

AUTOLISP(10.0版)教程

(初级·中级·高级)

刘银远 编著
董 华 审



北京希望电脑公司

643

AUTOLISP (10.0版) 教程

(初级·中级·高级)

刘银远 编著
董 华 审

北京希望电脑公司

一九九一年十二月

版权所有
不许翻印
违者必究

★北京市新闻出版局

准印证号：3204-90204

★订购单位：北京8721信箱资料部

★电 话：2562329

★电 传：01-2561057

★地 址：海淀影剧院北侧

★乘 车：320、332、302路至海淀黄庄下车

★办公地点：希望公司大楼101房间

★邮 码：100080

前 言

在微机CAD领域中,谈起AutoCAD已是家喻户晓了。在工程设计中要用到AutoCAD,对于CAD基础软件开发的研究者也要参考AutoCAD,之所以如此,是因为AutoCAD强大的图形编辑功能和开放的结构吸引了众多的CAD工作者。自从AutoCAD问世以来,全世界不知道有多少人都在对其进行二次开发,以满足具体的应用要求。从早期的DXF文件、SCR文件过渡到现在用AutoLISP来开发AutoCAD。由于AutoLISP和AutoCAD的巧妙结合,就使得用AutoLISP开发AutoCAD比以往其它任何方法都要简单,而且效率极高。Auto-DESK公司考虑到这一特点,在它以后推出的版本中也将继续延用AutoLISP语言作为AutoCAD的主要编程语言。而且随着AutoCAD版本的提高,AutoLISP语言的功能也在逐渐加强。因此,可以说在如今如果不懂得AutoLISP语言,那么就没有真正地掌握好AutoCAD,也就很难开发出水平较高的应用程序。

由于AutoCAD几年来版本更新比较快,作为嵌入在这个软件内部的AutoLISP语言的功能也在逐渐加强,但是,目前关于介绍AutoLISP9.0以上版本的资料既零散,内容又十分简单,而且大部分资料是从国外文献翻译过来的,其中存在不少语法和笔误。这给广大的AutoCAD用户,尤其是初学者带来了一定的困难。随着AutoCAD在各行各业中的广泛应用,人们越来越感到系统地掌握AutoLISP语言的必要,因为一个用AutoCAD开发的CAD系统离不开用AutoLISP语言编程。为满足广大CAD工作者的迫切需求,使他们有更多的时间去作创造性的工作,而不必把时间和精力花费在对AutoLISP的理解上,作者参考有关中外资料,以及作者多年利用AutoCAD开发应用软件和教学的经验,在对AutoLISP语言进行剖析的基础上,从用户编程的角度出发,编写了这本教程。

本书的编写具有以下特点:

一、在对每个系统函数的功能、用法的讲解上力求全面、系统、准确。所有例子均上机调试通过。例如,对于除法函数(/)的功能的描述和AutoLISP手册及其它参考书目描述的就不同,本书对其解释是比较准确的。

二、初、中、高级用户兼顾。在每个函数的讲解上,首先是讲解函数的功能、用法以及举一些简单的例子,这适用于初学者。为适用中、高级读者的需求,在每个函数或每节后均有一些例题,例题的内容可能涉及其它章节的内容,这就使读者可灵活运用所学知识,综合编程。

三、不但详尽地讲解了AutoLISP的系统函数,而且介绍了大量的编程技巧,几乎所有的例程都有实用价值。

四、介绍了AutoLISP的高级开发技术。例如,用AutoLISP实现屏幕光标控制和重定键盘功义能键;AutoLISP语言和高级语言的接口功能;AutoLISP程序和DOS命令结合使用;任意曲线曲线的绘制等。

全书共十四章。

第一章主要介绍如何在AutoCAD中使用AutoLISP,以及AutoLISP程序的编辑等,

05130/10

当学完这一章后，就可以准备工具开始编写程序了。

第二章是AutoLISP概述。主要介绍AutoLISP的数据类型以及求值过程等。学完本章就可对AutoLISP语言的特点有一个深入的了解。

第三章到第十一章全面系统地介绍了AutoLISP的系统函数的功能、应用以及相应的程序设计技巧。

第十二章介绍了AutoLISP语言的接口功能。主要介绍了AutoLISP与菜单文件、命令组文件、DOS系统以及其它高级语言的联接方法。

第十三章介绍了运行AutoLISP时的内存管理。

第十四章是综合举例。从五个方面的应用程序开发着手，说明了如何综合运用所学知识，进行高级程序设计。

每章后均附有思考题，作为练习。

书末有五个附录。列出了标准的出错信息，系统变量，AutoCAD10.0命令表，ASCII表以及AutoLISP系统函数和符号索引。

本书由大连铁道学院刘银远编写，由大连理工大学董华副教授主审。

本书在编写过程中，一直得到了中国科学院希望高级电脑技术公司秦人华经理、高级工程师的帮助与支持，作者在此谨致谢意。

大连铁道学院张致祥教授审阅了本书的全部初稿，并提出了许多宝贵意见。另外，大连铁道学院李力行教授、陈贺、魏伟、关天民老师以及大连理工大学徐闯老师等曾对本书的编写给予了很多关怀、帮助与支持。作者在此表示诚挚的谢意。

最后作者向能够理解和支持他，并为之付出艰辛劳动的妻子，大连理工大学的王小晶表示深深的谢意。

由于作者水平有限，书中错误与不当之处在所难免，恳请读者批评指正。

作 者

一九九一年八月

内 容 提 要

本书全面系统地介绍了微机辅助绘图软件AutoCAD内部的编程语言AutoLISP。主要介绍了AutoLISP (9.0 ~ 10.0版)系统函数功能和用法,并介绍了相应的程序设计技巧以及在开发AutoCAD中的应用。

全书共十四章。从介绍AutoLISP的数据类型、数据的存储结构以及AutoLISP的求值过程等入手,循序渐进地介绍了AutoLISP的基本函数,函数的定义,输入输出函数,AutoLISP的绘图功能,控制结构,字符串处理函数,数据文件,出错处理与程序调试,图形数据库编辑函数,AutoLISP语言的接口功能,以及内存管理功能。最后一章是一些应用实例。

本书在叙述上由浅入深,力求通俗易懂,函数功能叙述全面、准确且重点突出。书中含有大量的程序实例,图文并茂。书中的一些内容是作者多年研究的成果,含有最新的AutoLISP开发技术,内容较为丰富。每章后均附有习题。

本书可供工厂和科研单位从事CAD工程技术人员阅读,也可用作高等院校有关专业的研究生教材,或者作为本科生的选修课教材。本书亦可用于其它从事计算机绘图、学习LISP语言的有关人员参考。

本书特别适合作为强化AutoLISP语言的培训教材。

目 录

第一章 绪论

1.1 为什么要学习AutoLISP语言.....	(1)
1.2 如何在AutoCAD中使用AutoLISP.....	(3)
1.3 AutoLISP程序的编辑.....	(5)
1.4 关于使用本书的一些说明.....	(6)
1.4.1 记号约定说明.....	(6)
1.4.2 AutoLISP10.0和其以下版本的区分.....	(7)
思考题.....	(8)

第二章 AutoLISP概述

2.1 AutoLISP的数据类型.....	(9)
2.1.1 原子.....	(9)
2.1.2 点对.....	(12)
2.2 AutoLISP的数据存贮结构.....	(15)
2.3 AutoLISP的程序结构.....	(16)
2.4 AutoLISP的求值过程.....	(18)
2.5 AutoLISP程序的工作方式.....	(20)
2.5.1 “读—求值—打印”循环.....	(20)
2.5.2 批处理方式.....	(21)
2.6 AutoLISP的出错处理.....	(27)
2.6.1 表达式错误的处理.....	(27)
2.6.2 求值错误的处理.....	(28)
思考题.....	(28)

第三章 AutoLISP的基本函数

3.1 赋值与求值函数.....	(30)
3.1.1 赋值函数SETQ.....	(30)
3.1.2 赋值函数SET.....	(31)
3.1.3 禁止求值函数QUOTE.....	(32)
3.1.4 FUNCTION函数.....	(33)
3.1.5 再求值函数EVAL.....	(34)
3.2 数值函数.....	(36)
3.2.1 算术运算函数.....	(36)
3.2.2 三角函数.....	(41)
3.2.3 数的类型转换函数.....	(42)
3.2.4 整数位操作布尔函数.....	(43)

3.3 表处理函数	(50)
3.3.1 取表部分内容函数	(51)
3.3.2 表的构造与修改函数	(55)
3.3.3 联接表	(60)
3.3.4 测量表的长度函数	(63)

思考题	(64)
-----	--------

第四章 函数定义

4.1 DEFUN定义有名函数	(66)
4.1.1 函数的定义	(66)
4.1.2 函数的调用	(67)
4.1.3 函数的副作用	(67)
4.1.4 应用DEFUN注意事项	(71)
4.2 LAMBDA定义无名函数	(72)
4.3 增加和修改AutoCAD命令	(74)
4.3.1 增加AutoCAD命令	(74)
4.3.2 修改AutoCAD命令	(75)
4.4 自动执行函数	(77)
4.5 定义“可变参数”函数	(77)

思考题	(79)
-----	--------

第五章 交互式输入函数和屏幕输出函数

5.1 坐标系与角度测量	(80)
5.1.1 坐标系	(80)
5.1.2 角度的测量	(81)
5.2 交互式输入函数	(82)
5.2.1 GET族输入函数	(82)
5.2.2 其它输入函数	(96)
5.3 屏幕输出函数	(98)
5.3.1 用于屏幕和文件的输出函数	(98)
5.3.2 只用于屏幕的输出函数	(103)
5.4 扩展的屏幕和键盘控制	(104)
5.4.1 设备驱动程序“ANSI.SYS”的加载	(105)
5.4.2 用DOS命令实现扩展的屏幕和键盘控制	(105)
5.4.3 用AutoLISP实现扩展的屏幕和键盘控制	(108)

思考题	(112)
-----	-------

第六章 AutoLISP的绘图功能

6.1 COMMAND函数	(115)
6.1.1 参格格式	(115)
6.1.2 COMMAND函数的求值	(116)
6.1.3 为用户输入暂停	(117)

6.1.4	应用COMMAND注意事项	(118)
6.1.5	综合举例	(120)
6.2	图形处理函数	(124)
6.2.1	求相对角度函数ANGLE	(124)
6.2.2	求两点间距离函数DISTANCE	(124)
6.2.3	求极坐标点函数POLAR	(124)
6.2.4	求二线交点坐标函数INTERS	(126)
6.2.5	目标捕捉函数OSNAP	(127)
6.2.6	不同坐标系统的点转换函数TRANS	(131)
6.3	屏幕操作函数	(135)
6.3.1	文本、图形屏幕转换函数	(136)
6.3.2	重画屏幕图形函数REDRAW	(136)
6.3.3	清图形屏幕函数GRCLEAR	(138)
6.3.4	画矢量线函数GRDRAW	(139)
6.3.5	向屏幕文本显示区写文字函数GRTEXT	(141)
6.3.6	视窗说明函数VPORTS	(143)
6.4	访问输入设备函数	(144)
6.5	存取AutoCAD系统变量值函数	(149)
6.5.1	获取系统变量值函数GETVAR	(150)
6.5.2	设置系统变量值函数SETVAR	(150)
6.5.3	综合举例	(150)
6.6	获得AutoCAD环境变量值函数	(153)
	思考题	(154)

第七章 控制结构

7.1	谓词函数	(155)
7.1.1	数的谓词函数	(155)
7.1.2	数据类型谓词函数	(158)
7.1.3	等值谓词函数	(160)
7.1.4	逻辑谓词函数	(162)
7.1.5	判断从属关系的谓词函数	(165)
7.2	条件分支函数	(166)
7.2.1	基本的条件函数COND	(166)
7.2.2	IF条件函数	(170)
7.3	循环函数	(172)
7.3.1	WHILE函数	(172)
7.3.2	REPEAT函数	(179)
7.3.3	FOREACH函数	(183)
7.3.4	MAPCAR函数	(185)
7.4	调用函数的函数	(188)

7.5 顺序控制函数.....	(190)
7.6 函数的递归定义.....	(191)
思考题.....	(197)

第八章 字符串处理函数

8.1 字符、ASCII码互换函数.....	(198)
8.1.1 字符转换成ASCII码函数 ASCII.....	(198)
8.1.2 ASCII码转换成字符函数CHR	(198)
8.2 测量字符串长度函数.....	(200)
8.3 数、串互换函数.....	(200)
8.3.1 整数转换成整数串函数ITOA.....	(200)
8.3.2 数字串转换成整数函数atoi.....	(201)
8.3.3 数字串转换成实数函数atof	(201)
8.3.4 数转换成实数串函数rtos.....	(201)
8.3.5 角度转换成字符串函数angtos.....	(204)
8.4 字符串的链接函数.....	(205)
8.5 字符串截取函数.....	(207)
8.6 字符串大小写转换函数	(208)
8.7 字符串转换成表或原子函数.....	(210)
思考题.....	(211)

第九章 数据文件

9.1 文件的打开.....	(212)
9.2 关闭打开的文件.....	(214)
9.3 输入、输出函数.....	(215)
9.3.1 输出函数.....	(216)
9.3.2 输入函数.....	(218)
9.4 文件查找函数.....	(219)
9.5 综合举例.....	(221)
思考题.....	(228)

第十章 出错处理与程序调试

10.1 出错处理.....	(230)
10.2 程序调试.....	(233)
10.2.1 调试函数TRACE和UNTRACE.....	(233)
10.2.2 其它调试方法.....	(235)
思考题.....	(238)

第十一章 图形数据库编辑函数

11.1 基本概念.....	(239)
11.1.1 实体.....	(230)
11.1.2 实体名.....	(240)
11.1.3 实体标号.....	(240)

11.1.4 选择集	(240)
11.2 选择集操作函数	(241)
11.2.1 实体选择	(241)
11.2.2 选择集构造函数SSGET	(242)
11.2.3 选择集操作函数	(244)
11.3 实体名和选择集与AutoCAD一起使用	(248)
11.4 实体名操作函数	(250)
11.4.1 实体名搜索函数ENTNEXT	(250)
11.4.2 获得最后一个主实体名函数ENTLAST	(251)
11.4.3 选择任一实体函数ENTSEL	(252)
11.4.4 实体标号处理函数HANDENT	(254)
11.5 实体数据函数	(254)
11.5.1 获得实体定义数据函数ENTGET	(254)
11.5.2 修改实体在图形数据库中定义函数ENTMOD	(266)
11.5.3 更新复杂实体屏幕图象函数ENTUPD	(268)
11.5.4 删除或恢复实体定义函数ENTDEL	(269)
11.6 符号表访问函数	(270)
11.6.1 TBLNEXT函数	(271)
11.6.2 TBLSEARCH函数	(274)
思考题	(278)

第十二章 AutoLISP语言的接口功能

12.1 AutoLISP与AutoCAD菜单文件的接口功能	(279)
12.1.1 菜单文件调用AutoLISP	(279)
12.1.2 在AutoLISP程序中激活AutoCAD子菜单	(284)
12.2 AutoLISP与SCR文件的接口功能	(286)
12.3 AutoLISP与DOS的接口功能	(289)
12.4 AutoLISP与高级语言的接口功能	(293)
思考题	(295)

第十三章 内存管理

13.1 AutoLISP的内存分配	(296)
13.1.1 堆栈空间	(296)
13.1.2 合理分配堆栈空间	(297)
13.1.3 为Extended AutoLISP分配内存空间	(297)
13.2 回收节点空间	(298)
13.3 无用存储单元的收集	(300)
13.4 页式虚拟存储功能	(301)
13.5 节点和字符串空间的分配	(303)
13.5.1 设置段的大小	(303)
13.5.2 申请节点空间	(304)

13.6	统计内存状态.....	(304)
13.7	程序设计上的一些技巧.....	(305)
	思考题.....	(306)
第十四章 程序实例		
14.1	方程求根.....	(307)
14.2	中缀表示转换为前缀表示.....	(309)
14.3	任意线型曲线的绘制.....	(312)
14.4	空间曲面的绘制.....	(316)
14.5	排序程序及应用.....	(321)
14.5.1	插入排序.....	(321)
14.5.2	选择排序.....	(322)
14.5.3	排序应用.....	(323)
	思考题.....	(324)
附录A 标准出错信息		
		(325)
附录B 系统变量.....		
		(329)
附录C AutoCAD10.0命令一览表.....		
		(337)
附录D ASCII (美国标准信息交换码)表.....		
		(352)
附录E AutoLISP系统函数和符号索引.....		
		(355)
主要参考文献.....		
		(358)

第一章 绪 论

1.1 为什么要学习AutoLISP语言

LISP(LISt Processing language)是一种计算机的表处理语言,是迄今为止人工智能(Artificial Intelligence)学科领域应用得最为广泛的一种程序设计语言。自1960年由美国麻省理工学院的J. McCarthy提出至今已有三十多年的历史。在它的发展过程中,产生了以下几个典型的版本: MacLISP, InterLISP, ZetaLISP以及CommonLISP。其中的CommonLISP(通用LISP)是近几年来美国的几所大学(如MIT, STANFORD, CMU, RUTGERS等)和工业界(如Bell Lab., DEC, Hewlett-Packard等)的人工智能研究者协同推出的,它概括了MacLISP, InterLISP及ZetaLISP等版本的特点,因而功能较强且拥有其它版本的一些优点,它是目前LISP语言的较完整的版本。

LISP语言又称为符号式语言(Symbolic Language)、函数式语言(Functional Language)。在LISP语言中,最基本的数据类型是符号表达式(Symbolic-expression),处理符号是LISP的特性之一。LISP程序看起来是一个一个的函数的调用。用LISP很容易定义和调用一个用户编写的函数,且LISP函数可以使用递归来定义。递归是LISP的又一重要特性。从本书后面章节的介绍中,读者将会看到LISP的函数递归定义显示了它的强大功能。

LISP语言的一个突出特点是程序和数据二者都取符号表达式的形式,也就是说,一个LISP程序可以把另一个LISP程序作为它的数据处理。这就使得用LISP语言的编程十分灵活。

AutoLISP语言是一种嵌入在AutoCAD内部的LISP编程语言,它是和美国AUTODESK公司发行的装备有ADE-3软件的AutoCAD2.18或更高级的版本一起提供的,是LISP语言和AutoCAD有机结合的产物。

AutoLISP采用了和CommonLISP最相近的语法和习惯约定,它具有CommonLISP的一些特性,但它针对AutoCAD又增加了许多功能,例如,可以把AutoLISP程序和AutoCAD的绘图命令透明地结合起来,使设计和绘图完全融为一体。还可以利用AutoLISP语言编程实现对AutoCAD当前图形数据库的直接访问、修改,为对屏幕图形的实时修改,实现交互设计以及在绘图领域中应用人工智能提供了方便。

AutoLISP是开发AutoCAD的最主要的工具。AUTODESK公司在AutoCAD中嵌入了AutoLISP的目的就是为了帮助使用者充分利用AutoCAD,从而节省时间和提高效率。也许正是由于AutoCAD这种开放的结构,才使其在众多的CAD软件竞争之中长久立于不败之地。

利用AutoLISP开发AutoCAD的一个典型应用是实现参数化绘图程序的设计。参数化程序设计是关于一些特殊对象怎样构成的基本规则集合。例如,我们要画图1.1a所示结构的图形,首先要确定一些规则,以确定如何来表达这个图形。这是和设计这个零件图的设计者有联系的,例如,一条基本的规则是:一个图形在各个方向上尺寸标注必须是封闭的,即既没

有多余的尺寸（即可导出尺寸），又没有缺少的尺寸（尺寸链没封闭）。根据这个规则，我们可以把该图标注成如(b)和(c)所示的尺寸关系，这种图有时称作“盲图”，因它只反映了具有这种结构图的尺寸之间关系，其中的尺寸参数（L1, L2, W1, W2）称作生成这个图的独立参数或变量。参数化绘图程序就是以这些独立变量作为入口参数，根据不同的尺寸变量值生成相应的图形。

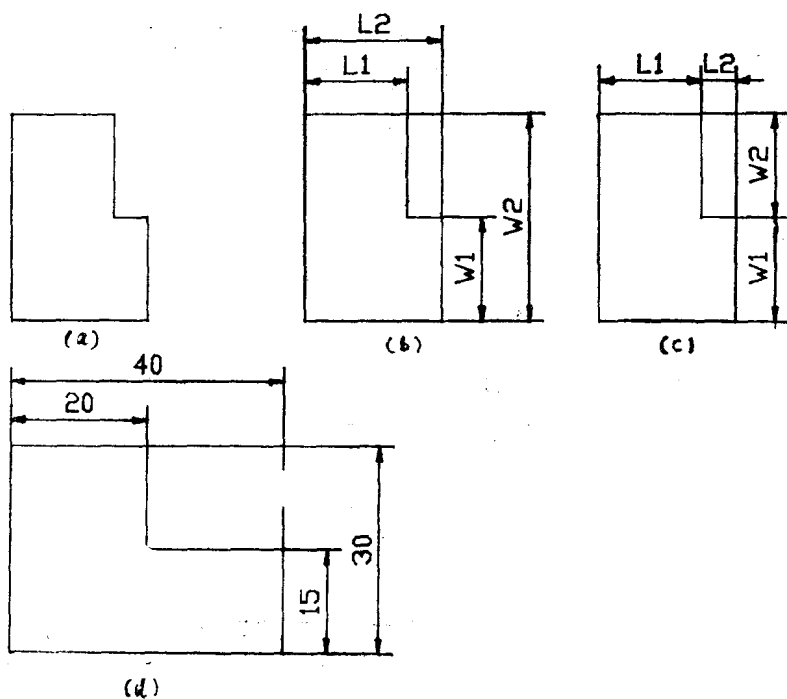


图1.1 参数绘图

下面我们就用AutoLISP编程，看看其是如何实现参数化绘图的：

```
(defun draw (bp l1 l2 w1 w2 / sp)
  (command "pline"
    bp
    (setq sp (polar bp 0.0 l2))
    (setq sp (polar sp (/ pi 2.0) w1))
    (setq sp (polar sp (- pi) (- l2 l1)))
    (setq sp (polar sp (/ pi 2.0) (- w2 w1)))
    (polar bp (/ pi 2.0) w2)
    "close"
  )
)
```

关于程序代码的意义在以后的章节中将介绍。这里用DEFUN（内部函数）定义了一个名为DRAW的函数，该函数带有5个形式参数：bp（图形左下角插入点），L1, L2, W1, W2。现在我们调用该函数画如图1.1(d)所示尺寸的图形：

Command: (draw' (20 40) 20 40 15 30)

回车后, 即可在屏幕的“20, 40”点处画出所需图形。

由于在实际工程设计中, 60%~80%的图形是通过修改已有的设计而形成新设计的, 而且多数是通过修如设计参数(如参数优化)来完成的, 所以, 参数化绘图程序设计具有广阔的应用领域。

上面只是通过一个例子来说明应用AutoLISP是如何开发AutoCAD的。实际上, AutoLISP的用途远不止这些。概括地说, AutoLISP综合了人工智能语言 LISP 的特性和 AutoCAD强大的图形编辑功能的特点, 可谓人工智能绘图语言。AutoCAD从2.18版本开始把LISP语言加入其中, 作为其主要的编程语言。至今AutoCAD版本已更新数次, 其中的AutoLISP语言也在不断完善。由于AutoLISP处理符号而非数字, 它是一个面向对象而不是过程的语言, 它是一个求值型的语言以及其具有简洁的语法, 丰富的数据类型等特点, AutoCAD随着版本的提高, 将一直会把AutoLISP作为其主要编程语言。每一个从事AutoCAD应用开发的工程人员, 学习和掌握AutoLISP语言是必要的, 甚至是必需的。

1.2 如何在AutoCAD中使用AutoLISP

AutoLISP语言同AutoCAD的ADE-3软件一起提供的, 它有两种版本, 一种称作标准AutoLISP, 它是AutoCAD10.0以下的版本所采用的; 另一种称作扩展AutoLISP(Extended AutoLISP), 它是标准AutoLISP的增强版, 由AutoCAD10.0版提供。

Extended AutoLISP的功能和普通的AutoLISP功能完全相同, 其唯一区别是: Extended AutoLISP驻留在扩展内存里, 而不是在基本内存里, 这样就为AutoCAD腾出更多的基本内存用作I/O分页空间。

使用Extended AutoLISP, 必须具有以下软硬件支持:

- 装有AutoCAD所支持的Intel 80286/80287或80386/80387微处理器的计算机, 内存至少为640KB以及硬盘。
- 至少为512KB的AT型扩展内存(非Lotus/Intel/Microsoft型扩存), 而且该部分扩展内存不能用于其它目的(如RAM盘)。
- PC-DOS/MS-DOS 2.0以上版本的操作系统。

如果以上条件均满足, 便可选用Extended AutoLISP而不用标准AutoLISP。Extended AutoLISP在80286/80386的“保护”方式下工作且常驻内存, 这便保证AutoLISP程序及其取用的数据量都远超过标准AutoLISP的限度。Extended AutoLISP不能在PC/XT型的计算机上运行, 这主要因为8088/8086 CPU不支持“保护”方式下的操作。

使用Extended AutoLISP前必须设置环境变量LISPXMEN的值, 它用于指定 Extended AutoLISP在扩展内存的驻留位置以及要占用多大空间用作堆栈空间。有关LISPXMEN环境变量的设置详见第13.1节。

Extended AutoLISP是作为一个独立的程序提供的, 即“EXTLISP.EXE”, 它是一种中断型内存驻留程序(TSR), 它有一部分驻存在扩展内存中, 它与AutoCAD的通讯是通过覆盖文件“ACADLX.OVL”来实现的。如果要使用Extended AutoLISP, 则必须在启动AutoCAD之前运行它。可将EXTLISP命令写入“AUTOEXEC.BAT”文件中, 使开机后就自动执行。

如果需要, 你还可将EXTLISP从内存中清除。AutoCAD10.0的发行盘上提供的“REMLISP.EXE”文件用于完成这项工作, 为此, 退出AutoCAD, 在DOS提示符下运行REMLISP, 它将还给DOS那部分曾被EXTLISP占用的内存空间。

如果没有足够的硬件支持, 则只能选用标准 AutoLISP, “ACADL.OVL”是标准 AutoLISP的覆盖文件。标准 AutoLISP占用基本内存空间。在使用标准 AutoLISP 前, 同样要进行环境参数设置, 以确定 AutoLISP使用的堆 (HEAP)、栈 (STACK) 空间的大小, 详见第13.1节。

无论决定使用何种 AutoLISP, 在使用前还必须对其进行配置。配置是从 AutoCAD 的主菜单开始进行的。下面是配置 AutoLISP 的对话过程:

Main Menu

0. Exit AutoCAD
1. Begin a NEW drawing
2. Edit an EXISTING drawing
3. Plot a drawing
4. Printer Plot a drawing
5. Configure AutoCAD
6. File Utilities
7. Compile shape/font description file
8. Convert old drawing file

Enter selection: 5

选择主菜单的第 5 项, 进入配置 AutoCAD。

Configuration menu

0. Exit to Main Menu
1. Show current configuration
2. Allow I/O port configuration
3. Configure video display
4. Configure digitizer
5. Configure plotter
6. Configure printer plotter
7. Configure system console
8. Configure operating parameters

Enter selection () : 8

选择8, 进入配置工作参数子菜单。

Operating parameter menu

0. Exit to configuration menu
1. Alarm on error
2. Initial drawing setup
3. Default plot file name
4. Plot spooler directory

5. Placement of temporary files
6. Network node name
7. AutoLISP feature

Enter selection (0) : 7

Do you want AutoLISP enabled? (N) Y

选择工作参数菜单子任务7, 允许用户接通或关闭AutoLISP特性。若对上面的提示回答Y, 则AutoLISP接通, 否则关闭。

若使用了Extended AutoLISP, 且对上面的提示回答了Y, 则AutoCAD又提示,

Do you want to use Extended AutoLISP? (current)

回答Y, 则使用Extended AutoLISP; 回答N, 则使用标准AutoLISP。

1.3 AutoLISP程序的编辑

AutoLISP程序是作为ASCII码文本文件来编辑的, 所以, 任何一种ASCII文本编辑器均可用来编辑AutoLISP的源程序。最早用的文本编辑器是DOS提供的EDLIN行编辑程序。用EDLIN编辑一个文本文件过程如下:

C: EDLIN (文件名)

New file

* |

1*(defun f (x y)

2* (+ (* x y) 2.0)

3*)

4* 按 CTRL-C 键

* E

其有关命令可参阅DOS使用手册。其中(文件名)必须符合DOS的有关规定, 且对于AutoLISP程序, 其扩展名一般取为“.LSP”。

用EDLIN编辑程序, 修改特别不方便, 目前, 市场上已有很多全屏幕编辑软件, 如“ED.COM”, 以及专用于编辑LISP程序的软件出售, 可以去选购一个, 作为快速编辑AutoLISP程序用。但应注意, 若选用的是字处理软件, 则必须保证是以ASCII文本文件的形式存盘, 这样AutoLISP才能识别。

由于一个AutoLISP程序往往需要调试多次才能成功, 如果反复地退出AutoCAD, 进入DOS进行编辑, 然后再进入AutoCAD调用调试, 势必浪费大量的机时, 实际上, 没有用这种方法调试程序的。在AutoCAD中提供了一个SHELL命令, 该命令允许用户在保留绘图编辑程序的同时, 执行AutoCAD外部的实用程序(如DOS命令)。当在AutoCAD的“Command:”提示下执行SHELL命令时, 其首先检查当前磁盘中是否存在一个“ACAD.PGP”的文件以及其中是否有SHELL命令的定义, 如果没有, 则SHELL命令不能执行。

“ACAD.PGP”文件是AutoCAD提供的可以修改的ASCII文本文件。文件中的每一行描述了一个在AutoCAD内部可以执行的外部程序。当用户输入一个命令, 若AutoCAD