

高校计算机教学系列教材

微机原理及汇编语言教程

杨延双 张晓冬 魏坚华 编著

北京航空航天大学出版社

内 容 简 介

本书详细介绍了PC系列的微型计算机原理与汇编语言程序设计的方法。全书共9章,内容包括: Intel系列处理器;汇编语言指令系统和汇编语言的程序设计;存储器;中断系统;输入/输出系统;总线技术。在每章后面均附有习题。全书后面附有7个上机实验,并在附录中给出了参考答案。内容丰富,深入浅出,注重实用,是面向高等职业教育的特点而编写的。本书适合大专、高等职业教育及非计算机专业本科作教材;相关专业技术人员也可作为参考书。

图书在版编目(CIP)数据

微机原理及汇编语言教程/杨延双等编著. —北京:

北京航空航天大学出版社,2002.6

ISBN 7-81077-190-6

I. 微… II. 杨… III. ①微型计算机—理论—教材②汇编语言—程序设计—教材 IV. TP36

中国版本图书馆CIP数据核字(2002)第020187号

微机原理及汇编语言教程

杨延双 张晓冬 魏坚华 编著

责任编辑 许传安

责任校对 陈 坤

北京航空航天大学出版社出版发行

北京市海淀区学院路37号(100083) 发行部电话:82317024 传真:82328026

<http://www.buaapress.com.cn>

E-mail:pressell@publica.bj.cninfo.net

北京市宏文印刷厂印装 各地书店经销

*

开本:787×1092 1/16 印张:14.5 字数:371千字

2002年6月第1版 2002年6月第1次印刷 印数:5000册

ISBN 7-81077-190-6/TP·107 定价:20.00元

总前言

科教兴国,教育先行,在全国上下已形成共识。在教育改革过程中,出现了多渠道、多形式、多层次办学的局面。同时,政府逐年加大教育的投入力度。教育发展了,才能有效地提高全民族的文化、科学素质,使我们中华民族屹立于世界民族之林。

计算机科学与技术的发展日新月异,其应用领域迅速扩展,几乎无处不在。社会发展的需求,促使计算机教育生气蓬勃。从普通高校的系统性教学,到远距离的电视、网上教学;从全面讲述,到不同应用领域的、星罗棋布的培训班;从公办的到民办的;从纸介教材到电子教材等等,可以说计算机教学异彩纷呈。要进行教学,就必须有教材。

面对我们这么大的国家和教学形势,在保证国家教学基本要求的前提下,应当提倡教材多样化,才能满足各教学单位的需求,使他们形成各自的办学风格和特色。为此,我们组织北京工业大学、北京航空航天大学、北京理工大学、南开大学、天津工业大学等高校的有丰富教学经验的教师编写了计算机教学的系列教材,将陆续与师生见面。

系列教材包括以下各项:

(一) 基础理论:离散数学。

(二) 技术基础:电路与电子技术基础;数字逻辑基础;计算机组成与体系结构;计算机语言(拼盘、选择使用),包括 C++ 程序设计基础、Visual Basic 程序设计基础、Matlab 程序设计基础、Java 程序设计基础、Delphi 语言基础、汇编语言基础等;数据结构;计算机操作系统基础;计算方法基础;微机与接口技术;数据库技术基础等。

(三) 应用基础:计算机控制技术;网络技术;软件工程;多媒体技术等。

(四) 技术基础扩展:编译原理与编译构造;知识工程——网络计算机环境下的知识过程。

(五) 应用基础扩展:计算机辅助设计;单片机实用基础;图像处理基础;传感器与测试技术;计算机外设与接口技术。

本系列教材主要是针对计算机专科教学编写的,供普通高校、社会民办大学、高等职业学校、业余大学等计算机专科使用。其中一部分教材也适合计算机本科教学或非计算机专业本科教学使用。在这部分教材的内容简介或前言中对使用范围均作了说明。

本系列教材在编写时,注重以下几点:(1)面对计算机科学与技术动态发展的现实,在内容上应具有前瞻性;(2)面对学以致用,既有系统的基础知识,又有应用价值的实用性;(3)具有科学性、严谨性。另外,力求排版紧凑,使有限的版面具有最大的信息量,以使读者得到实惠。

能否实现这些愿望,要由师生在教学实践中评价。我们期望得到师生的批评和指正。

高校计算机教学系列教材编委会

高校计算机教学系列教材编委会成员

主任:赵沁平

副主任(常务):陈炳和

顾问:麦中凡

委员(以姓氏笔划为序):

吕景瑜(北工大教授)

乔少杰(社长,副教授)

麦中凡(北航教授,教育部工科计算机基础教学指导委员会副主任、中专计算机教学指导委员会顾问)

苏开娜(北工大教授)

陈炳和(北工大教授)

张鸿宾(北工大博导)

郑玉明(北工大副教授)

金茂忠(北航博导)

赵沁平(北航博导,教育部副部长)

前 言

微机原理与汇编语言是学习和掌握计算机技术的重要内容。在传统的教学计划中是将“微机原理”和“汇编语言程序设计”分开单独设课。而近年来高等职业教育蓬勃发展,由于其校情、师生、生源等与普通高等教育有所不同,所以在课程设置上也有其特殊的需求和自身的特点。我们为适应这种教育发展形势而编写了这本《微机原理及汇编语言教程》。

本课程的前修课程为“数字逻辑”。本课程是“微机接口技术”、“操作系统”、“计算机体系结构”等课程的必要先修课。由于本课程在计算机专业必修课中的位置,更体现了它的重要性。读者通过本课程的学习,可深入了解微型计算机系统的组成、工作原理,掌握汇编语言程序设计技术,为微型计算机技术的应用打下良好基础。

全书由9章组成。第1章为概述;第2章全面介绍了Intel系列微处理器;第3~5章为汇编语言系统和汇编语言程序设计;第6章介绍了存储器;第7章对中断系统结构、工作原理和中断控制器进行了详细的论述;第8章介绍了输入/输出接口的概念和控制方式;第9章为总线技术,介绍了常用的总线标准。实验部分编排了7个上机实验,并在附录中给出了参考答案。

本书作者都是有多年的教学经验和实践经验的教师。本书内容丰富,深入浅出,注重实用,是面向高等职业教育的特点而编写的,有利于学生对未知领域的掌握和运用。

本书的第1,2,6,9章由杨延双编写;第3,4,5章由张晓冬编写;第7,8章及实验由魏坚华编写。全书由张载鸿教授主审。

在本书的编写过程中,得到了张载鸿教授的大力支持,并提供了宝贵意见;尹子赓、尹志军、刚冬梅承担了书稿的录入、校对等工作。在此一并感谢。

由于作者水平有限,书中难免存在错误及不妥之处,敬请专家和广大读者批评指正。

编 者

2001.10

目 录

第 1 章 概 述

1.1 微型计算机发展概况	1
1.2 微型计算机的特点与分类	2
1.3 微处理器、微型计算机和微型计算机系统	2
1.3.1 微处理器	2
1.3.2 微型计算机	2
1.3.3 微型计算机系统	3
习 题	3

第 2 章 Intel 系列微处理器

2.1 微处理器的基本结构	4
2.2 微处理器的工作原理	4
2.3 16 位微处理器	5
2.3.1 8086 的内部结构和引脚	5
2.3.2 8086 的存储器组织	10
2.3.3 80286 的内部结构	11
2.4 32 位微处理器	14
2.4.1 80386 的基本结构	14
2.4.2 80386 的引脚信号	18
2.5 奔腾(Pentium)微处理器	19
2.5.1 Pentium 的系统结构	20
2.5.2 Pentium 微处理器的技术特点	21
习 题	21

第 3 章 指令系统

3.1 80x86 的指令格式	22
3.1.1 操作码字段	22
3.1.2 地址码字段	24
3.2 80x86 的寻址方式	24
3.2.1 8086/8088 的寻址方式	25
3.2.2 80x86 的寻址方式	29
3.3 8086 指令系统	30
3.3.1 数据传送指令	30
3.3.2 算术运算指令	35

3.3.3	逻辑运算和移位指令	40
3.3.4	串操作指令	44
3.3.5	控制转移指令	49
3.3.6	处理器控制指令	52
3.4	80x86 增强和扩充的指令	53
3.4.1	80286 增强和扩充的指令	53
3.4.2	80386 新增加的指令	56
3.4.3	80486 新增加的指令	59
3.5	Pentium 指令集	60
	习 题	66

第 4 章 汇编语言程序格式

4.1	汇编程序功能	70
4.2	伪指令语句	71
4.2.1	符号定义伪指令	71
4.2.2	数据定义伪指令	73
4.2.3	段定义伪指令	75
4.2.4	过程定义伪指令	78
4.2.5	其它伪指令语句	78
4.3	宏指令	80
4.3.1	宏的使用	80
4.3.2	宏定义中所使用的其它伪指令	82
4.4	汇编语言程序格式	85
4.4.1	名字部分	85
4.4.2	操作符部分	86
4.4.3	操作数部分	86
4.4.4	注释部分	90
4.5	汇编语言程序的上机过程	91
4.5.1	建立软件环境	91
4.5.2	汇编程序	92
4.5.3	连接程序	94
4.5.4	程序的执行与调试	95
	习 题	99

第 5 章 汇编语言程序设计

5.1	程序设计的基本步骤	102
5.2	循环程序设计的基本步骤	102
5.2.1	循环程序的结构形式	102
5.2.2	循环程序设计方法	103
5.2.3	多重循环程序设计	109

5.3 分支程序设计	111
5.3.1 分支程序设计概述	111
5.3.2 分支程序设计方法	112
5.4 子程序设计	119
5.4.1 子程序概念	119
5.4.2 子程序的调用和返回	119
5.4.3 子程序的设计方法	122
5.4.4 嵌套与递归子程序	130
5.5 DOS 系统功能调用	132
习 题	133
第 6 章 存储器	
6.1 概 述	137
6.1.1 存储器的分类	137
6.1.2 存储器的主要性能指标	138
6.2 半导体存储器	139
6.2.1 读/写存储器 RAM	139
6.2.2 只读存储器 ROM	142
6.2.3 由 RAM 芯片组成微型机的读/写存储器	146
6.3 高速缓冲存储器(cache)	148
习 题	149
第 7 章 中断系统	
7.1 中断的基本概念	150
7.1.1 中 断	150
7.1.2 中断过程	151
7.2 8086 的中断结构	153
7.2.1 中断源类型	153
7.2.2 中断向量表	155
7.2.3 中断源优先级	160
7.2.4 BIOS 系统功能调用	161
7.3 可编程中断控制器	162
7.3.1 中断控制器的功能	162
7.3.2 8259A 的引脚及其编程结构	163
7.3.3 8259A 的编程控制	164
7.3.4 8259A 的工作方式	171
7.4 80286 的中断	181
7.4.1 中断或异常	181
7.4.2 中断或异常的响应过程	182
习 题	182

第 8 章 输入输出系统

8.1 概 述	184
8.1.1 I/O 接口	184
8.1.2 CPU 与 I/O 设备之间的信号	185
8.1.3 I/O 接口的基本功能	186
8.2 I/O 端口的编址方式	186
8.2.1 端口统一编址方式	186
8.2.2 端口独立编址方式	187
8.3 I/O 控制方式	188
8.3.1 程序控制方式	188
8.3.2 中断控制方式	192
8.3.3 DMA 方式	193
8.4 I/O 指令	196
8.4.1 输入指令	196
8.4.2 输出指令	197
习 题	197

第 9 章 总线技术

9.1 MULTIBUS 的信号和总线操作	200
9.1.1 MULTIBUS 总线的信号和定义	200
9.1.2 MULTIBUS 的总线操作	202
9.2 ISA 总线	204
9.3 EISA 总线	205
9.4 VESA 总线	206
9.5 PCI 总线	206
习 题	207

附 录

实验一 十六进制转换到十进制	208
实验二 十六进制转换到二进制	208
实验三 二位十进制加法	208
实验四 排 序	208
实验五 函数计算	209
实验六 ASCII 表生成	209
实验七 实时时钟显示	209
有关实验的参考程序	210

参考文献

第1章 概述

1.1 微型计算机发展概况

电子计算机按其体积和性能分为巨型机、大型机、中型机、小型机和微型机。微型计算机是以微处理器为核心,配上存储器、输入/输出接口电路和系统总线所构成。微型计算机的发展通常是以微处理器的升级而换代的。

微处理器的问世是在1971年。美国Intel公司生产的4004微处理器采用了PMOS技术。在 $4.2\text{ mm}\times 3.2\text{ mm}$ 的硅片上集成了2 250个晶体管,可进行4位二进制的并行处理。后来Intel公司正式生产了通用的4040微处理器。这种4位的微处理器以体积小,价格低而引起人们的兴趣。以4004为核心组成的MCS-4是世界上第一代微型计算机。

1974年—1978年,Intel公司推出的8080/8085、Zlog公司推出的Z80、Motorola公司推出的MC6800/6802等,被称为第二代微处理器。这一代8位的微处理器的特点是采用NMOS电路,集成度达5 000个晶体管/片以上,时钟频率为 $2\text{ MHz}\sim 4\text{ MHz}$ 。这时的微处理器的设计和生产技术已相当成熟。

第三代是以16位微处理器的出现(1978年)为标志的。典型产品为Intel的8086、Zlog的Z 8000和Motorola的MC68000。它们采用了HMOS工艺,集成度为(20 000~60 000,管/片,时钟频率为 $4\text{ MHz}\sim 8\text{ MHz}$,平均指令执行时间为 $0.5\ \mu\text{s}$ 。

第四代微型计算机(1985年—1992年)是32位微型机。典型的微处理器产品有80386/486、MC 68020、Z 80000等。

1993年Intel公司推出的Pentium微处理器,宣布了第五代微处理器的诞生。这种64位的Pentium微处理器芯片采用了新的体系结构。芯片的集成度达 $(5\times 10^6\sim 9.3\times 10^6)$ 管/片,时钟频率达 $150\text{ MHz}\sim 300\text{ MHz}$ 。

1995年Pentium II问世,1999年Pentium III公布。目前,市场上的主流产品是Pentium IV。更高性能的微处理器将不断推出,微型计算机的发展速度是惊人的,而其性能/价格比日渐提高,微型计算机技术的应用越来越广泛。

1.2 微型计算机的特点与分类

微型计算机与巨型机、大型机、中型机、小型机相比,最主要的特点是体积小、功耗低、价格低廉、可靠性高、硬件结构设计灵活、安装维修方便及具有丰富的软件。微型计算机的这些特点极大地赢得了用户的欢迎,使其应用日益广泛。

微型计算机的分类可以从不同角度进行:如果从制造工艺来分,可将微型机分为 MOS 型和双极型;若从组装形式划分,则可以分为单片、单板和多板微型机;按微处理器的字长来划分,通常可分为 4 位机、8 位机、16 位机、32 位机、64 位机及位片式等。位片式微处理器是以位为单位,由若干个位片组合而构成不同字长的微型计算机,其特点是结构灵活。目前,市场上的主流产品是 64 位机。

1.3 微处理器、微型计算机和微型计算机系统

1.3.1 微处理器

微处理器由一片或几片大规模集成电路组成,具有运算和控制功能的中央处理器部件(central processing unit)简称 CPU。

微处理器在内部结构上一般包括:

- 算术逻辑部件 ALU;
- 寄存器组;
- 程序计数器、指令寄存器和译码器;
- 时序和控制部件。

微处理器是微型计算机的核心。

1.3.2 微型计算机

微型计算机是以微处理器为核心,再配上存储器、输入/输出接口电路和系统总线,如图 1.1 所示。

- 存储器 包括只读存储器 ROM 和随机存取存储器 RAM。它们用来存储程序和数据。
- 输入/输出接口电路 用来控制微机与外部设备之间的信息交换。
- 系统总线 用来在微型计算机的部件和部件之间进行信息传输的一组总线,通常包括地址总线、数据总线和控制总线。

地址总线用来传送地址信息,为单向输出。地址总线的位数决定了 CPU 可以直接寻址

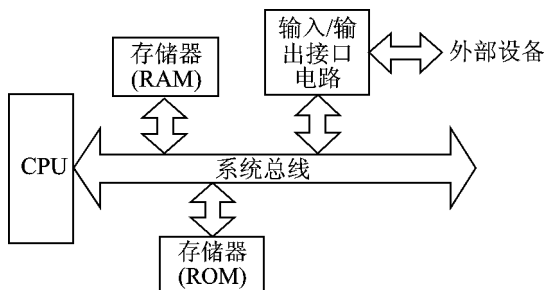


图 1.1 微型计算机的结构框图

的内存范围。如地址总线为 16 位时,可直接寻址范围为 $2^{16} = 64\text{K}$ 单元。

数据总线是用来传输数据的,双向。数据总线的位数和 CPU 的位数相对应。如对于 16 位的 CPU,其数据总线的宽度为 16 位。

控制总线用来传输控制信息。

1.3.3 微型计算机系统

以微型计算机为中心,再配上外部设备和相应的软件就组成了微型计算机系统。微机系统包括两大部分:硬件和软件,如图 1.2 所示。

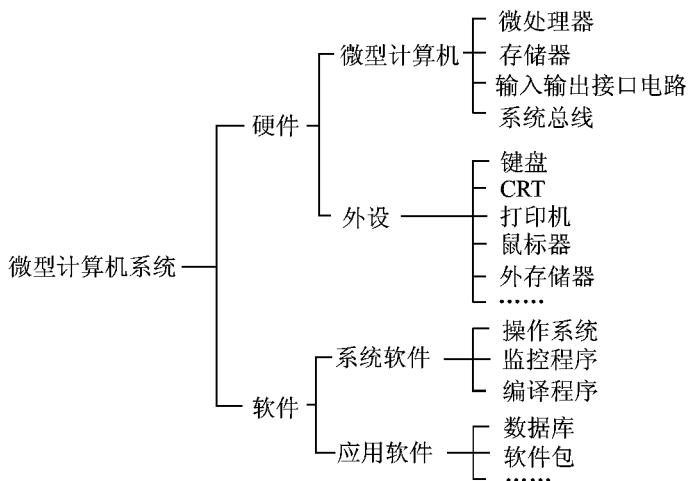


图 1.2 微型计算机系统

习 题

1. 微型计算机有哪些主要特点？
2. 说明微处理器、微型计算机和微型计算机系统有什么不同及三者之间的关系。

第 2 章

Intel 系列微处理器

2.1 微处理器的基本结构

从内部结构上,微处理器一般都包含下列功能部件:

- 运算器:算术逻辑部件 ALU;
- 寄存器:累加器和通用寄存器组;
- 程序计数器、指令寄存器和指令译码器;
- 时序和控制部件。

微处理器内部的算术逻辑部件是用来执行基本的算术运算和逻辑运算的。它可以进行加、减、乘、除的算术运算和与、或、非、异或等逻辑运算。累加器和通用寄存器组用来存放参加运算的数据、中间结果及存储运算结果的状态标志,也用来存放地址。程序计数器总是指向下一条要执行的指令;指令寄存器存放从存储器中取出的指令码。而指令译码器是对指令码进行译码和分析,以完成指定的操作。时序和控制部件具有指挥整个系统操作次序的功能。

现代的微处理器均为单片型,即由一片或几片超大规模集成电路制成。其集成度越来越高,性能也越来越高。

2.2 微处理器的工作原理

微处理器是通过执行程序来完成预定任务的,即逐条从存储器中取出程序中的指令并完成指令所指定的操作。

微处理器执行程序一般是通过反复执行以下步骤而实现的。

首先,从程序计数器所指向的存储器单元中取出一条指令(由于程序一般存放在内存的一个连续区域,所以顺序执行程序时,每取一个指令字节,程序计数器就自动加 1),存放到指定的寄存器。其次,由指令译码器对指令码进行译码和分析,来确定指令的操作。若指令要求操

作数,则确定操作数的地址,读出操作数。之后,执行指令内容(算术运算或逻辑运算)。最后,指令译码器译码时产生的相应控制信号送到时序和控制逻辑电路,控制 CPU 内部及整个系统来协调工作,从而完成指令所指定的操作任务。

2.3 16 位微处理器

8086 是 Intel 系列的 16 位微处理器。它采用高密度的硅栅 H-MOS 工艺制造,内部包含近 29 000 只晶体管。它采用 40 根引脚双列直插式封装,单一的 5 V 电源和单相时钟。8086 有 16 位数据线和 20 位地址线,可寻址空间为 2^{20} B,即 1 MB。

8088 是 Intel 公司继 8086 之后又推出的一种准 16 位的微处理器。8088 内部是 16 位 CPU,而外部的数据总线是 8 位的。这样就可以与当时已有的一整套 Intel 外围设备接口芯片直接兼容。

8086/8088 是 Intel 系列 CPU 中最具有代表性的 16 位微处理器。随后 Intel 公司陆续推出的 80x86 都是按其模式加以升级的,均保持与 8086/8088 兼容。

2.3.1 8086 的内部结构和引脚

2.3.1.1 8086 的内部结构

1. 框图

8086 的内部结构框图如图 2.1 所示。

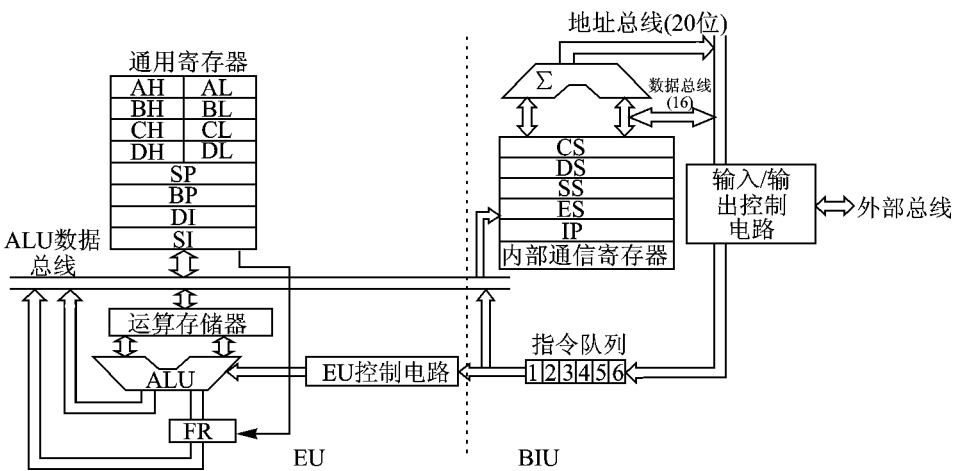


图 2.1 8086CPU 内部结构框图

8086 的 CPU 由两个功能部件:EU(Execution Unit)和 BIU(Bus Interface Unit)所构成。下面分别介绍这两个功能部件。

(1) 总线接口部件 BIU

总线接口部件由下列部分组成:4 个 16 位的段寄存器 CS,SS,DS,ES;16 位的指令指针寄存器 IP;20 位的地址加法器 Σ ;内部通信寄存器;6 字节的指令队列和输入/输出控制电路。

BIU 的功能负责与存储器、I/O 端口传送数据,即当指令队列空时,BIU 从内存取指令放入指令队列;当 CPU 执行指令时,BIU 配合 EU 从指定的内存单元或外设端口中取出数据供 EU 使用;当运算结束时,BIU 运算结果传送到指定内存单元或外设端口中。

BIU 中的地址加法器 Σ 是进行物理地址计算的,即将段寄存器中的 16 位数值和偏移量相加而得到 20 位物理地址。

(2) 执行部件 EU

执行部件包括 8 个 16 位通用寄存器、算术逻辑单元 ALU、标志寄存器 FR、运算寄存器和执行部件控制电路。

执行部件负责执行指令。EU 从指令队列取出指令代码,将其译码,发出相应的控制信息,并对通用寄存器和标志寄存器进行管理。

2. 寄存器

8086CPU 内部共有 14 个寄存器。下面分别加以介绍。

(1) 通用寄存器

8086CPU 有 8 个 16 位的通用寄存器。这 8 个通用寄存器可分为两组。一组称为数据寄存器,包括 AX,BX,CX 和 DX,用来存放数据或地址。这 4 个数据寄存器可以作为 16 位寄存器使用,也可以作为 8 位寄存器使用,即每一个 16 位数据寄存器都可分成两个独立的 8 位寄存器,见图 2.1 中的 AH 和 AL,BH 和 BL,CH 和 CL,DH 和 DL。其中 AH,BH,CH,DH 为高 8 位,AL,BL,CL,DL 为低 8 位。

通用寄存器的另一组包括堆栈指针寄存器 SP、基地址寄存器 BP、源变址寄存器 SI 和目的地址寄存器 DI。这 4 个寄存器只能作为 16 位寄存器使用,主要用来存放存储器或输入/输出端口的地址,也可以用来存放数据。

(2) 段寄存器

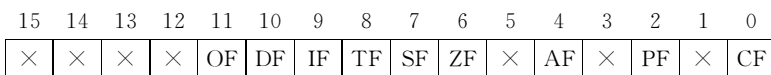
8086CPU 有 4 个 16 位的段寄存器:代码段寄存器 CS、数据段寄存器 DS、附加段寄存器 ES 和堆栈段寄存器 SS。段寄存器用来存放段基地址(或称为首地址),即 CS 中存放有当前执行程序所在段的首地址;DS 存放数据段首地址;SS 存放当前堆栈段的首地址;ES 存放当前附加段首地址。

(3) 指令指针寄存器 IP(16 位)

IP 用来指明将要执行的下一条指令的偏移地址。

(4) 标志寄存器 FR(16 位)

标志寄存器中的 7 位未用,所用 9 位的含义如图 2.2 所示。



×:表示该位未用。

图 2.2 8086 标志寄存器的标志

标志寄存器的 9 个标志位可按照功能的不同分为两类:状态标志和控制标志。

状态标志用来表示算术运算和逻辑运算结果的特征,有如下 6 位。

- CF:进位标志。当进行加或减法运算时,若最高位产生进位或借位时,则 CF 为 1;否则 CF 为 0。另外,循环指令、移位指令也会影响这一标志位。
- PF:奇偶标志。当本次运算结果中的低 8 位中 1 的个数为偶数时,PF 为 1;否则 PF 为 0。
- AF:辅助进位标志。当本次运算,如果第 3 位往第 4 位有进位或借位时,AF 为 1;否则 AF 为 0。此标志在 BCD 码运算中作为是否进行十进制调整的依据。
- ZF:零标志。当运算结果为 0 时,ZF 为 1;否则 ZF 为 0。
- SF:符号标志。当运算结果最高位为 1 时,表示结果为负数,SF 为 1;否则 SF 为 0。
- OF:溢出标志。当运算结果产生溢出,即运算结果超出了相应类型数据所能表示的范围,OF 为 1;否则 OF 为 0。

控制标志用来控制 CPU 的操作,有如下 3 位。

- TF:单步标志(跟踪标志)。当 TF 为 1 时,CPU 按单步工作方式执行指令。在单步工作方式下,CPU 每执行完一条指令就自动产生一次内部中断。此功能便于程序的调试。
- IF:中断标志。这是控制可屏蔽中断的标志。当 IF 为 1 时,允许 CPU 响应可屏蔽中断;当 IF 为 0 时,则 CPU 不能响应可屏蔽中断请求。
- DF:方向标志。用来控制串操作指令的执行。若 DF 为 1,则串操作按减地址方式进行操作;DF 为 0 时,则串操作指令按增地址方式进行操作。

2.3.1.2 8086CPU 的引脚

8086CPU 的引脚分配如图 2.3 所示。

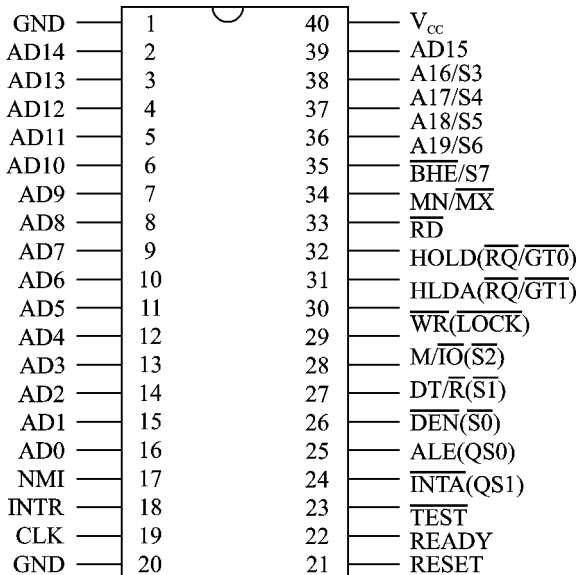


图 2.3 8086CPU 的引脚信号

由于 8086CPU 可有两种工作模式:最小模式(单 CPU 模式)和最大模式(多 CPU 模式),所以 8086CPU 的引脚 24 ~31 都有两种定义功能。图 2.3 括号中的引脚定义为最大模式下

的定义。

下面对 8086CPU 的各引脚功能作简要说明。

1) AD15~AD0(Address / Data BUS):分时复用的地址/数据总线。传送地址时,三态输出;传送数据时,可以双向三态输入/输出。在 DMA 操作期间,这些引脚为浮空状态。

2) A19/S6~A16/S3(Address / Status):分时复用的地址/状态线。作为地址线用时,A19~A16 与 A15~A0 一起构成访问存储器的 20 位物理地址。当 CPU 访问 I/O 端口时,A19~A16 均保持为 0。作状态线用时,S6~S3 输出状态信息。用 S4 和 S3 的不同编码表示当前使用哪一个段寄存器来对存储器寻址。S4,S3 编码与所使用的段寄存器的对应关系如表 2-1 所列。S5 表示可屏蔽中断允许标志的当前状态。当 IF=1 时,S5 为 1。S6 保持恒为低电平。在 DMA 操作期间,这些引脚均为浮空状态。

表 2-1 S4,S3 状态编码

S4	S3	段寄存器
0	0	ES
0	1	SS
1	0	CS(或未用任何段寄存器)
1	1	DS

3) $\overline{\text{BHE}}/\text{S7}(\text{BUS High Enable / Status})$:数据总线高 8 位有效信号。当 CPU 读/写存储器或 I/O 端口时, $\overline{\text{BHE}}$ 用作体选信号,与 AD0 配合来表示总线使用情况,如表 2-2 所示。在其它时间作为状态信号 S7,但 S7 未定义。

表 2-2 BHE 和 AD0 编码的含义

$\overline{\text{BHE}}$	AD0	总线使用情况
0	0	AD15~AD0 16 位数据总线上进行字节传送
0	1	AD15~AD8 8 位数据总线上进行字节传送
1	0	AD7~AD0 8 位数据总线上进行字节传送
1	1	无效

4) NMI(Non-Maskable Interrupt):非屏蔽中断请求信号输入,上升沿触发。非屏蔽中断不受 IF 的影响,不能用软件屏蔽。CPU 一旦检测到 NMI 请求有效,就会在当前指令结束后,执行中断类型为 2 的中断处理程序。

5) INTR(Interrupt Request):可屏蔽中断请求信号输入,高电平有效。CPU 在每条指令的最后一个时钟周期对 INTR 进行采样。一旦发现 INTR=1 时,并且当前中断允许标志 IF=1 时,则 CPU 在当前指令结束后,响应中断请求,转入中断响应周期。

6) $\overline{\text{INTA}}$ (Interrupt Acknowledge):中断响应信号输出,低电平有效。表示 CPU 响应了外部中断请求信号 INTR,发给请求中断的设备的回答信号。

7) $\overline{\text{RD}}$ (Read):读信号。三态输出,低电平有效。表示当前 CPU 正在读存储器或 I/O 端口。

8) $\overline{\text{WR}}$ (Write):写信号。三态输出,低电平有效。表示当前 CPU 正在写存储器或 I/O 端口。