

21世纪高等学校计算机专业规划教材

80X86 汇编语言程序设计

编著◎张君 高福祥 齐志儒



大连理工大学出版社 Dalian University of Technology Press

21 世纪高等学校计算机专业规划教材

80X86 汇编语言程序设计

张 君 高福祥 齐志儒 编著

大连理工大学出版社

图书在版编目(CIP)数据

80X86 汇编语言程序设计/张君,高福祥,齐志儒编著. —大连:大连理工大学出版社,2000.8

(21世纪高等学校计算机专业规划教材)

ISBN 7-5611-1753-1

I .8 … II . ① 张… ② 高… ③ 齐… III . 汇编语言-程序设计-高等学校-教材 IV .TP313

中国版本图书馆 CIP 数据核字(2000)第 23041 号

大连理工大学出版社出版发行
大连市凌水河 邮政编码 116024
电话:0411-4708842 传真:0411-4708898
E-mail: dulp@mail.dlptt.ln.cn
URL: http://www.dulp.com.cn
丹东日报印刷厂印装

开本:787×1092毫米 1/16 字数:470千字 印张:19.5
印数:1-6000册

2000年8月第1版 2000年8月第1次印刷

责任编辑:吕志军 责任校对:蒋浩
封面设计:孙宝福

定价:20.00元

内 容 简 介

本书以 PC 系列机为背景系统地介绍了汇编语言程序设计的基本理论和方法。全书正文由十一章组成。第一到第五章主要讲述了计算机的基础知识、8086/8088 的指令系统、汇编语言、程序的基本控制结构及程序设计理论、方法、技巧和宏汇编技术。第六章介绍了汇编语言在数值运算与代码转换方面的应用。第七章讲述了输入输出及中断程序设计的概念、方法和技巧,并以键盘和打印机为例详细介绍了其输入输出程序的设计方法。第八章介绍了 80286 的程序设计方法。第九章介绍了 80386、80486、Pentium(80586)的特点、指令系统及存储管理的有关知识。第十章介绍了 8087/80287/80387 的指令系统及程序设计方法,并给出了应用实例。第十一章介绍了 MMX 的程序设计方法。

每章都附有大量习题。

本书在编写上语言通俗易懂,叙述由浅入深,循序渐进,结构清晰严谨。

本书可作为高等学校计算机专业和自动化专业的教材和计算机专业自学考试的自学教材,也是工程技术人员自学进修和实际编程的极为实用的参考书。

出版说明

1989年,根据国务院关于高等学校教材工作分工的规定,按照原电子工业部教材办公室的安排,我社承担了全国高等学校工科电子类专业教材的出版工作,出版了《计算机网络》、《数据结构》、《数字电路逻辑设计》和《程序设计语言编译方法》四本电子类专业教材。这些教材来自于教学实践,又经编审委员会小组择优评选出来,所以质量较高。十多年来,这些教材经过大范围的推广发行,得到了广大读者的认可,受到各个学校的广大师生的一致好评,其中《计算机网络》第一版获得第二届全国优秀教材奖,《数字电路逻辑设计》获得原机械电子工业部电子类优秀教材二等奖。

近年来,我社在计算机图书的运作方面投入很大,每年都有大批计算机方面的图书问世,其中不乏各个层次的教材,可以说,我们的计算机教材出版发行工作已有了一个非常厚实的积累。在此基础上,借世纪之交计算机专业教学计划调整的机会,我们组织编写了这套《21世纪高等学校计算机专业规划教材》。

本套教材的编写宗旨是不求大而全,而求简而精。由于教学时数的限制,教材要做到面面俱到不现实,也没有必要,我们主要从先进性、实用性和可操作性角度把握,使本套教材能简捷精炼,重点突出。

本套教材的使用对象是普通高等学校的本、专科生。我们针对这一定位,结合普通高等学校的具体情况,对每本教材的具体内容进行了认真的研究,使其能紧密结合各个学校的教学实际。

为了保证这套教材的质量,我们精心组织了国内多所大学的知名专家、学者作为我们的顾问和编者,他们是清华大学、解放军理工大学、北京联合大学、哈尔滨工业大学、东北大学、大连理工大学等学校的教授,其中有的在本专业有很深的造诣,是国内相应领域的知名专家,有的多年工作在教学和科研第一线,有着丰富的教学经验。强大的编审者队伍给本套教材提供了强有力的技术支持和保证。

本套教材的特点是:

简洁——避开了高深的理论,简明扼要地介绍学生最需要的基础知识和技术;

通俗——通过通俗易懂的语言讲授计算机专业技术知识;

先进——在内容上吸收新技术、新动向,保持一定的前沿性;

实用——本套书能既适合于教,更适合于学,对普通高等学校计算机专业的教学具有较强的适用性。

真心希望本套教材能成为老师的助手,学生的良师益友。

大连理工大学出版社

2000年6月

前 言

汇编语言是一种强有力的语言,它能透彻地反映、巧妙而充分地运用计算机的硬件功能和特点,便于编程人员根据自己的需要灵活地编制高级语言能实现和无法实现的各种程序,随心所欲地控制计算机的运行。汇编语言是为计算机提供的最快、最有效的语言,也是能够利用计算机所有硬件特性的惟一语言,在许多对运行速度要求很高的场合,汇编语言是必不可少的,如操作系统、编译程序等多数是用汇编语言编写的。近年来,微型计算机的迅速发展,特别是它在实时控制和计算机网络与通信等方面的广泛应用,以及分析、消化、扩充、改造、移植、汉化、开发各种系统软件的需要,使汇编语言程序设计技术成为广大微型计算机用户不可缺少的看家本领。

《80X86 汇编语言程序设计》是高等学校计算机、自动化等专业学生的必修课程之一。它不仅是计算机原理、操作系统等其他课程的必要先修课,而且对于训练学生掌握程序设计技术、熟练上机操作和程序调试技术有重要的作用。

本书以 IBM PC 系列机为背景机,详细讲述了 8086、80286、80386、80486 以及 Pentium (80586)的程序设计方法,并介绍了 MMX 的程序设计。

本书是在东北大学等多所院校应用多年教材的基础上编写的,作者结合教学和科研经验,进行了反复认真的讨论,改变和充实了原教材的内容,使本书内容更加完善和充实。

全书共分十一章。全面清楚地讲述了 8086/8088 的宏汇编语言,详细讨论了程序的基本控制结构、设计原理及编程技巧,以大量的实例说明了数值运算和代码转换的算法,深入地剖析了 IBM PC 机的中断系统,并给出了实用程序;详细地介绍了 80286、80386、80486 和 Pentium(80586)的扩充功能、指令系统及存储管理的有关知识,讨论了保护虚地址方式下的程序设计方法;介绍了 MMX 的程序设计方法;最后,完整地介绍了 8087/80287/80387 的结构、指令系统,并以若干程序实例说明了其指令系统的应用。本书中给出的所有程序都是经过上机验证的。各章附有大量的习题。

本书在编写上力求语言通俗易懂,叙述由浅入深、循序渐进,结构清晰严谨。

本书可作为高等学校计算机专业和自动化专业的教材和计算机专业自学考试的教材,也是工程技术人员自学进修及实际编程的参考书。

全书由张君、高福祥、齐志儒编著。由大连理工大学陈连玉教授主审。

限于编者水平,本书中错误和不妥之处在所难免,敬请读者不吝批评指正。

编 者
2000 年 6 月

目 录

出版说明

前 言

第一章 概 述	1
1.1 微型计算机的发展过程	1
1.2 微型计算机的特点	2
1.3 计算机系统	3
1.3.1 硬件子系统	3
1.3.2 软件子系统	5
1.4 计算机语言	6
1.4.1 机器语言	7
1.4.2 汇编语言	7
1.4.3 通用语言(高级语言)	9
习 题	10
第二章 8086/8088 微型计算机的结构	11
2.1 8086/8088 微型处理机的结构	11
2.1.1 8086/8088 微型处理机的结构	11
2.1.2 8086/8088 寄存器和标志	12
2.2 8086/8088 的存储器管理	15
2.2.1 8086/8088 的存储器的分段结构	15
2.2.2 实际地址的产生	16
2.3 8086/8088 的寻址方式	17
2.3.1 操作数的种类	18
2.3.2 寻址方式	18
2.3.3 段更换和段跨越	20
2.4 8086/8088 指令系统	21
习 题	21
第三章 汇编语言	22
3.1 汇编语言的语句格式	22
3.1.1 字符集	22
3.1.2 汇编语句格式	22
3.2 汇编语言中数据的表示方法	25

3.2.1	数据在机内的表示方法和范围	25
3.2.2	汇编语言中数据书写形式	27
3.3	运算符号	28
3.3.1	算术运算符	28
3.3.2	逻辑运算符	28
3.3.3	关系运算符	28
3.3.4	分析算符	28
3.3.5	组合算符	30
3.3.6	字节分离算符 LOW 和 HIGH	30
3.3.7	记录专用算符	30
3.4	伪指令(汇编命令)	31
3.4.1	符号定义伪指令	31
3.4.2	数据定义伪指令	32
3.4.3	段定义伪指令	35
3.4.4	记录和结构定义伪指令	37
3.4.5	其他伪指令	39
习 题	40
第四章	8086/8088 汇编语言程序设计	43
4.1	程序设计的步骤	43
4.2	8086/8088 的指令系统	45
4.2.1	数据传送指令	45
4.2.2	算术运算指令	48
4.2.3	逻辑操作指令	54
4.2.4	移位操作指令	56
4.2.5	状态标志位操作指令	56
4.2.6	转移指令	57
4.2.7	循环控制指令	58
4.2.8	数据串操作指令	60
4.2.9	简单的 I/O 功能调用	63
4.3	顺序结构程序	65
4.4	分支结构程序设计	67
4.4.1	单分支结构程序设计	68
4.4.2	多分支结构程序设计	70
4.5	循环结构程序设计	74
4.5.1	循环程序的控制方法	75
4.5.2	单重循环程序举例	79
4.5.3	多重循环程序设计	81
习 题	83

第五章 子程序与宏汇编	90
5.1 子程序的引出	90
5.2 子程序(过程)定义伪指令	91
5.3 调用和返回指令	92
5.4 子程序设计方法	95
5.4.1 现场的保护和恢复	95
5.4.2 子程序说明文件	96
5.4.3 主程序与子程序之间的参数传递	96
5.5 子程序嵌套	103
5.6 递归子程序	106
5.7 程序的连接	109
5.7.1 程序连接伪指令	109
5.7.2 模块连接方法	110
5.8 条件汇编伪操作	112
5.9 宏伪操作	114
5.9.1 宏定义与宏结束伪操作命令	114
5.9.2 参数的使用	116
5.9.3 宏中的标号处理	118
5.9.4 宏嵌套	119
5.10 宏与子程序的区别	122
习 题	123
第六章 算术运算与代码转换	128
6.1 多字节加减运算	128
6.2 多字节整数乘除运算	129
6.2.1 一般整数乘法运算	129
6.2.2 多字节整数乘法运算	130
6.2.3 一般整数除法运算	132
6.2.4 多字节整数除法运算	132
6.3 十进制数的 ASCII 码串转换为二进制定点数	137
6.4 二进制定点数转换为十进制数的 ASCII 码串	140
习 题	142
第七章 输入输出与中断	143
7.1 输入输出指令	143
7.2 中断指令	144
7.3 外同步指令和空操作指令	145
7.4 CPU 与外设间的数据传送	146
7.4.1 输入输出接口	146
7.4.2 CPU 与外设之间的接口信号	147

7.4.3 CPU 与外设之间的数据传送方式	148
7.5 中断系统概述	148
7.5.1 中断请求与中断源	149
7.5.2 中断系统的功能	149
7.5.3 中断响应	150
7.6 8086/8088 的中断系统	150
7.6.1 外部中断	151
7.6.2 内部中断	152
7.6.3 中断向量表	153
7.7 8259A 中断控制器及其程序设计	154
7.7.1 8259A 的结构	154
7.7.2 8259A 的程序设计	155
7.7.3 8259A 在 IBM PC 系列机中的应用	159
7.8 中断程序设计方法	160
7.8.1 主程序设计	160
7.8.2 中断服务程序设计	160
7.8.3 中断服务程序设计中应注意的几个问题	161
7.9 中断程序设计举例	162
7.9.1 键盘程序设计	162
7.9.2 打印程序设计	166
7.10 IBM PC 的 BIOS 调用及 DOS 功能调用	171
7.10.1 BIOS 调用	171
7.10.2 DOS 中断和功能调用	173
习 题	177
第八章 80286 系统	179
8.1 80286 CPU 结构	179
8.1.1 80286 CPU 的内部结构	179
8.1.2 80286 CPU 内部寄存器	180
8.2 80286 系统存储器管理	182
8.2.1 80286 的操作方式	182
8.2.2 有关特权的概念	182
8.2.3 保护虚地址方式的存储管理	182
8.3 特权级	185
8.4 80286 的中断系统	187
8.5 80286 增强与扩充的指令	187
8.5.1 增强的指令	187
8.5.2 扩充的指令	188
习 题	191

第九章 80386/80486/Pentium 系统	192
9.1 80386 系统	192
9.1.1 80386 CPU 的内部结构	192
9.1.2 80386 寄存器	193
9.1.3 80386 存储器管理	196
9.1.4 80386 的中断和异常	198
9.1.5 80386 的寻址方式	201
9.1.6 80386 扩充的指令	202
9.2 80486 系统	205
9.2.1 80486CPU 结构	205
9.2.2 80486 的内存管理	206
9.2.3 80486 的片内高速缓存	206
9.2.4 80486 扩充的指令	206
9.3 Pentium 系统	206
9.3.1 Pentium CPU 结构	208
9.3.2 Pentium 扩充的指令	209
习 题	212
第十章 8087/80287/80387 程序设计	213
10.1 概 述	213
10.2 8087/80287/80387 的逻辑结构与运行	214
10.2.1 8087/80287/80387 的逻辑结构	214
10.2.2 8087/80287/80387 的运行	218
10.3 数据类型及格式	219
10.4 8087/80287/80387 的指令系统	221
10.4.1 数据传送指令	221
10.4.2 算术运算指令	223
10.4.3 比较指令	225
10.4.4 函数指令	226
10.4.5 常数指令	228
10.4.6 处理器控制指令	229
10.5 8087/80287/80387 程序设计举例	231
10.5.1 整数运算	231
10.5.2 实数运算	237
习 题	240
第十一章 MMX 的程序设计	241
11.1 MMX 简介	241
11.2 MMX 寄存器	242
11.3 检测 MMX 技术的存在	242

11.4	EMMS 指令	243
11.5	异常	243
11.6	MMX 指令	244
11.6.1	指令的句法	245
11.6.2	指令格式	245
11.6.3	表示法规则	245
11.6.4	MMX 指令	246
11.7	MMX 应用举例	269
11.7.1	Chroma Keying	269
11.7.2	矢量点乘积	270
11.7.3	矩阵乘积	271
11.7.4	使用 alpha 混合的图像分解	272
11.8	MMX 的程序设计	274
11.8.1	MMX 程序设计的步骤	274
11.8.2	程序设计举例	275
	习 题	280
附录 A	BIOS 调用说明	281
附录 B	DOS 功能调用说明	292
	参考文献	298

第一章 概述

本章要点

本章介绍了微型计算机的发展过程,微型计算机的特点及微型计算机系统的组成,分析了机器语言、汇编语言和高级语言的优缺点及适用场合,说明了汇编语言在微型计算机应用中的重要地位。

1.1 微型计算机的发展过程

微型计算机(Micro computer)简称微型机,从1971年美国 Intel 公司发明了第一台微处理器(Micro processor)以来,已经经历了20多年的历史,发展了五代产品。

第一代(1971~1973年)是低档的4位微处理器 Intel 4004 及由它组成的微型计算机 MCS-4。它使用机器语言和汇编语言,基本指令执行时间为 $10\sim 15\mu\text{s}$ 。虽然第一代微型计算机在结构和性能上还很不完善,但它获得了价格上的优势。微处理器和微型计算机的出现,标志着计算机进入了一个崭新的发展阶段。

第二代(1974~1977年)是8位微处理器和微型计算机。初期产品有 Intel 公司的 8080 和 Motorola 公司的 MC6800,1976年 Zilog 公司生产了性能较高的 Z-80,以这三种微处理器为 CPU 的微型计算机使用得较为普遍,指令系统比较完善,已具有典型的计算机体系结构、中断功能和 DMA 控制功能,除采用机器语言、汇编语言外,还逐渐配了 BASIC、FORTRAN 等高级语言及相应的解释程序、编译程序。

第三代(1978~1984年)是16位微处理器和微型计算机。初期产品有 Intel 公司推出的16位微处理器 Intel 8086,接着 Motorola 公司推出了 MC68000,Zilog 公司推出了 Z-8000。这三种微处理器是第三代微处理器的代表产品,也是国际市场最流行的三种16位微处理器。后来 Intel 公司又推出了新型的 80286 微处理器。第三代16位微处理器比第二代8位处理器的速度快 $2\sim 5$ 倍,每秒可执行100万条以上(1MIPS)的指令,赶上和超过了小型计算机。第三代微型计算机配备多种高级语言、完善的操作系统、大型的数据库。在事务管理、实时数据处理和实时控制领域中开辟了广泛的应用前景。

第四代(1985~1992年)是32位微处理器和微型计算机。初期产品有 Intel 公司推出的32位微处理器 80386,每秒可执行300万条以上($3\sim 4\text{MIPS}$)的指令,接着 Motorola 公司推出了 MC68020,后来 Intel 公司又推出了 80486。用32位微处理器构成的微型计算机系统的速

度和性能大为提高,每秒可执行 5000 万条以上(54MIPS)的指令,可靠性也大大增加,其功能足以同高档的小型计算机相匹敌。

第五代(1993 年以后)微处理机是 Intel 公司推出 Pentium 微处理机,它是继 8086/8088、80286、80386 和 80486 之后 X86 家族的又一个新成员,按理说应叫作 80586,它采用了全新的体系结构,运用超标量流水线设计,同时,还在原有 80486 体系结构的基础上做了一些改进,其总体性能大大超过了 80486,每秒可执行 10000 万条以上(112MIPS)的指令,Pentium 的推出,迎合了用户在图形图像、实时图像处理、语音识别和 CAD/CAM 等方面对高性能工作平台的需求。

第六代(1995 年末)微处理器是 Intel 公司推出的 Pentium Pro 微处理器(中文名字高能奔腾),它是针对 32 位软件设计的具有 RISC 核心但仍与 X86 指令相兼容的微处理器,运算速度可达 300MIPS。为了提高 Pentium Pro 的性能,Intel 公司采用了 3 路标量体系结构和 14 级流水线使得 CISC 的指令更加 RISC 化,并且允许动态指令执行。另外在 CPU 内部除了集成有一级 Cache 外,还增加了 512K 的二级 Cache,该 Cache 与 CPU 之间通过一条 64 位的专用总线相连,提高了 CPU 与 Cache 之间的数据传输速度。Pentium Pro 在运行 16 位和 16 位、32 位混合代码进程的性能与 Pentium 差不多,但运行纯 32 位代码时要快得多。

1997 年初,Intel 公司推出了编号为 P55C 的具有 MMX(Multi Media eXtension,多媒体扩展)指令集的 Pentium 处理器,以支持越来越多的多媒体应用。在这种处理器中,除了引进了新的多媒体指令外,还增加了 8 个 64 位的寄存器和 4 种新的数据类型。Pentium Pro 的 MMX 版本命名为 Klamath,即 Pentium II 也于 1997 年的上半年推出。1999 年上半年,Intel 又推出了 Pentium III,Pentium III 比 Pentium II 增加了 70 条新指令,增强了 3-D 数据处理能力和提高了浮点运算的速度。

目前我国国内流行的微型计算机多数是 Pentium、Pentium II 和 Pentium III 微处理器组成的微型计算机系统。微型计算机在我国各个领域已经得到了广泛的应用,对计算机科学和其他学科产生了变革性的影响,并在国民经济的建设中占有越来越重要的地位。

1.2 微型计算机的特点

微型计算机除具有电子计算机的运算速度快、计算精度高、有记忆能力和逻辑判断能力等特点外,它还有自己的特点:

1. 体积小,重量轻

一般微处理机芯片的面积只有绘图橡皮那样大,其重量只不过几十克,由微处理机组成的微型计算机系统,可以全部组装在一块印刷电路板上。

2. 简单灵活、可靠性高、功耗低

微型计算机硬件系统,可以根据不同的需要灵活地组成各种不同规模的微型计算机系统,由于微型计算机采用大规模集成电路,很多功能组装在一个芯片上,外部连线、开关都大为减少,功耗得到降低,同时又由于功耗小,发出热量少,又使微型计算机的可靠性大为提高。

3. 工作条件要求低

微型计算机系统对工作环境的条件要求不高,不像中小型计算机系统那样要求有专门

的计算机房、净化间,对室内温度都有严格的要求,而微型计算机在普通办公室或家庭居室都能正常工作。

4. 性能价格比高

随着微型计算机的发展,微型计算机系统的硬件越来越可靠,软件越来越丰富,功能也越来越完善,而价格越来越低,一般几千元就可购买一台,因此,一般单位都购有几台、几十台微型计算机系统,在办公室自动化、科学计算、工业控制等方面发挥着越来越重要的作用,在我国已开始进入家庭。

1.3 计算机系统

相关事物组成的集合体称为系统,组成系统的事物称为该系统的元件或部件,计算机系统由硬件子系统和软件子系统组成。

1.3.1 硬件子系统

硬件子系统是指组成计算机系统的所有电子的、机械的、光学的和磁性的元部件。硬件子系统通常包括中央处理器、主存储器、接口装置、外部设备(输入设备,输出设备,外存储器)、外围设备(A/D转换器、D/A转换器、开关量输入/输出)。

计算机硬件系统的示意图如图 1-1 所示。

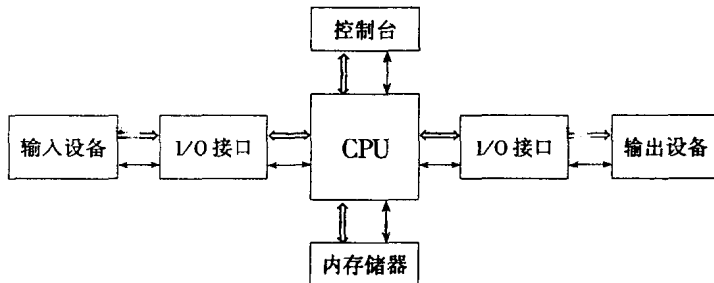


图 1-1 计算机硬件子系统示意图

1. 中央处理器

中央处理器通常由算术逻辑部件、寄存器组、标志寄存器和控制部件等组成。

算术逻辑部件完成算术和逻辑运算。

寄存器组由若干寄存器组成,有的用于寄存参加运算的数据、运算后的结果;有的用作控制指令执行的指令计数器;有的用作堆栈指针寄存器等。

标志寄存器用于保存运算结果的状态和 CPU 用的逻辑控制标志,以使 CPU 根据这些标志位进行判断,决定程序的执行顺序。

控制部件用于保存机器指令,对指令进行译码,产生控制各个部件的信号。中央处理机是计算机硬件系统的核心。

2. 存储器

存储器分内存储器和外存储器,简称为内存和外存。

内存又称主存储器,用于存储计算机当前正在运行的程序、正在处理的原始数据、中间结果及最终结果等。

主存储器能存放信息的容量为内存容量,通常用存储空间多少来表示。存储空间的基本计量单位为字节(Byte),1个字节由8位二进制位(bit)组成,每1024个字节称为1KB,每1024KB称为1MB。从使用计算机的角度上说,内存空间越大越好,但由于受到技术上的限制及价格等方面因素的影响,内存空间不能太大,以Z-80为CPU的微型机的内存空间最大为64KB,以8086/8088为CPU的微型机内存空间最大为1MB(1024KB),以80286为CPU的微型机内存空间最大为16MB,以80386和80486为CPU的微型机内存空间最大为4096MB,以Pentium为CPU的微型机内存空间可达16MB,32MB,64MB,128MB。

每个存储空间又称为一个单元,每个单元有一个惟一的编号,称为地址,向存储空间存数据或从存储空间读取数据时,都必须指出它的地址。

外存又称为辅助存储器,如磁盘(硬磁盘和软磁盘)和磁带等。以8086/8088为CPU的微型机一般配有硬磁盘(温氏盘)和软磁盘,硬磁盘容量通常为40MB、80MB、120MB、200MB或300MB等。软磁盘有两种:一种是5.25"软盘,每个盘片的容量为360KB或1.2MB,一种是3.5"盘,每个盘片的容量为720KB或1.44MB。当要存储的信息较多时,可使用多片软盘,从这个意义上说,它的容量是无限的,所以又称为“海量”存储器。80386以上的微型机还配置有光盘,光盘的容量为650MB。

3. 接口装置

人们和计算机交换信息都是通过外部设备进行的,外部设备一般是机械设备,其工作速度与中央处理器的工作速度比起来低得多,而且不同的外部设备传输数据的编码格式、所需要的控制信号种类、数量也不相同。因此,外部设备与CPU之间不能直接交换信息,需要有一个具有若干个寄存器和逻辑控制电路部件,作为二者交换信息的缓冲部件,该部件称为接口装置,简称接口。

接口又分为并行接口和串行接口。在并行接口中,每一位数据都有自己的通路(数据传输线),譬如,并行传送一个字节,需要8根数据传输线;在串行接口中,数据是一位接一位传输的,只要一根传输线,为了使接口电路具有通用性和灵活性,其中设有用于指定工作方式的控制寄存器,用户可以根据连接设备的种类及对控制信号的要求,对接口电路中的控制寄存器通过指令进行设置,这样的接口电路称为可编程接口电路。接口电路的通用性越强,内部的控制寄存器越多,需设置的信号越多,编程就越复杂。

不同的厂家为自己生产的计算机研制了配套的并行、串行接口电路,如Z-80系列的接口电路有PIO,SIO,CTC等,Intel系列的接口电路有8255,8251,8253,8259等等。

4. 输入输出设备

输入输出设备又称外部设备,是用来实现人与计算机交换信息的装置。输入设备是用来向计算机的主存储器或CPU送入程序或数据的装置。常见的输入设备有键盘,读卡机,光笔,数字化仪等;输出设备是用来将计算机处理的结果以某种形式输出给用户的装置,常见的输出设备有显示器,打印机,绘图仪和穿卡机等。

不同的计算机配置的外部设备种类、数量、所需的控制信号、数据编码格式等都不统一,使用时需参看有关技术说明书。

5. 外围设备

当计算机用于控制时,现场信号要进行输入/输出,这些信号分为模拟量和开关量,为区别一般的外部设备,完成这些信号输入/输出的设备,称为外围设备。

模拟量用于控制计算机,大量的现场信号经过传感器把非电量(例如温度,压力,流量,位移等)转换为电量,并经放大即得模拟电压和电流。这些模拟量必须先经过 A/D 转换,变为数字量才能送入计算机(位数由 A/D 转换精度确定)进行运算处理;计算机输出的控制信号,也必须先经过 D/A 转换,变为模拟量才能去控制执行机构。

开关量是一些两个状态的量,如计算机控制电机的运转与停止、电子开关的接通与断开、阀门的打开与关闭等等。这些量只要用一位二进制数即可表示,这就是开关量的输出。但有时在控制过程中也需要开关量的输入。字长为 8 位的机器一次可输入或输出 8 个这样的开关量。

1.3.2 软件子系统

软件子系统是指为了充分发挥计算机硬件子系统的功能、方便用户使用计算机、提高计算机系统效率而编制的各种程序。

软件子系统对于计算机系统来说是极为重要的,在硬件子系统已经确定的情况下,计算机系统功能的强弱,应用范围的大小,使用的方便性、灵活性等完全取决于软件功能的强弱。

软件子系统由系统软件和应用软件组成,而系统软件又分为三类:面向用户的软件,面向计算机维护人员的软件和面向计算机本身的软件;应用软件是用户为解决自己的问题而编制的软件。

1. 面向用户的软件

这类软件有利于用户使用计算机解决自己的问题,还有利于用户编制、调试、装配自己的应用程序。

(1) 语言加工软件。它用来把用户用各种计算机语言编制的源程序转换为计算机硬件能直接处理的机器代码(目标程序)。如:汇编程序、编译程序、解释程序等。

汇编程序是将汇编语言程序转换为目标程序的软件。

编译程序是将通用语言源程序转换为目标程序的软件。任何一种通用语言都有自己的编译程序。如:FORTRAN 语言有 FORTRAN 编译程序,Pascal 语言有 Pascal 编译程序等等。

(2) 辅助加工软件。用户借助辅助加工软件来编辑、修改自己的源程序,装配、连接、调试自己的目标程序,这类软件有编辑程序(行编辑程序,全屏幕编辑程序)、连接程序、纠错程序等等。

行编辑程序:用户利用此程序将自己的源程序送入内存,逐句逐字进行编辑操作,删除、插入、查找、替换字或字符串。IBM PC 微机的行编辑程序名是 EDLIN。

全屏幕编辑程序:其编辑功能与行编辑程序相同,只不过它可以使光标在整个显示屏幕上任意移动,对光标指示的任何字符进行各种编辑操作,它比行编辑程序更灵活、更方便。在 IBM PC 上有很多全屏幕编辑程序,如 WS,WPS,XE 等等。

连接程序:把经过汇编或编译产生的目标程序(一个或几个)连接在一起成为可执行的机器语言程序,以备装入运行。IBM PC 上的连接程序名为 LINK。

纠错程序:用户借助此程序来控制自己程序的执行,设置断点、逐条跟踪、检查每条指令或某段程序运行的结果、检查修改寄存器或存储单元的内容等。IBM PC 上的纠错程序名为 DEBUG。

2. 面向计算机维护人员的软件

这类软件主要有诊断调试程序、日常事务受理程序等。