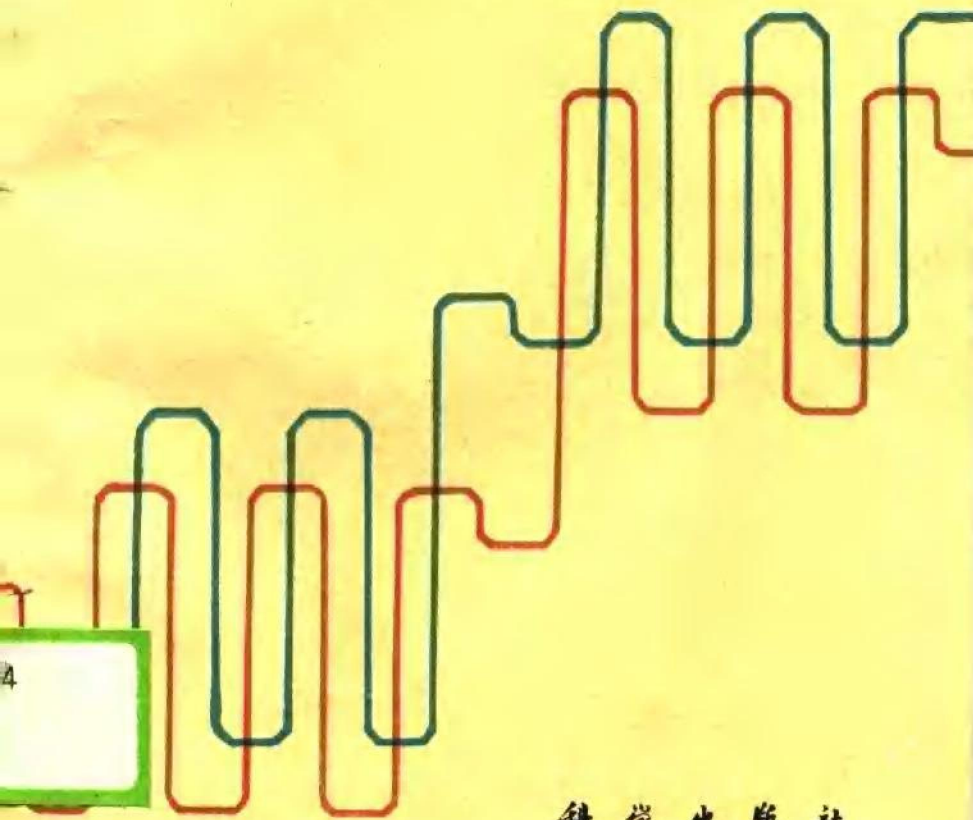


微控制器在传热学中的应用

俞忠原 编著



科学出版社

序

微型计算机技术迅猛发展,其应用日益深入到传热学数值计算、传热学实验测试及传热设备控制等各个方面。从事传热技术工作的读者迫切需要一本能快速掌握微型计算机在传热学中的应用的实用书籍,俞忠原同志的《微控制器在传热学中的应用》一书正是为此目的而编写的。书中从应用的角度,重点介绍了微处理器,时钟,存贮器,接口部件和 A/D、D/A 转换等微控制器的基本部件,实时数据采集与控制软件的编制方法,以及双机通讯、滤波、系统仿真和微控制器加固等实际应用技术。

本书的特点是将基础理论与实际应用有机地结合起来,遵照循序渐进的原则,由浅入深地加以阐述。同时,为了帮助读者理解与应用所学内容,书中还列举了许多从科研项目中精选出来的实例。相信从事传热技术工作并有心于微控制器应用研究的读者,定能从本书中获得有益的启示。

孔祥谦

1989 年 4 月于上海交通大学

前 言

随着微型计算机应用技术在各个领域的日益深入,微控制器在传热学实验数据采集与传热工程实时控制中也得到了广泛的应用。

本书从实际应用角度出发,介绍了微控制器硬件与接口电路设计,实时控制软件编制的基本技术,数据采集、滤波与标度变换的实用方法,数字控制器基本设计与实现办法,以及为了开发硬件电路与实时软件而进行的系统实时仿真与双机通讯技术。为了帮助广大读者尽快掌握并应用微控制器技术,书中还列举了许多实际科研项目,以此说明各种硬件构成的原理与实时软件的编制方法。

在本书编写过程中,承蒙中国科学院计算技术研究所栾毓敏副研究员,上海交通大学孔祥谦教授,哈尔滨船舶工程学院丁放副教授的指导与审阅,在此一并致谢。限于编者水平,书中缺点错误在所难免,恳请读者批评指正。

编著者

1988 年 10 月

目 录

第一章 微控制器的基本知识.....	1
§ 1-1 微控制器的构成与工作原理	1
§ 1-2 数制和编码	9
§ 1-3 应用软件与程序语言	18
§ 1-4 指令系统及其功能	22
第二章 实时应用程序设计基础.....	37
§ 2-1 程序设计的步骤与方法	37
§ 2-2 伪指令及宏定义	46
§ 2-3 中断及中断服务程序	52
§ 2-4 实时应用试验程序	60
§ 2-5 实时应用程序的开发	64
第三章 微控制器的硬件组装.....	69
§ 3-1 CPU 与存储器.....	69
§ 3-2 Z80 计数器/定时器 (CTC)	76
§ 3-3 Z80 并行 I/O 接口 (PIO)	85
§ 3-4 8212 与 8255 接口.....	96
§ 3-5 中断控制器 8259 与定时器 8253	105
§ 3-6 数字设备接口	113
§ 3-7 辅助电路	124
§ 3-8 双机通讯方法	134
第四章 过程通道和数据采集系统.....	144
§ 4-1 过程通道的组成和功用	144
§ 4-2 D/A 与 A/D 转换技术.....	157
§ 4-3 信号的处理方法	171
§ 4-4 数据的分析	186

§ 4-5 数据采集系统	203
第五章 实时控制系统	216
§ 5-1 实时控制系统与数学模型	216
§ 5-2 控制系统仿真	228
§ 5-3 数字控制器	241
§ 5-4 自校正调节器	254
§ 5-5 执行元件的控制	259
第六章 微控制器系统的加固	274
§ 6-1 加固微控制器的技术途径	274
§ 6-2 系统可靠性分析与设计	275
§ 6-3 信号测量、传输与抗干扰	283
§ 6-4 光电耦合器的应用	287
§ 6-5 电源系统及地线系统	292
§ 6-6 故障诊断与容错技术	296
参考文献	301

第一章 微控制器的基本知识

§ 1-1 微控制器的构成与工作原理

1-1-1 微控制器的工作原理

微控制器在传热学中的应用主要是数据采集与实时控制,即在传热实验或传热设备中,实时地进行数据采集,按传热模型与相应的计算方法进行数据处理,打印所要求的传热参数;同时可依据对象的不同要求,实时地进行动态控制,以构成高精度、智能化的控制仪表,例如风洞强迫对流换热测试仪,加热炉控制仪,导热系数测定仪等都是具体应用的实例。

微控制器是用于实时测试与控制的最小微机系统,其工作原理可简明地归结为:按序取指令,并依次执行指令。如图 1-1 所示,微控制器由微处理器 (MPU)、存贮器 (ROM 与 RAM)、时钟 (ϕ)、输入输出 (I/O) 接口、模/数与数/模 (A/D 与 D/A) 转换器组成,通过数据总线 (DB)、地址总线 (AB) 和控制总线 (CB) 连成整体。指令程序按地址顺序存放在只读存贮器 (ROM) 中;数据存放在随机存贮器 (RAM) 中,也按地址存放。进行实时数据采集时,一般要求温度、压力、流量等多路数据同时采集,故输入口要编上地址号码。同样,输出量经输出口显示、打印与控制,故输出口也要编上地址。这样,从微控制器的内部看,各输入/输出口和各存贮单元就如同旅馆里的房间,被全部编上地址号码;一般地址为 16 位,共有 64K ($1K = 1024$)。

微控制器按程序计数器 PC 指定的地址取出指令并执

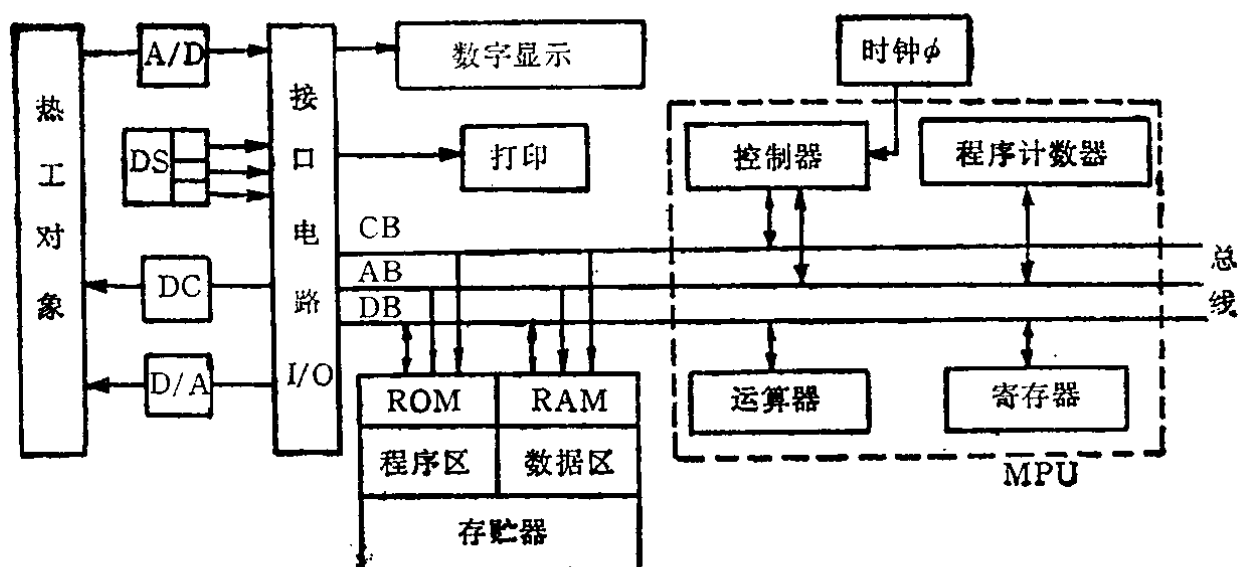


图 1-1 微控制器构成图

行，PC 原则上是一个加 1 计数器，它保证程序指令按序执行。

微控制器采集的数据是在运算器中进行加、减、乘、除和逻辑运算的，其计算结果暂存在数据寄存器里。要特别指出的是，采集的数据、运算的数据以及存放的指令程序都是二进制数码。从这个意义上讲，微控制器的功能就是对二进制数码进行采集、计算、控制及输出。在整个工作过程中，控制器协调和控制各个单元的动作，按时钟发出定时信号，以实现实时数据采集与控制。

微处理器和内存贮器通常称为主机，外围输入/输出设备称为外围设备(简称外设)，两者之间的电路称为接口电路，一般都是由大规模集成电路制成的芯片，因而简化了接口设计的任务且便于使用。

热工参数测量传感器的输出量一般为模拟量(电压或电流)，需通过 A/D 转换器变为数字量(二进制数码)才能输入主机。同样，输出的数字量也常需由 D/A 转换器变为模拟量，以驱动执行机构(电机等)。当然，有的执行机构(如步进电

机)可直接由数字量控制;也有数字量的测量传感器。

1-1-2 微控制器的硬件构成

微控制器由硬件与软件两部分组成,硬件主要由以下几部分构成。

1. 微处理器 (MPU)

微处理器也称中央处理单元 (CPU), 通常由一块芯片构成的,称为单片 CPU。CPU 具有两大功能: 1) 运算功能,包括四则运算、逻辑运算与移位等。2) 控制功能,包括指令的读出、译码、执行以及地址总线、数据总线、寄存器组的协调控制,因此其结构除控制逻辑电路外,主要由运算器、控制器、寄存器等经内部数据总线连接构成,如图 1-2 所示。

算术逻辑单元 ALU, 即运算器。它能完成加、减、乘、除,逻辑与、或、比较、判断、移位等操作,但由于其电路只是 8 位并行加法器,只有直接的加法、减法指令,没有乘除、开方等复杂的运算指令,这需要自行编制子程序来实现。

控制器由三部分构成: 1) 定时器由时钟脉冲与节拍脉冲发生器组成,作为定时协调器给各单元送入时间控制信号。2) 指令部件由指令寄存器、译码器、机器周期逻辑单元组成,它给出按序执行指令的地址,从内存中取出指令,放入指令寄存器,译成相应电平的控制信号,通过定时与控制电路,在规定时刻发出命令。3) 控制电路,接受定时器及指令部件的时序及地址信号,并分解成若干微操作过程,产生整机各部分协调工作的各控制电位。

累加器 A 常与状态标志寄存器 F 联用,在算术和逻辑操作时,A 中的内容必为一个操作数,运算结果又存放于 A 中,而运算结果的特征,如正、负、零或进位、溢出等状态特征寄存

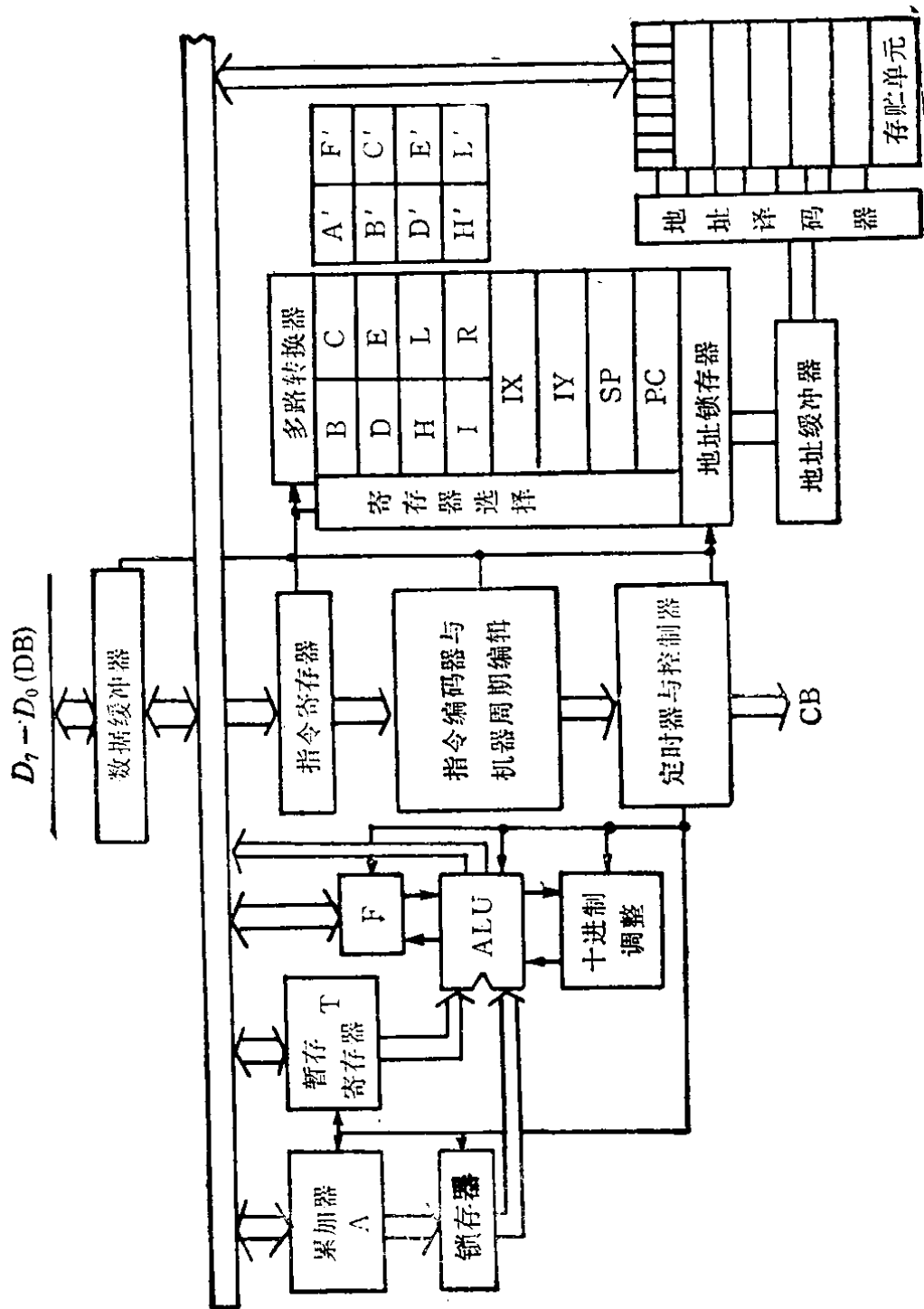


图 1-2 Z80-MPU 方框图

于F中。这样,在编制软件时可充分利用状态标志进行判别,以实现循环或跳转、分支等功能。

CPU 中有两组通用寄存器 B, C, D, E, H, L 与 B', C', D', E', H', L'。它们可分别作 8 位寄存器使用,接受从内存中调来的 8 位信息。若信息为指令,则送向指令译码器;若信息是数据,则送向 ALU 或累加器 A;若信息是地址,则送向 PC 计数器或地址寄存器 AR。它们又可以两个连用,形成 BC, DE, HL, BC', DE' 与 HL' 六对 16 位寄存器对,以存放操作数的 16 位地址或 16 位数据。这便于成组数据交换,提高操作速度。

专用寄存器有以下几种:

(1) 程序计数器 PC 是 16 位的,其内容总是下一条要执行指令的地址,它能指定 64K 字节的存贮空间地址。CPU 根据被执行指令字节的长度,自动使 PC 的值加 1, 2, 3 或 4。在执行转移指令或调用子程序时,又能自动把转向地址赋予 PC,故 PC 是指令地址形成部件。

(2) 堆栈指示器 SP 也是 16 位寄存器,用于指示存贮器的地址,它能变更较复杂的程序执行顺序。堆栈是 RAM 中划出一块按先进后出原则组织的存贮区域,其指针始终指向堆栈顶部。调用子程序时,PC 内容自动压入堆栈保存,子程序结束返回主程序时,能自动回到下一条指令所在的地址单元。在编制中断子程序时,需用 PUSH, POP 指令,将子程序中用到的寄存器内容压入堆栈,以保护现场。子程序结束后,这些内容再弹回原寄存器,以恢复现场。这样便可实现多级中断、子程序嵌套等复杂操作。

(3) 变址寄存器 IX 与 IY 是两个独立的 16 位寄存器,用于存放过渡性的地址,以达到自动变址的目的。例如,可用指令使它加 1 或减 1 来改变地址。变址寄存器作为地址计数

器,使用时可使它们存贮一个基地址,加上由程序指令给定的相对位移量来构成操作数的有效地址. IX 与 IY常与标志寄存器 F 联用,当其内容不断减 1 结果为零时, F 寄存器中的零位标志 Z 自动置 1. 利用这种特征来变址寻址可大大简化编程,提高机器效率与功能.

(4) I, R 寄存器是两个独立的 8 位寄存器, I 寄存器称为中断页面地址寄存器,用于存放中断矢量入口地址表的高 8 位地址(页地址),它与请求中断的外设提供的低 8 位地址一起构成中断矢量入口表的地址,一般由软件在初始化段中设定. R 称为刷新寄存器,机器对内存单元刷新时作地址计数器用.

微处理器的位数是按总线来分类的,即根据执行一条指令时机器每次接收和传送数据的位数决定的. 8 位机每次传送 8 位信息(Z80, 8085, M6800 等), 16 位机每次传送 16 位信息(8086, Z8000, M68000 等). 若在单片芯片里包含 CPU、存贮器和输入/输出,则称为单片机. 单片机也分为 4 位、8 位机.

2. 存贮器

存贮器是用来存放程序指令和数据的部件,所存信息均以二进制代码的形式存放. 存贮器分为两种:一种是只读存贮器(ROM),它在写入信息后只能进行读出操作,断电后信息仍能保持,故可用于存放固定不变的程序和数据,按其结构形式与写入方式又可分为: 1) 固体掩模编程 ROM,一般为生产厂家固化监控程序用,固化后不能再改写. 2) 用户可写一次而不可擦的 PROM. 3) 可用紫外线擦去内容,允许再写入新内容的 EPROM. 4) 可用电的方法擦去内容再写入的 E²PROM. 另一种是读写存贮器(RAM),又称为随机存贮

器,它能在系统(程序)控制下写入信息,又能读出信息。RAM按结构和性能分为静态和动态两类。静态 RAM 在不停电时信息永久保持,动态 RAM 内的信息只能保持几十毫秒,需每隔几毫秒刷新一次。无论静态或动态 RAM,在断电时信息自动丢失。因此在停电时仍需保持信息的场合,应采用不间断电源。目前广泛采用动态 RAM 的原因是因为它耗电少,封装密度高,价格便宜,速度快。

半导体存贮器的容量是用含有多少位来表示的,位数是2的乘幂,这些位可单独存取,或以4位或8位为一组存取。某型号的存贮器可能包含 1024×1 位,另一种可能为 256×4 位或 128×8 位,这几种都是 1024 位;但是 1024×1 位器件有 10 条地址引脚,要用 10 位地址来规定其中某一位的位置; 256×4 位器件只有 8 条地址引脚,用 8 位地址来规定 4 位一组的数据位置; 128×8 位器件则有 7 条引脚地址线来规定每个字节的位置。图 1-3 为存贮器的构成框图。这是

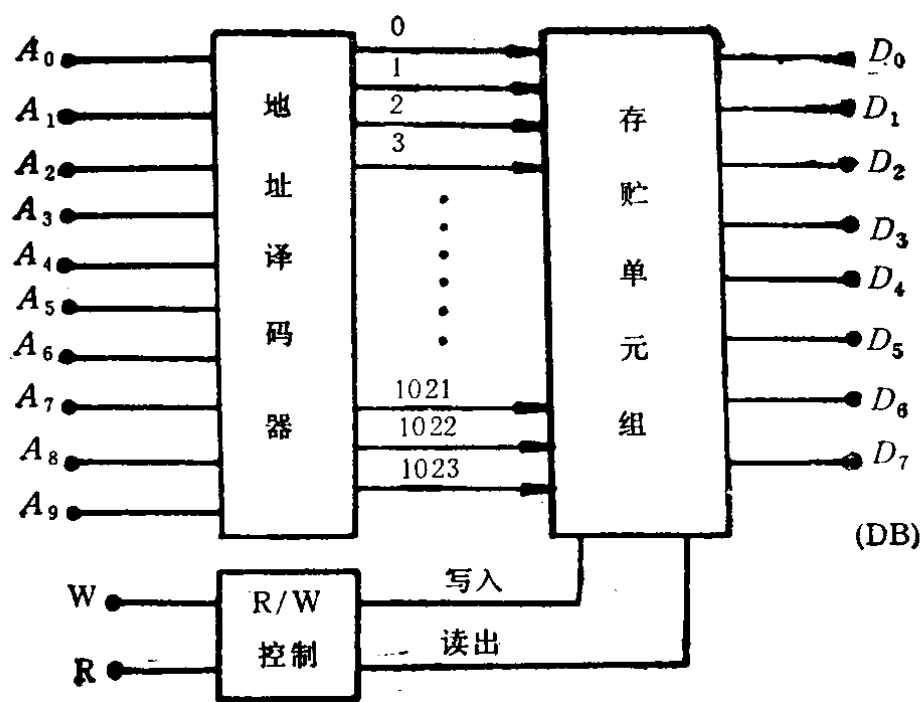


图 1-3 存贮器的构成

一个 1024×1 (1K 字节) 位的存贮器。 $A_0—A_9$ 为 10 个地址总线，经译码成 1024 根地址选择线。使用存贮器时，由地址总线选定相应的存贮地址，就可在数据总线上读出（或写入）8 位数据。一般 8 位机有 $A_0—A_{15}$ 共 16 条地址线，最多可指定存贮单元的地址为 $2^{16} = 65536$ 个。

存贮器的工作速度一般为 $2ns—2ms$ ，称为存取时间或存取周期。存贮器的速度与容量是微控制器设计时的两个重要指标。

3. 输入输出 (I/O) 接口

微控制器外围设备的费用往往占有很大比重，常用的输入设备有：按钮、数字开关，A/D 转换器、键盘和数字传感器等；常用的输出设备有：发光二极管 (LED)、打印机、D/A 转换器、光电隔离逻辑电路、声光报警装置等。输入输出接口的功能是：1) 在主机与外设之间进行数字量信息的串行或并行交换。2) 在主机与外围设备之间起时序控制与数据的缓冲作用。3) 进行模拟量(电压等)与数字量(二进制代码)的转换。4) 为主机提供外围设备的状态信息，如“准备好否”，“空与忙”等。5) 进行电平变换及 I/O 设备的地址译码。

由于大规模集成电路技术的发展，接口电路大都制成为通用的芯片，从而大大简化了接口设计的任务。这些接口大多是可编程序 I/O 接口，其使用十分方便。通用并行 I/O 接口芯片有 Z80 PIO, 8255, 6820 (PIA) 等，串行同步异步接口芯片有 Z80SIO, 8251, 6850 等。主机处理紧急事件的有效方式是中断方式，常用的中断芯片有 8214, 8259 等，计数和定时的中断接口片有 8253, Z80CTC, 6840 等。此外，专用外围接口芯片，如快速数据传送 Z80DMA，键盘/显

示控制器 8279, CRT 控制器 8275, 6845 等。

§ 1-2 数制和编码

微控制器的基本功能是进行二进制数的计算与处理。数包括指令、数值、地址、字符等信息,在微控制器的内部均以高、低电平表示,即 5V 与 0V 量化为 1 与 0 的二进制代码。例如,从压力传感器送来压力为 0—2kg 转换电流为 4—20mA 的信号,经 250Ω 标准电阻转换成 1—5V 电压信号,再经 8 位 A/D 转换器变换为 51—255 的二进制数,送入 8 位主机。因此在主机内二进制数 51 代表 0kg 压力,255 代表 2kg 压力。如果从差压传感器送来 ±0—2kg 差压作为控制信号,传感器输出也为 4—20mA,经 250Ω 标准电阻变为 1—5V 电压,经 A/D 变换为 51—255。差压是带符号(正、负)的数,在主机中计算时应以 +102 表示 2kg 差压,−102 表示 −2kg 差压,故需对 A/D 采入的数减去 153,得到带符号的数。在 8 位主机中,用第一位表示符号,0 为正,1 为负(数),其范围为 +127—−128。另外,在主机中运算的是二进制数,表示单元地址时常用十六进制数,而输出信息作数码显示时又常用十进制,因此需熟悉各种数制及码制,才能正确编制应用软件,使输入/输出各种信息的标度变换正确,以保证主机对有符号数或无符号数的计算及处理均正确无误。

1-2-1 数制

按位记数法的一般形式为:

$$N_b = \sum_{i=n-1}^{-m} a_i b^i \quad (1-1)$$

式中基数 b , 权数 b^i , 数字 a_i , 小数点前为 n 位, 小数点后为

m 位.

1. 二进制

基数为 2, 数字仅用 0 与 1, 按逢二进一原则进位. 例如:

$$N_2 = 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} = 11011.101B$$

显然, 这里 $b = 2$, 权 b^i 为 16, 8, 4, 2, 1, $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$; $m = 3$, 且 $n = 5$, $a_i = 1, 1, 0, 1, 1, 1, 0, 1, B$ 表示二进制数.

二进制数的运算法则如表 1-1 所示.

表 1-1

101	110	110	101
+ 111	- 011	× 011	011) 1111
1100	011	110	11
		110	011
		10010	011
			0

二进制数转化为十进制是按权展开的, 例如:

$$N_2 = 11.101B = 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} = 3.625$$

十进制数转化为二进制是用基数去除, 余数按反序排列, 例如:

$\frac{9}{2} = 4$
余数 1

$\frac{4}{2} = 2$
0

$\frac{2}{2} = 1$
0

$\frac{1}{2} = 0$
1

故 $9 = 1001B$

← 反序

2. 八进制

基数为 8, 按逢八进一的原则进位, 用 0, 1, 2, 3, 4, 5, 6, 7 共八个符号表示数, 也按权展开, 可转化成十进制. 例如:

$$N_8 = 15.76Q = 1 \times 8^1 + 5 \times 8^0 + 7 \times 8^{-1} + 6 \times 8^{-2} \\ = 13.969$$

Q 表示八进制数.

二进制数可按三位一组转换成八进制数, 例如: $N_2 =$
110 101 111 010 001B = 65721Q

反之, 八进制可转换成二进制.

3. 十六进制

基数为 16, 用 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F 共 16 个符号表示数, 按逢十六进一的原则进位. 例如:

$$N_{16} = 9CH = 9 \times 16^1 + 12 \times 16^0 = 163$$

是按权展开的十进制数, 式中 H 表示十六进制数. 二进制数若按四位一组, 便可构成十六进制数, 例如:

$$N_2 = \underline{1101} \ \underline{1111} \ \underline{1001} \ \underline{0100}B = DF94H$$

4. 二一十进制 (BCD 码)

用四位二进制数代替一位十进制, 便构成二一十进制数, 常用 (8421) BCD 码, 如表 1-2 所示.

1-2-2 码制

对带正负号的数, 在机器中运算时需规定格式, 即码制.

1. 原码

数的最高有效位为符号位, 0 表示正数, 1 表示负数. 正

表 1-2 二—十进制数码 (BCD 码)

十进制数	(8421) BCD 码	十进制数	(8421) BCD 码
0	0000	6	0110
1	0001	7	0111
2	0010	8	1000
3	0011	9	1001
4	0100	10	0001 0000
5	0101	11	0001 0001

数的原码为该数本身，负数的原码是符号位为 1，数本身不变。例如，0111 1111 是正数+127，10001000 是负数-8，8 位机中原码数值范围为+ 127——128。因此在 8 位机中作数值加法运算时要注意溢出，即超出带符号数的范围。若超出要作处理，以保证运算正确进行。

2. 反码

正数的反码与原码相同，如 + 58 的反码为 00111010，与原码相同；负数反码是符号位为 1，其余位取正数的反，如 -58 的反码为 11000101。0 的反码有两种形式，00000000 或 11111111，故在数据处理时需注意。

3. 补码

正数补码与原码相同，负数补码是将其反码的末位加 1，故 -58 的补码为 11000110。采用补码，0 只有一个00000000，可简化加减法运算，统一为 $[X + Y]_{补} = [X]_{补} + [Y]_{补}$ 。例如：

$[58]_{补} + [-10]_{补} = 00110000$ （进位自然丢失）

应用补码进行运算的硬件电路容易实现，因而硬件设计均采用它，图 1-4 所示为补码加法器。