微计算机丛书

〔日〕中村和夫 井出裕巳 著
陆玉库译

INTEL 8086 微处理器应用入门

電子工業出版社

INTEL 8086
微处理器应用入门

〔日〕中村和夫 井出裕巳 著
陆 玉 库 译

电子工业出版社

内 容 简 介

本书主要译自〔日〕《インターフェース》杂志，1982年第8期所刊载的INTEL 8086特集；部分内容选译自〔日〕オーム社出版的《8086の使い方》。

全书包括：8086的结构、指令系统、外围芯片、硬件设计、汇编语言程序设计、多总线及开发系统等内容。书后附录有INTEL 8086的指令详解。

本书内容充实、文字简明扼要、图表丰富。对于已了解8位微计算机的读者来说，该书是学习8086微计算机的一种入门读物；并可作为8086指令参考手册。

本书可供从事16位微计算机设计和使用的科技人员及大专院校有关专业师生阅读和参考。

INTEL 8086
微处理器应用入门
〔日〕中村和夫　井出裕已著
陆　玉　库　译
※
电子工业出版社出版
中国科技情报所印刷厂印刷
新华书店北京发行所发行
各地新华书店经售
※
1983年6月第一版　　开本：850×1168 32开
1983年7月第一次印刷　印张：11 1/8
印数：1—25,000册　　字数：294千字
统一书号：15290·3　　本社书号：P31·03
定价：1.60元

译 者 的 话

自从美国INTEL 公司于1971年制造出了单片的 4 位微处理器4004以来， 微处理器已经历了十多年的发展历程。 在这十多年的过程中，微处理器不仅在各个领域里得到了广泛的应用，而且微处理器本身不断更新换代，表现了它的强大生命力。继 4 位、 8 位微处理器之后，INTEL公司于1978年又首先发表了 16 位 微 处 理 器 8086，此后ZILOG公司、MOTOROLA 公司又相继发表了16位微处理器Z8000 (1979)、MC68000(1980)，使微处理器应用进 入 了一个新阶段。由于16位微处理器不仅具备以前微处理器所有优点，还吸取了不少小型计算机结构设计上的特点，从而使它的处理能力进一步加强，使微型机和小型机差别日趋减小，并有取代小型机的趋势。

目前，我国在广泛应用 4 位和 8 位微计算机的基础上已开始重视16位微机的研制开发和应用。但是在实际工作中深感资料不足。译者根据当前国内微机发展形势的需要，编译了《Intel8086微处理器应用入门》一书。如果本书能对于从事微机研制和应用的技术人员有所启发和帮助，为我国微机的发展起到点滴作用的话，将是译者的最大荣幸。

本书 1～6 章译自日文杂志《インターフェース》1982 年第 8 期； 7 、 8 、 9 章选译自〔日〕井出裕巳著《8086の使い方》一书。

北京工业大学计算中心李礼贤副教授对译文作了校订，并提供了附录资料，在此谨表谢意。

由于译者水平有限，缺乏实践经验，不妥之处，谨請读者批评指正。

1983.4 .

前　言

本书是以打算采用8086微处理器的人为对象，以8086的结构、指令系统、硬件设计、汇编语言程序设计及多路总线等在系统设计中应注意的事项为主要内容，并加以详细的说明。

书中所列举的CP/M操作系统，多总线为 DIGITAL RESEACH公司和INTEL公司的专利。

中村和夫

目　录

1 8086 微处理器的结构

8086是最早出现的第三代16位微处理器，它采用了8位微处理器中所没有的许多新结构。为此，本章首先对8086的结构进行概要的说明。

1.1 8086结构概要

1.1.1 地址总线和数据总线

8086是16位微处理器，其数据总线为16位，地址总线为20位。因此，存储器可以寻址1Mb（1兆字节)的地址空间。在进行I/O（输入/输出）存取时，20位的地址线中只有低16位有效。这样，I/O地址空间为64Kb（64千字节)。图1.1为8085A与8086的数据总线和地址总线的比较，从中可以看出8086的扩充情况如下：

总　线	8085A	8086
数据总线	8位	16位
地址总线（用于存储器）	16位	20位
地址总线（用于I/O）	8位	16位

1.1.2 以字节或字为单位的数据处理

在8086的所有运算指令和传送指令中，可以按字节为单位，也可以按字为单位处理数据。例如，16位数据传送指令可以传送8位数据；带符号或不带符号的乘除运算，可按8位进行，也可以按16位进行。为此，在这些运算指令和传送指令的OP(操作）码中设置

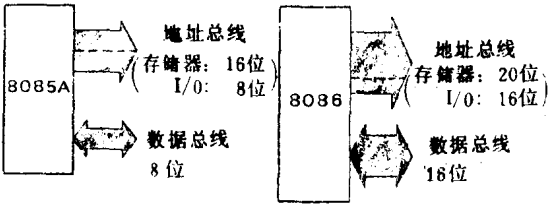

图1.1 8086与8085A的地址总线和数据总线

有W字段（段长为1位），当令W=0时为字节处理，而令W=1时为字处理。W字段在操作码中的位置如图1.2所示。在BCD（二-十进制）码数据和ASCII码（美国信息交换标准码）数据的运算时，只能按8位进行，这一点应该注意。

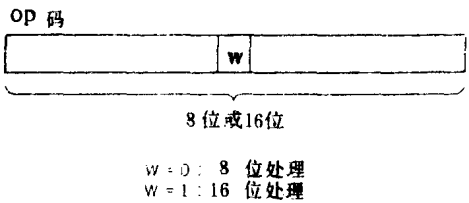

图1.2 W字段

8086中对存储器和I/O进行存取的情况可用图1.3来表示。图中（a）、（b）为字节处理的情况，其中（a）表示存取低位字节（D_7～D_0），（b）表示存取高位字节（D_{15}～D_8）。（c）为字（D_{15}～D_0）处理的情况。（d）也是一种字处理的情况，但它所存取的字跨存储器的两个地址，即包括一个地址的高8位和下一个地址的低8位的内容。8086对这样的字以一条指令进行处理时，要分两次对存储器进行存取，将两个地址分别所对应的高8位和低8位数据读出或写入。

在8086芯片单独使用时，除了乘法指令的积和除法指令的被除

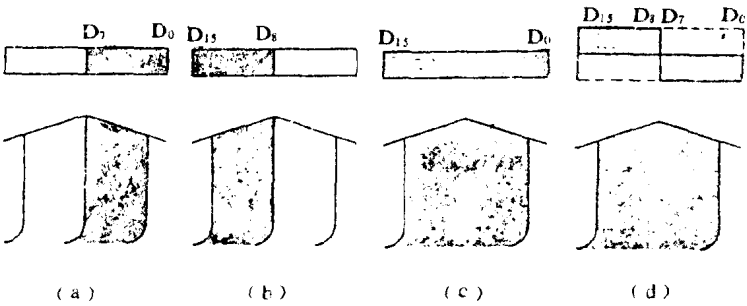

图1.3 8086数据处理的分类

数之外，都不能采用32位数据。如果采用数值运算处理器8087，不仅可以进行32位数据运算，也可以进行80位浮点运算。

1.1.3 以字节为单位的地址分配和$\overline{BHE}$信号

8086对存储器或I/O存取时，有图1.3所示的四种情况，其中(a)、(b)、(d)只能对存储器或I/O存取8位数据。因此，在写入时，必须注意不得改变其他8位数据的内容。为了解决这个问题，8086以字节为单位分配存储器和I/O的地址，并采用$\overline{BHE}$总线高8位允许信号以实现对高位和低位字节的选择。

图1.4表示存储器及I/O的地址分配情况，图中16位数据的低8位（D_7～D_0）分配在偶数地址，而高8位（D_{15}～D_8）分配在奇数地址。这样，通过20位的地址总线能够存取的存储器空间（00000_{16}～$FFFFF_{16}$）为1 Mb或512千字（字长16位）。低8位的偶数地址与8085A具有兼容性。比如在8085A上执行下述程序时，HL寄存器中装入的内容为：$H=12_{16}$，$L=34_{16}$。

```
      ORG 1234H
ADR   EQU $
        ⋮
      LXI H,ADR
```

Z 8000、68000刚好与此相反，高8位为偶数地址。

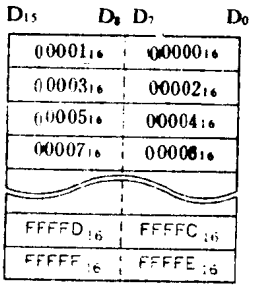

图1.4 以字节为单位的地址分配

表1.1表示8086微处理器执行存储器或I/O存取时与 A_0、$\overline{BHE}$ 信号间的关系。$\overline{BHE}$ 为表示存取 D_{15}～D_8 的信号，$\overline{BHE}$ 从8086输出为低电平有效。正如图1.5的逻辑图所示，通过地址译码器译码的片选信号与 A_0 及 $\overline{BHE}$ 信号相“与”作为相应存储器或I/O的片选信号，就可分别存取高8位（D_{15}～D_8）或低8位（D_7～D_0）数据。

表1.1 8086的存储器或I/O存取形式

A_0	$\overline{BHE}$	存取形式
0	0	存取16位（D_{15}～D_0）数据
0	1	存取低8位（D_7～D_0）数据
1	0	存取高8位（D_{15}～D_8）数据
1	1	不使用

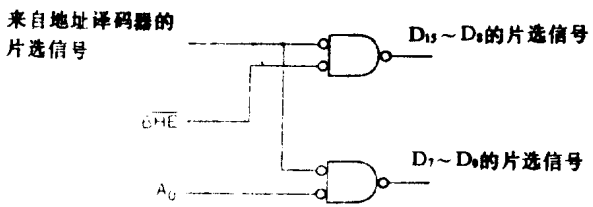

图1.5 片选信号逻辑图

1.1.4 8个16位通用寄存器

8086微处理器有8个16位通用寄存器，参阅图1.6(b)。其中AX～DX寄存器都分成高8位和低8位两部分，它们也可以分别作为独立的8位AL～DH寄存器来指定。图1.6（b）和（a）给出了8086与8085A寄存器的比较，（其中阴影部分与8085A相同）对此需要说明下述几点。

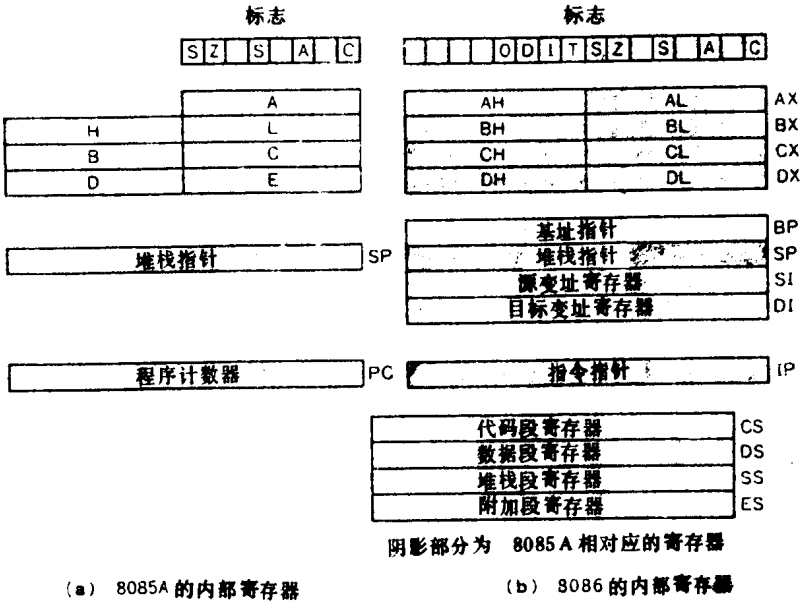

图1.6 8085A与8086内部寄存器的组成

（1）AL～DH寄存器（AH除外）对应于8085A的A～D寄存器。在8085A中只有A寄存器能作为累加器使用，而8086中AL～DH寄存器(8位)和AX～DI寄存器(16位)都可以作累加器使用。

（2）8086的BX寄存器和8085A的HL寄存器相对应。(在8086中，作为寄存器间接寻址使用的地址寄存器有BX、SI、DI寄存器。)

（3）AL～DH寄存器用于8位数据存取时，它们不再是高位字节、低位字节的关系。（在有些微处理器中，不能进行高位字节寄存器和低位字节寄存器间的传送和运算。）

（4）与8085A使用的程序计数器（PC）相对应，8086采用了名称不同的指令指针（IP），因此用监控程序调试时要注意。

上面已经说明AX～DI寄存器和AL～DH寄存器都可以作累加器使用，即可以用下述指令进行操作。

```
ADD    CX,   DX      ; CX←—CX+DX
SUB    SP,   2       ; SP←—SP-2
AND    DH,   AL      ; DH←—DH·AL
```

这些通用寄存器在有些指令中有时还起一些特殊作用。表1.2归纳了这些寄存器的作用。根据它们的特点分别规定了不同的名称，这些名称均采用了该寄存器的英文字头。例如：

AX：（Accumulator）累加器

BX：（Base Register）基址寄存器

CX：（Count Register）计数寄存器

DX：（Data Register）数据寄存器

从表1.2的说明中就可以理解这些寄存器的意义和使用方法。

表1.2 通用寄存器的特殊用法

AX，AL	• 在乘除指令中作累加器 • 在输入输出指令时，作为数据寄存器
AH	• 在LAHF指令（(AH)←(标志)）中作目的寄存器
AL	• 在BCD，ASCII码数据运算时作累加器 • 在XLAT指令（(AL)←((AL)+(BX))中作累加器
BX	• 在间接寻址时作地址寄存器 • 在间接寻址时作基址寄存器 • 在XLAT指令（(AL)←((AL)+(BX)）中作基址寄存器

续表1.2

CX	· 在循环、字符串指令中作循环次数的计数寄存器（指令执行后，CX寄存器内容发生变化）
CL	· 作为移位、循环指令的移位数、循环数的计数寄存器（指令执行后，CL寄存器的内容不变化）
DX	· 输入输出指令间接寻址时，作为地址寄存器 · 在乘除指令中作辅助累加器（当乘法的积为32位以及除法的被除数为32位时，存放高16位）
BP	· 在间接寻址时作为基址寄存器
SP	· 作堆栈指针
SI	· 在间接寻址时，作为地址寄存器 · 在间接寻址时，作变址寄存器 · 在字符串处理指令中，作源变址寄存器
DI	· 在间接寻址时，作地址寄存器 · 在间接寻址时，作变址寄存器 · 在字符串处理指令中，作目的变址寄存器

（注）此外，AX、AL寄存器作为累加器，在与立即数进行运算时，比用其他寄存器时指令短一个字节。

图1.6（b）中在指令指针（IP）的下面有4个16位寄存器，这些寄存器称为段寄存器。从图中可以看出这些寄存器比其他寄存器画得略向左错开一些，其作用将在本章的1.2中说明。

1.1.5 标志

8086比8085A新增加了3个标志位。在8085A中，标志位是与A寄存器内容一起进栈、出栈的。但在8086中，标志位的进栈、出栈是通过POPF、PUSHF指令进行的。而在中断时，它与指令指针、CS寄存器一起保留在堆栈中。

（1）进位标志CF

运算指令执行之后，当在最高位（字节运算时为D_7，字运算时