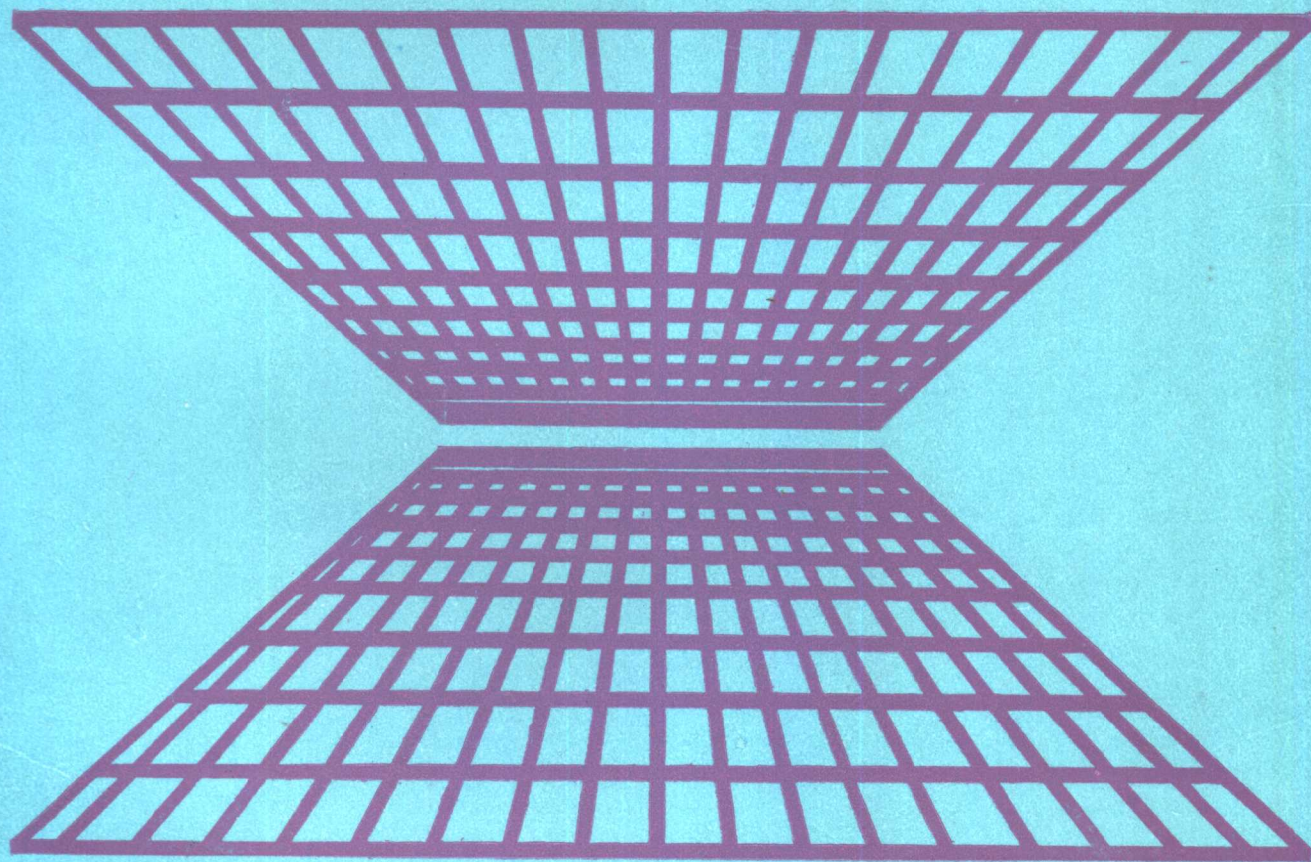


MCS 8098

# 单片微机技术基础

张振远等 编著



气象出版社

*MCS 8098*

# 单片微机技术基础

张振远 张玉华 汪大伟 编著  
卫金茂 刘云翔

气象出版社

(京)新登字 046 号

### 内 容 简 介

本书从零起点出发,详细地论述了 MCS8098 单片机的软、硬件工作原理及其应用方法,同时对 MCS8098 单片机的应用系统和开发系统做了全面的介绍。并且给出了大量的编程实例。全书内容系统完整,概念清楚,叙述新颖,理论联系实际,实用性强。

本书可作为大专院校相关专业开设单片机原理与应用课程的教材,同时对于从事单片机开发与应用的工程技术人员,也是一本难得的参考书。

### 图书在版编目(CIP)数据

mcs8098 单片微机技术基础/张振远等编著. —北京:气象出版社,1995.3

ISBN 7-5029-1900-7

I. m… II. 张… III. 单片微型计算机-概论 IV. TP368

.1

### MCS 8098 单片微机技术基础

张振远 张玉华 汪大伟  
卫金茂 刘云翔 编著

责任编辑:陶国庆 终审:周诗健

封面设计:席大光 责任技编:席大光 责任校对:陶国庆

\* \* \*

气象出版社出版

(北京海淀区白石桥路46号 邮政编码100081)

北京昌平环球印刷厂印刷

新华书店总店科技发行所发行 全国各地新华书店经销

\* \* \*

开本:787×1092 1/16 印张:14.375 字数:360千字

1995年4月第一版 1995年4月第一次印刷

印数:1—3500

ISBN 7-5029-1900-7/TP·0050

定价:14.60元

# 前 言

计算机,尤其是微型计算机对于当代中国人已经不再感到陌生。作为微型计算机家族中的重要成员和后起之秀——单片机,由于同样具有计算机的三大基本功能(运算功能、记忆功能和控制功能),尤其是更能体现出其控制功能,再加上它还具有软、硬件透明度高,价格低廉和容易开发普及等优点,所以应用领域非常广阔。此外,把单片机作为学习微机原理与应用的基础,也是非常合适的。因此,近年来在国内一个学习和推广应用单片机的热潮正日益兴起,方兴未艾。

作为大专院校的电子学及其相关的专业,义不容辞地担负着培养本专业现代化人才的使命。然而,由于历史的原因,在我国一提到学习微机原理与应用就会想到 Z80 和 51 系列的 8 位机。不可否认,Z80 及其单板机,对于在我国普及和推广微型计算机知识,尤其在教学领域中,曾经做出过不可磨灭的贡献。可是 Z80 在国际上早已被淘汰,在我国也已过时,被新型单片机所取代。至于 51 系列单片机,由于受到我国近年来国情的局限,在学校里并没有得到充分的普及。当今已经进入 90 年代中期,要想尽快缩短与世界先进水平的差距,“微机原理与应用”课程内容的改革势在必行,改革的出路在于积极引进当今世界上广泛流行的、具有先进性和代表性的 MCS8098 单片机,淘汰已经过时落后的机型和教学内容,让学生以最短的时间学到最先进的微机知识和技能,不必再去走那种先学 Z80,再学 51,最后才学 8098 单片机的弯路。

近年来国内虽然已经出版了许多有关 8098 单片机的书籍,但可惜的是,这些书基本上都是以前学 Z80 和 51 系列机的人员为对象而编写的,而且大多都不适合用作教材。根据这种情况,我们以初学者为对象编写了本书,作为微机原理与应用课的换代教材。

值得提出的是,我校早在 1989 年就引进了 MCS8098 单片机,经过几年来坚持不懈努力,不仅在教学上从研究生到本科生多次开设过相关的课程,取得了比较丰富的教学经验,而且还独立地研制成功了 DS98C 型单片机教学实验开发机,现已推向市场,受到广大用户好评,这些都为本书的编写奠定了可靠的基础。此外,作者多年来的科研成果也为本教材提供了丰富的实用内容。

本书面向从零起点开始直接学习和掌握 8098 单片机原理和应用的人员,具有系统性、完整性和实用性等特点,可用作大专院校的专科生、本科生、乃至研究生的教材。对于从事单片机开发的工程技术人员,也是一本不可多得的实用参考书。读者如对本书及 8098 单片机希望有更进一步的了解,可与作者联系。作者联系电话:(0431)5685085 转 2102。

在本书的编写过程中,还得到了周围许多同志的关怀和帮助,在此一并表示感谢。

由于作者水平有限,编写时间仓促,不足之处在所难免,欢迎读者指正。希望本书能对您的工作和学习有所帮助。

作 者

1994 年秋于长春

东北师范大学

# 目 录

前言

绪论..... (1)

## 基础篇

第一章 计算机常用的数制与码制..... (6)

§ 1.1 计算机中数的表示法 ..... (6)

§ 1.2 二进制数的四则运算..... (11)

§ 1.3 计算机的常用码制..... (14)

习题与思考题 ..... (15)

第二章 微型计算机基础知识 ..... (16)

§ 2.1 微型计算机的基本结构..... (16)

§ 2.2 存储器基础知识..... (17)

§ 2.3 微型计算机的 I/O 接口 ..... (20)

§ 2.4 I/O 数据传送方式 ..... (22)

习题与思考题 ..... (23)

## 原理篇

第三章 8098 单片机的硬件结构 ..... (24)

§ 3.1 8098 单片机的内部结构与引脚 ..... (24)

§ 3.2 中央处理器 CPU 结构 ..... (26)

§ 3.3 存储器空间的分布..... (29)

§ 3.4 存储器控制..... (31)

§ 3.5 芯片配置寄存器(CCR) ..... (33)

§ 3.6 I/O 口与接口功能部件..... (35)

§ 3.7 复位与掉电保护..... (38)

习题与思考题 ..... (41)

第四章 8098 单片机的指令系统 ..... (42)

§ 4.1 预备知识..... (42)

§ 4.2 指令系统分类详解..... (49)

习题与思考题 ..... (69)

第五章 汇编语言程序设计基础 ..... (70)

§ 5.1 汇编语言源程序的语句格式..... (70)

§ 5.2 汇编语言程序的基本结构..... (73)

§ 5.3 子程序..... (75)

§ 5.4 查表程序..... (76)

§ 5.5 数字与码制变换程序..... (80)

§ 5.6 线性搜索与排序..... (84)

习题与思考题 .....	(87)
--------------	------

## 接口篇

<b>第六章 8098 单片机的中断系统</b> .....	(88)
§ 6.1 中断源与中断向量 .....	(88)
§ 6.2 中断系统 .....	(90)
§ 6.3 中断系统编程举例 .....	(94)
习题与思考题 .....	(96)
<b>第七章 8098 单片机接口功能部件</b> .....	(97)
§ 7.1 定时器部件 .....	(97)
§ 7.2 高速输入部件(HSI) .....	(100)
§ 7.3 高速输出部件(HSO) .....	(107)
§ 7.4 A/D 变换器部件 .....	(117)
§ 7.5 脉宽调制输出部件 .....	(121)
§ 7.6 串行口部件 .....	(124)
习题与思考题 .....	(134)

## 系统篇

<b>第八章 8098 单片机的存储器扩展</b> .....	(136)
§ 8.1 常用存储器芯片 .....	(136)
§ 8.2 最小系统 .....	(143)
<b>第九章 8098 单片机的应用系统</b> .....	(146)
§ 9.1 常用 I/O 接口扩展芯片及其应用 .....	(146)
§ 9.2 显示器接口电路 .....	(157)
§ 9.3 键盘接口电路 .....	(166)
§ 9.4 可编程键盘和显示器接口芯片 8279 及其应用 .....	(169)
§ 9.5 微型打印机及其接口电路 .....	(177)
§ 9.6 8098 单片机与 CRT 的接口应用 .....	(184)
<b>第十章 8098 单片机开发系统</b> .....	(191)
§ 10.1 概述 .....	(191)
§ 10.2 DS98C 单片机开发系统的组成和特点 .....	(193)
§ 10.3 DS98C 单片机开发系统的硬件描述 .....	(195)
§ 10.4 DS98C 单片机开发系统的监控程序 .....	(199)
§ 10.5 DS98C 单片机开发系统的使用与操作 .....	(213)
<b>附录一 MCS 8098 单片机指令操作码总表</b> .....	(218)
<b>附录二 MCS 8098 特殊功能寄存器速查表</b> .....	(220)
<b>附录三 十进制与十六进制数转换表</b> .....	(222)
<b>附录四 ASCII 码表</b> .....	(223)

# 绪 论

## 一、单片机的由来和分类

### 1. 计算机

数字式电子计算机(以下简称计算机)自从 1946 诞生以来,至今不到半个世纪,但发展极为迅速。由于它具有运算、记忆和控制三大基本功能,在某种程度上可以代替和减轻人们的脑力劳动,故又称为电脑,并在科学技术和国民经济各个领域中都得到了广泛的应用。在当今世界上,一个国家的计算机科技、生产水平和应用、普及的程度,已经成为衡量这个国家现代化程度的重要标志。

计算机之所以有如此飞速的发展和进步,是和电子元器件的技术进步分不开的,尤其是和半导体以及集成电路技术的不断完善密切相关。随着组成计算机的电子元器件的发展,计算机至今大致已经经历了如下四个发展时代。即:

第一代(1946~1958):电子管时代;

第二代(1958~1964):晶体管时代;

第三代(1964~1971):中(MSI)、小(SSI)规模集成电路时代;

第四代(1971~至今):大规模(LSI)和超大规模(VLSI)集成电路时代。

目前计算机正在朝着人工智能计算机和神经网络计算机的更高层次发展。

不管使用什么样的电子元器件组成计算机,它都应包括输入、内部存贮器、运算器、控制器和输出等五大基本组成部分。如图 1 所示。

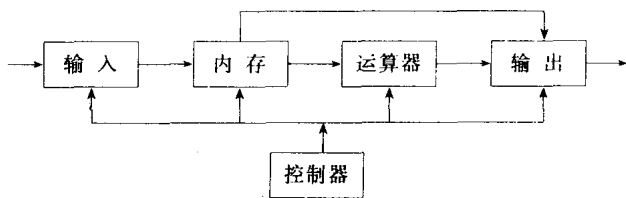


图1 数字式电子计算机基本组成

### 2. 微型计算机

在 60 年代中期,平面硅工艺的出现和薄膜技术的突破,产生了半导体集成电路。随着半导体工业突飞猛进的发展,又出现了大规模集成电路(LSI),使把运算器和控制器合在一起做到一个芯片上成为可能,于是就出现了微处理器(microprocessor,简称  $\mu\text{P}$ ),又称为 CPU(central processing unit)。微处理器虽然不能算做计算机,但它却包含了组成计算机的核心部分控制器和运算器。在微处理器的基础上,再适当配以存储器(ROM, RAM)和 I/O 接口芯片,便可组成计算机。从而诞生了微型计算机(microcomputer, 简称为  $\mu\text{C}$ )。在此基础上,再配备上适当的外部设备和电源等便可组成一个完整的微型计算机系统(microcomputer system)。

微型计算机的大致发展如下:

1971 年: Intel 公司 4004(四位机), PMOS 工艺, 组成 MCS-4 微机;

1972 年: Intel 公司 8008(八位机), PMOS 工艺, 组成 MCS-8 微机;

1973 年: Intel 公司 8080(八位机), NMOS 工艺;

Motorola 公司 6800(八位机), NMOS 工艺;

- 1974年: Intel公司 8085(八位机);
- Zilog公司 Z80(八位机), NMOS工艺;
- 1978年: Intel公司 8086(16位机), HMOS工艺, 组成MCS-4微机;
- 1980年: Intel公司 8088(准16位机), 组成IBM PC/XT微机;
- 1984年: Intel公司 80286(16位机), 组成286微机;
- 1985年: Intel公司 80386(准32位机), 组成386微机;
- 1990年: Intel公司 80486(32位机), 组成486微机;
- 1993年: Intel公司 80586(准64位机), 组成“奔腾”型微机;

### 3. 单片机

随着超大规模集成电路(VLSI)的不断进步和半导体工艺的进一步完善, 在微处理器的基础上, 实现了将部分存储器和I/O接口电路等一系列组成计算机的必要部分也一起集成到一个芯片中, 于是便出现了单片微型计算机(简称单片机), 又称作微控制器(single chip micro-controller)。由于单片机具有技术先进, 体积小, 重量轻, 功耗低, 使用灵活, 可靠性高, 价格低和便于开发等许多优点, 其应用领域更加广阔, 近年来发展非常迅速, 品种也日新月异。Intel公司是具有代表性的单片机生产公司, 以该公司的产品为例, 单片机的发展年代大致如下:

- 1987年: MCS48系列(8048), 低档8位机(HMOS), 主要应用于家用电脑;
- 1980年: MCS51系列(8051), 高档8位机(HMOS), 主要应用于工业控制;
- 1984年: MCS96系列(8096), 16位机(HMOS), 主要应用于工业控制, 智能仪器仪表和高速电机的控制;
- 1988年: MCS98系列(8098)准16位机(HMOS), 是8096系列的普及型;
- 1990年: MCS196/198系列(80C196/198), 16位机(CMOS), 是8096系列的低功耗改进型。

世界上生产单片机的厂家众多, 各厂家的产品系列和机型也很多, 归纳起来按用途, 位数等进行分类, 大致可以分成如下形式:

通用型	1位机——		
	4位机——	TMS-1000系列	(TI, 1975)
		MCS-48系列	(Intel, 1978)
		MC 6801	(Motorola)
	8位机低档机	{ Z8	(Zilog)
		{ 6500/1	(Rockwell)
	8位机高档机	{ TMS-7000系列	(TI)
		{ MCS-51系列	(Intel)
		{ 80C51系列	(Philips)
	16位机——	TMS-990系列	(TI)
		MCS-96系列	(Intel)
		MPD-78300系列	(NEC)
	32位机	MCS80960	(Intel)
专用型	模拟信号处理器	Intel 2920	
	数字信号处理器	NEC MPD7720	
		TMS 320系列	(TI)
	图形处理器	TMS 340系列	(TI)



## 二、MCS-96 系列单片机的家族及其应用范围

MCS-96 系列单片机是 Intel 公司推出的具有典型意义的第三代 16 位单片机系列。在该系列中近年来又很快出现了许多机型,可以将其再划分为  $8 \times 9 \times$  和  $8 \times C19 \times$  两个子系列。MCS-96 系列单片机的家族如表 1 所示。

表 1 MCS-96 单片机系列家族一览表

型号	引脚	工艺	版本	CPU	ROM	EPRM	XRAM	A/D	备注
8096	68	HMOS	90	✓					
8096BH	68	HMOS	BH	✓					
8097	68	HMOS	90	✓				8 路	42 $\mu$ s
8097BH	68	HMOS	BH	✓				8 路	带保持, 22 $\mu$ s
8097JF	68	HMOS	JF	✓			✓	8 路	带保持, 22 $\mu$ s
8094	48	HMOS	90	✓					
8094BH	48	HMOS	BH	✓					
8095	48	HMOS	90	✓				4 路	42 $\mu$ s
8095BH	48	HMOS	BH	✓				4 路	带保持, 22 $\mu$ s
8396	68	HMOS	90	✓	✓				
8396BH	68	HMOS	BH	✓	✓				
8397	68	HMOS	90	✓	✓			8 路	42 $\mu$ s
8397BH	68	HMOS	BH	✓	✓			8 路	带保持, 22 $\mu$ s
8397JF	68	HMOS	JF	✓	✓		✓	8 路	带保持, 22 $\mu$ s
8394	48	HMOS	90	✓	✓				
8394BH	48	HMOS	BH	✓	✓				
8395	48	HMOS	90	✓	✓			4 路	42 $\mu$ s
8395BH	48	HMOS	BH	✓	✓			4 路	带保持, 22 $\mu$ s
8796BH	68	HMOS	BH	✓		✓			
8797BH	68	HMOS	BH	✓		✓		8 路	带保持
8797JF	68	HMOS	JF	✓		✓	✓	8 路	带保持
8794BH	48	HMOS	BH	✓		✓			
8795BH	48	HMOS	BH	✓		✓		4 路	带保持, 22 $\mu$ s
8098	48	HMOS	BH	✓				4 路	准 16 位, 带保持
8398	48	HMOS	BH	✓	✓			4 路	准 16 位, 带保持
8798	48	HMOS	BH	✓		✓		4 路	准 16 位, 带保持
8 $\times$ C196	68 80	CMOS	KB/KC	✓	×	×		8 路	低功耗
8 $\times$ C194	52 80	CMOS	KB/KC	✓	×	×			高速
8 $\times$ C198	52 80	CMOS	KB/KC	✓	×	×		4 路	增加型

注: BH 版: A/D 变换器带采样保持电路, 其变换速度更快, 为 22 $\mu$ s。片内还增设了波特率发生器等。

JF 版: 片内有 0100H~01FFH 的内部可执行 RAM 空间, 该区域既可存放可执行的指令, 也可存储数据, 片内存储器空间扩展成 2000H~5FFFH。

由表 1 可知:

●  $8 \times 9 \times$  中第一个  $\times$  有三种可能, “0”为仅具有 CPU 的基本型, “3”为含有片内掩膜 ROM(8K)型, “7”为片内含有 EPROM(8K)型;

●  $8 \times 9 \times$  中的第二个  $\times$  有两种情况, 即“4”、“5”为 48 引脚型, “6”、“7”为 68 引脚型, 在这两种情况中, “5”和“7”为含有 A/D 变换器型;

● 每种机型又有不同版本(90 版, BH 版, JF 版), 这些版本既表示出现的早晚顺序, 又表示其功能的改进情况(详见表中注释);

● 对于 8098 系列, 只有三种基本机型, 即 8098BH、8398BH 和 8798BH;

● 对于  $8 \times C19 \times$  系列, 其中  $\times$  的含意同  $8 \times 9 \times$ 。 $8 \times C19 \times$  是 CMOS 工艺低功耗增强型, 其速度更快, 功能更多。是近年来才推出的机型;

●  $8 \times 9 \times$  系列的指令系统完全兼容,  $8 \times C19 \times$  的指令系统又有增加, 本教材仅论述 8098 的原理与应用;

●MCS-96 系列家族中各种机型的封装类型有很多种。8098 使用的是 48 引脚 DIP 双列直插封装。

8×9×系列单片机的设计思想,主要是针对 8 位机(如 51 系列)不能胜任驱动新型录像机、激光盘驱动器、高速马达和快速打印机等的客观需要而诞生的。由于其技术先进,功能齐全,获得了广泛的应用。Intel 公司推荐的典型应用范围如下:

工业控制方面:电机控制,工业机器人,离散与连续过程控制,数字控制,智能传感器;

仪器仪表方面:医疗机械,液体和气体色谱仪,示波器;

消费品方面:录像机,激光盘驱动,高级电视游戏;

电讯方面:调制解调器,智能线路控制;

导航与控制方面:图形终端,彩色与黑白复印机,温式硬盘驱动,磁带机驱动,打印机;

汽车方面:点火控制,变速器控制,燃料控制,防滑刹车,排气控制。

### 三、MCS8098 单片机的主要性能和特点

8096 系列单片机虽然得到了广泛应用,但是由于它通常采用 68 脚 PLCC(plastic leaded chip carrier)封装,价格昂贵,不利于普及。Intel 公司在 8096 系列推出之后,为了便于普及又推出了其简化型 MCS8098 单片机。MCS8098 单片机由于明显地具有高集成度,高抗干扰能力和高性能价格比这三高优势。因此一出现就受到了人们的青睐,在出现当年就生产了 20 万片,占了当年世界 16 位单片机市场的 90%。近年来发展一直很快,目前正在逐步取代 8 位单片机,成为单片机市场的主流机型。

MCS8098 单片机的主要性能特点如下:

(1)16 位的 CPU 由于 CPU 采用了先进的程序存储器与数据存储器合用的普林斯顿(Princeton)结构。它抛弃了 8 位机采用的传统累加器结构,使用了寄存器—寄存器结构,实现了 CPU 直接面向片内 256 个字节寄存器进行访问,并且其中 232 个可以用做通用寄存器,不用再分区,都可做累加器使用。从而在根本上消除了累加器的“瓶颈”现象,提高了操作速度和数据吞吐能力。

(2)丰富高效的指令系统 公开发表的指令系统共 100 条,可以支持字节(8 位)和字(16 位)操作,部分指令还可以支持双字(32 位)操作(如 16 位×16 位乘法指令和 32 位÷16 位的除法指令),还可以直接对有符号数和无符号数进行操作。此外还有整数符号扩展指令和长整数规格化指令(便于浮点数运算)等。因此可以说 8098 单片机的指令系统是丰富而高效的。

在典型时钟频率为 12MHz 的情况下,指令执行的状态周期 T 为时钟周期的 3 倍,即  $T = 3/12\mu s = 0.25\mu s$ 。因此一条指令的执行时间最短为  $1\mu s(4T)$ ,最长为  $9.5\mu s(38T)$ 。而 51 系列单片机( $T = 12/6\mu s = 2\mu s$ ),执行一条指令最短为  $2\mu s(1T)$ ,最长为  $8\mu s(4T)$ 。以同样执行一条 8 位无符号数乘法指令为例,51 机为  $8\mu s(4T)$ ,而 98 机仅需  $4.25\mu s(17T)$ ,可见 8098 单片机的运算速度是很快的。

(3)内含 4 路 10 位高速 A/D 变换器 完成一次 A/D 变换仅需  $22\mu s$ (BH 版,)并且带有采样保持器。这一点充分体现了 8098 单片机具有高性能价格比的优势,为实现高速数据采集提供了方便。

(4)提供了一路脉宽调制输出 PWM PWM 输出的脉宽调制信号,可以直接驱动步进电动机。将此输出信号进行滤波,又可获得 0~5V 的直流信号,从而可以实现 8 位 D/A 变换,不

必再外接 D/A 变换器。用此功能还可以产生锯齿波和三角波等许多有用的信号。

(5)具有全双工串行口 由于采用了双缓冲器结构,可以完成全双工(同时收发)串行接收和发送数据的操作(共有四种工作方式)。此外该机串行口功能还有两个特点:一个是设有独立的波特率发生器,不会因为串行口的使用而影响其他功能;一个是可以利用调整 HSO 功能构成软件串行口,从而使串行口功能更加丰富。

(6)具有高速 I/O 功能(HSI,HSO) 这是 8098 单片机除含有 A/D 变换器功能之外,与 MCS-96 系列其他单片机一样所具有的特殊功能。

HSI 功能可以利用相对于片内定时器 T1 产生的实时时钟,记录某个外部事件发生的时间(最多记录 8 个事件)。可以用来测量脉冲信号和马达转速等。

HSO 功能可按规定的时间去触发一个或多个内部或外部事件。如复位定时器 T2、启动 A/D、触发软件定时器、产生中断、输出 PWM 信号驱动步进电动机等。

这里“高速”的含义是,不用 CPU 干预就能自动完成,其运行速度必然很快。

(7)具有灵活多用的并行 I/O 口 8098 单片机在受 48 个引脚的局限下,为了尽可能地保留 8096 单片机的各种功能,就必须采取相应措施压缩一些并行口。即将 P0 和 P2 口都从 8 位压缩成为 4 位,P1 口全部删掉,P3 和 P4 口继续保留。因此 8098 共具有四个 I/O 口(共 24 个管脚)功能。其中大部分为复用口。

P3 与 P4 为地址/数据总线复用口。P3 既可作为 8 位数据总线口(8 位),也可作为地址总线的低 8 位口。P4 口既可作为地址总线高 8 位口,也可单独作为 8 位并行口使用。可见 8098 单片机是一种准 16 位机(地址总线 16 位,数据总线为 8 位)。之所以这样做,目的是为了与现行 8 位机的接口芯片相匹配,即所有 8 位机使用的接口芯片均可直接进行配接,于是既保留了 8096 单片机的先进功能,又便于普及和应用,从而降低了开发成本。

(8)具有 9 个中断向量 17 个中断源 在 9 个中断向量中,除软件陷阱中断由开发系统占用之外,其余 8 个全部提供给用户使用。其中有的一个中断向量还可以控制多个中断源。8098 共可提供 17 种中断源。

(9)设有三个 16 位硬件定时器 定时器 T1,可用做记录事件所发生的时间;定时器 T2,可用做对外部事件进行计数;监视定时器 WDT,可在系统发生软、硬件故障时,定时(16ms)自动地迫使系统复位,从而提高了系统软件的运行安全性和抗干扰能力。

(10)具有四个软件定时器功能 此功能是高速输出 HSO 功能中的一种特殊应用。它可以用来激活软件定时器产生中断,在某种意义上可以代替硬件定时器功能,使 MCS8098 单片机的定时器功能更加丰富。

(11)时钟频率典型值为 12MHz 从而保证了指令的运行速度。

## 习题与思考题

1. 数字电子计算机的基本组成和基本功能是什么?
2. 什么是微型计算机和单片机?
3. 8098 单片机的家族有几种机型,它们有何不同?
4. 8098 单片机的三高优势是何含义?
5. 8098 单片机的性能特点有那些?

# 基础篇

在学习 MCS8098 单片机之前,了解与熟悉微型计算机的有关基础知识,是非常必要的。可以为从零起点开始直接学习 MCS8098 单片机打下基础。

## 第一章 计算机常用的数制与码制

当今数字电子计算机(以后简称计算机)几乎都是用具有二值逻辑的数字电路制成的,因此计算机内部整个运行过程都必须采用二进制数,称之为机器数。也就是说,计算机中采用的最基本数是二进制数。有时为了表示方便,还要将其变换成十六进制或八进制数。

然而,人类在社会生活和科学计算中经常使用的是十进制数。十进制数的优越性是显而易见的。十进制数不仅可以表示数的大小,正负,还能表示整数和小数,定点数和浮点数等等,而且运算起来也很方便。那么如何才能让计算机中使用的二进制数也具有十进制数的同样功能,便提出了计算机中数的表示法问题。

不仅如此,计算机还要与外界事物发生联系。为了使这种联系更方便,通常还要对机器数进行适当编码。在计算机中经常使用的编码有 BCD 码和 ASCII 码等。

可见,首先学习和掌握计算机经常使用的数制与码制,是学习包括单片机在内的任何计算机所不可缺少的必备知识。

### § 1.1 计算机中数的表示法

#### 一、机器数的表示范围及其真值

计算机中数据的传送、存贮、运算都是以二进制数出现的。根据计算机数据总线的宽度可以确定计算机的字长,8 位机即为 8 位字长计算机,16 位机即为 16 位字长计算机。计算机字长一旦确定了,机器数的二进制数位数也就确定了。对于 8 位机,机器数的表示范围为  $(00000000)_2 \sim (11111111)_2$ ,对应十进制数为  $0 \sim 255$ 。对于 16 位机,机器数的表示范围为  $(00000000\ 00000000)_2 \sim (11111111\ 11111111)_2$ ,对应十进制数为  $0 \sim 65535$ 。

机器数在其变化范围内可以出现任意值,还常将其对应的十进制数称之为该机器数的真值。为了能够得到真值,必须首先知道二进制数的表达式,即一个任意正整数,可以用  $n$  位二进制数  $B$  表示成:

$$B = B_{n-1} \cdot 2^{n-1} + B_{n-2} \cdot 2^{n-2} + \dots + B_1 \cdot 2^1 + B_0 \cdot 2^0 = \sum_{i=0}^{n-1} B_i \cdot 2^i \quad (1.1)$$

式中:  $B_i$  为各位二进制数的取值,只能取“0”或“1”,  $2^i$  为二进制数各位所代表的权重。现将 16 位二进制数各位的权重列表如下:

i	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2 <sup>i</sup>	32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1

此表对以后的学习很有帮助。在 16 位二进制数范围之内,使用它实现二进制数与十进制数的正整数之间正、反变换就很容易了。例如:

$$(01010101)_2 = 64 + 16 + 4 + 1 = 85$$

$$(01110011)_2 = 64 + 32 + 16 + 2 + 1 = 115$$

$$206 = (11001110)_2$$

此外,在实际工作中为了表达方便,通常还把一个 8 位二进制机器数定义为一个字节 (BYTE),把一个 16 位二进制数定义为一个字 (WORD),把一个 32 位二进制数定义为一个双字 (DOUBLE-WORD)。

## 二、机器数的十六进制和八进制表示法

虽然计算机中都是按二进制数运算的,但它表示的位数很长,理解和记忆都很困难。如果将其变换成为八进制数(逢 8 进 1)或十六进制数(逢 16 进 1)来表示,这种困难将会得到解决,使用起来也更加方便,因此在实际工作中,十六进制数和八进制数都得到了广泛应用。

将二进制数变换成十六进制数,其实并不难。十六进制数的常用码符如下:

十进制数	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
十六进制数	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

对于 8 位二进制数  $(0000\ 0000)_2 \sim (1111\ 1111)_2$ ,对应的十六进制数为 00H~FFH。对于 16 位二进制数对应的十六进制数则为 0000H~FFFFH。

可见一个十六进制数码符所代表的数值,恰好与一个四位二进制数所代表的数值相当。因此只要将每四位二进制数分成一组,然后用与之对应的十六进制数码符来代换,就可以完成二进制数到十六进制数的变换。为了与其它进制数表示相区别,还需要加上后缀“H”。

将二进制数变换成八进制数的方法,与变换成十六进制数的方法类似,将每三位二进制数分为一组,然后用对应的八进制数代码替换即可。八进制数的表示后缀为“O”。

一并举例如下:

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
二进制数	0	0	1	0	0	0	0	0	1	0	1	1	1	0	0	0
对应八进制数	0	2		0		2		7		0						
对应十六进制数	2			0			B			8						
二进制数	0	0	1	1	1	1	1	1	1	0	1	1	0	0	0	1
对应八进制数	0	3		7		7		3		1						
对应十六进制数	3			F			D			9						

## 三、整数表示法

在计算机中,整数可以用无符号数和带符号数来表示。

### 1. 无符号数

上面提到二进制机器数就是无符号数,无符号数只用来表示正整数或绝对值,对于一个 n 位二进制数,其无符号数的表达式为:

$$D = \sum_{i=0}^{n-1} a_i \cdot 2^i \quad (1.2)$$

式中  $a_i$  为某一位的取值(0 或 1),  $a_{n-1}$  为最高位(MSB),  $a_0$  为最低位(LSB)。无符号数可以表达数的范围为:

单字节(一个字节为 8 位)  $0 \leq D \leq 2^8 - 1$  即 0~255

双字节(一个字为 16 位)  $0 \leq D \leq 2_{16} - 1$  即 0~65535

双字(32 位)  $0 \leq D \leq 2^{32} - 1$  即 0~4294967295

无符号数的表达精度可以用  $\pm 1\text{LSB}$  所引起的相对误差来表示,即单字节(单精度)为  $1/255 \approx 0.4\%$ ,双字节(双精度)为  $1/65535 \approx 1.5 \times 10^{-5}$ ,可见在一般情况下,采用双字节(双精度)无符号数来表示正整数已足够精确了。

## 2. 有符号数

在计算机中,只能表示正整数是不够的,还需要能表示负整数才行,为此引进了有符号数表示法。所谓有符号数,就是把  $n$  位二进制数的最高位规定用做符号位,该位为“0”表示是正数,该位为“1”则表示是负数。一个  $n$  位有符号数的表达式为:

$$D = -a_{n-1} \cdot 2^{n-1} + \sum_{i=0}^{n-2} a_i \cdot 2^i \quad (1.3)$$

式中  $a_{n-1}$  为符号位,  $a_{n-1} = 0$  表示其值为正数,  $a_{n-1} = 1$  表示其值为负数。有符号数可以表达的数的范围为:

单字节(8 位)  $-128 \leq D \leq +127$

双字节(16 位)  $-32768 \leq D \leq +32767$

四字节(32 位)  $-21474838 \leq D \leq +21474837$

## 3. 带符号数的原码、反码和补码表示法

在计算机中为方便地实现对带符号数进行软件或硬件快速运算操作,通常还要使用原码、反码和补码表示法。在微型计算机中主要使用的是补码表示法。

(1)原码表示法 采用上面提到的那种带符号表示法得到的有符号数,就是带符号数的原码。例如:

$$[+83]_{\text{原}} = 01010011$$

$$[-83]_{\text{原}} = 11010011$$

$$[+0]_{\text{原}} = 00000000$$

$$[-0]_{\text{原}} = 10000000$$

**注意:**原码表示法中,0 有  $[+0]_{\text{原}}$  和  $[-0]_{\text{原}}$  两种表示法。

原码表示法虽然简单易懂,但在计算机中对其进行运算操作却很困难,对两个原码进行运算,计算机必须首先判断符号,才能确定是进行加法还是进行减法运算,而且还不能使用同一种运算电路来实现加、减法运算,因此必然要影响运算速度。

为了克服原码表示法的这些缺点,从而又引入了反码和补码表示法。

(2)反码表示法 在已知带符号数原码的情况下,由原码求得其反码是很容易的。即:对于正数,反码与原码相同;对于负数,则应变换成与其原码相对应的反码。求反码的具体做法是,其最高符号位“1”保持不变,将其余各位逐位取反即可(原为“0”变成“1”,原为“1”变成“0”)。例如:

原 码	反 码
$[+101]_{原} = 01100101$	$\longrightarrow [ +101]_{反} = 01100101$ (相同)
$[-101]_{原} = 11100101$	$\longrightarrow [-101]_{反} = 10011010$
$[+127]_{原} = 01111111$	$\longrightarrow [ +127]_{反} = 01111111$ (相同)
$[-127]_{原} = 11111111$	$\longrightarrow [-127]_{反} = 10000000$
$[+ 0]_{原} = 00000000$	$\longrightarrow [ + 0]_{反} = 00000000$ (相同)
$[- 0]_{原} = 10000000$	$\longrightarrow [- 0]_{反} = 11111111$

**注意:**在负数的反码中,符号位之后各位不代表该数的真值,若想求得其真值,还需再将其还原成原码才行。上例中 $[-101]_{反}$ 的真值为 $-101$ ,而不是 $-26$ 。在反码表示法中,同原码一样,也有 $[+0]_{反}$ 和 $[-0]_{反}$ 之分,在微型计算机中,这种反码表示法很少使用,故不详述。

(3)补码表示法 在微型计算机中通常是使用补码形式来表示带符号数。补码的概念是从补数的概念引入的。一个系统所能表示的最大数(如测量系统的量程)通常称之为模。用 $K$ 来表示。当满足 $Z = nK + Y$ 的关系时(式中 $n$ 为正整数,一般取为1),则可称 $Z$ 和 $Y$ 互为补数。实际上 $Z$ 和 $Y$ 表示的是同一个事物。例如一个圆周角是 $360^\circ$ ( $K = 360^\circ$ ),对于 $+300^\circ$ 和 $-60^\circ$ 两个角度满足这种关系,即 $300^\circ = 360^\circ + (-60^\circ)$ ,因此 $+300^\circ$ 和 $-60^\circ$ 互为补数,且在物理上表示的是同一个角度。

对于一个 $n$ 位带符号数,其模为 $K = 2^n$ ,原码与其补码的关系表达式为:

$$[X]_{补} = 2^n + [X]_{原} \quad (1.4)$$

已知原码便可以求得补码,具体做法是(证明略),对于正数其补码与原码相同,(正数的原码、反码和补码都一样),对于负数其补码与原码不同。具体求补码的做法是:将原码的反码再加1即为其补码。亦可说成是,将原码符号位保持不变,其余各位逐位取反再加1,简称为“求反加1”法。例如:

原码	反码	补码
$[+103]_{原} = 01100111$	$\longrightarrow [ +103]_{反} = 01100111$	$\longrightarrow [ +103]_{补} = 01100111$ (相同)
$[-103]_{原} = 11100111$	$\longrightarrow [-103]_{反} = 10011000$	$\longrightarrow [-103]_{补} = 10011001$
$[+ 0]_{原} = 00000000$	$\longrightarrow [ + 0]_{反} = 00000000$	$\longrightarrow [ + 0]_{补} = 00000000$ (一样)
$[- 0]_{原} = 10000000$	$\longrightarrow [- 0]_{反} = 11111111$	$\longrightarrow [- 0]_{补} = 00000000$ (有溢出)

**注意:**负数符号位之后各位,并不代表其真值,而是其补数。上例中 $[-103]_{补}$ 的真值为 $-25$ ,不是 $[-103]_{原}$ 的真值 $-103$ ,两个真值之和恰为 $-128$ ,可见两个真值以模为 $-128$ ( $-2^7$ )互补。欲得补码对应原码的真值,还需将其再求补还原成原码才行。对上例有 $[[[-103]_{补}]_{补}]_{补} = 11100111 = [-103]_{原}$ 。在补码表示法中,0只有一种表示形式,即 $[0]_{补}$ 。

#### 四、小数表示法

计算机中只有“0”和“1”两种表示信息的数码,没有可用来表示小数点“.”记号的数码。为了能够同样表示小数,只能采取人为规定小数点的位置来解决。因此产生了定点(小)数和浮点(小)数两种表示法。

为了实现用二进制数能够表示小数,可以把一个二进制数 $N$ 表示成:

$$N = \pm S \times 2^{\pm i} \quad (1.5)$$

式中 $S$ 为数 $N$ 的尾数,是一个二进制小数,用来表示小数部分的全部有效位,其前面的符号

“±”为数符；j 为数 N 的阶码，是一个二进制整数，用来决定小数点的位置，其前面的符号“±”为阶符。

### 1. 定点数表示法

定点数表示法就是人为地将小数点规定在二进制代码中的一个固定位置上，对于整数，通常是将小数点固定（隐含）在数值部分最低位（LSB）的后面，用来表示整数；对于小数，则通常是将小数点固定（隐含）在数值部分的最高位（MSB）的前面，表示的是纯小数。用此法规定所得到的数称为定点数。对于定点数，表达式中的阶码  $j = \text{常数}$ ，即表示小数点位置固定。在计算机中经常使用的定点数是定点纯小数。对于纯小数  $j = 0$ ，则有

$$N = \pm S \quad (1.6)$$

例如：二进制小数可表示为：

$$\begin{array}{r} +0.1010110 \quad 0 \quad . \quad 1010110 \\ -0.1010110 \quad 1 \quad . \quad 1010110 \end{array}$$

数符    隐含小数点    尾数 S

因为小数的尾数部分共 7 位，各位所代表的真值如下：

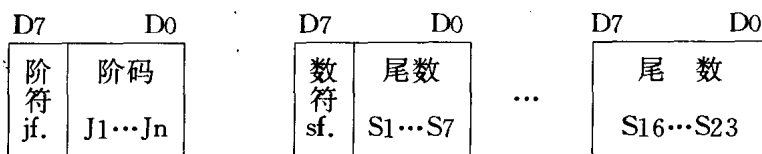
$2^{-1}$	$2^{-2}$	$2^{-3}$	$2^{-4}$	$2^{-5}$	$2^{-6}$	$2^{-7}$
0.5	0.25	0.125	0.0625	0.03125	0.015625	0.0078125

可得上述例中的二进制小数真值为  $\pm 0.671875$ 。对于  $n$  位二进制小数，其数值表示范围为  $2^{-n} \leq |N| \leq 1 - 2^{-n}$ 。由于其值总是小于 1，当计算机进行这种定点小数运算时，其结果必须保证小于 1 才行，否则将会产生“溢出”。

### 2. 浮点数表示

定点数表示法中小数点位置固定不变，而在浮点数表示法中，小数点的位置则不固定，可以浮动，故称之为浮点数。其表达式同(1.5)式，对于浮点数，阶码  $j \neq \text{常数}$ 。

以具有 32 位（四字节）的多字节浮点数为例，在计算机中的数据格式如下：



可见，浮点数是由阶码及其阶符，和尾数及其数符两个部分组成的。其中阶码部分（通常为一个字节）是一个带符号的整数（ $jf = 0$  表示阶码为正数， $jf = 1$  表示阶码为负数），用来表示尾数中的小数点向左或向右移动的位数，通常用补码形式表示。其中尾数部分（通常为 2~6 个字节）的结构，同上述纯小数定点数表示方式相同，即约定小数点在数符与尾数之间，尾数通常用原码形式表示。其值在 0~1 之间。

浮点数尾数的位数可以表示出数值的精度，精度要求越高，尾数的位数则应越多。精度和尾数位数  $n$  的关系为：

$$\text{精度} = \frac{1}{2^n}$$

例如尾数为 2 字节（16 位）的浮点数，其数值表示精度为  $1/65536$ 。这种浮点数的数值表示范围为：



$$\pm 1 \times 2^{-1} \sim \pm 1 \times 2^{+1}$$

对于具有 4 位尾数、2 位阶码的单字节浮点数,其数值表示范围为:  $\pm (1 \times 2^{-4} \sim 1 \times 2^3)$ , 即  $\pm (0.0625 \sim 8)$ 。对于具有 7 位阶码、15 位尾数的三字节浮点数,其数值表示范围为:  $\pm (1 \times 2^{-126} \sim 1 \times 2^{+127})$ , 即  $\pm (2.94 \times 10^{-39} \sim 1.70 \times 10^{38})$ 。

可见浮点数的数值表示范围是很大的,比具有相同尾数位数的定点数大得多。因此在对数值范围很大的数进行运算时,要使用浮点数表示法。

在计算机中使用浮点数表示法时,尾数通常用原码表示,因此可以直接使用四则运算指令来完成浮点数尾数的运算,从而大大提高了浮点数的运算速度。

在使用计算机进行浮点数运算时,为了提高精度,通常还要对浮点数进行规格(范)化处理,即必须对尾数进行移位操作,使其达到尾数最高位为“1”为止。(详见规格化指令 NORML 的用法)。

浮点数运算,同样也存在溢出问题,应予以注意。此外,浮点数表示法的格式有很多种,很难统一,使用时应注意,这里不再详述。

## § 1.2 二进制数的四则运算

运算功能是计算机的最基本功能之一,四则运算是一切数值计算的基础。学习和掌握二进制数四则运算的指令系统很重要,但是,如果不了解和掌握二进制数的四则运算的原理与方法,就不能很好地理解和运用好这些指令,必然要影响软件编程的质量。对于学习 8098 单片机也不例外。二进制数有无符号数和有符号数两种表示法,其四则运算方法也不同。

### 一、无符号二进制数的四则运算

无符号二进制数的四则运算与我们习惯使用的十进制数(正整数)的四则运算很相似。只不过这里仅有“0”和“1”两种代码。运算时是“逢二进一”,而不是“逢十进一”,是“借一为二”而不是“借一为十”。只要掌握好这一点,对无符号二进制数的四则运算就很容易理解了。下面举例加以说明。

#### 1. 加法举例

低位进位	111		
被加数	1010	1100	$\rightarrow (ACH)_{16} \rightarrow (172)_{10}$
加 数	0011	1010	$\rightarrow (3AH)_{16} \rightarrow (58)_{10}$
和	1110	0110	$\rightarrow (E6H)_{16} \rightarrow (230)_{10}$

可见这里每位相加时,要涉及到两个相加的数和低位进位共三个二进制数的求和。在计算机中实现这种运算的最基本硬件单元电路是“全加器”。

#### 2. 减法举例

向高四位借位	0001		
低位借位	1010	101	
被减数	1011	0010	$\rightarrow (B2H)_{16} \rightarrow (178)_{10}$
借位后被减数	0000	1000	
减数	0101	1101	$\rightarrow (5DH)_{16} \rightarrow (93)_{10}$
差	0101	0101	$\rightarrow (55H)_{16} \rightarrow (85)_{10}$