程序设计语言实用手册

主编 管会生
副主编 周虹
饶增仁

清华大学出版社

程序设计语言实用手册

主　　编：管会生
副 主 编：周　虹　饶增仁
编写人员：郭明超　李蜀娴　赵　玲
　　赵　鸿　马　俊

清华大学出版社

(京)新登字 158 号

内 容 简 介

本书汇集了微机广泛使用的程序设计语言和数据库语言。

全书共分 10 章：第一章微机汇编语言，第二章 Turbo BASIC 语言，第三章 FORTRAN 77 语言，第四章 COBOL 语言，第五章 Turbo Pascal 语言，第六章 Turbo C 语言，第七章 C++语言，第八章 GCLISP 语言，第九章 Turbo PROLOG 语言，第十章关系数据库语言。附录是各章按英文字母顺序排列的命令名、语句名和函数名索引。

图书在版编目(CIP)数据

程序设计语言实用手册/管会生主编. —北京：清华大学出版社，1994
ISBN 7-302-01630-5

Ⅰ.程… Ⅱ.管… Ⅲ.微程序设计语言-手册 Ⅳ.TP312-62

中国版本图书馆 CIP 数据核字(94)第 11031 号

出版者：清华大学出版社(北京清华大学校内，邮编 100084)
印刷者：北京密云胶印厂印刷
发行者：新华书店总店北京科技发行所
开 本：787×1092 1/16 印张：28.5 字数：712 千字
版 次：1994 年 11 月第 1 版 1994 年 11 月第 1 次印刷
书 号：ISBN 7-302-01630-5/TP・692
印 数：0001—5000
定 价：24.00 元

前　言

本书汇集了通用微型计算机上目前广泛使用的程序设计语言和数据库语言，力图体现既系统又实用两大特色。全书共分 10 章并附有各章按英文字母顺序排列的命令名、语句名和函数名索引。各章的安排及主要内容如下：

第一章：微机汇编语言，以 8086/8087 汇编指令集为主，兼顾 80x86/80x87，突出汇编语言在系统程序设计中的特点及在系统维护方面的实用性。

第二章：Turbo BASIC 语言，突出小型语言灵活实用的特点。

第三章：FORTRAN 77 语言，突出该语言在科学计算中的特点及应用。

第四章：COBOL 语言，以 COBOL 85 为标准，突出该语言在商业事务处理中的特点及应用。

第五章：Turbo Pascal 语言，以 6.0 版为主，突出结构化程序设计方法。

第六章：Turbo C 语言，突出 C 语言在应用系统和操作系统设计中的特点。

第七章：C++ 语言，突出面向对象的程序设计思想和方法。

第八章：GCLISP 语言，突出表处理语言在人工智能领域中的应用。

第九章：Turbo PROLOG 语言，以 Turbo Prolog 2.0 为主，突出逻辑程序设计的思想和方法。

第十章：关系型数据库语言，以 FoxPro 为主，兼顾 dBASE Ⅲ plus、FOXBASE 和 SQL，突出计算机在管理上的应用。

作为一册实用性较强的工具书，它适合于每一个从事计算机应用与开发的人员，既可作为程序设计初学者的编程指南与参考，又可作为软件开发和设计人员的简捷速查手册。

本书由主编管会生提出各章节的提纲和框架，并对全书进行了初审、修改和统稿。副主编周虹参加了初审并对全书文字进行了最后校订。参加各章编写工作的还有兰州大学计算中心的饶增仁、郭明超、李蜀娴、赵玲、赵鸿。马俊、徐玲承担了全部书稿的录入工作。在一年多日日夜夜的编写过程中，还得到了许多同行和朋友们的关心和帮助，在此表示衷心的感谢。

鉴于水平有限和成稿时间仓促，书中粗浅疏漏及叙述不够严密之处，恳请读者批评指正。

编　者

1994 年 3 月于兰州

使用说明

[1] 手册以 Turbo 系列语言为主，每种语言分一章，并按语言的主要功能分节、段。各章第一节“语言概要”中主要陈述该语言的创始人、时间、主要版本以及语言的基本要素。如：数据类型和说明、常量、表达式、操作符和保留字等等。

[2] 为方便程序设计人员查阅，每章各节不是按字母顺序而是按语言功能（如基本语句、I/O 语句、分支、循环、文件操作、绘图操作等）来划分和叙述的。对每一条语句、函数、命令按“功能”、“格式”、“说明”三段式进行叙述。如：

INPUT

功能：为一个或多个变量赋值。

格式：INPUT [;] [prompt string { ;|,}] variable list

说明：prompt string 是一个字符串常数。.....

[3] 多种格式的语句用“格式 1”和“格式 2”来区分。说明中通常包括格式说明、参数说明以及与其它版本的区别。为了文字叙述的方便和简洁，除各章符号约定的特别说明之外，对各章格式说明中通用的符号约定如下：

|	表示在前后两项中选择一项。
{ }	表示在大括号中的多项选择中必选一项。
[ ]	表示在方括号中的内容为可选项，即选一项或不选。
〈 〉	表示尖括号中的内容为表达式或变量名。如：〈算术表达式〉或〈自变量 2〉。
...	表示重复项，即前一项的重复。
F1—F10	表示程序功能键 1—程序功能键 10。
ENTER	表示回车键。

[4] 大写词通常用于语言中的保留关键字，而并非缩写词。如 FORTRAN 语言和 COBOL 语言就要求必须全部用大写词（某些微机版本放松了这一限制）。英文缩写词必要时作简要注释，放在圆括号内。

[5] 说明中“参见 XXXX”字样，表示需参阅 XXXX 语句的说明。两个或多个语句、函数说明部分相同时，一般只对第一个语句、函数加以说明，其余的则标出“同 XXXX”。

[6] 附录给出了按英文字母顺序排列的命令名、语句名和函数名索引。

目　录

第一章 微机汇编语言

§1.1 语言概要

1.1.1 汇编语言特点及程序书写格式

汇编语言特点

汇编语言是机器语言的一种符号表示，它用便于记忆的符号名来表示机器指令中的操作码和地址码等信息，由这些助记符组成的汇编源程序，需经汇编程序汇编（转换）成机器语言代码后才能被计算机自动解释执行。Microsoft 公司的宏汇编程序（MASM）是专为 IBM PC 系列微机及兼容机配备的，它具有丰富的宏伪指令处理及其它多种功能，已经历了多个版本的更新换代，并在继续扩充发展着，特别是 1987 年的 MASM 5.0 版本，它比以往版本更有许多增强功能，主要表现在：

- 支持 80386、80387 的所有指令和寻址模式。
- 新增的段定义伪指令，使编程更简便。
- 结构中允许有条件伪指令，使结构定义更加灵活。
- 增强了对错误信息的处理，信息的输出更具体清晰。
- 提高了汇编速度和符号空间的大小。

汇编语言比高级语言有两个显著优点：① 执行速度快，能产生最快的可执行代码。② 可以充分利用计算机的硬件软件资源。它能利用计算机的所有硬件特性，并能直接控制硬件，如对各种 I/O 接口的控制以及对存储器合理而有效的利用。目前国内流行的微机大都采用 Intel 公司的 8086 系列微处理器（CPU）。自 1971 年 Intel 公司生产出第一个微处理器 4004 到现在的 20 多年里，微处理器已更新了几代，从集成度、速度、指令性质到可直接寻址的内存空间都比以往产品有了很大提高。表 1.1 列出了 Intel 公司部分微处理器的基本进展情况。

表 1.1 Intel 系列微处理器进展

年代	型号	时钟（MHz）	内部数据位	外部数据线	最大物理内存（byte）	集成度（晶体管个数）
1978	8086	5,8,10	16	16	1 M	29000
1984	80286	6—25	16	16	16 M	10 万
1985	80386	16—40	32	32	4 G	40 万

汇编语言结构

8086 系列 CPU(8086/80286/80386)内设有 4 个段寄存器,使内存空间采用分段结构,以利于扩大寻址空间。反映在汇编语言程序中是数据与代码以及需要的堆栈都必须纳入某个段中。一个典型的汇编源程序应该包括堆栈段,数据段和代码段。一个源程序中至少应有一个代码段,数据段内容也可以放入代码段内。段又由指令或伪指令语句所组成。指令语句对应于机器指令,汇编程序都要为之产生机器指令代码;伪指令又叫指示性语句,它仅告诉汇编程序对后面的指令性语句应如何产生代码,其本身除了可申请一部分存储空间外,并不产生任何等价的机器码,在机器运行期间也不控制计算机的执行。代码段内可包括指令语句和伪指令语句,其它各段只能由伪指令语句组成。汇编语言中的段定义请参见 1.6.3。

汇编语句格式

语句格式为:[名字] 助记符[操作数[,操作数]][;注解]

① 名字项可以是标号或变量,表示本语句的符号地址。对于指令语句,名字项是标号,后有冒号(:),供转移指令作为操作数使用;对于伪指令语句,名字项可以是常量名、变量名、段名、过程名等。名字项最多由 31 个字母组成,可用的字符包括:A—Z a—z 0—9 ' _$. ? @ % 。

② 助记符,由 2—6 个字母组成,是指令或伪指令的符号表示,也可以是宏指令名。汇编程序对指令翻译成机器语言代码,对伪指令将根据其要求的功能处理,对宏指令将依据其定义展开。

③ 操作数可以是常量、寄存器、标号、变量、表达式等。操作数包含了要由指令进行操作的数据的一个或多个单元。当有多个操作数时,各操作数之间要用逗号(,)分开。

④ 注释项是为提高程序的可读性而加入的说明信息,一般是对程序中功能的解释,汇编程序对此不作处理。

1.1.2 操作数约定

(一) 指令中操作数的说明

常量

常量可以是数值常数和字符串常数两类。数值常数有二进制数、八进制数、十进制数和十六进制数,分别以后缀 B、O、D、H 加以区分。通常默认是十进制数,即无后缀时表示是十进制数。字符串常数由引号括起来的一个或多个 ASCII 字符组成,字符串常数中区分大、小写字母。

标号

标号对指令语句起标识作用,具体指向指令所在的地址。标号也正因为是指令地址,所以它有三个属性,即:段地址、段内位移量(或叫偏移量)和类型,类型有 NEAR 和 FAR 两种。标号常用于转移、调用和循环等地址改向指令中的操作数。

变量

变量是对内存中一个数据区起的名字,作为指令中的存储器操作数被引用,由数据区中伪指令来定义。

数值表达式

数值表达式由数值和运算符组成,其结果是一个确定数值,可作为指令中立即数和数据

区中的初值使用。运算符有：① 算术运算符(也可用于地址表达式中)，包括加/正数(+)、减/负数(-)、乘(*)、除(/)、取余(MOD)、右移(SHR)和左移(SHL)。② 逻辑运算符(按位操作)，只用于数值表达式。有：与(AND)、或(OR)、异或 (XOR)、非(NOT)四种。③关系运算符，作用于数值表达式或同一段内的两个存贮地址，运算结果是逻辑值真(1)或假(0)，有：相等(EQ)、不等(NE)、小于(LT)、大于(GT)、小于或等于(LE)、大于或等于(GE)六种。

地址表达式

地址表达式由常量、标号、变量、寄存器内容及运算符组成，其结果是存贮器地址。地址表达式中可含有数值表达式。地址表达式中常见的运算符及其用法见表 1.2。数值表达式和地址表达式中运算符的优先级见表 1.3。

表 1.2 地址表达式中的运算符及用法

运算符	格　式	说　明
[ ]、+	[表达式 1][表达式 2]… [表达式 1+表达式 2…]	各表达式值相加结果作为地址。
PTR	类型 PTR 地址表达式	对变量指定类型是 BYTE,WORD, DWORD…,对标号类型是 NEAR,FAR。
:	段名:地址表达式　或者： 段寄存器：地址表达式	段缀或叫段跨越运算符，临时给地址表达式指定一个段属性。
SHORT	SHORT 标号	当标号到指令的距离小于 128 字节时，短标号用在 JMP 指令中，可节省一个字节内容 。
THIS	THIS 类型	规定该操作数的地址与下一个存储单元地址相同，而类型由这里指定，可以不同。
SEG OFFSET TYPE	运算符　变量或标号	取变量或标号的段址、偏移量、型。
SIZE LENGTH	运算符　变量	取数据区中字节总数，数据项个数。
HIGH LOW	运算符　表达式	取表达式值的高 8 位或低 8 位作为结果。

表 1.3 运算符优先级

运　算　符	优先级	运　算　符	优先级
LENGTH,SIZE,(),[ ],<>	1 最高	加(+),减(-)	7
:	2	EQ,NE,LT,LE,GT,GE	8
PTR,OFFSET,SEG,TYPE,THIS	3	NOT	9
HIGH,LOW	4	AND	10
单目项的正(+),负(-)	5	OR,XOR	11
* ,/ ,MOD,SHL,SHR	6	SHORT	12

（二）本章指令格式中的符号约定

表 1.4　指令格式中的符号约定

符　号	说　　明
dest	代表目的操作数，可以是寄存器或存储器地址。
src	源操作数，可以是立即数、寄存器或存储器地址。
opr，opr1	操作数，操作数 1。
reg，reg8	寄存器操作数，可以是字节、字(16bit 或 32bit)。
reg16，reg32	长度分别是 8 位、16 位、32 位的寄存器操作数。
acc	累加寄存器 AL，AH，AX 或 EAX(仅对 80386)。
sreg	段寄存器。
mem，mem8	存储器操作数。
mem16，mem32	长度分别是 8 位、16 位、32 位的存储器操作数。
im，im8，im16	立即数，单字节立即数，双字节立即数。
instr	表示一条汇编指令。
label	符号，表示标号或变量名，也表示模块名、文件名、宏指令名、记录名、结构名，类型可以是 SHORT，NEAR，FAR。
port	端口地址。取立即数时，范围在 00H—FFH。当为 DX 时，DX 大小在 64K 范围内。
type	标号类型。可以是 BYTE，WORD，SHORT，NEAR，FAR。
expr	表达式。可为数值或地址表达式，也可以是： ① 字符串常数，或用逗号分开的常数表； ② ?。不规定初值的数据项(无确定初值)； ③ 重复项：n DUP expr [，expr，…]。其中 n 为正整数，表示 n 个重复数据项。
(x)	表示 x 的内容。
/	表示“或者”。
↔	互换。
←	替代。如 A←B，表示左端 A 被右端 B 的值所替换。
& 或 ∧	逻辑乘。也表示“同时(且)”。
∨	逻辑加。也表示“或者”。
∀	逻辑异或。
0	复位或逻辑“假”值，也表示数值“0”。
1	置位或逻辑“真”值，也表示数值“1”。
[ ]	表示选择项，即可有可无。

§ 1.2　8086 系列寄存器和寻址方式

1.2.1　8086 系列寄存器

（一）8086/80286 寄存器(如图 1.1 所示)

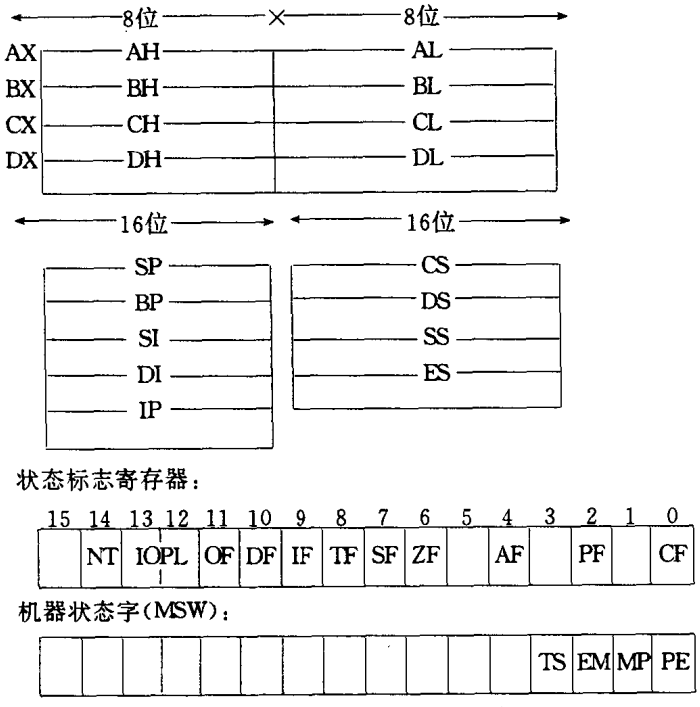

图 1.1　8086/80286寄存器

说明：

① 寄存器 AX,BX,CX,DX 分别是二进制位的累加、基数、计数、数据的 16 位寄存器，每一个均可拆分为高八位与低八位的寄存器独立使用，如 AX 可分为 AH 与 AL。

② 寄存器 SP,BP,SI,DI,IP 分别表示堆栈指针、基址指针、源索引(变址)、目的索引(变址)和指令指针寄存器。

③ 寄存器 CS,DS,SS,ES 分别表示代码段、数据段、堆栈段和附加段寄存器。

④ 状态标志寄存器各位意义为：

CF —— 进位标志位。CF＝1 表示有进位，CF＝0 表示无进位。

PF —— 奇偶性标志位。

AF —— 辅助进位标志位。

ZF —— 结果为零标志位。

SF —— 符号(或负数)标志位。

TF —— 陷井中断标志位。

IF —— 中断允许标志位。当 IF＝1 时，CPU 允许外部可屏蔽中断申请中断。

DF —— 方向标志位。

OF —— 溢出标志位。

IOPL —— 输入输出特权级(仅用于 80286 保护方式)。

NT —— 多任务嵌套标志位(仅用于 80286 保护方式)。

⑤ 机器状态字(MSW)仅用于 80286CPU 中，其意义如下：

PE —— 保护允许位。PE＝1 时处于保护方式，PE＝0 处于实地址方式。

MP —— 监控协处理器位。与 TS 位、EM 位合用，用于确定：当 MP＝0，TS＝0，EM＝0 时，表示 CPU 复位后，未装协处理器，也无仿真软件；当 MP＝

0,TS=0,EM=1 时,表示未装协处理器,但有仿真软件;当 MP=0,TS=1,EM=1 时,表示仿真协处理器时,CPU 若遇到 ESC 或 WAIT 指令则产生中断 7;当 MP=1,TS=0,EM=0 时,表示安装有协处理器;当 MP=1,TS=1,EM=0 时,表示协处理器存在,CPU 若遇到 ESC 或 WAIT 指令,则产生中断 7。

EM —— 仿真协处理器位。若 EM=0,则所有的协处理器操作码都将在实际的协处理器上执行。

TS —— 任务切换位。每当进行任务切换时自动置位,之后当 CPU 遇到 ESC 或 WAIT 指令且 MP 也置位时,则产生中断 7。

(二) 80386 寄存器(如图 1.2 所示)

说明:

① 8 个通用寄存器(EAX—EDI)已扩展到 32 位,其中每个寄存器又可用其低 16 位的寄存器(AX—DI),还可用前面 4 个寄存器低 16 位中的高 8 位寄存器(AH—DH)与低 8 位寄存器(AL—DL)。

② 指令指针(EIP)也扩展到了 32 位,但可以单独使用其低 16 位(IP)。标志寄存器也已扩展到了 32 位,其中低 16 位等同于 80286,另外新增的 RF 和 VM 分别是恢复标志(Resume Flag)与虚拟 8086 方式标志位。

③ 段寄存器除了与 80286 的 4 个段寄存器相同外,另外增加了 FS 和 GS 两个 16 位寄存器留给用户使用,目的是为了减轻 ES 寄存器的负担,并能更好地配合基址和变址寄存器。

④ 4 个控制寄存器 CR0,CR1,CR2,CR3 可用装载和存储指令对其进行存取操作。CR0 的低 16 位也就是 MSW,与保护方式下的 80286 兼容,只比 80286 高出两位 ET 与 NE,分别是扩展类型控制位与数值异常中断控制位。ET=1 时,表示实际组装有 80387。NE=1 时,协处理器指令发生故障时用中断 16 处理;当 NE=0 时,用外部中断处理。装入机器状态字指令 LMSW 和存储机器状态字指令 SMSW 与 80286 指令相同。当存取 CR0 时,应当用"MOV CR0,reg"指令。CR1 保留供未来的 Intel CPU 用。CR2 包含了一个线性地址,这是造成最后一次页故障的地址。CR3 包含了页目录表的物理基地址,因 80386 的页目录表总是页对齐的,故在写入 CR3 时,它的低 12 位被忽略。

⑤ GDT、IDT、LDT 和 TSS 分别是全局描述表(48 位)、中断描述表(48 位)、局部描述表(16 位)和任务状态段表(16 位)寄存器,用于访问由 80286/80386 保护方式所支持的一些表和段,属系统地址寄存器又可被分别命名为 GDTR,IDTR,LDTR 和 TR 。80286 也含有这四个寄存器。

⑥ 调试寄存器(32 位)DR0、DR1、DR2 和 DR3 包含有线性断点地址,DR4 和 DR5 为 Intel 公司所保留,DR6 和 DR7 分别包含断点状态与断点控制内容。另外,TR6 和 TR7 分别是测试控制寄存器(32 位)与测试状态寄存器(32 位)。

1.2.2 8086 系列寻址方式

8086 系列有以下 8 种寻址方式:

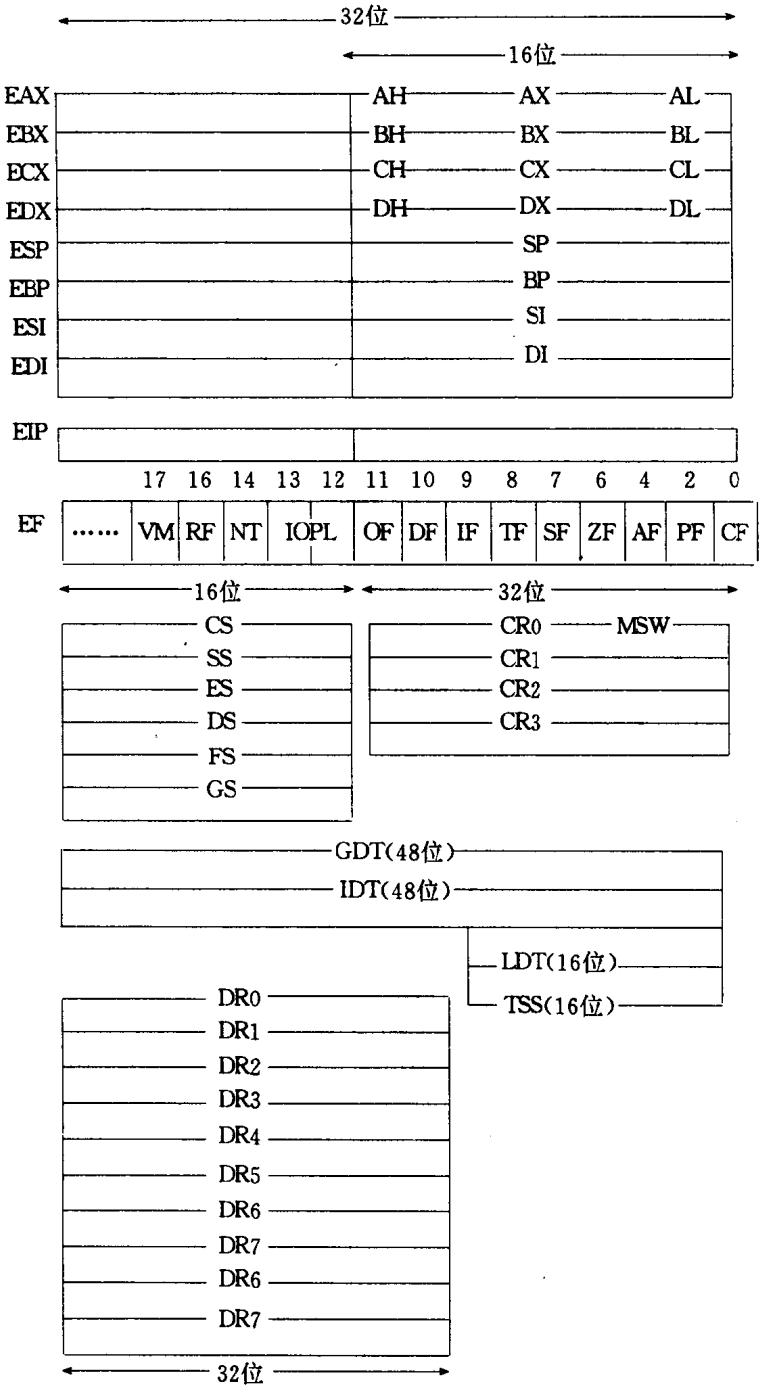

图 1.2 80386寄存器

① 立即寻址。操作数包含在指令中,不需要另外寻找。

如: MOV DX,600H

② 寄存器寻址。操作数在指定的寄存器中。

如: MOV DX,AX

③ 直接寻址。操作数在内存中,地址在指令中直接给出。

如: MOV AX,mydata （其中 mydata 为存储器地址）