

高等学校计算机专业教材精选·计算机硬件

计算机硬件技术基础 学习指南

颜志英 颜星悦 编著

清华大学出版社

高等学校计算机专业教材精选·计算机硬件

计算机硬件技术基础 学习指南

颜志英 颜星悦 编著

清华大学出版社
北京

内 容 简 介

本书是浙江省精品课程建设项目和浙江工业大学重点教材建设项目中的重要内容之一。本书的主要目的是帮助学生计算机硬件课程,内容包括微型计算机系统的组成、寻址方式、指令系统、系统功能调用、汇编语言程序设计、存储器组织与设计、输入输出接口与中断,以及一些常用的可编程接口芯片等。为读者提供了各部分内容的重点与难点讲解与分析,以及大量的例题精选,并给出了详细的解答。本书便于读者的学习和课后复习,也便于教师备课。

本书适合高等院校理工科的计算机科学与技术、电子、信息、自动化等各类专业的本科、专科、自学考试的学生使用,也可以为工程技术人员提供学习和参考,也适合初学者自学使用。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

计算机硬件技术基础学习指南/颜志英,颜星悦编著.--北京:清华大学出版社,2013.4

高等学校计算机专业教材精选·计算机硬件

ISBN 978-7-302-31342-7

I. ①计… II. ①颜… ②颜… III. ①硬件—高等学校—教学参考资料 IV. ①TP303

中国版本图书馆 CIP 数据核字(2013)第 012423 号

责任编辑:白立军 徐跃进

封面设计:傅瑞学

责任校对:焦丽丽

责任印制:杨 艳

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课件下载: <http://www.tup.com.cn>, 010-62795954

印 装 者:北京嘉实印刷有限公司

经 销:全国新华书店

开 本:185mm×260mm

印 张:10.25

字 数:248 千字

版 次:2013 年 4 月第 1 版

印 次:2013 年 4 月第 1 次印刷

印 数:1~3000

定 价:19.50 元

产品编号:043912-01

目 录

第 1 章 微型计算机系统的组成与结构	1
1.1 重点与难点讲解	1
1.1.1 微型计算机系统的组成	1
1.1.2 8086CPU 的寄存器组	2
1.1.3 微型计算机系统的主要性能指标	3
1.2 例题精选	4
第 2 章 汇编语言程序设计基础	8
2.1 重点与难点讲解	8
2.1.1 寻址方式	8
2.1.2 指令系统	9
2.1.3 汇编语言程序格式	12
2.1.4 汇编语言程序设计	15
2.2 有关寻址方式的例题精选	16
2.3 有关指令系统的例题精选	21
第 3 章 BIOS 和 DOS 系统功能调用	63
3.1 重点与难点讲解	63
3.2 例题精选	63
第 4 章 存储器组织结构	69
4.1 重点与难点讲解	69
4.1.1 有关存储器的概念	69
4.1.2 半导体存储器的性能指标	70
4.1.3 半导体存储器芯片的主要外部引脚	71
4.2 例题精选	72
第 5 章 输入输出接口与中断	77
5.1 重点与难点讲解	77
5.1.1 接口的基本概念	77
5.1.2 接口的基本功能	78
5.1.3 CPU 与外设之间的数据传送控制方式	78

5.1.4	输入输出端口的编址方式	80
5.1.5	中断的基本概念	80
5.1.6	中断优先级与中断嵌套	82
5.1.7	中断向量表	82
5.1.8	中断处理的过程	83
5.1.9	中断服务程序设计	85
5.2	例题精选	87
第 6 章	接口技术及其应用	95
6.1	重点与难点讲解	95
6.1.1	定时/计数控制技术	95
6.1.2	并行接口技术	97
6.1.3	串行接口技术	103
6.1.4	模拟量输入输出接口	107
6.2	例题精选	108
第 7 章	计算机总线技术	131
7.1	重点与难点讲解	131
7.1.1	计算机总线概述	131
7.1.2	系统总线	133
7.1.3	PCI 局部总线	134
7.1.4	PCI Express 总线	134
7.1.5	外部总线	135
7.2	例题精选	135
第 8 章	汇编语言程序设计的上机指导	136
8.1	上机环境	136
8.2	调试程序	137
8.2.1	动态调试程序 DEBUG 常用的命令	137
8.2.2	程序段的调试方法和举例	144
附录 A	标准 ASCII 码表	147
附录 B	PC 中断表	148
附录 C	DOS 功能调用 (INT 21H 的功能) [AH 中存放功能号]	150
附录 D	BIOS 功能调用	153
参考文献		156

第 1 章 微型计算机系统的组成与结构

1.1 重点与难点讲解

1.1.1 微型计算机系统的组成

计算机系统通常由硬件系统和软件系统两大部分组成。计算机的硬件系统和软件系统有着同样重要的作用,没有软件的硬件系统称为“裸机”,裸机是不能工作的。软件和硬件的结合才能构成一个完整的计算机系统,二者缺一不可。

由于微电子技术和计算机技术的不断发展,软件和硬件在相互转化,有些硬件的功能可以由软件实现,而有些软件的功能也可以由硬件实现,例如,现代操作系统中的一些重要的功能模块都已经固化成硬件了。

计算机软件系统是计算机系统的重要组成部分。软件系统是指为运行、管理、应用、维护计算机所编制的所有程序及文档的总和。依据功能的不同,软件系统包括系统软件和应用软件及其程序设计语言。其中,系统软件是面向计算机系统的,是由计算机的生产厂家提供给用户的一组程序,是为了使用和管理计算机所必需的软件。

软件系统的基本功能是保证计算机硬件的功能得以充分发挥,并为用户提供一个使用计算机的工作环境。

微型计算机的硬件系统的基本功能是接受计算机程序,并在程序的控制下完成数据输入、数据处理和输出结果等任务。

微型计算机的硬件系统主要由四大模块构成:微处理器系统(CPU)、主存储器系统、输入输出系统和总线。

微处理器是整个计算机系统的核心,微处理器芯片内集成了控制器、运算器、寄存器等。微处理器及其支持电路共同构成了计算机系统的控制中心,对系统的各个部件进行统一的协调和控制。

存储器是用来存放程序和数据部件。按存取信息的特点可以分为随机存取存储器(读写存储器)(Random Access Memory, RAM)和只读存储器(Read Only Memory, ROM)两大类。

输入输出设备是指计算机上配备的外部设备,简称外设,其功能是为计算机提供具体的输入输出手段。而输入输出接口电路用以完成信号变换、数据缓冲、与 CPU 的联络等工作。计算机系统中,输入输出接口电路一般是制作在电路插板上,俗称“卡”,只要将它们插入 I/O 总线插槽上就是把它们连接到了系统中。微型计算机的主板上一般会安排多个 I/O 总线插槽,供用户连接各种插卡和外设。

总线是模块之间传送信息的一组公用通道。CPU 通过总线与主存储器系统、输入输出系统交换信息。

时钟用于将 CPU 的内部操作与其他系统部件之间的同步。

1.1.2 8086CPU 的寄存器组

8086CPU 的寄存器组,包括一组通用寄存器组和专用寄存器。通用寄存器用于暂存参加运算的一个操作数,例如数据寄存器可以用来存放二进制操作数。这些操作数可以是参加操作的数据,操作的中间结果,也可以是操作数的地址,大部分算术和逻辑运算指令都可以使用这些数据寄存器。专用寄存器通常有指令指针 IP 或程序计数器 PC、标志寄存器、段寄存器等,专用寄存器功能比较单一。

8086CPU 内部包含有 8 个 16 位通用数据寄存器: AX、BX、CX、DX、SP、BP、SI、DI,每个寄存器都有它们各自专用的目的。其中的 AX、BX、CX、DX 寄存器又可以分别当做两个 8 位的通用寄存器使用。

通用数据寄存器可以存放操作数或运算结果,在指令系统中,这些数据寄存器又有它们隐含的使用规定:

AX 作为累加器运用在字乘法、字除法和字 I/O 中;而 AL 运用在字节乘法、字节除法和字节 I/O、十进制运算和查表转换中。在 I/O 指令中,AX 和 AL 分别存放 16 位和 8 位的传送数据。

BX 作为基址寄存器,主要用于查表转换和间接寻址时存放基址。

CX 作为计数寄存器,用于串操作和循环指令中的计数器。

DX 作为数据寄存器,在 I/O 间接端口寻址中存放 I/O 端口地址。在双字乘、除法运算中,DX 与 AX 一起存放一个 32 位的双字数据,其中,DX 存放高 16 位,AX 存放低 16 位。

通用寄存器中与偏移地址有关的寄存器是地址指针 SP、BP 和变址寄存器 SI、DI。其中,指针寄存器 SP、BP 用来存放当前堆栈段中的数据,但它们在使用上是有区别的,入栈(PUSH)和出栈(POP)操作时 SP 指出堆栈栈顶的偏移地址,该地址表明栈顶离堆栈段首地址的偏移量,所以 SP 称为堆栈指针(Stack Point)。而 BP 则表示数据在堆栈段中的基地址,以便访问堆栈中的其他数据,BP 称为基址指针寄存器(Base Point),主要用在子程序设计中,利用堆栈来传递参数。

变址寄存器 SI、DI 用来存放当前数据段的偏移地址,在串操作指令中,源操作数的偏移地址存放在 SI 中,所以 SI 寄存器称为源变址寄存器(Source Index);目的操作数的偏移地址存放在 DI 中,所以 DI 寄存器称为目的变址寄存器(Destination Index),此时 SI 和 DS 联用,DI 和 ES 联用,可以分别在数据段和附加段中寻址。

对 80386 以后的系统,通用寄存器是 32 位的,包括 EAX、EBX、ECX、EDX、ESP、EBP、EDI、ESI。它们可以用来保存不同宽度的数据,如可以用 EAX 寄存器保存 32 位数据,用 AX 寄存器保存 16 位数据,用 AL 或 AH 寄存器保存 8 位数据。

8086CPU 内部还包含有 4 个 16 位的段寄存器: CS、DS、SS、ES,分别指明代码段、数据段、堆栈段、附加段的首地址。

代码段寄存器(Code Segment,CS),用来存放代码段的段地址,CS 寄存器与指令指针寄存器 IP 一起构成 CPU 将要执行的下一条指令的物理地址 CS:IP。代码段是微处理器用来存放程序代码的一段存储区。

数据段寄存器(Data Segment,DS),用来存放当前使用的数据段的段地址,DS 寄存器与指令中的存储器操作数的偏移地址一起构成数据段中某一单元的物理地址。数据段用来

存放当前程序所用的数据。

堆栈段寄存器(Stack Segment,SS),用来存放当前堆栈段的段地址,SS寄存器与堆栈指针寄存器SP一起构成堆栈段中某一单元的物理地址(SS:SP)。堆栈是按先进后出的原则规定的一段特殊的存储区。

附加段寄存器(Extra Segment,ES),用来存放附加段的段地址,附加段通常也用来存放数据。在串操作指令中,附加段作为目的操作数的存储区。

8086CPU内部还包含一个标志寄存器FLAGS、一个指令指针寄存器IP,它们都是16位的寄存器。其中,IP寄存器存放着CPU将要执行的下一条指令的偏移地址,以实现代码段的访问,代码段存放着程序的指令序列,IP寄存器与CS寄存器一起用以确定CPU将要执行的下一条指令的物理地址CS:IP,以控制指令序列的执行流程。

8086的标志寄存器中有9个标志,分为6个状态标志、3个控制标志,用于反映指令执行的结果或控制指令执行的形式。

状态标志用来反映CPU执行算术和逻辑运算以后的结果特征。有进位标志CF、奇偶标志PF、辅助进位标志AF、零标志ZF、符号标志SF、溢出标志OF。

控制标志是用来控制CPU的工作方式或工作状态,它由程序根据需要用指令设置或由指令清除,用于控制处理器执行指令的方式。有单步操作标志TF、中断允许标志IF、方向标志DF。

1.1.3 微型计算机系统的主要性能指标

下面是微型计算机系统的主要性能指标。

1. 主频

主频就是CPU的时钟频率,主频越高,CPU的速度越快,主频的单位是兆赫兹(MHz),目前CPU的主频都在1GHz(1024MHz)以上,有的已经超过了3GHz。主频决定了计算机的运算速度。

2. 运行速度

运行速度的单位是MIPS(百万条指令/秒)。因为执行不同的指令所需要的时间不同,以前是以执行定点加法指令作标准来计算运算速度,现在用平均速度来衡量,影响机器运算速度的因素很多,主要是CPU的主频和存储器的存取周期。

3. 字长

字长就是计算机内部一次能够同时处理的二进制数据的位数,计算机的字长是由它的内存、通用寄存器、运算器的位数和数据总线的宽度决定的。字长越宽,计算机的运算精度就越高,计算机的运算速度就越快。字长也决定了指令直接寻址的能力。然而,字长越宽,计算机的硬件成本也越大。为了兼顾精度、速度与硬件的成本,有的计算机允许采用变字长运算。一般说来,机器的字长是字节的1、2、4、8倍,例如,机器的字长是8位、16位、32位、64位等。

4. 存储容量

计算机的主存储器中能够存储的信息的总字节数称为存储容量,人们通常把一个8位(bit)的二进制数称为一个字节(byte)。存储容量的单位通常有KB(1024B)、MB(1024KB)、GB(1024MB)等。存储容量是人们在购买计算机时关注的一个重要指标。

5. 存取周期

存储器进行一次读或写操作所需要的时间称为存储器的访问时间或读写时间,连续启动两次独立的读或写操作所需要的最短时间,称为存取周期。目前,微型计算机的存取周期约为几十纳秒。

6. 外设扩展能力

外设扩展能力是指计算机系统配接各种外部设备的灵活性和适应性。一台计算机允许配接多少外部设备,对于系统接口和软件开发都会有影响。在计算机系统中,外设配置需要考虑的问题有很多,包括打印机的型号、显示器的分辨率、外存储器的容量等。

7. 软件的配置

软件是计算机系统中的一个重要的组成部分,软件的配置直接影响到计算机的性能和效率。软件的功能、满足应用要求的操作系统、丰富的工具软件和应用软件等,都是配置软件系统时不可忽视的问题。

8. 其他指标

其他指标包括机器的兼容性(包括数据兼容、程序兼容、系统兼容、设备兼容)、系统的可靠性(平均无故障时间 MTBF)、系统的可维护性(平均修复时间 MTTR)以及性能价格比。

1.2 例题精选

例 1.1 微型计算机的硬件系统主要由哪些部分组成?

解答: 微型计算机的硬件系统主要由四大模块构成:微处理器系统(CPU)、主存储器系统、输入输出系统和总线。

例 1.2 冯·诺依曼计算机工作方式的基本特点是什么?

解答: 按地址访问,顺序执行指令。

例 1.3 简述微处理器中的运算器、控制器、寄存器的功能。

解答: 运算器的功能主要是:①对数据进行加工处理,包括算术和逻辑运算,如加、减、乘、除、与、或、非、移位等运算与操作。②暂时存放参与运算的数据和中间结果。

控制器的功能主要是控制和指挥计算机内各功能部件协同动作,完成计算机程序的功能。控制器主要包括程序计数器(IP)、指令寄存器(IR)、指令译码器(ID)和时序信号发生器。

寄存器包括一组通用寄存器组和专用寄存器。其中,①通用寄存器用于暂存参加运算的一个操作数。这些操作数可以是参加操作的数据、操作的中间结果,也可以是操作数的地址,大部分算术和逻辑运算指令都可以使用这些数据寄存器。②专用寄存器通常有指令指针寄存器 IP 或程序计数器 PC、标志寄存器、段寄存器等,专用寄存器功能比较单一。

例 1.4 8086 CPU 中的标志寄存器有哪些标志位? 状态标志和控制标志的意义是什么?

解答: 8086 CPU 的标志寄存器是一个 16 位的寄存器,只用了 9 个标志,其中 6 个状态标志是:进位标志(Carry Flag, CF)、奇偶标志(Parity Flag, PF)、辅助进位标志(Auxiliary carry Flag, AF)、零标志(Zero Flag, ZF)、符号标志(Sign Flag, SF)、溢出标志(Overflow Flag, OF)。3 个控制标志是:陷阱标志或单步操作标志(Trap Flag, TF)、中断允许标志

(Interrupt Flag, IF)、方向标志(Direction Flag, DF)。

状态标志用来反映 CPU 执行算术和逻辑运算以后的结果特征。

控制标志是用来控制 CPU 的工作方式或工作状态,它由程序根据需要用指令设置或由指令清除,用于控制处理器执行指令的方式。

例 1.5 有两个 8 位数 8FH 和 36H 相加后,其状态标志 CF、ZF、SF、PF、OF 的状态分别是什么?

解答: 两个 8 位数 8FH 和 36H 相加后的结果是 C5H,所以状态标志 CF、ZF、SF、PF、OF 的状态分别是: CF=0(运算结果没有进位), ZF=0(运算结果不是零), SF=1(运算结果最高位为 1), PF=1(运算结果中 1 的个数是偶数), OF=0(运算结果没有溢出)。

例 1.6 可以为存储器操作数提供偏移地址的寄存器组有哪些?

解答: 可以为存储器操作数提供偏移地址的寄存器组有 BX、BP、SI、DI。

例 1.7 一个 16 位有符号数的补码所表示的十进制数的范围是什么?

解答: $-32\,768 \sim +32\,767$

例 1.8 简述存储器系统的构成。

解答: 存储器系统采用分级结构来组织,包括 CPU 内部的寄存器、高速缓存器(cache)、主存储器、外存储器。

例 1.9 什么是 RAM 和 ROM?

解答: RAM 可以被 CPU 随机地读或写,所以 RAM 又称为读写存储器。这种存储器用于存放用户装入的程序、数据、一些系统信息。机器断电后原存储的信息都会丢失。

ROM 是只读存储器,其中的信息只能被 CPU 随机读取,而不能由 CPU 任意写入。机器断电后,其中的信息并不丢失。这种存储器主要用于存放各种固定的程序,如汇编程序、各种高级语言的解释或编译程序、监控程序、基本输入输出程序等,以及存放各种常用的数据和表格等。

例 1.10 如果一个 8KB 的 BIOS 程序放在以 FE000H 为起始地址的 ROM 存储区内,其末地址是什么?

解答: 末地址为 FFFFFH。

例 1.11 什么是输入输出系统?

解答: 输入输出系统包括输入输出接口电路和外部设备,是计算机与外部世界交换信息所必需的手段。

例 1.12 为什么计算机与外设之间,必须设置一种接口电路?

解答: 相对于 CPU 而言,外设的工作速度是比较慢的,外设能处理的信息是非常复杂的,有数字量,也有模拟量,外设的信息传送方式有串行方式,也有并行方式。

正是由于输入输出设备的数据形式、数据的传递方式以及传递速率的差异很大,因此,在计算机与外设之间,必须设置一种接口电路,以完成信号转换、数据缓冲、接受并解释和执行 CPU 的命令,使 CPU 与外设之间能协调工作,有效地完成 CPU 与外设的信息交换。

例 1.13 什么是端口地址?

解答: 计算机系统给每个端口分配一个地址,称为端口地址或端口号,与内存单元地址一样,端口地址具有唯一性。

例 1.14 IBM-PC 系统中,参与端口寻址的往往只有多少位地址线? 端口寻址空间是

多少?

解答: IBM-PC 系统中,参与端口寻址的只有 10 位地址,所以最多只有 $2^{10}=1024(1K)$ 个端口地址,端口寻址空间为 000H~3FFH。

例 1.15 什么是总线的功能? 计算机系统采用总线结构的好处是什么?

解答: 在计算机系统中,总线的功能就是完成计算机中各个部件之间的各类信息传送。系统的各个部件或模块是通过总线连接在一起的,模块之间的各种信息,如地址信号、数据信号、控制信号等是利用总线进行传送的,所以总线是计算机系统中各模块之间传送信息的通道。高性能的 CPU 必须有高性能的总线来支持,才能构成高性能的计算机系统。

计算机系统采用总线结构的好处是: ①能简化软硬件的设计。②简化了系统结构,便于系统扩展与更新。

例 1.16 什么是地址总线、数据总线和控制总线?

解答:

(1) 地址总线——传输地址信息,采用单向三态逻辑,是单向总线,它是 CPU 发出地址信息的总线,CPU 在地址总线上输出将要访问的内存单元或 I/O 端口的地址。地址总线的宽度决定了该总线的寻址能力,例如,某总线有 n 条地址线可寻址 2^n 个字节单元,16 条地址线可以访问 $2^{16}B=64KB$;20 条地址线可以访问 $2^{20}=1MB$ 。

(2) 数据总线——传输数据信息,采用双向三态逻辑,是双向总线。数据总线的宽度决定了该总线一次能传送的二进制数据的位数。而字长是计算机内部一次可以处理的二进制数据的位数,一台计算机的字长由它的通用寄存器、主存储器、ALU 的位数和数据总线的宽度决定。字长越长,数据处理的速度越高,一般情况下,CPU 内、外数据总线的宽度是一致的。

(3) 控制总线——传输控制或状态信息,用于控制数据和地址总线的访问和使用,也是 CPU 向主存储器系统或输入输出系统发出的控制命令,例如,I/O 读写信号,存储器的读写信号或中断信号等,主存储器系统或输入输出系统向 CPU 发出的状态信号、中断请求信号也是通过控制总线传送的。

例 1.17 总线的主要性能指标有哪些?

解答:

(1) 总线频率——总线的工作频率,以 MHz 为单位,工作频率越高其速率越快。

(2) 总线宽度——是指数据线的位数,以 bit(位)为单位。表明总线传输数据的能力。

(3) 总线的数据传输率——单位时间内总线上可以传输的数据总量,数据传输率的单位是 MB/s。

例 1.18 为什么地址总线是单向的,而数据总线是双向的?

解答: 地址总线是 CPU 发出地址信息的总线,CPU 在地址总线上输出将要访问的内存单元或 I/O 端口的地址,所以地址总线是单向的,它采用单向三态逻辑,是单向总线。

数据总线用来传输数据信息,采用双向三态逻辑,是双向总线。当 CPU 访问某一内存单元或 I/O 端口时,读出或写入的数据通过数据总线传送。

例 1.19 如果某微处理器有 20 条地址总线和 16 条数据总线。

(1) 若采用独立编址方式,则存储器的地址空间有多大?

(2) 数据总线上传送的有符号整数的范围有多大?

解答：

(1) 可以寻址 $2^{20}\text{B}=1\text{MB}$ 的主存储器空间。

(2) $-32\,768\sim+32\,767$ 。

例 1.20 什么是字长？

解答：字长就是计算机内部一次能够同时处理的二进制数据的位数，计算机的字长是由它的内存、通用寄存器、运算器的位数和数据总线的宽度决定的。字长越宽，计算机的运算精度就越高，计算机的运算速度就越快。字长也决定了指令直接寻址的能力。然而，字长越宽，计算机的硬件成本也越大。为了兼顾精度、速度与硬件的成本，有的计算机允许采用变字长运算。一般说来，机器的字长是字节的 1、2、4、8 倍，例如，机器的字长是 8 位、16 位、32 位、64 位等。

例 1.21 什么是存储容量？

解答：计算机的主存储器中能够存储的信息的总字节数称为存储容量，人们通常把一个 8 位(bit)的二进制数称为一个字节(byte)。存储容量的单位通常有 KB(1024B)、MB(1024KB)、GB(1024MB)等。存储容量是人们在购买计算机时关注的一个重要指标。

例 1.22 什么是中断？

解答：所谓中断(interrupt)，是指计算机具有响应不可预料事件的处理能力，就是当 CPU 正在执行程序时，某一中断源(可能来自内部中断源，也可能来自外部中断源)向 CPU 发出中断请求，CPU 暂停当前程序的执行，转移到对该中断源的服务(或称中断服务)程序，当中断服务程序运行结束后，返回原程序继续执行的过程。

例 1.23 采用 cache 技术主要解决什么问题？而采用虚拟存储技术主要解决什么问题？

解答：为了解决 CPU 与主存之间的速度匹配问题，在 CPU 和主存之间增设一个容量不大，但操作速度很高的存储器——高速缓冲存储器 cache。cache 是位于 CPU 与主存储器之间的临时存储器，它的容量比主存储器小但交换速度快，通常由 SRAM(静态随机存储器)构成，用来存放那些被 CPU 频繁使用的数据或指令。

所谓虚拟存储系统(Virtual Memory System, VMS)，就是通过软件、硬件的结合，将主存储器与辅助存储器统一成一个整体，使该存储器系统的速度接近于主存，而容量接近于辅存。这种可以将多种、多个存储设备统一管理起来，为使用者提供大容量、高数据传输性能的存储系统，就称之为虚拟存储技术。虚拟存储系统是由操作系统的存储管理软件辅助一些硬件实现的，这种硬件也称存储器管理部件(Memory Management Unit, MMU)，一般包含在 CPU 芯片中。

例 1.24 什么是指令流水线技术？

解答：为了提高计算机访问主存储器和执行指令的速度，可以使一些需要计算机处理的多项操作在时间上重叠进行或并行执行，可以设计多个功能相同或相近的功能部件对其处理，这就是流水线技术。

第2章 汇编语言程序设计基础

2.1 重点与难点讲解

汇编语言程序设计的基础是指令系统,指令系统是微处理器(CPU)所能执行的指令的集合,也是计算机提供给用户使用的一组指令集。它与微处理器有密切的联系,不同的微处理器有不同的指令系统。Intel 8086 微处理器的 16 位指令系统已经成为后来广泛使用的 80×86 微处理器的基本指令集。而 80386 微处理器在兼容 16 位指令系统的基础上,确立了 32 位指令系统的结构,是 Intel 公司作为 80×86 微处理器的指令系统的标准。

一条指令就是完成一种操作的命令,它由操作码和操作数两部分构成,用二进制数表示,操作码表示计算机所要执行的操作,操作数表示操作的对象,也就是指令执行操作的过程中所需要的数据或地址,所以操作数字段可以是操作数本身,也可以是操作数地址,或是地址的一部分,还可以是指向操作数地址的指针,或其他有关操作数的信息。

所以一条指令一定包含这样的信息:是做什么操作?操作数从哪里来?

2.1.1 寻址方式

所谓寻址方式就是寻找操作数的方式或过程,处理器设计了许多方式用来指明操作数的位置。根据操作数可能的存放位置,有各种不同的寻址方式。

一般来说,操作数的存放地址不外乎有以下几种可能:

(1) 操作数由指令直接携带。指令中的操作数就包含在指令中,称为立即数。这种寻址方式称为立即数寻址。立即数寻址只能用于源操作数的寻址,而不能用于目的操作数的寻址。因为操作数是直接从指令中取得,不需要执行总线周期,执行速度非常快。

(2) 操作数隐含在操作码中,是一种特定的单操作数指令,常常是对某一个固定的寄存器进行操作,而这个寄存器名又隐含在操作码中,称为固定寻址,这种寻址方式中操作数是被隐含在指令中的,不需要执行总线周期,执行速度快。

(3) 操作数也可以存放在 CPU 内部的寄存器中,称为寄存器操作数。指令中的操作数部分给出的是存放操作数的某个 CPU 内部的寄存器名。这种寻址方式称为寄存器寻址。寄存器寻址方式的指令编码短,无须从存储器取操作数,所以指令的执行速度快。

(4) 操作数由指令规定的某个内存单元提供,称为存储器操作数。指令中的操作数部分给出的是存放操作数的某个内存单元的地址,在执行指令时需要根据这个信息找到需要的操作数。这种寻址方式称为存储器寻址。包括存储器直接寻址、寄存器间接寻址、寄存器相对寻址、基址变址寻址、相对基址变址寻址等。

用存储器寻址的指令,其操作数一定在数据段、附加段、堆栈段的主存储器中,指令中一定包含有存储器单元的地址或产生存储器单元地址的信息。执行这类指令时,CPU 首先根据指令提供的地址信息,计算出偏移地址,用地址产生器产生能直接访问内存单元的物理地址,从内存中取得操作数,再执行指令规定的操作。对于双操作数指令的两个操作数中,只

能有一个使用存储器寻址方式。

(5) 操作数由指令规定的某个输入输出端口提供,这种寻址方式称为输入输出寻址或端口寻址。端口寻址分为直接端口寻址和间接端口寻址两种。

直接端口寻址的指令中包含 I/O 端口的直接地址,其端口地址用 00H~FFH 表示,共包含 256 个 8 位的 I/O 端口地址。

间接端口寻址的指令中包含 I/O 端口的间接地址,端口地址 ≥ 256 (16 位地址),可寻址 65 536 个端口。用于间接寻址端口的寄存器只能使用 DX 寄存器。与直接端口寻址一样,对 16 位系统,使用 AL 或 AX 寄存器分别传送 8 位或 16 位的数据;对 32 位系统,使用 EAX 传送 32 位的数据。

以上的各种寻找操作数的寻址方式是为了寻找执行指令所必需的操作数。当遇到转移指令的时候,就需要寻找下一条要执行的指令的地址。

(6) 与转移地址有关的寻址,是为了寻找下一条要执行的指令的地址在哪里,转移指令是用来改变程序的正常执行顺序的。指令中直接提供了转移地址的段地址和偏移地址,将其中的段地址送 CS 寄存器,偏移地址送 IP 寄存器,就找到了下一条将要执行的指令的物理地址。与转移地址有关的寻址,主要包括段内直接寻址、段内间接寻址、段间直接寻址、段间间接寻址。

2.1.2 指令系统

80×86 的指令系统可以分为数据传送类指令、地址传送类指令、算术运算类指令、逻辑运算类指令、串操作类指令、移位指令、控制转移类指令、处理器控制类指令等。

1. 数据传送类指令

数据传送类指令是最常使用的一类指令,它们负责把数据、地址或立即数传送到寄存器或存储单元中。数据传送类指令除标志寄存器操作指令外,其余指令均不影响标志位。

数据传送类指令包括:

- (1) 数据传送指令 MOV;
- (2) 数据交换指令 XCHG;
- (3) 堆栈操作指令 PUSH 和 POP;
- (4) 标志寄存器操作指令 LAHF、SAHF、PUSHF、POPF 等;
- (5) 输入输出指令 IN/OUT;
- (6) 换码指令 XLAT。

2. 地址传送类指令

地址传送类指令用于传送地址信息。该指令的源操作数只能是存储器操作数,指令的功能是完成把存储器操作数的地址(偏移地址或逻辑地址)送到指定的寄存器中。地址传送指令不影响标志位。

地址传送类指令包括 LEA、LDS、LES 等。

3. 算术运算类指令

算术运算类指令用来执行加减乘除以及求补和比较等操作,它们中有双操作数指令,也有单操作数指令。双操作数指令的规定与 MOV 指令一样,两个操作数中除源操作数可以是立即数的情况外,还必须有一个操作数在寄存器中,源操作数与目的操作数不能同为存储

器操作数,两个操作数的长度必须一致。单操作数指令不允许使用立即数寻址方式。

算术运算指令会根据运算结果影响状态标志位。

算术运算指令包括:

(1) 加减运算指令 ADD、ADC、SUB、SBB。

(2) 比较指令 CMP 与 SUB 指令一样,它也是执行两操作数相减,但它并不保存结果,只是根据两操作数相减后的结果影响标志位 OF、SF、ZF、AF、PF 和 CF 的状态,根据受影响的标志位状态就可以判断两个操作数比较的结果。

(3) 增量和减量指令 INC、DEC。

(4) NEG 指令的实质是做减法,其被减数是 0,减数是指定的寄存器或存储器操作数,并按减法操作影响标志;该指令执行后的结果是将正数变负数或负数变正数,但绝对值不变。是一种特殊的 SUB 指令。

(5) 乘法和除法指令 MUL、IMUL、DIV、IDIV;只指明其中一个操作数,另一个操作数隐含在 AX 或 DX、AX 中。分为有符号乘除法指令、无符号乘除法指令两种形式。乘法产生双倍长的乘积,除法的被除数必须是除数的双倍长。

(6) 十进制调整指令,计算机中常用 BCD 码表示十进制数,BCD 码表示有两种方法:压缩的 BCD 码(规定用 4 位二进制数表示一位十进制数),非压缩 BCD 码(用一个字节代表一位 BCD 码表示的十进制数,该字节的高四位为 0)。例如,十进制数 36,表示为压缩 BCD 码时为 36H;表示为非压缩 BCD 码时为 0306H,用两字节表示。专为完成十进制数运算而设置的十进制调整指令有 6 条:压缩的 BCD 码加法调整 DAA、压缩的 BCD 码减法调整 DAS、非压缩的 BCD 码加法调整 AAA、非压缩的 BCD 码减法调整 AAS、乘法后的 BCD 码调整 AAM、除法前的 BCD 码调整 AAD,这组指令在二进制计算的基础上做十进制调整,可以直接得到十进制的结果。

4. 逻辑运算类指令

逻辑运算类指令可以对字节或字数据进行按位的操作,逻辑运算指令的主要功能是:

(1) AND 指令用于对某些位清零,其余位保持不变;OR 指令用于对某些位置 1,其余位保持不变;XOR 指令使某些位取反,其余位保持不变;TEST 指令用于测试某些位而不保存结果;NOT 指令对操作数按位取反后再回送该操作数。

(2) 移位指令可以对字节或字数据进行移位操作,32 位以后的系统还可以对双字移位操作。移位指令可以改变操作数中所有位的位置,其功能是实现对寄存器或存储器操作数的移位,操作数不能用立即数寻址。

5. 串操作类指令

串操作类指令常用来完成两个存储器操作数之间的操作,实现一串数据的传送、比较、存储和扫描以及串输入、串输出等功能的操作。串操作类指令包括:

(1) 串传送指令(move string);

(2) 存串指令(store string);

(3) 取串指令(load string);

(4) 串比较指令(compare string);

(5) 串扫描指令(scan string);

(6) 串输入(input string);

(7) 串输出(output string)。

串操作类指令有以下特点:

(1) 可以对字节串、字串进行操作,32 位系统还可以对双字串操作。

(2) 源串默认在数据段,使用 SI(16 位地址)或 ESI(32 位地址)寄存器间接寻址,所以,源串的逻辑地址是 DS:SI 或 DS:ESI。而目的串默认在附加段,使用 DI(16 位地址)或 EDI(32 位地址)寄存器间接寻址,目的串的逻辑地址是 ES:DI 或 ES:EDI。

(3) 串操作时地址指针 SI(或 ESI)和 DI(或 EDI)将根据方向标志 DF 自动修改:DF=1 时 SI(或 ESI)和 DI(或 EDI)减 1(对应字节串)或减 2(对应字串)或减 4(对应双字)。DF=0 时 SI(或 ESI)和 DI(或 EDI)加 1(对应字节串)或加 2(对应字串)或加 4(对应双字类型串)。

(4) 执行串操作指令中的串传送、存串和取串指令后不影响标志位。而执行串比较和串扫描指令后将不保存结果,而是根据操作结果影响状态标志 CF、PF、AF、ZF、SF、OF。

(5) 有些串操作指令前允许加重复前缀,用以实现串操作的重复执行。

6. 移位指令

移位指令可以对字节或字数据进行移位操作,32 位以后的系统还可以对双字移位操作。移位指令可以改变操作数中所有位的位置,其功能是实现了对寄存器或存储器操作数的移位,操作数不能用立即数寻址。移位指令有以下几种。

SHL	OPR,CNT	;逻辑左移
SAL	OPR,CNT	;算术左移
SHR	OPR,CNT	;逻辑右移
SAR	OPR,CNT	;算术右移
ROL	OPR,CNT	;不带进位循环左移
ROR	OPR,CNT	;不带进位循环右移
RCL	OPR,CNT	;带进位循环左移
RCR	OPR,CNT	;带进位循环右移

其中,OPR 为寄存器或存储器操作数,不能用立即数。该指令支持 8 位、16 位、32 位寄存器或存储器操作数的左、右移位操作。8086 系统中,移位一次则移位次数 CNT 为 1,移位次数大于 1 时,CNT 预先保存在 CL 寄存器中。80286 以后的系统支持用一个立即数指定移位次数。使用 CL 或立即数指定的移位次数应小于或等于操作数的位数。

7. 控制转移类指令

一般情况下,程序是按顺序逐条执行指令的,但常常也会遇到需要改变程序的正常执行顺序。控制转移类指令就是用来控制或改变程序的执行顺序,它是通过改变 CS 和 IP 寄存器的值实现的,它主要用于实现程序的分支、循环、子程序调用等程序结构,根据转移范围的不同可以分为段内转移和段间转移,而根据转移地址的寻址方式的不同可以分为直接转移和间接转移。

段内转移是指当前指令的地址和将要转移的指令的地址都在当前代码段的 64KB 范围内,所以只要改变 IP 寄存器的值就能实现,不需要改变段地址 CS 的内容,也就是说,只要用新的转移地址代替原有的 IP 寄存器的值就能达到转移的目的。如果转移的地址可以用一个 8 位的位移量表示,称为段内短转移。因为需要满足向前或向后转移的需要,位移量是一个带符号数,也就是说段内短转移是从 -128~+127 字节范围内跳转。如果转移的地址

可以用一个 16 位的位移量表示,称为段内近转移,位移量是一个 16 位的带符号数,所以段内近转移是从-32 768 到+32 767 字节范围内跳转。

段间转移是指程序将从当前代码段转移到另一个代码段去执行,指令中直接给出转移地址的段地址和偏移地址,需要同时改变 CS 和 IP 寄存器的内容。

直接转移是指指令中直接包含了转移地址,用一个符号地址表示。分为段内直接近转移、段内直接短转移、段间直接远转移。

间接转移是指转移地址的寻址只能是寄存器或存储器寻址,即转移地址就是某一寄存器或存储单元的内容,分为段内间接近转移和段间间接远转移。

80×86 的控制转移类指令可以分为:

- (1) 无条件转移指令(JMP);
- (2) 子程序调用与返回指令(CALL 和 RET);
- (3) 有条件转移指令(JCC);
- (4) 循环控制指令(LOOP、LOOPZ、LOOPNZ);
- (5) 中断指令(INT)等。

8. 处理器控制类指令

1) 状态标志位处理指令

80×86 提供了一组用以设置或清除标志位的指令,它们只影响指令指定的标志,而不影响其他标志位。

CLC (clear carry)	;进位标志清零: CF=0
STC (set carry)	;进位标志置 1: CF=1
CMC (complement carry)	;进位标志求反
CLD (clear direction)	;方向标志置 0: DF=0
STD (set direction)	;方向标志置 1: DF=1
CLI (clear interrupt)	;中断标志置 0: IF=0, (关中断)
STI (set interrupt)	;中断标志置 1: IF=1, (开中断)

2) 其他处理器控制指令

NOP (no operation)	;空操作指令,常用来做软件延时,或取代其他指令做调试之用
HLT (halt)	;停机指令,CPU 暂停执行程序,等待硬件中断

学习指令系统时请重点关注:

- (1) 指令的汇编格式;
- (2) 指令的基本功能;
- (3) 指令支持的寻址方式;
- (4) 指令的执行对标志位的影响;
- (5) 指令的特殊要求。

2.1.3 汇编语言程序格式

利用处理器指令编写汇编语言源程序时,还需要伪指令的配合。形成一个完整的汇编语言源程序需要有基本伪指令和操作符,主要包含常量的表达、变量的定义和使用,以及组