https://ntrs.nasa.gov/search.jsp?R=19820024975 2020-03-21T06:26:46+00:00Z

View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by NASA Technical Reports Server

General Disclaimer

One or more of the Following Statements may affect this Document

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.
- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.
- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.
- This document is paginated as submitted by the original source.
- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

Produced by the NASA Center for Aerospace Information (CASI)

9950-721

(NASA-CR-169302) THE SMALL MODULE FIXED MIRROR DISTRIBUTED FOCUS (FMDF) PHOTOTHERMAL CONCENTRATOR STUDY Final Report (Arizona Univ., Tucson.) 32 p HC A03/MF A01 CSCL 10A

N82-32851

Unclas
G3/44 28906

THE SMALL MODULE FIXED MIRROR DISTRIBUTED FOCUS (FMDF) PHOTOTHERMAL CONCENTRATOR STUDY

(Modification No. 4 of 18 February 1980)

FINAL REPORT

A. B. Meinel

RECEIVED
SEP 1982
NASA STI FACILITY
ACCESS DEPT.

27 February 1981

JPL Contract No. 955162

Optical Sciences Center
University of Arizona
Tucson, Arizona 85721

This work was performed for the Jet Propulsion Laboratory, California Institute of Technology, sponsored by the National Aeronautics and Space Administration under Contract NAS 7-100.

This document contains information prepared by the Optical Sciences Center, University of Arizona, under JPL subcontract. Its content is not necessarily endorsed by the Jet Propulsion Laboratory, California Institute of Technology, or its sponsors.

ABSTRACT

This report covers the period of the contract extension, from 1 September 1979 to 31 August 1980. Tasks established and carried out during this period were: the development of a general ray trace evaluation program, called ICARUS; the study of novel Fresnel concepts; and the review of a report draft on novel Fresnel concepts. ICARUS is documented herein; reports on the novel Fresnel concepts have been submitted earlier.

NEW TECHNOLOGY

Fresnel Computation Program

A general ray trace evaluation program, designated ICARUS, has been developed, with the goal of widening the generality over that in commercially available optical evaluation programs. The main thrust of the effort has been to expand the number of geometrical shapes, tilts, and decenters that could be explored, to include conic, aspheric, and toric surfaces. To keep the program task within time and budget, the effect of facet errors was not to be included. This task has been performed principally by Steve Eckhardt, graduate student, under the supervision of A. B. Meinel. Documentation is appended to this report (Appendix A).

A review of the program led to the decision by us and by JPL's program management to terminate this task before completion; however, we were able to bring it to a usable state; examples of Fresnels that can be ray traced are shown in Appendix B. One reason for withdrawing from the hope of a fully general program was the conclusion that a specific program could be written for any novel Fresnel surface that could not be handled by an existing program.

It is important to know what has not been included in the program to date. Not included are the following capabilities: (1) finite groove size, (2) diffraction, (3) local groove effects (facet tilt error), (4) substrate effects (lens surface errors), and (5) general aspheric substrate shapes. The facet tilt error and surface errors are relatively easy to handle for reflective Fresnels in that whatever the angular distribution of assumed errors is, the effect at the focus is twice the angular distribution, subject to the geometrical cosine effect of oblique incidence on the focal plane. The effect is considerably more complicated for refractive Fresnels. We will examine the analytical expressions in future work if JPL requests it.

Questions of adding facet error statistics have yet to be incorporated. The analytical approaches to this are currently being evaluated and could appear as an addendum at the appropriate time.

Fresnel Optic Study

The rest of the current funding was expended in addressing novel Fresnel concepts, on which reports have been submitted, and in reviewing a report draft on this topic prepared by JPL. Several of the novel concepts appear to be patentable, and a report was written on this question, but no reply was received concerning JPL's desire to proceed with this question. In the absence of a reply we will, at an appropriate time, prepare a paper for journal publication on these novel concepts.

APPENDIX A

PROGRAM DOCUMENTATION

PROGRAM IDENTIFICATION

Program Title:	Optical Sciences Center Fresnel Evaluation Program
Program Code Name:	ICARUS
Writer:	Steve Eckhardt Optical Sciences Center University of Arizona Tucson, Arizona 85721
Date of Completion:	May 1980
Source Language:	Fortran (Data General FORTRAN 5)
Availability:	Program listing as computer printout (attached)
Abstract:	ICARUS is a ray trace program of greater generality than commercially available programs.

PROGRAM DESCRIPTION

Method of Solution:	Paraxial rays are traced by normal methods. Real rays are traced in accordance with the conic ray trace equations given in Table 1. Fresnels are treated by assuming that the point of intersection of the ray with the surface is in the tangent plane of the lens and that the angle of the normal to the ray is the same as if the Fresnel were an ordinary conic.
Program Capabilities:	Paraxial and real ray tracing, as well as radiant energy distribution computation, for conic, fresnel, toric, curved fresnel, toric fresnel, and curved toric fresnel surfaces. Not included are finite groove size, diffraction, local groove effects (facet tilt error), substrate effects (lens surface errors), and general aspheric substrate shapes.

Data Inputs:	Data are entered interactively at the terminal, in response to questions posed by the program. Tables 3 and 4 provide guides to input options and format.
Program Options:	See Table 3
Printed Output:	N/A
Other Outputs:	A file is created for each new lens. These can only be deleted outside of the program.
Flow Chart:	Figure 1 is a general system flow chart.
Sample Run:	Table 3 is an annotated "walk-through" of the program.

SYSTEM DOCUMENTATION

Computer Equipment:	Data General Eclipse
Peripheral Equipment:	N/A
Source Program:	Table 5 is a complete source code listing for ICARUS and its subroutines.
Variables and Subroutines:	See the source code listing for variables. Table 2 is a list of subroutines, with descriptions of their functions.
Data Structures:	N/A
Storage Requirements:	Minimal
Maintenance and Updates:	None

ORIGINAL PAGE IS
OF POOR QUALITY

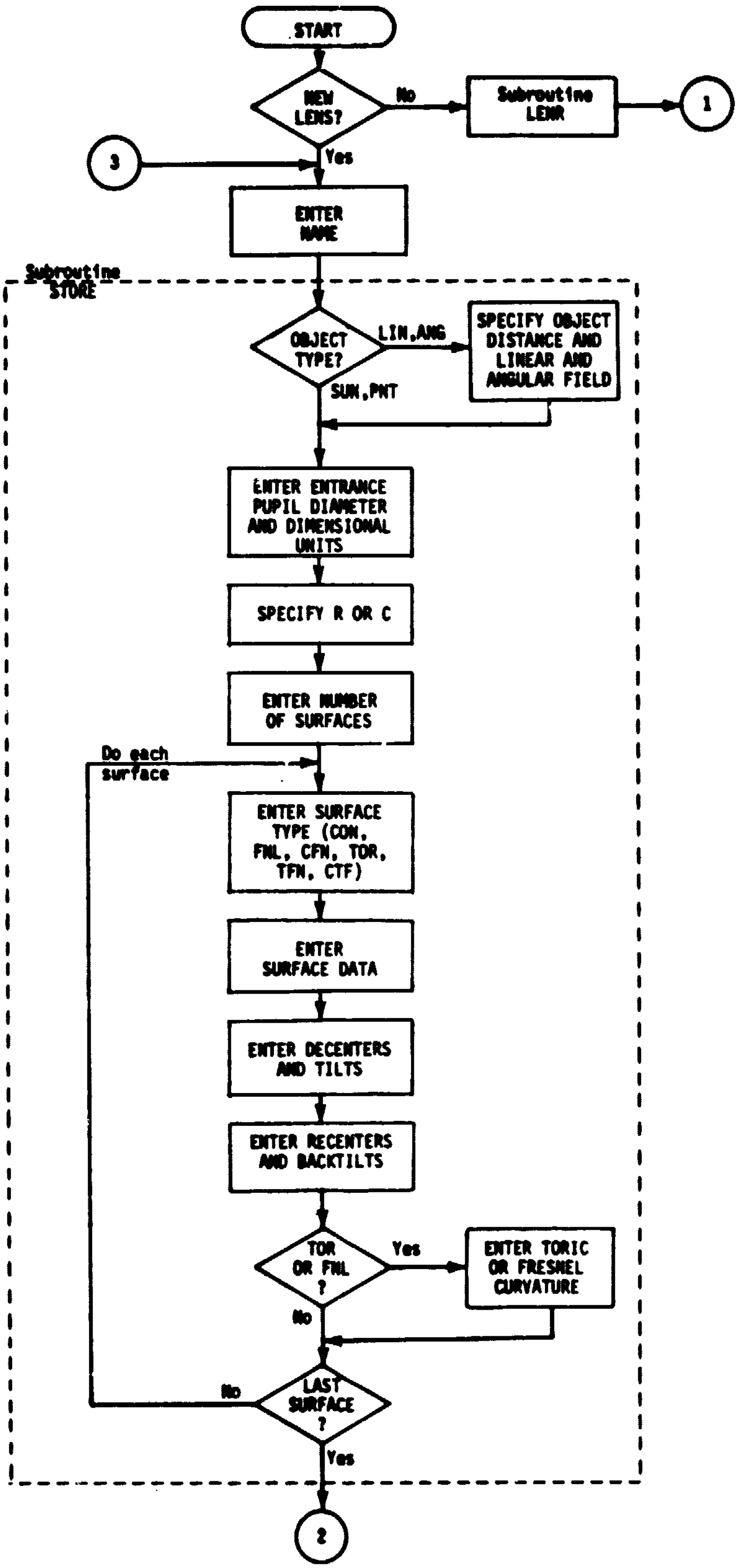


Fig. 1. ICARUS System Flow Chart

ORIGINAL PAGE IS
OF POOR QUALITY

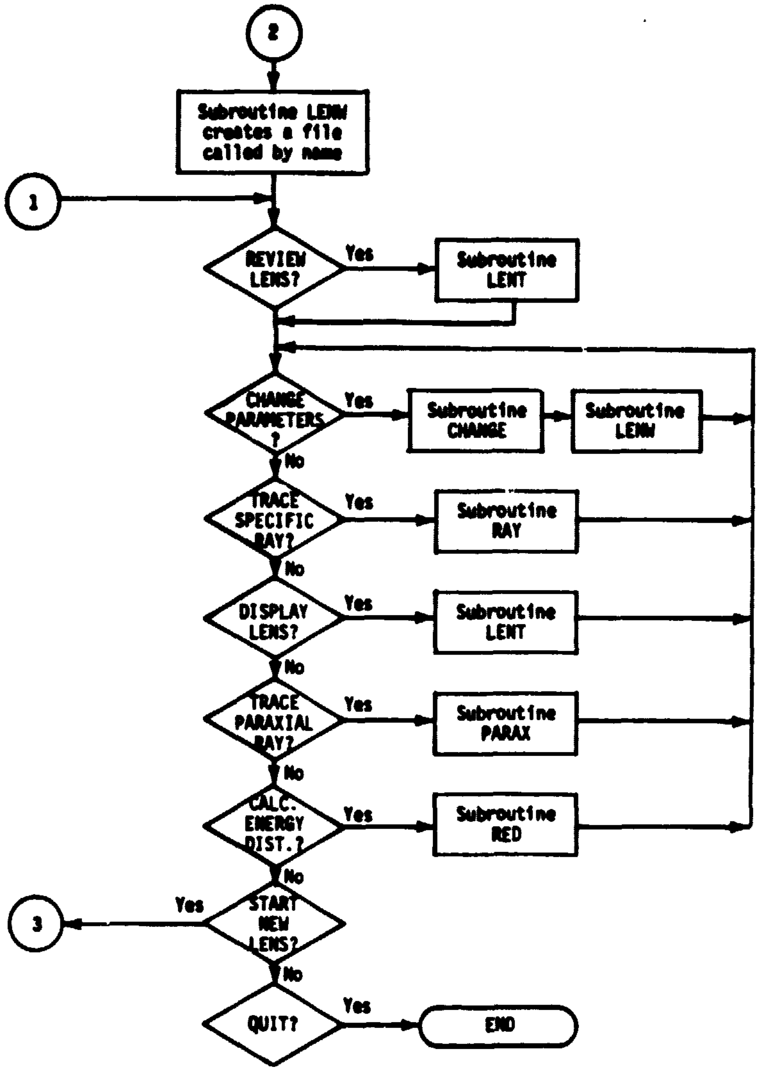


Fig. 1. ICARUS System Flow Chart (Continued).

ORIGINAL PAGE IS
OF POOR QUALITY

Table 1. Ray Trace Equations for Conics.

Opening:

$$A = [t^2 + (h_y - y_t)^2 + (h_x - x_t)^2]^{1/2}$$

$$L = \frac{h_x - x_t}{A}$$

$$M = \frac{h_y - y_t}{A}$$

$$N = t/A$$

Transfer to Surface:

$$B = c(x_t^2 + y_t^2)$$

$$C = N - c(Lx_t + My_t)$$

$$D = [C^2 - cB(1 + kN^2)]^{1/2}$$ $D^2 < 0$: ray missed surface

$$A = B/(C + D)$$

$$x_s = x_t + AL$$

$$y_s = y_t + AM$$

$$z_s = AN$$

Refraction:

$$E = [1 - 2ckz_s]^{-1/2}$$

$$\cos I = DE$$

$$\mu = n/n'$$

$$\delta = (1 - \mu^2 + \mu^2\cos^2 I)^{1/2} - \mu\cos I$$

$1 - \mu^2 + \mu^2\cos^2 I < 0$: total internal reflection

ORIGINAL PAGE IS
OF POOR QUALITY

Table 1. Ray Trace Equations for Conics (Continued).

$\alpha = -cE\, x_s$

$\beta = -cE\, y_s$

$\gamma = [1 - C(k+1)z_s]E$

$L' = \mu L + \delta\alpha$

$M' = \mu M + \delta\beta$

$N' = \mu N + \delta\gamma$

Transfer to Tangent Plane:

$x_t = x_s + L(t - z_s)/N$

$y_t = y_s + M(t - z_s)/N$

$A = [(t - z_s)^2 + (y_t - y_s)^2 + (x_t - x_s)^2]^{1/2}$

Variable Definitions:

x_t = ray height in x on tangent plane

y_t = ray height in y on tangent plane

x_s = ray height in x on surface plane

y_s = ray height in y on surface plane

t = distance from object to first lens surface, or distance between surfaces

h_x = object height in x

h_y = object height in y

L,M,N = x,y,z direction cosines for the ray before refraction

L',M',N' = x,y,z direction cosines for the ray after refraction

α,β,γ = direction cosines of the normal to the surface in x, y, and z, respectively

A = geometrical path length between a surface and the following tangent plane; also vice versa, the refractive index times the optical path length

I = the angle of incidence on the surface

n,n' = the refractive indices of the materials before and after the surface

B,C,D,E,δ = dummy variables

Table 2. List of Subroutines.

Name	Called by	Description
CHANGE	Main program	Interacts with user to make individual changes in the lens without requiring retyping of entire lens; inputs are as in subroutine STORE
LENR (lens read)	Main program, PARAX	Reads a currently existing lens file into the active area
LENT (lens type)	Main program	Displays the current lens on the terminal in tabular form
LENW (lens write)	CHANGE, STORE	Creates a file called by any name the user specifies, and writes the lens to it. Should the need arise, this file may be accessed for modification without use of the program
PARAX	Main program	Does a paraxial ray trace that enables the user to locate the Gaussian image plane, and in general to do a rough lens design
RAY	Main program, RED	Traces a specified ray through the lens (see TORT and SYMM)
RED	Main program	Calculates the radial energy distribution in the image plane
SORT	RED	Orders the rays by increasing radial distance from the paraxial image point
STORE	Main program	Interactively accepts data input
SYMM	RAY	Does refraction calculations for symmetrical surfaces
TORT	RAY	Does refraction calculations for toric surfaces
	Subroutines Programmed into the Data General Eclipse	
CLOSE	LENR, LENW	Subroutine for closing a file
FOPEN	LENR, LENW	Subroutine for opening a file

Table 3. Walk-Through of ICARUS Program.

To log onto the system:

USER NAME:

SUN [sample user inputs are indicated in this special type face]

PASSWORD:

STEVE [this can be changed]

After much printing, user may execute the program by typing:

MXEQ ICARUS 15 [this is system dependent]

IS THIS A NEW LENS? [Y or N]

Y

WHAT IS ITS NAME? [up to 7 alphanumeric characters]

LENS10 [for example]

If this were not a new lens, the identifier statement would cause subroutine LENR (lens read) to be called; this routine reads in stored lens data from memory.

The following comes from subroutine STORE:

ENTER OBJECT TYPE [the choices are: SUN (specifies sun as the object), PNT (point source), LIN (specifies an object defined in terms of its linear dimensions, ANG (specifies an object defined in terms of angular subtense); entering LIN or ANG triggers the following interactions]

LIN

OBJECT DISTANCE IS:

100 [or whatever]

LINEAR FIELD(S) =

10 5 [height in y and x directions, respectively]

ANG

OBJECT DISTANCE IS:

1000 [or whatever]

ANGULAR FIELD(S) =

5 0 [subtense in y and x directions, respectively; angles are in degrees]

SUN

ENTER ENTRANCE PUPIL DIAMETER [currently only circular optics can be handled; the entrance pupil diameter for most solar collectors will be the collector diameter]

10

ENTER DIMENSIONS [MM or IN or M or CM or FT, etc.; this could be changed to ENTER UNITS. Currently, this is of no use except perhaps to help the designer remember; in the future, program additions could make use of this feature to scale output]

DO YOU WISH TO SPECIFY RADII (R) OR CURVATURES (C)? [R or C: since the program works in curvatures, if R is entered, all curve inputs will be divided into 1 to obtain curvatures; this pertains to surface data, toric curvatures, and Fresnel curvature]

HOW MANY LENS SURFACES ARE THERE? [generally, the answer will be 1, but the program can handle several; this number does not include object or image surfaces; the image is assumed to be plane and normal to the optical axis]

2

1st surface

ENTER SURFACE TYPE [options are as follows]

CON: a conic section (ellipse, sphere, parabola, hyperbola)
FNL: a normal Fresnel lens; aspheres are permitted
CFN: curved Fresnel, a Fresnel lens that does not lie on a plane; only spherical base curves are permitted
TOR: a toroid, a lens with different curvatures in the x and y directions
TFN: toric Fresnel, a Fresnel with differing x and y curvatures
CTF: curved toric Fresnel, a toric Fresnel that does not lie on a plane; only spherical base curves are permitted

CON

ENTER SURFACE DATA [respectively R, CC, TH, N_1, N_2, and N_3, defined as follows; as in input elsewhere, numbers are separated by a space]

R: radius of curvature or curvature of surface

CC: conic constant: $CC = 0 \rightarrow$ sphere, $CC = -1 \rightarrow$ paraboloid, $CC > 0 \rightarrow$ oblate spheroid, $CC < -1 \rightarrow$ hyperboloid, $0 > CC > -1 \rightarrow$ ellipsoid

TH distance to vertex of next surface

N_1, N_2, N_3 indices of refraction of three wavelengths

200 -2 10 1.523 1.518 1.527

ENTER DECENTERS AND TILTS [respectively XDEC, YDEC (x- and y-decenters in common units), α, β, γ (tilts in degrees); these are currently not operational, but soon to follow are program additions that will incorporate them; since they are currently disregarded by the program, 0 is a good entry]

0 5 0 0 0

ENTER RECENTERS AND BACKTILTS [respectively REC_1, REC_2, α', β', γ'; again, these are not currently operational]

0 0 0 0 0

2nd surface

ENTER SURFACE TYPE

CTF

ENTER SURFACE DATA

0 0 100 1 1

[first two entries negated by choice of surface type; if a nontoric is requested in preceding line, then the R appears as the first entry here]

ENTER DECENTERS AND TILTS

0 0 0 0 0

ENTER RECENTERS AND BACKTILTS [same sign as original decenter or tilt cancels the former]

0 5 0 0 0

ENTER TORIC CURVATURES [CVX, CVY (x curve and y curve); this question appears only if a toroid has been specified]

190 210 [curvatures (C) or radii (R), as previously specified]

ENTER CURVATURE OF FRESNEL [base curve; appears if Fresnel is specified]

-150 [curvature (C) or radius (R), as previously specified]

Subroutine LENW (lens write) is now called; it creates a file called by the name the user specifies, and writes the lens to it. Should the need arise, this file may be accessed for modification without using the program.

Return to main program:

WOULD YOU LIKE TO REVIEW YOUR LENS? [Y or N; Y (yes) triggers subroutine LENT, which displays the tabulated data on the lens; this step is used to see whether any errors have occurred or whether the designer wants to change anything]

Y

Subroutine LENT (lens type): displays the current lens on the terminal in tabular form.

WOULD YOU LIKE TO MAKE ANY CHANGES? [Y or N; Y triggers subroutine CHANGE, which will then ask for the changes you'd like to make]

Y

Subroutine CHANGE: interacts with the user to make individual changes in the lens without having to retype the entire lens; all user inputs are as in subroutine STORE.

WHAT WOULD YOU LIKE TO DO NOW? [options are RAY, LENT, CHANGE, PARAX, RED, NEW, QUIT; handled by subroutines described below]

RAY

Subroutine RAY: traces a specified ray through the lens. RAY in turn uses two subroutines, TORT and SYMM, to do the refraction calculations for toric surfaces and axially symmetrical surfaces, respectively. Interaction is as follows:

WHICH COLOR IS THE RAY, 1, 2, or 3? [1, 2, or 3: refers to N_1, N_2, and N_3]

ENTER FRACTIONAL OBJECT HEIGHTS [FOBY, FOBX: the normalized (full field = 1) height on the object from which the ray is to start; excluded if the object is SUN]

ENTER FRACTIONAL PUPIL COORDINATES [RHOY, RHOX, the normalized (half of the entrance pupil diameter = 1) ray height on the entrance pupil; can be any coordinates, not the 0.7, 1.0 rays, which are built in when RED is required. The output is as follows:

RAY COORDINATES AT IMAGE PLANE: X = , Y = , R =

$R = (X^2 + Y^2)^{1/2}$

LENT

Subroutine LENT (lens type): displays the current lens on the terminal in tabular form.

CHANGE

Subroutine CHANGE: interacts with the user to make individual changes in the lens without having to retype the entire lens; described above.

PARAX

Subroutine PARAX: does a paraxial ray trace that enables the user to locate the Gaussian image plane, and in general to do a rough lens design. One question is asked:

IS THIS IN COLOR 1, 2, or 3? [1, 2, or 3: determines whether N_1, N_2, or N_3 is used for each surface in the trace. Output is in the following form:

SURF	AXIAL Y	CHIEF Y	AXIAL U	CHIEF U
0	*****	.	.	.
1	.	.	.	.
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.

where SURF is the surface number (0 = object; the last surface is the image surface), AXIAL Y represents the height of the ray that passes through the center of the object and the edge of the entrance pupil, CHIEF Y is the height of the ray that passes through the edge of the object and the center of the entrance pupil, AXIAL U is the angle the axial ray makes with the optical axis prior to refraction at specified surface, and CHIEF U is the angle of the chief ray; angles are measured counterclockwise from the axis]

RED

Subroutine RED: calculates the radial energy distribution in the image plane. The object is always assumed to be the sun. Interaction is as follows:

HOW MANY COLORS WOULD YOU LIKE? [1, 2, or 3: If 1 is specified, index N_1 is used and 96 rays are traced through the system. If 2 is specified, 96 additional rays are traced using N_2 at each surface. If 3 is specified, 96 rays are traced using N_3 in addition to those with N_1 and N_2. To do this, subroutine RAY is used. The radial coordinate in the image plane is then stored, and subroutine SORT is called. This subroutine arranges the points in order of increasing radius, and puts them into 10 bins. Each bin is now assumed to contain 10% of the energy entering the system.]

Output is as follows, where r_n = the radius containing successive percentages of ray intercepts:

PERCENT ENERGY	RADIUS
10	r_1
20	r_2
30	r_3
⋮	⋮
100	r_{10}

NEW [start on a new lens (or a different old one)]

QUIT [terminates the program; to restart it, type MXEQ ICARUS 15 (as before)]

WHAT NEXT? [options are any of the above]

QUIT

To log off Eclipse:

BYE

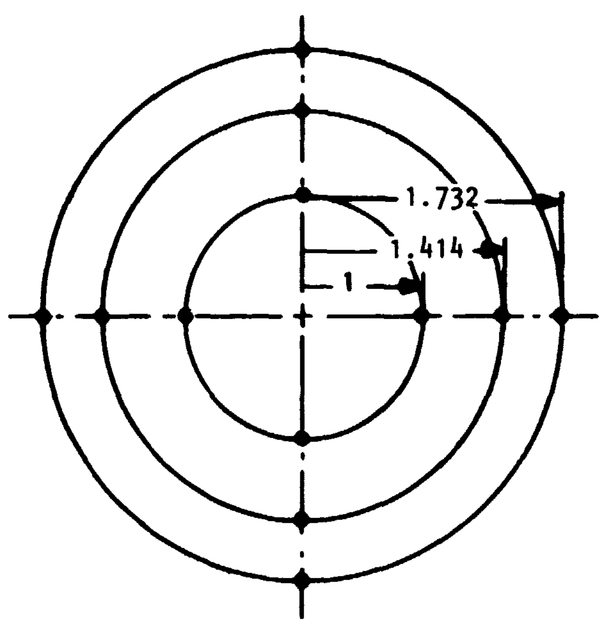
ORIGINAL PAGE IS
OF POOR QUALITY

Table 4. User's Notes on Program ICARUS.

---

1. The program takes 12 points to describe the sun, eight points on the lens in up to three wavelengths specified by three indices of refraction. This set then specifies 288 rays. If more points on the lens are desired, it will require recompiling the program and will involve considerably more computation time (unless a single wavelength is desired, thus permitting 24 points on the lens).

The pattern of points on the sun is as shown below. They are distributed in rings of equal area, each ring being weighted for a cosine solar limb-darkening effect. In reality, the power associated with limb darkening should be less than unity for closer approximations to the true sun, but for simplicity we have assumed a cos 1.0x relationship.



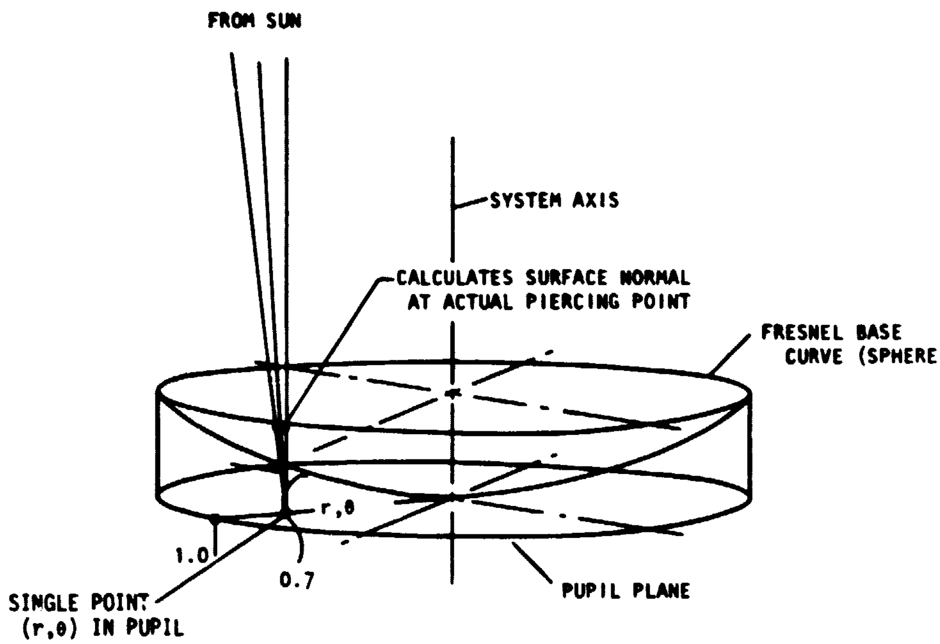
Pattern of points on the sun distributed
in weighted equal-area rings.

2. The choice of surface type encompasses plane and plane tilted (as in the faceted Fresnel panel concept).

ORIGINAL PAGE IS
OF POOR QUALITY

Note that the Fresnel surface has no finite grooves in this analytical program. The proper tilt of the Fresnel element at the point where the entering solar rays reach the lens surface is automatically calculated as discussed below.

The entrance pupil is in a plane normal to the vertex of the Fresnel base curve, as shown below. The present program takes eight rays: two radii (r = 0.7 and 1.0 the lens diameter at θ = each of 0, 90, 180, and 270 degrees. No central ray is traced.



Definition of the pupil plane and Fresnel base (substrate) curve.

The rays from the finite sun are directed to a single point (r,θ). When the rays are transferred to the specified Fresnel base curve, the r,θ values will be slightly different because the rays from the disc of the sun are tilted with regard to the normal at the pupil surface. The same effect arises from either a tracking error or a deliberate tilt.

The program calculates the normal at its intersection point of the Fresnel surface. The equivalent curvature of the Fresnel optical

ORIGINAL PAGE IS
OF POOR QUALITY

surface is calculated, as illustrated below. The equivalent Fresnel element surfaces are always considered as curved elements, curved to yield a point focus at the specified TH value given for the final optical surface (if more than one is involved).

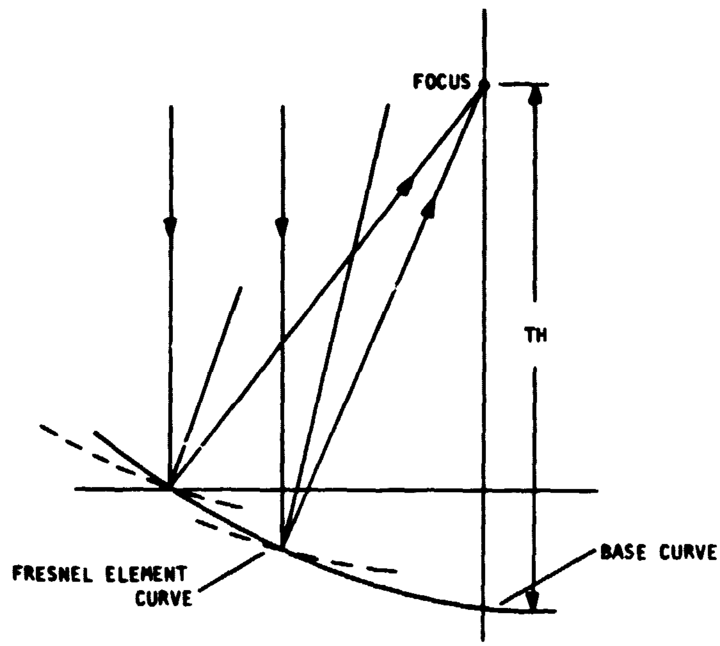


Illustration of the difference between the base (substrate) curvature and the equivalent element curvature.

The "proper tilt" of the Fresnel element has a subtle definition, as illustrated in the next figure. The equivalent Fresnel curvature has a constant value of RD (or CV) and CC, but is shifted by an amount Δ so that the slope at r brings the reflected ray at r to the focus. In other words, the program shifts the conic.

The radius and curvature are defined by the focal length F or the power P, by

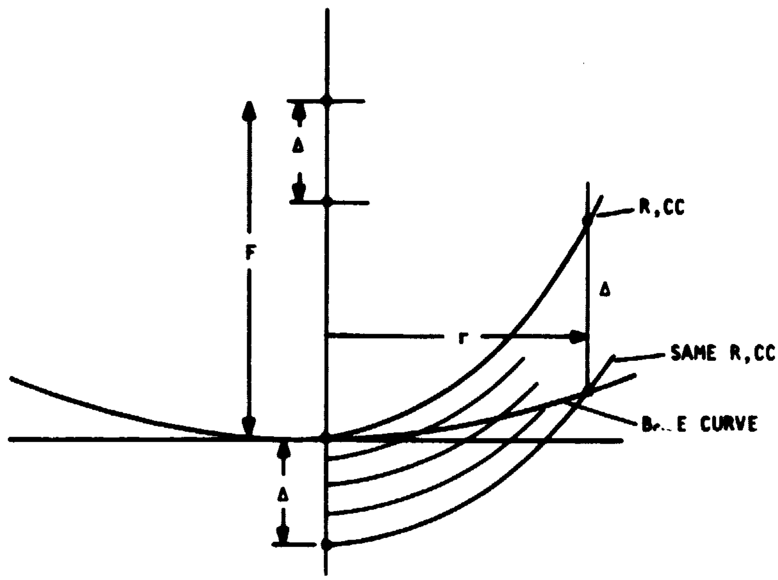
$$CV = P/2$$

$$P = 1/F$$

RD equals 2F for a reflector, or

$$\left.\begin{aligned} 1/F &= (n' - n)\ (1/r_1 - 1/r_2) \\ P &= (n' - n)\ (CV_1 - CV_2) \end{aligned}\right\} \text{ for a lens.}$$

ORIGINAL PAGE IS
OF POOR QUALITY



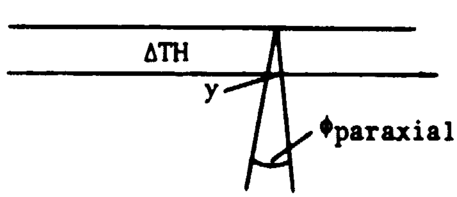
Geometry illustrating the shift in the
effective conic surface defining facet angles.

There is a strategy for finding the best value of CC for the equivalent facets, depending on the base surface used. For example, the best value of CC for a plane reflective Fresnel mirror is a hyperboloid, CC = -3.0. The strategy is to try a CC and examine the ray distribution in the focal surface, then change CC to find the best distribution.

The curvature is the inverse of the radius of curvature. The conic constant is:

hyperboloid	$CC < -1$
paraboloid	$CC = -1$
ellipse	$-1 \le CC \le 0$
sphere	$CC = 0$

3. To find a better value of TH for the focus, take the output value of the y intercept with the TH plane and paraxial angle ϕ, and calculate. Calculate an unproved TH using a pocket calculator.



$$\Delta TH = y\phi$$

ORIGINAL PAGE IS
OF POOR QUALITY

Table 5. Source Code Listing of Program ICARUS.

```
      PARAMETER NRAY=96,IF=11,OF=10,NF=2
      COMMON/NAME/ANAME,IREC,LG,IC,NCOL
      COMMON/OD/OBJ,DIM,EPD,LS,ANY,ANX,LNY,LNX,TH0
      COMMON/SD/SUR(10),CV(10),CC(10),TH(10),N(11,3)
      COMMON/TD/DEC(10,2),TILT(10,3),RDEC(10,2),RTILT(10,3),CVY(10),CVX(10)
      COMMON/TD/CVF(10)
      COMMON/RAYS/FOBY(NRAY),FOBX(NRAY),XTA(NRAY),YTA(NRAY),RA(NRAY),RAD(10)
      COMMON/RAYS/R(NRAY,3)
      REAL N,LNX,LNY
      LOGICAL LG
10    TYPE"IS THIS A NEW LENS?"
      READ(IF,900) IAGE
      TYPE"WHAT IS ITS NAME?"
      READ(IF,905) ANAME
      IF(IAGE.EQ."Y") CALL STORE
      IF(IAGE.EQ."N") CALL LENR
      TYPE"WOULD YOU LIKE TO REVIEW YOUR LENS?"
      READ(IF,900) LR
      IF(LR.EQ."Y") CALL LENT
      TYPE"WOULD YOU LIKE TO MAKE ANY CHANGES?"
      READ(IF,900) ICQ
      IF(ICQ.EQ."Y") CALL CHANGE
      TYPE"WHAT WOULD YOU LIKE TO DO NOW?"
      READ(IF,905) WORK
20    IF(WORK.EQ."RAY") GOTO 100
      IF(WORK.EQ."LENT") GOTO 110
      IF(WORK.EQ."PARAX") GOTO 120
      IF(WORK.EQ."RED") GOTO 130
      IF(WORK.EQ."CHANGE") GOTO 150
      IF(WORK.EQ."NEW") GOTO 30
      IF(WORK.EQ."QUIT") GOTO 40
      TYPE"SORRY, I DON'T KNOW HOW."
25    TYPE"WHAT NEXT?"
      READ(IF,905) WORK
      GOTO 20
100   LG=.TRUE.
      CALL RAY
      GOTO 25
110   CALL LENT
      GOTO 25
120   CALL PARAX
      GOTO 25
130   CALL RED
      GOTO 25
150   CALL CHANGE
      GOTO 25
30    GOTO 10
40    CONTINUE
900   FORMAT(S1)
905   FORMAT(S7)
      END
      SUBROUTINE STORE
      PARAMETER NRAY=96,IF=11,OF=10,NF=2
      COMMON/NAME/ANAME,IREC,LG,IC,NCOL
      COMMON/OD/OBJ,DIM,EPD,LS,ANY,ANX,LNY,LNX,TH0
      COMMON/SD/SUR(10),CV(10),CC(10),TH(10),N(11,3)
      COMMON/TD/DEC(10,2),TILT(10,3),RDEC(10,2),RTILT(10,3),CVY(10),CVX(10)
      COMMON/TD/CVF(10)
      COMMON/RAYS/FOBY(NRAY),FOBX(NRAY),XTA(NRAY),YTA(NRAY),RA(NRAY),RAD(10)
      COMMON/RAYS/R(NRAY,3)
      REAL N,LNY,LNX
```

ORIGINAL PAGE IS
OF POOR QUALITY

Table 5. Source Code Listing of Program ICARUS (Continued).

```
      TYPE"ENTER OBJECT TYPE"
      READ(IF,999) OBJ
      IF(OBJ.EQ."SUN") THØ=1.ØE+1Ø
      IF(OBJ.EQ."SUN") GOTO 1Ø
      ACCEPT"OBJECT DISTANCE IS ",THØ
      IF(OBJ.EQ."ANG") ACCEPT"ANGULAR FIELD(S) = ",ANY,ANX
      LNY=THØ*TAN(ANY/57.296)
      LNX=THØ*TAN(ANX/57.296)
      IF(OBJ.EQ."ANG") GOTO 1Ø
      IF(OBJ.EQ."LIN") ACCEPT "LINEAR FIELD(S) = ",LNY,LNX
      ANY=ATAN(LNY/THØ)*57.296
      ANX=ATAN(LNX/THØ)*57.296
1Ø    ACCEPT "ENTER ENTRANCE PUPIL DIAMETER ",EPD
      TYPE "ENTER DIMENSIONS"
      READ(IF,999) DIM
      TYPE "DO YOU WISH TO SPECIFY RADII (R) OR CURVATURES (C)?"
      READ(IF,998) IRQ
      ACCEPT "HOW MANY LENS SURFACES ARE THERE?",LS
      DO 2Ø I=1,LS
      TYPE "ENTER SURFACE TYPE"
      READ(IF,999) SUR(I)
      ACCEPT "ENTER SURFACE DATA: ",CV(I),CC(I),TH(I),N(I,1),N(I,2),N(I,3)
      ACCEPT"ENTER DECENTERS AND TILTS ",(DEC(I,J),J=1,2),(TILT(I,J),J=1,3)
      ACCEPT"ENTER RECENTERS AND BACKTILTS ",(RDEC(I,J),J=1,2),(RTILT(I,J),
     1 J=1,3)
      IF(SUR(I).EQ."TOR".OR.SUR(I).EQ."TFN".OR.SUR(I).EQ."CTF") ACCEPT
     1 "ENTER TORIC CURVATURES ",CVX(I),CVY(I)
      IF(SUR(I).EQ."CFN".OR.SUR(I).EQ."CTF") ACCEPT"ENTER CURVATURE OF
     1 FRESNEL ",CVF(I)
2Ø    CONTINUE
      IF(IRQ.EQ."C") GOTO 4Ø
      DO 3Ø J=1,LS
      IF(CV(J).EQ.Ø) GOTO 22
      CV(J)=1/CV(J)
22    IF(CVY(J).EQ.Ø) GOTO 24
      CVY(J)=1/CVY(J)
24    IF(CVX(J).EQ.Ø) GOTO 26
      CVX(J)=1/CVX(J)
26    IF(CVF(J).EQ.Ø) GOTO 3Ø
      CVF(J)=1/CVF(J)
3Ø    CONTINUE
4Ø    CALL LENW
998   FORMAT(S1)
999   FORMAT(S3)
      RETURN
      END
      SUBROUTINE LENR
      PARAMETER NRAY=96,IF=11,OF=1Ø,NF=2
      COMMON/NAME/ANAME,IREC,LG,IC,NCOL
      COMMON/OD/OBJ,DIM,EPD,LS,ANY,ANX,LNY,LNX,THØ
      COMMON/SD/SUR(1Ø),CV(1Ø),CC(1Ø),TH(1Ø),N(11,3)
      COMMON/TD/DEC(1Ø,2),TILT(1Ø,3),RDEC(1Ø,2),RTILT(1Ø,3),CVY(1Ø),CVX(1Ø)
      COMMON/TD/CVF(1Ø)
      COMMON/RAYS/FOBY(NRAY),FOBX(NRAY),XTA(NRAY),YTA(NRAY),RA(NRAY),RAD(1Ø)
      COMMON/RAYS/R(NRAY,3)
      REAL N,LNX,LNY
      CALL FOPEN(NF,ANAME)
      READ(NF,999) ANAME
      READ(NF,998) OBJ,DIM,EPD,LS,ANY,ANX,LNY,LNX,THØ
      DO 1Ø I=1,LS
      READ(NF,997) SUR(I),CV(I),CC(I),TH(I),N(I,1),N(I,2),N(I,3)
```

ORIGINAL PAGE IS
OF POOR QUALITY

Table 5. Source Code Listing of Program ICARUS (Continued).

```
      READ(NF,996) (DEC(I,J),J=1,2),(TILT(I,J),J=1,3),(RDEC(I,J),J=1,2),
     1 (RTILT(I,J),J=1,3)
      READ(NF,995) CVY(I),CVX(I),CVF(I)
10    CONTINUE
995   FORMAT(1X,3F10.6)
996   FORMAT(1X,10F10.6)
997   FORMAT(1X,S3,6F12.6)
998   FORMAT(1X,S3,2X,S3,4F10.6,3E10.4)
999   FORMAT(1X,S4)
      CALL CLOSE(NF,IER)
      RETURN
      END
      SUBROUTINE LENW
      PARAMETER NRAY=96,IF=11,OF=10,NF=2
      COMMON/NAME/ANAME,IREC,LG,IC,NCOL
      COMMON/OD/OBJ,DIM,EPD,LS,ANY,ANX,LNY,LNX,TH0
      COMMON/SD/SUR(10),CV(10),CC(10),TH(10),N(11,3)
      COMMON/TD/DEC(10,2),TILT(10,3),RDEC(10,2),RTILT(10,3),CVY(10),CVX(10)
      COMMON/TD/CVF(10)
      COMMON/RAYS/FOBY(NRAY),FOBX(NRAY),XTA(NRAY),YTA(NRAY),RA(NRAY),RAD(10)
      COMMON/RAYS/R(NRAY,3)
      REAL N,LNY,LNX
      CALL FOPEN(NF,ANAME)
      WRITE(NF,999) ANAME
      WRITE(NF,998) OBJ,DIM,EPD,LS,ANY,ANX,LNY,LNX,TH0
      DO 10 I=1,LS
      WRITE(NF,997) SUR(I),CV(I),CC(I),TH(I),N(I,1),N(I,2),N(I,3)
      WRITE(NF,996) (DEC(I,J),J=1,2),(TILT(I,J),J=1,3),(RDEC(I,J),J=1,2),
     1 (RTILT(I,J),J=1,3)
      WRITE(NF,995) CVY(I),CVX(I),CVF(I)
10    CONTINUE
995   FORMAT(1X,3F10.6)
996   FORMAT(1X,10F10.6)
997   FORMAT(1X,S3,6F12.6)
998   FORMAT(1X,S3,2X,S3,4F10.6,3E10.4)
999   FORMAT(1X,S4)
      CALL CLOSE(NF,IER)
      RETURN
      END
      SUBROUTINE LENT
      PARAMETER NRAY=96,IF=11,OF=10,NF=2
      COMMON/NAME/ANAME,IREC,LG,IC,NCOL
      COMMON/OD/OBJ,DIM,EPD,LS,ANY,ANX,LNY,LNX,TH0
      COMMON/SD/SUR(10),CV(10),CC(10),TH(10),N(11,3)
      COMMON/TD/DEC(10,2),TILT(10,3),RDEC(10,2),RTILT(10,3),CVY(10),CVX(10)
      COMMON/TD/CVF(10)
      COMMON/RAYS/FOBY(NRAY),FOBX(NRAY),XTA(NRAY),YTA(NRAY),RA(NRAY),RAD(10)
      COMMON/RAYS/R(NRAY,3)
      REAL N,LNY,LNX
      DIMENSION RD(10)
      WRITE(OF,999) ANAME
      WRITE(OF,899)
      WRITE(OF,998) OBJ,DIM
      WRITE(OF,898)
      WRITE(OF,997) EPD,ANY,ANX,LNY,LNX,TH0
      WRITE(OF,897)
      DO 10 I=1,LS
      IF(CV(I).EQ.0) RD(I)=1.0E+10
      IF(CV(I).NE.0) RD(I)=1/CV(I)
10    CONTINUE
      DO 20 I=1,LS
```

ORIGINAL PAGE IS
OF POOR QUALITY

Table 5. Source Code Listing of Program ICARUS (Continued).

```
      WRITE(OF,996) CV(I),RD(I),CC(I),TH(I),N(I,1),N(I,2),N(I,3)
20    CONTINUE
      WRITE(OF,896)
      K=0
      DO 40 I=1,LS
      SUM=DEC(I,1)**2+DEC(I,2)**2+TILT(I,1)**2+TILT(I,2)**2+TILT(I,3)**2
      IF(SUM.EQ.0) GOTO 25
      WRITE(OF,995) (DEC(I,J),J=1,2),(TILT(I,J),J=1,3)
25    SUM=RDEC(I,1)**2+RDEC(I,2)**2+RTILT(I,1)**2+RTILT(I,2)**2+RTILT(I,3)**2
      IF(SUM.EQ.0) GOTO 30
      WRITE(OF,994) (RDEC(I,J),J=1,2),(RTILT(I,J),J=1,3)
      GOTO 40
30    K=K+1
40    CONTINUE
      IF(K.EQ.LS) WRITE(OF,895)
      K=0
      WRITE(OF,894)
      DO 60 I=1,LS
      IF(SUR(I).NE."CFN".AND.SUR(I).NE."CTF".AND.SUR(I).NE."TOR".AND.SUR(I)
     1 .NE."TFN") GOTO 50
      WRITE(OF,993) CVY(I),CVX(I),CVF(I)
      GOTO 60
50    K=K+1
60    CONTINUE
      IF(K.EQ.LS) WRITE(OF,895)
894   FORMAT(2X,"Y CURVE",2X,"X CURVE",2X,"FRNL CURVE")
895   FORMAT(1X,"NONE")
896   FORMAT(2X,"Y DECENTER",2X,"X DECENTER",5X,"ALPHA",5X,"BETA",5X,"GAMMA")
897   FORMAT(1X,"CURVATURE",3X,"RADIUS",3X,"CONIC",2X,"THICKNESS",2X,"N1",4X,
     1 "N2",4X,"N3")
898   FORMAT(6X,"EPD",7X,"ANGULAR FIELDS",4X,"LINEAR FIELDS",5X,"OBJ. DISTANCE
     1 ")
899   FORMAT(1X,"OBJECT  UNITS")
993   FORMAT(1X,3F8.6)
994   FORMAT(1X,5F11.4)
995   FORMAT(1X,5F11.4)
996   FORMAT(1X,2F10.6,F6.3,F10.4,3F7.4)
997   FORMAT(1X,5F10.4,E10.4)
998   FORMAT(3X,S3,5X,S3)
999   FORMAT(1X,S7)
      RETURN
      END


      SUBROUTINE CHANGE
      PARAMETER NRAY=96,IF=11,OF=10,NF=2
      COMMON/NAME/ANAME,IREC,LG,IC,NCOL
      COMMON/OD/OBJ,DIM,EPD,LS,ANY,ANX,LNY,LNX,TH0
      COMMON/SD/SUR(10),CV(10),CC(10),TH(10),N(11,3)
      COMMON/TD/DEC(10,2),TILT(10,3),RDEC(10,2),RTILT(10,3),CVY(10),CVX(10)
      COMMON/TD/CVF(10)
      COMMON/RAYS/FOBY(NRAY),FOBX(NRAY),XTA(NRAY),YTA(NRAY),RA(NRAY),RAD(10)
      COMMON/RAYS/R(NRAY,3)
      REAL N,LNY,LNX
      INTEGER RO
      TYPE"WOULD YOU LIKE TO CHANGE THE OBJECT?"
      READ(IF,999) ITO
      IF(ITO.EQ."N") GOTO 10
      TYPE"ENTER NEW OBJECT"
      READ(IF,998) OBJ
10    TYPE"WOULD YOU LIKE TO CHANGE THE UNITS?"
```

ORIGINAL PAGE IS
OF POOR QUALITY

Table 5. Soruce Code Listing of Program ICARUS (Continued).

```
      READ(IF,999) ITQ
      IF(ITQ.EQ."N") GOTO 2Ø
      TYPE"ENTER NEW UNITS"
      READ(IF,998) DIM
2Ø    TYPE"WOULD YOU LIKE TO CHANGE OBJECT DISTANCE?"
      READ(IF,999) ITQ
      IF(ITQ..EQ."N") GOTO 3Ø
      ACCEPT"NEW OBJECT DISTANCE = ",THØ
3Ø    TYPE"WOULD YOU LIKE TO CHANGE EPD?"
      READ(IF,999) ITQ
      IF(ITQ.EQ."N") GOTO 4Ø
      ACCEPT"ENTER NEW EPD",EPD
4Ø    TYPE"WOULD YOU LIKE TO CHANGE FIELD SIZE?"
      READ(IF,999) ITQ
      IF(ITQ.EQ."N") GOTO 6Ø
      IF(OBJ.EQ."LIN") GOTO 5Ø
      ACCEPT"ANGULAR FIELDS = ",ANY,ANX
      LNY=THØ*TAN(ANY/57.296)
      LNX=THØ*TAN(ANX/57.296)
      GOTO 6Ø
5Ø    ACCEPT"LINEAR FIELDS = ",LNY,LNX
      ANY=57.296*ATAN(LNY/THØ)
      ANX=57.296*ATAN(LNX/THØ)
6Ø    TYPE"DO YOU WISH TO SPECIFY RADII (R) OR CURVATURES (C)?"
      READ (IF,999) RQ
      TYPE"DOES THE LENS STILL HAVE",LS,"  SURFACES?"
      READ (IF,999) ITQ
      IF(ITQ.EQ."Y") GOTO 7Ø
      ACCEPT"ENTER THE NUMBER OF SURFACES",LS
7Ø    DO 9Ø I=1,LS
      TYPE"DO YOU WISH TO CHANGE SURFACE ",I,"?"
      READ(IF,999) ITQ
      IF (ITQ.EQ."N") GOTO 9Ø
      TYPE"ENTER SURFACE TYPE"
      READ(IF,998) SUR(I)
      ACCEPT"ENTER SURFACE DATA: ",CV(I),CC(I),TH(I),N(I,1),N(I,2),N(I,3)
      ACCEPT"ENTER DECENTER DATA: ",(DEC(I,J),J=1,2),(TILT(I,J),J=1,3)
      ACCEPT"ENTER RECENTER DATA: ",(RDEC(I,J),J=1,2),(RTILT(I,J),J=1,3)
      IF(SUR(I).EQ."TOR".OR.SUR(I).EQ."TFN".OR.SUR(I).EQ."CTF")
     1 ACCEPT"ENTER TORIC CURVATURES",CVY(I),CVX(I)
      IF(SUR(I).EQ."CFN".OR.SUR(I).EQ."CTF") ACCEPT"ENTER FRNL CURVE",CVF(I)
      IF(RQ.EQ."C") GOTO 9Ø
      IF(CV(I).EQ.Ø) GOTO 82
      CV(I)=1/CV(I)
82    IF(CVY(I).EQ.Ø) GOTO 84
      CVY(I)=1/CVY(I)
84    IF(CVX(I).EQ.Ø) GOTO 86
      CVX(I)= 1/CVX(I)
86    IF(CVF(I).EQ.Ø) GOTO 9Ø
      CVF(I)=1/CVF(I)
9Ø    CONTINUE
1ØØ   CALL LENW
998   FORMAT(S3)
999   FORMAT(S1)
      RETURN
      END
      SUBROUTINE PARAX
      PARAMETER NRAY=96,IF=11,OF=1Ø,NF=2,LM=LS+2
      COMMON/NAME/ANAME,IREC,LG,IC,NCOL
      COMMON/OD/OBJ,DIM,EPD,LS,ANY,ANX,LNY,LNX,THØ
      COMMON/SD/SUR(1Ø),CV(1Ø),CC(1Ø),TH(1Ø),N(11,3)
```

ORIGINAL PAGE IS
OF POOR QUALITY

Table 5. Source Code Listing of Program ICARUS (Continued).

```
      COMMON/TD/DEC(10,2),TILT(10,3),RDEC(10,2),RTILT(10,3),CVY(10),CVX(10)
      COMMON/TD/CVF(10)
      COMMON/RAYS/FOBY(NRAY),FOBX(NRAY),XTA(NRAY),YTA(NRAY),RA(NRAY),RAD(10)
      COMMON/RAYS/R(NRAY,3)
      DIMENSION Y(12),CY(12),U(12),CU(12),PHI(12)
      REAL N,LNY,LNX
      ACCEPT "IS THIS IN COLOR 1,2,OR 3?",J
      Y(0)=0
      CY(0)=-LNY
      U(0)=EPD/(2*TH0)
      CU(0)=-LNY/TH0
      PHI(0)=0
      Y(1)=EPD/2
      CY(1)=0
      U(1)=(U(0)-Y(1)*(N(1,J)-1)*CV(1))/N(1,J)
      CU(1)=CU(0)/N(1,J)
      PHI(1)=(N(1,J)-1)*CV(1)
      DO 10 I=2,LS+1
      Y(I)=Y(I-1)+TH(I-1)*U(I-1)
      CY(I)=CY(I-1)+TH(I-1)*CU(I-1)
      IF(I.EQ.LS+1) GOTO 10
      PHI(I)=(N(I,J)-N(I-1,J))*CV(I)
      U(I)=(N(I-1,J)*U(I-1)-Y(I)*PHI(I))/N(I,J)
      CU(I)=(N(I-1,J)*CU(I-1)-CY(I)*PHI(I))/N(I,J)
      N(I,J)=ABS(N(I,J))
10    CONTINUE
      PHI(LS+1)=0
      U(LS+1)=U(LS)
      CU(LS+1)=CU(LS)
      WRITE(OF,999)
      DO 20 I=1,LS+2
      K=I-1
      WRITE(OF,998) K,Y(K),CY(K),U(K),CU(K),PHI(K)
20    CONTINUE
998   FORMAT(1X,I4,5F10.6)
999   FORMAT(1X,"SURF",2X,"AXIAL Y",3X,"CHIEF Y",3X,"AXIAL U",3X,"CHIEF U",
     1 5X,"POWER")
      CALL LENR
      RETURN
      END
      SUBROUTINE RAY
      PARAMETER NRAY=96,IF=11,OF=10,NF=2
      COMMON/NAME/ANAME,IREC,LG,IC,NCOL
      COMMON/OD/OBJ,DIM,EPD,LS,ANY,ANX,LNY,LNX,TH0
      COMMON/SD/SUR(10),CV(10),CC(10),TH(10),N(11,3)
      COMMON/TD/DEC(10,2),TILT(10,3),RDEC(10,2),RTILT(10,3),CVY(10),CVX(10)
      COMMON/TD/CVF(10)
      COMMON/RAYS/FOBY(NRAY),FOBX(NRAY),XTA(NRAY),YTA(NRAY),RA(NRAY),RAD(10)
      COMMON/RAYS/R(NRAY,3)
      COMMON/TRACE/YT,XT,A,L,M,ND
      COMMON/PUP/ RHOY(NRAY),RHOX(NRAY)
      REAL N,L,M,ND,MU,LNY,LNX
      LOGICAL LG,LG2
      IF(.NOT.LG) NR=NRAY
      IF(.NOT.LG) GOTO 30
      FOBX(1)=0
      FOBY(1)=0
      NR=1
      ACCEPT "WHICH COLOR IS THE RAY, 1,2 OR 3?",JC
      N(0,IC)=1
      IF(OBJ.EQ."SUN") GOTO 10
```

ORIGINAL PAGE IS
OF POOR QUALITY

Table 5. Source Code Listing of Program ICARUS (Continued).

```
      ACCEPT "ENTER FRACTIONAL OBJECT HEIGHTS",FOBY(1),FOBX(1)
      GOTO 20
10    LNY=0
      LNX=0
      ANY=0
      ANX=0
20    ACCEPT "ENTER FRACTIONAL PUPIL COORDINATES",RHOY(1),RHOX(1)
30    DO 200 I=1,NR
      LG2=.FALSE.
      HX=LNX*FOBX(I)
      HY=LNY*FOBY(I)
      YT=RHOY(I)*EPD/2
      XT=RHOX(I)*EPD/2
      A=SQRT(TH0**2+(HY-YT)**2+(HX-XT)**2)
      L=(HX-XT)/A
      M=(HY-YT)/A
      ND=TH0/A
      DO 190 J=1,LS
      IF(SUR(J).EQ."TOR".OR.SUR(J).EQ."CTF".OR.SUR(J).EQ."TFN") CALL TORT(J,LG2)
      IF(SUR(J).EQ."CON".OR.SUR(J).EQ."FNL".OR.SUR(J).EQ."CFN") CALL SYMM(J,LG2)
      IF(LG2) GOTO 200
190   CONTINUE
      RXY=SQRT(XT**2+YT**2)
      IF(LG) TYPE "RAY COORDINATES AT IMAGE PLANE:  X= ",XT,"  Y= ",YT,"  R= "
     1 ,RXY
      IF(LG) GOTO 200
      XTA(I)=XT
      YTA(I)=YT
      RA(I)=RXY
200   CONTINUE
      RETURN
      END

      SUBROUTINE TORT(J,LG2)
      PARAMETER NRAY=96,IF=11,OF=10,NF=2
      COMMON/NAME/ANAME,IREC,LG,IC,NCOL
      COMMON/OD/OBJ,DIM,EPD,LS,ANY,ANX,LNY,LNX,TH0
      COMMON/SD/SUR(10),CV(10),CC(10),TH(10),N(11,3)
      COMMON/TD/DEC(10,2),TILT(10,3),RDEC(10,2),RTILT(10,3),CVY(10),CVX(10)
      COMMON/TD/CVF(10)
      COMMON/RAYS/FOBY(NRAY),FOBX(NRAY),XTA(NRAY),YTA(NRAY),RA(NRAY),RAD(10)
      COMMON/RAYS/R(NRAY,3)
      COMMON/TRACE/YT,XT,A,L,M,ND
      COMMON/PUP/ RHOY(NRAY),RHOX(NRAY)
      REAL N,L,M,ND,MU,LNY,LNX
      LOGICAL LG,LG2
      Y=Y+DEC(J,1)-RDEC(J,1)
      X=X+DEC(J,2)-RDEC(J,2)
      N=SIN(ASIN(N)+TILT(J,1)/59.29578)
      M=SIN(ASIN(M)+TILT(J,2)/57.29578)
      L=SIN(ASIN(L)+TILT(J,3)/57.29578-RTILT(J,3)/57.29578)
      M=SIN(ASIN(M)-RTILT(J,2)/57.29578)
      N=SIN(ASIN(N)-RTILT(J,1)/57 29578)
      IF (SUR(J).EG."CTF") GOTO 10
      BX=CVX(J)*XT**2
      BY=CVY(J)*YT**2
      CX=ND-CVX(J)*L*XT
      CY=ND-CVY(J)*M*YT
      DSX=CX**2-(CVX(J)*XT)**2
      DSY=CY**2-(CVY(J)*YT)**2
      GOTO 20
```

ORIGINAL PAGE IS
OF POOR QUALITY

Table 5. Source Code Listing of Program ICARUS (Continued).

```
10      BX=CVF(J)*(XT**2+YT**2)
        BY=BX
        CX=ND-CVF(J)*(L*XT+M*YT)
        CY=CX
        DSX=CX**2-CVF(J)*BX
        DSY=DSX
20      IF(ABS(DSX).LE.1E-10) DSX=0
        IF(ABS(DSY).LE.1E-10) DSY=0
        DX=SQRT(DSX)
        DY=SQRT(DSY)
        A=(BX/(CX+DX)+BY/(CY+DY))/2
        IF(SUR(J).EQ."TFN") A=0
        XS=XT+A*L
        YS=YT+A*M
        ZS=A*ND
        CSIX=DX
        CSIY=DY
        MU=N(J-1,IC)/ABS(N(J,IC))
        DEXS=1-MU**2*(1-CSIX**2)
        DEYS=1-MU**2*(1-CSIY**2)
        IF(DEXS.LE.0.OR.DEYS.LE.0) GOTO 70
        DEX=SQRT(DEXS)-MU*CSIX
        DEY=SQRT(DEYS)-MU*CSIY
        ALPHA=-CVX(J)*XS
        BETA=-CVY(J)*YS
        L=MU*L+DEX*ALPHA
        M=MU*M+DEY*BETA
        ND=SQRT(1-L**2-M**2)
        XT=XS+L*(TH(J)-ZS)/ND
        YT=YS+M*(TH(J)-ZS)/ND
        A=SQRT((TH(J)-ZS)**2+(YT-YS)**2+(XT-XS)**2)
        GOTO 95
70      IF(LG) TYPE"TOTAL INTERNAL REFLECTION AT SURFACE",J
        GOTO 90
80      IF(LG) TYPE"RAY MISSED SURFACE",J
90      LG2=.TRUE.
95      RETURN
        END

        SUBROUTINE SYMM(J,LG2)
        PARAMETER NRAY=96,IF=11,OF=10,NF=2
        COMMON/NAME/ANAME,IREC,LG,IC,NCOL
        COMMON/OD/OBJ,DIM,EPD,LS,ANY,ANX,LNY,LNX,TH0
        COMMON/SD/SUR(10),CV(10),CC(10),TH(10),N(11,3)
        COMMON/TD/DEC(10,2),TILT(10,3),RDEC(10,2),RTILT(10,3),CVY(10),CVX(10)
        COMMON/TD/CVF(10)
        COMMON/RAYS/FOBY(NRAY),FOBX(NRAY),XTA(NRAY),YTA(NRAY),RA(NRAY),RAD(10)
        COMMON/RAYS/R(NRAY,3)
        COMMON/TRACE/YT,XT,A,L,M,ND
        COMMON/PUP/ RHOY(NRAY),RHOX(NRAY)
        REAL N,L,M.ND,MU,LNY,LNX
        LOGICAL LG,LG2
        Y=Y+DEC(J,1)-RDEC(J,1)
        X=X+DEC(J,2)-RDEC(J,2)
        N=SIN(ASIN(N)+TILT(J,1)/57.29578)
        M=SIN(ASIN(M)+TILT(J,2)/57.29578)
        L=SIN(ASIN(L)+TILT(J,3)/57.29578-RTILT(J,3)/57.29578)
        M=SIN(ASIN(M)-RTILT(J,2)/57.29578)
        N=SIN(ASIN(N)-RTILT(J,1)/57.29578)
        IF(SUR(J).EQ."CFN") GOTO 10
        B=CV(J)*(XT**2+YT**2)
```

ORIGINAL PAGE IS
OF POOR QUALITY

Table 5. Source Code Listing of Program ICARUS (Continued).

```
      C=ND-CV(J)*(L*XT+M*YT)
      DS=C*C-CV(J)*B*(1+CC(J)*ND*ND)
      GOTO 20
10    B=CVF(J)*(XT**2+YT**2)
      C=ND-CVF(J)*(L*XT+M*YT)
      DS=C*C-CVF(J)*B
20    IF(ABS(DS).LE.1E-10) DS=0
      D=SQRT(DS)
      A=B/(C+D)
      IF(OBJ.EQ."FNL") A=0
      XS=XT+A*L
      YS=YT+A*M
      ZS=A*ND
      E2=1/(1-2*CV(J)*CC(J)*ZS)
      E=SQRT(E2)
      CSI=D*E
      MU=N(J-1,IC)/ABS(N(J,IC))
      DELS=1-MU**2*(1-CSI**2)
      IF(DELS.LT.0) GOTO 70
      DEL=SQRT(DELS)-MU*CSI
      ALPHA=-CV(J)*E*XS
      BETA=-CV(J)*E*YS
      GAMMA=(1-CV(J)*(CC(J)+1)*ZS)*E
      L=MU*L+DEL*ALPHA
      M=MU*M+DEL*BETA
      ND=MU*ND+DEL*GAMMA
      XT=XS+L*(TH(J)-ZS)/ND
      YT=YS+M*(TH(J)-ZS)/ND
      A=SQRT((TH(J)-ZS)**2+(YT-YS)**2+(XT-XS)**2)
      GOTO 95
70    IF(LG) TYPE"TOTAL INTERNAL REFLECTION AT SURFACE ",J
      GOTO 90
80    IF(LG) TYPE "RAY MISSED SURFACE ",J
90    LG2=.TRUE.
95    RETURN
      END



      SUBROUTINE RED
      PARAMETER NRAY=96,IF=11,OF=10,NF=2
      COMMON/NAME/ANAME,IREC,LG,IC,NCOL
      COMMON/OD/OBJ,DIM,EPD,LS,ANY,ANX,LNY,LNX,TH0
      COMMON/SD/SUR(10),CV(10),CC(10),TH(10),N(11,3)
      COMMON/TD/DEC(10,2),TILT(10,3),RDEC(10,2),RTILT(10,3),CVY(10),CVX(10)
      COMMON/TD/CVF(10)
      COMMON/RAYS/FOBY(NRAY),FOBX(NRAY),XTA(NRAY),YTA(NRAY),RA(NRAY),RAD(10)
      COMMON/RAYS/R(NRAY,3)
      COMMON/PUP/RHOY(NRAY),RHOX(NRAY)
      REAL N,LNY,LNX
      LOGICAL LG
      DIMENSION FY(12),FX(12),RY(8),RX(8)
      DATA FY/.31,0,-.31,0,.57,0,-.57,0,.89,0,-.89,0/
      DATA FX/0,.31,0,-.31,0,.57,0,-.57,0,.89,0,-.89/
      DATA RY/.7,0,-.7,0,1,0,-1,0/
      DATA RX/0,.7,0,-.7,0,1,0,-1/
      LG=.FALSE.
      LNX=4.36E+7
      LNY=4.36E+7
      DO 20 I=1,12
```

ORIGINAL PAGE IS
OF POOR QUALITY

Table 5. Source Code Listing of Program ICARUS (Continued).

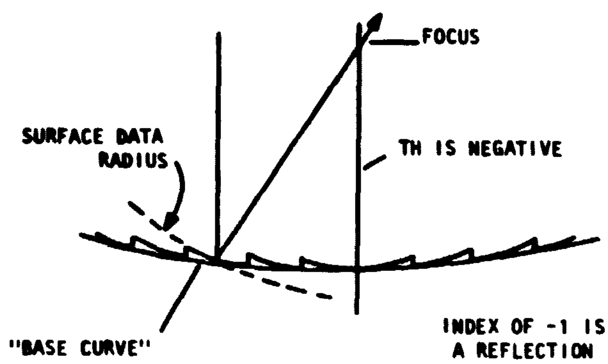
```
      DO 10 J=1,8
      K=8*(I-1)+J
      FOBY(K)=FY(I)
      FOBX(K)=FX(I)
      RHOY(K)=RY(J)
      RHOX(K)=RX(J)
10    CONTINUE
20    CONTINUE
      ACCEPT"HOW MANY COLORS WOULD YOU LIKE? ",NCOL
      DO 40 J=1,NCOL
      CALL RAY
      DO 30 I=1,NRAY
      R(I,J)=RA(I)
30    CONTINUE
40    CONTINUE
      CALL SORT
      WRITE(OF,999)
      DO 50 I=1,10
      J=10*I
      WRITE(OF,998) J,RAD(I)
50    CONTINUE
998   FORMAT(7X,I3,7X,F10.4)
999   FORMAT(1X,"PERCENT ENERGY",5X,"RADIUS")
      RETURN
      END
      SUBROUTINE SORT
      PARAMETER NRAY=96,IF=11,OF=10,NF=2
      COMMON/NAME/ANAME,IREC,LG,IC,NCOL
      COMMON/OD/OBJ,DIM,EPD,LS,ANY,ANX,LNY,LNX,TH0
      COMMON/SD/SUR(10),CV(10),CC(10),TH(10),N(11,3)
      COMMON/TD/DEC(10,2),TILT(10,3),RDEC(10,2),RTILT(10,3),CVY(10),CVX(10)
      COMMON/TD/CVF(10)
      COMMON/RAYS/FOBY(NRAY),FOBX(NRAY),XTA(NRAY),YTA(NRAY),RA(NRAY),RAD(10)
      COMMON/RAYS/R(NRAY,3)
      NRA=NRAY*NCOL
      DIMENSION A(288)
      REAL N,LNY,LNX
      DO 60 I=1,NRA
      A(I)=0
      DO 20 K=1,NCOL
      DO 10 J=1,NRAY
      A(I)=AMAX1(A(I),R(J,K))
10    CONTINUE
20    CONTINUE
      DO 40 K=1,NCOL
      DO 30 J=1,NRAY
      IF(A(I).NE.R(J,K)) GOTO 30
      R(J,K)=0
      GOTO 60
30    CONTINUE
40    CONTINUE
60    CONTINUE
      M=IFIX(NRA/10+.5)
      DO 70 I=1,9
      RAD(I)=A(NRA+1-M*I)
70    CONTINUE
      RAD(10)=A(1)
      RETURN
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

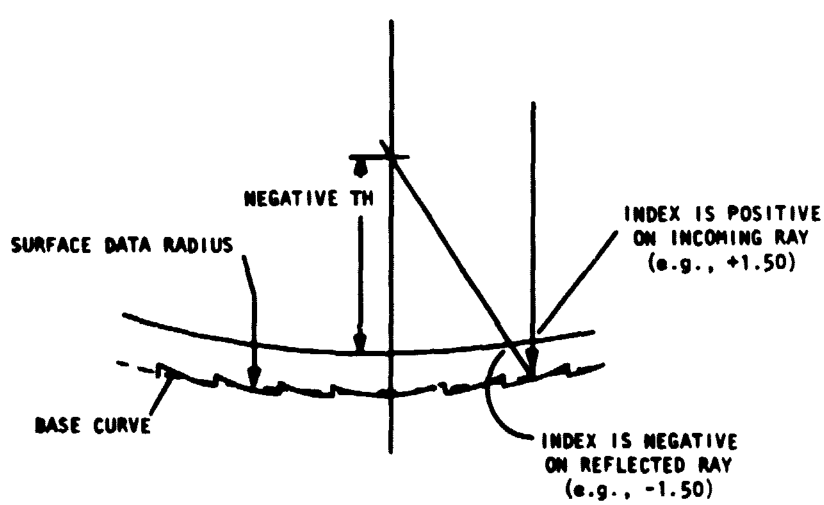
APPENDIX B

EXAMPLES OF LENSES THROUGH WHICH RAYS CAN BE TRACED

1. Front Surface Reflective Fresnel

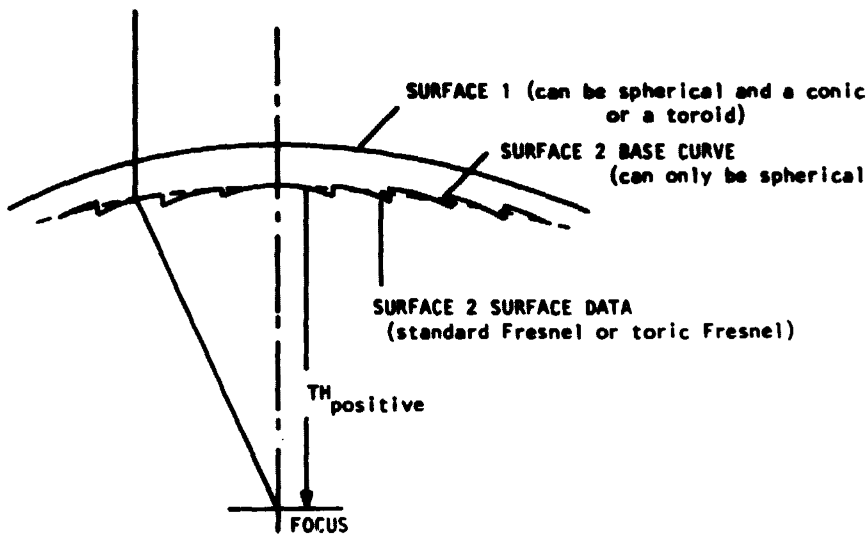


2. Rear Surface Reflective Fresnel



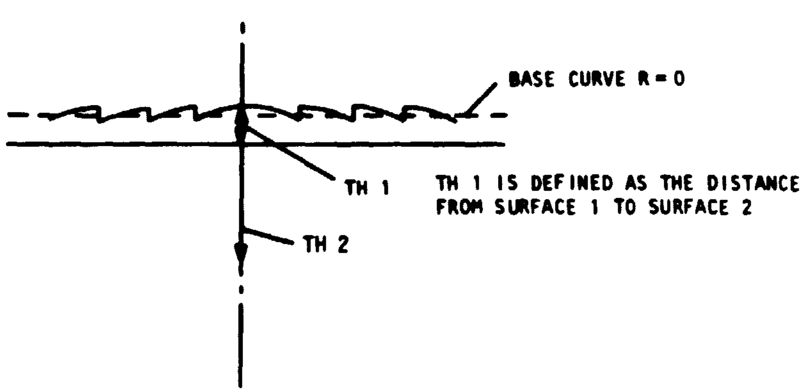
ORIGINAL PAGE IS
OF POOR QUALITY

3. Rear Surface Transmissive Fresnel



Note that, when "Fresnel" is specified, the limitation of only a spherical base curve enters. This limitation can be removed, but isn't in this edition. The Fresnel elements, as specified by the "surface data," can be conic, toric, etc.

4. Flat Transmissive Fresnel



Note that a flat surface can be designated as a conic even though it is flat, and it will be treated as an asphere with the departure from the plane being the same as the departure of the corresponding conic from the sphere whose curvature it approximates at its vertex.