

吴月珠

何 菁 编

王自强

南京大学出版社 NAN JING DA XUE CHU BAN SHE

微处理器系统及其应用

WEICHULIQIXITONGJIQIYINGYONG



内 容 简 介

本书基础篇全面地介绍了微处理器的一般原理和体系结构,并以 Intel8086 为典型 CPU,重点剖析了微处理器的硬件结构、时序特性及指令系统。在此基础上,讲述了利用外围芯片进行 I/O 处理、中断处理以及最终构成微处理器系统的方法。在程序设计部分,介绍了宏汇编语言的语法及常用程序设计方法。作为一个典型实例,介绍了 IBM PC 系列微机的体系结构、操作系统等内容。最后,对 X86 系列高档微处理器作了简单介绍和展望。实验篇部分提供了大量模块化软件和硬件实验,可对读者进行全面的训练。

本书结构新颖,内容翔实,是一本较好的微处理器学习和教学参考书,可作为理工科电子类专业微处理器课程的教学用书,也可供广大应用专业人员阅读和参考。

JS229/26

微处理器系统及其应用

吴月珠 何菁 王自强 编

南京大学出版社出版

(南京大学校内 邮政编码: 210093)

南京豪利电脑照排中心照排

江苏省新华书店发行 常熟市文化照相制版彩印厂印刷

*

开本 787×1092 1/16 印张 21 插页 1 页 字数 538 千

1996 年 1 月第 1 版 1998 年 1 月第 2 次印刷

印数 1501~3500

ISBN 7-305-02889-4/TM·4

定价: 25.00 元

(南大版图书若有印、装错误可向承印厂退换)

前 言

随着微电子技术的发展,微处理器的应用已渗透到各行各业的各个层次中去,《微处理器系统及其应用》也相应成为理工科类学生必须掌握的一门专业基础课。然而,微处理器发展的基本特征,即品种多、发展快,却给微处理器的教学带来了莫大的困扰。从位片机到 32 位机,从简单控制器到专用数字信号处理器 (DSP),眼花缭乱,日新月异。如何立足基本,剖析典型,掌握一般直至触类旁通,正是本书力图解决的问题。

初学微处理器时,常常会被其复杂的硬件结构,尤其是繁琐的硬件时序搞得晕头转向,不知所措,而一个简单的、与直观感觉较为接近的模型却是比较容易理解和接受的。本书即从这样一个模型机入手,使读者初步建立微处理器体系结构的基本印象。进而从硬件和指令系统两个方面对这一体系结构进行充实,使读者确立微处理器的完整概念。在此基础上,逐步设计出从简单到复杂的硬件系统和软件程序,使前面学到的硬件结构和指令系统得到进一步的强化和应用练习,并逐步建立微处理器系统的概念。

本书以 Intel 8086CPU 为典型微处理器进行分析,这样选择的原因一方面是因为 Intel 8086CPU 有较为规范的体系结构,成为 X86 系列微处理器的基础,因而具有一定的典型代表性。另一方面,是因为其有丰富的硬件支撑和软件开发环境,便于编排教学实验。但这决不意味着对微处理器的了解仅局限于 8086 就够了,事实上,我们的教学目的希望在学完本课程后,学生具备借助微处理器产品手册和开发系统开发任何一种微处理器应用系统的能力。

微处理器的学习中,经常会出现茫然的感受,这是正常的,因为微处理器是一门理论与实践并重的课程,不经过实验验证、强化和应用,想要学懂、学好是不可能的。本书包含基础篇和实验篇,部分实验内容超出或超前于课堂教学内容,藉此检验学生对一般微处理器触类旁通的能力。可与实验内容配套的有 WCS-1 型和 WCS-2 型实验系统,分别可用于开设 X86 和单片机系列实验,也可作为通用微处理器开发系统应用。这些实验系统可与本书作者联系获得。

本书由吴月珠同志负责组织和统稿。吴月珠(第一、二、三、四、九章及实验篇第一部分)、何菁(第五、六章)、王自强(第七、八章)、金永红(实验篇第二部分)、邵钺(第十章)参加编写。

本书的编写过程始终得到了严志华教授的直接指导和关心,并提出了许多宝贵而有益的意见,谨致谢忱。

徐岚同志为本书插图的绘制付出了艰辛的劳动,吴珩同志及电子科学与工程系的历届同学为本书出版及实验设计做了许多工作,在此一并致谢。

编 者

1995 年 1 月于南京大学

目 录

基础篇

第一章 微处理器应用基础	3
1.1 微处理器发展概况	3
1.2 有关微处理器的一些基本概念	4
1.3 运算基础	6
1.4 微处理器的工作.....	16
1.5 微处理器的外特性.....	24
第二章 8086/8088 的体系结构	27
2.1 8086/8088 的结构分析	27
2.2 8086/8088 的工作模式和引脚	30
2.3 8086/8088 的工作时序	36
第三章 8086/8088 的寻址方式和指令系统	41
3.1 8086/8088 的寻址方式	41
3.2 8086/8088 的指令系统	44
第四章 存储器组织与一个最简单的微处理器系统	72
4.1 存储器概述.....	72
4.2 输入/输出接口概述	76
4.3 存储器及输入/输出接口的寻址	77
4.4 存储器组织.....	79
4.5 一个最简单的微处理器系统.....	84
第五章 总线控制与中断	91
5.1 模块间输入输出的传递方式.....	91
5.2 查询型输入输出方式.....	93
5.3 中断型输入输出方式.....	95
5.4 直接存储器存取传递方式	101

第六章 通用型接口芯片	102
6.1 概述	102
6.2 并行通讯与可编程并行接口芯片 8255	104
6.3 串行通讯与可编程通讯控制器 8251A	116
6.4 可编程定时器/计数器 8253	127
6.5 可编程中断控制器 8259A	134
6.6 DMA 控制器 8237	143
第七章 汇编语言与程序设计	152
7.1 概述	152
7.2 宏汇编语言的基本语法	157
7.3 数据段的设计与使用	161
7.4 程序段的设计与使用	170
7.5 宏指令与其它常用的指令	177
7.6 汇编语言程序设计	182
第八章 IBM PC 微型计算机系统简单介绍	200
8.1 IBM PC 微型计算机系统	200
8.2 MS-DOS 操作系统	203
8.3 BIOS 中断功能调用	205
8.4 DOS 中断功能调用	215
第九章 微处理器应用系统的设计和调试	220
9.1 微处理器应用系统设计的两种方法	220
9.2 微处理器应用系统设计的几个问题	220
9.3 利用 PC 机作为开发工具设计应用系统的实例	223
第十章 X86 系列微处理器简介	230
10.1 Intel 8086/8088 微处理器	230
10.2 Intel 80286 微处理器	230
10.3 Intel 80386 微处理器	236
10.4 i486 微处理器	242
10.5 Pentium 微处理器	243
附录一 8086/8088 指令系统一览表	246
附录二 DOS 功能调用表	260
附录三 单片微型机 MCS-51 系列 8031 简介	265

实验篇

第一部分 微处理器应用实验..... 285

 一、汇编程序设计实验 285

 二、常用接口芯片及单片机研究实验 288

 三、综合实验 292

第二部分 WCS-1 和 WCS-2 型实验系统简介 296

附录一 ADC0809 DAC0808 及 8279 的简介 308

附录二 动态调试程序 DEBUG 316

附录三 实验常用芯片引脚图..... 326

参考书目..... 327

基础篇

第一章 微处理器应用基础

内容提要

微处理器是计算机中的核心部件,本章主要介绍微处理器的发展概况、有关的基础知识、运算基础、微处理器的工作及它的外特性,使读者对微处理器有一个初步的认识。

1.1 微处理器发展概况

微处理器产生于 70 年代,在此之前,计算机的发展,已经经历了四代更迭。第一代从 1946 年到 1954 年,其特征是以电子管作为逻辑元件,体积大,自身重,耗电多。例如 1946 年美国 ENIAC (Entectronic Numerical Integrator And Computer) 机共 18000 只电子管,1500 只继电器,重 30 吨,耗电 140 千瓦以上,占地 150 米²。第二代从 1955 年到 1964 年,以晶体管作为逻辑元件,代表机型如 IBM 公司的 IBM7094、CDC 公司的 6600 及 DEC 公司的 PDP-8 等。第三代从 1965 年到 1975 年,以中小规模集成电路作逻辑元件。出现了小型机,计算机技术走向成熟,代表的机型如 IBM360 和 PDP-11 等。1975 年以后计算机的发展进入第四代,大规模、超大规模集成电路成功地应用于计算机领域,出现了每秒亿次以上基本操作的巨型机及以微处理器为核心的个人计算机。进入 80 年代以来,计算机已经发展到第五代。计算机的前四代更迭,主要体现在使用的元器件及有关技术的巨大进步上,但是总的设计思想仍然基于 1946 年冯·诺依曼 (Von Neumann) 正式提出的“存贮程序原理”,此理论提出计算机由 5 个基本部分组成:存储器、算术逻辑单元、程序控制单元、输入及输出单元。而到了 80 年代,人们开始从否定冯·诺依曼理论的思路考虑设计具有推理、联想等思维能力的智能计算机即所谓神经计算机,将计算机的发展推进到一个新的时代。

而以微处理为核心的微型计算机自 70 年代产生以来,也已经有了飞速的发展。至今亦可把微处理器的发展划分四代。第一代 1971 年到 1972 年,以 Intel 公司的 4004 与 4040 为代表,时钟频率 750kHz,数据线宽 4 位,主要用于计算器。第二代 1973 年到 1975 年,以 Intel 公司的 8080A/8085, Zilog 公司的 Z80 及 Motorola 公司的 6800 为代表,时钟频率 1MHz~4MHz,数据线宽 8 位,设置了堆栈,具有中断功能,已经是一种成熟的计算机的中央处理单元 (CPU)。这些微处理器迅速在全世界得到广泛的应用与普及。第三代 1976 年到 1979 年,以 Intel 公司的 8086/8088, Motorola 的 68000, Zilog 公司的 Z8000 为代表,时钟频率提高到了 8MHz~10MHz, 16 位数据线宽。作为微型计算机的 CPU,它吸取了中小型机乃至大型机的特点,例如具有乘除法指令,支持高级语言,支持协处理器,可以组成多处理器系统等,所以很快风靡全世界。进入 80 年代以后,性能更优异的 80186、80286 相继问世,时钟频率为 12MHz—16MHz,把 16 位微处理器的水平提高到了一个新阶段。同时第四代微处理器,数据

线宽 32 位 Intel 公司的 80386、Motorola 公司的 68020、68030、68040 等也已进入市场。80386 时钟频率提高到 20MHz~55MHz。到了 90 年代更先进的 80486 (Intel 公司) 及 Pentium 等也已问世。

以上 CUP 构成的计算机都是 CISC (Complex Instruction Set Computers) 型的, 即随着寻址方式越来越复杂, 指令也越来越多, 机器的功能也越来越强。到了 80 年代中, 出现了 RISC (Reduced Instruction Set Computers) 型的 CPU, 它的指导思想是减少指令, 简化使用方法, 目前, RISC 型的计算机已商品化。还出现既具有 CISC 的优点又具有 RISC 的优点的新型 CPU。

微处理器的发展过程中, 一方面发展趋势如上面所述, 性能越来越高, 集成电路规模越来越大的微处理器竞相出现。而另一方面, 通用微处理器芯片在组成微型计算机或微处理器应用系统时, 还需要存储器芯片, 输入输出接口芯片, 定时/计数器芯片等, 于是人们将微处理器与存储器、定时/计数器、输入/输出接口等都集成在一个芯片上, 这就是目前广为应用的单片机。以 Intel 公司的产品为例, 自 1976 年单片机问世以来, 已发展了 MCS-48, MCS-51 和 MCS-96 单片机系列。其中 MCS-51 系列为 8 位数据线宽的高性能单片机, MCS-96 系列则是数据线宽为 16 位的单片机系列。

另外, 还有各种专用微处理器芯片, 位片机芯片在 70 年代到 80 年代获得了广泛应用。

1.2 有关微处理器的一些基本概念

一般计算机是由硬件与软件两部分组成。硬件是指组成计算机的集成电路、印刷板、电源、存储器、打印机等硬设备, 而软件是由算法和程序所组成, 包括系统软件与应用软件, 它们的基础是计算机的指令系统。其中系统软件是指操作系统、数据库管理系统、语言处理程序 (编译、解释程序), 还有一些诊断查错的实用程序等管理计算机工作的软件, 而应用软件是指由用户根据具体任务采用汇编语言或高级语言所编制的程序, 例如数据采集与处理程序、事务管理程序及过程控制程序等。作为一个整体, 计算机的一部分功能是由硬件来完成的, 另外一些功能是由软件来完成的, 不同的目的采用硬件与软件的决策是不同的, 并没有严格的规则来规定哪些功能由硬件来完成, 哪些功能由软件来完成。

下面将从硬件的角度来介绍有关微处理器的一些基本概念。

一、微处理器、微机、微机系统间的联系

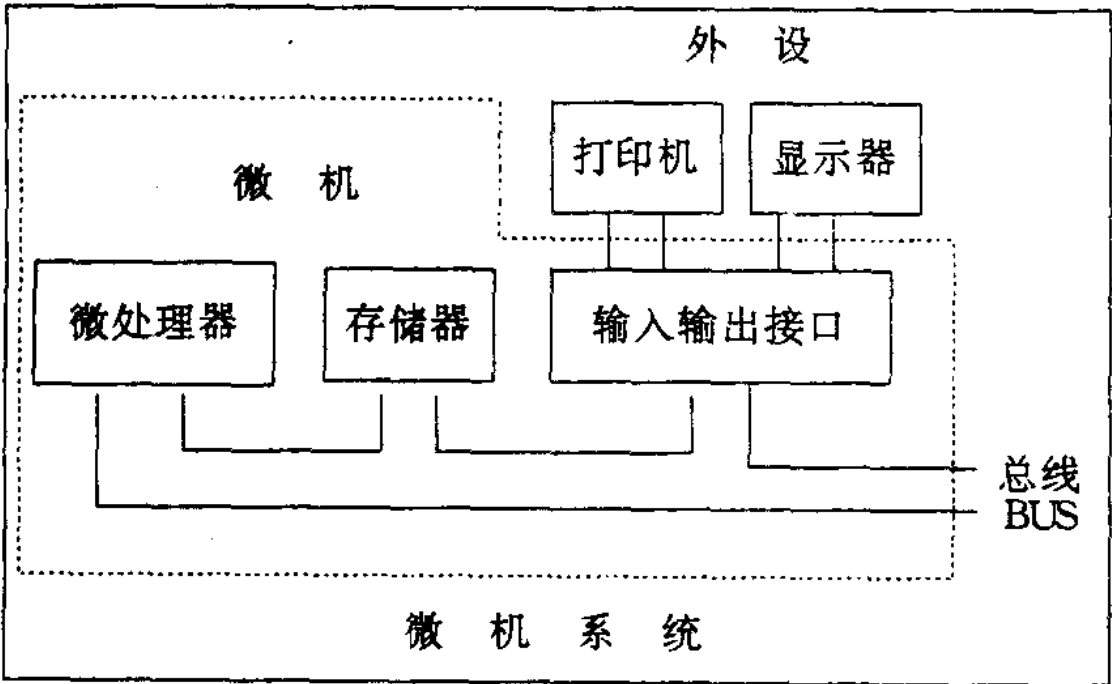


图 1.1.1 微机、微处理器、微机系统三者间关系的示意图

图 1.1.1 说明了微机、微处理器、微机系统间的联系。

从图可见, 一个微机是由微处理器、存储器、输入输出接口通过总线组成的。其中微处理器是微机的核心, 又称为中央处理器 (Central Processing Unit) 简称 CPU, 它是智能化的集成芯片。微机连上外设, 例如打印机、键盘、显示器等等, 就构成了微机系统。

二、微处理器

微处理器是微机及微机系统的核心部件，它是由算术逻辑单元（ALU）、寄存器阵列、指令寄存器、指令译码器及一些控制电路通过总线连接组成的。见图 1.1.2，其中 PC 是程序计数器，按它的值 CPU 到指定的存储单元去取指令，然后，PC 值自动增量，指向下一条指令的地址。指令寄存器及指令译码器是用来暂时存放指令及进行译码用的。参加算术逻辑运算的数可以取自 CPU 内部的寄存器也可以取自 CPU 外部的存储器。标志器是用来保存算术逻辑运算后的状态以便判别二个操作数之间的关系（例如大于、小于或等于），从而使程序分支。SP 是堆栈指示器，使用户可以把需要保护的数据或地址存放到存储器的恰当的地方。控制电路用来发出所需的控制信号以协调 CPU 内部各部分的工作。

三、存储器

存储器是构成微机系统必不可少的部件，它是用来存放 CPU 所要执行的指令、各种数据的“仓库”。一个存储器芯片内有许许多多单元，相当“仓库”内有许许多多房间。在微机系统中都要对它们“编号”，以便于从指定的单元中取出所需的信息或把信息存放到指定的单元。每一个存储单元可存放 8 位二进制数。

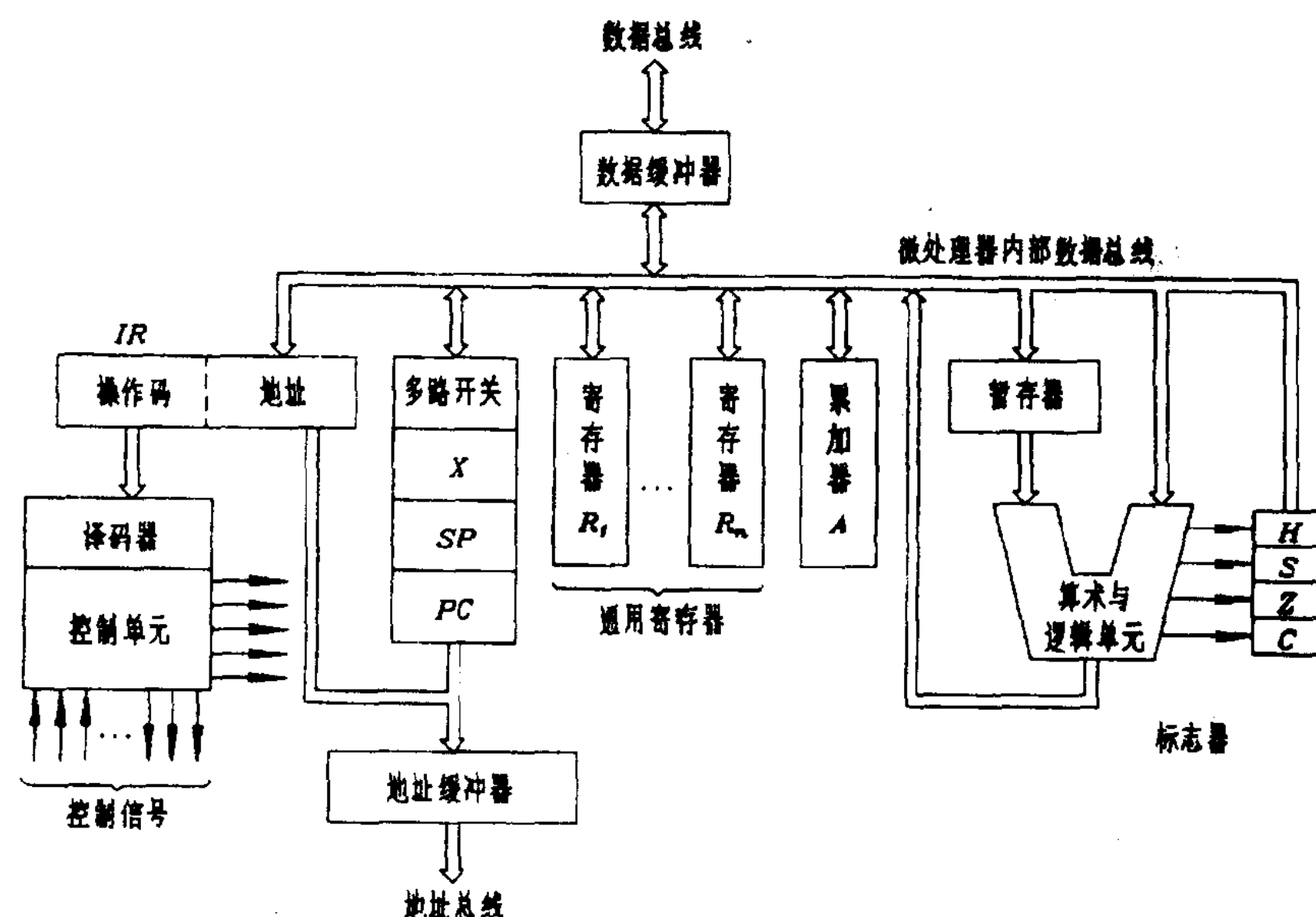


图 1.1.2 微处理器的典型结构

四、输入/输出接口

输入/输出接口是 CPU 与外设（例如打印机、键盘等）进行通讯的“桥梁”，最简单的输入/输出接口例如锁存器或缓冲器，可以利用它们把 CPU 的数据送给外设（例如打印机），也可以通过它们把外设（例如键盘）的数据传送给 CPU，制造商还提供了与 CPU 配套的可编程的接口芯片以供专用。

五、锁存器、触发器与寄存器

锁存器与触发器是具有“记忆”作用的数字电路，它们的差别在于前者是由电平触发的，

而后者是由沿触发的。多位触发器当每位的时钟端和清零端分别都连在一起时，就构成了寄存器。

六、总线与三态逻辑

总线是连结各部分的通道，可分为内总线、外总线，内总线指 CPU 或系统内部的总线，外总线指系统与外设间或系统间的总线。由于存放的信息不同又可分为数据总线、地址总线、及控制总线，有的总线也有复用功能。制造商还提供了一些专用的外部总线例如 RS-232、IEEE-488 等，供系统间采用，这样可以方便用户开发应用系统。

由于多个寄存器、存储器等都挂在同一总线上，如果每个寄存器都处于输出或输入状态，将有许多信息加在总线上，于是总线上的数据将发生混乱。为了解决此问题，一定要采用三态器件，所谓三态指除了‘0’、‘1’二态外还有第三态：高阻态。

图 1.1.3 与图 1.1.4 将说明利用三态器件来保证总线上能正确地传送信息。图 1.1.3 假设有 A、B、C、D 四个寄存器挂在同一条总线上，每个寄存器都有一个控制端 C_A 、 C_B 、 C_C 、 C_D ，图中用小圆圈表示这些控制端是低有效，即当该端为低电平时，寄存器与总线相通，当该端为高电平时，寄存器与总线不通（高阻态）。图 1.1.4 表明当 C_A 有效时，寄存器 A 的数据送上总线，当 C_D 有效时，寄存器 D 通过总线接收寄存器 A 发送的数据，又说明了寄存器 B 的数据如何传送到寄存器 C。一般三态器件都是“开路门”，只有此种器件才能组成“线与”逻辑电路。

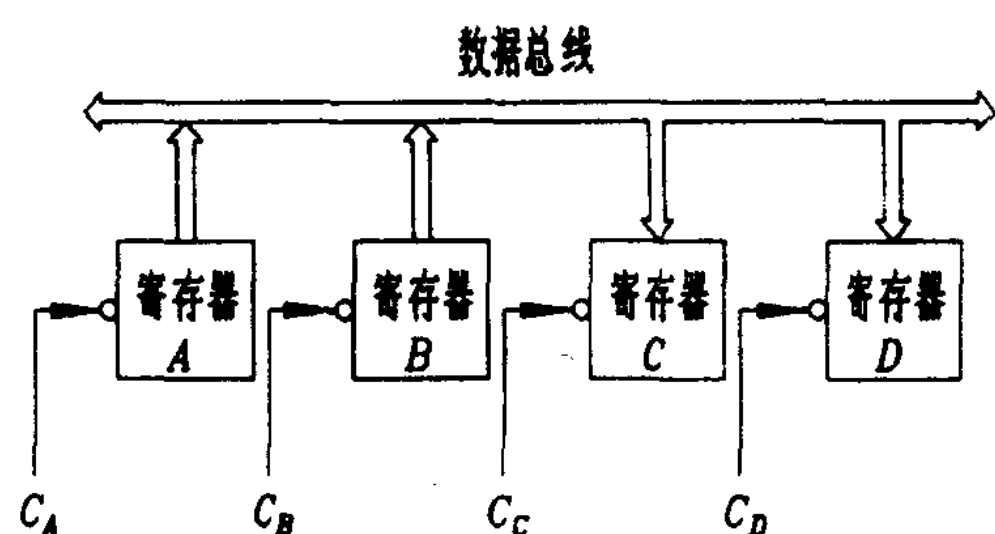


图 1.1.3 总线连结例

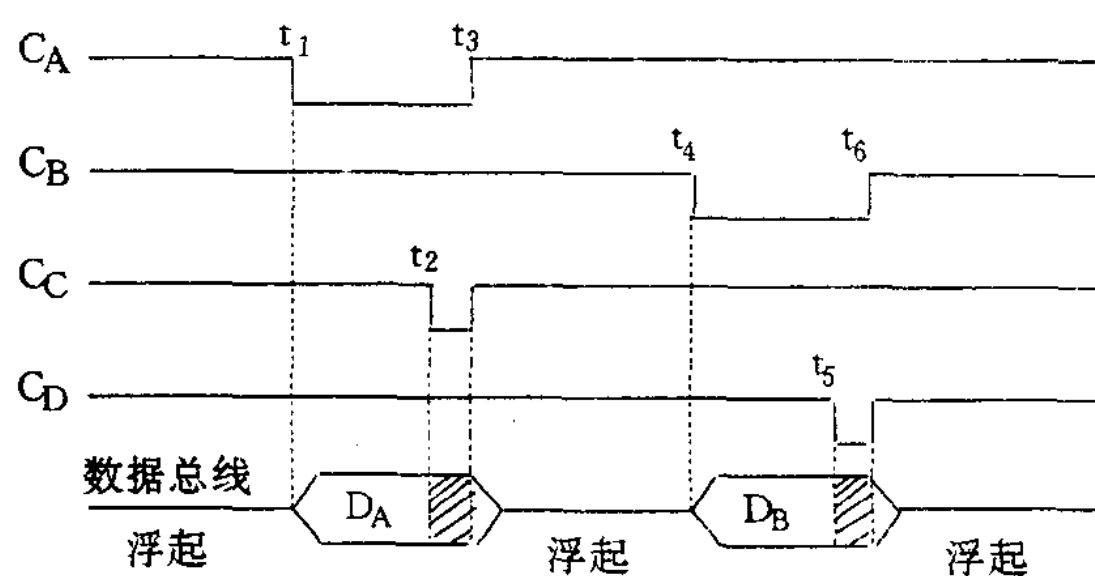


图 1.1.4 总线上数据传送例

1.3 运算基础

一、数制

1. 常用的数制

常用的数制有二进制、八进制、十进制、十六进制等。通用表达式如下：

$$N = A_{n-1}r^{n-1} + A_{n-2}r^{n-2} + \cdots + A_0r^0 + A_{-1}r^{-1} + A_{-2}r^{-2} + \cdots + A_{-m}r^{-m}$$

$$= \sum_{i=-m}^{n-1} A_i r^i$$

当 $r=2$ ， N 为二进制数，它的系数是 0、1。在数的尾部用 B 表示二进制数，例如 11111110B。

当 $r=8$ ， N 为八进制数，它的系数是 0、1、2、3、4、5、6、7。在数的尾部用 O 表示八

进制数，例如 276O。

当 $r=10$ ，N 为十进制数，它的系数是 0、1、2、3、4、5、6、7、8、9。在数的尾部用 D 表示十进制数，例如 254D。

当 $r=16$ ，N 为十六进制数，它的系数为 0、1、2、3、4、5、6、7、8、9、A、B、C、D、E、F。在数的尾部用 H 表示十六进制数。凡是以 A、B、C、D、E、F 开头的十六进制数都要在前面加 0。例如 0FEH，其中第一个 0 用来表示后面的 FEH 是十六进制数字不是字母。

计算机中常采用二进制数。

2. 数制间的转换

(1) 二进制与十进制数间的转换

二进制数转换成十进制数的方法是每位按权展开再逐位相加。例如

$$11111110B = 1 \times 2^7 + 1 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = 254D$$

十进制数转换成二进制数的方法是：整数部分为除 2 取余，小数部分为乘 2 取整。

例 1 将十进制数 175 转换成二进制数

	余数	读数次序
2 175		
2 87	1	
2 43	1	
2 21	1	
2 10	1	
2 5	0	
2 2	1	
2 1	0	
0	1	

所以 $175D = 10101111B$ 。

例 2 将十进制小数 0.8125 转换为二进制数

	整数	读数次序
$0.8125 \times 2 = 1.6250$	1	
$0.6250 \times 2 = 1.2500$	1	
$0.2500 \times 2 = 0.5000$	0	
$0.5000 \times 2 = 1.0000$	1	

所以， $0.8125D = 0.1101B$ 。

注意：将十进制小数转换为二进制数的过程可能会无限进行下去，这时可根据数据精度要求取足够位数即可。

如果十进制数有整数部分又有小数部分，应分开处理。如：

$$175.8125D = 10101111.1101B$$

(2) 二进制与八进制、十六进制数间的转换

由于二进制与八进制间的关系是 2 的三次方为 8，故把二进制数的整数部分从右向左 3 位一撇，小数部分从小数点开始向右 3 位一撇，不足 3 位添 0，就化为八进制数。例如

$$10101111.1101B \text{ 处理为 } 010,101,111.110,100$$

$$\text{所以 } 10101111.1101B = 257.64O$$

可以推得二进制转换成十六进制时，只要把二进制数的整数部分从右向左 4 位一撇，小数部分从小数点开始向右 4 位一撇，不足 4 位添 0，就化为十六进制数。例如

$$10101111.1101B \text{ 处理为 } 1010,1111.1101$$

$$10101111.1101B = 0AF.DH$$

反之，一位八进制数可以化为 3 位二进制数，一位十六进制数可以化为 4 位二进制数。例如

$$\begin{array}{ccccccc} 2 & 5 & 7 & . & 6 & 4 & O \\ 010 & 101 & 111 & . & 110 & 100 & B \end{array}$$

$$\text{即 } 257.64O = 10101111.1101B。$$

$$\begin{array}{ccccccc} 0 & A & F & . & D & H \\ & 1010 & 1111 & & 1101 & \end{array}$$

$$\text{即 } 0AF.DH = 10101111.1101B。$$

(3) 十进制数与八进制、十六进制数间的转换

从十进制数转换成二进制数得到启发，十进制数转换成八进制数的方法是其整数部分为除 8 取余，小数部分为乘 8 取整。十进制数转换成十六进制数的方法可类推。

反之，从八进制、十六进制数转化为十进制数的方法是按权展开即可。

二、BCD 码与 ASCII 码

1. BCD 码 (Binary Coded Decimal)

由于日常都使用十进制，为了在计算机内使用十进制数，就要用二进制来表示十进制，即采用 BCD 码的方式，每一位 BCD 码由 4 位二进制数表示。由于编码的方式不同（每位权不同），BCD 码也有不同的表示。表 1.1 为 8421 编码方式的 BCD 码与二进制数间的关系。

由于一位 BCD 码由 4 位二进制数来表示，实际上，4 位二进制数可以表示 0~15，作为十进制数只能表示 0~9，相差 6，于是用 BCD 码进行加减法运算时要进行调整。

$$\text{例如 } 3D + 7D = 10D$$

$$\text{用 BCD 码来表示 } 3D = 0011BCD$$

$$7D = 0111BCD$$

$$\text{而 } 0011B$$

$$+ 0111B$$

$$1010B$$

1010 大于 9D，需调整如下：

$$\begin{array}{r}
 1010 \\
 +0110 \quad \dots 6 \\
 \hline
 10000 \quad \dots 00010000\text{BCD}
 \end{array}$$

即可得到和为 10D，这样的结果才正确。

表 1.1 BCD 码与 4 位二进制数的关系

十进制数	二进制数	BCD 码
0	0000	0000
1	0001	0001
2	0010	0010
3	0011	0011
4	0100	0100
5	0101	0101
6	0110	0110
7	0111	0111
8	1000	1000
9	1001	1001
10	1010	—
11	1011	—
12	1100	—
13	1101	—
14	1110	—
15	1111	—

2. ASCII 码

字母和字符的二进制编码可以有几种方案，通常用的是 ASCII 码（American Standard Code For Information Interchange）即美国标准信息交换码。见表 1.2。

表 1.2 七位 ASCII 码代码表

行 \ 列	十六进制码高位	0	1	2	3	4	5	6	7
十六进制码低位	654位 3210位	000	001	010	011	100	101	110	111
0	0 0 0 0	NUL	DLE	SP	0	@	P	,	p
1	0 0 0 1	SOH	DC1	!	1	A	Q	a	q
2	0 0 1 0	STX	DC2	"	2	B	R	b	r
3	0 0 1 1	ETX	DC3	#	3	C	S	c	s
4	0 1 0 0	EOT	DC4	\$	4	D	T	d	t
5	0 1 0 1	ENQ	NAK	%	5	E	U	e	u
6	0 1 1 0	ACK	SYN	&	6	F	V	f	v
7	0 1 1 1	BEL	ETB	'	7	G	W	g	w
8	1 0 0 0	BS	CAN	(	8	H	X	h	x
9	1 0 0 1	HT	EM	)	9	I	Y	i	y
A	1 0 1 0	LF	SUB	*	:	J	Z	j	z
B	1 0 1 1	VT	ESC	+	;	K	[	k	{
C	1 1 0 0	FF	FS	,	<	L		l	
D	1 1 0 1	CR	GS	-	=	M	]	m	}
E	1 1 1 0	SO	RS	.	>	N	↑	n	~
F	1 1 1 1	ST	US	/	?	O	←	o	DEL

注：第 0、1、2、7 列中特殊控制功能符号的意义。

从表可见，ASCII 码是一种 7 位二进制编码，可以表示 128 个字符，包括十进制数 0~9，英文大、小写字母，及一些非打印字符，其中一些非打印字符的意义如下：

NUL	空
SP	空格
SOH	标题开始
STX	标题的结束及信息数据的开始
ETX	信息数据的结束
EOT	传输结束
ENQ	询问
ACK	承认
BEL	报警
BS	退一格
HT	水平制表
LF	换行
VF	垂直制表
FF	换页（走纸控制）
CR	回车
SO	移位输出
SI	移位输入
DLE	数据键换码
DC1	设备控制 1

DC2	设备控制 2
DC3	设备控制 3
DC4	设备控制 4
NAK	否定
SYN	空转同步
ETB	信息组传送结束
CAN	清除
EM	中程结束 (纸尽)
SUB	替换 (减)
ESC	换码
FS	文件分隔号
GS	组分隔号
RS	记录分隔号
US	单元分隔号
DEL	删除

三、二进制数的原码、反码、补码及其运算

1. 原码

计算机所用的数是用二进制表示的。如果用 8 位二进制数中的最高位表示该数的符号,当最高位为 1,表示该数为‘负’,当最高位为 0,表示该数为‘正’。用这种方式表示的数称为机器数,机器数的值称为真值。例如

+32D 的机器数表示为 00100000B 最高位为 0,故为正数。

-32D 的机器数表示为 10100000B 最高位为 1,故为负数。

上面这种机器数在计算机中又称为原码。

2. 反码

计算机科学规定:正数的反码等于原码,负数的反码等于原码除符号位外按位取反。例如

00100000B 的反码就为 00100000B

而 10100000B 的反码为 11011111B

3. 补码

从十进制运算中可知,如果限制结果为个位,那么

$$10D - 4D = 10D + 6D = 6D \quad \text{其中十位被舍去}$$

可见,只要把 4D 换成 6D,那么减法就可用加法来完成,这是因为 4D 与 6D 对模 10D 而言为互补。

推广到二进制运算中,引入补码的概念。规定正数的补码等于原码,负数的补码为它的绝对值取反加 1。例如

$$X = -32D = 10100000B$$

$$X \text{ 的补码} = \overline{X} + 1 = 11100000B$$

引入补码后,二进制减法就可以转换成加法。