

长

286

体系结构

386

城

汇编程序设计

艾德才编

天津大学出版社

长城286/386体系结构 与汇编程序设计

艾德才 编

天津大学出版社

内 容 提 要

本书介绍Intel 286/386体系结构,对寻址方式、数据类型、存贮器管理、总线等都做了详细的叙述。特别是对Intel 386微处理机的流水线结构、存贮器的四种管理方式、虚拟存贮器的管理与使用、与8086/8088等微处理机的兼容问题都进行了独到的论述。书中还介绍了用80286/80386微处理机汇编语言进行程序设计的方法,对伪操作、宏汇编、80287/80387协处理器的使用及汇编语言与高级语言的接口等问题,都以实例进行说明。

本书可作为32位微处理机教材,也可作为开发利用Intel 80286/80386的培训用书。可供计算机硬件、系统软件、应用软件工作人员,大专院校师生及其他科技工作者使用。

长城286/386体系结构

与

汇编程序设计

艾德才 编

*

天津大学出版社出版

(天津大学内)

河北省永清县印刷厂印刷

新华书店天津发行所发行

*

开本: 787×1092毫米^{1/16} 印张: 31 1/8 字数: 760千字

1988年5月第一版 1988年5月第一次印刷

印数: 1—15000

ISBN 7-5618-0039-8

TP·3

定价: 半精装 8.60元
平装 7.50元

前 言

计算机科学技术飞速发展,至今已经历了四十个春秋,更换了四代。对32位微处理机,众专家评说不一。有的说32位微处理机以其优良性能成为第四代计算机的代表;有的则说32位微处理机就是第五代计算机。32位微处理机是否是第五代计算机;应由实践作结论,但我赞成32位微处理机的时代已经到来的说法。32位微处理机比起8位、16位机将有较长的寿命。至少在本世纪末之前,32位微处理机会占领整个微型计算机市场。为使人们了解32位微处理机的体系结构和工作原理,为更好地开发利用32位微处理机,作者编写了本书。

本书的重点虽然是论述80386微处理机,但为满足目前国内的需要,也介绍了80286微处理机。书中详细介绍Intel 80286/80386的体系结构和工作原理,力求写出Intel 80286/80386的特色。因为设计80286/80386的宗旨之一是保持和Intel系列机向上兼容。所以本书在介绍80286/80386时注意到它们和前辈机的相同点与不同点。

本书既可作为教材,也可以作为开发利用Intel 80286/80386的培训用书。使用本书可以非常灵活,若短期培训,只花很短时间即可对80286/80386有全面了解。若欲详细了解Intel 80286/80386的体系结构和工作原理,可以详细阅读各个章节。若欲二次开发Intel 80286/80386,汇编程序设计部分则提供足够的基础知识和技能,从中可掌握汇编程序设计技巧。相比而言,本书对80286的描述虽然稍嫌简单,但非常实用。作者力求用最少的语言把80286的结构和工作原理描述清楚。

在编写中,承蒙天津大学计算机工程与科学系许镇宇教授的指导和帮助,全部书稿都经许镇宇教授仔细审阅,作者深表谢意。作者还感谢中国计算机发展公司卢明同志,他不仅给作者以热情鼓励,而且还为本书提供了十分有价值的文献资料。中国计算机发展公司市场部、中国计算机发展公司天津分公司对本书的出版发行给予了大力支持,天津大学李文彬同志为本书绘制了全部插图,作者在此一并致谢。

由于编者水平有限,时间又十分仓促,书中肯定有许多错误和不足之处,这无疑是作者的责任,敬请计算机界的老前辈、同行和读者批评指正。

编 者

1987年12月于天津大学

目 录

第一篇 80286 体系结构

概述.....	(1)
第一章 80286的基本体系结构	(4)
§1 寄存器.....	(4)
§2 寻址方式.....	(8)
§3 数据类型.....	(9)
第二章 80286系统结构.....	(11)
§1 80286 CPU.....	(13)
§2 80286物理存储器的构成.....	(14)
§3 80286的系统结构.....	(14)
第三章 80286的存储管理系统.....	(17)
§1 实地址存储管理方式.....	(17)
§2 保护虚拟地址存储管理方式.....	(18)
§3 特权.....	(29)
§4 描述符的访问和特权的确认.....	(31)
§5 保 护.....	(34)
第四章 特殊操作.....	(37)
§1 多任务处理的支持硬件.....	(37)
§2 任务转换操作.....	(37)
§3 处理机扩展上下文转换.....	(38)
§4 指针测试指令.....	(39)
§5 保护方式初始化.....	(39)
第五章 中 断.....	(41)
第六章 兼 容.....	(44)
§1 软件兼容.....	(44)
§2 兼容的实现.....	(44)
第七章 80286总线操作.....	(47)
§1 接 口.....	(47)
§2 总线操作.....	(47)
§3 $\overline{\text{READY}}$ 操作	(51)
§4 数据总线控制.....	(53)
§5 总线用法.....	(53)
§6 系统配置.....	(57)
§7 高级DMA控制器.....	(59)

第二篇 80386 体系结构

概 述.....	(67)
----------	------

第一章 80386基本体系结构	(69)
§ 1 寄存器.....	(69)
§ 2 80386寻址方式.....	(76)
§ 3 数据类型.....	(83)
第二章 80386流水线结构	(88)
§ 1 概 述.....	(88)
§ 2 流水线结构的逻辑部件.....	(90)
§ 3 80386的几个专用硬件.....	(94)
第三章 操作系统体系结构	(97)
§ 1 操作系统模型.....	(97)
§ 2 存储器管理.....	(97)
§ 3 实方式.....	(98)
§ 4 分段存储管理.....	(99)
§ 5 特权级.....	(104)
§ 6 分页存储管理.....	(112)
§ 7 80386存储管理系统的应用.....	(118)
§ 8 多任务处理和多环境.....	(120)
§ 9 保护规则.....	(121)
§ 10 分页交叉存取方式.....	(123)
第四章 虚拟存储器	(133)
§ 1 引 言.....	(133)
§ 2 虚拟存储器方案.....	(134)
§ 3 80386的虚拟存储方案.....	(134)
§ 4 虚拟存储器的作用.....	(138)
§ 5 虚拟机器.....	(139)
§ 6 为什么没有80386虚拟机?.....	(140)
第五章 软件兼容与支持兼容	(141)
§ 1 引 言.....	(141)
§ 2 与80286兼容.....	(143)
§ 3 与8086兼容.....	(143)
§ 4 支持8086和80286的软件.....	(143)
§ 5 虚拟8086环境和方式.....	(144)
第六章 80386的总线操作	(149)
§ 1 80386总线接口.....	(149)
§ 2 流水线操作和非流水线操作.....	(156)
§ 3 SRAM和高速缓冲存储器.....	(158)
§ 4 高速缓冲存储器的结构.....	(159)
§ 5 高速缓冲存储器的效率.....	(161)
§ 6 外围设备.....	(166)
§ 7 直接存储器存取.....	(167)
§ 8 I/O空间.....	(169)
§ 9 中 断.....	(170)

第七章 系统调试能力	(173)
§ 1 可测试性设计.....	(173)
§ 2 可编程逻辑阵列PLA测试.....	(174)
§ 3 控制ROM的测试.....	(175)
§ 4 TLB的测试电路.....	(175)
§ 5 对测试的支持.....	(175)
第八章 80386的指令系统	(178)
第九章 数值协同处理器	(228)
§ 1 微处理机/协处理器系统.....	(228)
§ 2 协处理器增强系统性能.....	(229)
§ 3 80287/80387协处理器的运行.....	(231)
§ 4 80287/80387的指令系统.....	(236)

第三篇 汇编程序设计

概 述	(257)
第一章 基础知识	(258)
§ 1 汇编语言的特点.....	(258)
§ 2 基本知识.....	(259)
§ 3 80386的数据类型.....	(262)
§ 4 寻址方式.....	(261)
§ 5 程序设计风格.....	(269)
§ 6 一个简单的汇编程序.....	(273)
第二章 汇编程序的运行	(275)
§ 1 引 言.....	(275)
§ 2 机器代码与汇编语言比较.....	(275)
§ 3 汇编程序的运行.....	(276)
第三章 简单的程序设计技巧	(281)
§ 1 算术运算程序.....	(281)
§ 2 逻辑运算.....	(302)
§ 3 专用表的查询.....	(305)
§ 4 用80386进行简单32位算术运算.....	(311)
§ 5 BIOS和DOS中断的使用.....	(314)
第四章 伪操作	(340)
§ 1 引 言.....	(340)
§ 2 伪操作.....	(340)
第五章 宏操作、过程和库	(360)
§ 1 宏操作.....	(360)
§ 2 过 程.....	(369)
§ 3 库.....	(378)
§ 4 几种选择的比较.....	(379)
第六章 高级程序设计技巧	(381)
§ 1 彩色屏幕绘图.....	(382)

§ 2	建立正确计量时间的程序.....	(385)
§ 3	生成简单菜单的驱动程序.....	(390)
§ 4	生成较复杂的菜单驱动接口程序.....	(393)
§ 5	高级字符串指令的应用.....	(400)
§ 6	磁盘文件的生成和应用.....	(403)
§ 7	实方式、保护方式、虚拟方式程序设计实例.....	(414)
第七章	80287/80387 协处理器的程序设计	(421)
§ 1	引 言.....	(421)
§ 2	整型数算术运算和协处理器.....	(423)
§ 3	实型数算术运算和协处理器.....	(433)
§ 4	傅里叶级数图形的绘制.....	(457)
第八章	与高级程序设计语言的接口.....	(469)
§ 1	STSC的APL.....	(469)
§ 2	Borland的Turbo PASCAL.....	(473)
§ 3	Microsoft的BASIC编译程序.....	(476)
§ 4	Microsoft的C语言编译程序.....	(479)
§ 5	IBM的FORTRAN编译程序.....	(483)
§ 6	IBM的PASCAL编译程序.....	(486)

第一篇 80286体系结构

概 述

80286是1982年推出的一种超级16位微处理机。比起它的前辈8086/8088来,它的速度要高出6倍多。它具有大批量数据、实时多任务和多道程序处理能力。80286芯片具有存储保护能力,支持操作系统和任务的分离并且可对程序和数据保密。80286最突出的优点是实现了许多用户所迫切希望的虚拟存储系统,它可以两种方式运行。一种是8086的实地址方式。另一种是保护下的虚拟地址方式。保护方式可以提供存储管理和保护手段。80286的实际存储空间为16兆字节,但每个用户任务可使用的虚拟存储空间可达一千兆字节。在运行时使用80286的存储管理能力,自动地把每项任务的一千兆字节虚拟地址空间转换成16兆字节的物理存储空间。这种保护方式也提供存储保护,以便将操作系统隔离,确保每项任务程序和数据的保密性,在实地址方式和保护虚拟方式下,80286所使用的基本指令系统相同,使用的寄存器和寻址方式也相同。采用实地址方式寻址,80286与当前8086/8088软件的目标码兼容。在采用保护虚拟地址寻址时,80386与8086/8088软件的原码兼容。为满足80286存储器的综合管理和保护的需要,还可以请求增加虚拟地址。为保证系统的高效率,80286还可提供专门的操作。例如,只需要一条指令就可以结束一项任务,并把它状态存放好,再转换成一项新任务,把新任务的状态装入,接着再启动这一项新任务。80286还可以通过所提供的段指令和一条可以再起指令支持虚拟存储系统。

80286与其它微处理机相比,有3个不同于其它微处理机的关键属性。它们是:

(1) 80286是一个高速的处理机,其系统信息流量是其它16位机系统信息流量的2~5倍。这是因为在80286内有两个其它处理机所不可及的优点。其一是,80286在技术上采用在内部结构超前处理工艺,使得80286的速度比现有的16位微处理机要快10倍之多。其二是,增加了全新的“操作系统加速器”指令,这就增加了系统软件的功能。

(2) 80286提供了16位微处理机以前从未采用的新功能。在CPU芯片内有存储管理和保护功能。这种结构与处理机的高性能是配套的。虽然对存储器中的任务或用户地址需进行隔离,但80286仍能提供非常高的系统信息流量。

(3) 80286使用起来非常容易。由于80286具有新功能和高性能,简化了编写操作系统和系统软件的工作,例如,80286仅用一条指令就可响应中断,并且启动一个新任务处理中断。

一、应 用

80286在如下3个方面应用起来具有特色。

1. 重要应用

80286有能力处理所给出的事件,或者在指定的时间内完成给定批量的事物处理。这种应

用通常是嵌入一个微处理机，象微处理机与大型计算机间的通讯和中心办公室与电话局的转接就是这种应用的例子，这是利用80286的高性能的典范。

2. 多用户又可改编程序

这种情况通常是多个用户与/或各自独立执行的任务都出自同一存储空间。这种应用通常都需要存储管理和保护。举例来说，规模为中等的多用户事物系统就要应用80286的存储保护机构来完成对用户的隔离。因为80286的存储保护机构的使用降低了10%的性能。如果增加事物系统就会增加支持用户的数量。

3. 增加了以8086为基础的系统

为8086所编写的程序可以直接在80286上运行，因为不需要修改软件，所以所增加性能为2至5倍。

二、带有关键性的几个特征

1. 非常高的性能

80286的性能比起8086/8088或其它相竞争的16位微处理机来要高出2~5倍，其原因是有许多指令，尤其是那些运算指令，执行起来要比8086快10倍。80286所提供的优良性能，与提高了以8086为基础的系统通道也是分不开的。

2. 数值运算能力强

80286可配置数值协处理器，大大增加数值计算的能力和速度。在定向应用方面，80286配置上数值协处理器比起8086/8087来性能要高出2倍之多。

除提供高性能外，80286在16位微处理机领域里与其它微型机相比还提高了更高档次 的功能。这样就有效地支持了存储管理、存储保护和CPU芯片内的虚拟存储器。

存储管理、存储保护和虚拟存储等概念，对微处理机来说并非全是新的。可事实上，在80286的CPU芯片上加进这些能力，就变成了80286CPU的一个组成部分。

之所以把存储管理和保护放到CPU内是基于如下三个原因。首先，也是最主要的，80286保护系统的性能比起把CPU与保护机构分成两个芯片来要高许多。其次，虚拟存储管理在CPU芯片内实现起来干净利落毫无拖泥带水之嫌。第三，用硬件实现起来也非常简单，而且在80286上还可以有效利用速度比较慢的存储器。

3. 存储管理和保护

在带有多用户或者多项任务的可重新编写程序的系统内，进行存储管理和存储保护是人们所希望的也是经常需要的。众所周知，在一个多用户或多项任务处理的系统内，有效地对存储器进行管理是很困难的。但是对每项任务提供保护，或者因为用户都共享同一存储空间而引起的相互干扰等原因，实现存储保护就更加困难。

80286的寻址方式还是很直观的。80286使用少量的软件就可以保护和管理着几乎近8000项任务。其中还包括对地址空间和违反访问权的访问进行验证。如果任务有越权访问的企图，就要给出标志，并且自陷进操作系统。这一优点使用户在未被保护的系统内，用付出一点性能降低的代价而换得系统的完整性。

可以在实方式下管理任务，也可以在虚拟方式下管理任务。

4. 虚拟存储

80286芯片内具有支持虚拟存储的能力，所有引起段异常的指令都可以重新启动。这样

就可以使目前不在存储器的段从辅助存储器恢复到存储器中，而且任务可以在出现异常的那一点重新启动。

虚拟存储器在需要大量存储空间的程序或大量数据的系统内特别有用。一般说来，存储器中所有的程序或数据并不是在任一给定时间内进行处理。由于减少了主存储器的大小规模，这样就把众多程序和数据都存放到辅助存储器上。此举随着很小的性能降低而显著地减少了系统成本。在处理时只需把需要的那部分程序从辅助存储器调入主存储器就可以了。

使用汇编语言进行程序设计的人们会发现，使用80286非常容易。³80286有八个非常直观的16位的通用寄存器、众多的寻址方式和优秀的指令系统。

80286还为那些计划编写自己的操作系统的人提供方便。80286为系统设计人员提供了以前微处理机从来没有的功能，这是进行系统设计的强有力的保障。80286还大大简化了所编写的操作系统。特别是在多用户或多任务处理系统内更加突出。

存储管理和保护是80286功能的一个重要组成部分，有三个关键事实为证。

(1) 仅需要一条22微秒的指令就可以完成任务切换。也就是把旧的任务参数保存起来同时启动一项新任务。

(2) 对于那些违反了段的界限、非法访问、或者不适当地使用数据的那些访问存储器的指令，都要由CPU对其进行检查。

(3) 操作系统的服务调用非常迅速，只需一条指令即可完成。

系统设计人员可以把以上所列特点，都集中到他所编写的最好的操作系统中去。80286也处理许多那些“脏”的任务。而且80286处理这类问题比起程序设计人员来更加有效。这种处理办法的优点是：

(1) 可以构造功能更强的操作系统。举例说来，可以用没有保护的同一系统在性能稍下降一点的情况下实现存储保护。

(2) 系统人员可以更多出成果。因为他们的代码处于一个比较高的级别上，这样就缩短了研制周期。

(3) 确保系统的高流量，这样就使每个用户可利用更高的处理能力，或者在每个系统内支持更多的用户。

站在应用程序设计人员的立场上来看，80286的使用也很简单。它支持多种高级语言，象PL/M、PASCAL、FORTRAN以及80286汇编语言。熟悉8086汇编语言的程序人员只要稍加训练就可掌握80286汇编。其它汇编程序人员也能很快地学会80286的汇编程序设计。

80286有效地支持应用程序设计，特别是在象PASCAL或Ada这样的高级语言上更是如此。80286的寄存器和指令系统非常适宜于编译程序代码的生成。而且所生成的代码比其它的16位微处理机所需的存储空间要少。

顺便强调一下，80286排除了事实上的对汇编程序语言中应用软件(相对系统软件而言)的需要。80286的性能级别如此之高，以至于象PASCAL这样的高级语言，在汇编程序代码内可以使用。程序设计人员用高级语言编写程序比起用汇编语言效率要高得多，使用高级语言编写程序也减少了软件研制费用和维护成本。

本篇内不叙述80286的指令系统。80286的指令系统请见第三篇。

第一章 80286的基本体系结构

80286基本体系结构包括寄存器、数据类型、寻址方式和指令系统。它形成了高级语言编译代码的生成基础，同时也是汇编语言应用程序设计的基础。

§ 1 寄 存 器

80286共有15个寄存器，按其功能可把它们分成以下几类：

通用寄存器

段寄存器

基址和变址寄存器

状态和控制寄存器

描述符表寄存器

一、通用寄存器

80286有8个16位通用寄存器，如图1.1所示。它们的结构和用法与8086几乎完全一样。

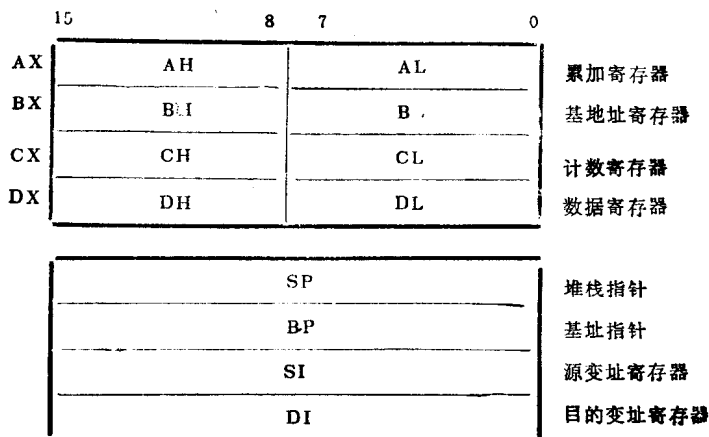


图1.1 80286的通用寄存器

通用寄存器可用于算术运算，逻辑运算。其中累加寄存器AX、基址寄存器BX、计数寄存器CX和数据寄存器DX，既可以处理16位数，也可以把它们编址成8个8位寄存器，其中高位为H，低位为L。

16位的堆栈指针和基址指针为寄存器对，在堆栈操作时才用到它们，其内保存进入当前堆栈的偏移量。通常把源变址寄存器和目的变址寄存器称为变址寄存器。在比较复杂的数据结构中，增值或减值时用SI和DI作为变址值。

二、段寄存器

80286支持同时对四个叫做段的代码模块的存取。这四个段可通过代码段寄存器CS、数据段寄存器DS、堆栈段寄存器SS、附加段寄存器ES四个16位的寄存器寻址。这四个寄存

器如图1.2所示。

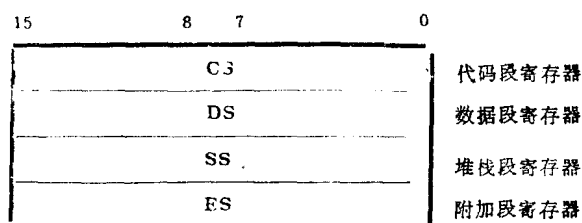


图1.2 80286段寄存器

用代码段寄存器CS对当前正在执行的驻留在存储器的程序寻址。而用数据段寄存器DS对当前数据段的基地址寻址，用来存放中间结果的堆栈和子程序的调用都必须在它们自己的存储段内给出。所以在堆栈寄存器内应该存放激活堆栈段的当前基地址。若程序设计人员还要访问第二个并行操作的附加段，就必须用附加段寄存器寻址。

在选中的段内对一个元素寻址，要首先选中这四个段寄存器中的一个当前活动段，然后从基址寄存器提供一个16位偏移量地址。把16位段地址和16位偏移量结合起来就形成32位虚拟地址指针的高半段和低半段。一旦选中了一个段，只需在指令内说明低端16位偏移量地址。

段地址的说明各不相同，这要取决于80286的操作方式。在实地址方式下，段寄存器内保存的是实际物理地址。在虚拟保护方式下，段寄存器内保存的是虚拟存储器的地址。虚拟方式下需要把逻辑地址转换成物理地址。

三、基址和变址寄存器

在80286的8个通用寄存器中，还包括四个即可通用又可专用的寄存器。它们分别是堆栈指针寄存器SP、基址指针寄存器BP、源变址寄存器SI和目的变址寄存器DI。如图1.1所示。前已谈到，所选中段内的任一给定的物理地址，都是由段址和偏移量组合而成，偏移量可以存放在任何一个指针、基址或变址寄存器内。

由于使用了堆栈段选择符SS和堆栈指针SP或者基地址指针BP寄存器对，使堆栈操作很方便。应用源变址寄存器SI和目的变址寄存器DI再连同当前激活的数据段，可对更加复杂的数据进行操作。

四、状态和控制寄存器

1. 标志寄存器FLAGS、指令指针寄存器IP和机器状态字寄存器。

这三个寄存器80286各有一个。图1.3所示的标志寄存器共使用了11个标志字段。其中六个标志字段叫状态标志。它们分别是进位标志CF、奇偶标志PF、辅助进位标志AF、零标志ZF、符号标志SF和溢出标志OF。这些标志为算术运算和逻辑控制提供必要的信息。另外三个标志分别是自陷标志TF、允许中断标志IF和方向标志DF，用来指出某些操作。当自陷标志置1时，处理机就处于单步操作方式，并且允许调试程序。当把允许中断标志IF置1时，就允许外部中断。当把IF置0时，则禁止外部中断。方向标志DF用来控制字符串的操作方向。若把DF置0，则SI或DI寄存器就自动向前增值。若把DF置1时，则SI和DI寄存器就自动地减值。

余下来的最后两个标志是I/O特权级IOPL和嵌套任务标志NT。只有当处理机处于保护方式下才可以使用这两个标志。两位的输入/输出特权级IOPL用来授权要完成的那些操作。

一位的嵌套任务标志NT用来表示是不是执行另一项任务内嵌套的当前任务。若把NT置1,则当前所嵌套的任务与前一项任务进行有效的连接。

由上可知,标志寄存器的各定义位和各字段控制某些状态并说明80286的状态。为清楚起见,下面再把各字段意义描述如下。

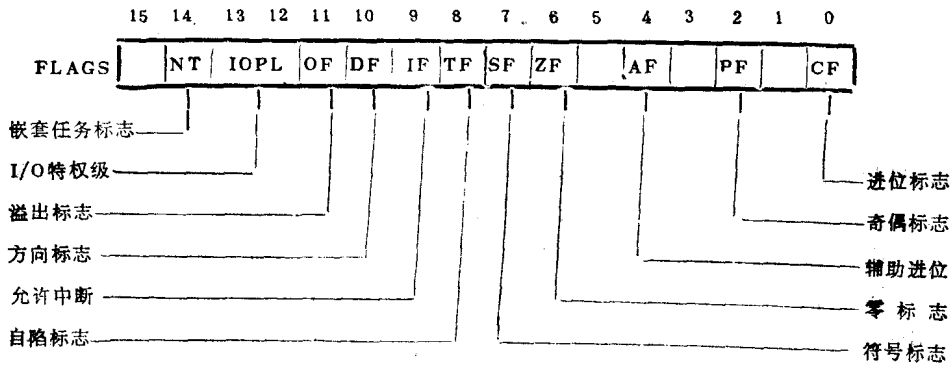


图1.3 标志寄存器

CF 进位标志, 第 0 位

如果由于操作引起向最高位有进位(加法时)或向最高位有借位(减法),则把CF置1,否则把CF清0。CF也可以在多位移和环移位指令中使用,并且保存从寄存器移出的那位。

PF 奇偶标志, 第2位

若在低8位中包含1的个数为偶数,则PF置1。若在低8位中所包含1的个数为奇数,则PF置0。PF只是检查低8位,与操作数的长度无关。

AF 辅助进位标志, 第4位

这个进位标志用于简化压缩BCD的加法和减法。若操作结果使第3位出现进位(加法)或借位到第3位(减法),则AF置1,否则AF置0。AF不顾及操作数的长度是8位还是16位。

ZF 零标志, 第6位

当结果各位都为0时ZF置1,否则置0。

SF 符号标志, 第7位

当把结果的最高位置1时,也把SF置1,否则把SF置0。对8位、16位操作来说,SF分别反映了第7、第15位的状态。

TF 自陷标志, 第8位

一旦TF置1,就在执行完下一条指令之后,发生单步中断,并且由单步中断处理程序把TF置成0。

IF 允许中断标志, 第9位

当IF位为1时,可识别INTR管脚上的外部中断信号。并把控制转移到中断向量所指定的存储单元。若IF位为0,则不识别INTR上的外部中断信号。

DF 方向标志, 第10位

当DF置成0时,则自动给指定的变址寄存器增值。当DF置1时,则自动给指定变址寄存

器减值。

OF 溢出标志, 第11位

若操作结果引起带符号溢出, 把OF置1。也就是说当计算结果对于目的操作数来说是很大的正数或很小的负数时, OF置1, 否则OF置0。

IOPL 输入输出特权级, 第12~13位

这个两位字段仅适用于保护方式。用它来说明I/O操作处于0~3级特权的哪一级。

NT 嵌套任务标志, 第14位

这个标志仅适用于保护方式, 若NT置1, 则说明该任务嵌套在另一个任务中执行。执行完该任务后, 要再返回到原来的任务中去。否则把NT置0。该值的置1和清0都是通过向其它任务的控制转移实现的。

2. 机器状态字寄存器

80286有一个16位机器状态寄存器如图1.4所示, 目前仅用了低四位, 剩下的八位由Intel保留。下面就来介绍已定义的四位。

PE 允许保护位, 第0位

用PE位激活微处理机的保护方式。如果把PE位置1, 80286就进入保护方式, 若把PE复位, 则80286在实地址方式下操作。

MP 监控协处理器, 第1位

用MP位与TS位一起确定WAIT操作码是否将要生成一个协处理器不可用的异常7, 当然此时的TS应为1。

EM 协处理器仿真, 第2位

若EM置1就会使所有的协处理器操作码生成协处理器不可用的异常7, 若EM复位, 则所有协处理器操作码都在一个实际的80287协处理器上执行。

TS 任务转换, 第3位

每当完成一次任务转换就把TS置1, 如果在TS置1时, 也把MP置1, 则协处理器的操作码将会产生一个协处理器不可用的自陷。此时自陷处理程序通常保存前一任务的80287的相关信息, 装入现行任务的80287状态, 并在返回到这个引起故障的协处理器操作码之前将TS清0。

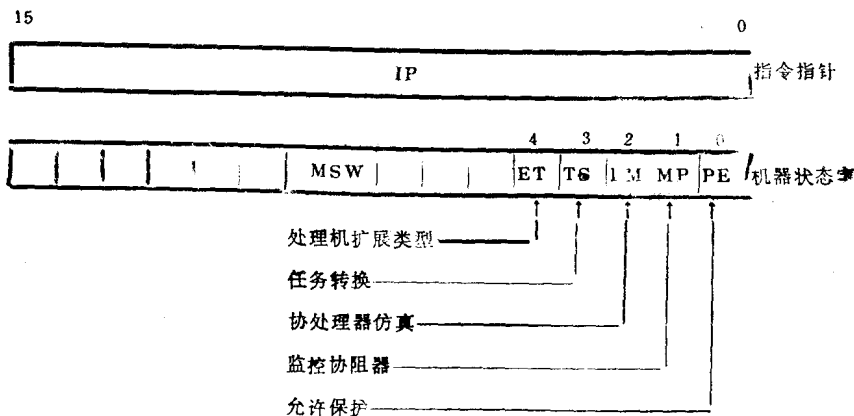


图1.4 指令指针和机器状态字

3. 指令指针寄存器IP

16位的指令指针内保存着在当前运行代码段内要执行的下一条指令地址的偏移量。

四、任务寄存器

任务寄存器用来表示当前正在执行的任务状态。当一项任务与另一项任务进行切换时，用它就可以自动地保存和恢复机器状态。它总是指着表示基地址、访问权和当前任务状态段大小的那个系统段描述符。

当切换任务时，就要使用任务寄存器。每次任务切换都要把整个机器的状态存储到由一个系统段描述符所指定的区域内。这个系统段描述符就是由任务寄存器所指示的，于是就装入任务寄存器，以便保存由它所指出的新任务区域，用时把来自这个区域的新机器状态装入。

五、描述符表寄存器

描述符表寄存器共有三个。它们是全局描述符表寄存器GDTR、局部描述符表寄存器LDT R和中断描述符表寄存器IDTR。它们总是指着包括有各种段描述符的描述符表。

§ 2 寻址方式

80286为指令指定操作数总共提供了八种寻址方式。经过优化，这些寻址方式可以有效地执行BASIC、FORTRAN、PASCAL等高级程序设计语言，可以覆盖高级语言所需要的绝大部分数据访问。

一、寄存器方式和立即方式

这种寻址方式专门处理寄存器操作数和立即操作数。

1. 寄存器寻址方式

操作数放在那8个8位的通用寄存器或8个16位的通用寄存器中。

2. 立即寻址方式

当操作数作为操作码的一个组成部分时，用此方式寻址。

二、存储器寻址方式

除上述外，还有六种寻址方式。这六种寻址方式要求说明操作数在存储器某存储段中的地址。80286寻址方式规定，存储器中操作数地址应有两个16位的分量组成，即段选择符和偏移量。段选择符是由段寄存器提供的，段寄存器既可用寻址方式隐含选择，也可以用一段优先前缀直接选择。与80286寻址方式一样，80286的地址偏移量也是分别由3个地址分量的任意一种组合求得的。这三个地址分量分别是：

(1) 位移量：跟随在指令后面的8位或16位立即数。

(2) 基地址：通用寄存器BX或基地址寄存器BP中的内容。

(3) 变址：源变址寄存器SI或目的变址寄存器DI中的内容。

在计算地址时，由16位加法产生的进位都忽略不计，8位的位移量，在运算时用符号位扩至16位。正是利用三个地址分量的不同组合，才定义了六种寻址方式。

1. 直接寻址方式

操作数偏移量以8位或16位位移量的形式并作为指令的组成部分直接包含在指令中。

2. 寄存器间接寻址方式

在这种寻址方式中，操作数在存储器中，但操作数地址的16位偏移量包含在寄存器SI、

DI、BX或BP之中。

3. 基址寻址方式

某一基址寄存器 (BX或BP) 的内容与 8 位或16位位移量相加之和构成操作数的偏移量。

4. 变址寻址方式

变址寄存器 (SI或DI) 的内容与 8 位或16位位移量相加之和, 构成了操作数的偏移量。

5. 基址变址寻址方式

一个基址寄存器的内容与一个变址寄存器的内容相加, 构成操作数的偏移量。

6. 带有位移量的基址变址方式

把一个基址寄存器的内容与一个变址寄存器的内容和一个 8 位的或16位位移量三种分量相加, 构成操作数的偏移量。

§ 3 数据类型

80286直接支持大多数高级语言具有的基本数据类型。存储器的基本存储单位是字节, 一个16位的量称为一个字。一个字占用存储器中两个相邻的字节, 低位字节放在较低的地址, 高位字节在较高的地址。所谓一个字的地址就是指的低位字节地址。现将80286的数据类型介绍如下。

一、序数

序数是不带符号的数, 是存放在一个 8 位字节内或一个16位字节中的二进制数。如果其值范围是0到65536, 就对应于一个16位字。如果它的数值小于0, 则对应于一个字或字节值。若计算结果溢出, 则产生一次异常事故。

二、指针

一个指针是一个存储器地址。它是一个32位的数, 是由一个16位的段选择符和一个16位的偏移量组合而成。

三、整数

整数是一个带符号的数, 取值范围从 - 32768到 + 32767。80286也支持字、字节的整数。在80286中使用的是带符号的 8 位或16位二进制数, 而且还规定了整数是用2的补码记数来表示。80286的数值数据处理器还支持32位或64位整数。

四、浮点数

80286加上80287浮点协处理器, 不仅扩充了指令系统, 而且可以直接支持带符号的32位、64位或88位的IEEE标准浮点算术运算。

五、ASCII

采用ASCII标准字符表示法, 来表示字母、数字和控制字符。

六、BCD

用字节来表示的二进制编码的十进制数位 (0~9)。

七、压缩的BCD

用一个压缩的字节表示两个二进制编码的十进制数位0~9, 每半个字节存放一个十进制数。