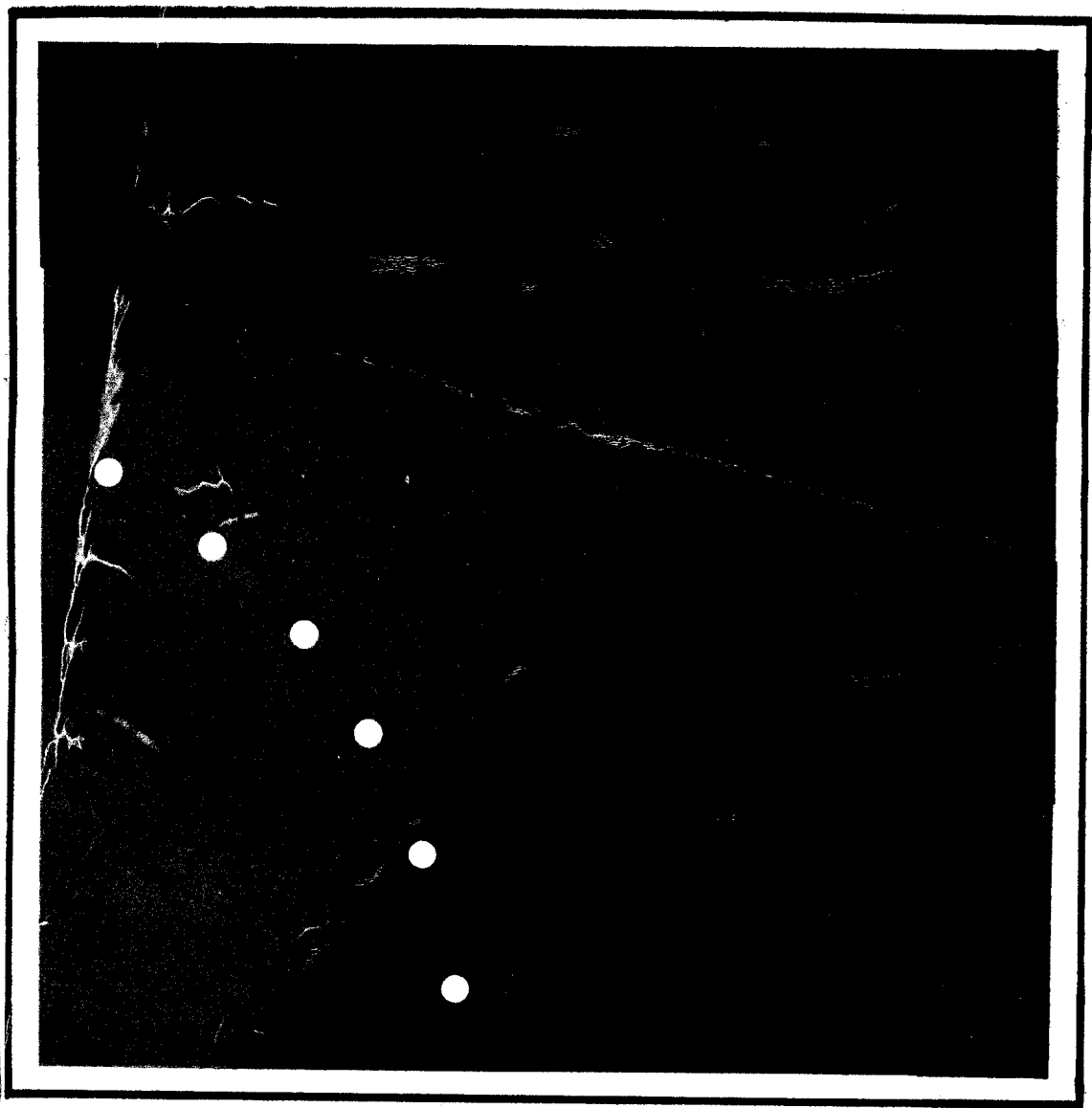


GH

中等专业学校 规划教材
工科电子类

汇编语言程序设计

武马群



电子科技大学出版社

[川]新登字 016 号

内 容 简 介

本书以 IBM PC 微型计算机为背景,通过全面介绍汇编语言程序设计来阐明程序设计的基本概念和方法。主要包括:电子计算机基础知识;计算机指令系统;汇编语言程序设计;上机操作与实验等。此外,对 80286 的指令系统及编程做了一个简要的说明。

本书通俗易懂,内容切合教学实际。在内容的安排上,以保证中专教学水平为前提,由浅入深、循序渐进。

本书既可作为中专电子计算机硬软件专业的教材,也可以供具有一般逻辑电路知识的人员自学之用。

中等专业学校教材 汇编语言程序设计

武马群

*

电子科技大学出版社出版

(成都建设北路二段四号)邮编 610054

电子科技大学出版社印刷厂印刷

四川省新华书店经销

*

开本 787×1092 1/16 印张 20.75 字数 505 千字

版次 1994 年 11 月第一版 印次 1994 年 11 月第一次印刷

印数 1—15000 册

中国标准书号 ISBN 7-81016-822-3/TP·95

定价:13.90 元

出版说明

根据国务院关于高等学校教材工作的规定,我部承担了全国高等学校和中等专业学校工科电子类专业教材的编审、出版的组织工作。由于各有关院校及参与编审工作的广大教师共同努力,有关出版社的紧密配合,从1978~1990年,已编审、出版了三个轮次教材,及时供给高等学校和中等专业学校教学使用。

为了使工科电子类专业教材能更好地适应“三个面向”的需要,贯彻国家教委《高等教育“八五”期间教材建设规划纲要》的精神,“以全面提高教材质量水平为中心,保证重点教材,保持教材相对稳定,适当扩大教材品种,逐步完善教材配套”,作为“八五”期间工科电子类专业教材建设工作的指导思想,组织我部所属的九个高等学校教材编审委员会和四个中等专业学校专业教学指导委员会,在总结前三轮教材工作的基础上,根据教育形势的发展和教学改革的需要,制订了1991~1995年的“八五”(第四轮)教材编审出版规划。列入规划的,以主要专业主干课程教材及其辅助教材为主的教材约300多种。这批教材的评选推荐和编审工作,由各编委会或教学指导委员会组织进行。

这批教材的书稿,其一是从通过教学实践、师生反映较好的讲义中经院校推荐,由编审委员会(小组)评选择优产生出来的,其二是在认真遴选主编人的条件下进行约编的,其三是经过质量调查在前几轮组织编写出版的教材中修编的。广大编审者、各编审委员会(小组)、教学指导委员会和有关出版社,为保证教材的出版和提高教材的质量,做出了不懈的努力。

限于水平和经验,这批教材的编审、出版工作还可能有缺点和不足之处,希望使用教材的单位,广大教师和同学积极提出批评和建议,共同为不断提高工科电子类专业教材的质量而努力。

电子工业部电子类专业教材办公室

前 言

本教材是由机械电子工业部中专计算机专业教学指导委员会评选审定,并推荐出版的。

该教材由北京无线电工业学校武马群编写,贵州无线电工业学校黄大胜主审。编审者都是依据中专计算机教材编委会审定的编写大纲进行编写和审阅的。

本课程的参考学时数为 85 学时。其主要内容是以 IBM PC 微型计算机为背景,介绍汇编语言程序设计的基本概念和方法。具体有以下几个部分:基础知识;8086/8088 指令系统;简单程序设计;循环程序设计;子程序设计;高级汇编语言技术;输入/输出程序设计;BIOS 中断和 DOS 功能调用。此外,对模块化程序设计,80286 指令系统及编程进行了适当的介绍。最后,对 IBM PC 机的汇编上机过程以及本课程的实验和课程设计做了说明。使用本教材时应注意对 8086/8088 指令系统的处理。为了保证教材的系统性和完整性,将机器的全部指令集中在第 2 章中叙述。但是,考虑到大量的指令集中讲授会使学生难以掌握,因此,在第 3 章以后的各章中,采取了分散引用指令方法,即先从一些基本的指令出发讲述程序设计的概念和方法,随着内容的进展,每 1 章都用到一些新的指令。这样安排便于任课教师灵活地处理教材内容。例如,在讲授第 2 章的内容时,可以把重点放在给学生建立起完整的指令系统概念上,并介绍一部分 8088 指令。而其他的指令则分别在以后的章节中进行介绍。为适应指令的分散启用,第二章中打破了传统的 8088 指令分类,而按照启用这些指令的顺序来分类加以说明,以便与后面各章的内容安排相呼应。

在本教材的编写过程中,得到曾玉崑、周岳山等同志的多方指导和帮助,在此表示诚挚的感谢。由于编者水平有限,书中难免还存在一些缺点和错误,敬请读者批评指正。

编者

1993 年 10 月

目 录

第1章 基础知识

§ 1.1 引言	(1)
§ 1.2 电子计算机的基本组成	(2)
1.2.1 一般计算机的组成结构	(2)
1.2.2 微型计算机简介	(4)
§ 1.3 存储程序控制概念	(6)
§ 1.4 计算机中数的表示	(7)
1.4.1 计算机中常用的数制	(7)
1.4.2 各种数制之间的转换	(12)
1.4.3 计算机中数的表示	(16)
§ 1.5 信息编码	(22)
1.5.1 ASCII 码	(22)
1.5.2 EBCDIC 码	(22)

第2章 8086/8088 指令系统

§ 2.1 8088 微处理器结构	(26)
2.1.1 概述	(26)
2.1.2 通用寄存器	(26)
2.1.3 控制寄存器	(27)
2.1.4 段寄存器	(30)
§ 2.2 微型计算机中数的表示	(31)
§ 2.3 指令的编码格式和寻址方式	(37)
2.3.1 指令的编码格式	(37)
2.3.2 寻址方式	(39)
§ 2.4 8088 指令系统	(45)
2.4.1 指令的分类	(45)
2.4.2 8088 指令的编码格式和执行时间	(46)
2.4.3 8088 指令系统说明	(50)

第3章 简单程序设计

§ 3.1 程序设计概述	(88)
§ 3.2 汇编语句格式	(89)
3.2.1 名字项	(89)
3.2.2 操作码项	(90)
3.2.3 操作数项	(90)
3.2.4 注释项	(92)
§ 3.3 伪操作	(92)
3.3.1 数据定义及存储器分配伪操作	(92)

3.3.2	等价伪操作	(94)
3.2.3	段定义伪操作	(95)
3.3.4	命名和程序结束伪操作	(97)
3.3.5	指定地址伪操作	(97)
3.3.6	基数控制伪操作	(98)
§ 3.4	顺序程序设计	(98)
§ 3.5	算法和框图	(102)
3.5.1	算法	(102)
3.5.2	框图	(104)
§ 3.6	分支程序设计	(105)
§ 3.7	程序举例	(111)
3.7.1	代码转换	(111)
3.7.2	十进制算术运算	(114)
3.7.3	双精度数运算	(116)
第4章 循环程序设计		
§ 4.1	循环程序的结构	(118)
§ 4.2	单重循环程序	(120)
4.2.1	循环次数已知的单重循环程序	(120)
4.2.2	循环次数未知的单重循环程序	(124)
§ 4.3	多重循环程序	(128)
§ 4.4	程序举例	(134)
第5章 子程序设计		
§ 5.1	主程序和子程序	(143)
§ 5.2	子程序的调用和返回	(146)
§ 5.3	寄存器内容的保护与恢复	(151)
§ 5.4	主程序和子程序之间的参数传送	(152)
5.4.1	用寄存器传送参数	(152)
5.4.2	通过堆栈传送参数或参数地址	(153)
5.4.3	通过共同数据存储区传送参数或参数地址	(154)
§ 5.5	子程序的嵌套与递归	(156)
5.5.1	子程序的嵌套	(156)
5.5.2	子程序的递归	(160)
§ 5.6	DOS 系统功能调用	(167)
§ 5.7	程序举例	(167)
第6章 高级汇编语言技术		
§ 6.1	汇编程序	(179)
6.1.1	宏汇编程序(MASM)的功能	(179)
6.1.2	汇编过程	(180)
6.1.3	汇编列表文件	(181)
6.1.4	交叉参考列表文件	(186)

§ 6.2 宏汇编	(186)
6.2.1 宏定义和宏调用	(186)
6.2.2 与宏指令有关的其他伪操作	(187)
6.2.3 宏指令与子程序的区别	(190)
§ 6.3 重复汇编	(190)
6.3.1 重复伪操作	(190)
6.3.2 不定重复伪操作	(191)
§ 6.4 条件汇编	(192)
第7章 输入/输出程序设计	
§ 7.1 I/O 设备及端口	(195)
7.1.1 I/O 设备	(195)
7.1.2 I/O 端口	(199)
§ 7.2 程序直接控制 I/O 方式	(200)
7.2.1 直接 I/O 方式	(200)
7.2.2 查询 I/O 方式	(202)
§ 7.3 中断 I/O 方式	(205)
7.3.1 中断处理的一般过程	(205)
7.3.2 中断 I/O 程序设计举例	(212)
7.3.3 中断优先级和中断嵌套	(215)
§ 7.4 软件中断	(218)
§ 7.5 程序举例	(220)
第8章 BIOS 中断和 DOS 功能调用	
§ 8.1 概述	(227)
§ 8.2 显示器输出	(229)
8.2.1 视频显示标准	(229)
8.2.2 CGA 字符显示编程	(233)
8.2.3 CGA 图形显示编程	(238)
§ 8.3 键盘输入	(242)
§ 8.4 打印机输出	(245)
§ 8.5 串行通信口输入/输出	(248)
8.5.1 并行传送与串行传送	(248)
8.5.2 异步通信控制规程	(249)
8.5.3 串行通信口输入/输出操作	(250)
第9章 模块化程序设计	
§ 9.1 程序设计的步骤	(253)
9.1.1 分析问题	(253)
9.1.2 确定算法和输入/输出方式	(253)
9.1.3 程序结构的设计	(254)
9.1.4 编写程序	(254)
9.1.5 调试运行	(254)

§ 9.2 模块化程序设计概述.....	(254)
9.2.1 模块化程序设计.....	(255)
9.2.2 结构化程序设计.....	(256)
§ 9.3 8088MASM 的多模块编程问题	(258)
9.3.1 多模块程序中段的连接.....	(258)
9.3.2 模块间的交叉访问和信息传送.....	(262)
§ 9.4 模块化程序设计举例.....	(264)
第 10 章 80286 指令系统及编程	
§ 10.1 80286 微处理器结构	(273)
§ 10.2 80286 的中断系统	(276)
§ 10.3 80286 增强与增加的指令	(277)
10.3.1 80286 增强的指令	(277)
10.3.2 80286 增加的指令	(278)
§ 10.4 80286 的编程	(281)
第 11 章 上机操作与课程实验	
§ 11.1 汇编语言上机所必备的软件	(287)
§ 11.2 汇编语言程序的上机操作过程	(288)
§ 11.3 源文件的建立	(289)
§ 11.4 源文件的汇编	(293)
§ 11.5 目标文件的连接	(295)
§ 11.6 用户程序的运行	(296)
§ 11.7 调试程序 DEBUG 的主要命令	(297)
§ 11.8 实验与课程设计	(300)
11.8.1 实验	(301)
11.8.2 课程设计	(302)
附录一 8086/8088 指令系统编码格式	(303)
附录二 中断向量地址一览表	(313)
附录三 DOS 功能调用	(314)
附录四 BIOS 中断	(319)
附录五 IBM PC ASCII 码字符表	(323)
附录六 常用的程序流程图符号	(324)

第 1 章 基础知识

§ 1.1 引言

电子计算机是 20 世纪 40 年代的一项伟大的科学发明。电子计算机的出现,给科学技术带来了日新月异的变化,有力地推动了社会生产力的发展,大大加快了人类进入信息社会的进程。

长期以来,为了满足科学和生产发展的需要,人们创造了各种各样的计算工具,如:算盘、计算尺、手摇计算机等等,运用这些计算工具进行计算,比起手工计算来有很大的进步。但这些计算工具的运算速度、工作效率由于受机械运动的限制,远不能满足现代科学技术发展的需要。随着电子工业的飞速发展,人们开始了电子计算机的研制工作。1946 年美国研制成功了世界上第一台电子计算机 ENIAC。它是由一万九千多个电子管和几十万个其他电器元件组成的,占地面积 167m²,体积庞大,耗电量达 150kw,但它的运算速度比过去所有的计算工具有相当大的提高,达到了每秒钟 5000 次。经过四十多年的努力,人们现在研制的电子计算机的运算速度每秒可达几千万次,甚至几亿次,其他方面的性能也比初期的电子计算机大大提高了。

电子计算机是由各种电气元件组成的。它能迅速而精确地对信息进行搜集、整理、变换、存储和处理,而这些工作过去都要由人的脑力劳动去完成。所以,如果说蒸汽机的发明把人类从繁重的体力劳动中解放出来,那么电子计算机的出现就意味着把人类从复杂的脑力劳动中解放出来。正因为如此,人们才投入了大量的人力、物力去研制电子计算机,从而形成了一门重要的学科,并得到飞速的发展。现在计算机的应用已经远远超出了其名字的含义,伸展到了国计民生的各个方面。电子计算机仅仅作作为一种高速计算工具的时代早已过去,它已成为各门科学技术向前发展的先导,在数据处理、自动控制、办公自动化等领域中都起着支柱作用。因此,计算机的发展水平和普及程度,已成为衡量一个国家科学技术发展水平的重要标准之一,也是一个国家现代化程度的重要标志。

一台完整的电子计算机系统由两部分构成,即“硬件”和“软件”。所谓硬件就是由电磁元、器件和机械装置组成的所有计算机设备,它们也称为硬设备。软件则是指为方便用户和发挥计算机效率而设计的各种系统程序和应用程序以及和这些程序有关的资料(包括程序结构的注释、使用说明文件等等),它们也称为软设备。在计算机的发展过程中,硬件为软件的发展奠定了基础,而软件的发展对硬件又提出了更高的要求。它们相互促进,使电子计算机向着更加完善的方向发展。在电子计算机的应用方面,软件在一定程度上弥补了硬件功能的不足,从而扩大了电子计算机的应用范围。

电子计算机之所以能够代替人去做某些工作,主要是因为它具有高速自动计算和逻辑判断的能力。而这种能力是基于“存储程序”这一重要原则的。计算机的每一操作完全是在

程序控制之下进行的,因此,对于一台通用电子数字计算机来说,在一定的条件下,输入了什么样的程序,它就可以做什么样的工作。好比一个人要做好某项工作,必须先学习这项工作的方法,了解工作性质及工作目的一样,计算机做某项工作时,也必须先对其交待工作方法和运算步骤,遇到问题如何判断,对不同情况如何处理等等。所有这些都要先由人编出具体的程序,然后送入计算机,计算机执行这个程序就可按照人们规定的方法、步骤进行工作。所以,电子计算机能否胜任某项工作,很重要的是人们能否为它编制出完成这项工作的正确的程序。程序设计技术是掌握软件和运用电子计算机的基础。本书的目的是使读者掌握程序设计的基本知识、基本方法和基本技巧。

§ 1.2 电子计算机的基本组成

1.2.1 一般计算机的组成结构

一般计算机的组成结构如图 1-1 所示。图中“ $\diamond$ ”表示代码传送路线,“ $\rightarrow$ ”表示控制路线。

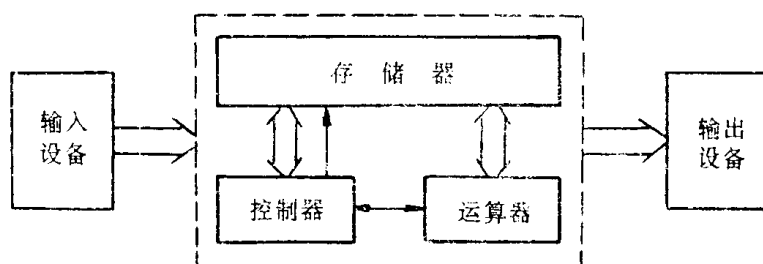


图 1-1 一般计算机的组成结构

下面分别介绍图中几个重要部件的功能。

一、运算器

运算器是对数据进行运算的部件,它能够快速地对数据进行加、减、乘、除等基本算术运算以及“与”、“或”、“非”等逻辑运算。

在运算过程中,运算器不断地得到由存储器提供的数据,进行计算后把结果(包括中间结果)送回存储器保存起来。整个运算过程是在控制器统一指挥下,按程序中编排的操作顺序有规律地进行的。

运算器主要由算术逻辑单元、寄存器以及一些控制数据传送的电路组成。算术逻辑单元是运算器中实现算术逻辑运算的电路;寄存器是运算器中的数据暂存器。运算器中往往设置多个寄存器,每个寄存器都能够保存一个数据。寄存器可以直接为算术逻辑单元提供参加运算的数据,运算的中间结果也可以保存在寄存器中。这样,一个简单的运算过程就可以在运算器内部完成,不需频繁地与存储器打交道,从而提高了运算速度。

二、存储器

存储器是用来存储大量信息(指令和数据的二进制代码)的部件。

在解题之前,先将程序和数据从计算机外送入存储器保存起来。在计算过程中,计算机不断从存储器取出指令送往控制器,控制器分析和解释指令的含义,并据此向运算器或其他部件发出相应的命令,控制各部件执行指令规定的操作。与此同时,存储器不断地向运算器提供运算所需要的数据,保存从运算器送来的计算结果。

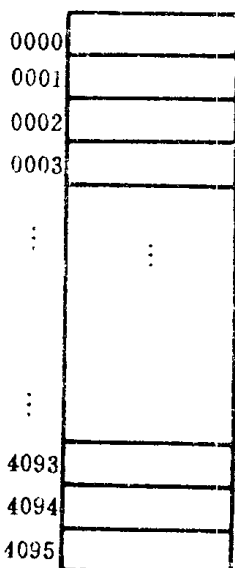


图 1-2 存储器单元编号示意图

存储器分为内存储器(简称“内存”)和外存储器(简称“外存”)。内存储器是计算机用来存放当前求解题目的计算程序及数据的主要存储器,故又称为主存储器。它和运算器、控制器有直接的联系。内存储器一般有磁心存储器和半导体存储器两种。整个内存储器由许多个存储单元构成,每个单元可以存放一串二进制代码。为区分存储器中不同的单元,需把它们按照一定的顺序进行编号,如图 1-2 所示。

图中的每一个小格表示一个存储单元,左边对应的数字就是这个单元的编号,按照这些编号,我们把各个存储单元分别称为 0 号单元,1 号单元,……,4095 单元。人们形象地把这些存储单元的编号称为“地址”。犹如每个人的家庭住址,它指出了每个人的居住位置,我们可以按照住址去寻找某人一样,存储器也是以“按地址存取”代码的方式进行工作的。当一个代码送到某个存储单元中,或从某个单元中取出一个代码时,先要给出存储单元的地址,由存储器找到相应的存储单元之后,才能把代码存进去或取出来。

存储器有一个重要的特点:当从某一单元取出(又称为“读出”)时,这一单元中存储的代码不变,但是,如果向这一单元存入(又称为“写入”)新代码,则原来存储的代码就被这新的代码所代替。这相当于录音磁带,放音时磁带上的内容不变,而录音时原有的内容就被新的内容所代替。存储器的这个特点,可以用一句话来概括,即“取数不变,存数更新”。

由于受到计算机某些因素的限制,内存的容量(存储单元的数量)不能做得很大。为了加强计算机的处理能力,人们设计了许多形式的外存储器,常见的有磁盘、磁带等,它们的存储容量可以做得很大,能够有效地提高计算机存储信息的能力。一台计算机可配备多种形式的外存储器,当这些外存与内存配合起来使用时,可以使存储容量变得“无穷大”。外存储器同样具有“取数不变,存数更新”的特点,但外存储器存取代码的方式与内存不同,这里就不再介绍了。

三、控制器

控制器的主要作用是使计算机能自动地执行程序。控制器从存储器顺序地取出指令,并对指令代码进行翻译,然后向各部件发出相应的命令,使指令规定的操作得以执行。因此,控制器是统一指挥和控制计算机各部件进行工作的中央机构。它一方面向各部件发出执行指令的命令,另一方面又接受“执行部件”向控制器发回的有关指令执行情况的“反馈信息”,如

运算器向控制器报告计算结果的大小是否超出预定界限等等。这种由运算器、存储器、输入和输出设备发回控制器的“反馈信息”，将对计算机的下一部工作状态产生影响，控制器将根据这些反馈信息来决定下一步发出哪些操作命令。

控制器主要由指令寄存器、指令译码器、指令计数器以及其他一些电路组成。当计算机执行程序时，指令计数器保存着要执行的下一条指令的地址，控制器根据这个地址，从内存中取出指令并送到指令寄存器。指令译码器对指令寄存器中的指令代码进行翻译后，发出各种相应的操作命令，指挥计算机的有关部件进行工作。

指令计数器是控制器中的一个重要部件，它的功能是指示程序的执行过程。指令计数器可以自动修改其自身的内容，这使得计算机能够自动地执行程序。指令计数器内容的修改方式一般有两种。一种是指令计数器本身自动加 1，即以指令计数器的内容为地址，对内存进行一次读操作后，指令计数器就自动加 1，为取下一条指令做好准备。程序一般是按指令的编排顺序存放在内存中的。也就是说，指令的存放地址是顺序递增的。在计算机工作时，大部分时间都按顺序执行程序中的指令，指令计数器的加 1 功能就是为此而设计的。这样的加 1 操作一直进行下去，直到程序执行完毕。另外一种方式是通过某些指令的操作来修改指令计数器的内容，以实现程序本身对指令执行顺序的调整。

四、输入输出设备

输入输出设备是计算机和使用者相互联系的“窗口”。

输入输出设备的工作，一方面受主机的控制，另一方面又受计算机使用人员的操纵，因此它们的工作情形与其他部件有所不同。

输入设备能把数据和程序转换成电信号，并通过计算机的接口电路将这些信号顺序地送入计算机的存储器中。键盘是广泛使用的输入设备，用户可以直接从键盘上输入程序或数据信息，使人和计算机直接进行联系。

输出设备能把计算机所产生的结果送往机外，把从存储器取出的电信号转换成其他形式输出。如把数字符号印刷在纸上或显示在屏幕上等。它和输入设备的区别在于代码进出的方向不同。常见的输出设备有行式打印机、绘图仪及屏幕显示设备等。

软盘驱动器和磁带机既可以看作输入设备也可以看作输出设备。微型计算机 IBM PC 及其兼容机所使用的 $5\frac{1}{4}$ 英寸的软磁盘，可以存储 360KB 或 1.2MB。磁带可以存储大容量的信息，从几十兆字节到一百多兆字节，但从磁带上输入数据和输出数据到磁带上的速度相对磁盘方式要慢些。

为了使用方便，通常一台计算机要配备若干种输入输出设备。这些设备的工作速度各不相同，所需要的控制信息也不一样。因此，需要有专门控制输入输出过程、协调外部设备与主机之间的工作过程的部件，即输入输出控制部件，它也是计算机组成结构中不可缺少的重要部件。

通常，人们把运算器、内存储器和控制器合在一起称为计算机的主机，而把各种输入输出设备和外存储器统称为计算机的外围设备，简称外设。

1.2.2 微型计算机简介

微型计算机是 1971 年出现的，是大规模集成电路发展的产物。人们把计算机的核心部

件控制器、运算器集中起来做在仅有几十平方毫米的硅片上,并称之为微处理器,又称中央处理器 CPU。以 CPU 为中心,用存储器、输入输出接口、时钟发生器等大规模集成电路组成的一种电子计算机。就称为微型计算机。

微型计算机的组成结构如图 1-3 所示。

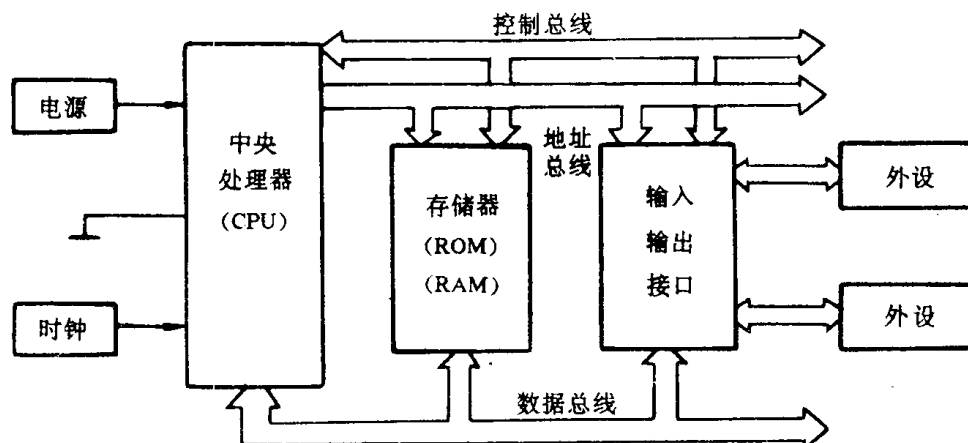


图 1-3 微型计算机组成结构

由图 1-3 可见,微型计算机由中央处理器、存储器和输入输出接口等三大部分组成。

中央处理器包括运算器和控制器。存储器包括只读存储器 (ROM) 和随机存储器 (RAM)。ROM 只能读出不能写入,因此,一般用来存放固定的程序;RAM 既可读出又可写入,是微型计算机的主要存储器。输入输出接口包括外部设备接口和通信接口,它们是微型计算机与外部交换信息的通道。

此外,CPU 和存储器所需的时钟发生器,以及连接各部件的总线也是必不可少的。时钟发生器一般是由晶体振荡器构成的,CPU 的操作时序、存储器的操作时序等都是由晶体振荡器发出的时序脉冲来确定的。

微型计算机的主要特点是:体积小、价格便宜和灵活性大。

体积小是很明显的,由于采用了大规模集成电路,所以尺寸大大缩小。微型计算机这种超小型的特征,使得它非常适合装在仪器仪表等装置内部作为一个数据处理部件,从而大大地扩大了计算机的应用范围。

国际上微型计算机的现行价格只有几十至几百美元,随着大规模集成电路的发展,其价格还会有所下降。

微型计算机的灵活性很大,它和一般计算机一样采用了程序存储、程序控制和自动优先中断等技术,能够灵活地组成各种系统。

由于微型计算机在使用时不要求严格的环境条件,而且具备上述特点,这就给微型计算机的普及应用提供了极有利的条件。现在,它正不断地渗透到各行各业,甚至人们的社会生活中,从而给人类的生产和生活带来了极其深远的影响。

§ 1.3 存储程序控制概念

把解题程序送入内存,由计算机自动地执行,使计算机在解题过程中避免人的干预,这就是存储程序控制的概念。

最初的电子计算机,存储器单纯地用来存放数据,计算过程要由人来控制。这样,完成一项计算工作所需要的时间,主要取决于人的操作速度。在大量的计算过程中,人们深感快速、广泛和重复计算的需要,因而产生了“存储程序”的思想。存储程序思想的实现是电子计算机的一项重大革新,它对以后计算机的发展起了巨大的推动作用。

随着存储程序控制概念的出现,程序设计的任务就被相应地提出来了。如何用指令来描述一个计算过程,这并不是一个简单的问题。由于计算机结构等各方面的限制,一台机器的指令是很有限的,一般有几十条,多则几百条。要用这些有限的指令来描述无限多的题目,是有很多困难的。因此,用计算机指令来描述解决问题的过程是一门技术,这就是所谓的程序设计技术。

最初的程序设计使用的是二进制代码形式的指令。这种指令能够直接被计算机接受并执行,因此,称这种形式的指令为“机器指令”。全部机器指令构成计算机的“机器语言”。用机器指令编制的程序,称为计算机的“机器语言程序”。机器语言程序可以直接在计算机上执行,但由于机器指令不便记忆,阅读和书写都很困难,因此,用机器语言编制程序很费时间。于是,人们改用符号表示二进制代码形式的指令,这就出现了“汇编语言”,可以说汇编语言是机器语言符号化的结果。由于符号形式的指令要比长串的二进制代码容易记忆,便于阅读和书写,所以,汇编语言在一定程度上克服了机器语言的缺点。但是,汇编语言编制的程序必须翻译成计算机能理解的机器语言程序才能在计算机上执行。为此,人们设计了一种叫做“汇编程序”的翻译程序,计算机配上汇编程序以后,就能自动进行程序的翻译工作,提高了编程序的效率。

此外,人们还创造了一些更有效的计算机程序设计语言——算法语言。算法语言同自然语言和数学语言比较接近,它容易为人们掌握,用它来描述一个解题过程或某一问题的处理过程将更加方便、灵活,并且具有一定的通用性。同样,用算法语言编写的程序也必须翻译成机器语言程序才能在计算机上执行。为此,人们设计了一种叫做“编译程序”的翻译程序,计算机配上编译程序之后,就能够自动地将用算法语言编制的程序翻译成等价的机器语言程序。

由上面的叙述可以知道,计算机只能执行由机器指令构成的程序,而用汇编语言或算法语言编写的程序必须翻译成机器语言程序才能在计算机上执行。因此,人们把用汇编语言或算法语言编写的程序称为“源程序”。源程序经过翻译后得到的可以在计算机上执行的机器语言程序称为“目的程序”。机器语言、汇编语言和算法语言统称为计算机语言,而程序设计就是用计算机语言描述某个问题的求解过程。

§ 1.4 计算机中数的表示

1.4.1 计算机中常用的数制

一、十进制数

在日常生活中人们最常用的是十进制数。十进制数的数值部分是用十个不同的数字符号 $0, 1, \dots, 9$ 来表示的, 我们把这些数字符号叫做数码。在数中, 一个数码所代表的意义与它所处的位置有关。例如, 780.43 这个数, 小数点左边的第一位代表个位, 表示它本身的数值 0; 左边第二位是十位, 表示 8×10^1 ; 左边第三位是百位, 表示 7×10^2 ; 而小数点右边的第一位“4”表示 4×10^{-1} ; 第二位“3”表示 3×10^{-2} 。因此, 这个数可以写成:

$$780.43 = 7 \times 10^2 + 8 \times 10^1 + 0 \times 10^0 + 4 \times 10^{-1} + 3 \times 10^{-2}$$

一般, 对任意一个正的十进制数 S , 都可以表示为:

$$S = K_{n-1}(10)^{n-1} + K_{n-2}(10)^{n-2} + \dots + K_0(10)^0 + K_{-1}(10)^{-1} + \dots + K_{-m}(10)^{-m} \quad (1.1)$$

或

$$S = \sum_{j=-m}^{n-1} K_j(10)^j \quad (1.2)$$

其中, K_j 可以是 $0, 1, \dots, 9$ 这十个数码中的任意一个, 它由 S 决定; m, n 为正整数; 括号内的 10 称为记数制的基数, 它表示在十进制数中可能用到的数码只有 10 个, 最大的是 9, 每位计满十就向高位进一, 即“逢十进一”。

一般地, 若 p 是大于 1 的整数, 则任一数 N 总可以用下式来表示:

$$N = K_{n-1}(p)^{n-1} + \dots + K_1(p)^1 + K_0(p)^0 + K_{-1}(p)^{-1} + \dots + K_{-m}(p)^{-m} \quad (1.3)$$

或

$$N = \sum_{j=-m}^{n-1} K_j(p)^j \quad (1.4)$$

其中, K_j 可以是 $0 \sim (p-1)$ 中的任意一个数码; m, n 为正整数; p 为基数。当 p 取不同数值时, N 为不同进制的数。

$p = 10$ 就是十进制的表示形式, N 称为十进制数。

$p = 8$ 就是八进制的表示形式, N 称为八进制数。

$p = 2$ 就是二进制的表示形式, N 称为二进制数。

二、二进制数

1. 二进制数的表示

例 1.1

$$\begin{aligned} (11011.101)_2 &= 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} \\ &= (27.625)_{10} \end{aligned}$$

在二进制数中, 基数为 2, 因此二进制数中可能出现的数码只有 0 和 1, 并且每一位上都

是“逢二进一，借一当二”。表 1-1 给出了二进制数与十进制数之间的对应关系。

表 1-1 二进制数与十进制数对照表

二进制数	十进制数	二进制数	十进制数
0	0	101	5
1	1	110	6
10	2	111	7
11	3	1000	8
100	4	1001	9

2. 二进制数的运算

二进制数只有 0 和 1 两个数码，其运算法则和运算过程比其他任何记数制的数都要简单。

(1) 加法

$$0+0=0; 1+0=1; 0+1=1; 1+1=10$$

例 1.2

$$10011.01 + 100011.11 = 110111.00$$

算式如下：

$$\begin{array}{r} 10011.01 \\ +) 100011.11 \\ \hline 110111.00 \end{array}$$

(2) 减法

$$0-0=0; 1-0=1; 1-1=0; 10-1=1$$

例 1.3

$$10110.01 - 1100.10 = 1001.11$$

算式如下：

$$\begin{array}{r} 10110.01 \\ -) 1100.10 \\ \hline 1001.11 \end{array}$$

(3) 乘法

$$0 \times 0 = 0; 0 \times 1 = 0; 1 \times 0 = 0; 1 \times 1 = 1$$

例 1.4

$$1101.01 \times 110.11 = 1011001.0111$$

算式如下：

$$\begin{array}{r} 1101.01 \\ \times) 110.11 \\ \hline 1101 \ 01 \\ 11010 \ 1 \\ 000000 \\ 110101 \\ 110101 \\ \hline 1011001.0111 \end{array}$$

(4)除法

$$0 \div 1 = 0; 1 \div 1 = 1$$

例 1.5

$$1101.1 \div 110 = 10.01$$

算式如下:

$$\begin{array}{r} 10.01 \\ 110 \overline{) 1101.1} \\ \underline{110} \\ 1 \\ \underline{1 } \\ 0 \end{array}$$

二进制数的四则运算,象通常的十进制数的四则运算一样,做加、减法时,小数点要对齐,做乘、除法时,先当成整数进行乘或除,算完之后再确定小数点的位置。二进制数的乘、除法运算,实际上就是二进制数的加、减法运算的重复。

电子计算机采用二进制数进行计算,一方面是因为二进制数运算简单,容易在机器上实现。另一方面是因为二进制数中仅有的 0、1 两个数码,可以很方便地用具有两种稳定状态的电或磁的元件来表示。例如:灯泡的亮与灭,继电器的吸合与释放,开关的接通与断开,电子管、晶体管的导通与截止等等。用这些元件的两种不同的稳定状态,可以控制电压的高低、电流的通断、脉冲的有无,以及磁化的顺向与逆向。这些物理状态均可分别采用二进制的 1 和 0 来表示。此外,采用二进制数,使计算机中的数据传送和存储也非常简单。

三、八进制数和十六进制数

由前面的叙述可以知道,在计算机中采用二进制数有节省元件、表示方便以及运算简单等优点。但是,二进制数也有一个很大的缺点,它书写冗长、不便阅读。因此,人们在书写程序和数据时,常常使用八进制和十六进制数。因为 8 和 16 都是二的整次幂,所以八进制和十六进制数同二进制之间的转换非常方便。

1. 八进制数的表示

八进制数就是式(1.3)中 p 等于 8 的情形。

例 1.6

$$\begin{aligned} (327)_8 &= 3 \times 8^2 + 2 \times 8^1 + 7 \times 8^0 \\ &= (215)_{10} \\ (328.135)_8 &= 3 \times 8^2 + 2 \times 8^1 + 8 \times 8^0 + 1 \times 8^{-1} + 3 \times 8^{-2} + 5 \times 8^{-3} \\ &= (216.18164)_{10} \end{aligned}$$

在八进制数中,基数为 8,因此八进制数中可能出现的数码为 0,1...,7,并且运算时数的每一位上都是“逢八进一,借一当八”。

2. 八进制数的运算

八进制数的运算与十进制数运算稍有不同,下面我们举例说明。

1) 加法

八进制数的加法运算规则可由表 1-2 表示。