯洛伐克社会主义共和国微处理机及有关电路的发展与生产情况。在苏联和东欧社会主义国家中,微处理机的开发与生产是互相合作、彼此分工的,因而本文也涉及苏联微处理机的一些情况。编写本文主要参考了捷克“自动化”1984年第5期刊载的“捷克斯洛伐克微处理机的发展”一文,也参考了捷克“通信技术”、“弱电评论”和斯洛伐克“电工技术期刊”等刊物近三、四年的有关文章与资料。

## 一、概述

近二十年来所有工业发达国家的技术进步都是以电子学发展为条件的。没有一个工业部门不利用集成电路(主要是微处理机;存储器和其它有关器件)的微电子学先进性来实现革新、生产过程的合理化和提高效率。

微处理机是一种通用集成电路,其功能可用程序来规定和改变。它是1969年计划开发的,1971年研制出来第1个4位微处理机,Intel公司的I4004;不久又研制出第1个8位微处理机I8008;随后又研制出最有名的七十年代应用最广泛的微处理机I8080。由于集成度不断提高,1976年国外出现了一块芯片上包括整个微型计算机的产品,即单片微型计算机。它除微处理机外,还包括时钟电路, RAM和ROM或EPROM存储器, I/O电路和其它支持电路。

由于Intel、Motorola、Zilog等公司和其它三十个生产厂的发展带来惊人的成就。出售的整个系列的单片微型计算机,不仅有4位和8位的,而且有16位的。微处理机除4位、8位、12位和16位的各种产品外,还出现了多种32位产品。

集成电路的迅速发展和半导体工艺的进步是分不开的。这种工艺持续发展使结构缩小,集成密度提高,信号处理速度提高和功耗降低。工艺发展可近似地用芯片上器件数目的增长来表征:

1976年-8位微处理机-9 000 器件/片

1978年-16位微处理机-50 000器件/片

1982年-32位微处理机-500 000器件/片

## 二、微处理机在捷克斯洛伐克的发展

国营企业TESLA Rožnov、TESLA Piešťany\*和TESLA通信技术研究(VÚST)于1977至1980年建立工艺工场,为单极性和双极性工艺的进一步发展和在芯片上集成几千个器件的集成电路的研制与生产打下基础。国家任务RVT中规定了单极性和双极性工艺的发展。

在单极性工艺方面,1980年开始研制NMOS工艺的大规模集成电路(微处理机和有关电路MHB 8080A、MHB8251、MHB8255A和存储器MHB 2102A、MHB 8708、MHB 4116)和CMOS工艺的系列4000/4500(与RCA的产品相当)中小规模集成电路。从1980年起也用CMOS工艺开发了几种大规模集成电路:MHB1902A、MHB8804——带译码器的开关矩阵 $8 \times 4$ 、MHB 5085——频率选择发送器等。

微处理机和所有CMOS电路(22个型号)的设计都是与TESLA VÚST和CMOS工艺的开发一起解决的,NMOS工艺的开发是在TESLA Piešťany进行的。

在双极性工艺方面, TESLA Rožnov开发

---

\* TESLA是捷克斯洛伐克国营电工电子企业的名称。TESLA本是一位南斯拉夫著名科学家的名字,他曾在捷克工作过。Rožnov和Piešťany都是城市的名称。

了数字和模拟电路用的schottky TTL和 $I^2L$ 工艺。1980年开始开发以schottky TTL为基础的位片式微处理机系统MH3000并逐步投产。该系统由八个型号的数字电路组成,由这些电路构成微程序设计的中央处理机和控制器。由标准schottky存储器ROM和PROM实现的微程序控制存储器的内容,确定了上述中央处理机和控制器的控制功能。MH3000系列最重要的大规模集成电路(LSI)是微程序控制单元MH3001和中央处理单元MH3002。微程序控制单元用来编制微指令、执行进程并控制到中央处理单元的数据传送/位移。中央处理单元MH3002是中央处理装置的两位宽的处理位片,它并行地排列,可以构成任意字长的处理机。MH3000系统中还包括下列电路:MH3003、MH3205、MH3212、MH3214、MH3216和MH3226(见表1)。对构成微程序用存储器,可使用所有的预制型号快速双极性存储器,有时也可利用静态或动态MOS存储器RAM、ROM、PROM或EPROM的某些产品。TESLA Rožnov 1984年底将MH3000系统中的存储器品种扩展到容量1K位的RAM和2K位的ROM和PROM,还准备开发容量4K位的ROM和PROM和8K位的PROM。七五计划(1981~1985)还开发了采用 $I^2L$ 和 $I^3L$ 工艺面向用户的电路。1984年投产了线性与数字功能混合集成的新集成电路类型,还有构成8位D/A变换器用的电路。

1982年开始生产单极性工艺的8位微处理机系统的三种电路:MHB8080A(CPU)、MHB8251(USART)和MHB8255A(PPI),均采用NMOS工艺。该系统还补充了几种已在双极性微处理机系统中使用的电路MH8205、MH8212、MH8214、MH8216、MH8226以及另外的双极性电路MH8224时钟和MH8228系统控制电路和总线控制器(见表1)。

上面介绍的微处理机电路型号都和美国Intel公司的相应电路型号对应。捷克斯洛伐克微处理机的发展面向Intel的电路,不论是单极性的还是双极性的,主要追求和原型号电路在功能上兼容。1978/79年制订国家任务RVT时选择微处理机电路典型,受下列事实影响:

——当时Intel公司是世界范围内双极性微处理机I3000系列和8位微处理机8080系列的最大生产者;

——I8080电路具有最广泛和最受欢迎的软件;

——8080电路当时是捷克斯洛伐克应用最多的微处理机电路,而且对它的需求正在增长;

——为了开发8080电路的软件,已从Intel公司引进Intel公司的全套开发系统,而且微型计算机企业(FMEP)已准备生产类似的开发系统;

——苏联的一些微处理机生产厂也确定了上述电路型号。

面向Intel的微处理机电路型号,是捷克斯洛伐克微处理机发展的一个主要特点。另一个主要特点是和经互会国家特别是苏联合作。由于捷克斯洛伐克半导体工业不能开发全部所需的微处理机电路型号(品种),过去、现在和将来都面向合作,互通有无。这样保证国民经济许多部门能够使用现代微处理机的各种电路。

捷克斯洛伐克开发并从1981年逐步投产了一些存储器。有两种型号的1K位静态存储器RAM,2102A采用NMOS工艺,1902A采用CMOS工艺,两者在速度和功耗方面有区别。还生产了16K位动态存储器RAM(相当于4116)和8K位可用紫外线擦去的可编程存储器EPROM(相当于8708)。

1981年根据对世界发展趋势的估计和对捷克斯洛伐克企业TESLA VÚST和TESLA Rožnov的基本情况和前景的估计,将“单片微型计算机系统”的任务编入国家1982至1985年技术发展计划。该任务的目标是将选定型号的单片微型计算机和大容量存储器投产并保证其应用的基本条件。

上述任务中规定要在1982至1984年间将下列电路研制出来并投产。

**微型计算机:**由TESLA VÚST设计,国营企业TESLA Piešťany承担工艺;

8048——带存储器RAM  $64 \times 8$ 位、ROM 1K  $\times$  8位的单片8位微型计算机、机器周期2.5微秒,27条输入/输出线、电源5V、功耗1.5W,双列直插式封装——40条引线,振荡器频率5兆赫;

8035——同上但没有存储器ROM(EPROM);

8748——单片8位微型计算机,与8048相同,只是将1K  $\times$  8位ROM改为EPROM,1985年投产;

8243——扩展器,能将输入/输出数目扩大16个,即( $4 \times 4V/V$ );

**存储器:**

表1 捷克斯洛伐克的七五计划中生产和将要生产的集成电路

电路型号	名 称	生产年代
MHB8080A	8位中央处理机	生产中
MHB8255A	PPI——可编程序的外部设备接口	生产中
MHB8251	USART——通用同步和异步发送器—接收器（通信接口）	生产中
MH3212 8212	输入/输出电路	生产中
MH3214 8214	优先级中断电路	生产中
MH3216 8216	4位不倒相总线驱动器	生产中
MH8224	CPU8080A用的时钟源	生产中
MH3226 8226	4位倒相总线驱动器	生产中
MH8228	CPU8080A用的系统控制电路和总线驱动器	生产中
MH3205 8205	8中取1译码器	生产中
MH3001	微程序控制电路	生产中
MH3002	两位处理单元（ALU）	生产中
MH3003	传输（进位）加速	生产中
MHB8048	单片8位微型计算机	1984
MHB8748	带存储器EPROM的单片8位微型计算机	1985
MHB8035	无存储器ROM的单片8位微型计算机	1984
MHB8243	输入/输出扩展器	1984
MHB2102A	1千位（1k×1）静态存储器RAM	生产中
MHB1902A	1千位（1k×1）静态存储器RAM	生产中
MHB6561	1千位（256×4）静态存储器RAM	1984
MHB2114	4千位（1k×4）静态存储器RAM	生产中
MHB8708	8千位（1k×8）存储器EPROM	生产中
MHB2716	16千位（2048×8）存储器EPROM	1984/85
MHB4116	16千位（16k×1）动态存储器RAM	生产中
MHB8155	2048位（8×256）静态存储器RAM+输入/输出扩展器	1986
MHB40114	64位（16×4）存储器RAM	1985
MH74188	256位（32×8）存储器PROM	生产中
MH74S287	1024位（256×4）存储器PROM	生产中
MH74S571	2048位（512×4）存储器PROM	生产中
MH74S370	2048位（512×4）存储器ROM	生产中
MH74S187	1024位（256×4）存储器ROM	生产中
MH7489	64位（16×4）存储器RAM	生产中
MH74S201	256位（256×1）存储器RAM	生产中
MH82S11	1024位（1024×1）存储器RAM	生产中
MH93425	1024位（1024×1）存储器RAM	1985

注：在开发方面还有在1985年安排的项目双极性存储器ROM和PROM（512×8）以及ROM和PROM（1024×8）。标志MH的电路已经或将要在TESLA Rožnov生产；标志MHB的电路已经或将要在TESLA Piešťany生产（表1取自1984年的文章）。

**2716**——采用硅栅NMOS 工艺的静态存贮器RAM, 1K×4位, 访问时间200毫微秒, 供电电压5V;

**6561**——采用硅栅CMOS工艺的静态存贮器RAM 256×4位, 访问时间350毫微秒, 消耗功率0.1W, 供电电压5V;

**8155II**——容量256×8位静态存贮器RAM和22条输入/输出的扩展器, 采用HMOS工艺, 访问时间400毫微秒, 供电电压5V。

该任务的部分内容是开发能给单片微型计算机编制程序的学校实验室用微型计算机TEMS48和仿真器TEMS49并投入生产; 仿真器使开发系统能够让实时系统在直到255指令周期内运行, 调整存贮器RAM中的程序并改变8048系统寄存器的内容。同时要建立程序库并进行系统用户培训。为实现该项目, 要将单片微型计算机在消费和投资电子学中的典型应用硬件系统装置和软件投产。

对于大量的应用, 典型的单片微型计算机只需要数目有限的外设和简单的应用软件, 例如用于部分工艺过程、工具机、汽车发动机和调速箱、专用实验室及医学设备、消费电子设备、电子游戏机、玩具、录音机、电唱机、彩色电视机和类似设备的控制。这种单片微型计算机可能形成重大的技术和经济革新, 和8080微处理机相比, 它把整个控制与计算系统集成到单片上, 可靠性提高, 成本、重量、体积和功耗减小。

微处理机和单片微型计算机已在捷克斯洛伐克国民经济的许多部门开始应用。为了给采用微处理机和单片微型计算机设备的设计人员创造条件, 从1980年起开始进行培训工作。培训基地TESLA ELTOS, TESLA IMA工厂有充分的能力, 不仅配备了合格的教师和必需的资料, 也装备了教学用微型计算机; TESLA IMA还提供相应的咨询服务。

### 三、与苏联合作

前面已提到, 捷克斯洛伐克在集成电路方面依靠经互会国家范围内的合作来扩大品种。已经和民主德国、波兰与苏联订有条约, 保证捷克斯洛伐克某些型号的集成电路在国外的销售, 一些型号的集成电路现在正在进口或将要进口到捷克斯洛伐克。

但在微处理机与单片微型计算机的开发与生产

方面, 只与苏联合作。

在苏联, 微处理机和微型计算机电路的发展于1976到1983年间达到空前的繁荣。除4位微型计算机系列K145的各种型号外(K145IK18和K145IK19), 还生产8位微处理机系列K580(和Intel 8080相当)和16位微处理机系列K1810(与Intel 8086相当)。

4位微型计算机系列K145适合在一切对计算速度和存贮容量要求不高的地方使用。因此存贮器ROM中存放的程序主要用来解决要求不高的设备的自动控制问题。根据苏联的经验, 它用于工艺过程、消费电子设备 and 机器人的控制方面是最经济的。要在电路制造过程中写入ROM存贮器中的程序, 因而要决定开发和制造一种型号的4位微型计算机必须保证在经济上是有利的, 即每年要生产几万块。

苏联K145系列电路的基本数据

	K145 IK18	K145 IK19
存贮器ROM 容量	5500位	4000位
存贮器RAM 容量	76×4位	34×4位
指令数	40	32
时钟频率	100/200KHZ	30/40 KHZ
供电电压	27V	27V
引线数	48	48

上述系列的辅助电路在表2中列出。

8位微处理机方面, 捷克斯洛伐克生产并供应苏联的有下列型号的电路:

K589 IR 12——通用8位输入/输出(锁存器), 等效于3212;

K589 IK 14——优先级中断电路, 等效于3214;

K589 AP 16——4位双向不倒相总线驱动器, 等效于3216;

K580 IK 51——USART通用同步异步接收器/发送器, 等效于8251;

KR580 VI 53——可编程序定时电路, 等效于8253;

KR580 IK 55——可编程序并行接口, 等效于8255;

KR580 IK 57——可编程序直接存贮器访问控制器, 等效于8257;

KR580 VN59——可编程序中断控制器, 等效于



8259。

(符号用拉丁字母写出)

大多数这种电路已在捷克斯洛伐克的国营企业  
TESLA Piešťany和TESLA Rožnov 生产  
(参看表1)。

TESLA Rožnov从1984年起为微处理机

8086生产下列电路:

MH 8282—  
MH 8283— } 八位锁存器  
MH 8286—  
MH 8287— } 八位总线接收器/发送器

表2 微处理机系列K145 (K145 1K18和K145 1K19)

K145	IK1801	主要用于和计算系统的连接; 系列145带外部发送器(接口)
K145	IK1802	用于在以K145型为基础的计算器系统中对波兰DK278型电机 械式打印机的控制
K145	IK1803	用于对穿孔纸带信息输入/输出设备的控制
K145	IK1805	用于对“ELEKTRONIKA UTP-15”型热敏打印机的控制
K145	IK1807	用于对自动洗衣机、高频烤炉、控制器、各种工艺设备和各种过 程的控制
K145	IK1808	用于各种设备中模拟数字信息变换, 处理与检验
K145	IK1809 和1810	微型计算机用的I/O处理器, 该微型计算机具有EIEKTRO- NIKA60语言的宽广的指令系统和灵活的寻址系统。
K145	IK1811	用于数量高达100操作数的数据采集与存放设备的处理器
K145	IK1901	用于台式电子钟——有两种激励方式的闹钟
K145	IK1903	用于汽车维修电子学, 用于速度、燃料消耗参数瞬时值的自动计 数与检验
K145	IK1905	用于十进制代码信息的接收, 保护、发出和检验, 例如在电话方 面用于32个用户的自动连接
K145	IK1906	用于盒式磁带录音机中磁带控制机构活动的检验与控制
K145	IK1907	可编程序定时信号源, 用于在99小时99分到0的范围内以1分的准 确度进行各种过程的实时控制。
K145	IK1908	可编程序定时信号源, 用于在99小时99分到0的范围内以1秒的准 确度进行各种过程的实时控制
K145	IK1909	可编程序定时信号源, 用于在9999秒到0秒范围内以1秒或0.01秒 的准确度进行各种过程的实时控制。

## 四、结束语

前述微处理机系统MH3000、MHB8080的生产  
和单片微型计算机系列8048的开发实际上已耗尽  
了捷克斯洛伐克微电子学器件基地在第七个五年计  
划期间的生产与开发能力。上述品种进一步扩展将  
在完成研究开发规划时考虑, 在生产方面主要采取  
现有品种革新的道路, 即用新的型号代替老的。

在第八个五年计划(1980~1990)期间, 与苏联

合作扩大单片微型计算机系列8048, 例如采用CMO  
S工艺的电路80C48, 也考虑带集成A/D变换器的  
微型计算机8022、微处理机8087、8088以及其他型  
号的微处理机。也考虑研制质量更好的微型计算机  
系列8051。这个系列的电路用HMOS II工艺实现,  
和系列8048相比, 使用值大约提高十倍。

选择要开发的具体电路型号取决于捷克斯洛伐  
克工业企业和经互会的合作国家的实际需要。在评  
估现代微型计算机的质量参数与实际情况的基  
础上, 提出在第八个五年计划期间捷克斯洛伐克微电

# Sun-3工程工作站的系统结构 (续1)

## 二、软件体系结构

### 1. 软件构成及操作系统

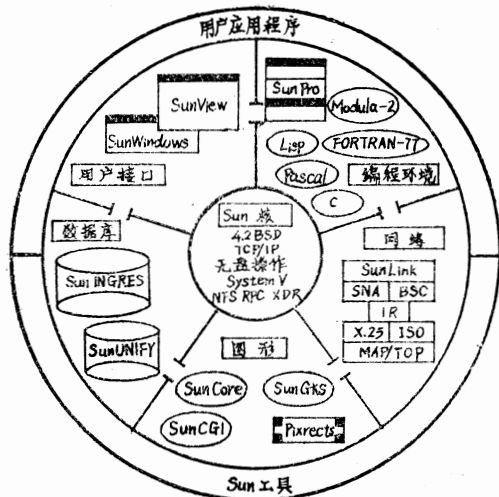


图1 Sun-3软件构成

(1) 软件构成 Sun-3工作站给最终用户及软件开发者提供了通用而灵活的环境。

采用各种标准及开放的系统结构是 Sun 的特点。遵从工业标准原则，指导Sun公司在Sun-2和Sun-3的每一个层次的系统结构中作出相应的选择。Sun的操作系统、语言、数据库、网络规约和服务程序及图形接口都是在开发高性能工作站中受标准广泛影响的例子。在没有工业标准的地方，Sun致力于建立相应的标准。

Sun提供完整的用户和编程环境。在 Sun 软件结构的每一级上提供有文件说明的编程接口，同

时又允许在每一层适当调整软件环境。第三厂家提供了上百种产品，保证在各应用领域有丰富的软件包在Sun的工作站上运行。

Sun使用Berkeley 4.2BSD的UNIX操作系统作为Sun软件构成的基础。它提供整套语言，包括FORTRAN-77、Pascal、C及汇编。Sun也为人工智能和专家系统的开发提供Common Lisp作为一个选项。

Sun的系统软件不但包括一组编程语言，还包括Sun的编程环境，Sun Pro™。SunPro与UNIX成为一体，是快速产生程序原型的理想工具。SunPro与SunView™，即Sun可见的集成窗口环境充分结合。SunView为软件工程师提供先进的用户接口工具，可用于改进软件开发环境并能很容易地把先进的用户工具结合到程序中。Sun的数据库环境分享SunView灵活的数据存取特色，并增加了关系数据库结构中数据的独立性。

对软件结构的评价越来越取决于它的通讯能力。Sun公司相信Sun-3工作站仅配备了开放网络的一部分。Sun的结构允许与其它厂家的系统结合。Sun的网络文件系统(NFS)是一个公共领域规约，它和SunLink™支持对采用多种工业标准规约的各种其它系统进行访问。

位映象图形也是Sun环境的一部分。标准和性能是Sun开发图形支持功能的指导思想。SunCore、Sun CGI™和Sun GKS™遵守三个工业标准的图形编程方法。

Sun软件的这些组成部分和Sun开放系统的宗旨相一致，以最廉价的方式为Sun-3工作站提供优化的性能。

### (2) 操作系统

产这三种系列。

总之，捷克斯洛伐克发展微处理机有三个特点：(1) 面向Intel公司的微处理机系列；(2) 重视单片微型计算机；(3) 与苏联合作。

1) 结构 操作系统是系统软件结构的核心,它是最基础的软件,提供多个抽象的层次,使用户不直接和使用不便的硬件接口打交道。操作系统作为通信量控制和管理者,负责在各种应用程序所调用的进程间进行进程控制及资源管理。软件设计者在操作系统基础上编写软件,可以避免与硬件直接打交道,也不必知道其他进程的情况,但它们被限制在操作系统的能力范围之内。应用的性能取决于围绕着硬件核心的操作系统的抽象程度。性能和抽象的层次不可避免地发生冲突。然而如果缺乏由操作系统提供的防止硬件层影响的有效保护,会使软件变得不可移植而与硬件相关。操作系统的设计本身可能是这样与硬件相关的,因而它基本上是不可移植的。

2) Sun对UNIX的扩充 Sun的编程环境,以窗口软件为基础的图形用户环境, Sun的网络环境及图形接口等均来源于UNIX的灵活性。每一种情况中, Sun对UNIX的扩充都增加了其功能。增强的诸方面,特别是窗口和图形接口已经通过剪裁用于高性能的工作站。一个重要的辅助机种是无盘工作站。无本地盘的工作站能够在全网络范围内自举,并通过网络在全网络范围的存储设备中进行分页和交换。不同硬件结构的多代硬件产品能够透明地共存于同一网络上,即Sun-2, Sun-3工作站共享同一网络上的同一目录,应用程序和数据库。网络的管理以及两代工作站的管理象仅有单一结构那样处理。Sun系统为工作站结构自举引导相应的操作系统。工作站能够透明地访问远程文件,进行远程过程调用并执行所有网络存取功能而不论安装在源或目的节点上的Sun的硬件结构如何。

3) UNIX的未来 首先,要发表的Sun的UNIX版本将和AT&T System V的UNIX在应用接口一级兼容,保留Berkely 4.2BSD版本的功能。System V应用接口将充分结合到Sun操作系统的环境中去。程序员及用户将能同时获取4.2BSD及System V的应用接口的优点; Sun也将继续支持及加强统一的UNIX环境。

第二条统一途径(可能对发展UNIX是最重要的)是Sun同意和AT&T建立单一的,兼容的UNIX接口。这个新的UNIX将有4.2BSD的全部功能并能利用运行在System V上庞大的应用软件包。

## 2. 数据库软件

Sun的数据库环境与其它Sun环境是一个整体。数据库访问可通过一个与Sun语言和编程工具一致的编程接口来进行。Sun网络工作站的特点是要求采用分布式的数据访问。在Sun的软件结构中,关系数据库模型是这个完整数据库环境的基础,这个数据库环境利用了Sun的用户接口、图形和网络。

(1) 关系数据库 信息的存储与检索是计算机系统的基本功能。将“低级”的基本能力转变为易使用的服务是数据库的目的。数据存储本身是必要的,但还不够充分,组织和访问是解释信息的关键。数据必须按照一定的逻辑方式构成,并且可按易定义和易理解的一组规则和命令来检索。灵活性是基本的特性;当不能增加,重新定义,删除部分信息时,必须重新构成或重新装入完整的数据库,必须识别信息的动态性质。信息的外观应能很容易改变而不导致计算机和人的巨大开销。还应能迅速、有效地寻找和更新记录。

为解决这些,引进了数据的关系模型。数据的独立性(即用户的数据结构和数据的表示方式是分离的)及用户管理所存数据的容易程度被认为是数据库设计的主要出发点。关系模型提供了一种数据模型,这种数据模型与数据的内部表示无关。数据模型允许重新定义用户的数据结构或视图,而不影响数据实际存储的格式。关系模型仅保持一种数据结构(一组矩形表格)来存储用户信息。非过程的命令(或查询)语言是定义用户视图和组成数据检索命令的工具。

关系数据库模型所依据的原理是Sun结构特点的核心。关系模型在改动数据库环境中的灵活性与容易改动Sun的环境是相呼应的。Sun用户环境的一个重要特点是提供了一个公开的数据操作接口。数据内部表示的一致反映出使Sun的用户环境统一的接口属性值的清晰。

为给最终用户和OEM提供适当的数据库工具, Sun提供了二个关系数据库系统, SunUNIFY™和SunINGRES™。

(2) SunUNIFY SunUNIFY的主要特点是网络关系数据库的高性能。通过对32位工作站能力的开发, SunUNIFY在单独使用(或多用户)和网络配置中获得了明显的交互性能的优点。

SunUNIFY将网络数据库系统中的指针结

构结合进关系模型中。指针结构通过避免频繁的索引查找来减少额外开销。通过使用原始 (raw) 文件方式, (这种方式去掉了负担在UNIX的那一部分开销) 进一步增强了性能。在原始文件方式中, 数据库记录可以完全绕过UNIX文件系统, 储存在相邻的物理页上。SunUNIFY的相邻文件结构与UNIX共同存在盘上, 但物理位置不同。由于相邻地存储数据, SunUNIFY极大地改进了查询和数据输入的响应时间。

在分布式处理环境中, UNIFY核心放在一个专用服务器上, 而数据库应用软件和工具在数据库用户工作站上运行。Sun操作系统的远程过程调用和记录锁定功能控制着数据的完整性, 并允许对数据库进行快速修改。

数据库的性能也借助于SunUNIFY的结构查询语言 (SQL)。为了快速查询, SQL动态地选择最有效的连接次序, 访问方法和表达式估值顺序。如有必要, 程序员可使用一个专用接口ERIC直接进行自身的优化。

1) 扩展的数据模型 近年来, 关系模型已日益普及, 原因是其基本概念 (表) 容易理解。数据用行 (或记录) 和列 (或段) 表示, 列可容纳各种广泛的数据类型, 包括整数, 日期, 计费 和 字符串。

SunUNIFY把结合段和参考段增加到数据库中, 扩充了常规的关系模型。结合段在各数据段中规定一种层次关系。参考段提供了“参考的完整性”, 它允许系统自动检验参考其它记录时, 记录的准确性。参考的完整性保证了参照体的存在, 允许数据物理表示的优化以便更快地访问, 并能通过把很多表中的信息组成逻辑实体的方法, 使用户对它的透视得到简化。

可以给各种表指定主关键字 (一个段或多个段, 但对每一行来说, 它们必须是唯一的) 以保证逻辑数据的完整性。辅助关键字 (索引) 则可增强查询性能。

2) ERIC程序员的接口 除了标准的 UNIFY程序员接口 (用以保证与现有的UNIFY的应用软件兼容) 之外, 一个增强的SunUNIFY程序员接口 “ERIC” 也可供使用。ERIC (扩展的记录级接口约定) 是一个由Sun提出的标准, 可使数据库访问在大范围数据库系统间进行。

ERIC提供了90多个用C语言写的子程序帮助

构成一个应用的数据操作, 屏幕显示和报告。使用ERIC, 程序员可去掉在其它数据库系统上所需要的查询语言的开销, 使ERIC适宜最优性能的应用。

3) 内部的数据库表示 SunUNIFY中的记录能以二进制形式存入备份和从备份装入内存, 从而改进了备份的速度。为作备份, 记录也能以ASCII格式进行读、写, 以便在SunUNIFY和其它关系的和非关系的数据库系统间直接进行数据传递。

4) 数据库的特性 在SunUNIFY的数据库中最大保存2千兆记录。每个记录可达25600字节。数据类型包括长、短数字、计费、浮点、字符串、日期和时间。一秒钟能修改10个记录, 每秒钟可检索200个记录。一个简单查询所用的实际回答时间几乎不到100毫秒, 其速度相当于相竞争的其它系统的10倍。

SunUNIFY不仅限于Sun工作站。它是由UNIFY公司的数据库产品中派生出来的, 目前有100多种计算机可使用SunUNIFY。

(3) SunINGRES SunINGRES是一种全关系型数据库, 它通过QUEL查询语言提供查询。SunINGRES提供了一种表格工具, 使用和组织数据库命令很容易。SunINGRES表格化应用接口支持快速编制程序原型, 它允许迅速生成并测试以QUEL编写的应用软件。

FORTRAN和C编程接口允许在FORTRAN和C源程序中嵌入SunINGRES命令。这些语言和QUEL的功能通过嵌入的QUEL (EQUEL) 结合在一起。报告生成工具允许SunINGRES的应用软件使用缺省格式或用户编制格式生成报告。

INGRES/NET允许产生一个基本的网络数据库系统。用户可以访问远程Sun工作站上的SunINGRES数据库。同SunUNIFY一样, INGRES关系数据库系统具有广阔的市场 (特别是在DEC VAX共同体中) 使用Sun工作站上的SunINGRES就在Sun工作站和VAX VMS或UNIX操作系统环境下的INGRES之间建立一条链路。

1) 事务处理和视图 SunINGRES极大地简化了多用户中数据库事务处理的并发控制。在硬、软件失效时数据的完整性得到进一步保证。

SunINGRES让用户选定自己的数据视图。

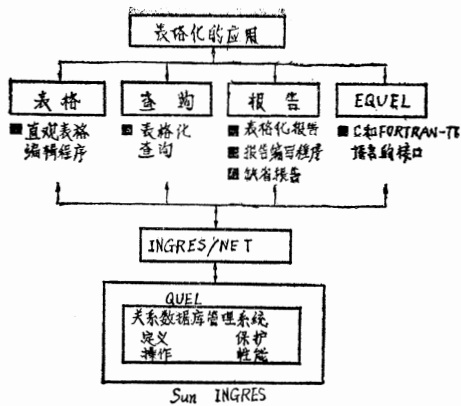


图2 Sun INGRES数据库

用户可以只看表中的部分数据，也可看不同表的组合数据。这个特点消除了不同表中数据的重复。同时视图也提供了一个机构，保证某一表中的保密数据得到控制。

(4)未来的数据库环境 Sun的INGRES 和 UNIFY的信息检索工具仅仅是Sun提供的一个通用数据库系统的初始步骤，这一通用数据库系统利用了Sun工作站。Sun设想的信息管理设施的范围要比目前正在使用的数据库管理系统更广泛。

SunView和SunPro为在Sun用户接口和编程环境下提供与图形和程序目标的一致的，开放的接口奠定了基础。Sun数据库的策略是继续朝关系数据模型向数据目标和数据库结构相一致的接口方面发展。NFS（网络文件系统）和RPC（远程过程调用）在分布式数据库管理中为进一步通用化奠定了基础。

在一种数据库结构中，用户需要处理不同类型的数据（文本，统计，图象，声音）并显示它们的方法（浏览，图形，文本寻找）。要继续进行目标和工具接口的通用化。符合标准并和多个厂家兼容是Sun图形和网络环境的特点；保持标准数据格式，并与外来数据库协作，是与Sun坚持提供通用数据库环境的立场相一致的。

### 3. 图形开发台面

高分辨率、位映象显示监视器正迅速取代专业技术人员惯用的台式字符显示终端。随着向更先进技术的过渡，图形应用的需要正逐级上升。甚至用二维能力的字符CRT，图形程序对越来越多的应用来说已经变得不够用了。为满足这些依赖于图形的程序的需要，软件工程师需要有效的图形开发程

序，以便进行工作。

(1) 图形工具系列 Sun公司已研制了一个图形开发环境，它综合了图形标准，先进的窗口系统，网络文件系统访问，高速图形流水线硬件等功能。这种综合意味着图形应用软件能充分利用系统工具而不牺牲图形标准提供的可移植性，Sun的综合图形环境提供了贯穿产品系列（从简单帧缓存到高速图形流水线）的功能的一致性以及对设备的独立性。

几年来，Sun已提供了一个高速的、对设备独立的位映象图形软件的Pixrect（象素矩形）层以及Sun Core（Sun的SIGGRAPH Core图形标准的实现）。1984年11月，Sun发表了Sun CGI，它以美国国家标准学会（ANSI）推行的图形标准为基础。1985年7月Sun宣布 Sun GKS，它是GKS图形软件的一种实现，GKS是ANSI推荐的另一个标准。

每个图形软件包都与SunWindows™结合在一起，SunWindows是Sun的多窗口用户接口，它使写在这些图形标准软件包上的应用程序能在窗口环境中运行。所有这些图形软件包还可利用带有硬件图形处理器选件的Sun工作站的优越性能。

(2) 整体的图形 过去，典型的图形应用被限制在使用单一图面的一个进程中，通过它一个用户能够操作和检查数据目标。通常，用户接口是由一批大的、不灵活的、粘在数据小平板上的菜单组成，高性能工作站的出现已极大地改变了这种情况。一个工作站多种、可重叠的窗口系统，可修改的菜单（例如上托式菜单）和位映象工具（如图标及光标造型）已增加了使用图形软件进行应用程序开发的可能性。

现在，一个工作站上的各种窗口能执行不同的作业或允许多种数据表示方法，通过工具，如鼠标器，用户不用从屏幕上移开目光就能观看上下文，在它们之间进行切换及执行复杂的交互动作。

在下述各方面，Sun充分利用这一技术并加强了图形的编程环境。

·图形标准 在工作站工业中，现有的新用户接口工具有一个可移植性问题。过多的窗口系统和用户接口使应用程序的编写者难以保护他们在增值软件中的投资。为解决这一可移植问题并提供一个开放的交互台面，以便在台面上建立各种应用软件，



Sun将关键的、对设备独立的图形标准与Sun Window结合在一起。

- 交互速度 对于广泛的图形应用软件，性能是头等重要的，与复杂目标有效的交互是基本的需要。Sun的图形设计使用高性能整数坐标的低级图形标准——SunCGI以及一个低成本，高性能的图形加速器——Sun的图形处理器来满足这个需要。

- 网络图形 大的设计工程需要很多人联合努力。设计者、浏览者、检查员、技术文件写作人员和管理员都需要对共享数据进行访问。Sun的图形环境包括网络范围内的网络文件系统（NFS）能力、透明的文件访问和远程过程调用（RPC），这些都为分布式图形方式奠定了基础。

为在这些方面有效地工作，Sun已建立了一个分层的开发台面。Sun图形环境满足广泛的图形开发需要（从低级的Pixrects库到复杂的网络图形软件）。

(3) 位映象图形——Pixrects Pixrect库是最低一级的与设备无关的图形支持。它为Sun的图形软件奠定了高性能基础。很多重要的应用，包括SunWindows，都存在于Pixrect层之顶部。Pixrects使用一种面向目标的编程风格，它隐藏了具体显示设备的特性，以使所有象素矩形呈现同样的统一接口。这种统一接口适用于存贮在主存中的Pixrects，也适用于帧缓存显示设备上的Pixrects。目标以整数象素坐标定义，所有的Pixrects对延伸出他们边界的目标均加以剪裁（除非明显地不允许剪裁）。

Pixrect操作包括对向量，多边形、图象、文本、纹质及曲线封闭区域填充的操作。还包括一个漏印板操作，它用在任意形状区域内，绘制纹质，颜色及图象。彩色和单色的Pixrects都可供使用，各类型的Pixrect有同样的统一接口。

Pixrects规定颜色表或颜色查找表是一个统一的概念。它按照帧缓存中象素的内容为输出颜色赋值。颜色表（与一个平面屏蔽一起）允许作双重缓冲，以使动画画面平滑。平面屏蔽限制对帧缓存中指定平面存储器进行的写操作。

Sun的pixwins与pixrects相似。这个图形软件功能很象pixrects，它也处理SunView窗口系统所需要的锁定及矩形框裁剪。

(4) 图形标准 虽然Pixrects是高效的，但它并不提供工业界图形标准的可移植性。为满足这些需要，Sun提供三种图形标准软件包，同时打算在将来增加新的标准软件包。Sun Core为三维交互图形规定了一套输出图元和输入功能。Sun CGI（以ANSI和ISO计算机图形接口绘图标准为基础）为有效地产生二维图象提供了一套丰富输入和输出功能。为支持交互的二维图形，Sun GKS采用建立好的ISO和ANSI图形核心系统（GKS）标准。

1) SunCore SunCore以计算机协会（ACM）的Core规范为基础，这是美国首次在图形标准方面作的努力，虽然还没有一个版本的标准被承认，但它已经成为工业上一个重要的事实上的标准。SunCore为在二或三维浮点用户坐标中表现目标提供各种图元，并为从不同透视方向和不同光照条件下观察这些目标提供各种功能。SunCore也允许把一个图形映象描述分割成若干图元群，称之为段。段就是一群输出图元和属性的集合。用户通过鼠标器的指点能够选择建立、删除、变换及突出某些段。

在SunCore中，可供使用的图形输出元件包括向量、多边形、标记、文本和光栅。每一输出图元都可有其属性，如颜色和纹质。虚拟存贮Z缓冲器允许绘制平滑浓淡及隐面消除的多边形。

2) Sun CGI CGI（计算机图形接口）是由ANSI主持正在研制的一个标准。在写这本材料时，这一标准已接近完成，Sun的版本非常接近预期的最后规范。在窗口系统集成，与C语言的联结及输入功能方面，Sun已对ANSI设计的这一标准作了多方面贡献。SunCGI有一个二维整数虚拟设备坐标系，它避免了在Core和GKS中所需的浮点变换，因此，SunCGI在低端工作站上是有效率的。

Sun CGI提供了一组扩充的输出图元，包括复合线、多边形、多标记、园、园弧、椭圆、椭圆弧、填充纹质或图象的饼图，园及椭圆与弦构成的弓形、文本、光栅、单元阵列及位块传递（bitblts）。还提供了一些输入例行程序以及处理采样、请求或异步事件队列输入。异步输入对于事件驱动的、与图形实体的交互操作是方便的。

3) SunGKS GKS（图形核心系统）是一个二维的标准，它是国际上承认的图形标准。Sun

GKS是一个完整的2C级版本的GKS标准。GKS目前还是一个二维标准，现在标准委员会正计划进行有限的三维扩充。

GKS在结构上近似于Core。GKS的主要优点在于它是一个非常精确、有很好规范的标准。它还有一个处理属性的更灵活的方法，使它更不依赖于设备。GKS有一个显示表，它象Core一样，是段的一个线性表。可以看到段在不同视图面的各种变形。GKS支持事件驱动输入并通过指点选择屏幕上目标的输入功能。这个标准也规定了中间文件（GKSM）的格式，用于传递和归档图形目标。SunGKS包含着对GKSM的支持。

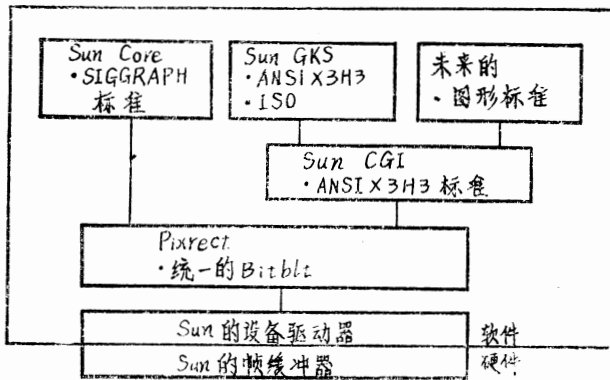


图3 Sun的图形标准

(5) 窗口中的图形 在窗口中，综合图形标准允许开一个或多个窗口作为视图面，通过使用Sun Windows接口，用户可以控制视图面，如定位、改变其尺寸或减小成为图标。标准委员会最初的意图是要把一个实际的显示表面作为一个视图面来对待，每一个实际显示只有一个视图面。

为了把Sun Core与窗口系统结合起来，Sun的设计把窗口作为虚拟设备来处理，并把视图面与窗口紧密结合在一起。这种方法的优点是一个单独的应用进程可以在工作站显示器上使用多个重迭的视图面。当一个窗口移动、揭去复盖或改变尺寸时，SunCore或SunGKS就从SunWindows上取来一个信号，对图形的图象再次进行比例缩放，以适合窗口的大小。这样做不需要应用软件介入，这意味着按照标准写的程序在窗口中运行时不需要修改。这些程序的优点是他们的视图面在任何时候都能独立定尺寸，独立定位并可制成图标。

除了目前没有段处理外，SunCGI与窗口系统的结合几乎同Sun Core的情况一样。当窗口揭去复盖或规定尺寸时，应用程序必须重新绘制它的窗口。Sun CGI也在用一种Pixwin兼容的方式工作，使用窗口坐标并允许把视图面附到现存的子窗口上。

(6) 硬件图形处理器 Sun提供一个图形处理器（GP）插板和图形缓冲器（GB）插板，它们把图形操作的性能提高到原来的2—250倍（视操作而定）。图形处理器使用一个二级流水线和一组浮点器件进行变换、裁剪、比例缩放，并使用向量、多边形和矩形进行上屏操作。图形缓冲器选件板增加了三维隐面的功能。

图形处理器在窗口和多进程访问中支持图形。它剪取一组矩形以支持重迭窗口并为处理多进程访问提供一个共享存储器环形缓冲器。高速度的图形在多窗口中同时运行。执行二维操作的应用只从GP中受益，不能从GB中得益。GB是加在GP上的可选部件，对于包括三维图形的应用，如实体模型或机械CAD，GB改进了上屏处理的速度。使用Sun图形库编写的应用软件可以在有（或没有）GP或GB情况下运行，因为GP及GB硬件对于应用是透明的。

图形处理器使低价工作站具备了在窗口中做实时三维图形的功能。Pixrects，Sun CGI，SunGKS和Sun Core都与图形流水线相结合。Sun Core具有最大的优越性，它使用图形处理器的峰值速度（每秒4兆浮点操作）去作矩阵乘法和三维裁剪。Sun Core能读它的显示表并以20 000/秒以上的速度绘制向量。Sun CGI用图形处理器画复合线和多边形。

SunCGI也使用软件的Pixrect层，它得到了图形处理器的加强。Pixrect库利用图形处理器绘制二维多边形，包括凸、凹、重迭的多边形及带洞的多边界多边形。图形处理器画二维多边形的速度达4000/秒，它也加速了Pixrect绘制向量和光栅的操作。用向量画图的象素速率提高了7倍。使用恒定填充颜色的光栅操作或在彩色帧缓存内从一处到另一处复制操作的速度增加2~3倍。

(7) 在一个网络中的图形 图形工程的生产率象其他数据库及软件开发工程一样，可由局部区域网络增加，由Sun设计的NFS、网络上的远程文件系统能够安装在本地工作站的文件系统上，以

# 港口外贸费收管理系统的分析与设计

港口费收管理是港口生产经营活动中一项十分重要的业务工作。涉及面广,政策性强,费收管理水平直接影响港口的经济效益和信誉。

人工港口费收管理是一项繁重的脑力劳动,需配置一定数量训练有素的业务人员参加此项工作。但由于工作量大,往往造成计费不及时和错收、漏收等现象。因此,提高港口费收管理水平是增加港口经济效益、促进港口管理现代化水平的当务之急。

## 一、人工费收管理的现状

### 1. 费收管理的组织机构

天津港费收管理工作主要在公司一级。各港埠公司主管费收管理工作的是货运科和财务科。货运科计费组专门负责各种费收项目的计算。财务科负责各种费收款项的结算及与费收有关的各种财务统计工作。

外轮理货公司的费收管理由业务科负责。

港务局的费收和费率管理由业务处商务科负责。

### 2. 费收管理的业务功能

- 收集、整理与费收有关的各种资料,其中包括仓单、发货单、理货单和各种作业单等。

- 管理各种费规和费率。

- 计算各种费收项目并向各付款单位(船主和货主)核收款项。

- 各种费收的统计(单船统计和分货类统计)。

### 3. 费收的内容

港口费收分两大类:远洋运输(外贸)和沿海运输(内贸)。

(1) 远洋运输 远洋运输包括:进口卸船、出口装船、进口装车、杂项作业费和理货费。

远洋运输工具有外轮和国轮,计费对象有船方付费和货方付费。计费项目近廿项,按不同货类分别计费,部颁费规、费率均不相同。

(2) 沿海运输 除理货费外,外贸运输的费收内容均包括在内,只是各种费率不同。另外还包括运输费及换装费的计收,其货类的划分及费率均与其他费收不同。

### 4. 费收管理业务流程图

远洋运输费收管理业务流程如图1,图2所示。各种费收的计算方法均依据交通部和理货总公司颁布的费规与费率执行。(图见后页)

## 二、系统分析

### 1. 现系统功能的分析

目前天津港的人工费收管理系统是几十年形成的,功能齐全,故可以完成由计费所需的信息收集、整理、计算和统计工作。问题是功能较弱,工作质量和速度对港口经济管理均有一定的影响。

### 2. 对现系统信息结构的分析

根据系统调查,了解到费收管理的信息结构特

便能在很多用户和设计者中间共享图形数据库,不必作多份拷贝。这就消除了对同一信息的多次拷贝中保持一致的要求。数据库文件,比如GKS中间文件,很容易供网络的其他用户使用。

对网络图形另一个理想的特点是在网络上使用一个高性能、高端服务器存放数据库,而在本地工作站上进行图形交互操作,这可以通过使用NFS中的远程过程调用(RPC)例行程序实现。RPC在计算集中部分和应用的交互及显示部分之间形成

了过程连接。目前Sun正在改进在不同图形包上的图面概念,以便更好地利用以RPC为基础的交互的潜力。对于需要图形的应用,网络图形环境提供了共享数据,改进安全,数据库一致性及节省存储的有利条件。

(未完待续)

天津市电子计算机研究所

袁野 汪霞 编译

黄侃 校

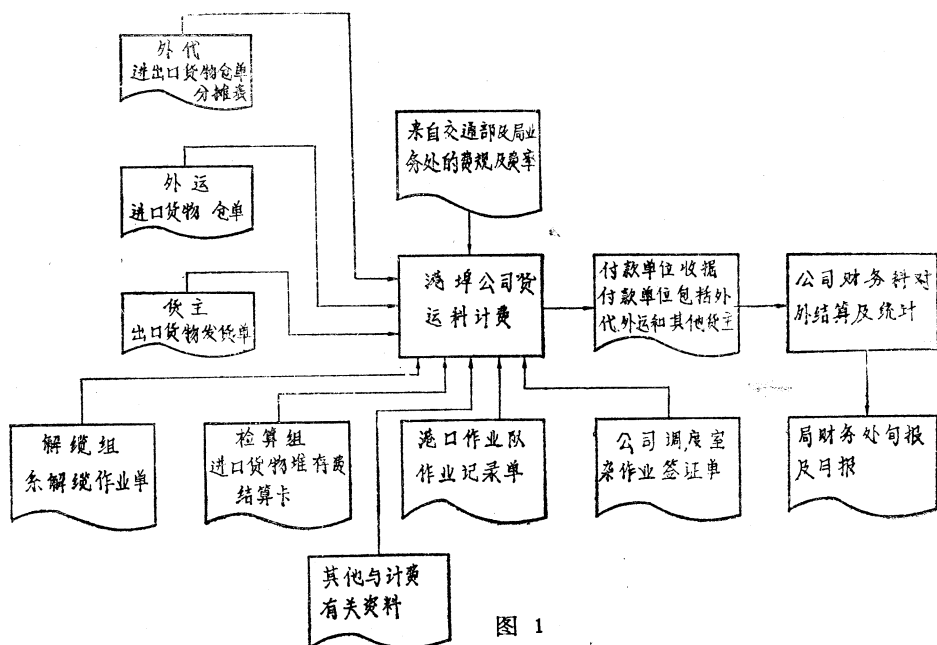


图 1

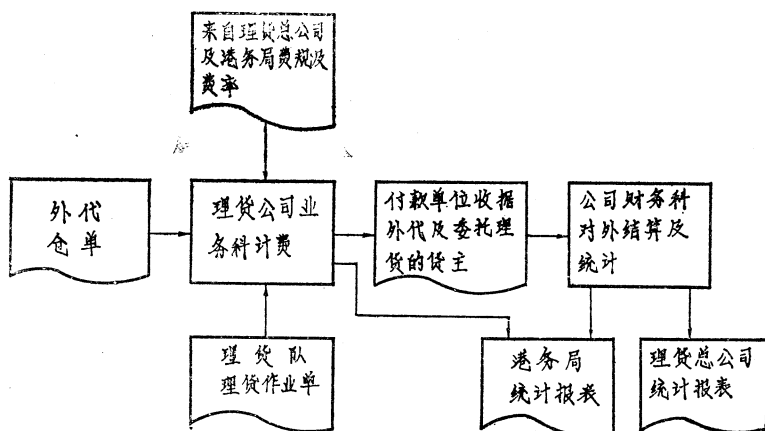


图 2

点是既有大量的静态信息又有大量的动态信息。静态信息不是绝对不变的，在费收计算中做常数使用。

静态信息主要是部颁费规和费率中包含的信息。主要费率表有：装卸费率表；货物港务费率表；港口货物堆存费率表；港口货物陆运装卸费率表；杂作业费率表；港建费率表；理货费率表。此外，还有常用的代码、货名、及有关的船主、货主等信息。

以上费率均与货类有关，每种费收项目对货类分法也不相同。有的费率表要求几十种货物分类，

有的要求几种。因此费率信息量较大。在人工管理系统中，较有经验的费收人员，对经常接触的主要货类的费率有记忆，但大多数费率使用时要查表。一旦费率变化，由于计费人员需要熟悉新的费率表，往往一段时间内计费工作效率有所降低。

动态信息是随着每船进出口货物运输而发生的，其信息量是一个累积量。时间愈长，进出口货物愈多，信息量愈大。

动态信息主要包括以下内容：有关的船舶特性；分摊表；仓单；发货单；各种作业单；经过费收计算产生的各种付款收据；各种统计数据报表。

每条船的信息量不同。一般船装的货物的票数多、货物种类多、货主多，信息量就大。不仅输入信息量大，输出信息量也大。出口货物品种多，货主多，因此一般讲出口船的信息量要大于进口船的信息量。各种统计数据报表，信息量相对讲比较一定，只是保存的时间愈长，积累的报表越多，积累信息量就愈多，与时间成正比。

### 3. 现系统环境分析

费收管理系统从业务功能上讲实质是一个数据处理系统，它是港口管理信息系统中货商子系统的一个比较大的数据加工点。

现人工系统是计费人员将各种计费依据（数据）收集整理后（见业务流程图），经过计算，开出各付款单位的收据，并按旬、月进行统计。

因此，现人工系统工作时，计费人员通过信息交换与外界环境发生供需关系。这种关系既有局（业务处，财务处）和公司（货运科、财务科、计费组）之间的纵向关系，又有公司内部、公司和外代、外运、银行、各货主之间的横向关系。在人工管理中，这些关系彼此之间比较协调，保证了费收管理工作顺利进行。目前的问题是有时渠道不通，信息收集不及时，往往影响计费不及时，这是靠人工传递信息时致命的瓶颈问题，在没有实行计算机管理网络化之前，难以彻底解决。

此外，在人工管理中，费收计算与财务统计分属于货运科和财务科，这增加了信息的交换，这是现编制造成的问题。

## 三、新系统的设计

### 1. 设计思想

新系统首先着眼于解决目前费收管理中存在的主要问题。新系统代替人工系统时，应见到明显效益，这是系统成败的关键。

新系统应智能化，尽量减少操作人员的脑力劳动量。系统输出信息要英汉结合，输入信息的汉字量压缩到最低限度。操作过程中，系统要汉字提示，减少操作错误。系统应具有自动确定费率的能力。目前交通部与港务局颁布的费规和费率都要纳入此系统。

应模拟人工，高于人工。现行系统符合交通部统一的费规和费率，有许多可取之处，新系统基本内容要模拟人工。但新系统利用的计算机较算盘和计算器有明显的特点：大存贮能力；高运算速度；

不需要人的干预；不仅可进行算术运算，而且可以进行逻辑运算；对数据的自动寻址和快速传递信息等。因而，计算机管理完全可以按操作性质集中处理，归结为“输入—计算—输出”标准的计算机处理过程。目前还没有实现计算机联网，操作人员比较多的工作量是数据录入工作，“计算—输出”由计算机自动进行。由于操作简单，不仅可提高工作效率，而且每个计费人员都可胜任全部计费工作。

新系统应有较强的应变能力，很快地适应费率变化。因此，新系统的设计对费率表管理要给予充分注意。

新系统对用户友好，简便易学。原计费人员经过基本培训，就能独立操作。为此，尽量采用当前业务人员已经熟悉的术语、代码，以及为广大用户所欢迎的菜单驱动、汉字屏幕、表格方式、数据填入等操作方式。

具有通用性。系统经一个公司试运行成功后，其他公司经移植可直接使用。系统设计在符合部颁各种费规和费率的前提下，尽量照顾各公司已经形成的习惯做法，设计出比较规范的操作程序。

### 2. 系统逻辑结构

我们以外贸港口费收管理系统说明系统结构。系统由二个相对独立的系统组成：

- 远洋运输船舶港口费收管理系统（称系统一）

- 外轮理货港口费收管理系统（称系统二）

二个系统结构分别见图3，图4。两个系统结构基本相似，对应模块的功能也相似。下面分别将四个模块做统一说明。

#### （1）费率表管理模块

本系统中费率表作为固定数据，以表格形式存放。将每个费率表设计成一个数据库，存放在硬盘上。为适应修改费率需要，设计了较强的管理功能，即增加、修改、删除、显示列表和帮助。这样用户可以方便、迅速、实时地修改、删除和检查各种费率。

两个系统的费率表管理模块程序结构完全相同，只不过管理的费率表的种类不同。

#### （2）软盘初始化模块

软盘初始化的目的是在软盘上建立存放船舶输入数据及计算结果的空表格。本系统将每张软盘分成四个分目录，每个分目录存放一条船的数据。为存放不同费收项目的数据库，设计了不同的数据库，



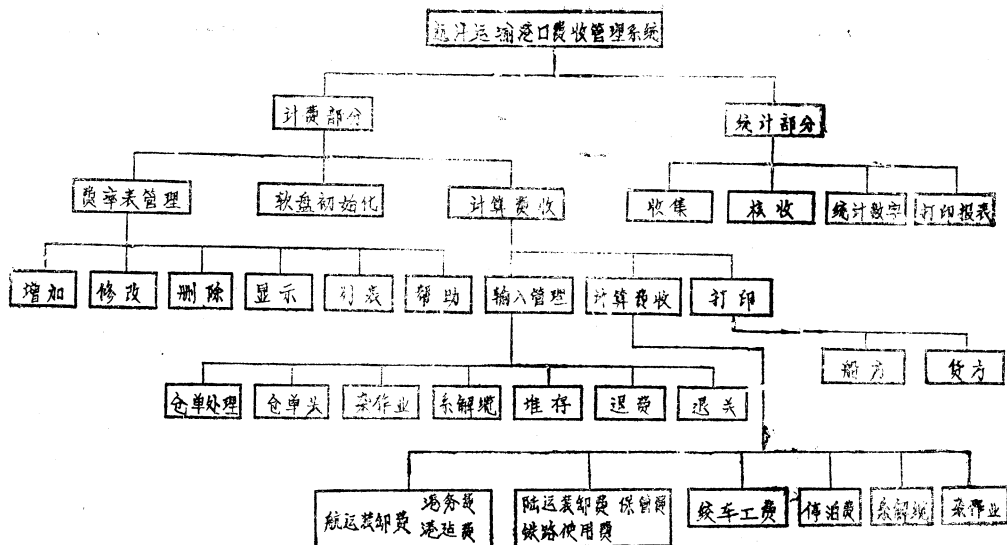


图3

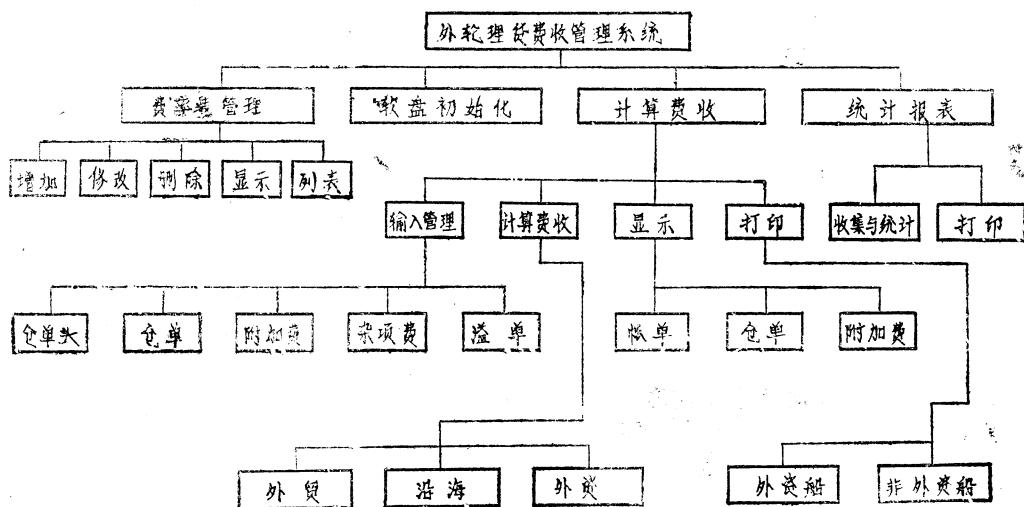


图4

它们的库结构存放在硬盘上，初始化时，将其拷贝到软盘上，即在软盘上建立起输入和计算时所需的表格。

### (3) 计算费收模块

计算费收模块是费收管理中的主模块。进行费收计算时，启动该模块，前两个模块属于服务性的。

计算费收模块，是典型的数据处理过程，“输入—处理（计算）—输出（显示和打印）”。

1) 输入模块是对仓单头，仓单及其它计费用数据进行管理。

2) 计算模块的目的是根据已输入计算机的船

舶特性、仓单及其它信息计算各种费收项目。由于两个系统计费项目不同，这个模块区别较大。《系统二》不仅可以在外贸计费使用，在内贸中如船货主委托也可以用于费收计算。

输出模块的目的是打印或显示计算的结果。根据用户需要，《系统一》只设计打印输出，《系统二》设计了打印输出和屏幕显示。

### (4) 统计模块

数据统计模块的目的是对计费输出的数据进行收集、汇总、打印报表（旬报和月报）。目前该系统是单机运行，数据收集通过传递软盘介质进行。操作人员将每条船的软盘依次通过统计模块送到对

应数据库中。收集完毕后汇总，再根据操作员命令打印各种报表。

《系统一》在信息收集增加核收工作，这是港埠公司财务工作要求，也就是说参加汇总的数据要求是已经财务结算完毕的数据。外理公司无此项要求。《系统二》未设计此项功能。内贸费收系统的逻辑结构要增加运费和换装费管理。

### 3. 系统的物理结构

#### (1) 运行本系统的组织结构

目前本系统为单机工作。运行本系统后，录入数据量还较大，原人工系统计费的组织结构可不作大的变动。人员分工可以不必更细。

(2) 选定IBM PC/XT及其兼容机支撑本系统，该机属较高档微型机，运行此系统按目前要求是可以的。

支撑系统的系统软件是CCDOS操作系统和dBASEⅢ关系数据库系统。

### 4. 数据库文件结构

由于系统开发选用了dBASEⅢ关系数据库系统，因此本系统的输入数据、输出数据及其它工作过程中所需的数据，均采用dBASEⅢ数据库文件结构。

设计中我们尽量减少数据的冗余量。必要的冗余量是为提高数据处理速度。

### 5. 代码设计

代码设计主要考虑了货名代码。由于目前交通部没有定型的货名代码标准，为便于原计费人员操作，系统一采用装卸费率表中的货类代码，即“数字-字母-数字”编码。为使某些货类代表一组货名，增加一位数字编码，扩充为“数字-字母-数字-数字”，做为输入货名使用。

系统二计费时不要求细致分货类，只是为统计需要设计一位字母代码(A-M)。

### 6. 输入输出设计

输入设计指设计出一个什么样式的界面，使操作人员能迅速准确地将计费的原始数据送到计算机中。为此，我们在充分利用系统提供的全屏蔽编辑功能的基础上，采取了一些方便措施，如以表格形式列出应输入的值；尽量采用代码输入和增、删、改功能等。

输出设计的格式设计主要参考原单据和报表的

内容格式，并结合目前所用打印机的特点。

### 7. 效益分析

经天津港第二港埠公司和外理公司试运行结果表明：利用计算机进行费收管理，不仅有社会效益而且还有明显的经济效益。

(1) 使业务人员从繁琐、重复的脑力劳动中解放出来。可以有比较充裕时间，学习业务和研究其他改进费收管理问题。

(2) 打印出的单证规格统一、字体清晰、数字准确，得到中外客户的欢迎，为国家赢得了荣誉。

(3) 初步做到了数据共享。通过计费程序一次输入仓单和其处理结果，经软盘介质传递，为财务、计划、统计等部门汇总提供有价值的信息。

(4) 通过计费程序得到的统计数据较人工要准确得多，提高了信息的使用价值，有效地支持了经济决策。

(5) 直接经济效益比较明显。减少了错收、漏收和计算错误，直接维护了港口与用户的利益。本系统运行后，一条船计费时间一般可提前两天，加速了资金周转，给港口带来了经济效益。在一定时间内，不仅可以收回开发系统所用的资金，也对疏港有一定作用。

## 四、 结束语

天津港港口外贸费收管理系统1986年10月已由天津港务局和天津市电子仪表工业管理局联合组织通过技术鉴定。与会的计算机专家、水运专家、管理信息系统专家和交通部有关部门负责同志，对研制成果给予了充分肯定。它是我国用于港口外贸费收管理获得成功的第一个计算机应用系统，设计成果在港口外贸费收管理领域居国内领先地位，具有推广价值。

该系统选择了当前最普及的IBM PC/XT优选机种，资源丰富，易于推广使用。

参加研制的人员是天津电子计算机研究所的解永兴、肖力、王晶同志。天津港务局唐本政、周倜、刘皖、古翌等同志在研制和推广工作中做了大量工作。

天津港港口费收管理系统研制组

解永兴执笔

# 产品结构优化系统

刘 辉 吴振庆 刘於勋

赵利思 张 帆 王进兴

(郑州机械专科学校)

如何根据本单位现有的生产能力合理地选择产品结构,使企业既能保证国家计划的完成,又能在市场调节作用下获得最大的经济效益,是计划工作中的一个重要课题。长期以来,这一工作大都凭经验和历史资料进行定性分析,这存在很大的主观性和盲目性。由于计算量很大,手工实现十分困难,我们在 IBM PC 机上开发的“产品结构优化系统”(简称PSES系统)较满意地解决了这个问题。

## 一、产品结构优化的数学模型与结果分析

### 1. 数学模型

产品结构优化的问题可归结为:在  $n$  个车间(或工序)生产  $m$  种产品时,已知各车间的生产能力(工时)、各车间生产单位产品的工时定额、每种产品的最小必须生产量和最大可能销售量、产品的单位利润,如何安排各种产品的产量结构,使企业获得的经济效益最高。

其数学模型为

$$\text{约束条件} \begin{cases} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \dots + a_{1m}x_m \leq b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \dots + a_{2m}x_m \leq b_2 \\ \vdots \\ a_{n1}x_1 + a_{n2}x_2 + a_{n3}x_3 + \dots + a_{nm}x_m \leq b_n \\ x_j \geq D_{j1} \quad (1 \leq j \leq m) \\ x_j \leq D_{j2} \end{cases}$$

$$\text{目标函数} \quad \max(F) = \sum_{j=1}^m C_j \times x_j$$

从上式可以看出,这是一个线性规划问题,

式中:  $a_{ij}$  ( $i: 1 \sim n; j: 1 \sim m$ ) 为第  $i$  个车间(或工序)生产第  $j$  种产品时的单位产品工时定额;  $b_i$  ( $i: 1 \sim n$ ) 为第  $i$  个车间(或工序)的生产能力(有效工时  $\times$  定额完成系数);  $D_{j1}$  和  $D_{j2}$  ( $j: 1 \sim m$ ) 分别为第  $j$  种产品的最小必须生产量和最大可能销售量;  $C_j$  为第  $j$  种产品的单位利润;  $x_j$  ( $j: 1 \sim m$ ) 为欲求的各产品产量。

利用单纯形法可以求出各种产品产量的最优解和企业的最大利润。

在实际运用过程中,有些产品的最小必须生产量和最大可能销售量完全相等(即  $D_{j1} = D_{j2}$ ),因此它们已成为已知量( $D_{j1}$  又称为计划内产量)。对于二者不等的产品,只要求出计划外产品(即  $D_{j2} - D_{j1}$ ) 即可,这样,上述数学模型可简化为:

$$\text{约束条件} \begin{cases} a_{11}y_1 + a_{12}y_2 + \dots + a_{1m}y_m \leq B_1 \\ a_{21}y_1 + a_{22}y_2 + \dots + a_{2m}y_m \leq B_2 \\ \vdots \\ a_{n1}y_1 + a_{n2}y_2 + \dots + a_{nm}y_m \leq B_n \\ y_j \leq D_{j2} - D_{j1} \quad (1 \leq j \leq M) \\ y_j > 0 \end{cases}$$

$$\text{目标函数} \quad \max(F') = \sum_{j=1}^m C_j \times y_j$$

式中:  $y_j$  为第  $j$  种产品的计划外产量;  $B_i$  ( $i: 1 \sim n$ ) 为第  $i$  个车间(或工序)的生产能力减去计划内产品所消耗的生产能力;  $m$  为全部产品数减去计划外产品产量为零的产品数。

在对上式求解的基础上可进行汇总,把求出的最优解  $y_1 \sim y_m$  加上各产品对应的计划内产量,同时列出计划外产品产量为零的那些产品的计划内产量,便得出全部产品的最优结构(即  $x_j$ )。

最大利润  $F = F' + F''$

其中  $F'$  为计划外产量的最大利润,  $F''$  为全部

产品计划内产量的利润。

PSES系统由于采用了简化后的产品结构优化数学模型，在运行中大大减少了时间开销。

2. 结果分析

结果分析包括能力分析 & 利润、产值分析两个方面。

(1) 能力分析 主要指生产计划工时与生产能力之间的对比。

对于第*i*个车间（或工序）来说，生产计划工时  
$$= \sum_{j=1}^m a_{ij} X_j$$
（式中*X<sub>j</sub>*即为求出的各产品的最佳产量）

车间的生产能力是已知的，二者之差即车间的剩余能力，二者之比说明了该车间的负荷情况。

(2) 利润、产值分析 已知某企业的目标利润、目标产值、各种产品的单位售价*E<sub>j</sub>*；产品的最佳产量结构及最大利润已经求出。则：

$$\text{最佳搭配产值} = \sum_{j=1}^m E_j \cdot X_j$$

通过目标利润与最大利润、目标产值与最佳搭配产值的对比，可分析该企业（或车间）在产品结构优化后的利润创收和产值情况。

二、产品结构优化  
系统程序设计

1. 初始单纯形表的形成

初始单纯形表（简化后）的形式如表1所示。  
表1在以后的运算过程中存放在工作区P数组

表1 简化后数学模型的初始单纯形表

列 (j)	1	2	3	.....	M	M+1	M+2	.....	n+2M	n+2M+1	n+2M+2
行 (i)											
1	a <sub>11</sub>	a <sub>12</sub>	a <sub>13</sub>	.....	a <sub>1M</sub>	1	0	.....	0	B <sub>1</sub>	
2	a <sub>21</sub>	a <sub>22</sub>	a <sub>23</sub>	.....	a <sub>2M</sub>	0	1	.....	0	B <sub>2</sub>	
⋮	⋮					⋮				⋮	
n	a <sub>n1</sub>	a <sub>n2</sub>	a <sub>n3</sub>	.....	a <sub>nM</sub>	⋮				B <sub>n</sub>	
n+1	1	0	0	.....	0	⋮				B <sub>n+1</sub>	
n+2	0	1	0	.....	0	⋮				⋮	
⋮	⋮					⋮				⋮	
n+M	0	0	0	.....	1	0	0	.....	1	B <sub>n+M</sub>	
n+M+1	-C <sub>1</sub>	-C <sub>2</sub>	-C <sub>3</sub>	.....	-C <sub>M</sub>	0	0	.....	0	0	

中，其中在*i* = 1~*n* + *M*，*j* = 1~*M*这一部分放约束条件的系数，在*j* = *M* + 1~*n* + 2*M*这一部分放单位矩阵（*M*为*D<sub>i2</sub>* - *D<sub>i1</sub>* ≠ 0的产品个数）。  
程序运行时，首先从数据文件中取出单位产品工时定额*α<sub>ij</sub>*、产品单位利润*C<sub>i</sub>*和产品单位售价*E<sub>j</sub>*，并分别放在A、C、E数组中，A数组和C数组的数即在表一中所示的*a<sub>ij</sub>*和*C<sub>i</sub>*（*i* = 1~*n*，*j* = 1~*M*）所对应的位置，其中舍去了*D<sub>i1</sub>* = *D<sub>i2</sub>*的产品。*B<sub>1</sub>*~*B<sub>n</sub>*为所对应的各车间（或工序）生产能力与计划内产品所耗能力的差值，*B<sub>n+1</sub>*~*B<sub>n+M</sub>*为所对应的*D<sub>i2</sub>*与*D<sub>i1</sub>*的差值（*D<sub>i2</sub>* ≠ *D<sub>i1</sub>*，*j* = 1~*M*），*B<sub>1</sub>*~*B<sub>n+M</sub>*放在B数组中，并均放入P数组*n* + 2*M* + 1列中。表1的其余部分为自动形成，没有赋值的数自动清零。

表1的零列表示约束条件的个数，零行表示对应的计划外产品产量不为零的产品号及引入的松弛变量号。

2. 求最优解

上述初始单纯形表形成以后，首先要判别是否有解。在*i* = 1~*n* + *M* + 1，*j* = 1~*M*范围内，如果有任意一列*k*的值*P*（*i*，*k*）< 0，则该问题无解，否则有解。

如果有解，还要判别此时是否为最优解。如果表中*n* + *M* + 1行中所有的数均不小于零，则此时即为最优解，否则找出该行中为负数且绝对值最大的数，以该数所在的列为主元列*W*；主元列中所有的数除*n* + 2*M* + 1列中所对应的*B<sub>i</sub>*（*i* = 1~*n* + *M*）值，找出商数（如果*P*（*i*，*w*）≤ 0，商数自动用一

个大的正数填充,全部放在P数组 $n+2M+2$ 列)中  
为正数且最小的数,以该数所在的行为转换行S,  
则同时处于主元列和转换行的数为P(S,W),转  
换行中的每个数均乘 $1/P(S,W)$ ,从而使P(S,  
W)数变为1,主元列中剩余的数P(i,W)按照有  
关公式均变 为零,其它的数 也按相应 公式进行变  
换。单纯形表经过上述处理以后,再判别是否为最  
优解。

这个过程循环往复地进行,直至得出最优解,即  
最优的计划外产品结构。

### 3. 最佳产品结构的生成

在上述过程中,求出的最优解存放于y数组,  
而 $D_{i1} = D_{i2}$ 的产品所对应的y数组中的元素为零。

对求出的y数组中的数进行汇总,即每个元素  
加上各自对应的 $D_{i1}$ ,就形成了全部产品最佳生产  
结构,并存入x数组中。

据全部产品的最佳结构,按式 $\max(F)$

$$= \sum_{j=1}^m C_j x_j \text{ 即可计算出最大利润。}$$

## 三、PSES系统功能

### 与工作流程

#### 1. PSES系统功能

PSES在CCDOS支持下运行,在运行中附有  
菜单显示和汉字提示。

PSES的主控程序下有4个子系统供用户调用。

(1)清除原有数据文件 在解决不同的产品结  
构优化问题时,调用本子系统可将原有的数据清除。

(2)建立数据文件 PSES建立的数据文件,  
用于存放各车间每种产品的单位产品工时定额 $a_{ij}$ ,  
产品的单位利润 $C_j$ 和单位售价 $E_j$ 。以避免在运 行  
中大量数据的重复输入。

(3)修改数据文件 PSES提供了对建立的  
数据文件进行修改、删除、添加等功能。

(4)最佳生产产品结构的生成 该子系统是  
PSES最重要的功能模块,它主要完成简化后数学  
模型的原始单纯形表的建立、求最优解、最后汇总  
出产品的最佳结构,打印输出最佳生产产品 结 构  
表,打印输出能力和利润、产值分析表。

#### 2. 工作流程

这里主要给出最佳生产产品结构生成子系统的

程序框图,如图1所示。

其中用单纯形法求最优解的程序框图如图2所  
示。

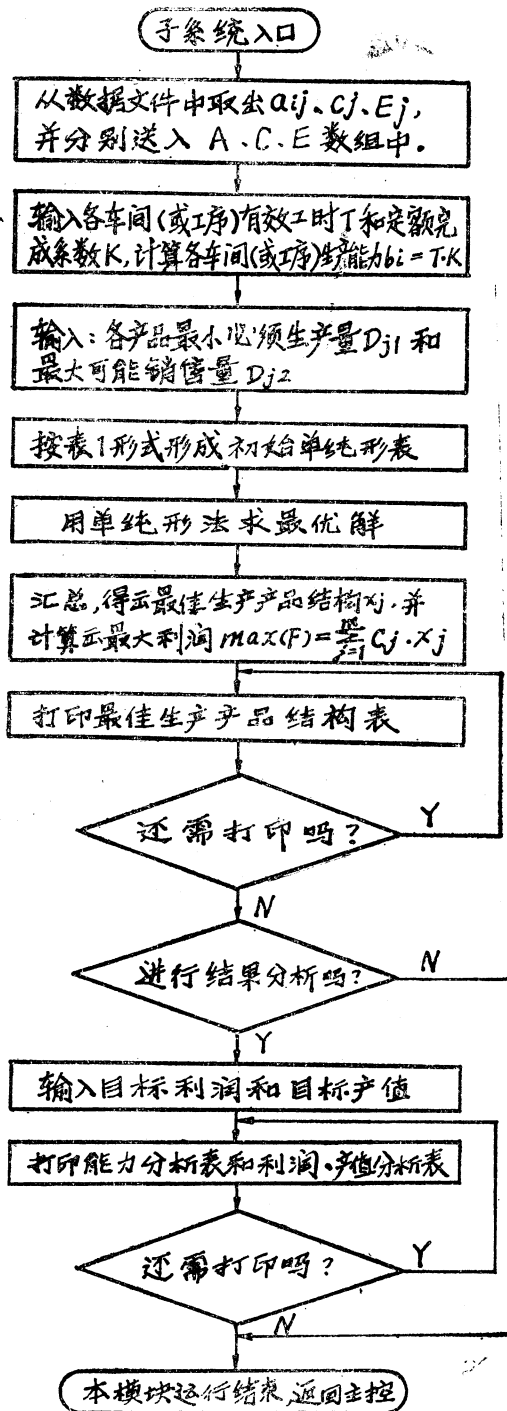


图1 最佳生产产品结构生成子系统



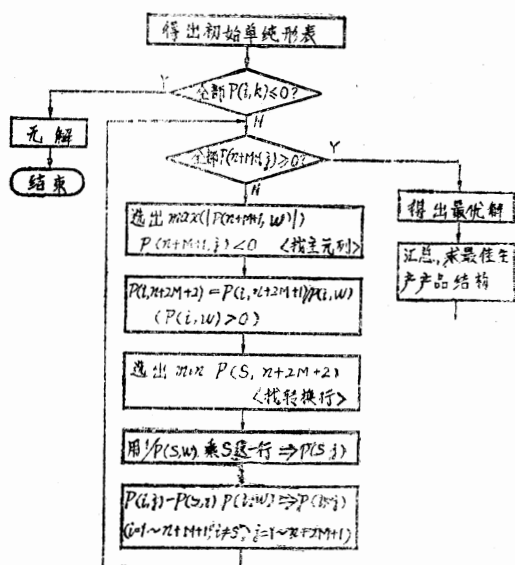


图2 求最优解的程序框图

## 四、应用实例

PSES系统具有操作简便、计算精确、效率高等优点。

无锡某机床厂共计11个车间（其中一个车间有两种指标），生产41种主要产品。根据已知条件建立起来的初始数学模型中，含变量41个，约束条件90多个。对于这样一个复杂的线性规划求解问题，用人工计算存在极大的困难。而采用PSES系统进行计算，仅运行30多分钟即可得到所需的数据和表格。

表2 为PC机输出的该厂的最佳生产产品结构表；表3和表4分别为能力分析表及利润、产值分析表。

PSES系统的推广应用将为企业运用现代化的科学管理方法提高经济效益带来方便。

表2 最佳生产产品结构表

产品号	最少必需生产 (台)	最多可能销售 (台)	最佳生产量 (台)	备注
× (1)	50	100	50	*
× (2)	250	300	299	*
⋮				
× (41)	55	55	55	

备注中\*号表示该产品没有满足市场需求  
全年最大利润 = 1.610613E + 07

表3 能力分析表

车间或 工序号	生产能力 (工时)	生产计划 工 时	剩余能力 (工时)	负 荷 (%)
1	5050	5033	17	99.66
2	155075	155003	72	99.95
3	60996	45071	15925	73.89
4	60000	59109	891	98.51999
5	110160	98322	11838	89.25
6	230000	229859	141	99.94
7	210000	209068	932	99.56
8	294316	279535	14781	94.97999
9	253368	229199	24169	90.46
10	345000	344822	178	99.95
11	117504	105214	12290	89.54
12	67565	46109	21456	68.24

注：第一行的能力单位为吨。

生产能力 = 有效工时 \* 定额完成系数。

表4 利润，产值分析表

目标利润 (元)	最大利润 (元)	目标产值 (万元)	最佳搭配产值 (万元)
1.2E + 07	1.610848E + 07	3850	3888.12

### 说明

按最佳方案生产能达到目标利润。

- (1) 若能力有余说明企业还有潜力。
- (2) 若能力没有富余说明如此生产恰到好处。
- (3) 按最佳搭配生产能完成产值指标。

# 计算机在收购粮食工作中的应用

钱维勤 赵敦玉 钟淑敏

(吉林粮油食品专科学校)

**摘要** 本文以收购粮工作中的重要业务环节——收购粮计算与结算为例介绍了微型机在粮食收购工作中的应用。并简要叙述了“收购粮计算软件”的设计思想、数学模型、操作过程、和给企业带来的明显的经济效益,以及微型计算机在粮食企业应用的广阔前景。

## 一、引言

秋收时节是粮食部门最繁忙的时候,但最忙碌的还要数业务计算员。长期以来计算员以算盘为工具,每天计算上千枚票据,特别是晚上结帐时,需统计每天每个品种各个等级中各项的合计及每个品种各项的总计,和每个品种不同水份粮入库情况。这是一项计算量大而又枯燥的工作。统计工作完成后,还要对帐,若对不上帐,就得在上千枚票据中查找,如大海捞针。为了解决这种现状,我省曾在粮食系统推广使用单板收粮机,但由于该机程序固定,不能方便地随意修改,满足不了目前灵活的收粮政策。为了从根本上解决这一问题,我们在APPLE-Ⅰ机上研制的《收购粮计算汉字应用软件》在实际应用中收到了较好的效果。

有人曾做过统计,业务计算员用算盘计算一张票据平均需3分钟,复核员再复核一遍需1.5分钟,计4.5分钟。而计算机最多用15秒钟。特别是各个项目的合计和整个品种各项的总计,及按水份统计数量等,均可根据需要随时输出,准确无误。若把节省的计算员统计在内,效率就提高几百倍,同时大大改善了工作条件,实现了收粮计算工作的电脑化。

## 二、设计思想及实现过程

收购粮食的计算工作,要依据粮食政策。对农

民送来的各不同品种的粮食进行检斤、验等、检测水分、确定杂质、辨别色泽气味及不完善粒等工作。然后根据这些原始数据确定粮食的单价。单价以标准等级为基础,每增加一等按一定比例加价;每减少一等则按一定比例减价。若超过标准水份按一定比例扣量;若低于标准水份则按一定比例加价。若杂质超过限度按一定比例扣价;若低于限度按一定比例加价。再按规定计算整晒费、运费、利息、补贴。把上述各项综合在一起,计算出应付金额。结算时需将当天收购的各品种粮食的扣量前净重、水份扣量、扣量后净重、金额、补贴款、利息、运费、收回予支款、净付金额等各项汇总,存档备查。

对于上述这样一个品种较多,随机出现计算量大,时间性强的问题,我们在程序设计中,采用了主-子结构的作法。由主程序承担品种选择,数据分类,汇总,和输出结果工作。不同品种的计算工作由不同的子程序来担任,子程序将计算后的各项数值转给主程序输出,同时进行汇总。按照一定的操作可输出需要的各种汇总。

采用这种主-子结构,使程序层次清晰、结构紧凑、运算速度快、并可根据实际需要方便地增删计算各品种的子程序。从而能够很好的适应各种收粮计算工作的变化。

程序的结构框图见图1。

收购粮食计算工作,因质量检验标准不一致,计算上各个品种有一定的差异不能一概而论。这里仅以玉米计算为例,给出其解析模型(参见图2)。

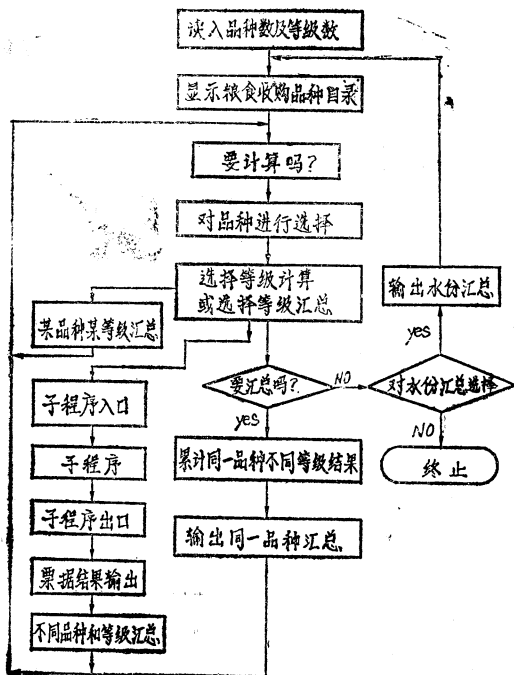


图1 程序结构框图

$$B = \begin{cases} C(1 + (2 - J)0.04)(1 + [W_1/0.5] \cdot \\ \times 0.75/100 + (18 - S)/100) & \text{当 } 1 \leq J \leq Q \\ D \times 0.096/91(1 + [W_1/0.5] \cdot \times 0.75/100 \\ + (18 - S)/100) & \text{当 } Q_1 \leq D \leq 91 \end{cases}$$

$$A = \begin{cases} A = x & \text{当 } S \leq 18 \\ A = (1 - (S - 18)/100)X & \text{当 } S > 18 \end{cases}$$

$$H = \begin{cases} (S - 18) \times 0.3/100 \times A \times B & \text{当 } 18 < S < 28 \\ (S - 18) \times 3/100 \times A \times B & \text{当 } S \geq 28 \end{cases}$$

$$K = (0.0198 + 0.00033X_1) V$$

其中: C为基价, J为等级,  $W_1$ 为杂质数, S为水份数, D为纯粮率, Q为最大等级数,  $Q_1$ 为最低纯粮率, V为尾欠款, B为单价, A为净重, X为输入净重, H为扣整晒费, K为利息扣款(这里计算式以85年规定为准,  $X_1$ 为10月1日到扣尾欠款时的天数)。[J\*为取整函数。]

### 三、使用简介

收购粮计算程序是应用苹果机, 采用 BASIC 语言编写的汉字应用软件。该软件被制成软盘、使用保管极为方便, 用户只要有一套APPLE—II基本系统, 就可利用该软件完成粮食收购的计算工作。

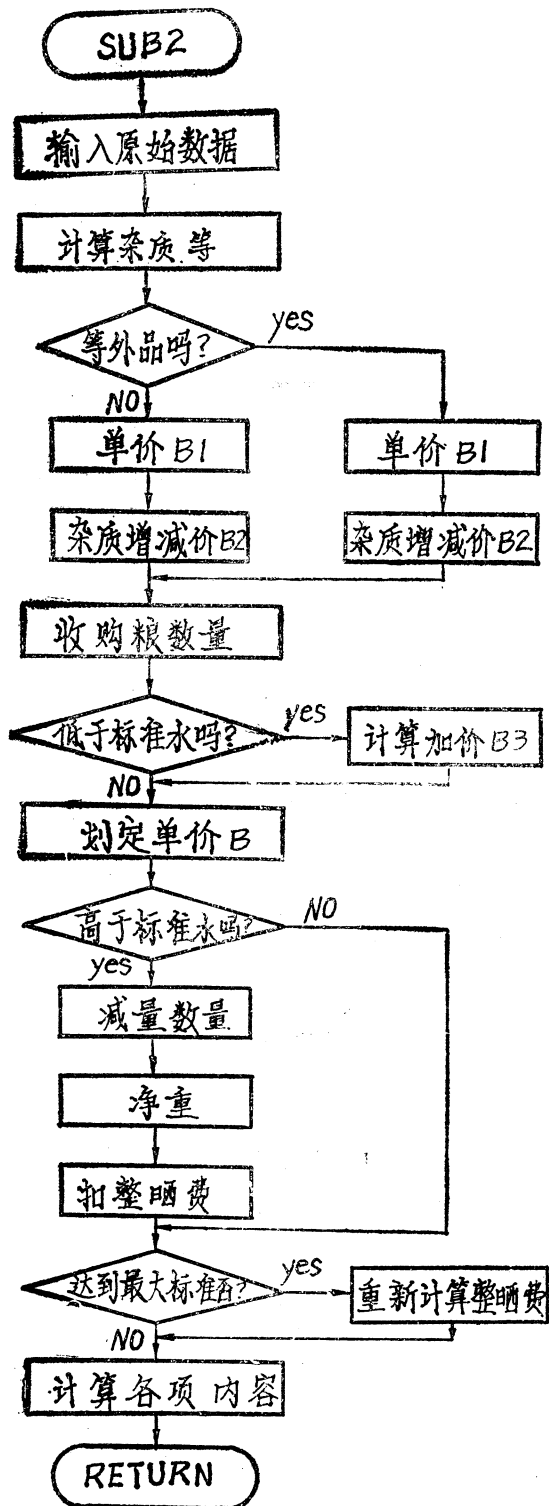


图2 玉米程序实现粗框图

下面我们用法1中的数据说明使用过程。

表 1

玉米	等级	水份	杂质	扣前净重	收回予支款	天数	色泽气味
2	2	23	0.5	1164	15.93	77	0

注：色泽气味正常输入0，不正常输入1。

## 1.对票据中各款项做计算

首先将应用软件放到驱动器中，启动机器。当进入汉字状态后，键入RUN↵(↵代表RETURN键，下同)，这时荧屏上显示：

- 1 玉米民代国储      2 玉米进货计算
- 3 高粱进货计算      4 谷子进货计算
- 5 糜子进货计算      6 小麦进货计算
- 7 稻谷进货计算      8 大豆进货计算

你要计算吗 (Y/N) ?

如果你要计算就键入Y↵，这时荧屏显示：

品种数 = ?

这是问你现在需要计算的是什么品种粮，如果要计算玉米进货，就由键盘敲入2↵；如果要计算小麦就键入6↵；现在我们键入2↵这时荧屏显示：

等级 = ?

这是问你进货的玉米是几等的，按照表1我们敲入等级数2↵，这时荧屏显示：

水份，杂质，扣前净重，收回予支款，天数，色泽气味 = ?

这是向你询问上述各个项目的具体数值。有的就由键盘敲入实际数值，没有的就在相应的位置敲入0。由表1我们依次敲入23，0.5，1164，15.93，77，0↵。这时荧屏显示：

扣量前净重 = 1164

水份扣量 = 58

扣量后净重 = 1106

单价 = .136

金额 = 150.42

补贴 = 2.21

利息 = .72

扣整晒费 = 2.26

收回予支款 = 15.93

净付金额 = 134.83

同时问你：

你要修改吗 (Y/N) ?

这是让你检查一下输入的原始数据有无错误，若有错误键入Y↵，则返回重新计算；若无错误则键入N↵，表示不用修改可继续计算。这时荧屏显示：

你要计算吗 (Y/N) ?

如果还要计算其它品种的粮食可重复上面的操作过程，就可得到相应的结果。

## 2.对某一品种不同等级分别汇总

结算时，要计算某一品种不同等级的汇总，可按下面的操作步骤进行。当荧屏显示：

你要计算吗 (Y/N) ?

假如你要进行玉米各个等级的分别汇总可键入Y↵，这时荧屏显示：

品种数 = ?

键入2↵，荧屏显示：

等级数 = ?

键入-J↵(J=1, 2...。这里用-J代表相应的等级汇总指令)。假如我们键入-2↵，这时荧屏便显示出到目前为止所有的2等玉米各项数据汇总，如要汇总其它等级，可以重复以上步骤，即可得到相应的输出。

## 3.对同一品种的各项汇总

如果现在要计算每个品种的各项汇总，在荧屏显示：

你要计算吗 (Y/N) ?

时，可键入N↵，荧屏显示：

你要汇总吗 (Y/N) ?

键入Y↵，即可得到每个品种的各项汇总数据。

## 4.按水份统计粮食入库量

如果你想知道到目前为止某一品种粮某种水份含量的粮入库数量情况。当荧屏显示：

你要汇总吗 (Y/N) ?

时，可键入N↵，这时荧屏显示：

你要计算品种——水份吗 (Y/N) ?

键入Y↵，荧屏显示：

品种数，水份数 = ?

这是问你了解的是什么品种数的粮食？多少水份含量？你可以依次键入相应的数值，这时荧屏给你计算出相应的数值。同时询问你是否继续了解其它水份含量粮食的入库数量情况，若了解可重复上述过程，若不了解可在输入水份数值时键入100↵，

# 计算机在能源审计中的应用

张更明

(济南汽车制造总厂企管处电算站)

随着我国经济体制的改革,节能和能源管理工作的不断加强,能源管理正朝着靠信息、靠经济手段、靠科学进步的现代化方向发展。

能源审计是能源管理的一个部分,是企业用能的一种宏观统计分析方法。其主要内容是以企业的二级能源计量为基础,计算不同层次的产品能耗指标和节能量指标,分析企业现状,评价企业能源利用水平;通过企业能源审计工作,寻找降低消耗,提高能源利用率的途径,为企业进行能源利用,设备的更新改造提供依据,从节能的角度上帮助企业寻求最佳的运行方式和最优的工艺路线。

企业的能源审计工作是一项经常性、重复性的工作,审计过程中要收集、处理并保存大批数据。为了解决审计中复杂的数据处理,我们在PC/XT微机上,用Cdbase—Ⅱ开发了能源审计数据处理系统,提高了数据处理的准确性和快速性,为促进企业节约能源,提高能源管理水平打下良好的基础。

本系统由系统初始化,数据输入,数据处理,报表输出等功能组成。采用模块化结构,操作灵活,维护方便,通用性强。能满足一般企业的不同

要求。

下面就本系统的能源实物平衡关系分类和各模块的功能分别加以论述。

## 一、实物平衡关系分类

企业消费的能源有购入能源和转换能源,分别被产品生产、能源转换、辅助生产和非生产部门耗损。在本系统中,购入能源有15项,转换能源有6项,每一项表示的能源由用户在使用前决定,各部门所消耗的能源数据分为三部分:

**1.基本系统** 它反映了企业能源总的收支情况。包括期初库存,期内输入,期末库存,期内输出,辅助、非生产消耗及损耗,净消费量等。

**2.能源转换系统** 即转换能源的消费量。它由每一种转换能源所消耗的购入能源和转换能源组成。

**3.产品生产系统** 即生产过程中消耗的能源。它由每一种产品所消耗的购入能源和转换能源组成。

这时荧屏重新显示粮食收购品种目录。

### 5.脱机

当一切计算完毕需脱机时,在荧屏显示:

你要计算品种——水份吗(Y/N)?

时,键入N↵,即可脱机,计算工作结束。

## 四、结束语

《收购粮计算汉字应用软件》是为了便于基层

粮库业务计算员和管理人员学习掌握,而采用普及型高级语言——BASIC语言编写的,程序设计力求按照计算人员的思考和工作方式形成块状结构。一个没有学过微机知识的初、高中毕业生,只需短期培训就可掌握程序的结构思想,并能随心所欲的修改子程序,以满足收粮计算工作的需要。





### 品生产系统的消耗

每次输入数据时,操作人员只需输入 期 内 输入, 期内输出, 辅助、非生产消耗和损耗的各项能源数据, 其它各项, 系统将根据它们之间的相互关系自动生成。

(2)能源转换系统的数据输入。收集各种转换能源所消耗的购入能源及转换能源的数据, 转换能源最多有 6 种。

(3)产品生产系统的数据输入。收集各种产品的能源消耗数据及产品产量。产品种类不限。

**3. 数据处理模块** 审计中各种数据之间的联系是复杂的, 因而数据处理是能源审计中工 作 量 最大、计算最复杂的一个部分。

在本系统中, 我们将转换系统的实物能耗数据作为一个矩阵  $Z$ 。

$$Z = \begin{pmatrix} Z_{11} & Z_{12} & Z_{13} \cdots Z_{1m} & Z'_{11} \cdots Z'_{1n} \\ Z_{21} & Z_{22} & Z_{23} \cdots Z_{2m} & Z'_{21} \cdots Z'_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ Z_{n1} & Z_{n2} & Z_{n3} \cdots Z_{nm} & Z'_{n1} \cdots Z'_{nn} \end{pmatrix}$$

$$Z_{nm} = \begin{pmatrix} Z_{11} & Z_{12} & Z_{13} \cdots Z_{1m} \\ Z_{21} & Z_{22} & Z_{23} \cdots Z_{2m} \\ \vdots & \vdots & \vdots \\ Z_{n1} & Z_{n2} & Z_{n3} \cdots Z_{nm} \end{pmatrix}$$

$$Z_{nn} = \begin{pmatrix} Z'_{11} & Z'_{12} \cdots Z'_{1n} \\ Z'_{21} & Z'_{22} \cdots Z'_{2n} \\ \vdots & \vdots \\ Z'_{n1} & Z'_{n2} \cdots Z'_{nn} \end{pmatrix}$$

其中,  $Z_{nm}$ 是转换系统消耗的购入能源, 它是一个  $n \times m$  阵。 $Z_{nn}$ 是转换系统消耗的转换能源和产生的转换能源, 它是一个  $n \times n$  阵。

令购入能源和转换能源的折标系数为一个  $R$  列矩阵。

$$R = \begin{pmatrix} r_1 \\ r_2 \\ \vdots \\ r_m \\ r'_1 \\ \vdots \\ r'_n \end{pmatrix}$$

其中  $r_1 \sim r_m$  是购入能源的折标系数。

$r'_1 \sim r'_n$  是转换能源的折标系数。

则  $Z \cdot R = 0$

解上列矩阵方程, 求得各种转换能源的折标系数。

用  $N$  表示转换能源折购入能源的实物系数, 它是一个  $n \times m$  矩阵。

$$N = \begin{pmatrix} n_{11} & n_{12} & \cdots n_{1m} \\ n_{21} & n_{22} & \cdots n_{2m} \\ \vdots & \vdots & \vdots \\ n_{n1} & n_{n2} & \cdots n_{nm} \end{pmatrix}$$

则  $Z_{nn} \cdot N = Z_{nm}$

解上列矩阵方程, 便求得各种转换能源折购入能源的实物系数。

用  $F$  表示辅助生产、非生产的能耗和损耗数据, 它是一个  $m+n$  阶行矩阵; 用  $F_t$  表示折进厂能源实物分摊系数, 它是一个  $m$  阶行矩阵。

$$F = [f_1, f_2, f_3, \cdots f_m, : f'_1, f'_2, \cdots f'_n]$$

$$F_m = [f_1, f_2, \cdots f_m]$$

$$F_n = [f'_1, f'_2, \cdots f'_n]$$

$F_m$  为消耗的购入能源,  $F_n$  为消耗的转换能源。

$$F_t = [f_{t1}, f_{t2}, \cdots f_{tm}]$$

$$\text{则 } F_s = F_m + F_n \cdot N$$

$F_s$  是辅助、非生产消耗的各种折算购入能源, 它是一个  $m$  阶行矩阵。

$$f_{ti} = \frac{f_{si}}{e - b} \quad i = 1 \sim m$$

$f_{si}$  是辅助、非生产能耗折第  $i$  种购入能源的实物量。 $e$  是企业总综合能耗, 即净消费量折标量之和。 $b$  是辅助、非生产能耗的折标量之和。

折标煤的分摊系数为:

$$f = \frac{b}{e - b}$$

设产品生产系统的实物能耗数据是一个  $P \times n$  矩阵  $G$ ,  $P$  为产品的种类。

$$G = \begin{pmatrix} g_{11} & g_{12} & g_{13} \cdots g_{1m} & g'_{11} & g'_{12} \cdots g'_{1n} \\ g_{21} & g_{22} & g_{23} \cdots g_{2m} & g'_{21} & g'_{22} \cdots g'_{2n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ g_{p1} & g_{p2} & g_{p3} \cdots g_{pm} & g'_{p1} & g'_{p2} \cdots g'_{pn} \end{pmatrix}$$

对  $G$  矩阵每行分别除以各自的产量, 则得 实物单耗矩阵  $D$ 。

$$D = \begin{pmatrix} d_{11} & d_{12} & d_{13} & \cdots & d_{1m} & d'_{11} & d'_{12} & \cdots & d'_{1n} \\ d_{21} & d_{22} & d_{23} & \cdots & d_{2m} & d'_{21} & d'_{22} & \cdots & d'_{2n} \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ d_{p1} & d_{p2} & d_{p3} & \cdots & d_{pm} & d'_{p1} & d'_{p2} & \cdots & d'_{pn} \end{pmatrix}$$

其中  $d_{ij} = \frac{g_{ij}}{p_i}$   $j = 1 \sim m+n$   
 $i = 1 \sim p$

则产品的生产比能耗

$$H_s = \begin{pmatrix} h_{s1} \\ h_{s2} \\ \vdots \\ h_{sp} \end{pmatrix} = D \cdot R$$

企业综合比能耗

$$H_q = \begin{pmatrix} h_{q1} \\ h_{q2} \\ \vdots \\ h_{qp} \end{pmatrix} = H_s \cdot (1 + f)$$

将D矩阵分为产品生产过程中消耗的购入能源矩阵D<sub>g</sub>和转换能源矩阵D<sub>z</sub>。则产品生产系统消耗的折购入能源实物单耗为:

$$M = D_g + D_z \cdot N + H_s \cdot F_t$$

节能量计算:

(1) 生产系统的节能量

某种产品的产品节能量 = (基期产品比能耗 - 对比期产品比能耗) × 产品产量

(2) 企业的节能量

某种产品的企业节能量 = (基期企业比能耗 - 对比期企业比能耗) × 产品产量

数据处理过程中, 需要处理大量的矩阵运算, 而PC/XT微机上没有发现有能进行矩阵运算的语言。我们通过分析Cdbase-Ⅱ的功能, 编制了一个命令文件, 解决了数据处理过程中的矩阵运算问题, 也提高了数据处理的速度。

**4. 报表输出模块** 它是系统的最后一部分, 输出四种报表, 即企业能源消费平衡表, 转换系统比能耗表, 生产系统能耗表, 包括实物单耗、折标量、折标单耗、生产比能耗、企业比能耗等数据, 和节能量计算表。这四种表基本上描述清楚了企业的能源消费情况和能源利用水平。

以上介绍的是计算机在能源管理中的基本应用, 对于企业合理使用能源, 降低能源消耗, 提高管理水平, 起到了促进作用。并有一定的使用和推广价值。

## 《微小型计算机开发与应用》征稿

本刊是电子计算机专业的中级技术性刊物。重点介绍国内外微、小型计算机的实用技术, 最新产品和发展动态。

根据国内外电子计算机的发展状况, 本刊特征集与微、小型电子计算机开发与应用有关的各类技术文章。征稿内容如下:

1. 中华学习机: 包括硬件; 软件及各种典型的应用程序等;
2. 四代机技术: 包括四代机体系结构; 第四代语言; 第四代操作系统及四代机网络等技术。

# DBASE II 与高级语言关系的探讨

林坚 于京平

(冶金部钢铁研究总院计算机室)

**摘要** 本文通过对数据库DBASE II系统特点和功能的分析,初步探讨了DBASE II在应用控制系统中的作用,解决了高级语言与数据库DBASE II之间的通讯问题,提出了利用DBASE II系统实现高级语言程序覆盖的方法。这对于扩大DBASE II的应用范围有着普遍的意义。

## 一、前言

数据库是近年来发展极为迅速的计算机软件,是数据信息管理的最新技术,它广泛应用于国民经济、文化教育、军事情报、科学研究、人工智能以及计算机辅助设计等领域,为计算机应用开辟了广阔的天地。

我国微型机上较为流行的数据库管理系统就是DBASE II。DBASE II系统适用于仓库管理、工资管理、账目、档案、报表、小型图书资料的查询与检索以及中、小型银行储蓄系统等。但是,由于DBASE II系统的命令功能有限,计算能力差以及计算速度较慢等因素,使得其应用范围受到限制。要使得DBASE II在应用控制系统中充分发挥其作用,就迫切需要寻找一种方法来解决DBASE II系统计算能力差的问题,为了达到这一目的,我们自然会想到将应用控制系统中的应用程序用高级语言(如: BASIC, FORTRAN, COBOL, PASCAL, PL/I等)进行编写,而后与DBASE II相结合形成一个完整的系统。实现这种方法的主要问题在于如何使高级语言与DBASE II系统进行有效的数据通讯。解决了这个问题,我们便可以把高级语言与DBASE II系统有机地结合起来,从而实现各种应用控制系统,并且将其应用于实际的现场模拟、控制、数据处理或精化,反映实际现场的数学模型等领域。这对于扩大DBASE II在工程上的应用范围有着重要的意义。

## 二、DBASE II 系统在实际应用控制系统中的作用

我们知道,设计数据库保存大量数据的目

仅仅是扩充人们的记忆,更主要的是依靠这些数据去帮助人们控制与之相关的事物,因此,数据库往往不是孤立存在的,在它的周围通常存在许多应用程序,它们一起便构成了完整的应用控制系统。

根据DBASE II现有的命令功能以及有限的计算能力,利用DBASE II独立实现一个庞大的应用控制系统是极为困难的,但是,如果应用程序采用高级语言编写,使得复杂的计算都由高级语言程序完成,而DBASE II则实现对数据的有效管理,并为应用程序提供必要的数据资源,那么实现这样一个应用控制系统将是完全可能的。图1表示了DBASE II系统与其它各部分之间的关系:

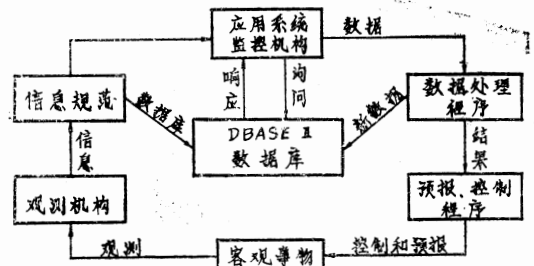


图1

从图1中我们看到,人们可以通过对客观事物的观测,从中得到大量的原始信息,而后对这些原始信息进行记录、整理和归类(统称为规范),最后将信息数据化并存入DBASE II系统中保存起来,其中一部分信息可能直接送入监控机构进行处理。另一方面,监控机构向数据库DBASE II系统发出询问,并从DBASE II中取出数据进行处理,在处理过程中可能产生新的数据要求保存,则可再次送入DBASE II系统中。数据处理后的结果可用于预报和控制相应的客观事物,此时通过预报、控制程序来加以实现。

监控机构与DBASE II的关系主要是询问和响

应,其结构较为简单,但是,DBASE II与监控机构、信息规范、数据处理的关系都归结为应用程序与DBASE II的数据传递关系。

### 三、DBASE II与高级语言程序的数据关系

DBASE II与它的关系实际上是数据的通讯关系,它解决了DBASE II与高级语言之间的数据传递,也就是说,在DBASE II中建立的数据文件如何在高级语言环境下得到应用,同样,在高级语言程序中经过数据处理以后产生的新数据又如何存入DBASE II系统中且被DBASE II系统有效管理。下面就这两个问题进行分析:

#### 1. DBASE II系统中文件的特点

文件是存贮在某种设备中的信息的集合。DBASE II系统中的文件都是标准的CP/M文件,它有以下几种类型:

数据文件: (•DBF);  
存贮文件: (•MEM);  
文本输出文件: (•TXT);  
命令文件: (•CMD);  
报表格式文件: (•FRM);  
索引文件: (•NDX);  
格式文件: (•FMT);

其中本文讨论的主要是数据库的数据文件(•DBF)和文本输出文件(•TXT)。数据库的数据文件由结构记录以及0~65535个数据记录组成。其中结构记录实际上是数据记录格式图。文件中最多含有32个不同的数据场(也称为字段),文件的记录结构包括数据场名、数据场中的数据类型,数据场的大小和记录中数据的位置。DBASE II系统中的命令可直接对此文件进行插入、删除、修改、检索、索引和排序等操作,而对于文本输出文件,DBASE II系统则无法对它直接进行上述操作。但是,可以把它转换成数据库文件(•DBF)处理,文本输出文件可作为DBASE II的“外部”文件而被高级语言程序所使用。

#### 2. 高级语言读取DBASE II数据文件的方法

从DBASE II文件特点看出,高级语言程序是可以读取DBASE II系统中数据文件记录的,但不能直接读取•DBF文件中的记录,因为,•DBF文

件记录是以内部形式(ASCII码)存放的,对于高级语言程序这个“外部程序”来说,它都属于字符型,读出的数据也都是字符型的,这样,若要对其中数值型数据进行数值计算处理时,还须进行必要的转换,这对于用户来说是极为不方便的。那么要想正确读取文件记录,只有将•DBF文件先转换成•TXT文件,因为•TXT文件是DBASE II的外部输出文件。其关系如图2所示:

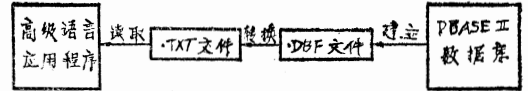


图2

从图2中看出,实现的方法是:先在DBASE II系统中利用CREATE命令建立一个数据库文件•DBF,同时在DBASE II系统下对它实施各种操作,最后利用COPY命令将•DBF文件转换成文本输出文件•TXT。其命令格式为:

COPY TO <文件名> SDF

执行这条命令后便可自动生成名为“<文件名>•TXT”的文本输出文件,此后,高级语言程序便可对此文件进行读取操作。

注意事项:

(1) 高级语言读取•TXT文件时,其结构必须与•DBF中规定的结构完全一致,否则可能产生错误。

(2) •TXT文件记录长度为128个字符,从中还要扣除2个字符作为换行码,实际长度为126个字符,因此建立的•DBF文件其记录长也应为126个字符,剩余位要求用一个空变量补足,否则将产生读取的混乱。

(3) 对于一个较大的文件,可将其分解成若干文件进行处理。

如在DBASE II系统中建立一个名为EXAD-ATA1.DBF的数据库数据文件,其结构如表1,

表1

数据场名	类型	宽度	小数位数
NAME	C	20	
AGE	N	3	
SEX	C	4	
MARK	N	5	1
NCHR	C	94	

其中NCHR为一个空变量,其用途是为保证.TXT文件的记录长为126个字符;要求利用FORTRAN语言将其读出并打印。

DBASEⅡ的命令程序: 建立一个数据库数据文件EXADATA1.DBF且把它转换成文本输出文件(.TXT)为EXADATA1.TXT;

程序清单: EXADBF1.CMD  
CREATE EXADATA1

USE EXADATA1  
APPEND  
COPY TO EXADATA1 SDF  
QUIT

FORTRAN 读数程序: 打开文件EXADATA1.TXT, 并读出文件中的所有记录, 同时利用打印机输出其结果。

程序清单: EXAFOR1.FOR (DATAMAX=8000 FORTRAN IV)

```
PROGRAM EXAFOR1
LOGICAL NAME (20), NCHR (94), SEX (4)
INTEGER AGE
REAL MARK
CALL OPEN (5, 'EXADATA1.TXT', 1)
WRITE (2, 5)
5  FORMAT (1X, ' ', ' ')
   WRITE (2, 10)
10  FORMAT (1X, '|-----|-----|-----|-----|',
*        ' ', '-----| ')
   WRITE (2, 20)
20  FORMAT (1X, '| 编 号 |', ' ', '姓 名', ' ', '年 龄 | 性 别 |',
*        ' ', '成 绩 | ')
   WRITE (2, 10)
   I = 1
30  READ (5, 40, REC = I, END = 60) (NAME (L), L = 1, 20), AGE, (SEX (L),
*        L = 1, 4), MARK, (NCHR (L), L = 1, 94)
40  FORMAT (20A1, I3, 4A1, F5.1, 94A1)
   WRITE (2, 50) I, (NAME (L), L = 1, 20), AGE, (SEX (L), L = 1, 4), MARK
50  FORMAT (1X, ' | ', I1, ' ', ' ', 20A1, ' | ', I3, ' ', ' ', 4A1, ' | '
*        ' ', F5.1, ' | ')
   WRITE (2, 10)
   I = I + 1
   GOTO 30
60  WRITE (2, 70)
70  FORMAT (1X)
END
```

EXAFOR1.FOR程序的执行结果为:

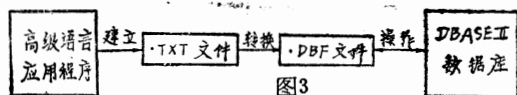
编 号	姓 名	年 龄	性 别	成 绩
1	林韦栋	22	男	99.0
2	于萃芝	23	女	98.5
3	林京平	24	女	96.0
4	李 杉	35	男	86.5

3.DBASEⅡ系统对高级语言程序所建立的

数据文件进行管理的方法

在DBASEⅡ系统下管理由高级语言程序建立的数据文件实际上是上述问题的逆过程。因为, DBASEⅡ系统无法直接对文本输出文件(.TXT)进行操作, 而高级语言程序也无法直接建立.DBF文件供DBASEⅡ系统管理, 因此, 同样也要进行

文件的转换。其关系如图3所示:



由图3可知,实现的方法是:先在高级语言程序中建立一个数据文件,其扩展名为.TXT,同时输入必要的数 据,而在DBASE II系统中先建立一个数据库文件.DBF的文件结构,其结构必须与高级语言程序中的文件结构完全一致,而后在命令程序中利用DBASE II系统的命令 APPEND 将此.TXT文件中的所有数据逐个 加入到 .DBF 文件中。最后便自动生成了.DBF文件,此时DBASE II便可对此数据库文件施行各种操作了。把.TXT文件中的数据自动添加到.DBF文件中的命令为:

## APPEND FROM 文件名.TXT SDF

转换后的目标文件名为“文件名.DBF”。

注意: .TXT文件与.DBF文件的结构必须完全一致,且.TXT文件记录长度为126 个字节,不足的话,则要用一个空变量补满。若记录太长,则要求将文件分解成若干个文件来处理,各文件之间可用记录号进行连接。

例如,在高级语言程序中建立一个名为EXADATA2.TXT的数据文件,其文件结构与表(1)相同,并利用DBASE II系统中的命令程序将此文件转换成EXADATA2.DBF文件,同时对其进行检索操作,打印出所有记录数据。

FORTTRAN建文件程序:建立数据文件EXADATA2.TXT并输入数据,最后写入EXADATA2.TXT文件中。

程序清单: EXAFOR2.FOR ( DATAMAX—8000 FORTRAN IV)

```
PROGRAM EXAFOR2
LOGICAL NAME (20) , NCHR (94) , SEX (4)
INTEGER AGE
REAL MARK
DO 1 I=1, 94
1  NCHR (I) = 1H
CALL OPEN (5, 'EXADATA2TXT', 1)
10 WRITE (1, 15)
15 FORMAT (1X, 'NUMBER: ')
READ (1, 25) I
25 FORMAT (I1)
IF (I.EQ.0) GO TO 110
WRITE (1, 20)
20 FORMAT (1X, 'NAME (1--20): ')
READ (1, 30) (NAME (L) , L=1, 20)
30 FORMAT (20A1)
WRITE (1, 40)
40 FORMAT (1X, 'AGE: ')
READ (1, 50) AGE
50 FORMAT (I3)
WRITE (1, 60)
60 FORMAT (1X, 'SEX (1--4): ')
READ (1, 70) (SEX (L) , L=1, 4)
70 FORMAT (4A1)
WRITE (1, 80)
```



```

80  FORMAT (1X, 'MARK : ')
    READ (1, 90) MARK
90  FORMAT (F5.1)
    WRITE (5, 100, END=110) (NAME (L) , L=1, 20) ,AGE, (SEX (L) ,L=1,4),
    *      MARK, (NCHR (L) , L=1, 94)
100  FORMAT (1X, 20A1, I3, 4A1, F5. 1, 94A1)
    GO TO 10
110  ENDFILE 5
    END

```

DBASEⅡ的命令程序：把EXADATA2.TXT文件转换成EXADATA2.DBF文件，对EXADATA2.DBF文件进行检索操作并打印出所有记录数据。

程序清单：EXADBF2.CMD

```

CREATE EXADATA2
USE EXADATA2
APPEND FROM EXADATA2. TXT SDF
ERASE
SET FORMAT TO PRINT
STORE 1 TO I
STORE 5 TO M
STORE 6 TO N

```

```

@ 1, 10 SAY " "
@ 2, 10 SAY " | "
@ 3, 10 SAY " | "
@ 4, 10 SAY " | "

```

编 号	姓 名	年 龄	性 别	成 绩
-----	-----	-----	-----	-----

```

GOTO 1
DO WHILE .NOT.EOF ()
@ M, 10 SAY " | "
@ M, 14 SAY I USING '9'
@ M, 18 SAY " | "
@ M, 19 SAY NAME USING 'XXXXXXXXXXXXXXXXXXXXXXX'
@ M, 39 SAY " | "
@ M, 43 SAY AGE USING '99'
@ M, 47 SAY " | "
@ M, 49 SAY SEX USING 'XXXX'
@ M, 55 SAY " | "
@ M, 57 SAY MARK USING '999.9'
@ M, 63 SAY " | "
@ N, 10 SAY " | ----- | ----- | ----- | ----- | "
STORE M+2 TO M
STORE N+2 TO N
STORE I+1 TO I
SKIP
IF EOF ()
    EXIT

```

```

ENDIF
ENDDO
STORE N+1 TO N
@ N, 10 SAY " "
RETURN

```

DBASE I 命令程序的执行结果为:

编 号	姓 名	年 龄	性 别	成 绩
1	杨荣堡	45	男	93.0
2	林 力	23	男	91.5
3	林非平	20	女	94.0
4	宋力薇	25	女	86.5

#### 四、利用DBASE II 实现 程序覆盖的方法

覆盖技术主要是用于程序大而内存小的 问题中,使得大程序能在小内存上运行,从而提高内存的利用率。覆盖的概念就是一个程序的若干个程序段或几个程序的某部分共享某段存储空间。目前,出现了许多实现程序覆盖的软件方法,在此将介绍一种利用DBASE I 系统实现程序覆盖 的软件方法。

这种方法是根据 DBASE I 中的QUIT 命令特点进行的。QUIT命令的完整格式为:

QUIT [TO <COM文件表>]

其功能是关闭所有的数据库文件,命令文件或其他类型的文件,回到操作系统状态,若含有TO短语,则所有 <COM文件表> 中的程序将在CP/M下顺序执行,这个功能使你在退出 DBASE I 系统的同时,还能链接其它软件。对于链接程序或软件的个数并无限制,只要CP/M命令不超过254 个字符即可。

根据这个特点,我们便可实现编译型高级语言的程序覆盖。实现这种程序覆盖的方法是先将程序按功能分类并且按它的程序形式进行独立编写,编译和链接,并用SAVE命令形成各自的 .COM 文件存入磁盘中,而后利用DBASE I 系统中的命令编写一个监控程序,利用QUIT命令分别调用各个高级语言程序的.COM文件运行,以此达到覆盖的目的。运行完成后便继续返回监控命令程序中。当然,整个系统在运行前要先调入DBASE I 系统,同时在DBASE I 状态下运行监控命令程序。即可在操作系统状态下输入:

A>DBASE< <监控程序文件名>

这样可直接进入监控程序状态。

现以FORTRAN IV语言为例,假定三个应用程序分别为:

PROGRAM1.FOR:

```

PROGRAM PROGRAM1
WRITE (1, 10)
10 FORMAT (1X, ' PROGR-
AM-----1' )
END

```

PROGRAM2.FOR:

```

PROGRAM PROGRAM2
WRITE (1, 10)
10 FORMAT (1X, 'PROGR-
AM-----2' )
END

```

PROGRAM3.FOR:

```

PROGRAM PROGRAM3
WRITE (1,10)
10 FORMAT (1X, 'PROGR-
AM-----3' )
END

```

监控命令程序则为: EXADBF3.CMD

```

SET FORMAT TO SCREEN
DO WHILE .T.
ERASE
STORE " " TO N
@ 6, 15 SAY " . . . . INFORMATION

```

ANAGEMENT SYSTEM . . . . "

```

@ 8, 25 SAY "1 : PROGRAM1"
@ 10, 25 SAY "2 : PROGRAM2"
@ 12, 25 SAY "3 : PROGRAM3"
@ 14, 25 SAY "0 : SYSTEM STOP"
@ 22, 2 SAY "PLEASE ENTER BLOCK'S NUMBER : "GET N PICTURE 'X'
READ
DO CASE
CASE N="1"
    QUIT TO "PROGRAM1", "DBASE EXADBF3"
CASE N="2"
    QUIT TO "PROGRAM2", "DBASE EXADBF3"
CASE N="3"
    QUIT TO "PROGRAM3", "DBASE EXADBF3"
CASE N="0"
    QUIT
ENDCASE
ENDDO

```

---

运行前先要对PROGRAM1, PROGRAM2, PROGRAM3程序分别进行编译、链接并形成各自的COM文件PROGRAM1.COM, PROGRAM2.COM, PROGRAM3.COM,而后在操作系统状态下输入: DBASE EXADBF3,便可进入监控程序的菜单系统,此时要求用户输入模块号。

(1) 若输入1 则退出DBASEⅠ执行PROGRAM1程序,此时屏幕将显示“PROGRAM.....1”,程序执行完后将返回监控程序的菜单系统中。

(2) 若输2 执行PROGRAM2 程序;而后

返回监控程序的菜单系统中。

(3) 若输3 执行PROGRAM3程序;而后返回监控程序的菜单系统中。

(4) 若输0 则停止执行,返回操作系统中。这样就实现了程序覆盖功能。

上述介绍的DBASEⅠ系统与高级语言的各类关系均已得到实现,并在工作中已得到应用,效果良好。由于实现了上述各类关系,人们便可以利用DBASEⅠ系统去设计各种的应用控制系统,这样就扩大了DBASEⅠ在工程上的应用范围。因此,这种将DBASEⅠ系统与高级语言相结合的方法具有较高的使用和推广价值。

# 微机应用中被删除文件的恢复

崔秀起

(北京大学)

**摘要** 应用微机时,由于不慎误删磁盘文件的情况是时有发生。怎样恢复因不慎而误删的文件是许多用户非常关心的问题,本文介绍一种简单易行的方法。

## 一、原理概述

为了恢复被删除的文件,首先要了解文件被删除以后磁盘上信息的变化。为此,先介绍一下一般软磁盘的文件组织。

一般软磁盘系统存在两个基本文件:文件目录和文件分配表(FAT)。操作系统对软磁盘上文件的建立、修改、删除都是以文件目录表和FAT作为依据的。一般说来,恢复软磁盘上被删除的文件,也必须从这两个文件入手。下面以应用较广的IBM-PC微型计算机及其兼容机为例说明。

当一个软磁盘用FORMAT命令格式化时,磁盘被格式化为如图1所示的五个区域。删除文件和恢复文件的信息变化就在FAT和文件目录中。

引导记录区
FAT第一拷贝区
FAT第二拷贝区
目录区
数据区

图 1

### 1. 文件目录

整个磁盘上的所有文件目录项形成一个文件放在磁盘的目录区中。每一个文件目录项,由32个字节(0~31)组成。它记录了一个文件的文件名(0~7字节)、扩展名(8~10)、文件属性(11)、文件建立时间(22、23)、文件建立日期(24、25)、文件起始簇号(26、27)及以字节表示的文

件长度(28、31)。其中第0字节表明了该文件目录项的当前状态,表示如下:

00H——表示这个目录项从未被使用过。

E5H——表示这个目录项曾被使用过,但该文件目录目前已被删除。

2EH——表示这个目录项是一个子目录。

若为其他字符,则是文件名第一个字符的ASCII码。(字节数码后的字符H表明该数码是以十六进制表示的数,以下内容中意义相同)

### 2. 文件分配表FAT

FAT是磁盘操作系统DOS用来为文件分配磁盘存储空间,每次分配一个簇。对于一个文件的所有簇在FAT中是链在一起的。对于单面磁盘一簇是一个扇区(512个字节),双面磁盘是一簇含有两个扇区。对磁盘上的每一簇有1个12位的登记项(1.5字节)。前两个FAT登记项是对目录的一个区段映象,这些登记项包含磁盘的长度和格式指示。第三个FAT登记项开始数据区的映象。每个登记项包含三位十六进制字符:

000H——表示该簇是未用的,或是被删除的文件释放的,可以使用。

FF0~FF7H——是用来标识保留的簇,其中FF7H表明一个坏的簇,它是不分配的。

FF8~FFFH——标识文件的最后一个簇。用来表明该文件的结束。

xxxH——除上述之外的任何其他的十六进制字符。这是文件中下一个簇的簇号。文件中第一个簇号是保留在文件的目录登记项中。

FAT的两个拷贝一致。一个拷贝跟在另一个拷贝之后,当需要时FAT被读入DOS的一个缓冲区中,根据FAT调用文件。

由此可知,删除文件并不是把磁盘上所有存入的该文件的信息内容全部擦除,只是把该文件目录项中第0字节改为E5H(对应小写英文字母e),

表示该文件已被删除，不再使用。同时收回软磁盘中为该文件分配的存贮空间，变为可使用的自由空间，即FAT中该文件各簇登记项改为000H，以待分配给其他新文件。因此，只要在文件目录表中被删文件名尚未被新文件名覆盖，该文件所在的信息存贮空间还没有被新文件占用之前，在文件目录表中恢复文件名，进而恢复文件全部信息是不难做到的。

二、恢复步骤

恢复被删除的软磁盘文件可以用子程序调用的编程方法，也可用调试程序DEBUG来恢复，下面仅就利用调试程序DEBUG恢复软磁盘文件的方法和步骤介绍给读者。

用调试程序DEBUG恢复被删除的软磁盘文件的操作，可按以下步骤进行：

- 1.调用调试程序DEBUG，在DEBUG命令的提示符下，用装入命令L把目录区的信息装入微计算机的内存；
- 2.用显示内存命令D显示内存中被装入的目录区内容，寻找0字节为E5H（即文件名首字符为e）的文件目录，记下该字节的内存地址；
- 3.用修改内存命令E，把有删除标记E5H的内存单元内容修改为原文件名首字符的十六进制ASCII码；
- 4.用写命令W将改写过的内容从内存写回到磁盘的目录区中，然后用Q命令退出DEBUG，返回操作系统DOS。

至此，软磁盘中被删除的文件的名字已经恢复，文件内容也可以由系统命令调用了。但是以上

操作仅是恢复了文件名，而还没有把分配给该文件的磁盘空间从软磁盘的自由空间表中脱离出来，FAT中该文件信息占用的簇的登记项仍是000H，因此仍有被重新分配给新文件，由新文件覆盖的危险。

5.把已恢复原名的文件连同已经可调用的文件信息利用copy命令复制到另一个软磁盘上，被复制出来的文件就是已恢复的原来不慎被误删的文件。

如果需要，可以把原来软磁盘上该文件删除，把复制的恢复文件再复制到原来盘上去。这样既有了该文件的副本，原盘上文件也已恢复，就不怕再误删了。

三、应用举例

这里所举文件恢复应用例子是在pc兼容机ABACUS上进行的。应用MS-DOS2.10版本磁盘操作系统。原来的文件名为CUI02.ASM，操作过程如下：

```
A>DEBUG*           ; 调用调试程序。

-L 100 0 0 8; 从A驱动器将0扇区到
                        8H扇区的内容装入
                        100H开始的内存单元中。
-D 08FF:0B80; 从指定地址08FF:
                        0B80开始显示。

于是在计算机显示屏幕上出现图2所示的信息。
```

08FF:0B80	44 45 42 55 47 20 20 20-43 4F 4D 20 00 00 00 00	DEBUG COM...
08FF:0B90	00 00 00 00 00 00 00 60-54 07 32 00 80 2E 00 00	.....T.2.....
08FF:0BA0	41 53 4D 20 20 20 20 20-45 58 45 20 00 00 00 00	ASM EXE .....
08FF:0BB0	00 00 00 00 00 00 3D 00-21 00 3E 00 72 4A 00 00	.....=.!.>.rJ..
08FF:0BC0	E5 55 49 30 32 20 20 20-41 53 4D 20 00 00 00 00	eUI02 ASM..
08FF:0BD0	00 00 00 00 00 00 90 0C-21 00 59 00 12 02 00 00	.....!.Y.....
08FF:0BE0	43 55 49 30 31 20 20 20-4D 41 50 20 00 00 00 00	CUI01 MAP...
08FF:0BF0	00 00 00 00 00 00 78 00-21 00 58 00 EE 00 00 00	...x.!.X.n...

图2 (下转封三)

\* 横线上的字符是键入的命令，键完命令后按回车键。

# 汉字AST—PCNET宽域网

刘京松

(中国电子技术研究院系统工程部)

## 一、局域网与标准化

汉字AST—PCNET宽域网(以下简称PC—NET)是建立在美国加州AST公司的PCNET及RSN(Resource Sharing Network)网络基础上的。它比原网扩充了汉字处理、点到点实时通讯、网际通讯、分布式数据库、网络管理等多项功能,性能得到很大的提高,并移植到二十多种IBM PC类原装及兼容机上。

PCNET同以太网及PLAN系列网相比有较多的优越性。它是一个开放系统,遵循ISO/OSI协议,与IBM NETBIOS完全兼容,并在1986年6月14日的美国NCC会议上被IBM指定为定型网络,1985年美国网络评比中PCNET名列前茅。

PCNET与OSI的参考模型如下:

CCDOS 3.1及网络命令 ////////////////////	应 用 层
NETBIOS	表 示 层
	会 话 层
GATEWAY	传 输 层
CSMA/CD	网 络 层
	数 据 链 路 层
	物 理 层

PCNET是个开放式系统,网络软件完全支持CCDOS下开发的应用程序。其性能价格比高,安装简便,易学易用,软硬件的兼容性、可维护性、可扩展性、可读性及可开发性都是其它网络不可及的。

## 二、PCNET的运行环境

### 1. 硬件支持

PCNET是总线型的资源共享网。传输速率为

1Mbps或5Mbps(RSN),站间距离可达2.1公里(不加中继器),传输介质用75欧姆CATV同轴电缆,可连计算机多至250台,机器为IBM PC类机器或兼容机,采用CSMA/CD协议。加上中继器,总线可延伸到6.6公里,加上Modem,通过电话网、专用网或公用数据网,可进行远程工作站与网通讯及网际通讯,速率为1200波特或2400波特。通过在PC站上扩充AST GATEWAY仿真器,可使PCNET共享IBM 43XX/303X/308X系列机、DEC PDP-11/VAX11系列机、WANG VS系列机等大、中、小型机的资源,使PCNET中多达32台PC机作为大机器的终端来使用,同时PCNET上的各PC站还可相互通讯,共享局域网内资源。

### 2. 可扩展性

PCNET以网络总线为中心,连在网上的设备均是由PC类机器构成的PC系统。每个PC系统是一个工作站,可共享它站资源。又为网上提供可共享的资源。这里,PC机的最小配置是主机、单软盘、512KB RAM、单显。在此基础扩充的任何软、硬资源均可供全网共享。

PCNET拓扑范例如图1:

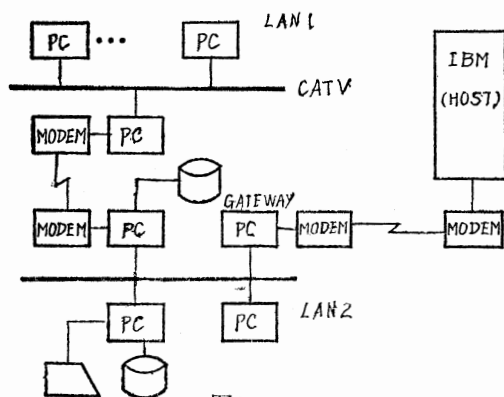


图1

### 3. 软件支持

现在的PCNET版本为V3.40, 兼容于PCNET以前开发的版本, 如HDCCNET, 汉字AST-PCNET局域网V3.2/3.32/V3.33。该版兼容于西文版本AST公司的PCnet V2.21/3.0/3.1/3.32, Orchid公司的PCnet V3.0及其后继版本。

支持PCNET的汉字操作系统: 电子部六所开发的CCDOS2.0/2.1/3.0/3.1及以后版本, 西文DOS 2.0/2.1/3.0/3.1/3.2。

PCNET软件是开放式系统, 层次化, 结构化, 并配有详细的系统分析、设计资料、框图及汇编或C语言网络源程序。读者可在其上根据自己的应用要求, 设计并实现自己的应用系统。

网络软件与用户界面分三级:

- (1) 菜单方式
- (2) 网络命令 (和使用DOS命令一样)
- (3) 语言开发工具语言, 如C-dBASE II/, 汉字AST-NDBMS, C-Lotus 1-2-3; 高级语言, 如C-C, C-BASIC, 汉字AST-NBASIC, C-COBOL等; 汇编语言。

### 三、PCNET的层次结构

PCNET是开放式系统, 各个层次结构清晰, 层内模块可以组合扩展, 易插入新的模块。

PCNET接口板是AST公司或Orchid公司推出的产品及兼容产品。CCBIOS使该网具有汉

应用软件 (MIS, OA)
网络C-dBASE II / I 等语言
网络管理
Real Time NET
CCDOS 2.0/3.1/及网络命令
NETBIOS及IBM <sup>®</sup> DOS
Datagram及NNC
IBM BIOS 及CCBIOS
PCNET 适配器
PC类计算机

字通信及处理能力。

IBM DOS 支持所有在其上开发的西文应用软件, CCDOS 2.0/2.1/3.0/3.1及HDCCDOS KH-CCDOS等, 使PC类机器上开发的中文及西文应用软件, 不经修改即可在PCNET上运行。

网络命令提供了多种功能: 如点到点实时通信软件, 文件保护、记录及并发控制的接口命令, 远程通信、执行、打印、作图等接口命令。

### 四、网络软件的可开发性

PCNET 及CCDOS 组成的网络CCDOS中断调用关系如图2所示:

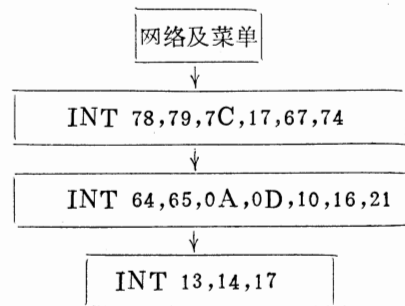


图2

读者利用这些汇编接口, 可开发出自己的应用程序, 如菜单程序, 网络管理, 分布式数据库 (DDB), 各种外设共享程序等。这些接口与NETBIOS及DOS中IBMDOS、IBMBIOS完全兼容。

为了支持高层通信软件, PCNET中利用INT 78实现了网内点到点通讯, 利用INT74实现了网与远程工作站及网与网间通讯, 其逻辑通信状态示意图如图3 (见下页):

### 五、网络数据库与分布式数据库

在PCNET上直接支持C-dBASE II/I, 在其上开发的应用软件通过前述的文件名更换及并发控制加锁机制的嵌入实现了网络数据库环境, 并已投入实用。如在某银行, 通过在C-BASIC应用软件中增加上述功能的少许语句达到了文件及记录级的并发控制及操作。某车站用的C-dBASE II编写的货运调度程序利用上述原理作了机械性扩



充，达到了文件及记录级的并发控制及操作。解放军某部用C-dBASE II编写的指挥管理程序，同样实现了文件及记录级的并发控制及操作。PCNET支持LAN: Data Core多用户（或网络）数据库及其它数据库。

INT 78/74 stream 4X/5X

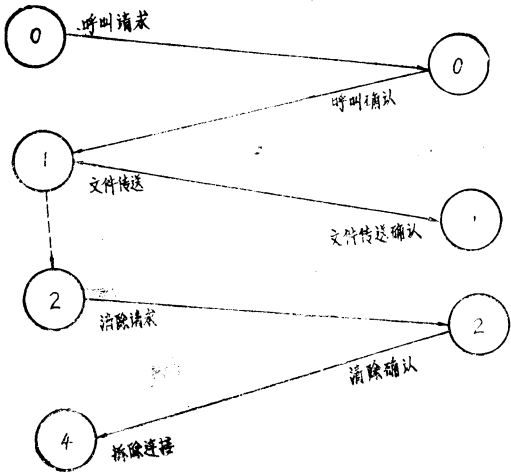


图3

在PCNET上，我们开发出了两个数据库系统。

具有分布处理能力的C-dBASE II叫做“汉字AST-NDBMS”，共享数据库站上的主要执行文件为DDBS，使用数据库站的执行文件称为D-DBU。此网络数据库具有并发控制多个文件及记录的能力，与C-dBASE II完全兼容。

具有分布处理能力的C-BASIC叫做“汉字AST-NBASIC”，主要执行文件称为NBASIC。它与解释或编译BASIC完全兼容。用它开发出的应用程序可在PCNET上实现文件及记录的并发读写，保证了数据的安全性及实用性。

上述两个网络应用软件均已投入实用，大大方便了使用PCNET的用户。对于已用C-dBASE II或C-BASIC开发应用软件的用户，只需在应用软件源程序中机械性的增加少许语句（提供新的几条语句），再使用该软件或是直接用此软件编写应用程序。这样，您的应用软件就会从集中式数据处理走向分布式数据处理，即建立了真正的OAS或MIS。当然，PCNET还可支持NC，NPASCAL，NCOBOL等分布式处理语言。

## 六、点到点实时通信

利用PCNET中INT 67的数据报收发机制（类似于X.25）的INT 78/79/74基础通信软件，以及INT 0A/0D中断处理，我们实现了点到点通信，逻辑拓扑结构如图4所示：

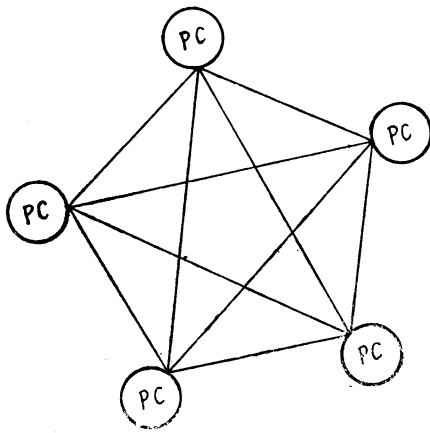


图4

此结构实现了全图，即各站均可同其它各站进行实时通信，共享网中所有软硬件资源，形成没有固定网络总控制机的完全分布式系统，这种产品在市场上是很少见的。

在此结构基础上，我们接管并扩充了INT10，13，16，21，7C，OE，开发了具有实时功能的菜单文件pcnet.exe，其中的查询文件、传输文件、站间对话、广播通信、超级用户、屏幕传送功能都是实时处理的，即通信中不破坏对方站的工作，自动保护屏幕和现场，当通信完后自动恢复原来的工作。对于通信中的交互信息，我们大量调用INT 13读写硬盘，极少调用int 21，因为使用int 21时，读写盘中断就会出现“死锁”。现在开发的各项实时功能经用户实用考验，解决了这一问题，进入了商品化。

## 七、网络管理与安全保密

为了适应用户单位管理人员（如工厂的厂长，部队的首长等）分层管理的需要，PCNET提供了站间层次差别。

进入网络系统前用password文件可建立各用户站口令，高优先级用户（一般是厂长或首长）利

用password所建的口令进入网络,可中断正通信的低优先级用户。同样还可建立三层、四层的管理层次;在网中各站通信过程中,主动站提出拷贝被动站的文件或要求通话时,须得到对方站的允许;各站在系统生成时,可由用户选择本站资源的读/写/执行许可权,因而实现了对资源的保护。

读者利用SPCDIR、UPCDIR、NETDIR等命令查看网络各站的权限和配置;利用SPCINST、UPCINST、NETINST等命令改变各站的权限和配置。

## 八、假脱机打印与远程执行

PCNET提供了假脱机打印系统,打印机安装在任一站上均可供全网共享。当用户使用网络打印文件时,还可进行其它工作,这时的PC站进行前后台处理。

PCNET中,RE·EXE实现了远程命令执行,即共享其它站的CPU,利用此低级接口,实现了假脱机打印、远程作图等项功能。远程执行是一站向另一站发一命令使对方执行,结果可自动回送本站。电子驱动器允许用户在RAM中仿真驱动器对此操作就象操作本站的A盘一样,但其速度比物理驱动器快十倍。读者可根据需要,利用MA-KEVOL命令对大磁盘分成多个虚拟卷。

## 九、PCNET与大机器接口

PCNET中的NETBIOS介面支持PCNET与IBM的大机器进行通信。通过AST-SNA/BSC Gateway使PCNET连接IBM3270、43xx、303x及308x等大机器于SNA/BSC 3270操作环境,使整个网上的PC站作为IBM3278 MODEL 2或IBM 3279 MODEL 2A终端使用,这样的终端可达16/32台。

同理,PCNET也能以类似的方式与PDP11系列、VAX11系列,WANGVS系列等大、中、小型计算机连网通信,共享其软、硬件资源。

## 十、PCNET的前景

PCNET以国家选型的长城0520机、IBM PC类机器及兼容机为基础,同时可以通过公用或专用电话网/数据网进行远程通信,也可共享国家选型的IBM大型机及VAX系列小型机软、硬件资源。功能上优于其它一些局域网及远程网。网络协议标准符合国标,适用于大、中、小型规模的网络。软件上支持标准的CCDOS、分布式的C-dBASE I/II,点到点实时通讯等功能。PCNET在国内已进入教育、企事业、军事、银行、交通运输,科研机构等部门的MIS或OA系统的实用阶段,安装了近1千个工作站。

# 可编程控制器S5-115U的CPU硬件结构

金广业

(冶金部自动化研究所)

本文在现有资料<sup>[1]</sup>的基础上,结合所做的开发研制工作,对西德西门子公司生产的可编程控制器S5-115U的CPU941硬件结构作了分析。

## 一、序言

可编程控制器是一种用于过程控制的工业控制装置。它既有逻辑运算、计时、计数、分支程序、子程序等顺序控制功能,也有数字运算、数据处理及模拟量调节等功能。它具有结构简单、性能优越、编程简单、可靠性高等优点。广泛用于工业生产过程的自动控制,如机械制造业的加工自动线、特种工艺线;冶金工业的烧结、冶炼、连铸、轧钢等过程;电力工业中火电厂各种热力系统的程序控制以及在食品加工业、煤炭工业、造纸工业中都广泛应用。目前世界上许多厂家推出了各种系列的可编程控制器,如日本的富士、日立、东芝;西德的西门子;美国的西屋公司、TI公司等。纵观各公司的产品,我们不难发现它们有许多相同之处,硬件结构也雷同。因此,若对其中一个产品有所了解,就能较容易地了解掌握其它产品。

在众多系列产品中,近几年我们选用西门子公司U系列产品用于冶金工业生产过程,特别是将其应用于高炉控制系统中<sup>[2]</sup>,收到了良好效果。由于各公司对其产品的技术资料管理严格,出售产品时配备的资料很少,因此在引进应用的基础上,本着消化吸收的方针,我们对西德西门子的可编程控制器CPU941进行了开发、分析。

## 二、主要技术参数

CPU941是S5-115U的中央处理模板,它将各输入端的状态读入,处理存放在用户程序中的STEP5指令,再去控制与被控对象相连的输出端。其主要技术参数如下<sup>[1]</sup>:

电流损耗:

$U_N = 5V \pm 2\%$ , 最大1.2A

$U_N = 5.2V \pm 2\%$ , 最大0.6A (当与编程器605/615相连时)

$U_N = 24V \pm 2\%$ , 最大40mA (用编程器615U对EPROM写程序时,为100mA)

处理时间:

位指令:  $2.2\mu S$

字指令:  $40 \sim 180\mu S$  (详见指令表)

最大开关量输入/输出: 512/512点

最大模拟量输入/输出: 64/64路

1024个暂存单元 (相当于中间继电器) 作为保护型。

1024个暂存单元作为非保护型。

定时器: 128路

$T_{0.4} \sim T_{0.9}$  可作为保护型

$T_{0.4} \sim T_{1.7}$  是不能保护的

时标有0.01秒, 0.1秒, 1秒, 10秒

扫描倍率: 0~999

定时范围: 0.01秒~9990秒

计数器: 128路

$C_{0.4} \sim C_{0.9}$  可作为保护型

$C_{0.4} \sim C_{1.7}$  为不保护型

计数范围: 0~999

## 三、基本工作原理

CPU941是由一个位处理器和一个字处理器组成。字处理器统一管理所有的编程器接口功能,管理内部定时器,监视扫描时间,处理字节指令,控制S5总线。同时字处理器也负责对位处理器的控制。二个处理器的时钟频率是分开的。

CPU941中除有操作系统本身所需的内存外,还备有1K的内存供用户程序使用,同时备有存储器插件接口。用户可选用4K或8K插件块供编程使

用。

除此以外，CPU941还有总线接口，操作及指示，时钟与控制等。其方块图如图1。

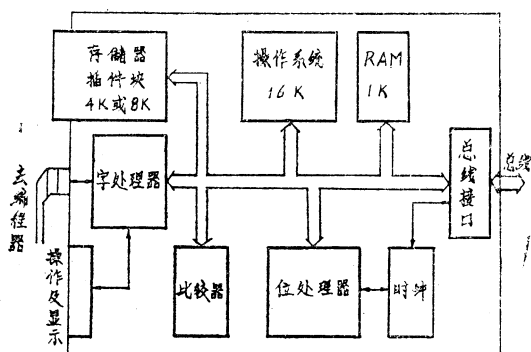


图1 CPU941原理图

## 1. 二个处理器之间的关系

前面已经叙述了字处理器负责对位处理器的控制，那么字处理器是怎样控制位处理器的呢？

CPU941的字处理器采用的是8031单片机。对于8031单片机大多数读者都比较熟悉，我们不再详述。

在CPU941中，8031单片机将位处理器看作它的一个外设。位处理机通过中断请求，分时地处理位指令。另外通过比较器将位处理器中的内容与数

据总线上的数据比较，再去控制二个处理器之间的通讯。由此可见，字处理器的作用相当于一个主处理器，它全面控制，协调CPU941的各个部分。而位处理器在这一点上相当于一个从处理器。在机器运行过程中，位处理器仅处理位指令。在执行指令的速度上，位处理器比字处理器更快。

## 2. 16K字节的操作系统的扩展

我们知道8031单片机本身不带有程序存储器，所以必须外接程序存储器。16K字节的操作系统是由EPROM27128构成的，其构成电路如图2。

8031的EA接地，CPU总是执行外部EPROM的程序。CPU在访问外部程序存储器时，P<sub>2</sub>口输出PC的高8位地址，P<sub>0</sub>口分时传送PC的低8位地址和指令字节。在ALE为高电平时，P<sub>0</sub>口输出的地址有效，并由ALE的下降沿锁存到地址锁存器74LS373中。接着外部程序存储器选通信号PSEN出现负脉冲，选通相应的EPROM存储器，相应的指令字节出现在EPROM数据线上并输入到P<sub>0</sub>口，8031CPU将指令字节读入指令寄存器。

由于扩展的是16K程序存储器，所以仅用14位寻址。这样只要8031寻址此16K程序存储器，其第16位必然是“0”，所以利用此位来控制这个程序

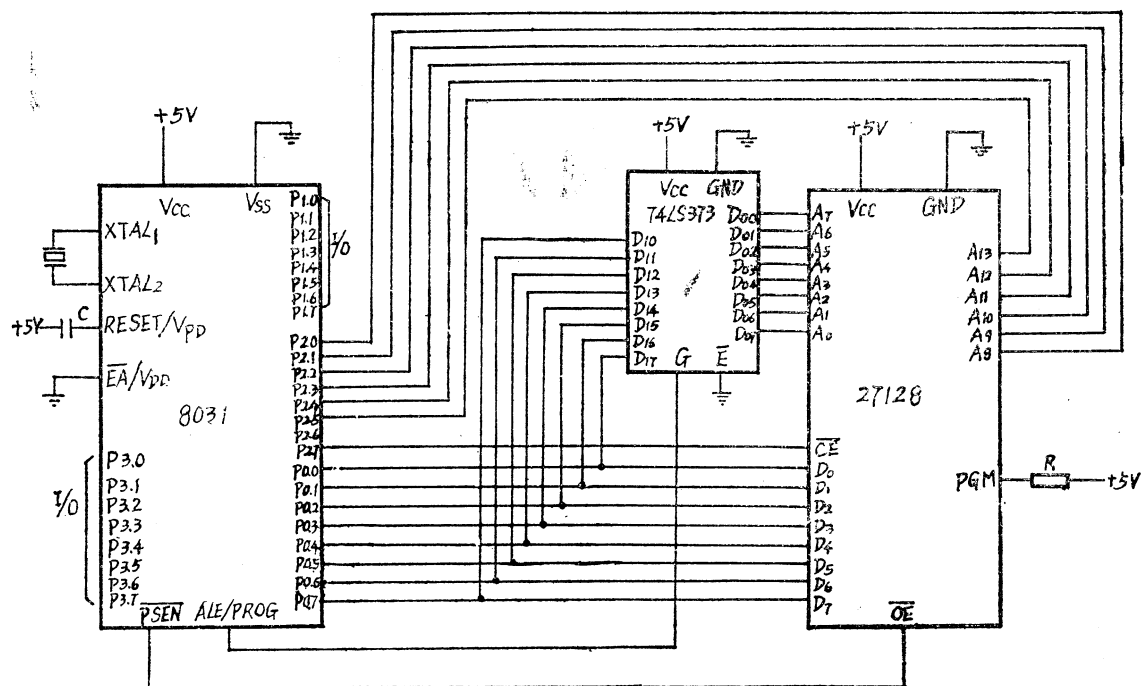


图2 用27128扩展16K操作系统

存储器的 $\overline{CE}$ 信号。

这里所采用的锁存器是74LS373,它是带有三态门的8位锁存器。 $\overline{E}=0$ 时三态门导通,输出线上为8位锁存器的状态; $\overline{E}=1$ 时输出线呈高阻状态。 $G$ 为锁存信号输入, $G=1$ 时锁存器输出等于 $D$ 端输入; $G$ 输入端的负跳变将输入信息锁存到8位

锁存器中。实际上,类似的8位锁存器都可以在此种情况下使用。

### 3. 用TC5565扩展8K数据存储器

此SN74LS373的作用与扩展16K程序存储器时的作用一样。且 $P_2$ 口也是输出PC的高8位地址, $P_0$ 口也是分时传送PC的低8位地址和数据。其电路组成如图3。

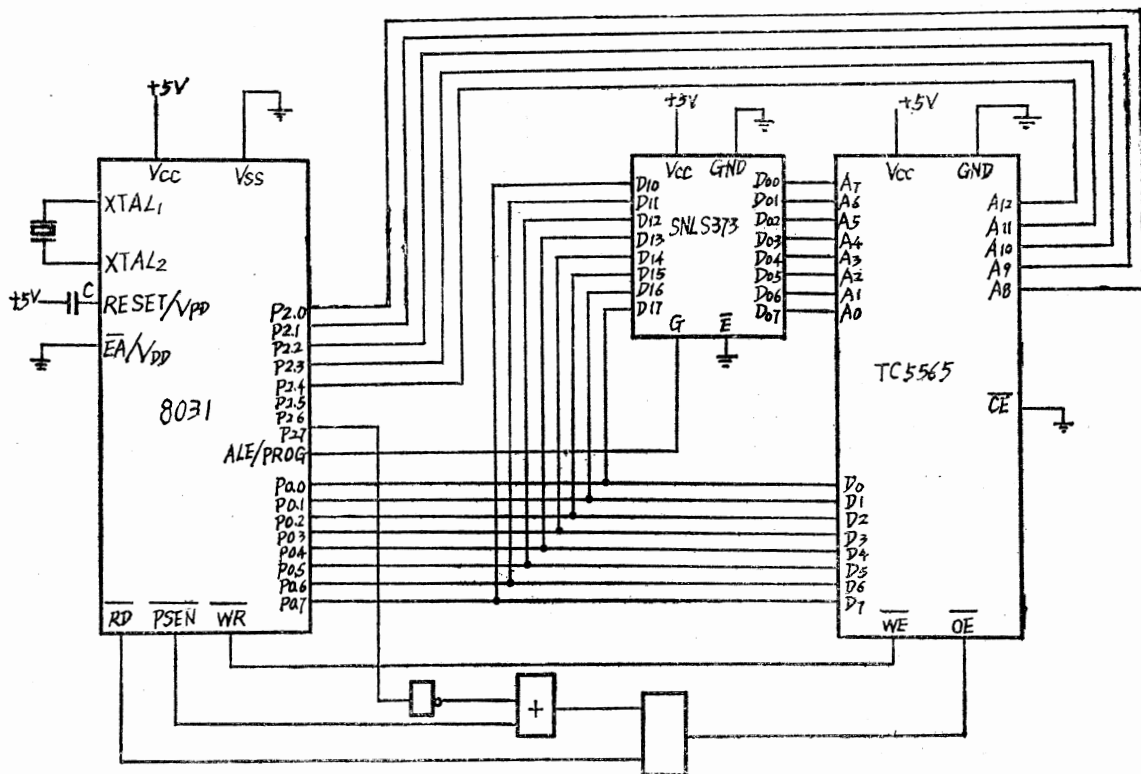


图3 用TC5565扩展8K数据存储器

由于扩展的是8K数据存储,所以仅需13位寻址。TC5565的 $\overline{WE}$ 为写输入允许线,这里由8031的外部数据存储器写脉冲直接控制。而TC5565的输出允许线则由下述信号控制。

$$\overline{OE} = \overline{RD} \cdot [\overline{PSEN} + \overline{P2.7}]$$

由于这里是13位寻址,所以在查询此数据存储器时 $\overline{P2.7}=1$ ,以上公式简化为

$$\overline{OE} = \overline{RD} \cdot \overline{PSEN}$$

输出允许线是由 $\overline{RD}$ 和 $\overline{PSEN}$ 的逻辑与来控制的。这样做是因为用户需要执行用于存放数据的存储器中的程序。这二个信号的逻辑与将8031的外部程序存储器和外部数据存储器的空间合并了,使得用户在此数据存储器中既可以存放数据也可以存放指

令。

需要指出的是:这里扩展的虽然是8K数据存储器,但仅有1K可供用户占用。笔者认为其它7K空间是作为操作系统所占用的数据区,在使用中用户应注意到这一点。

前面我们简单地介绍了S5-115U的CPU941是怎样扩展程序存储器和数据存储器,这些扩展都是在模块本身实现的。为进一步扩展用户的存储空间,CPU941还备有存储器插件块接口,用户可按需要插接4K或8K的存储器插件块。在掌握前面二种扩展技术后,参阅有关文献<sup>[3]</sup>即可了解CPU941中的存储器插件块接口,这里就不再详述了。

### 4. 操作、显示及时钟

图4表示操作、显示控制电路。图中 $D_1$ 为带

清除端的可重触发单稳态触发器， $D_2$ 为带清除端的上升沿D触发器。 $K_1$ 为停止及运行开关， $K_2$ 为预置开关。 $L_1$ 为绿灯，CPU处于运行状态时亮。 $L_2$ 为红灯，CPU处于停止状态时亮。 $T$ 为外部电路来的复位信号输入端，此复位信号同时输入到8031的RESET端和 $D_2$ 的T端。

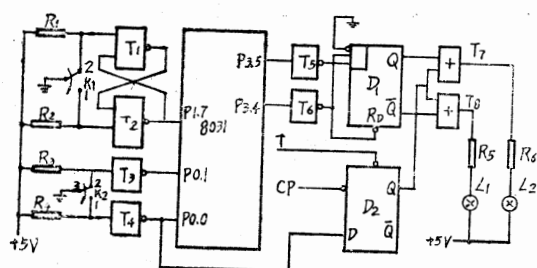


图4 操作、显示控制电路

当 $K_1$ 位于2端时， $T_2$ 输出低电平。8031收到这个低电平信号后，按操作系统指令运行，使CPU进入停止状态。并由8031的 $P_{2.4}$ 和 $P_{3.5}$ 端输出信号，使 $D_1$ 的Q端输出低电平从而 $T_7$ 也输出低电平导致 $L_2$ 亮。这仅是由 $K_1$ 置于2端时，CPU进入停止状态的情况。在编程器上调用STOP功能和出现故障或错误时也将自动进入停止状态。当 $K_1$ 位于1端时， $T_2$ 输出高电平。8031接收到这个高电平后，按照操作系统中的指令运行，使各部分进入运行状态。并由8031的 $P_{2.4}$ 和 $P_{3.5}$ 端输出信号，使 $D_1$ 的 $\bar{Q}$ 端输出低电平， $T_8$ 也输出低电平，导致 $L_1$ 亮。同样用编程器调用RUN功能也可使CPU进入运行状态。

$K_2$ 为预置开关，位于1端时， $T_4$ 输出高电平，8031接收到这个信号后，在系统初始起动时，所有暂存单元，定时器及计数器全部清零，并且不对系统进行保护。 $K_2$ 位于2端时， $T_3$ 输出高电平， $T_4$ 输出低电平。8031收到这个信号后，则对系统进行保护，并在系统初始起动时，将能保持的暂存单元、定时器及计数器的内容保持住，其余的清零。

开关 $K_2$ 若置于中位时，即 $T_3$ 、 $T_4$ 均输出低电平，此时若将开关 $K_1$ 连续二次由2端拨向1端，即2端→1端→2端→1端，则系统进行原始清零。将存在内存RAM中的用户程序，全部数据及错误指示全部清除。

CPU941中分为二个时钟，其中8031的时钟是用一般的外接晶振组成的<sup>[3]</sup>，而位处理器和总

线中的节拍是由下述时钟的电路组成的（见图5）。

此电路由一个晶体控制振荡器组成。外接晶振频率是7MHz。晶体控制振荡器的输出同时提供给位处理器和总线接口。

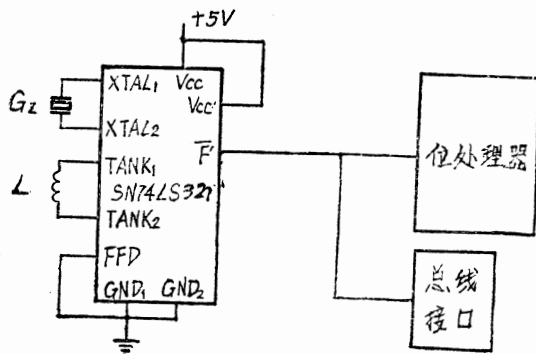


图5 位处理器及总线中的时钟电路

## 5. 编程器接口和总线接口

编程器接口与8031的串行口直接相连，关于8031串行口的工作方式已有介绍<sup>[3]</sup>。

数据总线和高8位地址线都经过缓冲器和锁存器。低8位地址线经地址锁存器后再经缓冲器和锁存器。

7根I/O模板使能线是由地址的第3位到第7位译码获得再经缓冲器输出的。

其它控制线，电源线等都是经缓冲器的。

## 四、总线结构

CPU941采用的是S5总线，共有64根线。总线具体排列如图6。总线的符号、功能等如表1。

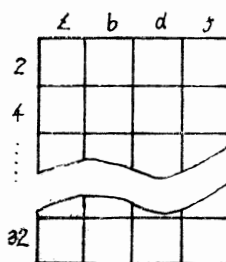


图6 S5总线排列

S5总线有8根数据线，16根地址线，7根I/O使能线，5根中断线，8根电源线及其它的控制线。对于各控制线，表中明确给出了其功能，读者可作进一步分析。

## 五、结论

通过上述分析，可得出如下结论及问题。

表1 S5总线符号、引脚、功能

引脚号	符 号	名 称	功 能
Z2	5V	5伏电源	各模板的正电源
Z4	TAKT	节 拍	各模板的节拍信号
Z6	RESET	复 位	对各模板用的清零脉冲
Z8	<u>MRD</u>	存储器读	读存储器信号, 每个读操作时都出现此信号
Z10	<u>MWR</u>	存储器写	写存储器信号, 每个写操作时都出现此信号
Z12	<u>RDY</u>	读写响应	准备好, 对MRD及MWR的回答信号
Z14~Z28	DB0~7	数据线0~7	第0~7位数据
Z30	M24V	24伏地	24伏接地端
Z32			未规定
b2	M	地	5伏及5.2伏接地端
b4	PESP	外设存储	外设操作信号, 存取每个外设存储时出现此信号
b6~b28	ADB0~11	地址0~11	第0~11位地址
b30	BASP	禁止输出	CPU进入停止状态时, 有此信号, 它阻止数字信号输出
b32	M	地	5伏及5.2伏接地端
d2	5.2V	5.2伏电源	编程器605U及615U用的电源
d4	UBATT	3.4伏电池电压	RAM备用的3.4伏电池电压
d6~d12	ADB12~15	地址12~15	第12~15位地址
d14	<u>IRA</u>	中断A	中断信号A, 来自智能模板
d16	<u>IRB</u>	中断B	中断信号B, 来自智能模板
d18	<u>IRC</u>	中断C	中断信号C
d20	<u>IRD</u>	中断D	中断信号D
d22	<u>BAU</u>	电池失效	电池不正常(没有电池或电池电压低于要求值)
d24	<u>NAU</u>	系统断电	系统断电信号, 仅电源故障时来此信号
d26	<u>PEU</u>	外设失效	外设不正常。扩展单元电源失效时有此信号
d28			未规定
d30			未规定
d32	BASPA		通讯模板、智能模板及扩展连接模板控制信号
f2	5V	5伏电源	各模板的正电源
f4~f16	<u>F0~6</u>	I/O使能 0~6	I/O模板使能线0~6
f18	<u>ASF</u>	接口空闲	中心单元可以在没有扩展单元连接时工作。 此时必须将专用的终端匹配器插入IM插座位置。
f20	<u>HOLD</u>		S5总线由智能处理控制模板占用
f22~f26	<u>HOLDA1~3</u>	总线允许1~3	S5总线使能信号, 用于智能处理控制模板
f28	<u>PRAL</u>	过程中断	来自开关量I/O模板的硬件中断信号
f30	24V	24伏电源	20mA接口电源和编程器615U用于写程序的电源
f32	<u>ASG</u>	扩展模板信号	表示在IM插座上有扩展连接模板



1. CPU941是由一个位处理机和一个字处理机组成的双机系统。由字处理机8031管理、控制整个系统。位处理机则负责高速地处理位指令。

2.所有的程序存储器、数据存储器都是采用8031的存储器扩展技术实现的。

3.总线结构是结合8031单片机设计的。结构紧凑、合理,适用于S5-115U这种模块式结构的系统。

#### 参考文献

- [1] SIEMENS: SIMATIC S5-115U Programmable Controller, Manual, 1985

- [2] 葆鸿熙等:可编程控制器SIMATIC S5-115U在高炉装料系统中应用,高炉自动化学术会议论文集,中国金属学会冶金自动化学会,1986

- [3] INTEL: MCS-51 Family of Single Chip Microcomputers User's Manual, 1981

- [4] 胡仁安:可编程序控制器及其发展,高炉自动化学术会议论文集,中国金属学会冶金自动化学会,1986

### 动态与信息

## 第六代计算机——生物计算机

本世纪八十年代以来,美国、日本和苏联等国的科学家在着手研制第五代计算机——人工智能计算机的同时,也开始研制第六代计算机——生物计算机。

生物计算机的主要原材料是借助生物工程技术(特别是蛋白质工程)生产的蛋白质分子,以它作为生物集成电路——生物芯片。在生物芯片中,信息以波的形式传播。当波沿着蛋白质分子链传播时,会引起蛋白质分子链中单键、双键结构顺序的改变。因此,当一列波传播到分子链的某一部位时,它们就象硅集成电路中的载流子那样传递信息。由于蛋白质分子比硅芯片上的电子元件要小得多,彼此相距甚近,所以生物元件可小到几十亿分之一米,元件的密集度可达每平方厘米 $10^{15}$ 至 $10^{16}$ 个,

甚至 $10^{18}$ 个门电路,再加上生物芯片具有天然的立体化结构,其密度比平面型的硅集成电路高三至五个数量级。这就意味生物计算机完成一项运算,时间仅为目前集成电路的万分之一。更可贵的是,由于生物芯片的原料是蛋白质分子,使得生物计算机既有自我修复的功能,又直接可与生物活机结合。

同时,生物芯片具有发热小,功耗低,电路间无信号干扰等优点。现在,有些科学家把数千台微型计算机联结起来,借以获取高速度,称为“并行处理”,而生物芯片本身就具有并行处理的功能。

人们对生物计算机将登上二十一世纪科技舞台抱有很大的希望,从而对未来世界产生不可估量的影响。

## 新书介绍与征订

《IBM PC系统软件与应用软件汇集》将于1988年2月由天津科学技术出版社出版。

IBM PC微型机在我国科技、厂矿、机关等部门几乎尽人皆知。它是美国IBM公司于1981年正式推出的。其后，由于IBM公司的声誉，使很多用户与软件公司纷纷为其开发系统软件与应用软件，因此IBM PC很快就风靡全世界，到1984年，它已占微型机市场的第一位，这又促使IBM PC的各种兼容机应运而生。IBM PC成功的主要原因是配有大量商品化的应用软件包。目前，我国国内已经装备了大量IBM PC微型机，迫切需要开发与引进一些实用的应用软件。

《IBM PC系统软件与应用软件汇集》由天津电子计算机联合公司和天津电子计算机研究所编译，是一本系统介绍IBM PC微型机各种软件的新书。本书汇集了1984年一季度前美国市场上所有IBM PC微型机的应用软件包。由于应用软件包必须在操作系统支持下运行，故本书首先对IBM PC机所配用的各种版本的操作系统作了介绍，以帮助用户掌握应用软件的选用。为了帮助用户自己编写应用程序，掌握所需语言，本书对IBM PC微型机所配用的各种语言也作了介绍。书中大部分篇幅着重介绍了对用户非常实用的应用程序。

本书共分十四个部分，包括：操作系统、语言、实用程序、财务分列表程序、字处理程序、数据管理程序、图形程序、通信程序、教育程序、金融应用程序、会计程序、企业管理程序、工业应用程序和作业专用应用程序。书中所介绍的各种软件除适用于IBM PC机外，也适用于它的各种兼容机。目前，0520系列微机已成为国产机的主流机型之一，它是IBM PC的一种全兼容机。本书对帮助上述各机种用户在自己的系统上开发更广泛的应用，提高系统功能和利用率，提高用机效率及效益无疑是新的佳音。本书还是IBM PC机及其兼容机用户、软硬件开发人员不可缺少的参考书和工具书。

本书所介绍的各种软件均由专家编写，深入浅出，容易掌握。全书共约40多万字，16开本，约19印张。书中各章节基本上互相独立，读者可根据需要选读。（欲订者，请填写背面的预订单）


天津科学技术出版社  
1987.6.

附三联单和说明

报 销 凭 证 (一联)

订阅单位:

198 年 月 日

订 阅 资 料 名 称	单 价	订 阅 份 数	金 额	发 行 单 位 财 务 公 章
IBM-PC系统软件 与应用软件汇集	4元3角			
合 计 (大 写)	元 角 分			
<p>★: 此联与银行汇款单据一并作报销凭证, 不另开发票。                      开户银行: 天津市河西区大沽路分理处账号: 601—67045</p>				

此联订阅者撕下报销用

请将二、三联填好与银行信汇单一起由银行转回本服务部

订 阅 单 (二联)

198 年 月 日

订 阅 单 位					开 户 银 行		
详 细 地 址					帐 号		
IBM-PC系统软件 与应用软件汇集	单 价	4元3角	份 数		金 额	订 阅 单 位 公 章	
经 办 人	合 计 (大 写)				元 角 分		

此联寄回我部作记帐凭证

发 行 单 (三联)

198 年 月 日

订 阅 单 位					开 户 银 行 及 帐 号		
详 细 地 址					收 件 部 门		
IBM-PC系统软件 与应用软件汇集	单 价	4元3角	份 数		金 额	订 阅 单 位 公 章	
经 办 人	合 计 (大 写)				元 角 分		

此联寄回我部作发行凭证

该书16开本, 定价4.30元, 1988年2月出书。欲订者, 请填好订单, 加盖公章, 将二、三联附在信汇单后面寄回。汇款由银行或邮局均可。单位购书, 加购书款的10%作为邮费; 个人购书, 加0.12元的挂号费。截订日期: 1987年12月底。

开户银行: 天津市河西区大沽路分理处。帐号: 601—67045

地址和联系人: 天津市和平区赤峰道130号, 天津科技出版社刘自强。