

高等学校试用教材

单片微型计算机 及其应用

席先觉 李忠民 程琳琳



高等教育出版社

36811

XJ/1

高等学校试用教材

单片微型计算机及其应用

席先觉 李忠民 程琳琳

高等教育出版社

内 容 提 要

本书主要介绍MCS-48和MCS-51单片机的基本结构、工作原理及应用实例。内容包括：计算机简介，MCS-48和MCS-51的基本结构和原理、指令系统及汇编语言程序设计，单片机的外部扩展及接口电路，传感器及其应用，单片机的开发及应用实例。本书讲授细致清楚，内容丰富。书中给出了大量的应用实例，读者可直接参考引用。每章均配有习题。

本书是为高等学校非计算机专业编写的教材，也可作为计算机专业的教材或参考书，对从事自动化控制方面的科技人员也有很好的参考价值。

高等学校试用教材

单片机原理及其应用

席先觉 李忠民 程琳琳

高等教育出版社出版

新华书店总店北京科技发行所发行

四川省金堂新华印刷厂印装

*

开本 787×1092 1/16 印张18.5 字数 460 000

1989年4月第1版 1991年4月第3次印刷

印数 7 451—11 176

ISBN 7-04-002084-X/TP·47

定价 4.85 元

前 言

随着大规模集成电路技术的发展,出现了各种性能优良的单片微型计算机(简称单片机),就其结构而言,单片机本身就是一台计算机。单片机具有功能强、体积小、节省能源、价格低廉以及工作性能可靠、使用方便等优点。这些优点比普通的微型计算机、单板机更为突出。目前它已广泛地应用于工农业生产的自动控制、智能化仪表、数据采集等各个方面,在国内已成为计算机控制应用的重点发展方向之一。

本书是作为高等院校非计算机专业学生的教科书编写的,也可作为计算机专业的学生及从事微机应用的广大科技人员的参考书。编者希望本书能有助于读者利用单片机为本门学科服务,因此在本书的编写中坚持由浅入深的原则,以便于自学。同时注意理论联系实际,尤其注重实际应用,书中例举了较多的实际电路和实用程序供读者参考引用。

目前单片机的品种很多,我们选用近几年世界上产量最大(86年占50%以上)、国内应用最广的MCS-48及MCS-51系列8位单片机作为本书的典型机种。鉴于MCS-51系列是在MCS-48系列的基础上发展而成的高档单片机系列,其基本结构与MCS-48系列有很多相同之处,故在本书中先详细介绍MCS-48系列单片机,然后再简要介绍MCS-51系列单片机。

本书的第一、二章介绍有关计算机的基本原理和概念。第三章至第六章介绍MCS-48系列单片机的结构原理、指令系统、编程方法和系统扩展方法。在此基础上,再在第十章介绍MCS-51系列单片机。使读者有能力根据需要,选择合适的单片机进行工作。第七章详细介绍与单片机应用密切相关的各种传感器。第八章例举了各种应用实例供读者参考和借鉴。单片机应用系统的研制必须要有开发工具的支持,故第九章详细介绍单片机开发装置的原理和单片机的开发应用方法。

本书由兰州大学计算机系席先觉同志主编,并参加编写了第一、二、三、十章。第四、五章由甘肃工业大学程琳琳同志编写。第六章至第九章由兰州铁道学院李忠民同志编写。

天津大学的许镇宇、何莉,北京工业大学的李大友,西安交通大学的刘甘娜及南京化工学院的陈季琪等同志参加了本书的初审,并由刘甘娜同志进行了复审。他们提出了许多宝贵的意见和建议,在此表示衷心的感谢。

由于编者的水平有限,时间仓促,缺点和错误在所难免,敬请读者批评指正。

编 者

1987年11月于兰州

绪 言

单片微型计算机是将微处理器、存储器、输入/输出接口等功能部件集成在一块大规模电路内而构成的微型计算机,简称单片机,亦称微控制器。

1975年,美国TEXAS仪器公司发表了TMS1000系列4位单片机,宣告了单片机时代的到来。单片机作为微型计算机领域的一个分支,以它体积小、功能强、耗电低、使用方便灵活、价格低廉等特点,占领了很多应用领域。同时也促使单片机本身获得飞速发展。1976年,美国INTEL公司即推出了8位单片机,并陆续生产了第一个8位单片机系列MCS-48,1980年又推出了8位高档单片机系列MCS-51。在INTEL公司的带动下,很多生产厂家都争相推出各自的8位单片机,如著名的Zilog公司推出了Z8系列单片机, Motorola公司推出了MC6801, MC6805系列单片机, Rockwell公司推出了6500/1系列单片机,日本的NEC公司推出了 μ COM-83和 μ COM-87系列单片机。

1982年开始推出的16位单片机,虽然性能优越,但由于价格昂贵,开发工具跟不上等原因,尚未得到用户的重视。目前主要的产品有Intel公司的MCS-96系列, NS公司的HPC16040及NEC的783XX以及Thomson公司的68HC200等16位单片机。

此外,近年来也出现了一些32位单片机,如TEXAS仪器公司的TMC320系列单片机,它专供数字信号处理之用。

目前,国际上出现的单片机品种繁多。据国外统计,到1985年10月止已有50个系列,373个机种。从产量来看,8位单片机的产量最大,1985年为1.7亿片,1986年为2.1亿片,占整个单片机的60%以上。Intel公司的MCS-48及MCS-51系列的产量占整个8位单片机的50%以上。而16位单片机今年的产量约为20万片。但它的价格正在逐步下降,预计它的应用面将会逐渐扩大。

正是由于上述原因,各生产厂家都把开发重点放在8位机上,改进原有机种的性能,并增加新的功能,以吸引用户。

新的单片机主要从下列几个方面采取改进措施:

一、片内存储器的改进

早期的单片机内读写存储器RAM的容量一般为64~128个字节,只读存储器ROM的容量为1k~2k字节,扩展寻址范围为4k字节。新的单片机内RAM为256字节左右,ROM可达12k字节,如TEXAS仪器公司的70120。而日立公司的6301Y,其ROM高达16k字节。扩展寻址范围增大到128k字节(RAM和ROM各64k字节)。

早期单片机内可编程ROM都是使用由高压编程和紫外线擦抹的EPROM。新的单片机内开始改用低压编程和用电擦抹的EEPROM,如Motorola公司的68HC805C₁(4k字节的EEPROM)、68HC11A₂(2k字节的EEPROM),TEXAS仪器公司的77C82(8k字节的EEPROM)。Intel公司也准备在UPI-452和8096中采用EEPROM。由于EEPROM读写方便,掉电后片内的信息不会被破坏,因此,有的生产厂家(如National Semiconductor公司)采用EEPROM作为片内RAM。

• 1 •

为了给用户编程提供保密,使EPROM编程不被复制,一些生产厂家对片内的 EPROM 或EEPROM采用加锁技术,如Intel 公司的 $8 \times C252$ 和Seeq 公司的72710.加锁以后就无法从片外读取.若要开锁,就得先擦除EPROM或EEPROM 内的全部信息,才能恢复未加锁时的功能.

二、片内输入输出功能的加强

早期的单片机,其片内接口只有并行输入输出接口,定时器/计数器接口能力较差.实用中常需扩展其他接口电路.新的单片机内增加了各种功能的接口: A/D转换器、D/A转换器、直接存储器存取(DMA)控制器、字符发生器、波特率发生器、比较器、液晶显示管(LCD)驱动器、萤光显示管(VFD)驱动器、发光二极管(LED)驱动器、CRT控制器、声音发生器、正弦波发生器、脉宽调制器(PWM)、锁相环、频率合成器以及各种特殊的串行通讯接口等.一个单片机内选用其中几种接口,供用户使用时选择.

三、软件固化

单片机的软件发展很快,并且固化到单片机内的ROM.如RCA公司的 68HCO5D2固化了键盘管理程序, Rockwell公司的R65F11及R65F12固化了FORTH语言, Zilog 公司的 Z8671及National Semiconductor公司的8073都固化了Tiny BASIC语言,有的公司甚至还固化了PASCAL语言.

四、制造工艺上的改进

早期的单片机采用PMOS工艺,其后逐渐采用NMOS, HMOS及CMOS 工艺,以提高单片机的性能(包括提高集成度;增加功能部件;提高时钟频率加快工作速度;降低功耗等).

随着 8 位单片机性能的提高,片内接口电路品种的增多,用户使用便愈加方便,这将会进一步推动 8 位机的推广应用.目前 8 位单片机的发展和应用正处在兴旺上升时期,在今后若干年内,单片机的应用仍将以 8 位机为主,8 位机的产量还会逐年增加,预计到1990年全球 8 位机的年产量将达 7 亿片.

我国对单片机的研制和应用起步较晚,但由于单片机特别适合我国的国情,自从单片机引进我国以后,即受到各方面的重视,并应用于国民经济的很多部门.发展的速度极快.目前国内单片机的应用也是以 8 位机为主,使用的机型主要有Intel 公司的MCS-48 和MCS-51 系列单片机,少数单位也使用Motorola公司的6801, 6805单片机和Zilog 公司的Z8系列单片机.国产的单片机也已有商品上市,这将会进一步促进我国单片机的发展.

目 录

绪言	1
第一章 计算机基础	1
1.1 计算机的基本结构	1
1.2 计算机的工作过程	4
1.3 中断	6
1.3.1 中断的作用	7
1.3.2 中断过程	7
1.4 堆栈	9
习题与思考题	11
第二章 运算基础	12
2.1 进位计数制	12
2.1.1 一般进位计数制	12
2.1.2 二进制数	12
2.1.3 八进制数和十六进制数	13
2.1.4 二进制数与八进制数及十六进制数之间的关系	14
2.2 二进制数与十进制数之间相互转换	15
2.2.1 十进制数转换成二进制数	15
2.2.2 二进制数转换成十进制数	17
2.2.3 八进制数、十六进制数与十进制数之间的转换	17
2.3 机器数	19
2.3.1 数值的表示方法——原码、反码、补码	19
2.3.2 小数点的表示方法——定点、浮点	21
2.4 编码问题	22
2.4.1 二进制数(BCD)	22
2.4.2 字母数字代码	22
2.5 运算方法	22
2.5.1 加、减法	23
2.5.2 乘、除法	26
习题与思考题	39
第三章 MCS-48单片机的结构和 工作原理	30
3.1 运算器	30
3.1.1 算术逻辑单元(ALU)	30
3.1.2 累加器、暂存器、标志寄存器	30
3.1.3 二十进制调整单元	32
3.2 存储器	34

3.2.1 程序存储器(ROM)	34
3.2.2 数据存储器(RAM)	35
3.3 控制部分	36
3.3.1 程序计数器(PC)	36
3.3.2 程序状态字寄存器及堆栈	36
3.3.3 指令寄存器及译码器	37
3.3.4 控制及定时部件	37
3.4 中断逻辑	40
3.5 输入/输出(I/O)端口	43
3.5.1 端口1(P ₁)和端口2(P ₂)	43
3.5.2 总线(BUS)	44
3.6 定时器/计数器	45
3.7 单片机的工作方式	46
3.7.1 复位工作方式	46
3.7.2 程序执行方式	47
3.7.3 单步方式	47
3.7.4 掉电保护方式	48
3.7.5 休眠方式(闲置方式)	49
3.8 引脚功能说明	50
习题与思考题	52

第四章 MCS-48指令系统	53
4.1 寻址方式	53
4.2 指令系统概述	56
4.3 数据传送及交换类指令	58
4.4 算术与逻辑运算类指令	62
4.5 累加器循环移位类指令	66
4.6 输入/输出类指令	67
4.7 程序转移类指令	70
4.8 定时器/计数器类指令	74
4.9 控制类指令	75
4.10 标志操作类指令	77
4.11 扩充的指令	79
习题与思考题	81

第五章 汇编语言程序设计	83
5.1 汇编语言的语句结构	83
5.2 简单程序设计	84
5.3 分支程序设计	86
5.4 循环程序设计	87

5.5 子程序设计	92	第八章 单片机应用实例	152
5.6 查表程序设计	95	8.1 多通道D/A转换与控制	152
习题与思考题	97	8.1.1 基本原理	152
第六章 外部扩展及接口电路	99	8.1.2 实用电路	152
6.1 概述	99	8.1.3 软件构成	153
6.2 存储器的扩展	99	8.2 仪器-计算机接口	155
6.2.1 程序存储器的扩展	99	8.2.1 接口工作原理	155
6.2.2 数据存储器的扩展	103	8.2.2 接口硬件电路	156
6.3 输入/输出接口扩展	106	8.2.3 接口软件	157
6.3.1 专用I/O扩展器件	107	8.3 智能声级计	159
6.3.2 用标准外设扩展I/O	109	8.3.1 硬件构成	160
6.3.3 标准外设扩展举例	112	8.3.2 软件设计	161
6.3.4 使用组合器件扩展方式	121	8.4 步进电机实时控制	164
6.3.5 使用标准TTL器件扩展I/O	123	8.4.1 步进电机控制原理	164
6.4 数/模与模/数转换接口	125	8.4.2 步进电机控制软件	169
6.4.1 单片机与D/A的接口	125	8.5 育秧控制	171
6.4.2 单片机与A/D的接口	130	8.5.1 控制原理	171
6.5 系统扩展小结	134	8.5.2 控制电路	171
习题与思考题	134	8.5.3 控制软件	174
第七章 传感器及其应用	136	8.6 程控调度电话汇接分机	176
7.1 传感技术概述	136	8.6.1 汇接分机专用机硬件电路	177
7.1.1 传感技术的发展	136	8.6.2 汇接分机软件构成	178
7.1.2 固态传感器的特点与结构类型	138	8.6.3 软件调制与解调的实现	179
7.2 力敏传感器及其应用	138	8.6.4 CMOS单片机的休眠控制	180
7.2.1 半导体压力传感器	138	习题与思考题	182
7.2.2 压电式力敏传感器	140	第九章 单片机的开发	184
7.3 热敏传感器及其应用	141	9.1 开发装置的结构与工作原理	185
7.3.1 热敏传感器的常用类型	141	9.1.1 系统组成与功能	185
7.3.2 半导体热敏电阻及其应用	142	9.1.2 系统的联接方式	186
7.3.3 晶体管温度传感器与可控硅热敏开关	142	9.1.3 接口板TPZCPU	187
7.4 光敏传感器及其应用	143	9.1.4 开发板TP80MCS	189
7.4.1 类型与应用特点	143	9.1.5 用户样机试验板TPMCSU	196
7.4.2 光电导型传感器	144	9.2 开发装置的使用方法	199
7.4.3 光电势型传感器	145	9.2.1 开发装置软件TPMCSA	199
7.5 磁敏器件及其应用	146	9.2.2 键盘与显示器	200
7.5.1 半导体霍尔器件	147	9.2.3 TPBUG监控程序的支援功能	201
7.5.2 其他磁敏器件	148	9.3 运用开发装置研制单片机应用系统的方法	202
7.6 气敏、湿敏传感器及其应用	149	9.3.1 总体设计	202
7.6.1 气敏传感器及其应用	149	9.3.2 硬件设计	203
7.6.2 湿敏传感器及其应用	150	9.3.3 软件设计	203
习题与思考题	151	9.3.4 设计举例	204
		9.3.5 源程序编译成目标程序	205

9.3.6 硬件调试	207	10.3 定时器/计数器	229
9.3.7 软件调试	208	10.4 串行口	233
习题与思考题	208	10.5 中断	238
第十章 MCS-51单片机	210	10.6 指令系统	241
10.1 8051的结构	210	10.7 程序设计举例	255
10.1.1 8051CPU	210	习题与思考题	263
10.1.2 片内存储器	214	附录1 MCS-48指令系统表	264
10.1.3 输入/输出端口	216	附录2 MCS-48指令矩阵表	270
10.1.4 复位与掉电方式	218	附录3 MCS-51指令系统表	272
10.2 存储器	218	附录4 MCS-51指令矩阵表	274
10.2.1 存储器组织	218	附录5 TP801MCS单片微型机开发	
10.2.2 寻址方式	220	装置键盘命令	277
10.2.3 存储器扩展	228		

第一章 计算机基础

在具体介绍单片机之前，先介绍一下电子计算机的基本结构和工作过程，使读者首先对计算机有一个初步的总体概念，有利于单片机的学习。

1.1 计算机的基本结构

现代的电子计算机种类繁多，按其规模可分为巨型机、大型机、中型机、小型机、微型机、单板机及单片机等。他们在结构、性能及配置等各个方面都有很大的差别。但是从它们组成结构的总体来看，这些千差万别的计算机都有共同之处。任何一台计算机，都由五大部分组成，即：运算器、控制器、存储器，输入部分和输出部分。通常把运算器和控制器两部分合称为中央处理器，简称CPU。

图 1-1 是计算机的结构框图，图中表示了这五部分相互间的联系。其中双线表示数据信息传输线，单线表示控制线，箭头表示信息的传递方向。

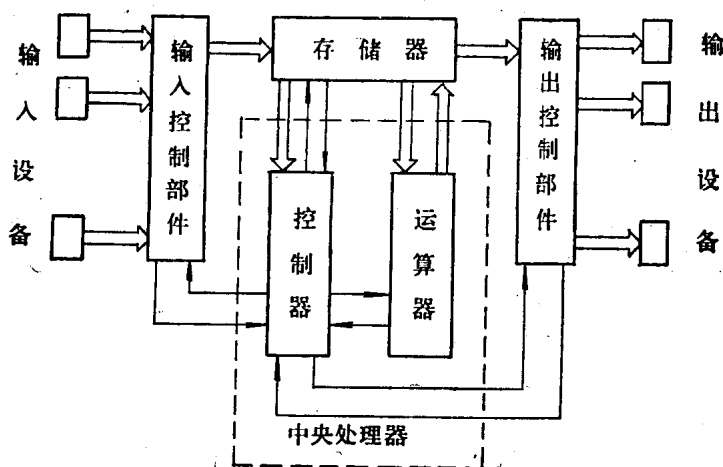


图 1-1 计算机结构框图

为了能了解计算机的大致工作过程，现对各个部分分别作一简要说明。

一、运算器 运算器的主要功能是作算术运算和逻辑运算。算术运算主要有加、减、乘、除四则运算，逻辑运算主要有与、或、非、异或等运算。对于大多数微型机及单片机来说，算术运算中只有加、减法运算，没有乘、除法运算，乘、除法需靠软件来实现。高档的微型机及单片机具有乘法和除法运算功能，如MCS-96系列的单片机，能进行16位乘16位的乘法和32位除16位的除法运算。MCS-51系列单片机有8位乘、除法运算功能。

运算器主要由一个算术逻辑单元（ALU）、两个以上的寄存器及一些控制门组成。图 1-2 是一个只有两个寄存器及一个算术逻辑单元组成的运算器简图。两个寄存器用来存放即将参加运算的数据。运算时，由控制器发来的控制信号，打开门1及门2，使寄存器A及B中的

数据进入算术逻辑单元 (ALU) 进行算术运算或逻辑运算。运算结果经门3存入寄存器A。由于运算结果总是存放到寄存器 A, 故又把寄存器 A 称为累加器。这种运算器的结构称为累加器结构, 单片机大都采用累加器结构。

在实际计算机中, 寄存器都不止两个, 而是有着一组寄存器, 每一个寄存器都可以直接参加运算, 这些寄存器称为通用寄存器。

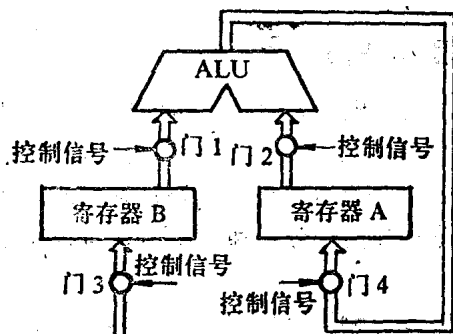


图 1-2 运算器简图

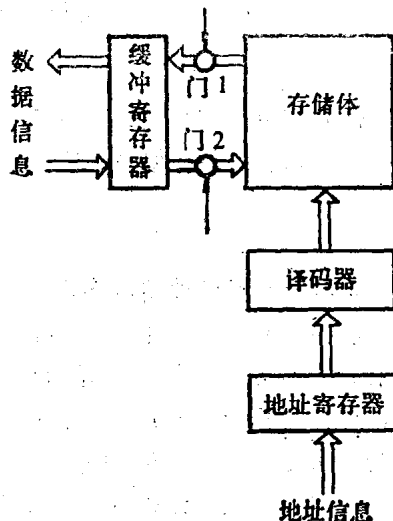


图 1-3 存储器简图

二、存储器 存储器的主要功能是存放程序、原始数据、中间结果和最终结果等, 图1-3是存储器的简图。其中最主要的部分是存储体。它由很多记忆元件组成, 用以记忆信息, 每个记忆元件存储一个二进制位的信息。通常以字节作为存储单元, 一个字节有八个二进制位。每个存储单元都有一个编号, 称为地址。当我们要存放信息 (程序或数据) 时, 先把该信息送到缓冲寄存器, 并把要存放信息的存储单元的地址号送到地址寄存器, 经译码器译码后, 在存储体内找到该地址单元, 然后由控制器发来的“写”控制信号把门1打开, 把缓冲寄存器内的信息送到地址寄存器所指示的存储单元。此过程称为“写入”。当我们要从存储体的某个地址单元中取出一个信息时, 先把该地址号送入地址寄存器, 经译码器译码后找到该地址单元, 然后由控制部件发来“读”控制信号, 打开门2, 把该地址的信息送到缓冲寄存器。此过程称为“读出”。

早期的计算机中主要采用磁心作存储器件。随着大规模集成电路技术的发展, 出现了半导体存储器, 它的集成度很高, 一片半导体芯片上能集成百万个以上的记忆元件。故目前的计算机基本上都采用半导体存储器。半导体存储器的种类比较多。最常用的有随机存取存储器 (简称RAM) 和只读存储器 (简称ROM)。随机存取存储器允许用户对它读取信息, 也允许用户对它写入信息。只读存储器是只允许用户读取信息, 不允许用户写入信息。如一些专用的程序, 不希望用户改动它, 就可以放到ROM中去。ROM又分为掩模ROM、可编程ROM(PROM) 及可擦除的ROM(EPROM)。掩模ROM是制造厂家在生产时将信息固化在片内, 制成后就只能读不能改写。PROM可由用户对它进行一次性编程脱机写入, 此后就不能再写, 只能读。对 EPROM 用户可根据需要将其内容擦除后重新再写, 擦除的方法应根据产品的种类, 有的用紫外线擦除, 有的用电擦除 (EEPROM)。正常工作时只能读, 不能写, 重写时需要额外的条件, 先擦后写。

三、控制器 电子计算机是根据指令来进行操作的, 所谓指令即是指挥计算机进行操作的命令。因此, 它必须告诉计算机进行什么性质的操作 (操作码) 及参加运算的数据 (操作

数) 放在存储器的哪个单元。所以指令通常由两部分组成: 一是操作码部分, 表示该指令的操作性质; 另一个是操作数地址部分, 指出操作数所在的地址。控制器的作用是把指令取来, 并对指令的性质进行分析(指令译码), 如需要, 再把操作数取来, 然后根据操作性质, 按时间先后发出各种控制信号, 使计算机的各个部分有条不紊地执行该指令所要求的一系列操作。当一条指令执行完后, 又自动地取下一条指令来执行, 直至全部指令执行完。

图 1-4 是控制器简图。图中程序计数器(也称为指令计数器)用以存放即将执行的指令

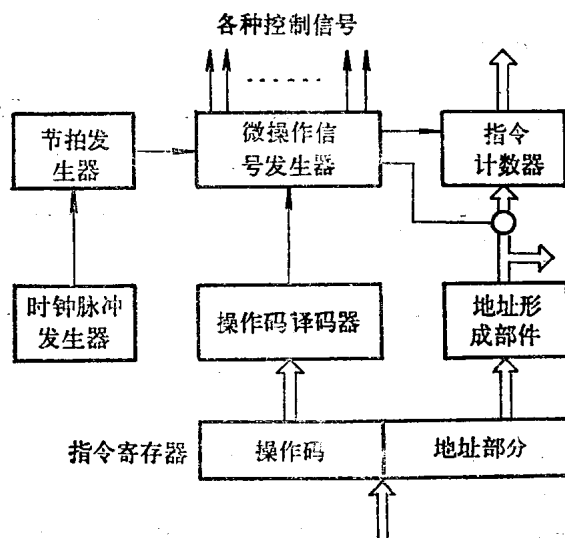


图 1-4 控制器简图

地址。当要取一条指令时, 就把该地址码送到存储器, 从存储器中取出的指令放到图中的指令寄存器中。控制器一旦取得指令, 就用操作码译码器对指令的操作码进行译码, 同时控制程序计数器进行加 1 操作, 使它指向下一条要执行的指令地址。这是因为构成程序的指令序列在存储器中是按顺序存放的。操作码译码后, 判定操作性质, 把译码器输出信号送微操作信号发生器, 由微操作信号发生器发出该指令所需的全部控制信号。各控制信号发出的先后顺序由节拍发生器控制。

根据操作码的性质, 如果需要操作数, 就由地址形成部件形成操作数存放的地址, 送存储器, 由存储器读出操作数, 送运算器参加运算。有时指令不按顺序执行而要跳到另一地址去执行该地址的指令, 即发生所谓转移。此时由地址形成部件产生转移地址, 送程序计数器, 这样, 计算机就执行新地址的指令。

图中的时钟脉冲发生器产生某一频率的矩形波, 称为时钟脉冲, 供给节拍发生器。他们的波形如图 1-5。这里假定一条指令需要 5 个节拍才能执行完, 故有 5 个节拍脉冲 $W_0 \sim W_4$ 。每个节拍内需完成什么操作, 就应在该节拍的脉冲作用期间发出相应的微操作控制信号。由此可见, 图 1-5 中的这些信号是用来协调整机工作的。计算机内的部件很多, 工作速度又很快, 因此工作的时序要求严格。时间上的不协调会使计算机工作不正常, 尤其在时钟频率很高的情况更是如此, 因此要求时钟频率也很稳定。目前一般计算机的时钟发生器都采用晶体振荡器, 以保证得到很高的频率稳定度。

四、输入/输出设备 使用计算机时, 必须先把编制好的程序和原始数据送入存储器,

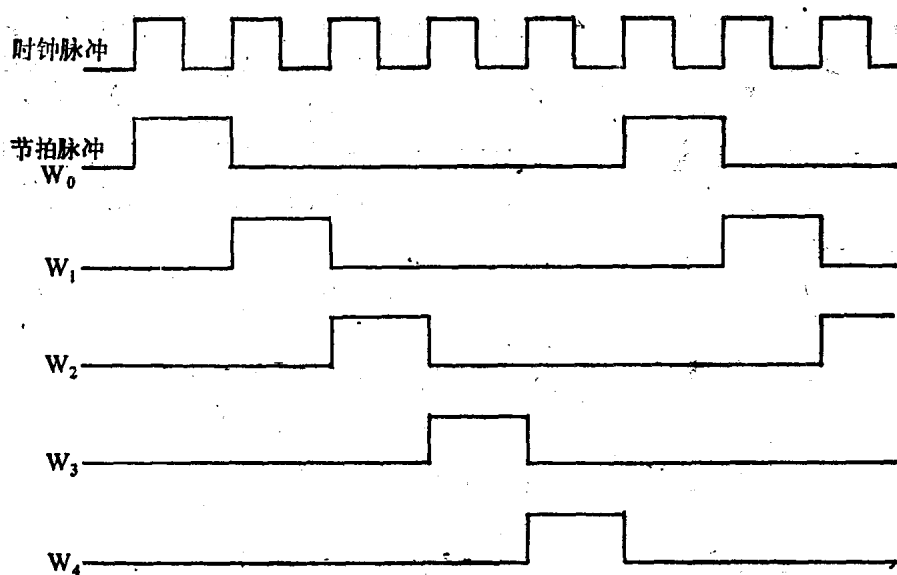


图 1-5 时钟脉冲和节拍脉冲波形图

并要求计算机把运算的结果按我们所要求的方式输出。这些任务是由输入设备和输出设备来完成的。

常用的输入设备有键盘、磁盘机、磁带机、光笔以及模数转换器等。常用的输出设备有显示终端、打印机、磁盘机、磁带机、绘图仪以及数模转换器等。由上可见，有些设备既是输入设备又是输出设备，如磁盘机、磁带机等。此外，象键盘和显示终端也常常放在一起，因此输入部分和输出部分常合在一起讨论。为了使用方便，一台计算机常配备多台输入输出设备，而且有时需要多台设备同时工作。为此就需要有专门管理输入输出设备工作的控制部件，用于管理这些设备的工作启停，以及与CPU及存储器的信息交换，实现人一机通信等。在微型机和单片机中与输入输出设备相连接的电路称为接口电路。

我们把图1-2至图1-4填入图1-1中，可得到一个计算机的整体框图，如图1-6所示。它有两簇公用线把各部分连接起来，一簇线用来传送地址码，称为地址总线，另一簇线用来传送数据信息，称为数据总线。图1-6粗略地给出了信息流通的大致情况。

1.2 计算机的工作过程

计算机的工作过程主要是指执行程序的过程。计算机运行时，先从存储器内取出事先存放好的一条指令，然后执行该指令，执行完后再取下一条指令，再执行此指令，如此循环，直至执行完最后一条指令。下面我们对图1-6，通过具体例题来说明计算机的工作过程，以了解计算机的各个部分是如何协调工作的。

例 计算 $y = a \times b + c \times d$ y 为计算结果， a, b, c, d 是原始数据。

解题步骤为：首先根据题意，确定算法，并分配内存单元；然后编制计算程序，把程序和原始数据都输入存储器，再让计算机去执行。

我们把从地址K开始的存储单元用于存放原始数据、中间结果及最终结果，具体分配如

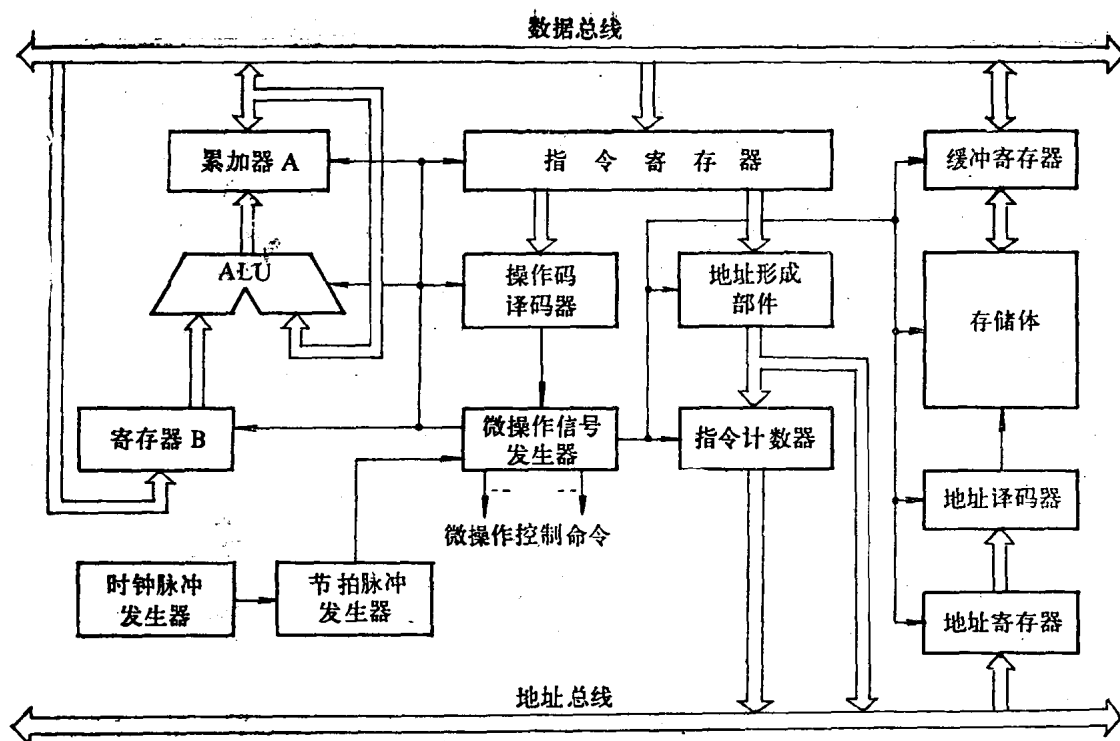


图 1-6 计算机整机框图

下:

地址	存放内容
K	a
K+1	b
K+2	c
K+3	d
K+4	$a \times b$ (中间结果)
K+5	y (最终结果)

编制好的计算程序放置在从地址 m 开始的存储单元, 具体分配如下:

地址	指令内容	指令内容说明
	操作码 地址	
m	取数 A, K	从存储单元 K 中取操作数 a 送入运算器中的累加器 A , 即 $(K) \rightarrow A$
$m+1$	取数 $B, K+1$	从存储单元 $K+1$ 中取操作数 b 送入运算器中的寄存器 B , 即 $(K+1) \rightarrow B$
$m+2$	相乘 A, B	A, B 两寄存器的内容相乘, 其积放在累加器 A , 即 $A \times B \rightarrow A$
$m+3$	送数 $K+4, A$	把累加器 A 的内容送存储器的 $K+4$

m+4	取数	A, K+2
m+5	取数	B, K+3
m+6	相乘	A, B
m+7	取数	B, K+4
m+8	相加	A, B
m+9	送数	K+5, A
m+10	停机	

单元, 即 $A \rightarrow K+4$

从存储单元 $K+2$ 中取操作数 c 送入累加器 A , 即 $(K+2) \rightarrow A$

从存储单元 $K+3$ 中取操作数 d 送入寄存器 B , 即 $(K+3) \rightarrow B$

A, B 两寄存器的内容相乘, 其积存放在累加器 A , 即 $A \times B \rightarrow A$

从存储单元 $K+4$ 中取操作数 $(a \times b)$ 的积) 送寄存器 B , 即 $(K+4) \rightarrow B$

A, B 两寄存器的内容相加, 其和存放在累加器 A , 即 $A+B \rightarrow A$

把累加器 A 的内容送存储单元 $K+5$, 即 $A \rightarrow K+5$

运行结束

其中地址号外面加括号 (如 $(K+3)$) 表示为该存储单元的内容, 即存放在该单元的数据。为了便于说明解题的过程, 这里指令分得比较细, 实际计算机指令的功能比较强, 可以把上述几条指令合成一条去执行。

通过输入设备, 把原始数据和程序输入到存储器中指定的单元, 这时准备工作已经完成, 现在可以由控制台下达运行命令。当计算机接收到运行命令后, 立即把程序计数器置为第一条指令所在的地址值 m , 接着就开始取指令的过程: 一开始先把程序计数器的内容 m 传送到地址总线, 再送入存储器的地址寄存器, 经地址译码器译码后找到存储单元 m , 在“读”命令控制下, 把 m 单元的内容送到缓冲寄存器, 再由缓冲寄存器经数据总线送入指令寄存器, 然后由操作码译码器进行译码, 判别操作性质。同时, 程序计数器的内容加1, 为取下一条指令作好准备。上述动作都是由控制器按时间顺序先后发出微操作控制命令来完成的。这个过程对任何指令都是一样的, 所以叫做公共操作。操作性质判定以后就要执行该指令。第一条指令要求完成的操作是 $(K) \rightarrow A$, 即要求从存储单元 K 取操作数送累加器 A 。于是由地址形成部件形成地址值 K , 经地址总线, 送入存储器的地址寄存器, 译码后找到 K 单元, 在“读”命令控制下, 把 K 单元的内容 a 送到缓冲寄存器, 再经数据总线送到运算器中的累加器 A , 此时累加器 A 的内容即为 a 。至此, 第一条指令所要求的全部操作已执行完毕。

上述指令一旦执行完, 控制器就又重复取指过程, 把 $m+1$ 单元中的指令取来, 并执行之。然后, 再取下一条指令, 如此循环, 直至执行到第 $m+10$ 单元的停机指令, 使计算机运行结束。每次取指令的过程都是相同的, 执行指令的过程却因指令的性质不同而有所差异, 它由控制器发出各种不同的微操作控制命令来完成各种指令所要求的全部操作。

由此可见, 计算机的操作过程是取指令并执行指令的循环过程, 这些都是在控制器的控制下完成的。由控制器发出各种微操作控制命令, 首先把事先存放在存储器中的指令取出, 然后再控制运算器完成该指令所要求的运算操作。

1.3 中 断

当输入/输出设备要与CPU进行信息交换时, 常用中断的方法来实现。所谓中断是指这

样的过程：当CPU正在执行某个程序时，由于某种原因促使CPU停止现行政程序的执行，转去执行一个特定的程序，作某种必要的处理，处理完毕后，再返回来继续执行原来的程序。

1.3.1 中断的作用

中断是用来提高计算机效率的一种有效手段。一般处理器的运算速度相当快（约微秒数量级），而外部设备的操作速度很慢（约毫秒数量级以上），二者存在着速度不匹配的矛盾，致使整机工作效率低。采用中断技术，可使这个矛盾得到很好解决。当外设完成其辅助操作需要与CPU进行信息交换时，它就向CPU发出一个信号，请求CPU进行信息交换，这个信号称为外设的中断请求。CPU接到中断请求信号后，就暂时停止执行现行政程序，转去执行该外设所要求的中断处理程序，完成后就返回中断点，CPU继续执行原程序。而外设在与CPU交换信息后，又按自己的规律进行操作，操作完毕后，再向CPU发出中断请求。这样，实现了CPU与外部设备的并行工作，从而解决了二者速度上的矛盾，并且大大提高了CPU的工作效率。

中断技术也能实现计算机的实时控制。如实时控制现场的参数和控制信息需要CPU处理，可随时向CPU发出中断请求，CPU就立即响应并执行处理程序，完成所要求的操作后，返回原程序。

中断技术也广泛用于处理紧急事件。如电源掉电等，计算机可以利用中断系统自行处理。

通常，我们把引起中断的原因称为中断源，象外部设备、实时控制设备、故障源（如电源掉电）等都称为中断源。一台计算机通常有很多中断源，若有几个中断源同时发出中断请求，则CPU在响应这些中断源的中断请求时，应有个先后次序，这个次序是根据各种中断源的优先级排定的，优先级高的优先得到响应。用户可按实际应用中的中断事件的紧急程度及重要性等条件来确定优先级。如电源故障最为紧急，需最先处理，安排的优先级最高。又如实时控制比外设紧急，则实时控制的优先级比外设高。一般微处理器都有几根接受中断请求的中断引线，每根中断引线的优先级由生产厂家规定，用户可把各中断源挂到相应优先级的中断线上。

CPU对各中断请求的处理原则是：

- 一、不同优先级的中断源同时申请中断时，按优先级别的高低依次处理。
- 二、当CPU正在处理某中断过程时，如发生优先级比它高的中断请求，则应立即暂停现行中断处理，而先去处理高级中断，等高级中断处理完毕后，再返回来处理原中断过程，这种情况称为多重中断或中断嵌套。
- 三、当CPU正在处理某中断过程时，如发生优先级比它低的中断请求，则应等到本中断过程处理完后响应优先级低的中断请求。
- 四、中断优先级相同的几个设备同时申请中断时，则按事先规定的次序逐个处理。

1.3.2 中断过程

为了阐明中断处理的基本过程，我们仅讨论最简单的中断情况。其基本过程一般分为中断请求、中断响应、中断处理及中断返回四个阶段。分述如下：

一、中断请求

当中断源需要CPU给予中断服务时，就向CPU的中断输入线发出中断请求信号。

二、中断响应

CPU在执行现行指令的最后时刻，检测中断输入线是否有中断源申请中断。当CPU检测到有中断请求信号，并且又处于允许中断状态，CPU就立即响应中断。

CPU在中断响应期间，一般能自动完成下列工作：

1. 关中断。即禁止再发生新的中断。CPU在响应中断后，需要进行一系列的处理工作，为了避免发生差错，在这些处理工作未完成之前，不允许发生新的中断。单片机8048，8051的中断系统较简单，不能自动关中断，可根据需要，用软件实现。

2. 保护断点。所谓断点是指发生中断时程序计数器PC的内容。即现行主程序中下一条指令地址。发生中断时，应把它保存起来，以便中断处理完毕后能返回到主程序。

3. 将该中断源的中断处理程序入口地址送入程序计数器，使CPU执行中断处理程序。

三、中断处理

中断处理程序一般包含四个部分：

1. 保护现场。也称为保护运行环境。所谓现场是指主程序中中断前它所占用的全部资源的状态。主要是一些通用寄存器和专用寄存器的内容。这些资源可能要被中断服务程序占用，因此需把它们的内容先保护起来，以供返回时恢复原来状态之用。

2. 中断服务程序。这是中断处理程序的主要部分。各中断源申请中断希望CPU执行的全部操作，都编制在中断服务程序中。由于各个中断源所要处理的问题各不相同，因此对应每个中断源都编制有特定的中断服务程序。

3. 恢复现场。把保护现场时保存的状态数据全部送回原资源中，使运行状态全部恢复到中断前主程序的运行状态。

4. 开中断。中断处理已经结束，应允许CPU响应新的中断。如允许中断嵌套，则应在保护现场之后就开中断。

四、中断返回

把中断响应时保存起来的断点地址送回程序计数器PC，使PCU继续执行主程序。有专门的中断返回指令来实现中断返回操作，因此中断处理程序的最后一条指令应是中断返回指令。

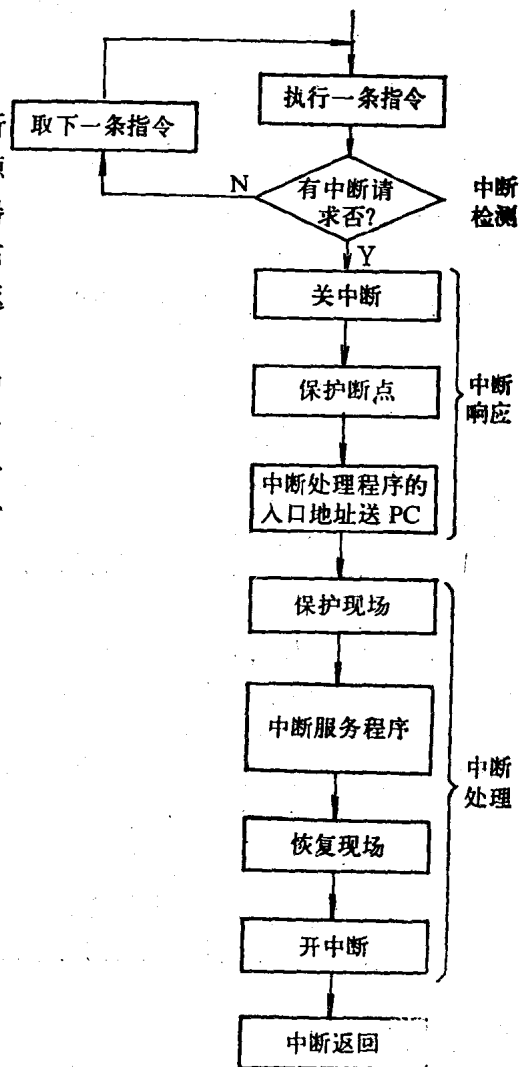


图 1-7 中断流程图

图 1-7 所示的流程图粗略地说明了整个中

甲