

微计算机教程

WEIJISUANJIJIAOCHENG

叶念国 编

W

湖北科学技术出版社



WEIJI SUANJISUANJI JIAOCHENG WEIJI

微计算机教程

叶 念 国 编

湖北科学技术出版社

微计算机教程

叶念国 编

•

湖北科学技术出版社出版发行

湖北省新生印刷厂印刷

787×1092毫米 16开本 22.5印张 1插页 560,000字

1984年10月第1版 1984年10月第1次印刷

印数: 1—29,000

统一书号: 15304·46 定价: 3.55元

前 言

电子计算机是现代化科技与工业竞争的产物，它的出现又反过来大大促进了现代科技与工农业的发展。随着半导体技术和集成工艺的迅速发展，在七十年代初期微型计算机以崭新的姿态出现在计算技术的历史舞台上。微型计算机以其功耗小、功能强、成本低、可靠性好及体积小一系列优点、无可争辩地，强烈地冲击着科学研究、国防技术、工农业生产以及社会生活的一切领域，它已成为电子计算机社会化的一种重要手段。美国八十年代的经济衰退使得百业萧条，而以电子计算机特别是以微型计算机为主体的电子行业却异常兴旺。据报道，仅一九八二年微型计算机的交易额就达49亿美元，为一九八一年的一倍，预计一九八三年将增加到121亿美元。美国应用微型计算机每年完成的工作量相当于4000亿人年，即相当于两千倍美国现有人口一年的工作量。可以想象，在这样的社会里离开了微型计算机，整个社会将陷于瘫痪。无怪乎人们把美国现在的社会称之为“两C社会”，即竞争（COMPETITION）和计算机（COMPUTER）的社会。

自1971年第一个四位微处理器 INTEL4004问世以来，只经过了短短的十二年，微处理器就经历了四代的更新，其功能和集成度每年成倍的提高，而成本则成倍的下降。32位微处理器已于1981年推出，预计1985年将出现单片64位微处理器，那时的微型计算机系统将达到七十年代大型计算机的性能水平。微处理器以如此惊人的速度发展，它已广泛应用于数据采集、数据处理、过程控制、企业管理、智能仪表、办公自动化、交通车辆管理、汉字处理、诊断和医疗系统、仿真技术及分布式系统等领域。以微处理器组成的个人计算机，以其优异的性能价格比走向社会和家庭，微处理器的应用领域不断在开拓，以致我们难以确切预言它将会有哪些应用。人们把微处理器称之为第二次工业革命的推动者，这毫无夸张之意。第一次工业革命时，蒸汽机给人类以动力，第二次工业革命则是微处理器给人类以智能。

迅速在我国推广微型计算机的应用已是迫不及待的当务之急，所以在各个技术领域普及微型计算机的知识就显得非常必要。这本书是根据编者为武汉水利电力学院电力系统及其自动化专业的学生、研究生以及十几个为工程技术人员开办的微型计算机原理及应用短训班所使用的讲稿经过整理而编写的，考虑到主要是作为工科院校“电力系统继电保护及自动化”专业，“发电厂及电力系统专业”和“电力系统及其自动化”专业的教材，因此认为读者已具备数字逻辑及电路的知识。全书从应用的基点出发，着重介绍微型计算机的基本原理及结构、介绍汇编语言程序设计方法及接口技术。由于微型计算机发展极为迅速，国内外的有关文献资料也极为丰富，在有限的篇幅里无法反映众多的内容。因此本书将主要围绕微型计算机的应用这个中心展开，不打算也不可能罗列形形色色的微型计算机的资料，这或许是教材和参考书的主要差异。

全书的结构是前面着重介绍微型计算机有关硬件电路的基本原理，然后以汇编语言程序设计方法为主线，介绍软件的基础知识，最后围绕应用介绍接口技术，书末提供最必要的附录。

当前微型计算机的类型极多，但基本原理是相近的，故本书选择了目前国内外都用得较多的美国 Zilog 公司生产的 Z-80 八位机作为主线条来介绍微型计算机的基本原理。有了这些基础不难做到触类旁通。

本课程的特点是各部分具有较强的逻辑联系，并与科技应用紧密相关。学习时务必不要把注意力放在具体电路的分析上，而是要着眼于微型计算机各部份的逻辑关系及如何与应用结合起来。

本书编写的时间极为仓促，加之编者水平、学识的限制，缺点及错误必定不少，敬希读者不吝指正。

编 者

目 录

第一章 概 论	(1)
§ 1—1 微型计算机的发展概况	(1)
§ 1—2 电子数字计算机的基本工作原理	(2)
§ 1—3 必要基础知识的简要回顾	(4)
一、数制与编码	(4)
二、基本逻辑运算	(16)
三、触发器	(18)
四、三态门	(21)
五、基本逻辑部件	(22)
§ 1—4 本课程的学习方法	(25)
第二章 微处理器	(26)
§ 2—1 微型计算机的组成及工作概况	(26)
一、几个名词定义	(26)
二、微型计算机的主要组成部分	(26)
三、微型计算机的工作过程	(28)
§ 2—2 微处理器的结构	(31)
一、Z—80CPU 的内部结构	(31)
二、Z—80CPU 寄存器组	(33)
三、算术逻辑部件	(36)
四、控制部件	(37)
五、总线	(38)
§ 2—3 Z—80微处理器结构分析	(41)
§ 2—4 微处理器的发展概况	(45)
第三章 存 贮 器	(47)
§ 3—1 概述	(47)
§ 3—2 存储器与 CPU 的联接	(47)
一、半导体存储器的内部结构	(47)
二、存储器和 CPU 的联接	(49)
§ 3—3 半导体存储器的分类	(53)
一、随机存储器 (RAM)	(55)

二、只读存储器 (ROM)	(60)
§ 3—4 磁盘存储器	(65)
一、软磁盘的结构	(66)
二、软磁盘存储器与微型机系统的联接	(67)
三、软盘上的信息存储方式	(68)
四、软盘的记录格式	(69)
五、微型计算机软磁盘及软磁盘驱动器的联接	(71)
第四章 微处理器的指令系统	(74)
§ 4—1 指令的格式	(74)
§ 4—2 指令的寻址方式	(75)
一、直接寻址	(75)
二、间接寻址	(76)
三、寄存器直接寻址	(77)
四、寄存器间接寻址	(78)
五、变址寻址	(78)
六、相对寻址	(79)
七、立即寻址	(81)
八、零页寻址	(81)
九、位寻址	(82)
十、隐寻址	(83)
§ 4—3 微计算机面向程序员的结构	(83)
§ 4—4 指令的分类	(83)
一、数据传送和交换类	(84)
二、数据运算类	(84)
三、控制程序转移类	(85)
四、移位和按位操作类	(85)
五、CPU 的状态控制类	(86)
§ 4—5 Z—80的指令系统	(86)
一、Z—80CPU 的时序概念	(86)
二、Z—80的数据传送和交换类指令	(90)
三、Z—80的数据运算类指令	(108)
四、Z—80的控制程序转移类指令	(111)
五、Z—80的移位和按位操作类指令	(113)
六、Z—80的 CPU 状态控制类指令	(115)
第五章 汇编语言程序设计	(117)
§ 5—1 程序设计及计算机语言概述	(117)
一、计算机语言	(117)

1. 常用计算机语言的分类	(118)
2. 汇编语言与高级语言的比较	(119)
二、关于软件的几个问题	(119)
1. 什么是软件	(119)
2. 软件的基本分类	(120)
3. 编制程序的步骤	(120)
§ 5—2 汇编语言程序的格式	(121)
一、标号段	(121)
二、操作码段	(121)
三、操作数段	(121)
四、注释段	(123)
§ 5—3 伪指令	(123)
一、为什么要伪指令	(123)
二、伪指令语句的格式	(123)
1. 汇编起始地址说明伪指令	(123)
2. 符号定义说明伪指令	(124)
3. 数据说明伪指令	(125)
4. 条件汇编伪指令	(125)
5. 汇编结束伪指令	(126)
§ 5—4 宏指令	(127)
§ 5—5 汇编语言程序的设计方法	(128)
一、程序设计的步骤	(128)
二、程序的分类	(128)
1. 简单程序	(129)
2. 分支程序	(129)
3. 循环程序	(130)
4. 子程序	(134)
5. 查表程序	(136)
6. 汇编语言程序举例	(145)
三、程序设计中要注意的一些问题	(148)
1. 程序的分块结构	(148)
2. 存储单元的分配	(149)
3. 程序的正确性检查	(149)
§ 5—6 汇编程序	(150)
一、汇编程序的作用	(150)
二、源程序的手工汇编过程	(151)
三、汇编程序的流程图	(154)
1. 第一次扫描	(154)
2. 第二次扫描	(156)

第六章 微计算机与外设的连接	(158)
§ 6-1 概述	(158)
§ 6-2 I/O 端口的编址方式	(160)
一、存储器对应编址方式	(160)
二、I/O 端口专用编址方式	(160)
§ 6-3 CPU 与不同工作速度外设的配合方式	(161)
一、CPU 与极慢外设的配合	(161)
二、CPU 与中速外设的配合	(161)
三、CPU 与较快速外设的配合	(161)
§ 6-4 CPU 与外设交换数据的控制方式	(162)
一、Z80—CPU 的输入/输出时序	(162)
二、以查询方式实现数据的输入或输出	(163)
三、以中断方式实现数据的输入与输出	(166)
1. 中断的过程	(166)
2. 外设中断申请信号的产生和中断的屏蔽	(168)
3. 外设提出中断请求以后的过程	(169)
4. 中断服务程序的结构	(170)
5. 中断源的识别	(172)
6. 中断优先权的处理	(174)
7. Z80 的中断方式	(179)
四、以存储器直接存取方式 (DMA) 实现数据的输入与输出	(182)
1. DMA 过程是如何开始的	(182)
2. DMA 过程	(183)
第七章 输入/输出接口电路	(185)
§ 7-1 输入/输出接口电路的功能及构成	(185)
一、数据端口	(185)
二、状态端口	(185)
三、控制端口	(185)
四、地址逻辑	(185)
五、中断控制逻辑	(185)
六、读/写控制逻辑	(185)
§ 7-2 输入/输出接口电路的分类	(187)
一、按功能分类	(187)
二、按连接的物理量分类	(187)
三、按信息形式分类	(188)
四、按适用范围分类	(188)
§ 7-3 Z80—PIO 可编程并行接口电路	(188)

一、功能	(188)
二、PIO的结构	(189)
三、PIO各引脚的功能	(189)
四、PIO的初始化	(193)
五、PIO的时序	(198)
1. 写周期时序	(198)
2. 读周期时序	(199)
3. 输出方式时序	(199)
4. 输入方式时序	(199)
5. 双向方式时序	(200)
6. 位控方式时序	(201)
7. 中断控制时序	(201)
§ 7-4 串行接口电路	(203)
一、串行通讯的特点	(203)
二、串行数据位的识别	(204)
三、串行通讯的同步方式	(205)
1. 同步传送方式	(205)
2. 异步传送方式	(206)
四、奇偶校验	(207)
五、可编程串行通讯接口	(209)
1. 8251的功能	(209)
2. 8251的结构和引脚功能	(209)
3. 8251的初始化	(212)
4. 8251与CPU和外设的连接	(213)
§ 7-5 计数器/定时器电路	(216)
一、CTC功能概述	(216)
二、CTC的结构	(216)
1. CTC的总体结构	(216)
2. 通道的结构	(218)
3. CTC的引脚功能	(219)
4. CTC的初始化	(220)
5. CTC的时序	(223)
§ 7-6 微计算机与模拟量的接口电路	(225)
一、D/A转换电路及其与CPU的连接	(225)
1. D/A转换电路的基本原理	(225)
2. D/A转换电路的实例	(227)
二、A/D转换电路及其与CPU的连接	(230)
1. A/D转换电路原理概述	(230)
2. A/D转换电路的实例	(233)

3. ADC—0809的应用举例	(236)
§ 7—7 总线标准	(238)
一、Sloo 总线	(239)
二、EIA—RS232C 串行总线	(239)
第八章 微计算机的接口技术	(242)
§ 8—1 概述	(242)
§ 8—2 微计算机与开关或接点的接口	(242)
一、硬件的配置	(242)
二、程序设计	(245)
1. 监视开关接通的程序	(245)
2. 监视开关变位的程序	(247)
§ 8—3 微计算机与 LED 显示器的接口	(249)
§ 8—4 微计算机与键盘的接口	(255)
一、编码键盘	(255)
二、非编码键盘	(255)
§ 8—5 微计算机与盒式磁带机的接口	(259)
一、硬件配置	(260)
一、程序设计	(260)
§ 8—6 微型计算机与模拟电路的接口	(267)
一、用软件按逐级逼近法实现 A/D 转换	(267)
二、用 D/A 转换器产生锯齿波电压	(269)
第九章 微计算机应用举例	(271)
§ 9—1 微计算机数据采集系统的结构	(271)
§ 9—2 数据采集系统主要硬件电路的原理	(272)
一、多路模拟开关	(272)
二、可编程增益隔离放大器	(276)
三、采样—保持电路	(277)
四、数控滤波器	(278)
§ 9—3 采集数据的处理	(280)
§ 9—4 数据采集系统的主要功能	(281)
一、模拟量采集	(281)
二、对采集数据进行中位值滤波	(281)
三、A/D 转换电路的误差校正	(281)
四、越限检查	(281)
五、标度变换	(281)
六、事件顺序记录	(283)
七、实时时钟	(284)
八、制表打印和参数显示	(284)

九、趋势分析.....	(285)
十、通过键盘实现人机联系.....	(285)
§ 9—5 硬件配置.....	(285)
一、地址逻辑电路.....	(285)
二、定时电路.....	(285)
三、开关量输入电路.....	(286)
四、模拟量输入电路.....	(286)
五、打印机接口电路.....	(286)
六、智能终端接口电路.....	(286)
§ 9—6 软件设计.....	(288)
一、基本程序的组成.....	(288)
1. 初始化程序.....	(288)
2. 模拟量采样子程序.....	(288)
3. 中位值滤波子程序.....	(288)
4. 数据合理性检查子程序.....	(288)
5. 自校程序.....	(288)
6. A/D 零漂修正子程序.....	(288)
7. 数据传送子程序.....	(288)
8. 越限检查子程序.....	(288)
9. 越限、复限处理子程序.....	(289)
10. 打印子程序.....	(289)
11. 实时时钟中断服务子程序.....	(289)
12. 开关量采样中断服务子程序.....	(289)
13. 键盘操作中断服务子程序.....	(289)
二、内存分配.....	(289)
1. 只读存储器的分配.....	(289)
2. 随机存储器的分配.....	(289)
三、主程序流程.....	(289)
四、子程序举例.....	(289)
1. 模拟量采样子程序.....	(289)
2. 中位值滤波子程序.....	(293)
3. 数据传送子程序.....	(294)
4. 越限检查子程序.....	(295)
§ 9—7 关于抗干扰的若干技术问题.....	(297)
一、干扰的途径.....	(297)
二、抗干扰措施.....	(297)
1. 来自系统内部的干扰抑制方法.....	(297)
2. 来自外部的干扰抑制方法.....	(299)
附 录.....	(301)

第一章 概 论

§ 1—1 微型计算机的发展概况

五十年代后期出现了集成电路,它较之电子管和分立元件电路具有体积小、重量轻、功耗低、可靠性高、速度快及成本低等一系列优点。因而集成电路在计算技术领域获得了越来越广泛的应用。微型计算机就是在大规模和超大规模集成电路的基础上产生和不断发展的。自1971年美国INTEL公司推出第一个微处理器Intel 4004以后,在短短的十二年里,微处理器以惊人的速度向前发展。字长由4位、8位、16位发展到32位。Zilog公司的16位微处理器Z8000芯片面积 40mm^2 ,集成17500个元件,Intel公司的16位微处理器8086,芯片面积 325mm^2 ,集成了29000个元件。Motorola公司的16位微处理器MC68000,是内部32位外部16位数据总线的准32位机,芯片面积 42.5mm^2 ,集成了68000个元件。而Intel公司的32位微处理机iAPX-432的集成度已达到每片250000个元件。预计到1990年一个芯片上的集成度将由现在的几十万个元件发展到上千万个元件。电路的速度也在不断提高,一种称为MAC-32的32位微处理器,执行一个32位的加法运算只需60ns(毫微秒即 10^{-9} 秒),它是以32兆赫的时钟频率工作的。由于集成电路的工艺不断提高,MOS逻辑电路的开关速度达到0.5ns,ECL逻辑电路的开关速度达到0.1ns。速度更快的砷化镓和约瑟夫逊结已研制成功,它比硅器件的速度快50倍,耗电量约小1000倍,日本NEC公司研制成功的约瑟夫逊结的开关速度已达10.8ps(微微秒)。当然,目前对于约瑟夫逊结的实用性,还存在着一定的争议。

推动微型计算机发展的重要因素有三个,一是集成电路技术,二是软件技术,另一个则是社会对微型计算机的广泛需求,即应用技术。自1946年第一台电子计算机诞生,至今经历了四代,即电子管计算机、晶体管计算机、中小规模集成电路计算机和大规模集成电路计算机,目前正在向第五代的“智能计算机”进军。自1971年第一台微处理器问世以来,至今也经历了四代。第一代是从1971年至1974年出现的用PMOS工艺制造的4位和8位微处理器,如Intel公司的4004、8008即是。第二代是1975年至1977年出现的用NMOS工艺制造的8位微处理器,Intel公司的8080、Motorola公司的M6800等即是。第三代是1978年至1981年出现的以NMOS工艺、CMOS工艺、HMOS工艺及 $1.2\mu\text{L}$ 双极工艺等生产的16位微处理器,如Intel公司的8086、Zilog公司的Z8000、Motorola公司的MC-68000、National Semiconductor公司的NS16000等即是。第四代是1982年以后出现的32位微处理器,如Intel公司的iAPX-432、National Semiconductor公司的NS16032即是。

微型计算机正朝着下述方向发展:

1. 增加字长。因为字长对指令的功能、处理数据的精度、速度以至于寻址范围都有极大的影响。
2. 提高处理速度。处理速度对于微型计算机的应用有举足轻重的影响,大量的数据处理及实时性要求很高的过程控制,对于处理速度有时会提出相当苛刻的要求。

3. 提高集成度。微型计算机的基础是大规模及超大规模集成电路, 集成度指标又直接影响着微型计算机的功能、体积、速度、功耗和可靠性。现仅从集成度对可靠性的关系来看, MC68000微处理器中集成了68000个元件, 以每个元件三条引线计算, 则当用分立元件构成该处理器时, 至少有 $3 \times 68000 = 204000$ 个焊接点及相应的联线。而实际的 MC68000 微处理器芯片对外部的引脚只有64条。可以想到, 这两者的故障概率将相差多大。

4. 扩大存贮量。存贮容量关系到计算机的解题规模和能力, 特别对那些求解复杂数学模型的任务尤为突出。

5. 增强中断和数据交换能力。这是提高微型计算机灵活性、实时性和快速性的重要途径。

6. 加强软件支持能力。软件是微型计算机赖以发挥作用的重要保证, 有完善的硬件而无强有力的软件支持, 微型计算机是难以发挥其功能的。

7. 提高性能价格比。这是微型计算机能否发展的关键指标, 高性能但却昂贵的机器是难以普及的, 只有价廉物美, 才具有强大的生命力。当今微型计算机正以性能每两年提高一倍, 而价格却降低一倍的速度发展。

§1—2 电子数字计算机的基本工作原理

常用的计算工具大致可分为两类, 一类是模拟计算机, 一类是数字计算机。计算尺就是一个模拟计算机, 它是通过尺身与滑尺的长度来完成运算。由于采用按对数的分度方法, 因此, 以尺身与滑尺的两个长度的相加或相减, 就实现了乘、除运算。由于长度是一个模拟量 (AnaLog Value), 即连续变化的量。因此, 计算尺是以模拟量为基础的模拟计算机。这类计算机的精度与尺身和滑尺的长度及分度粗细有关。算盘则属数字计算机, 它是用一定位数的数字值来表示一个量的大小。它的精度与位数有关。显然, 提高数字计算机的精度比模拟计算机要容易得多。数字计算机的抗干扰能力也比模拟计算机强, 计算尺的读数与人们的经验、视角等有关。然而, 人们不会因为算盘的算珠没有完全拨到顶端而不把它读成1, 却读成0.8。电子计算机也分为电子模拟计算机和电子数字计算机两大类。人们常常说的电子计算机, 实际上指的是电子数字计算机。

电子计算机在组成结构上完全是按照人们进行运算的方式形成的。为了完成一道数学题的演算, 需要将数学式子、原始数据、运算步骤写在纸上, 运算的中间结果和最终结果也要写在纸上。当然还需要一个计算工具, 例如算盘。再就是控制运算过程的人。这三大大部分在电子计算机里就是存贮器、运算器及控制器, 如图1—1所示。为了能够把原始数据, 运算步骤 (程序) 等送入计算机, 以及将计算结果送出来告知人们, 还需要配置输入设备和输出设备。

数据和指令 (完成某种操作的命令) 都是以一定位数的二进制码来表示, 通过输入设备 (例如纸带读入机、卡片输入机等) 将原始数据和由一条条指令所组成的程序送到具有相当数量存贮单元的存贮器中存放起来, 每个存贮单元都有一个确定的二进制码形式的地址。然后在控制器的自动控制下, 通过运算器把存贮器中的数据按程序所规定的操作和步骤进行运算, 最后将运算结果通过输出设备 (例如打印机、显示器等) 以人们习惯的形式输出。为了保证各个部分的协调工作, 控制器在执行程序的过程中, 还要监视各个部件的工作状况, 这

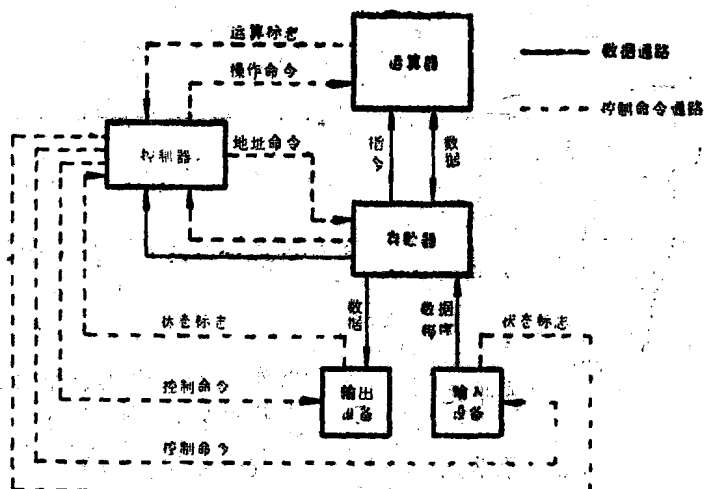


图 1—1 数字计算机框图

就是图 1—1 中运算标志和状态标志的作用。例如控制器按程序要求，需将在存储器中某地址单元的数据输出到打印机去打印，这个操作必须在该打印机的状态标志表明打印机已准备就绪的情况下才能执行。

现以计算 $ax^2 + bx + c$ 为例，式中 a 、 b 、 c 及 x 均为已知数，并分别通过输入设备将它们存放在存储器。由于一般运算器只能做简单的四则运算，所以我们按以下步骤安排计算该式的程序：

- 第一步：进行 a 与 x 的乘法运算，得到 ax ；
- 第二步：将 ax 与 b 相加，得到 $ax + b$ ；
- 第三步：将 $ax + b$ 与 x 相乘，得到 $ax^2 + bx$ ；
- 第四步：将 $ax^2 + bx$ 与 c 相加，得到 $ax^2 + bx + c$ ；
- 第五步：将运算结果输出到打印机打印；
- 第六步：停机。

我们看到，每一步都完成对某两个或一个数的某种操作，例如第三步是将 $(ax + b)$ 与 x 这两个数作乘法操作，第五步则是将 $(ax^2 + bx + c)$ 这一个数作输出打印操作。计算机是以不同的指令实现各种操作的，无疑，指令应该提供的信息应包含两个内容，即做什么操作和被操作的数在哪里。因而当用二进制代码表示指令时（计算机是由数字电路构成的，它只识别以高、低电平形式出现的二进制代码），指令就被划分为操作码和地址码两部分，如图 1—2 所示。地址码可以是存储器某单元的地址，也可以是运算器中的某寄存器的编号。

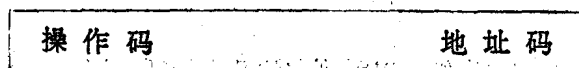


图 1—2 指令的格式

表 1—1 中列出了完成前述运算的存储器内容分配情况。在第 100 号到 106 号地址的存储单元里存放着由各种指令组成的程序，在第 350 号到 353 号地址的存储单元里分别存放着数据

a、b、c、x。

表 1—1

存贮单元地址	存贮单元内容		操 作 说 明
	操作码	地址码	
100	取数	350号单元	取350号存贮单元中的数a送到运算器的寄存器A中
101	乘	353号单元	将353号存贮单元中的数x与寄存器A中的数a相乘得 ax ，并保留在A中
102	加	351号单元	将351号存贮单元中的数b与A中的数相加得 $ax+b$ ，并保留在A中。
103	乘	353号单元	将353号存贮单元中的数x与A中的数 $ax+b$ 相乘得 $(ax+b)x=ax^2+bx$ 并保留在A中。
104	加	352号单元	将352号存贮单元中的数c与A中的数 ax^2+bx 相加得 ax^2+bx+c ，并保留在A中。
105	打印	2号打印机	将A中的内容即计算结果输出到2号打印机打印
106	停机		控制器停止向下一存贮单元取指令
350	a		
351	b		
352	c		
353	x		

从这一例子可以看出，计算机不同于常规布线逻辑的电子仪器，它不是靠改变接线（即改变电路元件的联接）实现功能的改变，而是靠存放在存贮器中的程序来完成各种不同的任务，因而它较之布线逻辑有极大的灵活性。计算机能够自动地进行工作，显然是因为能在存贮器中存放指挥计算机工作的程序。

§ 1—3 必要基础知识的简要回顾

对数字电路的有关内容作一简要的回顾，有助于对后续微型计算机原理的学习。由于篇幅限制，下面仅作结论性的回顾，不再作推导和证明。

一、数制与编码

1. 进位计数制

从人们最常用的十进制数我们可以看到它的几个特征，即，

(1) 有十个基本的有序数字符号（或称元素）0、1、2、3、4、5、6、7、8、9和一个小数点符号。

例如786.42

(2) 每一个数字符号所代表的值或权（Weight）取决于它在数中的位置。

例如786.42中的4代表0.4, 8代表80等等。

(3) 当两数相应位的和大于最大单个数字符号的值9时, 可按逢十进一的原则表示这个数。

例如 $79 + 38 = 115$

(4) 数中的各数字符号每向左移一位, 则数的值增长10倍。而向右移一位, 则缩小10倍。

例如976.42向左移一位变为9764.2, 而向右移一位则变为97.642。

(5) 一个数常用表示法有位置记数法和多项式记数法。

例如976.42即为位置记数法。它以各个数字符号在数中的位置不同来表示数的大小。它的多项式表示形式则为:

$$9 \times 10^2 + 7 \times 10^1 + 6 \times 10^0 + 4 \times 10^{-1} + 2 \times 10^{-2}$$

式中的“10”称为十进制计数制的基数。

任何其他进制的数都有与十进制数类似的特征, 这些特征可归纳为:

(1) K进制数具有K个基本的有序数字符号和一个小数点符号。K称为该进制数的基数。

例如二进制数有两个数字符号0与1。八进制数有八个数字符号0、1、2、3、4、5、6、7。十六进制有十六个数字符号0、1、2、3、4、5、6、7、8、9、A、B、C、D、E、F等等。

(2) K进制计数制中最大数字符号的值为基数K减1。例如十进制为9, 二进制为1, 八进制为7, 十六进制为F(相当十进制数的15)。

(3) 两个K进制数的加法运算按“逢K进一”的原则, 减法运算则按“借一当K”的原则。例如二进制数 $10 + 11 = 101$, 即逢二进一。八进制数 $54 + 76 = 152$, 即逢八进一。十六进制数 $5A + FB = 155$, 即逢十六进一。又如二进制数 $10 - 01 = 01$, 即借一当二。八进制数 $62 - 37 = 23$, 即借一当八。十六进制 $C4 - 6B = 59$, 即借一当十六。

(4) K进制数每向左移一位, 其值增大K倍, 向右移一位, 则减少K倍。例如二进制数011的值为三, 向左移一位变成110, 其值就变为六即增大一倍, 向右移一位变成01.1, 其值为十进制数1.5, 即缩小一倍。八进制数075的值相当于十进制数61, 向左移一位变成750, 其值则相当于十进制数488, 即增大八倍。向右移一位变成07.5, 其值相当于十进制数7.625, 即缩小八倍。十六进制数0D2的值相当于十进制数210, 向左移一位变成D20, 其值相当于十进制数3360, 即增大16倍。向右移一位变为D.2, 其值相当于十进制数13.125, 即减少16倍。

(5) 一个K进制数N的位置记数法表示形式为:

$$N = N_{n-1}N_{n-2}N_{n-3}\cdots N_1N_0.N_{-1}N_{-2}\cdots N_{-m}$$

它的多项式表示形式为:

$$N = N_{n-1}K^{n-1} + N_{n-2}K^{n-2} + N_{n-3}K^{n-3} + \cdots + N_1K^1 + N_0K^0 + N_{-1}K^{-1} + \cdots + N_{-m}K^{-m}$$

$$\text{或写成: } N = \sum_{i=-m}^{n-1} N_i K^i$$

式中n代表整数部分的位数, m代表小数部分的位数, N_i 是K进制数K个数字符号中的