

## 使用BASIC编译程序开发光学设计程序

陈海清 严国萍

(华中工学院)

BASIC编译程序支持大多数解释的BASIC语言。本文叙述了编译调试BASIC解释光学设计程序的过程,强调了处理方法;通过光学设计子程序及程序段描述了解释程序和编译程序在BASIC语言上的不同之处。

### 一、引言

微型机的发展是近代重大科技成就之一,正日益深入各个领域。目前我国已引进相当数量IBM-PC个人计算机。PC机的突出的优点之一,具有BASIC编译程序的功能。该机的BASIC编译程序是一个优化编译程序,由此而建立的应用程序具有两个主要优点。

1. 对大多数程序来说大大地增加其执行速度。例如,对一个十一个面的照相物镜而言,计算完整的光学参数:高斯分布值、各面初级象差系数、轴上点、轴外点所有象差(五个孔径、六个视场)、子午、弧矢三种色光的横向象差、各视场各孔径的镜面高和 $tg(U)$ ,未经编译的BASIC源程序需1小时的运行时间,经编译之后的光学设计程序仅需运行5分钟,打印时间6分钟。

2. 增加了BASIC源码的安全性、保密性,使其免遭未经允许的改变程序和泄露。

### 二、使用BASIC编译程序开发光学设计程序

PA II-846程序是一个通用的BASIC光学计算程序,运行可靠,计算准确,功能齐全。该程序采用经典几何光学象差理论导出的一整套计算公式。本程序的功能齐全。

1. 指定波长折射率的计算;
2. 近轴、空间光线追迹;
3. 结组计算;
4. 整体缩放;
5. 整组弯曲;
6. 计算实际晕瞳中心;

收稿日期:1986年9月28日。

7. 离焦;
8. 单个透镜几何参数计算;
9. 五种不同的输出格式;
10. 孔径、视场分割系数控制。

现把上述 PA II -846 程序的文件名取为“HAI.BAS”。这是一个已建立并调试成功的 BASIC 源程序,因此接着就可编译程序、连接光学设计程序所需要全部模块、运算编译好的用户程序。

#### 1. BASIC 编译程序包的内容与编译方法

BASIC 编译程序包包括:

两个 BASIC 和 LIBRARY 的软盘。

BASIC 软盘包括下列文件:

BASCOM.COM-BASIC 编译程序、LINK.EXE-LINK 连接程序、DEMO.BAS-示范程序、样本批文件。

LIBRARY 软盘包括下列文件:

BASCOM.LIB-BASIC 程序库、BASRUN.EXE 运行期间模块、BASRUN.LIB 运行期间模块程序库、IBBASRUN.OBJ 运行期间模块程序库、IBMCOM.OBJ 通信模块。

通常情况下由以下五个文件组成:

BASCOM	LIB	103936
BASCOM	COM	41600
LINK	EXE	41856
BASRUN	LIB	3072
BASRUN	EXE	31744

文件扩展名后面的数字表示该文件所占的字节数。

使用 BASIC 编译程序开发文件名为“HAI.BAS”光学设计程序。为了使开发过程顺利、方便,需在带有硬盘的 PC/XT 机上操作。

#### 1. SAVE “HAI.BAS”, A

然而,使用 System 命令退出 BASIC。

#### 2. 在 MS-DOS 状态下打入命令

A> BASCOM HAI,

进行编译、产生目标文件 HAI.OBJ

#### 3. 用连接程序连接,打入命令

A> LINK HAI,

生成 HAI.EXE 文件

#### 4. 运行 HAI.EXE 文件

A> HAI

执行 HAI.EXE 文件,进行运算。在 A 盘上如果没有 BASRUN.EXE 文件,则屏幕上显示:

Connot find A: BASRUN, EXE

Enter new driver letter,

此时只要输入BASRUN.EXE文件即可。

## 2. 光学设计初始数据的输入

光学设计是需要多次修改数据的，尤其在透镜个数较多的情况下，数据的输入量是较大的。在编译情况下，绝不能按解释 BASIC 源程序的方式输入，仅以为按以下两种方式输入为好。

1. 数据文件 在 PC 机系统中，数据文件主要通过磁盘 BASIC 的文件管理系统来实现的。数据文件存取方式分为两大类，顺序存取方式和随机存取方式。在顺序存取方式中，把数据写入磁盘或磁盘读出数据时，按记录号从前向后顺序地进行。从光学设计的具体要求而言，建议采用顺序存取方式。

在存取数据文件之前，必先打开文件。打开文件用 OPEN 语句，格式为：

OPEN <方式>, [#] <文件号>, <文件名> [, <记录长>] 或

OPEN <文件名> [FOR <方式>] AS [#] <文件号> [LEN = 记录长度]

在程序中打开了文件，则使用完毕后，必须关闭文件。文件关闭语句的格式为：

CLOSE [#] [, <文件号>] ...

下面创建光学设计初级数据的一个顺序文件，文件名为 DATA.BAS，在 B 盘上生成。使用 TYPE 命令，显示顺序文件清单如下：

```
LIST
10 OPEN "O", #1, "data"
20 READ A
30 IF A = 1E+14 THEN 1000
40 PRINT #1, A
50 GOTO 20
100 DATA 3, 5, 3, -140.47, -0.03, -7.5, 0, 0
110 DATA 1, 0, 0
120 DATA 1, 1.5163, 1.6725, 1, 1, 1
130 DATA 1, 1.52196, 1.68747, 1, 1, 1
140 DATA 1, 1.5139, 1.6666, 1, 1, 1
150 DATA 0, 2.7, 1, 0.01, 0
160 DATA 13.116, -10.3, -35.21, 0, 0
990 DATA 1E+14
1000 CLOSE 1
OK
```

数据区由 100 行开始至 990 行，皆可根据需要填写。修改数据简单、方便。对于该数据文件不进行编译。

2. 管道 (piping) 技术 管道的概念是伴随 I/O 重定向而来的。一个管道 (pipe) 是一条连接两个程序的计算机处理线，就是把第一个程序的输出变为第二个程序的输入。

创建光学参数输入的管道方式：

A>EDLIN B:DATA

New file (对新建立的文件)

或End of input file (对盘上已有的文件)

尚若是新建的文件

New file

• I

1: •, 3, 5, 3, -140.47, -0.03, -7.5, 0, 0

2: 1, 0, 0

3: 1, 1.5163, 1.6725, 1, 1, 1

4: 1, 1.52196, 1.68747, 1, 1, 1

5: 1, 1.5139, 1.6666, 1, 1, 1

6: 0, 2.7, 1, 0.01, 0

7: 13.116, -10.3, -35.21, 0, 0

• E

按上述方法创建的两种原始光学数据输入方式后,即可执行编译通过的光学程序CHEN.  
EXE

A>CHEN

3. 输出打印或屏显方式的处理

编译通过的光学设计程序执行时,  命令不起作用。为此为满足光学设计的需要,

1. 不一定要每次都打印计算结果, 仅需终端显示, 提供观察修改即可。

2. 需要打印与显示同时进行。

这里同时编译了两个光学设计程序, HAI1.EXE, 只显示不打印, HAI2.EXE, 既显示又打印, 这两个文件通过主程序 CHEN.EXE 调用, 控制自如。

主程序 CHEN

LOAD" LIST

10 PRINT " == PRINT DATA ? == "

20 INPUT A\$

30 IF A\$ = "N" THEN CHAIN "HAI1"

40 IF A\$ = "Y" THEN CHAIN "HAI2"

50 GOTO 10

60 END

OK

或者按批作业方式进行

A>COPY CON:R.BAT

PAUSE ARE YOU PRINT (Y/N)

L<L

```

IF L = Y THEN LOOP
HAI1
LOOP:
HAI2
END
^Z

```

启动磁盘后，只要键入R即可。

A > R

综上所述，若采用数据文件及主程序调用的结构形式，用户工作盘的文件有：

A > DIR /W

Volume in drive A has no label

Directory of A:\

COMMAND.COM BASCOM

LIB BASCOM.COM

LINK EXE BASRUN

BASRUN EXE HAI1 EXE

HAI2 EXE CHEN EXE

DATA

DATA.BAS

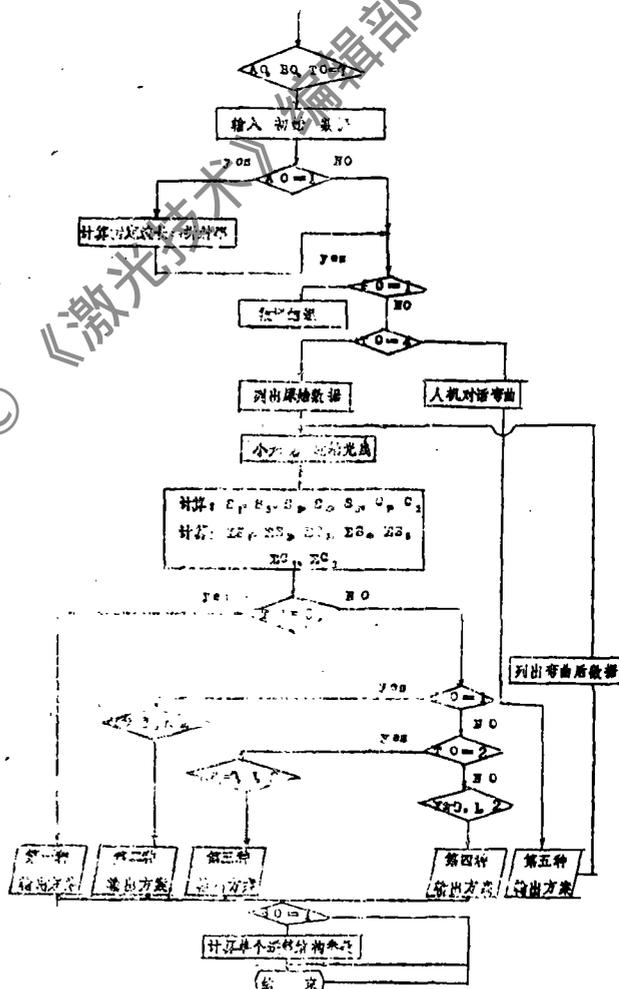
11 File(s) 26624 bytes

free

#### 4. 光学设计 BASIC 源程序主框图

BASIC 语言具有会话性的特点。在 BASIC 源程序中，设置了 INPUT 语句，通过键入不同的常量，使源程序完成选择性计算，选择性打印。例如，程序一开始，设置有 AO、BO、CO 三个人机对话的变量；AO = 1 或者 0，或计算指定波长的折射率或跳过该子程序，BO = M 或者 0，或计算 M 个单透镜的结构参数或跳过该子程序，TO = 0、1、2、3、4 即有 5 种不同的计算途径并与之对应的 5 种不同的打印格式可供选择。从另一方面考虑，人机对话的数据也不宜过多，往往会给新操作者带来不便。

光学计算程序主程序框图



该源程序占内存大约为26K，总程序行约1000行。

### 三、BASIC 语言在编译与解释程序状态下的主要差别

在编译情况下，说明语句 COMMON、DEF、DIM 和 REM 语句一般应放在程序的首部，对于几个程序模块共同使用的公共变量的说明部分，通常放在一个单独的包含文件中，并在这几个程序模块前部用包含命令 \$INCLUDE 把包含文件嵌入。

某些说明语句有一些特殊的要求，例如：

1. COMMON 语句必须出现在所有可执行的语句之前。
2. DIM 语句中的下标范围只允许使用整型常数。
3. REM 语句后随 <命令表>表示编译命令。

例如上例中提到的包含命令可表示为：

```
10 REM $INCLUDE 'DEFINE.HD'
```

4. CHAIN 不允许包含 ALL、MARGE、DELETE 和 <行号>等任选项，所有公共变量都必须用 COMMON 语句说明。

5. 不允许使用 ON PLAY (n) 和 ON TIMER 语句。

6. FOR-NEXT 语句的特殊区别。

在 IF-THEN 的条件语句之后，使用 FOR-NEXT 语句时，PC 机的 BASIC 以及编译情况下的 BASIC 都与一般 BASIC 语句之间存在着重要的区别。下面通过一个子程序例来说明——即光学系统的半径、间隔（包括厚度）、折射率的序号的前后对调，以便反向计算焦距、截距等参数的需要。

说明：N 表示为总面数，包括象面。光栏在系统最前面不计为面数，在系统中间计为一面。

- R (N)            N个面的曲率半径。  
D (N)            N个面之间的间隔（或厚度）。  
D (N+1)         N+1个主色光的折射率。

(1) 基本或扩展 BASIC 的序号对调子程序：

```
1100 IF INT(N/2) = N/2 THEN 1130
1110 FOR I = 1 TO (N - 1)/2
1120 GOTO 1140
1130 FOR I = 1 TO N/2
1140 R1 = - R(N - I) : R(N - I) = - R(I) : R(I) = R1
1150 NEXT I
1160 IF INT(N/2) = N/2 THEN 1190
1170 FOR I = 2 TO (N + 1)/2
1180 GOTO 1200
1190 FOR I = 2 TO N/2
1200 D1 = D(N + 1 - I) : D(N + 1 - I) = D(I) : D(I) = D1
1210 N = 1N(N + 1 - I) : N(N + 1 - I) = N(I) : N(I) = N1
```

1220 NEXT I

1230 RETURN

由以上子程序可见，在 IF-THEN 条件语句之后，1110和1130两程序行，两个 FOR 可共用1150程序行的一个 NEXT，是允许的。

## (2) PC 机的 BASIC 及编译状况下的 BASIC

若源程序在直接调用上述编制的子程序，在PC机上RUN或编译时，则打印出错误信息：

FOR 1110 without NEXT 或者

FN

表示在1110程序行没有 NEXT 与 FOR 配对，即此时认定1150程序行的 NEXT 仅只与1130程序行的 FOR 配对，而不管 IF-THEN 的执行情况。改写之后的子程序如下。不过这种一一配对的改写方法显然有些繁琐，我们还可以作另一种处理，即把子程序中重复出现的程序行再编写为子程序，那么程序就简练多了。

```
1100 IF INT(N/2) = N/2 THEN 1130
1110 FOR I = 1 TO (N - 1) / 2
1120 R1 = -R(N - I) : R(N - I) = -R(I) : R(I) = R1 : NEXT I
1130 FOR I = 1 TO N / 2
1140 R1 = -R(N - I) : R(N - I) = -R(I) : R(I) = R1
1150 NEXT I
1160 IF INT(N/2) = N/2 THEN 1200
1170 FOR I = 2 TO (N + 1) / 2
1180 D1 = D(N + 1 - I) : D(N + 1 - I) = D(I) : D(I) = D1
1190 N1 = N(N + 1 - I) : N(N + 1 - I) = N(I) : N(I) = N1 : NEXT I
1200 FOR I = 2 TO N / 2
1210 D1 = D(N + 1 - I) : D(N + 1 - I) = D(I) : D(I) = D1
1220 N1 = N(N + 1 - I) : N(N + 1 - I) = N(I) : N(I) = N1
1230 NEXT I
```

## 参 考 文 献

- [1] 徐 军译，《BASIC编译程序》，辽宁省电子计算机学会，1984年。
- [2] 陈海清，用于Apple- II 微型机多功能光学计算程序，《光仪通讯》，1985年，第3期，第11~18页。

(下转第22页)

张兴德、任大翠、陈铁民教师给予许多帮助，在此致谢。

### 参 考 文 献

- [1] J.A.P., 1978, Vol. 49, P. 1031.
- [2] J.A.P., 1976, Vol. 47, P. 4578.
- [3] IEEE J.Q.E., 1981, QE-17, P.178.
- [4] J.A.P., 1977, Vol. 16, P. 205.
- [5] IEEE J.Q.E., 1979, QE-15, P. 718.
- [6] IEEE J.Q.E., 1981, QE-17, P. 646.
- [7] W. J. Devlin et al., Inst. Phys. Conf. Ser., Chapter12 Symp., Japan, 1981, No.63, P.567.
- [8] R.A. Logan et al., Inst. Phys. Conf. Ser., 1982, No.18, P.895.
- [9] M.T. Oomuk et al., Inst. Phys. Conf. Ser., 1981, No.16, P.566.

#### An investigation for the steady state characteristics of buried crescent InGaAsP/InP laser

Wang Xiangwu

(Changchun Optics and Fine Mechanism College)

#### Abstract

The steady state characteristics for buried crescent InGaAsP/InP laser are investigated by effective refraction index method and numerical method. The carrier distribution for various structure parameters and the relationship between threshold current and structure parameters are given.

---

(上接第59页)

#### Using BASIC compiler to develop optical design program

Chen Haiqing, Yan Guoping

(Huazhong University of science and Technology)

#### Abstract

Basic compiler supports the most functions of interpreting Basic language. In this paper compiling process for Basic interpreting the optical design program is described, the processing method is emphasised, the difference in the Basic language between interpreting and compiler is discussed with optical design subroutine and multiple—statement lines.