

高等学校教材

高档微型计算机

白素怀 白英彩 编著

电子工业出版社

高 等 学 校 教 材

高 档 微 型 计 算 机

白素怀 白英彩 编著

電子工業出版社

内 容 提 要

本书全面、系统地论述了高档微型计算机的系统结构及应用。全书共分十一章，分别介绍了16位微处理器的性能及结构特点、高性能32位微处理器的结构及工作方式、高档微型计算机系统、RISC技术、工程工作站、多微机系统、高档微型计算机系统软件及应用举例。

本书可作为高等院校计算机专业及相关专业高年级本科生和研究生的教材，也可作为有关工程技术人员的参考书。

高 档 微 型 计 算 机

白素怀 白英彩 编著

责任编辑 徐云鹏

电子工业出版社出版(北京市万寿路)

新华书店北京发行所发行 各地新华书店经销

山东电子工业印刷厂印刷 (淄博市周村)

开本: 787×1092毫米1/16 印张: 21.375 字数: 560千字

1991年3月第1版 1991年3月第1次印刷

印数: 2500册 定价: 5.65元

ISBN7-5053-1202-2/TP·193

前 言

本教材系按电子工业部的工科电子类专业教材1986—1990年编审出版规划,由《计算机》教材编审委员会征稿,推荐出版,责任编委为西安交通大学胡正家教授。

本教材由西安电子科技大学白素怀副教授与上海交通大学白英彩教授合编。西安电子科技大学李学干副教授担任主审。

本课程的参考时数为50至60学时。全书共分十一章。第一章概述了高性能的32位微处理器的性能及特点,第二、三章介绍了16位微处理器Intel 8086、80286的结构特点,第四、五章介绍了高性能的32位微处理器Intel 80386及MC68020的结构及性能,第六、七、八、九、十、十一章介绍了以32位微处理器为基础构成的微型计算机系统、RISC技术、工程工作站、多微机系统、高档微型计算机系统软件及应用举例。

本教材白素怀执笔编写了第一至第五章,白英彩执笔编写了第六至第十一章,最后由白素怀统编全稿。

本教材初稿完成后,主审李学干副教授对本书结构、内容及文字上提出了许多重要的宝贵意见。在编写本书过程中受到全国工科电子类计算机教材编委会的关心和指导,特别是本书责任编委胡正家教授为该书的编写和出版给予了多方面的帮助。孙德文副教授校订了后六章初稿并对全书编写提出了有益的建议,金正宜、张韧为后六章初稿的编写工作给予大力协助。在此,谨向他们表示诚挚的感谢。由于编者水平有限,书中难免还存在一些缺点和错误,殷切希望广大读者批评指正。

编著者

1990年3月

目 录

第一章 概述.....	(1)
1.1 引言	(1)
1.2 32位微处理器	(2)
一、32位微处理器芯片结构特点.....	(2)
二、典型的32位微处理器的性能比较.....	(2)
1.3 32位微型计算机	(3)
第二章 Intel 8086的结构.....	(5)
2.1 8086 CPU的结构.....	(5)
一、8086 CPU的功能结构	(5)
二、8086 的寄存器结构	(7)
2.2 8086外部引线	(9)
2.3 8086两种系统工作方式	(12)
一、最小系统工作方式.....	(12)
二、最大系统工作方式.....	(14)
2.4 8086总线时序分析	(16)
一、总线周期.....	(17)
二、最小系统工作方式读/写周期.....	(18)
三、最大系统工作方式读/写周期	(20)
2.5 8086存贮器组织	(22)
一、存贮器的结构.....	(22)
二、存贮器的段结构	(23)
三、实际地址的产生	(24)
2.6 8086的指令系统	(25)
一、8086的寻址方式.....	(25)
二、8086指令格式.....	(29)
三、8086指令系统.....	(30)
2.7 8086汇编语言及汇编语言程序设计	(45)
一、8086汇编语言语句结构.....	(45)
二、命令语句(伪指令).....	(52)
三、8086汇编语言程序设计举例.....	(60)
2.8 8086 中断系统.....	(63)
一、外部中断.....	(63)
二、内部中断.....	(65)
三、中断矢量表.....	(65)
四、单步中断.....	(66)
五、断点中断.....	(66)
第三章 Intel 80286的结构	(66)

3.1	80286 CPU 结构	(67)
一、	80286 功能结构	(67)
二、	80286 寄存器组	(67)
3.2	80286外部引线	(70)
3.3	80286指令系统	(73)
3.4	80286实地址方式	(74)
一、	存储器容量	(74)
二、	存储器寻址	(74)
三、	保留的存储单元	(74)
四、	中断	(75)
3.5	80286虚地址保护方式	(75)
一、	存储器容量	(75)
二、	存储器寻址	(76)
三、	描述符	(76)
四、	描述符表	(82)
第四章	Intel 80386 的结构	(85)
4.1	80386的内部结构	(85)
一、	80386的功能结构	(85)
二、	80386寄存器组	(86)
4.2	80386总线结构	(90)
一、	80386外部引线	(90)
二、	80386总线周期	(93)
4.3	80386 指令系统	(96)
一、	寻址方式	(96)
二、	数据类型	(97)
三、	80386指令系统	(98)
四、	80386汇编语言程序设计	(105)
4.4	实地址方式	(113)
一、	存储器寻址	(113)
二、	实地址方式下保留的存储空间	(114)
三、	中断	(114)
4.5	存储器管理	(114)
一、	存储类型	(115)
二、	分段	(115)
三、	分页	(120)
四、	线性地址到物理地址的变换机构	(123)
4.6	保护方式	(124)
一、	特权	(124)
二、	描述符访问和特权检查	(126)
三、	保护	(128)
四、	存储器保护	(129)

五、支持多任务·····	(130)
4.7 中断系统·····	(131)
一、80386中断与异常·····	(131)
二、调试与诊断·····	(132)
4.8 系统设计·····	(132)
一、80386的工作方式·····	(134)
二、软件兼容·····	(134)
第五章 MC68020系统结构 ·····	(136)
5.1 MC68020的结构·····	(136)
5.2 MC68020总线结构·····	(139)
一、MC68020外部引线·····	(139)
二、MC68020总线操作数传送机构·····	(143)
5.3 MC68020寻址方式·····	(151)
一、存贮器间接寻址·····	(152)
二、程序计数器存贮器间接寻址·····	(153)
5.4 MC68020指令系统·····	(154)
一、数据类型·····	(154)
二、指令格式·····	(155)
三、MC68020指令系统·····	(155)
5.5 处理器状态·····	(160)
一、特权状态·····	(161)
二、异常处理·····	(163)
5.6 协处理器接口·····	(163)
一、协处理器的概念·····	(163)
二、协处理器状态·····	(163)
三、协处理器操作·····	(164)
四、协处理器总线·····	(165)
五、协处理器接口寄存器·····	(166)
六、协处理器指令·····	(169)
第六章 32位微型计算机系统 ·····	(169)
6.1 概述·····	(169)
6.2 IBM PS/2系列微型个人计算机·····	(171)
一、PS/2的由来·····	(171)
二、PS/2系列机的新特征·····	(174)
三、PS/2的设计安装和运行·····	(177)
四、PS/2各型机器的比较·····	(180)
五、作为32位机的PS/2-80型机·····	(182)
6.3 IBM PS/2在系统结构上的几个特点·····	(188)
一、微通道与总线共享·····	(188)
二、磁盘高速缓冲技术与硬盘的交叉系数·····	(191)
6.4 IBM PS/2的微通道结构·····	(192)

一、什么是微通道结构	(192)
二、微通道结构与PC总线的比较	(194)
三、用户意义下的微通道结构	(195)
四、未来的微通道结构	(197)
第七章 RISC 技术	(199)
7.1 引言	(199)
一、RISC的由来	(199)
二、的RISC 设计思想	(200)
三、RISC 的特性和发展前景	(201)
7.2 RISC 寄存器管理	(204)
一、概述	(204)
二、RISC寄存器堆的结构	(205)
三、寄存器窗口的重迭技术	(205)
7.3 典型的RISC技术产品	(207)
一、RISC 的第一代产品—RISC- I	(207)
二、RISC的第二代产品—Am29000	(207)
三、RISC的第三代产品—MC88000	(208)
四、一个值得注意的RISC个人计算机系统—IBM PC/RT	(209)
7.4 一种对计算机更新换代具有重要意义的芯片 —IMS T414	
Transputer	(210)
一、概述	(210)
二、IMS T414算元	(210)
第八章 高档微型计算机工程工作站系统	(216)
8.1 工程工作站的特点	(216)
8.2 Sun 工程工作站	(217)
一、Sun-3系统的硬件结构	(217)
二、物理设备结构	(217)
三、图形系统的结构	(222)
第九章 高档微型计算机多机系统	(223)
9.1 概述	(228)
9.2 Stratus连续处理计算机系统	(228)
一、Stratus的系统结构	(232)
二、虚拟操作系统 VOS	(233)
9.3 STARLET 多机系统	(235)
一、STARLET系统的逻辑结构	(237)
二、STARLET系统的硬件结构	(238)
三、STARLET系统的软件结构	(239)
四、STARLET的结构性能和发展	(241)
第十章 高档微型计算机的系统软件	(242)
10.1 概述	(245)
10.2 32位微型机 IBM PS/2的操作系统OS/2简介	(245)

一、OS/2操作系统的设计	(246)
二、OS/2的特点及开发环境	(252)
三、新一代的程序设计语言 COBOL/2	(256)
10.3 UNIX 操作系统	(258)
一、UNIX 操作系统概述	(258)
二、UNIX 的优缺点	(260)
第十一章 高档微机应用举例	(261)
11.1 利用Transputer 实现多机系统和分布式系统的构想	(261)
一、基于32位RISC机的多处理机系统	(261)
二、利用 IMST414实现分布式系统	(262)
11.2 其它领域的应用	(265)
一、工业控制中的应用	(266)
二、CAD/CAM中的应用	(269)
附录一、8086指令系统表	(273)
附录二、80386指令系统表	(306)
附录三、MC68000 的指令系统表	(324)
附录四、MC68020 的指令系统表	(326)
参考文献	(331)

第一章 概 述

1.1 引 言

自从1971年研制出 Intel4004 微处理器以来,在短短的十几年中随着半导体工艺技术的改进,微处理器已从4位、8位、16位发展到32位。芯片集成度不断提高,存储器芯片容量不断增加,加之外围芯片大量出现,使得微型计算机得到了蓬勃的发展及广泛的应用。

目前世界主要半导体厂家竞相研制32位微处理器。32位微处理器基本上采用了32位超级小型机以及中、大型机中的某些技术。但有两种发展途径:一种是继承了16位微处理器的结构特点,某些指令格式与现在16位微处理器基本兼容;另一种是继承了32位超级小型机和中型机的结构特点,力图在软件上兼容。这两种发展途径的共同之处,是采用超大规模集成电路(VLSI)技术,让性能不低于以往的超级小型机和中型机,但价格却大大下降。

对32位微处理器硬件结构及软件有如下的要求:

硬件上的要求

字长——32位

地址总线位数——(24~32)位

数据总线位数——32位

存储器直接存取空间——(4~16)兆字节

虚拟存储空间——(400~500)兆字节

整数加/减法时间——100毫微秒以内

整数除法时间——400毫微秒以内

浮点运算时间——32位加32位 800毫微秒以内

32位乘32位 1200毫微秒以内

64位乘32位 2400毫微秒以内

存储器直接访问(DMA)——4个以上的通道

中断系统——中断向量表,硬件中断7级 软件中断1级

高速缓冲存储器——容量范围为1K字节

寄存器个数——32个字长为32位

软件上的要求

操作系统具有通用性,能适应多种处理器指令系统,多种数据处理格式,易于安装调试和移植。

面向高级语言,可采用的高级语言尽可能多,因此需要配备的编译和解释程序尽可能多。

可使用多窗口软件。

1.2 32位微处理器

一、32位微处理器芯片结构特点

在32位微处理器中大多数都包括：高速缓存器、存贮器管理部件(MMU)、指令流水线等结构，它们的功能和执行水平几乎等价于大型机。

1. 高速缓存器(CACHE)

32位微处理器引入高速缓存器减少了对内存的访问，提高了运算速度。在高速缓存器中可存放指令/数据。若存放指令，其作用等效于指令先行缓存，从而提供指令执行的灵活性。由于缓存中的内容可以随机存取，所以指令/数据均可缓存。

目前大多数32位微处理器有带有256字节的高速缓存。Z80000的高速缓存能对指令/数据提供缓存，保证指令/数据的流水线操作；MC68020的高速缓存只对指令缓存，对数据没有缓存能力；80386设计者认为256字节高速缓存容量太小，容量至少在4K字节以上，若要把那么大容量的高速缓存做在芯片上，势必大大增加其复杂度，因此80386只提供一个16字节长度的预取指令缓冲，而宁可把高速缓存做在片外。

2. 存贮器管理部件(MMU)

32位微处理器芯片采用了小型机常用的两种存贮器管理方法：段式管理方法和请求页式管理方法。

段式管理——仅在需要时，才从磁盘上把数据或代码读进内存中。

请求页式管理——在操作系统控制下，确定或修改代码/数据时，由MMU部件进行地址分配。MC68020和NS32032 CPU芯片与MMU部件是分离的，而80386是将MMU部件制作在CPU芯片内。

3. 指令流水线

高速缓存器减少了读内存操作数时间，使微处理器可重迭工作。目前微处理器芯片体系有不少采用流水线特性的处理单元，如算术运算流水线，大大加速了运算速度。流水线的复杂程度取决于流水线级数，通常32位微处理器都具有3~6级流水线控制。Z80000有6级流水线，包括取指令、指令译码、计算地址、取操作数、执行和接收操作数等允许6条指令同时在流水线中处理，采用同步控制，每一级流水线操作，只需一个总线周期(2个时钟周期)。MC68020在取指令和指令译码部分采用3级流水线，它有3个32位，ALU，使计算指令地址、操作数地址及数据运算并行操作。80386有4级流水线，包括总线接口、指令译码、执行和地址变换。

二、典型的32位微处理器的性能比较

典型的32位微处理器有：美国国家半导体公司(NS)的NS32032、Motorola公司的MC68020、Zilog公司的Z80000、Intel公司的80386。下面以MC68020和80386为代表，将其性能作一简单介绍。

MC68020片内具有指令高速缓存器，采用3级流水线操作，数据总线宽度动态可调，具有为提高执行速度的桶形移位器，有3个ALU可提供并行处理能力。在MC68020上可用的高级语言有C、Pascal等。

80386 在CPU芯片上设置了存储器管理部件 (MMU), 可对虚拟存储器和物理存储器进行分页和分段管理, 并提供软件保护功能, 采用4级流水线操作, 支持多任务和 多用户, 数据总线宽度动态可调。表1-1列举了几种典型32位微处理器性能。

表1-1 几种典型32位微处理器性能

型 号	NS32032	MC68020	Z80000	80386
生产公司	美国国家半导体公司	Motorola公司	Zilog 公司	Intel 公司
半导体工艺	NMOS	NMOS/CMOS	NMOS	CHMOS I
元 件 数	约7万个	约20万个	约12万个	约27万个
封 装	64个引脚LCC 注1	114个引脚PGA 注2	68个引脚	132个引脚PGA
时钟频率	6/10MHz	12.5/16.67MHz	10~25MHz	12/16MHz
逻辑地址空间	16兆字节	4千兆字节	4千兆字节	64兆兆字节
地址/数据总线	24/32(复用)	32/32(分离)	32/32(复用)	32/32(分离)
通用寄存器	8	16	16	8
寻址方式	线 性	线 性	段式/线性	段式/线性
指令系统	128种	99种	122种	123种
存储器管理方式	分 页	分 页	分页/分段	分 段
高速缓存器	无	256字节(指令)	256字节(指令/数据)	指令高速缓存
浮点处理部件	NS32081	MC68881	Z8070	80387
存储器管理部件	NS32082	MC6885	内 含	内 含
协处理器接口	最大256个	最大8个	最大4个	不 详
指令流水线	3级	3级	6级	4级

注1:LCC—无引线载片式封装。

注2:PGA—网络阵列式封装。

1.3 32位微型计算机

32位微型计算机系统结构, 大量采用了超级小型机和大型主机的结构技术, 所以32位微型计算机可以认为是超级小型机和大型主机技术微型化的产品。32位微型计算机中采用一种新的结构概念——开放系统。即可把软件界面和硬件的内部总线接口设计成符合国际通用标准, 并向用户公开, 鼓励用户围绕该系统进行开发。

32位微型计算机另一结构特点是多处理机结构。浮点运算往往用协处理器, 而图形处理。I/O 控制、数据库管理、文件处理、阵列处理等均用专门的处理器, 通过系统总线与主处理器连接, 形成模块化结构。最近几年不少32位微处理器设计中采用 RISC (Reduced Instruction set Computer 精简指令系统计算机)技术, 使系统结构与指令趋向统一化, 更有利于支持高级语言。

32位微型计算机主要应用图象处理、工程工作站、并行处理、多用户系统、实时处理等。除了32位微型计算机本身处理能力强, 效率及速度都有很大提高外, 还与 32 位

CPU芯片配套的外围芯片有关。

32位微型计算机有丰富的外围芯片，例如存贮器管理部件、中断控制器、时钟产生器、DMA 控制器以及各种系统接口部件。此外更重要的外围芯片是浮点运算部件——协处理器。引入协处理器，对开发软件有两大优点：一是随着集成电路技术的发展，协处理器可以进入CPU芯片时，所有软件无需修改，并且运行速度更快。其次系统设计者不使用协处理器，而用仿真软件，同样可以构成系统。有了协处理器以后，大量浮点运算可用它来完成。

目前大多数32位微处理器芯片上都有存贮器管理部件MMU和ALU协处理器。例如80386具有片内MMU，而ALU为80387协处理器；MC68020的MMU为MC68887，而ALU为MC68881；NS32032的MMU为NS32082，而ALU为NS32081。

微型计算机发展趋势是将硬件、固件、系统软件和应用软件作为一个整体来进行设计。80年代微型计算机发展的突出特点是走向“统一化”和“标准化”。标准化的产品才能形成大批量生产，只有大批量的产品才能采用各种先进的设备和工艺，提高产品质量、降低产品价格。从应用角度来看，硬件产品应具有互换性和通用性，软件应具有兼容性，可降低软件产品的价格，大量已开发的软件可以继续使用，这样避免大量重复开发工作。例如 Lotus 1-2-3、DBASE、Wordstar 等几种主要支撑软件，已形成标准化得到了广大用户的认可。

在80年代末推出的NEXT微型计算机，充分利用各种微处理器的特点，其中CPU是采用MC68030，浮点运算采用MC68882协处理器，外围接口使用NCR53C90提高I/O处理速度并处理各种I/O通道，采用信号处理机DSP56001来处理语音和图象，因此NEXT微型计算机实际上是一个分布式多机处理系统。NEXT微型计算机对I/O管理十分重视，设置了12个DMA通道，包括两个以太网通道、两个磁盘通道（一个为光盘，另一个为硬磁盘），一个打印机通道、一个存贮器到DMA通道、一个DMA寄存器到存贮器通道及两个声音通道，可以看出新一代微型计算机的重点不仅在于CPU的功能，更重要的是微型计算机与外界的联系。NEXT微型计算机采用了大容量的海量存贮器，其中有一个256兆字节的光盘和一个670兆字节的硬磁盘。

新一代的微型计算机应具有超级微型计算机的运算速度，荧光屏上图文并茂，可以输入和输出声音，有巨大容量的内存与海量外存，智能化的I/O处理能力，采用UNIX操作系统。

第二章 Intel 8086的结构

Intel 8086(简称8086)是在Intel 8080/8085 的基础上发展起来的一种16 位微处理器。它能处理16位数据(具有16位的ALU和16位运算指令,以及16位的数据总线宽度),也能处理8位数据,能执行 8080/8085的全部指令。所以,在汇编语言上与 8080/8085 是兼容的,同时又增加了不少新的16 位操作指令。有20根地址总线,直接寻址能力达到1兆字节(2^{20} 字节)。

2.1 8086 CPU的结构

一、8086 CPU的功能结构

8086 CPU从功能上分成执行部件EU(Execution Unit)和总线接口部件BIU(BUS Interface Unit)两大部分,如图2-1所示。

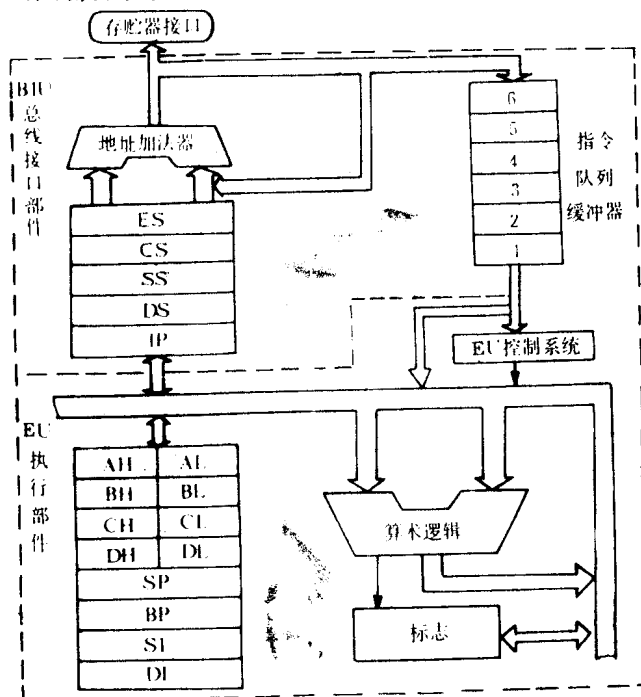


图2-1 8086 CPU功能框图

BIU由段寄存器、指令指示器、地址加法器和指令队列缓冲器等四部分组成。

BIU负责8086 CPU与存储器、I/O端口之间的数据传送。从存储器指定区域取出指令,送到指令队列缓冲器中排队,在执行指令时所需的操作数,由BIU从存储器指定区域取出,传送给EU去执行。

EU由ALU(运算器)、通用寄存器、标志寄存器等三部分组成。

EU负责指令的执行。向BIU提供数据和地址，并对通用寄存器和标志寄存器进行管理。

当EU准备执行指令时，从BIU的指令队列缓冲器中取出头一个字节的指令目标代码，如果指令队列缓冲器是空的，则EU就处于等待取指令状况。在指令执行过程中，如果需要访问存储器或I/O端口时，EU就请求BIU进入访问存储器或I/O端口的总线周期。

在CPU中，EU和BIU的操作是互相独立的。当指令队列缓冲器里6个字节中，有两个以上字节是空的，并且EU也没有要求BIU进入总线周期时，BIU就执行取指令周期，把指令队列缓冲器填满。取指令部分与执行指令部分是分开控制的，这样在一条指令执行过程中，就可以预取出下一条(或多条)指令，在指令队列缓冲器中排队等待。当一条指令执行完后，可以立即执行下一条指令，以减少CPU为取指令而等待的时间，提高了CPU的利用率和运行速度。

在8080/8085以及标准8位微处理器中，程序的执行是由取指令和执行指令交替完成的。执行的顺序为取第一条指令，执行第一条指令；取第二条指令，执行第二条指令；……直至取最后一条指令，执行最后一条指令。在每条指令执行完以后，CPU必须等待，直到下一条指令取出来以后才能执行。执行顺序如图2-2所示。

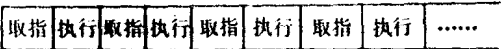


图2-2 8080/8085程序执行顺序

在8086中，由于BIU和EU是分开的，所以取指令和执行指令可以重叠。执行顺序如图2-3所示。这样可以大大减少了等待取指令所需的时间，提高了CPU的利用率，同时

也降低了对存储器存取速度的要求。这种重叠技术，过去只在大型计算机中才使用。

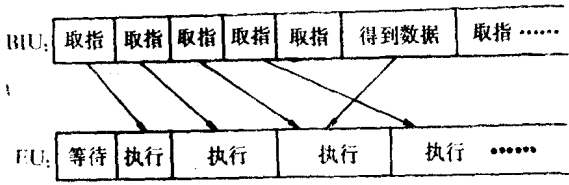


图2-3 8086程序执行顺序

下面举个例子，来说明8080/8085取指令和执行指令采用交替工作方式与8086取指令和执行指令采用重叠工作方式的差别。这两种微处理器在执行

同样三条指令时的流程如图2-4所示。

8080/8085取指令和执行指令采用交替工作方式。在第一个时间单元CPU执行第一条指令，总线不工作。在第二个时间单元CPU利用总线将第一条指令执行的结果写到存储器中，此时总线“忙”，处于工作状态。在向存储器写结果时，CPU中完成指令执行的那部分电路不工作。在第一条指令执行结果写到存储器以后，在第三个时间单元CPU才能利用总线从存储器中读第二条指令。当第二条指令取出来之后，总线就空闲起来，CPU中指令执行的那部分电路开始工作。假设，这条指令不需要写操作，指令执行完毕后，CPU开始经由总线读取第三条指令，此时总线“忙”。如果第三条指令要求从存储器中读取一个操作数，CPU利用总线读取操作数。然后，CPU执行第三条指令，这时总线又空闲了。从图2-4中看出，执行上述三条指令，在8080/8085CPU中，要用7个单元时间，而总线

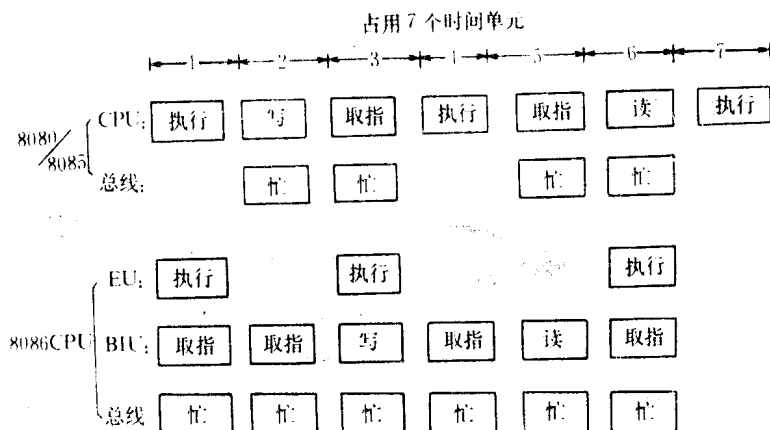


图2-4 取指令和执行指令采用交替与重叠两种不同工作方式的流程

在其中4个单元时间中工作，因此总线并没有得到充分的利用，相应也就延长了指令总的执行时间。

8086取指令和执行指令采用重叠工作方式。在第一个时间单元里EU执行已经取得的第一条指令的同时，BIU经由总线取第二条指令，总线处于“忙”。第二个时间单元BIU就可取第三条指令。第三个时间单元EU执行第二条指令，同时BIU经由总线把第一条指令执行结果写到存储器中，故总线仍处于“忙”状态。接着第四个时间单元BIU就可以取第四条指令了，第五个时间单元根据第二条指令的要求，EU请求BIU使用总线，读取第二条指令所需的操作数，第六个时间单元EU就可以执行第三条指令，而BIU同时去取第五条指令。从图2-4可以看出，8086 CPU执行同样三条指令的过程中，总线一直处于“忙”状态，总线得到充分的利用。只用了6个时间单元，便可执行这三条指令，同时还将其后面第四条和第五条指令取到BIU指令队列缓冲器中排队，从而减少了总的指令执行时间。

二、8086的寄存器结构

8086的寄存器结构如图2-5所示。通用寄存器组由AX、BX、CX、DX 4个16位数据寄存器组成。其中，AX为累加器，其它3个为16位通用数据寄存器。它们的用途可用表2-1来说明。

8086也能处理8位数，图2-5中的4个16位通用寄存器也可作为8个8位寄存器使用，图中打斜线部分相当于8080/8085中

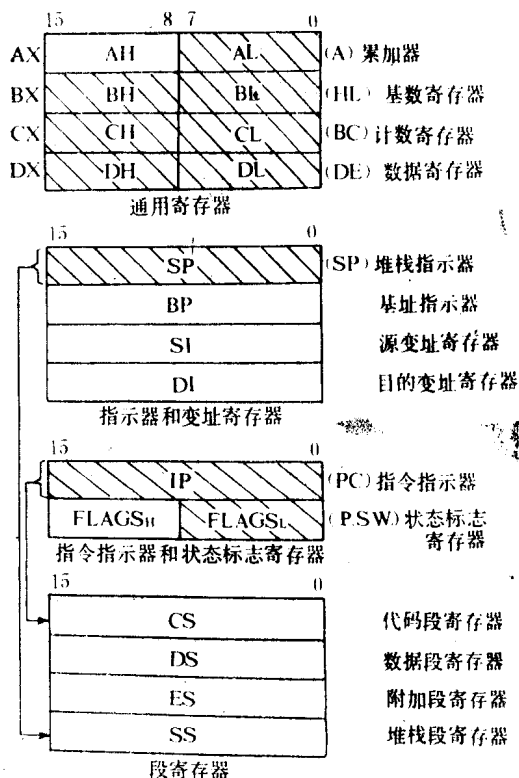


图2-5 8086寄存器结构

表2-1 通用寄存器的用途

寄 存 器	操 作
AX	字乘法, 字除法, 字 I/O(输入/输出)
AL	字节乘法, 字节除法, 字节 I/O, 转移, 十进制算术运算
AH	字节乘法, 字节除法
BX	转移
CX	在循环或字符串操作中, 作循环次数计数器
CL	在移位或循环移位操作中, 作移位位数或循环次数计数器
DX	字乘法, 字除法, I/O间接寻址时作地址寄存器

的通用寄存器, 图右侧括号内的寄存器均为8080/8085的8位通用寄存器, 其中 A为累加器, 其余均为通用数据寄存器。

指示器和变址寄存器组由 SP、BP、SI、DI 组成。其中 SP、BP 为指示器, SI、DI 为变址寄存器。在8086中的堆栈指示器SP类似于8080/8085中的堆栈指示器, 用于确定堆栈操作时, 确定堆栈在内存中的位置。但在8086中, SP还必须与 SS 堆栈段寄存器一起才能确定堆栈的实际位置。基址指示器 BP, 用来存放在堆栈段中的一个数据区的“基址”偏移量。变址寄存器SI存放源操作数偏移地址, 故称源变址寄存器; 变址寄存器DI存放目的操作数偏移地址, 故称目的变址寄存器。

段寄存器组由CS、DS、SS、ES 4个16位段寄存器组成。CS为代码段寄存器, 指向当前程序的代码段的起点, 取指令就靠它进行寻址。DS 为数据段寄存器, 指向当前使用的数据段起点, 一般来说程序中的变量存放在这个段中。SS 为堆栈段寄存器, 指向当前使用的堆栈段起点, 堆栈操作时就靠它寻址。ES 为附加段寄存器, 指向当前附加段起点, 在字符串操作时使用。

指令指示器IP和状态标志寄存器。在8086中指令指示器IP类似于8080/8085 中的程序计数器PC。但是它们略有区别, 在8080/8085中的程序计数器 PC是指向下一条即将执行的指令, 而指令指示器IP是指向下一次要取出的指令, 这两者是有区别的。在8086中IP要与CS寄存器相配合, 才能形成要取出的指令的物理地址。8086中的状态标志寄存器含有9个标志位, 其中包含8080/8085的5个标志位, 又增加了4个标志位, 使状态标志寄存器增加到一个字(两个字节), 如图2-6所示。

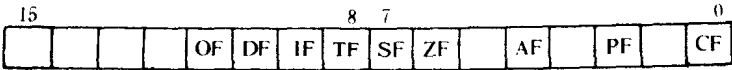


图2-6 状态标志寄存器

1. 辅助进位标志位AF

在字节操作时, 由低半字节(一个字节的低4位)向高半字节有进位或借位时, 则AF置“1”, 否则为“0”。这个标志用于十进制算术指令中。

2. 进位标志位CF

当指令操作的结果最高位(字节操作D₇或字操作时的D₁₅)产生一个进位或借位时, 则CF置“1”, 否则为“0”。这个标志主要用于多字节数的加法或减法运算。循环移位指令也能够从存储器或寄存器中抽出一位, 放入进位标志位CF中。