

AD-A034 239

SCHOOL OF AEROSPACE MEDICINE BROOKS AFB TEX
RETINAL THERMAL MODEL OF LASER-INDUCED EYE DAMAGE: COMPUTER OPE--ETC(U)
NOV 76 A R MERTZ, B R ANDERSON, E L BELL

F/6 9/2

UNCLASSIFIED

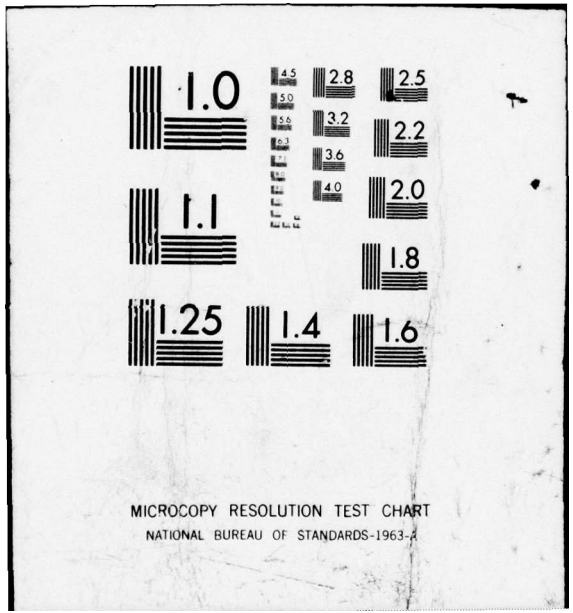
SAM-TR-76-33

NL

1 OF 2

AD
A034239

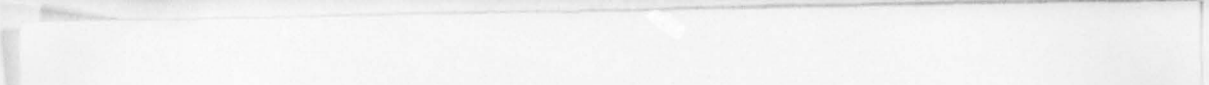
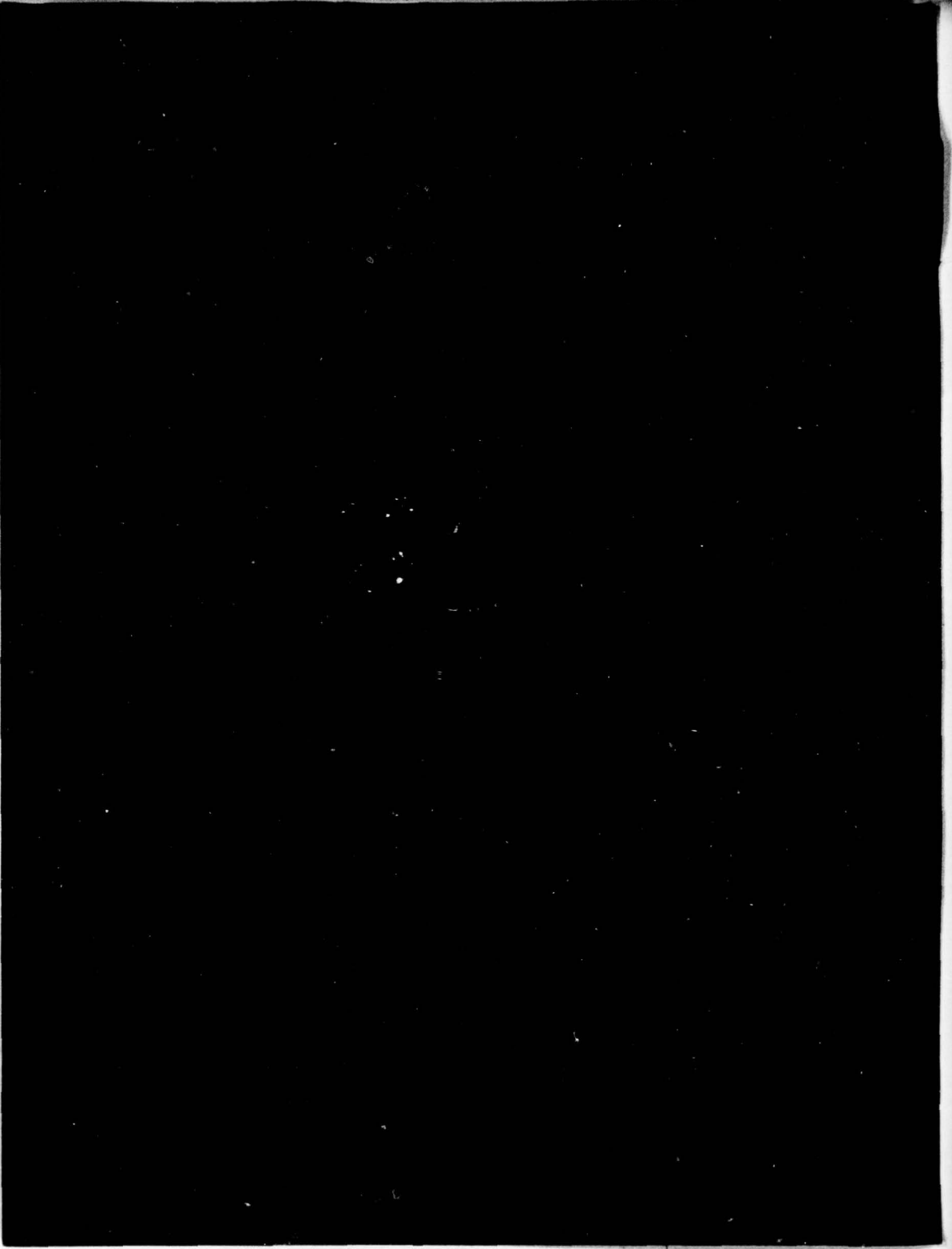




MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

ADA 034239

SAMUEL



Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER SAM-TR-76-33	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) RETINAL THERMAL MODEL OF LASER-INDUCED EYE DAMAGE: COMPUTER PROGRAM OPERATOR'S MANUAL		5. TYPE OF REPORT & PERIOD COVERED Final Report - June 1975 - June 1976
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) 10 Alan R. Mertz, Capt, USAF Bruce R. Anderson, 1st Lt, USAF Earl L. Bell, M.S. David E. Egbert, Capt, USAF		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS USAF School of Aerospace Medicine (RAL) Aerospace Medical Division (AFSC) Brooks Air Force Base, Texas 78235		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 62202F 16 6301-00-51 17 00
11. CONTROLLING OFFICE NAME AND ADDRESS USAF School of Aerospace Medicine (RAL) Aerospace Medical Division (AFSC) Brooks Air Force Base, Texas 78235		12. REPORT DATE 11 November 1976
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		13. NUMBER OF PAGES 136 (12) 139 p.
		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Ocular damage Retinal thermal model Laser effects Temperature rise predictions		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) A user-oriented description is given of a computer program for predicting thermal eye damage caused by exposure to laser radiation. This report describes the parameters necessary to run the program and provides a suggested range of values for the parameters. The input and output are described in detail, as are many of the capabilities and limitations of the program. The parameters and required input steps for a plot routine used to plot the temperature rises within the ocular structures are also presented. The architecture of the program and the interpretation of the input and output can be found in the IIT Research Institute technical report, "Thermal Model of Laser-Induced Eye Damage" (AD-A017-201).		

DD FORM 1 JAN 73 1473

EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

317 000

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

6/2

TABLE OF CONTENTS

	Page
INTRODUCTION	3
CAPABILITIES AND LIMITATIONS	5
INPUT REQUIREMENTS	7
OUTPUT FORMAT.	13
APPENDIXES	
A--GLOSSARY OF PARAMETERS.	17
B--PLOT ROUTINE.	41
C--INPUT-OUTPUT PROCESSES.	51
D--PROGRAM LISTING	61

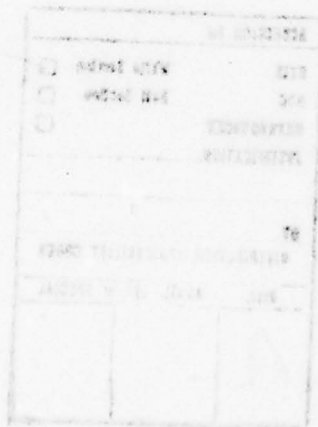
List of Illustrations

Figure		Page
B-1	Temperature Rise versus Axial Depth	47
B-2	Temperature Rise versus Radial Extent	48
B-3	Three-Dimensional View of Temperature Rise.	49
C-1	Sample Card Deck for Running Retinal Program RE2.	57
C-2	Sample Card Deck for Compiling and Running Retinal Program RE2	58
C-3	Sample Card Deck for Running the Plot Routine	59

ACCESSION FOR		
RTIS	White Section	<input checked="" type="checkbox"/>
636	Both Section	<input type="checkbox"/>
UNANNOUNCED		
JUSTIFICATION.....		
BY.....		
DISTRIBUTION/AVAILABILITY CODES		
Dist.	AVAIL. and/or	SPECIAL
A		

List of Tables

Table		Page
1	Ocular Structures Modeled.	8
2	Program Output Sections.	11
3	Input Deck for RE1	12
4	Input Deck for RE2	13
5	Printed Output for RE1	14
A-1	Optical Parameters for Caucasians and Negroes.	36
A-2	Optical Parameters for Rhesus Monkeys.	37
A-3	Thickness of Ocular Media.	38
A-4	Parameters for Computing Time Intervals.	38
A-5	Normalized Temperature Rise Decays	39
A-6	Zero-Order Bessel Function	40
A-7	Refractive Indexes	40
B-1	Sample Plot Input (with Comments).	46



RETINAL THERMAL MODEL OF LASER-INDUCED EYE DAMAGE:
COMPUTER PROGRAM OPERATOR'S MANUAL

INTRODUCTION

The Retinal Thermal Model is a mathematical model that predicts the thermal eye damage resulting from an exposure to laser radiation. This program, developed by the Illinois Institute of Technology Research Institute, is a result of many years of improvements in thermal damage modeling techniques. The mathematical basis for temperature predictions computed in the model is the standard heat-conduction equation in cylindrical coordinates

$$\left[\rho C \right] \frac{dv}{dt} = q(z,r,t) + K \left[\frac{1}{r} \frac{\partial v}{\partial r} + \frac{\partial^2 v}{\partial r^2} \right] + \frac{\partial}{\partial z} \left[K \frac{\partial v}{\partial z} \right]$$

where C = specific heat

ρ = density

q = rate of heat deposition from the laser

K = thermal conductivity

r = radial distance

z = axial distance

t = time

v = temperature rise above the initial temperature

The heat-conduction equation is approximated by finite differences and then solved with an explicit-implicit alternating-direction technique developed by D. W. Peaceman and H. H. Rachford (1). This technique solves the finite-difference equations explicitly in z and implicitly in r for odd time steps, and implicitly in z and explicitly in r for even time steps. In explicit calculations, existing temperatures are used to represent thermal gradients; in implicit calculations, future temperatures are used. This approach results in a set of equations that are solved using ordinary matrix algebra. Larger time intervals can be used with this technique than with standard explicit finite-difference methods. The model uses the predicted temperature rises to determine irreversible tissue damage by applying Henrique's damage criteria

$$\Delta \Omega(z,r,t) = C_1 \exp[C_2/T_a(z,r,t)] \Delta t$$

1. Peaceman, D. W., and H. H. Rachford, Jr. The numerical solution of parabolic and elliptic differential equations. J Soc Indust Appl Math 3:28-41 (1955).

where $\Delta\Omega(z,r,t)$ = incremental damage at point z, r

C_1 and C_2 = rate constants

$T_a(z,r,t)$ = absolute temperature

Δt = increment of time.

Irreversible tissue damage is defined as occurring whenever the integral of $\Delta\Omega$ over all time is greater than or equal to 1. From this mathematical basis, the model has the capability of predicting temperature rises, damage thresholds, and the extent of damage for specified sets of spatial coordinates within the ocular media. The model also has the capability to predict the retinal intensity distribution from the intensity distribution at the cornea. This optical spread capability has its basis in scalar diffraction theory, using the Fresnel approximation and adding terms to account for defocusing and ocular aberrations.

The Retinal Thermal Model has been divided into two programs, RE1 and RE2. Both programs perform the same tasks with one exception--RE1 contains the subroutine MXGRAN, which models the melanin granules, while RE2 does not contain MXGRAN.

Designed for maximum flexibility, the model offers wide variability in both input and output. It accommodates variations in laser radiation characteristics and in optical, thermal, and physiological properties of the eye. The model's design enables the user to specify his region of interest within the retinal layers and to print out only those portions of the output information which he desires.

The purpose of this manual is to give the user a basic understanding of the model's capabilities and how to use it within the limits of those capabilities. A meaningful description of a model of this type and flexibility, however, cannot be written without some complexity; and an individual will usually need some study and practical experience before feeling comfortable with the model. Additional information on the code can be obtained from the IITRI Technical Report, "Thermal Model of Laser-Induced Eye Damage" (2).

This manual briefly describes (1) the capabilities and limitations of the model as they pertain to the source, the eye, the mechanics of the program, and the output desired; (2) the basic input required, listing the required cards, their order, and appropriate formats; and (3) the printed output, including its format and the options available to the user. Appendix A is a glossary of all parameters that are either input or output, plus some parameters used internally in the program.

2. Takata, A. N., et al. Thermal model of laser-induced eye damage. Final Technical Report, IIT Research Institute Contract F41609-74-C-0005, 8 Oct 1974, USAF School of Aerospace Medicine, Brooks AFB, Tex. AD A017201.

Appendix B briefly describes the PLOT routine that can be used with the retinal model to obtain two- and three-dimensional plots of the predicted temperature rises. Appendix C covers the steps necessary to run the program on the IBM 360/65 computer at the San Antonio Data Service Center (SADSC), a computer facility available through a remote job-entry terminal located in the Biometrics Division, USAF School of Aerospace Medicine (USAFSAM), Brooks AFB, Tex. A description is included of the job-control language cards required to enter the program on the computer. Appendix D is a listing of the RE1 and RE2 programs and PLOT.

This manual is designed as a user's reference for the IITRI retinal model as it existed in November 1975. This version differs mainly in output format from the version described in the IITRI Technical Report.

CAPABILITIES AND LIMITATIONS

The user is responsible for adequately describing the exposure conditions to be modeled and the predictions (retinal intensity distribution, temperature rises, damage thresholds or extent of damage) he desires from the model. He must describe, or model, the incident radiation, the ocular media, the mechanics (temporal and spatial grid) of the program, and the output desired. This section presents a broad overview of the capabilities and limitations as they pertain to these four areas.

In developing the program, several major assumptions are made. First, the eye geometries are simulated in cylindrical coordinates, approximating the retina as a flat surface. Second, the relative retinal-intensity radial distribution is used at all depths of the eye below the retina, assuming that the incident radiation is coherent and dispersion of the beam through the retina will be minimal. Third, all reflected radiation is considered to move along axial directions; also, only first-order reflections are considered to be important to the total temperature rise. Fourth, the rates of retinal-tissue damage used in the damage integral are assumed to equate to the rates of skin-tissue damage; extensive work has been done with skin tissue in this area while very little has been done with the retina. Other assumptions will be discussed in later sections.

The model has a number of features which give the user flexibility in describing the incident radiation in terms of its spatial, spectral, and temporal properties. The model is designed only for monochromatic, coherent radiation. The spatial profile of the beam may be designated as uniform, gaussian, or irregular. Symmetry about the axis of propagation is always assumed. The user may specify the profile at either the cornea or the retina. For uniform or gaussian profiles, the user specifies the beam radius and the total power incident during a single pulse. For irregular profiles, the user constructs the desired beam profile by specifying the total power incident during a single pulse and the intensity (absolute or relative) and associated radial distances from the center of the beam.

The temporal properties of the incident radiation are specified by selecting the duration of a pulse, the repetition rate, and the number of pulses. Therefore, both single- and multiple-pulse exposures can be modeled. The model assumes all pulses to be square with respect to time; multiple-pulse trains will be composed of simple periodic, 100% modulated pulses. Because the model only predicts damage based on thermal mechanisms, the recommended time range for a pulse duration is between 10^{-7} and 10^3 seconds. For pulse durations outside this range, nonthermal damage mechanisms may become significant. The model will accommodate shorter pulse durations; however, only thermal effects will be predicted. The model automatically converts pulses with durations less than 3×10^{-9} seconds to pulses with 3×10^{-9} -sec pulse widths. The conversion is accomplished assuming energy conservation. The associated power conversions are also made internally in the model. Even though the model makes these conversions, the output is always given relative to the original input with one exception: the value for POW, the total power incident on the corneal surface, is given relative to the 3×10^{-9} -sec pulse.

The ocular media can be described in terms of the optical, thermal, and physiological characteristics of the eye. The optical parameters include the coefficients for absorption, reflection, aberration, and refraction. Scattering of the radiation within the ocular media is ignored, and all reflected radiation is along axial directions. The user has the option to either specify an optical system that calculates the retinal intensity distribution or specify the retinal intensity distribution directly. The optical system of the eye is specified in terms of focal length, distances between optical surfaces, refractive indexes, aberration coefficients, and pupil size. The relative retinal-intensity radial distribution is assumed to remain constant at all depths below the retina.

The thermal parameters include heat capacity and conductivity of the individual ocular layers, initial ocular temperature, and blood flow within the choriocapillaris and tissues surrounding the eye. The user may also specify the decay of temperature rises within the melanin granules for program RE1.

The user specifies the physiological characteristics of the eye by selecting the radius and thicknesses of the various homogeneous ocular layers. All surfaces are assumed to be flat and perpendicular to the laser beam axis. The user may divide the pigment epithelium into two sublayers; also, the user has the option of placing 1- μ m-cube melanin granules, separated by 1 μ m, in either one of the sublayers of the pigment epithelium. The melanin granules are modeled as absorbing all energy incident upon them; the surrounding media absorb like the choroid. The choice of the sublayer to contain the melanin granules is used to differentiate between the human and the monkey eye. In the monkey eye, the melanin granules are located in the anterior portion of the pigment epithelium; in the human eye, in the posterior portion. The melanin granules contribute to the temperature rise calculations only for pulse widths less than 10^{-5} seconds. For times greater than 10^{-5} seconds, the heat diffusion from one granule to another has already taken place; so the temperature rises are

constant across the entire granulated layer, and the granules do not contribute significantly to the temperature calculations. For this reason, the Retinal Thermal Model has been divided into two programs, RE1 and RE2. RE1 contains the subroutine MXGRAN which models the melanin granules within the pigment epithelium. RE2 does not contain MXGRAN but does allow division of the pigment epithelium into two homogeneous layers. RE2 requires less computer central processing time and core memory than does RE1.

The model automatically selects the temporal and spatial grid points in the ocular media; however, the user specifies the size of the increments and extent of the temporal and spatial grid. The dimensions of the spatial and temporal increments represent the limits of resolution of the model predictions. The spatial grid has uniform increments in the region of highest temperature rise and constantly expanding increments away from the highest temperature regions. The temporal grid has constantly expanding time steps from the beginning of the pulse. The predicted retinal intensity distribution and temperature rises are all relative to the specified spatial and temporal grid points. The user selects the range and spacing of spatial coordinates used to print the temperature rises, threshold powers, and extent of damage; also, the user has the option to print and plot temperature rises at any selected time. A separate plotting routine uses input data cards punched by the retinal model and user control cards to plot the temperature rises. Entire sections of the complete printed output can be deleted. Available options are described in the Output Format section.

The retinal model has a variety of input/output capabilities. The user can batch a sequence of exposures by varying the pulse-repetition frequency and/or number of pulses while keeping all other parameters constant. For such sequence of exposures, the program is initiated only once, thus conserving operating time.

INPUT REQUIREMENTS

This section describes the input required to specify the source, the ocular media, the mechanics of the program, and the formatting of the output. All parameters required by the user as input, along with the appropriate formats and data card numbers, are given for retinal models RE1 and RE2. All parameters are defined in Appendix A. When solving a problem, the user must model both the incident radiation and the ocular media to fit the desired situation, and must also specify the parameters governing the mechanical operation of the program, such as the size of the grid required and the output desired.

To simulate the radiation incident on the ocular media, the user has the option of specifying, via IPROF, a uniform-, gaussian-, or irregular-beam irradiance profile. Axial symmetry for all beam profiles is assumed. For uniform-beam profiles, the beam radius (RIM) must be specified; it is used with LIM (the number of radial intervals from the center of the beam to a specified radius) to establish the minimum radial grid increment (DR). For gaussian-beam profiles, RIM must be specified at a particular relative

intensity point (CUT). For irregular-beam profiles, the absolute or relative irradiance profile must be specified on a point-by-point basis by listing the irradiance value, $PX(L)$, at the radial distance, $RX(L)$, from the center of the beam. $PX(L)$ cannot have a value of zero at the center of the beam. The total number of specified irradiance points is equal to LR . The model will integrate the beam profile at radial intervals ($RINT$). We suggest that the irradiance points be specified at radial intervals equal to multiples of $RINT$ to avoid interpolation.

The total power per pulse in the beam at the cornea (POW) must be specified for all beam profiles. In effect, the user has the capability of specifying the divergence, which is a function of the distance of the pupil from the nearest beam waist (ZO). ZO is input only if the spread function is used ($IFIL=1$). In addition, the pulse width ($DPULSE$), the pulse repetition frequency ($REPET$), the number of pulses entering the ocular media ($NPULSE$), and the wavelength ($WAVEL$) must also be specified for all simulated exposures. However, $WAVEL$ is used only in the image spread-function calculation ($IFIL=1$). For single-pulse exposures, a value for $REPET$ must be supplied; however, it will not be used in calculations in the model.

The eye is modeled as a cylinder with its axis coincident with the axis of radiation propagation. The various layers of the eye lie perpendicular to the cylinder axis, with flat boundaries between the layers. The radial extent of the eye is specified as RVL . The various ocular media modeled in the eye are listed in Table 1. The thickness, transmittance, reflectance, and absorption coefficients of the various layers are all input. The user must also supply the thermal conductivity, $CONX(L)$, and heat capacity, $VSHX(L)$, for each ocular layer. Values for these parameters are given in Appendix A.

TABLE 1. OCULAR STRUCTURES MODELED

<u>Label</u>	<u>Ocular structures</u>
1	Everything from the anterior portion of the eye to the pigment epithelium
2	Pigment epithelium
3	Choriocapillaris (vascular layer)
4	Choroid
5	Sclera
6	Tissue posterior to the sclera

The pigment epithelium layer may be divided into two sublayers, with the user specifying the thickness and the absorption coefficient for each sublayer. IGX is the parameter used to specify the absorption coefficients. For a simulation of the human eye (IGX=1), the absorption coefficient for the anterior sublayer (APE1) is set equal to the absorption coefficient for the choroid (ACH); the absorption coefficient for the posterior sublayer (APE2) is calculated within the model. For the simulation of the monkey eye (IGX=0), the absorption coefficient for APE1 is calculated within the model, while APE2 is set equal to ACH. The distinction of two sublayers with differing absorption coefficients is due to the assumption that most of the absorption within the pigment epithelium occurs in the posterior section for human eyes and in the anterior section for monkey eyes. The relative thickness of the two sublayers is specified by RPE.

The user can transform the distribution of the beam incident on the cornea to a retinal distribution by using the spread function (IFIL=1) or can ignore that transformation (using IFIL=0) and specify the intensity distribution at the retina. The spread function simply transfers the beam from the cornea to the retina, simulating ocular focusing and optical aberration effects. The initial temperature of the eye is specified as T0. The radius of the pupil of the eye (PUPIL) is specified by the user and implemented within the model to define the initial beam intensity profile. Only when the spread function is used must Z0, FLO, FC, NB, PP, CABER, PC, J0, and NA be input.

The effects of blood flow are assessed in two ocular structures--the choriocapillaris and the tissue surrounding the eye. Within both structures, the blood is treated as a heat sink--the extraction of heat from the adjacent tissue by the blood as it enters that tissue. The user inputs the specific heat of blood (SHB), the total blood flow to the choriocapillaris (CFLOW), and the rate of blood flow to the tissue surrounding the eye (XFLOW). The model computes the temperature rise resulting from the heating of the blood as it enters both ocular structures. The user can also account for the radial transport of heat by the radial flow of blood within the choriocapillaris. To do this, statement number 31 of the RE1 and RE2 programs (Appendix D) must be deleted and replaced by statements to establish specific values for XFLOWO(L1) L1=1,6. XFLOWO(L1) is defined as the total blood flow per unit area leaving the choriocapillaris at a given radial distance R. It is given in units of $g \cdot cm^{-2} \cdot sec^{-1}$. Without this change, the model will only treat blood as a heat sink.

The model also computes the temperature effects of heat absorption in the melanin granules. The user must specify TS(L), which determines how the average temperature rise of the granules decays with time, and LTMAX, which controls the time beyond which the temperature rises of the melanin granules are completely dissipated. The temperature-rise contributions are specified at time increments of 3×10^{-9} seconds. In the output section of the model, XPD(K) represents the degree to which the melanin granules affect the temperature in the pigment epithelium.

To predict the power required to produce irreversible damage by the damage-integral method, the coefficients for the rates of damage (DAMAGE) must be specified. The model will also compute the power required to raise the temperature of the tissue to TSTEAM and will repeat the calculation at temperature intervals (DTSTM) until the power to produce irreversible damage, as determined by the damage-integral method, is reached.

The mechanics of model operation include determining the spatial grid system and the time intervals used in the temperature-rise computation. The spatial grid system, used to specify the locations at which an evaluation of the temperature-rise and damage-threshold predictions is desired, has both a uniform and a constantly expanding portion. The uniform portion of the grid is positioned at the center of the beam in the pigment epithelium, usually the region of highest temperature rise. The grid then constantly expands away from this region. The user specifies the size of the uniform radial grid interval (DR) by using LIM and LESION for irregular or gaussian profiles, and LIM and RIM for uniform profiles. The uniform axial grid interval is about one-sixth of the thickness of the pigment epithelium. Upon this grid the physiological layers are constructed. The various ocular layers and the labels used to assign absorption, conductivity, and heat-capacity values to these ocular layers are listed in Table 1. For damage threshold calculations, the range of grid locations at which calculations are made is determined from LIMAX and MAXPRT for axial locations and from RMAX for radial locations. LIMAX and RMAX must be chosen so that $(ID2-ID1+1)*JM < 27$ where $ID1=IMAX-LIMAX$, $ID2=IMAX+LIMAX$, and JM is the index of the first radial grid point beyond RMAX. IMAX is the axial grid point at which peak temperature rises occur at the conclusion of the pulse.

The time intervals used in calculating temperature rises, the maximum time during which temperature rise calculations and damage threshold predictions are made, and the time intervals used to subdivide the pulse widths are complex and intricately related. Unless the reader is experienced with the program, we suggest that the values supplied in Appendix A for FTIME, EDT1, EDT2, NPT, XCT, and KTT be used. Appendix A contains some of the relationships between these parameters for those who need to change the suggested values.

To reduce computation time of the program, the user may group all exposures which differ only in REPET and NPULSE. This is done by specifying the desired values for REPET and NPULSE and specifying the total number of exposures so grouped in NTEST, with a maximum of 7 pairs of values per group.

The formatting of the output includes selecting the output sections to be printed and the times and spatial ranges within the grid system for temperature rise printouts and/or plotting. Using IPRT codes, as shown in Table 2, the user specifies the output sections desired. For multiple-pulse exposures, temperature rises are printed only for the first pulse

TABLE 2. PROGRAM OUTPUT SECTIONS

<u>Code</u>	<u>Section</u>
IPRT(1)	Grid information
IPRT(2)	Laser profile
IPRT(3)	Data identification
IPRT(4)	Blood flow and heat deposition
IPRT(5)	Temperature rises
IPRT(6)	Normalized temperature rises
IPRT(7)	Normalized temperature rises of melanin granules
IPRT(8)	Predicted threshold laser power
IPRT(9)	Axial extent of damage
IPRT(10)	Radial extent of damage

incident on the retina; however, prediction of damage is based on multiple-pulse effects. The range of grid locations at which temperature rise calculations are printed is determined from ID1 and ID2 for the axial range, and from JD1 and JD2 for the radial range. The user has the option of printing the temperature rises at all the time intervals determined within the model (ITYPE=1), only every nth time interval (ITYPE=n), and/or at any selected times (KTYPE0=1) within the maximum time used by the model. The total number of selected times for printing is equal to KTYPE, while the selected times for printing are specified in TIMEX. Temperature rise calculations are always printed at the beginning and end of the pulse and at the time interval TIME. When plots of temperature rises at selected times are desired (KTYPE0=0), the model will always provide printouts in addition to the plots. The range of axial and radial grid locations desired for plots is determined by II1, II2, and JJ1, JJ2, respectively. II3 enables the user to mark (by an asterisk) a specific axial depth on plots for easy identification and comparison with other plots.

Tables 3 and 4 provide a quick reference to the parameters required as input to the model as well as their required formats and order. Data cards with an asterisk as a prefix to the data card number should be checked to ensure they simulate the desired exposure; data cards without the prefix asterisk generally remain constant from exposure to exposure. Data cards with an asterisk as a suffix to the data card number are not input unless an irregular profile (IPROF=2) or the spread function (IFIL=1) is used. When the uniform and gaussian profiles are used (IPROF=0,1), data cards 20, 21, and 22 are not input. When the spread function is not used (IFIL=0), data cards 23, 24, and 25 are not input.

TABLE 3. INPUT DECK FOR RE1

<u>Data card number</u>	<u>Format</u>	<u>Parameter</u>
1 (4 cards)	11F7.2	FTIME(L) L=1,38
*2 (1 card)	10I7	IPRT(I) I=1,10
*3 (1 card)	F7.4,3I7	RIM, LIM, IFIL, IGX
*4 (1 card)	F7.2,2I7,F7.2	RMAX, LIMAX, MAXPRT, LESION
*5 (1 card)	I7,3E7.2	IPROF, POW, CUT
*6 (1 card)	10E7.2	DPULSE
*7 (1 card)	10I7	NTEST, NRUN(L) L=1, NTEST
*8 (1 card)	10E7.2	REPET(L) L=1, NTEST
*9 (1 card)	10I7	NPULSE(L) L=1, NTEST
*10 (1 card)	10I7	ID1, ID2, JD1, JD2, ITYPE
11 (1 card)	11F7.2	TO, EDT1, EDT2
*12 (1 card)	11F7.2	TOM, APE, AVL, ACH, ASC, ATS, RCO, RRT, RSC, RPE, WAVEL
*13 (1 card)	11F7.2	TAV, TPE, TVL, TCH, TSC, RVL
14 (1 card)	11F7.2	CONX(L) L=1,6
15 (1 card)	11F7.2	VSHX(L) L=1,6
16 (4 cards)	10I7	NPT(L) L=1,38
17 (4 cards)	10F7.3	XCT(L) L=1,38
18 (4 cards)	10I7	KTT(L) L=1,38
19 (1 card)	10E8.3	PUPIL
20(1 card)	I7	LR
21(1-3 cards)	10E7.3	RX(L) L=1,LR
22(1-3 cards)	10E7.3	PX(L) L=1,LR
23(1 card)	10E8.3	ZO, FLO, FC, NB, CABER, PP, PC
24*(4 cards)	10F8.5	JO(L) L=1,32
25*(3 cards)	10F8.5	NA(L) L=1,22
26 (1 card)	10F7.3	SHB, XFLOW, CFLOW
*27 (1 card)	I7,3E7.2	KTYPEO
*28 (1 card)	I7,3E7.2	KTYPE
*29 (1 card)	10E7.2	TIMEX(K) K=1,KTYPE
*30 (1 card)	10I7	II1, II2, II3, JJ1, JJ2
31 (1 card)	I7,3E7.2	LTMAX
32 (22 cards)	10F7.3	TS(L) L=1,LTMAX, 10
33 (1 card)	11F7.2	DAMAGE(L2,1) } DAMAGE(L2,2) } L2=1,2 TSTEAM, DTSTM

Prefix * indicates parameters most often altered for specific exposures.

Suffix * indicates parameters not input unless irregular profile or spread function is used.

TABLE 4. INPUT DECK FOR RE2

1-30	Same as RE1	Same as RE1	} L2=1,2
31(1 card)	11F7.2	DAMAGE(L2,1)	
		DAMAGE(L2,2)	

OUTPUT FORMAT

The printed output of the retinal models, RE1 and RE2, are arranged into 11 sections. The printout format and the IPRT codes for each section are listed in Table 5. The user has the option of printing only the sections desired, as specified in Table 2, except for one. The unlabeled section listed prior to Predicted Threshold Laser Power on Table 5 is always printed. (The definitions for the program parameters are given in Appendix A.) Not every parameter in Table 5 will be printed every time. In the Laser Profile Section, if the spread function is not used, all parameters dealing with it will not be printed. On the first line of this section, only RIM is printed for a uniform beam (IPROF=0); SIGMA, RIM, and CUT for a gaussian beam (IPROF=1); or RINT for an irregular beam (IPROF=2). For a single-pulse exposure (N=1), trainlength and repetition rate are deleted from the unlabeled section.

In the Temperature Rises section (Table 5), numbers printed at each axial and radial grid point represent the temperature rise ($^{\circ}\text{C}$) above the initial temperature of the eye (T_0) at the time indicated. For the Normalized Temperature Rises section, the temperature rise ($^{\circ}\text{C}$) is divided by the input power. Therefore, the numbers printed at each grid point represent the temperature rise per watt of input power. For pulses of less than a 3×10^{-9} -sec duration, POW is converted to a power relative to a 3×10^{-9} -sec pulse; the converted POW is used to normalize the temperature rises.

For the axial and radial extent of damage, the model selects the appropriate statement from those listed in Table 5. If the input power produces no damage within the grid range specified by LIMAX, the model will print NO DAMAGE--LASER POWER TOO LOW. If the grid range specified by LIMAX contains the most anterior point at which damage occurs, the model will print MINIMUM DEPTH OF DAMAGE = (the value given will be relative to the anterior boundary of the pigment epithelium). If the maximum posterior point at which damage occurs is contained within the grid range specified by LIMAX, the model will print MAXIMUM DEPTH OF DAMAGE = (again, the value given will be relative to the anterior boundary of the pigment epithelium). If damage is present, the radial extent of damage will be printed for each axial grid point specified by LIMAX. The value for the radial extent of damage will be relative to the center of the laser beam.

TABLE 5. PRINTED OUTPUT FOR RE1

GRID INFORMATION IPRT(1)

R2 =
 R1 = ZM =
 ID1 = ID2 = JD1 = JD2 =
 DR = DZ =
 IPA = IPC = IPE = IPS = IPT = IPV =
 LPA = LPC = LPE = LPS = LPV =
 M = M1 = N = N1 =
 R =
 Z =
 ZH =

LASER PROFILE IPRT(2)

SIGMA = RIM = CUT = RINT =
 ZO = FLO =
 CABER = CABER2 = PP =
 PC = NB = NC =
 FC = WAVEL =
 QP =
 HR =

DATA IDENTIFICATION IPRT(3)

REPET =
 NPULSE =
 AAV = ACH = APE = ASC = ATS =
 RCO = RRT = RPE = TOM = AVL =
 TAV = TCH = TPE = TSC = TVL =
 IGX = IFIL = IPROF = LIM = NTEST =
 POW = DPULSE = RIM = RMAX = TIME =
 CFLOW = XFLOW = SHB = EDT1 = EDT2 =
 DT = KM = KT = PTIME = XC =
 IKX = AP = APE1 = APE2 = IG =
 RVL = PUPIL = TO = LIMAX = MAXPRT =

BLOOD FLOW AND HEAT DEPOSITION IPRT(4)

FLOWI =
 FLOWX =
 S =
 S =
 .
 .
 .
 S =

TABLE 5. (Continued)

TEMPERATURE RISES IPRT(5)

TIME = K =
R =
Z =
Z =
.
.
.
Z =

NORMALIZED TEMPERATURE RISES IPRT(6)

TIME = K = POWER = 0.1000E+01 WATTS
R =
Z =
Z =
.
.
.
Z =

NORMALIZED TEMPERATURE RISES OF MELANIN GRANULES IPRT(7)

LTMAX = BT =
XPD =

(unlabeled section)

WAVELENGTH = DAMAGE =
NRUN = TRAINLENGTH = SEC PULSE WIDTH = SEC
NUMBER OF PULSES = REPETITION RATE = PULSES/SEC
IMAGE RADIUS = LESION RADIUS = CM

PREDICTED THRESHOLD LASER POWER IPRT(8)

R =
Z = QD =
Z = QD =
.
.
.
Z = QD =

TABLE 5. (Continued)

TEMPERATURE RISES AT SELECTED TIMES TIMEX(K)

TIME =
 R =
 Z =
 Z =
 .
 .
 .
 Z =

AXIAL EXTENT OF DAMAGE IPRT(9)

NO DAMAGE--LASER POWER TOO LOW
 or
 DEPTHS OF DAMAGE BEYOND BOTH SPECIFIED DEPTHS
 or
 MINIMUM DEPTH OF DAMAGE = CM
 and/or
 MAXIMUM DEPTH OF DAMAGE = CM

RADIAL EXTENT OF DAMAGE IPRT(10)

Z = CM RADIAL EXTENT OF DAMAGE GREATER THAN CM
 or
 Z = CM RADIAL EXTENT OF IRREVERSIBLE DAMAGE = CM

APPENDIX A

GLOSSARY

All parameters used as either input or output in the retinal models, and some used internally, are listed in alphabetical order, with appropriate units and suggested input values. The equations provided are in FORTRAN IV language, where ALOG represents the natural logarithm and ** represents "raised to the power." For some of the parameters, numerical values are tabulated in the tables at the end of the glossary.

AAV--The absorption coefficient for the ocular media from the cornea to the retina.

Units: inverse cm

$$AAV = \text{ALOG}(TOM)/TAV$$

ACH--The absorption coefficient for the choroid.

Units: inverse cm

Suggested input value: Tables A-1 and A-2

AP--The fraction of heat that, deposited in the granulated pigment epithelium, is absorbed by the melanin granules. AP is calculated from ACH, RPE, TPE, APE1, and APE2. It is printed and used only when the subroutine MXGRAN in RE1 is used.

Units: unitless

APE--The absorption coefficient of the pigment epithelium.

Units: inverse cm

Suggested input value: Tables A-1 and A-2

APE1--The absorption coefficient for the anterior sublayer of the pigment epithelium.

Units: inverse cm

$$APE1 = (APE - ACH * (1. - RPE)) / RPE \text{ for IGX}=0$$

$$APE1 = ACH \text{ for IGX}=1$$

APE2--The absorption coefficient of the posterior sublayer of the pigment epithelium.

Units: inverse cm

$$APE2 = ACH \text{ for IGX}=0$$

$$APE2 = (APE - ACH * RPE) / (1. - RPE) \text{ for IGX}=1$$

ASC--The absorption coefficient for the sclera.

Units: inverse cm
Suggested input value: same as ACH

ATS--The absorption coefficient for the tissue posterior to the sclera of the eye.

Units: inverse cm
Suggested input value: same as ACH

AVL--The absorption coefficient for the choriocapillaris (vascular layer).

Units: inverse cm
Suggested input value: same as ACH

BT--The time interval during which heat conduction from the granules is insignificant. It is the time interval used to evaluate the contributions of the melanin granules to the temperature rises. BT is set equal to 0.3×10^{-8} .

Units: sec

CABER--A constant in the spherical aberration term used in the spread function. The spherical aberration term is $CABER \rho^4 / \lambda$, where ρ is the radius in the pupil plane and λ is the wavelength. CABER is printed only when the spread function is used (IFIL=1).

Units: $\text{cm}^{-4} \cdot \text{nm}$
Suggested input value: $-3.0\text{E}+6$

CABER2--A spherical aberration constant calculated by dividing CABER by the wavelength (nm) of the incident radiation. CABER2 is printed only when the spread function is used (IFIL=1).

Units: cm^{-4}

CFLOW--The total blood flow to the choriocapillaris.

Units: $\text{g} \cdot \text{sec}^{-1}$
Suggested input value: 0.024

CONX(L), L=1,6--The thermal conductivity of the Lth ocular media as listed in Table 1 (text).

Units: $\text{cal} \cdot \text{cm}^{-1} \cdot \text{sec}^{-1} \cdot \text{C}^{-1}$
Suggested input value: 0.0012

CUT--The fraction of the peak intensity in the beam cross-sectional distribution at which the beam radius, RIM, is specified for gaussian profiles (IPROF=1). CUT can be any fraction of the peak intensity, but RIM must be specified at the same point.

Units: unitless

Suggested input value: $1.35E-1$ ($1/e^2$ intensity points of a gaussian profile)

DAMAGE (L2,1), DAMAGE (L2,2), (L2=1,2)--The DAMAGE array contains the coefficients for the damage-rate integral.

For temperatures below 50°C:

$$\text{Rate} = \text{EXP} (\text{DAMAGE}(1,1) - \text{DAMAGE}(1,2)/(\text{VC}+273+\text{T0})).$$

For temperatures above 50°C:

$$\text{Rate} = \text{EXP} (\text{DAMAGE}(2,1) - \text{DAMAGE}(2,2)/(\text{VC}+273+\text{T0})).$$

VC is the temperature rise (°C) at the specified grid points. T0 is the initial temperature (°C) of the eye, and the number 273 converts degrees Celsius to Kelvin. The values provided are for skin tissue; but they are assumed to equate to the damage-rate constants for retinal tissue.

Units: DAMAGE(L2,1): unitless

DAMAGE(L2,2): sec^{-1}

Suggested input values: DAMAGE(1,1) = 149.
DAMAGE(1,2) = 50,000.
DAMAGE(2,1) = 242.
DAMAGE(2,2) = 80,000.

DPULSE--The exposure duration of an individual pulse.

Units: sec

Suggested input value: $3.0E-9$ to $1.0E+3$

DR--The radial increment in the uniform portion of the grid network.

Units: cm

DR = LESION/LIM for gaussian and irregular beam profiles (IPROF=1,2)

DR = RIM/(LIM-.5) for uniform profiles (IPROF=0)

NOTE: For IFIL=1, since RIM is a corneal dimension, LIM must be large to obtain a small DR.

DT--The initial time interval after the start of a pulse at which temperature rise calculations are made. Successive calculated time intervals are increased by the stretching factor XC.

Units: sec

DT = DPULSE * (XC-1.)/(XC**NP-1.)
NP = NPT(L1) for single pulse
XC = XCT(L1) for single pulse
L1 = ALOG(DPULSE)/.69315 + 29.
NP = 5 for multiple pulse
XC = 1.4 for multiple pulse

DTSTM--The temperature increment used to increase TSTEAM in calculating the power required to produce the temperature TSTEAM in the melanin granules. Successive calculations and printouts of Predicted Threshold Power will be made at each increment of TSTEAM until the power required to produce the temperature TSTEAM exceeds the power required to produce irreversible damage as predicted by the damage integral method. Reducing DTSTM results in increased computation time and printout.

Units: °C
Suggested input value: 200.

DZ--The axial increment in the uniform portion of the grid network.

Units: cm
DZ = TPE/M1 - 1.E-25
M1 = 6

EDT1, EDT2--Parameters used to determine the time intervals at which temperature rises are calculated. The model divides the computed time interval into 2*IKX subintervals to insure stability and accuracy. IKX is dependent upon TIME, EDT1, and EDT2. The suggested values are adequate except for pulse widths greater than 10³ sec.

Units: unitless
Suggested input values: EDT1 = 0.16; EDT2 = 1.

IKX = TIME**EDT1 + EDT2

FC--The focal length of the cornea measured in the ocular media. FC is required only when the spread function is used (IFIL=1).

Units: cm
Suggested input value: 3.12E0--humans
2.43E0--rhesus monkeys

FLO--The second principal focal length at a 500-nm wavelength. The second principal focal point is the point at which parallel light incident upon the anterior portion of the eye will focus. FLO is required only when the spread function is used (IFIL=1).

Units: cm
Suggested input value: 2.242E0--humans
1.684E0--rhesus monkeys

FLOWI(J), J=1,JVL--The flow of blood into a unit volume of the chorio-capillaris at some radial point, R(J). JVL is the index such that R(JVL)=RVL.

Units: $g \cdot cm^{-3} \cdot sec^{-1}$

FLOWX(J), J=1,JVL--The product of the radius at some radial point, R(J), and the net flow of blood per unit area in the radial direction at point R(J). JVL is the index such that R(JVL)=RVL.

Units: $g \cdot cm^{-1} \cdot sec^{-1}$

FTIME(L), L=1,38--The array FTIME is used for multiple-pulse exposures to determine the time interval (TIME) over which the damage integral is evaluated. TIME is a function of FTIME; each element of FTIME is associated with a range of pulse widths (DPULSE).

Units: unitless

Suggested input value: 1.8 for all elements

TIME = FTIME(L1) * X1 for multiple pulse

L1 = ALOG(DPULSE)/.69315 + 29.

X1 = NPULSE(L)/REPET(L) The largest value for any NTEST.

Therefore, to increase the time interval (TIME) over which the damage integral is evaluated, one should increase the Lth element of FTIME associated with the specified pulse width (DPULSE).

HR(J), J=1,N--The normalized retinal irradiance at radial position R(J). Symmetry about the axis is assumed.

Units: unitless

ID1, ID2--Input and output parameters. As input parameters, ID1 and ID2 are integers used to determine at what axial positions the temperature rises are to be printed. The temperature rises will be printed at axial positions with indexes from I=IPE+ID1 to I=IPE+ID2. As output parameters, ID1 and ID2 are actual grid index points relative to the first grid point located anterior to the cornea. Temperature rises are to be printed from grid point ID1 to point ID2.

Units: unitless

Suggested input values: dependent upon the user

ID1 (output) = IPE + ID1 (input)

ID2 (output) = IPE + ID2 (input)

IFIL--The parameter that allows the user to decide whether or not to use the spread function. The spread function is used to transfer the beam distribution from the cornea through the ocular media to the retina. When the spread function is used, all input (RIM, POW, CUT, RX(L), PX(L), LR, ZO) beam characteristics must apply to the beam at the cornea. When the spread function is not used, the input spatial beam characteristics are assumed to apply to the beam distribution at the retina, with the exception of POW and PX(L) which always apply to the cornea.

Units: unitless

Suggested input value: 1--spread function is used.
0--spread function is not used.

IG--The index of the initial grid point in the melanin granules. It is used and printed only in program RE1.

Units: unitless

IGX--The selection parameter for the absorption coefficients of the two sublayers modeled in the pigment epithelium. For IGX=1, the absorption coefficient for the anterior sublayer (APE1) is equal to ACH. The absorption coefficient for the posterior sublayer (APE2) is computed assuming it contains most of the melanin granules. For IGX=0, APE2=ACH and APE1 is computed assuming the anterior sublayer contains most of the melanin granules.

Units: unitless

Suggested input value: 1--a human eye
0--a monkey eye

II1, II2--The indexes used to specify the range of axial grid values desired for a plot. These indexes are the actual indexes of grid points, with II1 closer to the anterior part of the eye.

Units: unitless

Suggested input values: dependent upon the user

II3--An identification index used in the plotting routine. An asterisk can be placed on the curve at the axial depth associated with grid point II3 in a plot and allows easy reference for comparing similar curves in more than one plot. II3 is the index of an actual grid point.

Units: unitless

Suggested input value: dependent upon the user

IKX--The number of times the temperature calculations are repeated to insure stability.

Units: unitless

IKX = TIME **EDT1 + EDT2
TIME = FTIME(L1)*X1 for multiple pulse
L1 = ALOG (DPULSE)/.69315 + 29.
X1 = NPULSE/REPET the largest value for any NTEST
TIME = DT * (XC **KT -1.)/(XC-1.) for single pulses
KT = KTT(L1)
XC = XCT(L1)

IPA--The index of the initial grid point located in the cornea. Its value is always 2.

Units: unitless

IPC--The index of the initial grid point in the choroid.

Units: unitless

IPE--The index of the initial grid point in the pigment epithelium. Its current value is 10.

Units: unitless

IPROF--The parameter used to describe the laser intensity profile. If a uniform profile is specified, RIM and POW must be specified. For a gaussian profile, RIM, CUT, and POW must be given. Irregular profiles require PX(L), RX(L), LR, and POW.

Units: unitless
Suggested input value: 0--uniform profile
1--gaussian profile
2--irregular profile

IPRT(I), I=1,10--The parameter which gives the user the choice of printing or not printing each of 10 separate output sections described in text Output Format section.

Units: unitless
Suggested input value: 0--printing is not desired.
1--printing is desired.

IPS--The index of the initial grid point in the sclera.

Units: unitless

IPT--The index of the initial grid point in the tissue posterior to the sclera.

Units: unitless

IPV--The index of the initial grid point in the choriocapillaris.

Units: unitless

ITYPE--Used to determine the time indexes (K) at which the temperature rises will be printed. The total number of times the temperature rise calculations can be printed is equal to KT. If the temperature rises are to be printed at all times (XT(K) K=2,KT), ITYPE must equal one. If temperature rises are to be printed at every nth time, ITYPE must equal n. Temperature rise printouts will always be provided at the first time (K=2) after initiation of the pulse, at the conclusion of the pulse (K=KM), and at the final time over which damage is assessed (TIME=XT(KT)). ITYPE must never equal zero.

Units: unitless

Suggested input value: dependent upon the user

JD1, JD2--The radial indexes used to determine the radial positions from the center of the laser beam at which the temperature rises are to be printed. The model will print the temperature rises starting at radial position JD1, out to radial position JD2. JD1=1 corresponds to the z-axis or the center of the beam. All 14 radial grid points can be printed; but only 9 will be printed on a single line, with the other 5 printed in consecutive order on the second line.

Units: unitless

Suggested input values: dependent upon the user

JJ1, JJ2--The indexes of grid points used to specify the range of radial grid values desired for a plot. JJ1 is the index closer to the center of the beam.

Units: unitless

Suggested input values: JJ1 = 1, and JJ2 = 5

JO(L), L=1,32--The value of the zero-order Bessel function for argument values to 3.1 in 0.1 increments. It is used in constructing the spread function.

Units: unitless

Suggested input values: Table A-6

K--An index of the expanded times, XT(K)--times at which temperature rise calculations are made.

Units: unitless

KM--The index indicating the temperature rise printout occurring at the end of the pulse ($XT(KM)=DPULSE$).

Units: unitless

KT--The maximum number of times at which temperature rise calculations are computed.

Units: unitless

$XT(KT) = TIME$

$KT = KTT(L1)$ for single pulse

$L1 = ALOG(DPULSE)/.69315 + 29.$

$KT = [ALOG(1.+TIME*(XC-1.)/DT)/ALOG(XC)+1.]+1$ for multiple pulses

KTT(L) $L=1,38$ --An array of the number of steps used to reach the total time (TIME). The suggested values were calculated to reduce error and increase stability in solving the finite-difference equations in the model.

Units: unitless

Suggested input values: Table A-4

KTYPE--The total number of temperature rise plots or selected time printouts desired. If no plots or selected time printouts are desired, set $KTYPE=0$. $KTYPE$ has a maximum value of 10. A printout of the temperature rise is automatic with each requested plot.

Units: unitless

Suggested input value: dependent upon the user

KTYPE0--The parameter that controls the punching of data cards used as input to the plot routine.

Units: unitless

Suggested input value: 0--card punching

1--no card punching

LESION--The radius of the retinal lesion. It is used only to determine DR for efficient grid structuring. It is not used for uniform beam profiles ($I\text{PROF}=0$).

Units: cm

Suggested input value: dependent upon the user

$DR = \text{LESION}/LIM$ for gaussian and irregular beam profiles ($I\text{PROF}=1$ or 2).

LIM--The number of radial intervals from the center of the beam to RIM for uniform beam profiles (IPROF=0), or to LESION for gaussian and irregular profiles (IPROF=1 or 2). LIM is used to determine the size of the smallest uniform radial-grid increment (DR). There are only four uniform radial-grid intervals.

Units: unitless
Suggested input value: 5

LIMAX--A parameter that determines the range of axial distances at which damage calculations are printed. It is used in conjunction with MAXPRT. For single-pulse exposures, LIMAX has a maximum value of 9 for MAXPRT=2 or 3, and a maximum value of 4 for MAXPRT=1. For multiple-pulse exposures, LIMAX has a maximum value of 2 regardless of MAXPRT.

For MAXPRT = 1, axial distance = IMAX - 2 LIMAX to IMAX
MAXPRT = 2, axial distance = IMAX + LIMAX
MAXPRT = 3, axial distance = IMAX to IMAX + 2 LIMAX
IMAX = the axial grid point at which peak temperature rises occur at the conclusion of the pulse.

Units: unitless
Suggested input value: dependent upon the user

LPA--The index of the last grid point located in the vitreous humor.

Units: unitless

LPC--The index of the last grid point in the choroid.

Units: unitless

LPE--The index of the last grid point in the pigment epithelium.

Units: unitless

LPS--The index of the last grid point in the sclera.

Units: unitless

LPV--The index of the last grid point in the choriocapillaris.

Units: unitless

LR--The total number of profile values to be specified (LR has a maximum value of 30). For irregular beam (IPROF=2) distributions only, the user must specify the intensity distribution of the beam on a point-by-point basis by giving the profile irradiance value, PX(L), and associated radial distances, RX(L).

Units: unitless
Suggested input value: dependent upon the user

LTMAX--The parameter that controls the time beyond which the temperature rises of the melanin granules are completely dissipated. LTMAX must be large enough to allow the temperature rises in the melanin granules to decrease to an insignificant value. The suggested value has been found to be adequate, and it is recommended that LTMAX not be less than 2191.

Units: unitless
Suggested input value: 2191

M--The total number of grid spaces in the axial direction, an even integer; currently, M = 28.

Units: unitless

M1--Half the number of uniformly spaced axial increments; currently M1 = 6.

Units: unitless

MAXPRT--The parameter which gives the user the option to control the printing of the predicted threshold laser powers and extent of damage. If MAXPRT equals 1, predicted threshold laser power calculations will be printed only at axial positions anterior to the position of the peak temperature rise, IMAX(IMAX-2 LIMAX to IMAX). If MAXPRT equals 2, the printouts will be for axial positions both anterior and posterior to the peak temperature rise position (IMAX-LIMAX to IMAX + LIMAX). If MAXPRT equals 3, printouts will be made only for axial positions posterior to IMAX (IMAX to IMAX + 2* LIMAX).

Units: unitless
Suggested input value: 1--anterior side of peak temperature
2--both sides of peak temperature
3--posterior side of peak temperature

N--The total number of grid spaces in the radial direction; currently, N=13.

Units: unitless

N1--The number of uniform grid increments in the radial direction; currently N1=4.

Units: unitless

NA(L), L=1,22--The refractive index of the ocular media as a function of wavelength. They should be placed on the data card in increasing wavelength sequence from 350 nm, at 50-nm increments.

Units: unitless
Suggested input values: Table A-7 (for water)

NB--The index of refraction of the ocular media at a 500-nm wavelength.
NB is required only if the spread function is used (IFIL=1).

Units: unitless
Suggested input value: 1.336E0 (mainly for water)

NC--The index of refraction of the ocular media for wavelength (WAVEL).
NC is printed only if the spread function is used (IFIL=1).

Units: unitless

NP--Constant used within the program.

Units: unitless

NPT(L), L=1,38--The number of incremental times used to subdivide DPULSE.
It is associated with specific values of FTIME(L), XCT(L), and KTT(L), all of which are associated with a specific range of values of DPULSE and DT. The suggested values were calculated to keep the errors small and satisfy stability requirements for solving the heat-conduction boundary value problem through the use of finite differences.

Units: unitless
Suggested input values: Table A-4

NPULSE(L), L=1, NTEST--The number of pulses associated with a specified test exposure identified by NRUN(L). All other parameters except pulse repetition rate must remain constant for all NRUN(L).

Units: unitless
Suggested input values: dependent upon the user

NTEST--The number of test exposures run which differ only in pulse repetition rate and/or number of pulses. All other parameters must remain fixed from test exposure to test exposure. This allows reducing computation time if only the pulse repetition rate and/or number of pulses differ from run to run. For single-pulse exposures, NTEST=1.

Units: unitless
Suggested input value: dependent upon the user, MAX = 7

PC--The distance from the pupil to the cornea. PC is required only if the spread function is used (IFIL=1).

Units: cm
Suggested input value: 4.0E-1 for humans
3.6E-1 for rhesus monkeys

POW--The total power per pulse incident on the corneal surface; assumed to be constant during the exposure.

Units: watts

Suggested input value: dependent upon the user

PP--The distance between the pupil and the second principal focal plane. It is required only when the spread function is used (IFIL=1).

Units: cm

Suggested input value: 1.35E-1 for humans
1.2E-1 for rhesus monkeys

PTIME--The uniform time increment into which DPULSE is divided for multiple-pulse calculations. For single-pulse exposures, PTIME is not used.

Units: sec

PTIME = DPULSE/NP for multiple pulses.
NP = 5.

PUPIL--The radius of the pupil of the eye.

Units: cm

Suggested input value: 3.5E-1

PX(L), L=1,LR--The absolute or relative irradiance incident on the cornea for an irregular profile at the radial distance from the center of the beam, RX(L). Symmetry with respect to the axis is assumed. PX(L) cannot have a value of zero at the center axis of the beam.

Units: watts·cm⁻²

Suggested input value: dependent upon the user

QD--All QD values in program RE2, and those associated with the last two TSTEAM values in program RE1, are the power per pulse at the specified grid points required to cause irreversible damage as determined by the damage integral. The other QD values in program RE1 are the powers required to raise the temperature to TSTEAM. QD is set equal to 1.0E+20 when the temperature rise is less than 10⁻³°C.

Units: watts

QP--The laser intensity at R(1), the center of the beam, entering the eye after the corneal reflection.

Units: cal·cm⁻²·sec⁻¹

R(J), J=1,N+1--The radial distance measured from the center of the beam.

Units: cm

R1--The exponential stretching factor in the axial direction for the non-uniform part of the grid system.

Units: unitless

R2--The exponential stretching factor in the radial direction for the non-uniform part of the grid system.

Units: unitless

RC0--The fraction of light reflected from the cornea.

Units: unitless

Suggested input value: Tables A-1 and A-2.

REPET(L), L=1, NTEST--The repetition rate associated with the specific test exposure identified by NRUN(L). All other parameters except the number of pulses must remain constant for all NTEST runs. For a single pulse exposure, set REPET(L)=1. If both NPULSE and NTEST=1, REPET is read but not used in the program.

Units: Hz

Suggested input values: dependent upon the user

RIM--The beam radius at the cornea if the spread function is used (IFIL=1) or at the retina if the spread function is not used (IFIL=0). It is specified at CUT for gaussian profiles (IPROF=1). Although not used for irregular profiles (IPROF=2), a value must always be specified for RIM. For uniform profiles (IPROF=0), it is used with LIM to establish the minimum radial grid increment DR.

Units: cm

Suggested input value: dependent upon the user

RINT--A radial interval used in the input-profile evaluation and in the spread-function integration. It is only printed when the spread function (IFIL=1) or irregular profile (IPROF=2) is used.

Units: cm

$RINT = PUPIL / (LI - 1)$, LI=500

RMAX--The maximum radial distance at which damage assessments are to be made. The model assesses damage at all grid points from R(J)=0 to the first grid point beyond R(J)=RMAX.

Units: cm

Suggested input value: 0.001

RPE--A fraction, ranging from zero to one, used to determine the thickness of the two sublayers of the pigment epithelium. RPE is used in conjunction with IGX to determine the absorption properties (APE1 and APE2) of the two sublayers; it cannot equal IGX, thus avoiding a division by zero. RPE represents the fraction of the total thickness of the pigment epithelium (TPE) occupied by the anterior sublayer.

Units: unitless

Suggested input value: 0.--no anterior sublayer
0.33--monkey eye
0.67--human eye
1.0--no posterior sublayer

$RPE \cdot TPE$ = thickness of anterior sublayer

$1 - RPE \cdot TPE$ = thickness of posterior sublayer

RRT--The fraction of light reflected from the retina.

Units: unitless

Suggested input value: Tables A-1 and A-2

RSC--The fraction of light reflected from the sclera.

Units: unitless

Suggested input value: Tables A-1 and A-2

RVL--The radial extent of the eye; the boundary where no temperature rise occurs.

Units: cm

Suggested input value: 0.7

RX(L), $L=1, LR$ --The radial distance from the center of the beam, that is associated with the profile irradiance value, $PX(L)$, for irregular beam profiles (IPROF=2).

Units: cm

Suggested input values: dependent upon the user

S--The rate of heat deposition from the incoming beam per unit volume at axial distances $Z(I)$ and radial distances $R(J)$. The S printout is given for N radial positions on one line for each axial grid point except those at the boundaries.

Units: $\text{cal} \cdot \text{cm}^{-3} \cdot \text{sec}^{-1}$

SHB--The specific heat of blood.

Units: $\text{cal} \cdot \text{cm}^{-3} \cdot ^\circ\text{C}^{-1}$

Suggested input value: 0.92

SIGMA--The radius of the beam at the point where the intensity is $1/e^2$ of the maximum value. It is used only for gaussian profiles (IPROF=1) and is specified at the cornea if the spread function is used (IFIL=1). If the spread function is not used (IFIL=0), all profile values are considered to be at the retina.

Units: cm

TAV--The thickness of the ocular media from the cornea to the vitreous humor inclusive; the distance from the cornea to the retina.

Units: cm

Suggested input value: Table A-3

TCH--The thickness of the choroid.

Units: cm

Suggested input value: Table A-3

TIME--The maximum time for temperature rise calculations and damage-integral evaluation.

Units: sec

TIME = $DT*(XC**KT-1.)/(XC-1.)$ for single pulse

DT = $DPULSE*(XC-1.)/(XC**NP-1.)$ for single pulse

KT = KTT(L1) for single pulse

NP = NPT(L1) for single pulse

XC = XCT(L1) for single pulse

L1 = $ALOG(DPULSE)/.69315 + 29.$

TIME = FTIME(L1)*X1 for multiple pulse

X1 = NPULSE/REPET largest fraction in all tests

TIMEX(K), K=1,KTYPE--The time at which a plot or selected-time printout of the temperature rises is desired. A separate value of TIMEX(K) must be supplied for each plot or selected-time printout. All values of TIMEX(K) must be less than or equal to the total time over which damage is assessed (TIME).

Units: sec

Suggested input values: dependent upon the user

T0--The initial temperature of the eye.

Units: °C

Suggested input value: 37.

TOM--The transmittance of the ocular media from the anterior surface of the cornea to the pigment epithelium.

Units: unitless

Suggested input value: Tables A-1 and A-2

TPE--The thickness of the pigment epithelium.

Units: cm
Suggested input value: Table A-3

TS(L), L=1,LTMAX,10--The normalized temperature rise decays with respect to time for the melanin granules. They are normalized to the power required to raise a homogeneous pigmented layer an average of 1°C per unit volume and are given in increments of 10BT or 3×10^{-8} sec. Values in Table A-5 were computed for melanin granules 1 μ m wide with a 1.5 μ m separation between adjacent granules.

Units: °C
Suggested input values: Table A-5

TSC--The thickness of the sclera.

Units: cm
Suggested input value: Table A-3

TSTEAM--A temperature defined by the user according to the particular subject being studied. The model computes the power necessary to raise the temperature of the tissue at specified grid points above the temperature TSTEAM. The model will increment TSTEAM by DTSTM and recompute the required power to exceed the new TSTEAM temperature. TSTEAM continues to be incremented by DTSTM until the power to produce irreversible damage predicted by the damage-integral method is less than the power required to raise the tissue above the temperature TSTEAM. At this point, the power predicted by the damage-integral method is printed. When this occurs twice in sequence, the computation is stopped. This parameter allows the user to determine what powers are necessary to raise the tissue above specified temperatures and to determine the power needed to cause irreversible damage in the tissue.

Units: °C
Suggested input value: 200.

TVL--The thickness of the choriocapillaris.

Units: cm
Suggested input value: Table A-3

VSHX(L), L=1,6--The heat capacity of the Lth ocular media.

Units: $\text{cal} \cdot \text{cm}^{-3} \cdot ^\circ\text{C}^{-1}$
Suggested input value: 1.0

WAVEL--The wavelength of the laser radiation in air.

Units: nm
Suggested input value: 400.-1200. nm

XC--The stretching factor for time intervals associated with temperature calculations.

Units: unitless

XC = XCT(L1) single pulse
L1 = ALOG(DPULSE)/.69315 + 29.
XC = 1.4 multiple pulse

XCT(L1), L1=1,38--An array of expansion factors for calculating time intervals in a single-pulse exposure run.

Units: unitless
Suggested input values: Table A-4

L1 = ALOG(DPULSE)/.69315 + 29.

XFLOW--The rate of blood flow to the tissues surrounding the eye.

Units: $\text{g}\cdot\text{cm}^{-3}\cdot\text{sec}^{-1}$
Suggested input value: .001

XFLOWO(L1), L1=1,6--The total blood flow per unit area leaving the choriocapillaris at a given radial distance.

Units: $\text{g}\cdot\text{cm}^{-2}\cdot\text{sec}^{-1}$

XPDK(K), K=1,KT--The normalized temperature rise of the melanin granules at times XT(K). The temperature rises are normalized to the average temperature rise that would occur if the melanin granules were not present. Therefore, if the effects of the melanin granules are not significant, the values for XPDK(K) will be 1.0. XPDK(K) values are printed in consecutive order for each time that temperature rises are printed.

Units: unitless

XT(K), K=1,KT--The time following the start of an exposure.

Units: sec

Z--In the program output sections Temperature Rises, Predicted Threshold Laser Power, and Radial Extent of Damage, Z is the axial depth from the anterior boundary of the pigment epithelium at which temperature rise and/or damage predictions are printed. Positive and negative numbers indicate axial distances posterior and anterior, respectively,

to the boundary of the vitreous humor and pigment epithelium. In the Grid Information output section, Z is the axial distance from the front of the cornea to the individual grid points.

Units: cm

ZH(I), I=1,M--An axial distance from the cornea to points located half-way between the axial grid points Z(I) and Z(I+1).

Units: cm

ZM--Half the length of the z-axis of the modeled eye.

Units: cm

Z0--The distance of the pupil from the nearest laser beam waist. It must be a positive value; i.e., only diverging beams are applicable. Z0 is required only when the spread function is used (IFIL=1).

Units: cm

Suggested input value: $2 \cdot \text{RIM} / \text{full-angle divergence at RIM}$, angle in radians.

TABLE A-1. OPTICAL PARAMETERS FOR CAUCASIANS AND NEGROES*

Wavelength nm (WAVEL)	Total transmission from cornea to retina (TOM)	Absorption coefficient (cm^{-1}) for				Reflection from		
		pigment epithelium (APE)		choroid (ACH)		cornea (RCO)	retina (RRT)	sclera (RSC)
400.0	.094	Caucasian 1838.	Negro 1838.	Caucasian 240.	Negro 240.	.025	.078	-
500.0	.763	1827.	1355.	106.	203.	.025	.091	.530
514.5	.777	1745.	1261.	99.	195.	.025	.097	.543
520.8	.782	1711.	1223.	97.	194.	.025	.099	.549
530.0	.791	1664.	1170.	93.	192.	.025	.104	.557
600.0	.823	1371.	850.	68.	179.	.025	.133	.620
647.1	.848	1253.	743.	61.	168.	.025	.160	.620
694.3	.853	1144.	643.	54.	159.	.025	.186	.620
700.0	.854	1132.	632.	53.	158.	.025	.189	.620
800.0	.840	974.	433.	43.	136.	.025	.289	.580
900.0	.763	524.	271.	36.	120.	.025	.398	.520
1000.0	.394	373.	205.	32.	109.	.025	.409	.450
1060.0	.492	247.	131.	31.	106.	.025	.450	.426
1064.0	.499	238.	126.	31.	105.	.025	.453	.424
1100.0	.562	164.	79.	30.	103.	.025	.478	.410
1200.0	.082	191.	223.	29.	100.	.025	.394	.320

*From Takata, "Thermal model of laser-induced eye damage."

TABLE A-2. OPTICAL PARAMETERS FOR RHESUS MONKEYS*

Wave length nm (WAVELENGTH)	Total transmission from cornea to retina (TOM)	Absorption coefficient (cm ⁻¹) for			Reflection	
		pigment epithelium (APE)	choroid (ACH)	cornea (RCO)	retina (RTT)	sclera (RSC)
400.0	.077	1852.	187.	.025	.080	.360
500.0	.826	1545.	169.	.025	.070	.325
514.5	.836	1485.	166.	.025	.070	.318
520.8	.841	1460.	164.	.025	.070	.315
530.0	.847	1425.	163.	.025	.070	.310
600.0	.877	1194.	151.	.025	.070	.265
647.1	.882	1108.	145.	.025	.075	.242
694.3	.887	1028.	141.	.025	.079	.231
700.0	.887	1019.	140.	.025	.080	.230
800.0	(.892) ^a	838.	123.	.025	.095	.250
900.0	(.878)	605.	114.	.025	.144	.245
1000.0	(.790)	434.	110.	.025	.210	.240
1060.0	(.814)	363.	108.	.025	.252	.252
1064.0	(.816)	258.	108.	.025	.255	.253
1100.0	(.830)	313.	107.	.025	.280	.260
1200.0	(.315)	303.	100.	.025	.260	.215

^aExtrapolated values

*From Takata, "Normal model of laser-induced eye damage."

TABLE A-3. THICKNESS OF OCULAR MEDIA*

<u>Code</u>	<u>Eye media</u>	<u>Thickness in cm</u>	
		<u>Monkey</u>	<u>Man</u>
TAV	Cornea	$5.16 \cdot 10^{-2}$	$5.86 \cdot 10^{-2}$
	Aqueous humor	$2.9 \cdot 10^{-1}$	$3.1 \cdot 10^{-1}$
	Lens	$3.5 \cdot 10^{-1}$	$3.6 \cdot 10^{-1}$
	Vitreous humor	1.157	1.697
TPE	Pigment epithelium	$1.2 \cdot 10^{-3}$	$1.4 \cdot 10^{-3}$
TVL	Choriocapillaris	$1.0 \cdot 10^{-3}$	$1.2 \cdot 10^{-3}$
TCH	Choroid	$1.68 \cdot 10^{-2}$	$1.42 \cdot 10^{-2}$
TSC	Sclera	$1.0 \cdot 10^{-1}$	$1.0 \cdot 10^{-1}$

*From Takata, "Thermal model of laser-induced eye damage."

TABLE A-4. PARAMETERS FOR COMPUTING TIME INTERVALS*

<u>L</u>	<u>NPT(L)</u>	<u>CT(L)</u>	<u>KTT(L)</u>	<u>L</u>	<u>NPT(L)</u>	<u>XCT(L)</u>	<u>KTT(L)</u>
1	1	1.2	47	20	39	1.15	55
2	3	1.2	47	21	40	1.15	56
3	5	1.2	47	22	41	1.15	57
4	7	1.2	47	23	42	1.15	58
5	10	1.2	47	24	43	1.15	59
6	14	1.2	47	25	44	1.15	60
7	18	1.2	48	26	45	1.15	61
8	21	1.2	48	27	46	1.15	62
9	25	1.2	49	28	47	1.15	63
10	28	1.2	49	29	48	1.1	64
11	30	1.2		30	49	1.1	64
12	31	1.2		31	50	1.1	65
13	32	1.2		32	51	1.1	66
14	33	1.2	51	33	52	1.1	67
15	34	1.15	52	34	53	1.1	68
16	35	1.15	52	35	54	1.1	69
17	36	1.15	53	36	55	1.1	69
18	37	1.15	54	37	56	1.1	70
19	38	1.15	54	38	57	1.1	70

*From Takata, "Thermal model of laser-induced eye damage."

TABLE A-5. NORMALIZED TEMPERATURE RISE DECAYS*

L	TS(L)									
1-91	8.000	7.359	6.815	6.349	5.944	5.590	5.277	4.999	4.749	4.523
101-191	4.319	4.132	3.960	3.802	3.656	3.520	3.394	3.276	3.166	3.062
201-291	2.965	2.874	2.787	2.706	2.629	2.557	2.488	2.423	2.361	2.303
301-391	2.247	2.194	2.144	2.096	2.051	2.007	1.966	1.927	1.889	1.853
.	1.819	1.786	1.755	1.725	1.697	1.670	1.644	1.619	1.595	1.572
.	1.550	1.529	1.509	1.490	1.472	1.454	1.437	1.421	1.405	1.390
.	1.376	1.362	1.349	1.336	1.324	1.312	1.301	1.290	1.280	1.270
.	1.260	1.251	1.242	1.233	1.225	1.217	1.209	1.202	1.195	1.188
.	1.181	1.175	1.169	1.163	1.157	1.152	1.147	1.141	1.137	1.132
1001-1091	1.127	1.123	1.119	1.115	1.111	1.107	1.103	1.100	1.096	1.093
.	1.090	1.087	1.084	1.081	1.078	1.075	1.073	1.070	1.068	1.066
.	1.063	1.061	1.059	1.057	1.055	1.053	1.052	1.050	1.048	1.047
.	1.045	1.043	1.042	1.041	1.039	1.038	1.037	1.035	1.034	1.033
.	1.032	1.031	1.030	1.029	1.028	1.027	1.026	1.025	1.024	1.024
.	1.023	1.022	1.021	1.021	1.020	1.019	1.019	1.018	1.017	1.017
.	1.016	1.016	1.015	1.015	1.014	1.014	1.013	1.013	1.012	1.012
.	1.012	1.011	1.011	1.010	1.010	1.010	1.009	1.009	1.009	1.008
.	1.008	1.008	1.008	1.007	1.007	1.007	1.007	1.006	1.006	1.006
.	1.006	1.006	1.005	1.005	1.005	1.005	1.005	1.005	1.004	1.004
.	1.004	1.004	1.004	1.004	1.004	1.004	1.003	1.003	1.003	1.003
2001-2091	1.002	1.002	1.002	1.002	1.002	1.002	1.002	1.002	1.002	1.002
2101-2191	1.002	1.001	1.001	1.001	1.001	1.001	1.001	1.001	1.001	1.001

*From Takata, "Thermal model of laser-induced eye damage."

TABLE A-6. ZERO-ORDER BESSEL FUNCTION*

<u>L</u>	<u>Zero Order Bessel Fn J0(L)</u>	<u>L</u>	<u>Zero Order Bessel Fn J0(L)</u>
1	1.00000	17	.45540
2	.99750	18	.39798
3	.99002	19	.33998
4	.97762	20	.28181
5	.96039	21	.22389
6	.93846	22	.16660
7	.91200	23	.11036
8	.88120	24	.05553
9	.84628	25	.00250
10	.80752	26	-.04838
11	.76519	27	-.09680
12	.71962	28	-.14244
13	.67113	29	-.18503
14	.62008	30	-.22431
15	.56685	31	-.26005
16	.51182	32	-.29206

*From Takata, "Thermal Model of laser-induced eye damage."

TABLE A-7. REFRACTIVE INDEXES*

<u>L</u>	<u>Wavelength nm</u>	<u>Refractive index NA(L) (WATER)</u>	<u>L</u>	<u>Wavelength nm</u>	<u>Refractive index NA(L) (WATER)</u>
1	350	1.357 (not water)	12	900	1.328
2	400	1.346 (not water)	13	950	1.327
3	450	1.341 (not water)	14	1000	1.326
4	500	1.336	15	1050	1.325
5	550	1.334	16	1100	1.324
6	600	1.332	17	1150	1.3235
7	650	1.331	18	1200	1.323
8	700	1.330	19	1250	1.322
9	750	1.329	20	1300	1.321
10	800	1.328	21	1350	1.320
11	850	1.327	22	1400	1.320

*From Takata, "Thermal Model of laser-induced eye damage."

APPENDIX B

PLOT ROUTINE

The plot routine was developed to display two- and three-dimensional temperature rise profiles as a function of radial and axial coordinates at selected times. At specified grid points, the routine utilizes card-punched temperature data that are output by the retinal program, and will generate, for each data set, as many plots as desired. For each plot, the user specifies the physical size of the plot area on the Model 1765 Calcomp plotter and can view the profiles at any angle desired by using a succession of rotation, scaling, and translation commands.

The R-axis, Z-axis, and T-axis of the temperature rise plots refer to the radial coordinates, the axial coordinates, and the temperature rises, respectively. The permanent viewing axes (x,y,z) are set up in a right-hand coordinate system with the permanent x-axis horizontal to the right, the y-axis vertical and up, and the z-axis coming perpendicularly out of the viewing plane. Initially the RZT axes and the permanent xyz axes have the same orientation and origin. All rotations and translations are in relation to the permanent axes and independent of any previous commands. Good three-dimensional views are obtained by a succession of these commands.

The input deck for the plot routine for a single set of temperature data can be separated into two sections. The first section consists of the necessary temperature data on cards that are punched as output from the retinal program. The data cards, for each selected time, should be placed as input to the plot routine in the order in which they are punched with one exception. After the retinal program punches the cards containing the temperature data for the selected times from any one run, it punches MAX RGV CARD(S) FOLLOW on a comment card. This is followed by a number of cards, equal to the number of selected times (KTYPE) and each containing the maximum temperature rise. This maximum rise is used to determine a scaling factor for the temperature rises. The scaling factor is a power of 10 chosen internally to put the maximum temperature rise in the 1-to-12 range. The comment card (MAX RGV CARD(S) FOLLOW) should be discarded. A maximum-temperature-rise card must be placed at the end of the set of temperature data for each selected time. When cards have been punched for more than a single selected time (KTYPE > 1), the end of each set of temperature data can be found by locating the initial card of the succeeding set. This initial card contains NRUN, NPULSE, and REPET and is the only card with the format 2I7, E10.4. When only one selected time (KTYPE=1) has been punched, the only deck manipulation is to discard the comment card.

The second section contains the command data for plotting. These commands scale, rotate, and translate the axes and establish the viewing screen for the desired plots. A uniform format for all commands and their associated parameters is used:

Columns 1-4	Columns 11-20	Columns 21-30	Columns 31-40
keyword	1st	2d	3d
left-justified	parameter	parameter	parameter, etc.

To identify the desired command, a keyword in alphanumeric format is entered in columns 1-4. Parameters, as applicable, are entered as floating-point numbers in 10-character-wide fields starting with column 11. A blank entry is always read as a floating number with value zero.

Several commands normally precede any others when the sequence of input plotting commands is set up. The first command, DUM, has no effect on the actual plot setup, but requests a summary of all the points in the data base--with the low, high, and mean values for the R, Z, and T ranges of data to be printed.

The second command, SCRN, sets up the size and position of the display area as measured on the Calcomp plotter. Without this command, the program will not plot.

The third command in the sequence, BOX, scales the object to fit the viewing area established by SCRN and centers the object on the origin. This eliminates losing plots due to disparity between coordinate magnitudes.

Without any further information, the program would plot an isometric R-Z view of the object, giving a plot of the radial vs axial grid points. Table B-1 contains a sample input deck used to obtain an R-T view (radial vs temperature), a Z-T view (axial vs temperature), and a good three-dimensional isometric view. Cards 1 through 17 contain the data and information supplied by the Retinal Thermal Model. Within this section, cards 11 through 16 contain the actual temperature data to be plotted. Card 17 is the maximum RGV value card. Cards 18 through 33 contain the individual plot commands. These can be used with any set of input data to obtain the same basic results. The plots generated by these commands are shown in Figures B-1 through B-3.

The three rotation commands, PITC, YAW, and ROLL (about the permanent xyz axes) are the most commonly used commands to move the object and obtain the desired view. The command TRAN can also be used to move the object through a translation relative to the permanent origin.

In addition to the positional commands, several commands can be used to scale the temperature rise profiles and change the viewing perspective. Two commands (besides BOX) have a scaling effect on the plots: SCAL can rescale the R, Z, and T coordinates independently; and FACT simply blows up or shrinks all plotting by applying the same scaling factor to all three of the axes. The command DIST, used to determine the viewing perspective of the plotted object, allows the viewer to adjust his position and distance relative to the permanent origin and to specify his distance

from the plane onto which the three-dimensional object has been projected. Without DIST, the program assumes an isometric view, with the permanent origin lying in the projection plane. If an enlargement of a portion of a plot is desired, the WIND command is used. This automatically scales up the area of interest to fill the screen, and the rest of the plot is cut off.

To obtain any plotting, the command PLOT must be used. This calls on the plot subroutine to plot the current view of the object as defined by the previously built-up commands. Normally, the plot includes all lines whether or not they would be seen by the observer of the three-dimensional object. The hidden lines can be dashed or totally removed by using the command HIDE. The visibility of a line is determined by the surface normal vectors entered in the plot file, which can be reversed by the command SIGN. After a plot command, a quick reinitialization of the transformation matrix is achieved by the command REIN. This erases all of the previously built-up results from the positional and the scaling commands.

The sequence of plotting commands listed in Table B-1 is generally adequate for plotting temperature profiles; however, the commands and their sequence can be changed at any time to fit the user's desire. The list in Table B-1 is given as a description of each input command and its associated parameters and is presented in sequence of general usage.

DUM command--Requests a summary of the current number of points in the data base and of the R, Z, and T ranges of the data. DUM is usually the first command entered in any command sequence and has no parameters associated with it.

SCRN(A B C D E) command--Sets up the physical size of the display area and draws a border around that area for every plot. Without either this or the window command, the Calcomp plotter will not plot. For each set of temperature data, the screen command remains in effect and is affected only by subsequent screen commands. Parameters A and B are the coordinates of the lower left corner of the screen in reference to the permanent origin; C and D describe width and height; and E, the fifth parameter, may be entered to define a three-dimensional rectangular box with E as the depth (units are all in inches).

BOX(A B C) command--Causes the object being plotted to fill a fraction of the screen area. The object is first moved so that its center of gravity is coincident with the permanent origin and then rescaled from there to fill a proportion of the available viewing area, as determined by parameters A, B, and C. When only A is entered, a single scale is applied to all three axes. If all three parameters are entered, the object is scaled to fit the A, B, and C proportions of the specified x, y, and z screen dimensions respectively. The parameters are generally set at values slightly less than unity; such as 0.9 or 0.85.

ROLL(A) command--Indicates that the object should move counterclockwise in the viewing plane by an angle of A degrees. The permanent z-axis is the axis of rotation.

PITC(A) command--Indicates that the object should rotate A degrees around the permanent x-axis so that the top part of the screen will come toward the viewer.

YAW(A) command--Indicates that the object should rotate A degrees around the permanent y- (or vertical) axis so that the rightmost portion of the screen will move away from the viewer.

PLOT(A B) command--Causes the current view (as defined by BOX, DIST, ROLL, PITC, YAW) of the object to be plotted. Parameters A and B define the relative X and Y advance on the Calcomp plotter for a permanent new origin of coordinates. A and B are interpreted as real inches. So that the title of the plots and scaling information will be appropriately displayed for each set of plotting data, A=12.75 must be on the first PLOT command card used. Also, B=0 must be on every PLOT command card after the first so that succeeding plots have a common baseline.

DIST(A B X Y) command--Adjusts the distance of the observer from the object. If no parameter or zero-valued parameters are entered, the view will be isometric. If both A and B are nonzero, A is the distance of the viewer from the projection plane and B is the distance of the viewer from the permanent origin. When A is nonzero and B is zero, parameter A is applied to both distances. Optional third and fourth parameters, X and Y, may be added to allow the viewer to shift his viewing position with respect to the z-axis. (All four parameters are in units of inches.)

REIN command--Reinitializes the object to its original position by unitizing the transformation matrix. All previously built-up results from roll, pitch, yaw, scale, box, and translation commands are lost.

HIDE(A) command--Calls for a change in the use of the hidden-line calculation. Through this calculation, lines not normally seen by an observer of a three-dimensional object may be dashed or removed. If parameter A is zero, the hidden-line calculation is not used and all lines are drawn. If A is 1.0, the hidden lines are removed; and if A is 2.0, the hidden lines are dashed. The most recent HIDE command will remain in effect until it is superseded by another HIDE command.

FACT(A) command--Simply expands or shrinks all plotting dimensions along all axes by factor A.

SCAL(A B C) command--Rescales the current object. If factors B and C are both zero, all three dimensions are rescaled uniformly by factor A. In this situation, the commands SCAL and FACT are identical. Otherwise, the R, Z, and T coordinates are independently scaled by factors A, B, and C respectively.

SIGN (A) command--Used to reverse the sense of the surface normals entered in the data base. To do this, A should be set equal to -1.0.

TRAN (A B C) command--Effects a translation of the current object position through a vector (A,B,C) relative to the permanent origin. A, B, and C are in terms of inches along the permanent x, y, and z axes, respectively.

WIND (A B C D) command--Used to zoom in on any portion of the current plot. A and B are the lower left-hand coordinates of the windowed area, and C and D give the horizontal and vertical extent of the windowed area in terms of the permanent display coordinates. The windowed area is then blown up to fill the entire screen area. If the screen command has not been effected, this command acts as a screen with A, B, C, and D having the same meaning as their equivalents for SCRN. The window command is only in effect for the immediately following plot, but can be reactivated by entering a WIND card with no parameters. In this case, the previous window, with its parameters, is put into effect.

TABLE B-1. SAMPLE PLOT INPUT (WITH COMMENTS)

Information and Data from Retinal Model									
1.	3	1	1.0						
2.	.200E-07	.530E 03	.200E 00						
3.	9	14	10	1	6				
4.	11	29							
5.	0.0	0.0003	0.0007	0.0010	0.0013	0.0030	0.0109	0.0496	0.2387
6.	5.6637								1.1613
7.	-0.9575	0.9575	1.5664	1.7599	1.8215	1.8411	1.8473	1.8493	1.8501
8.	1.8503	1.8505	1.8507	1.8509	1.8511	1.8513	1.8515	1.8517	1.8519
9.	1.8523	1.8529	1.8549	1.8611	1.8807	1.9423	2.1358	2.7447	4.6597
10.	.200E-07								
11.	0.840000E-03	0.640000E-03	0.640000E-03	0.330000E-03	0.160000E-03	0.900000E-04	0.100000E-04		
12.	0.461873E 01	0.348269E 01	0.348269E 01	0.178623E 01	0.893650E 00	0.490790E 00	0.545600E-01		
13.	0.209787E 01	0.158187E 01	0.158187E 01	0.811320E 00	0.405900E 00	0.222920E 00	0.247800E-01		
14.	0.565800E-01	0.426600E-01	0.426600E-01	0.218800E-01	0.109500E-01	0.601000E-02	0.670000E-03		
15.	0.541700E-01	0.408400E-01	0.408400E-01	0.209500E-01	0.104800E-01	0.576000E-02	0.640000E-03		
16.	0.524300E-01	0.395300E-01	0.395300E-01	0.202800E-01	0.101400E-01	0.557000E-02	0.620000E-03		
17.	0.461873E 01								
Command Data for Plotting									
18.	DUM								
19.	SCRN	-4.1	8.	6.	6.				(Dump summary of all 3-D points)
20.	BOX	1.0	1.0						(Establish screen size and position)
21.	PITC	-90.							(Rescale object to fill screen)
22.	BOX	0.9	0.9						(Pitch object -90° for an X-Z view)
23.	PLOT	12.5	6.						(Scale view to fill 0.9 of X and Y dimensions of screen)
24.	YAW	-90.							(Advance 12.5 in (31.8 cm) in X, 6 in (15 cm) up in Y, and plot)
25.	BOX	0.9	0.9						(Yaw by -90° to give Y-Z view)
26.	PLOT	10.	0.						(Scale view to fill 0.9 of X and Y dimensions of screen)
27.	REIN								(Advance 10 in (25 cm) in X, and plot)
28.	BOX	1.0	1.0						(Reinitialize to give an isometric X-Y view)
29.	PITC	-90.							(Scale object to fill screen)
30.	YAW	-45.							(Pitch object -90°)
31.	PITC	30.							(Yaw object -45°)
32.	BOX	0.9	0.9						(Pitch object 30°)
33.	PLOT	10.	0.						(Rescale view to fill 0.9 of screen)
									(Advance 10 in (25 cm) in X, and plot)

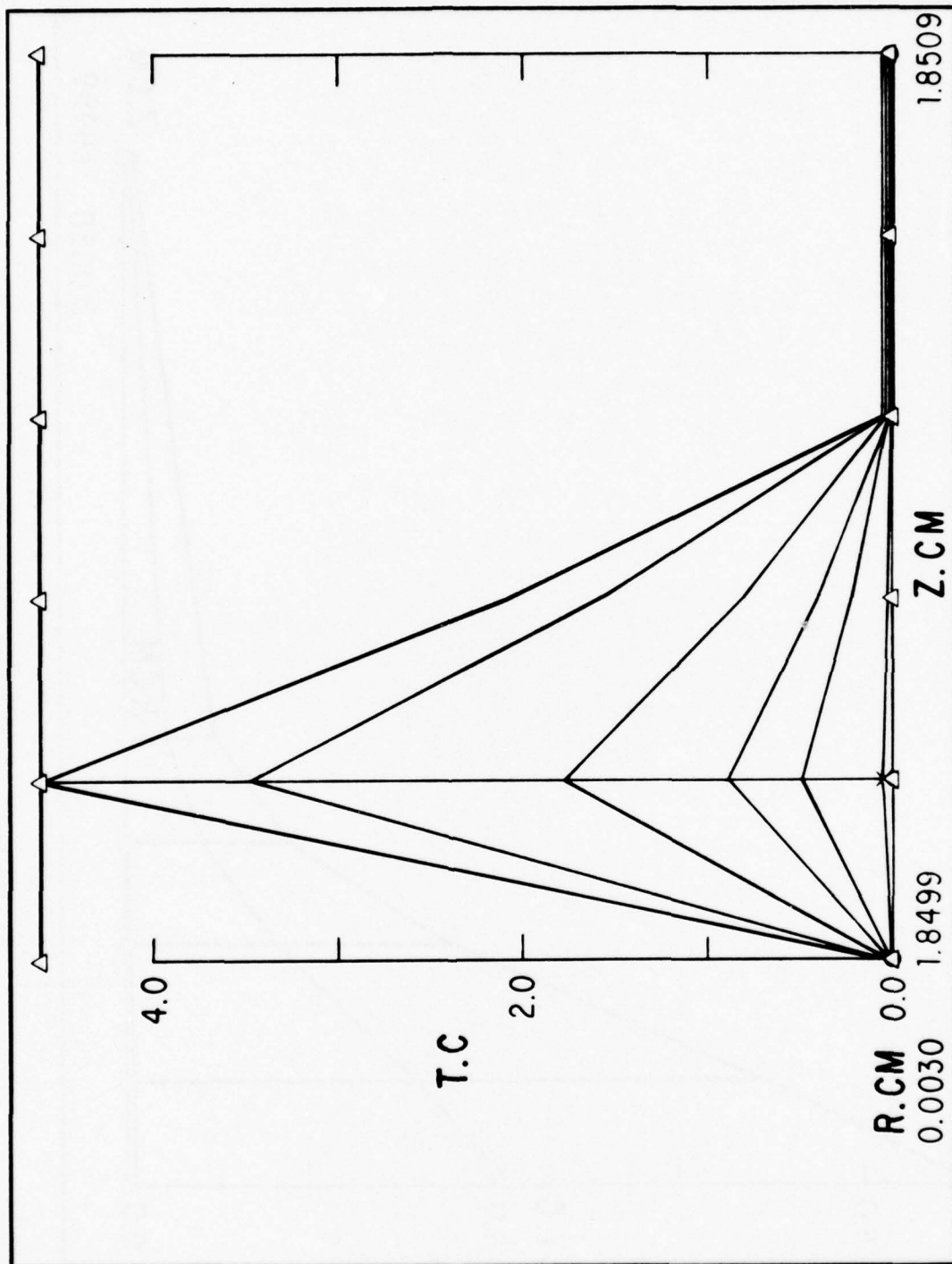


Figure B-1. Temperature rise versus axial depth.

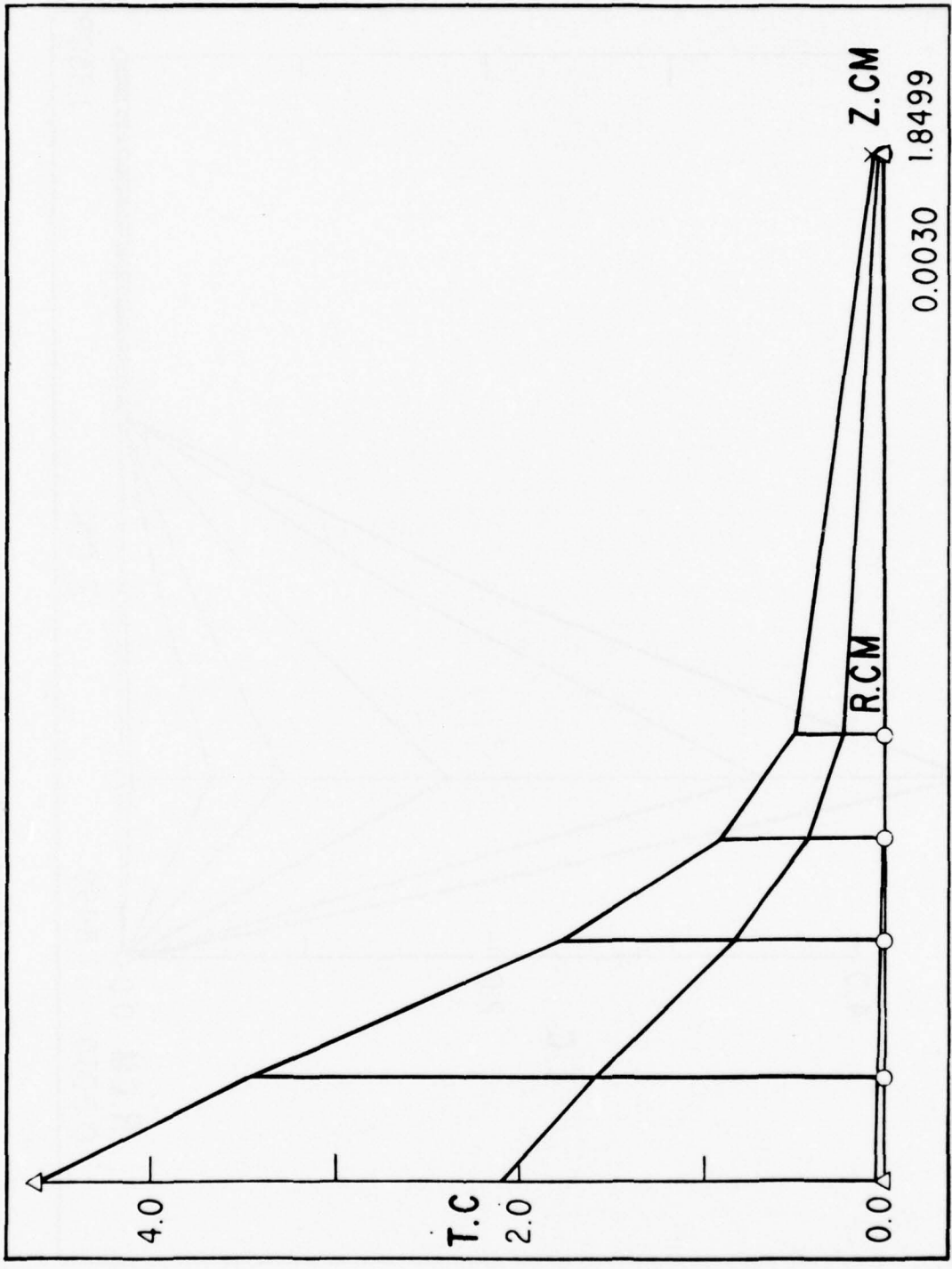


Figure B-2. Temperature rise versus radial extent.

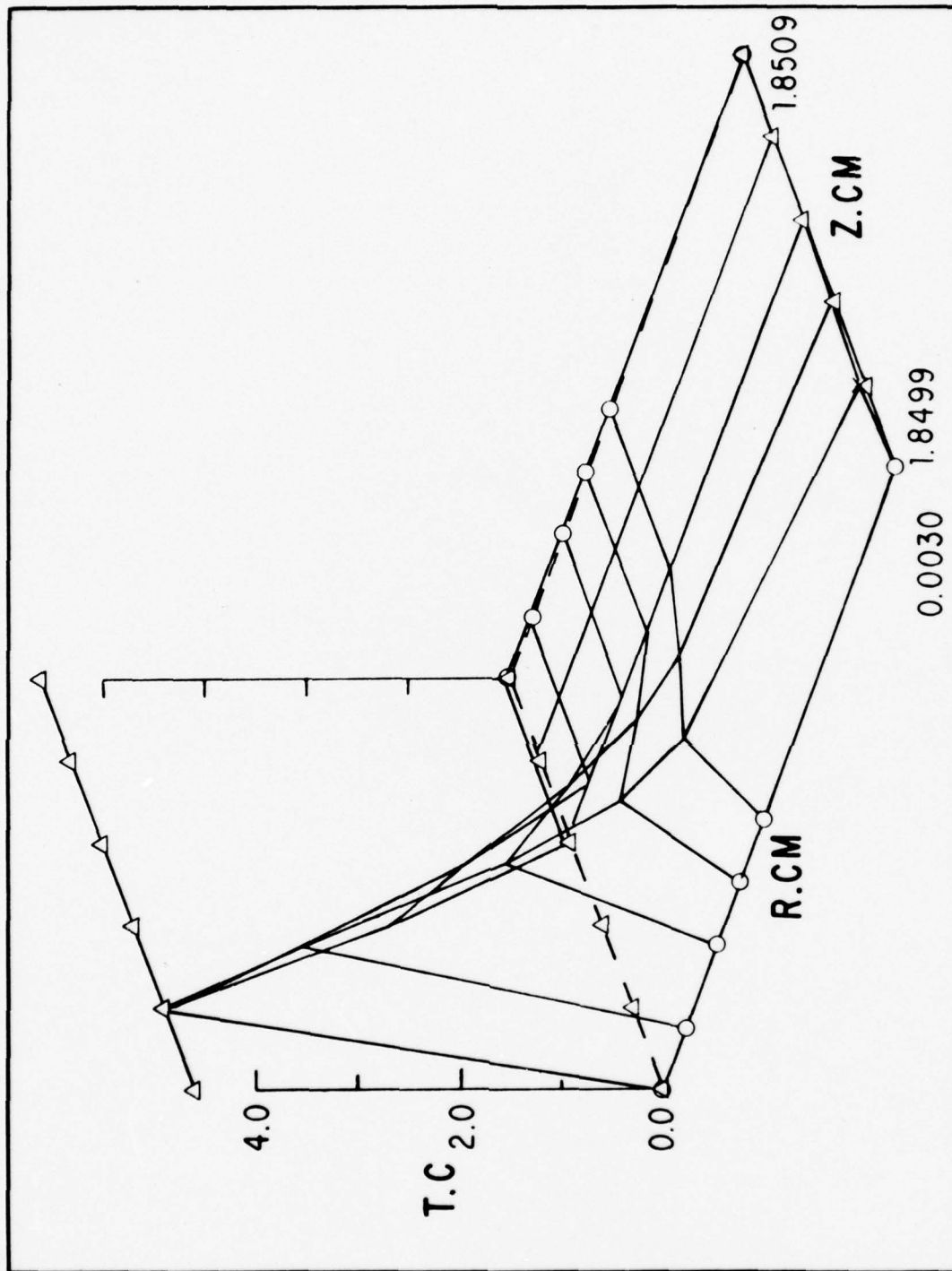


Figure B-3. Three-dimensional view of temperature rise.

APPENDIX C

INPUT-OUTPUT PROCESSES

This appendix will provide the user at Brooks AFB with the basic information necessary to run the retinal model on an IBM 360/65 computer, and will serve as an example for similar setups. One such computer is located at the San Antonio Data Service Center (SADSC), with a remote terminal at Brooks AFB. The prospective user should be familiar with the required input data cards as outlined in text, Input Requirements section.

At Brooks AFB, the retinal models (HBRØ1RE1 and HBRØ1RE2) are stored on a computer disk library. This eliminates having to submit and recompile the entire program for each set of data cards. Therefore, in addition to the data cards, only the Job Control Language (JCL) cards are necessary. These are used to call the stored program and to set up the program for operation on the IBM 360/65. Figure C-1 is an example of a deck used to call up a stored program.

Of the JCL cards, the job (JOB), execute (EXEC), and data definition (DD) cards are required by any IBM 360 operating system. The SETUP card is required by SADSC for long-running programs and programs requiring large core. In addition, several of the parameters on the JOB, EXEC, and DD cards are controlled by SADSC. The following is a list of these JCL cards with the parameters and formats required to call and run the retinal program (HBRØ1RE2) on the computer.

(1) JOB. The job cards identify the beginning of a new job; therefore, they must always be the initial cards in the deck setup. They are variable-field control cards, but have certain requirements placed on them by SADSC. They should fit the following format:

```
//HBaaabbbØJOBØ(3HØ1,BØ2Ø,cccc,ddd,eeee,,Y,ff),'HBgggggg1ØRØhhhhhhhh',
//ØCLASS=H,PRTY=5,MSGCLASS=A,MSGLEVEL=(2,Ø)
```

The parameters that are variable, depending on the user and the job being run, are represented by the lowercase letters above and are as follows (Ø must be blank):

aaa - Unique user code assigned to each user for identity.

bbb - Up to 3 alphanumeric characters (plus @, #, and \$ when desired) assigned by the user to identify the job. This and the user code, together, make up the job name.

cccc - Job execution time in minutes. This is the total estimated time for job execution--the sum of the central processing unit (CPU) time, wait time, and input/output (I/O) time requirements--and may consist of up to 4 digits. Details on the SADSC job class requirements (set forth later in the sequel) will help the user arrive at a suitable time estimate. A suggested time estimate is 1.5 times the amount of CPU time entered on the EXEC card.

ddd - Estimated output line count (in thousands of lines), consisting of up to 3 digits. It is recommended that this value be set at 9 and changed as experience dictates. SADSC operators will automatically cancel the job if the specified line count is exceeded by 9000.

eeee - Estimated card count (in hundreds of plot data cards to be punched), consisting of up to 4 digits. A card count based on an average of 30-40 plot cards for each set of temperature rise values corresponding to a selected time is recommended. SADSC operators will cancel the job if the card count is exceeded by 3000.

ff - Maximum number of lines to be printed per page. This may consist of up to 2 digits, up to a value of 61. To fully utilize the output paper, the value of 61 is recommended.

gggggg - Cost-accounting code associated with the particular work unit under which the job is being run. If this code is less than 6 characters, it must be left-justified, with @ signs acting as fill characters on the right to complete the 6-character subfield.

hhhhhhh - User's last name. Up to 8 alphabetic characters may be used.

The cost-accounting code is the only item requiring its associated subfield to be complete. When not used to the maximum, other subfields should be closed up to include only that portion being used. All other parameters and values on the JOB cards should be included and left as they are. However, the time requirements may necessitate a change in "CLASS=" designation as set forth later in the sequence.

(2) COMMENT. Cards having /* in the first three columns may be used as comment cards to supply information concerning the program to the user. They should be placed after the JOB cards but before any data definition cards. They have no effect upon the running of the program.

(3) SETUP. Special resources required during job execution are indicated by the setup card. It is listed on the computer console when the job enters the system, alerting the SADSC computer operator of any requirements for large amounts of CPU time and/or core storage. For retinal model RE2, the format for this card is:

```
/*SETUP#####'376K CORE ROD, aaa CPU MINS'
```


Columns 8-15 on the card should always be blank, and the message must be in quotes. The number of CPU minutes depends on the number of data sets being run and the data itself. A trial value of 1 minute for each NRUN set of data is suggested. For 4 sets of data, the CPU time would be 4. If CPU time limit exceeds 10 minutes, the job should be submitted as two or more jobs.

(4) EXEC. The execute card tells the computer what type of action the user wants on the source or data deck which will follow. For RE2, the EXEC card has the following format:

```
//STEP1EXECFORTGO,PROGRAM=HBR01RE2,REGION.GO=376K,TIME.GO=aaa
```

This card identifies the GO step as the step to be executed. The GO step calls for execution of the program named HBR01RE2, which has been compiled in FORTRAN IV language. This card further requests a core size of 376K and sets a CPU time limit (aaa) on the execution of the GO step. This time limit should equal the time requirement quoted in the SETUP message. If either the core size or the CPU time limit request is exceeded, program execution will be terminated.

(5) DD. The data definition cards basically supply the computer with descriptions of data sets. Two such cards are required in core loading and running the RE2 program. They are as follows:

```
//STEPLIBDDDSN=SYS1.TESTLIB,DISP=SHR  
//GO.SYSINDD*
```

The first card identifies the system library (TESTLIB) in which the program mentioned in the EXEC statement is stored. The second card identifies the cards which follow it as data cards for the GO step.

(6) DELIMITER (/ *). A card with / * in the first two columns (referred to as a delimiter card) must follow the data card deck. It serves as the end-of-file card for the card deck.

Occasionally, the user may need to recompile the program (RE2) and restore it in the computer library. To do this with a data run would require a deck setup as in Figure C-2. The JOB, SETUP, EXEC, and DD cards require some changes and additions:

(1) JOB. The only change required in the job card for compiling and running the RE2 program is in the estimated job execution time. The usual total estimated job time should be increased by 2 minutes in order to satisfy compiler and linkage editor time requirements.

(2) SETUP. As on the JOB card, the addition of compiler and linkage editor time requirements necessitates an increase in the quoted CPU time requirement. The usual time requirement quoted on the SETUP message for running from the disk library should be increased by 2 minutes to satisfy the extra time requirement.

(3) EXEC. To compile and run the RE2 program, the execute card has the following format:

```
//STEP1 EXEC FTGICLG,REGION.FORT=164K,REGION.LKED=114K,REGION.GO=376K,
//TIME=aaa
```

This card identifies FORT (FORTRAN), LKED (linkage editor), and GO (execution) as steps to be executed. The FORT step compiles the program, the LKED step edits and stores the program, and the GO step executes the program. The card requests core sizes of 164K for FORT, 114K for LKED, and 376K for GO; and sets a CPU time limit (aaa) to accomplish steps FORT, LKED, and GO. This time limit should be equal to the time requirement quoted on the SETUP message.

(4) DD. Compiling, storing, and running the RE2 program requires three data definition cards. They are formatted as follows:

```
//FORT.SYSIN DD *
//LKED.SYSMOD DD DSN=SYS1.TESTLIB(HBR01RE2),DISP=SHR
//GO.SYSIN DD *
```

The first DD card identifies the cards that follow it as source cards for the FORT step. A delimiter card follows the source, or program, deck. Immediately after the source-deck delimiter card, the second DD card directs the computer to store the program in system library TESTLIB under the name HBR01RE2. The last DD card identifies the cards that follow it as data cards for the GO step. A delimiter card is at the end of the data deck.

The SADSC IBM 360/65 computer system has a scanning procedure in operation to detect JCL card error. Detection of a single JCL error by the scanner terminates further processing of the job. One such error detected is a job class error. Job class is determined by use of core requirements and CPU characteristics. Specifically, the ratio of estimated job time (on JOB card) to the time request entered on the EXEC card is considered as $\geq 2:1$ or $< 2:1$, and the job is considered I/O bound or CPU bound according to these ratio values. The user selects the proper job class by using the following table of job class requirements:

<u>Core requirements</u>	<u>$\geq 2:1$</u>	<u>$< 2:1$</u>	J O B C L A S S e s
Max Region \leq 74K (DEFAULT)	A	B	b
75K \leq Max region \leq 150K	C	D	C
151K \leq Max region \leq 300K	E	F	a
301K \leq Max region	G	H	s
Special classes not verified	O,N,T,J	O,N,T,J	s

If the user wants to run the RE1 program, which uses the MXGRAN sub-routine, the following changes must be made:

(1) The name of the program changed from HBR01RE2 to HBR01RE1.

(2) The core requirement for execution (GO) increased from 376K to 436K on both the SETUP and the EXEC cards. All other parameters would be used as outlined for the RE2 program.

The plot routine is handled in the same manner as the main retinal program and is stored in the computer library; therefore, it has the same basic JCL card setup as has been outlined for the retinal program. An example of a deck used to call and run the plot routine is shown in Figure C-3. The changes that are required are:

(1) The name of the program is HBR01PLT.

(2) The core required by the GO step is 148K. This change should be reflected on the SETUP and EXEC cards.

(3) For normal running, FORTGO on the EXEC card should be replaced by PLOTGO. For compiling, the equivalent of FTG1 on the EXEC card is PLOTG, and the core request for FORT should be REGION.FORT=120K.

(4) The TIME.GO entry on the EXEC card should be approximately 0.05 times the number of plots desired.

(5) Set both estimated time and line count to 10 on the job card, and adjust as experience dictates. The number of cards to be punched should be set to zero.

(6) The ratio of estimated job time (on JOB card) to the time request entered on the EXEC card must be evaluated to determine the proper job class as outlined above. This is controlled by the parameter "CLASS=" on the JOB card.

(7) A delimiter card goes at the end of each set of data.

(8) For a single data set, a DD card (//GO.FT05F0020DD0*) must follow the data-set delimiter card and, in turn, must be followed by a delimiter card. For multiple sets of data to be run for any given job, a DD card having the following format must precede each data set subsequent to the first set:

```
//GO.FT05Faaa0DD0*
```

A 3-digit number (aaa) indexes the sets of data in sequential fashion; for example, aaa=002 for the second set of data [TIMEX(2)], aaa=003 for the third set [TIMEX(3)], etc. A DD card of this format must also follow the last data-set delimiter card and must have the proper index number for an additional data set, but with a delimiter card following it.

The computer terminal and its related facilities at Brooks AFB are controlled by Biometrics Division of the USAF School of Aerospace Medicine.

Before running any jobs, the user should familiarize himself with the area where card decks are submitted and returned and output is picked up. Two tables serve these purposes. Decks to be run are placed in the tray on the input table. Also on the input table is a log sheet on which the user must record each job submitted, and small punch/plot cards which must be filled out and placed with the card deck whenever punched cards or plots are expected as output.

All output, whether printed, plotted, or punched, is placed on an output table. Card decks which have been run are placed in trays on this table, with each tray filed corresponding to a range of user-code initial characters. The space on this table is limited, so users should pick up their card decks and output within a reasonable time.

A requirement for using the computer is having a valid user code. A prospective user can get a user code from the director of the programmers, who can also help in identifying or setting up the proper cost-accounting codes assigned to different work units. If either of these codes are invalid on the JOB card, the job will not run.

Several computer-terminal operators are constantly in the input/output area. Questions regarding any part of the input/output process and requests for assistance with any of the machines may be directed to these operators.

/*
// GO.SYSIN DD *
// STEPLIB DD DSN=SYS1.TESTLIB, DISP=SHR
// STEPI EXEC FORTGO, PROGRAM=HBRO1RE2, REGION.GO = 376K, TIME.GO = 4
// * SETUP
// * TEMPERATURE AND DAMAGE PREDICTIONS IN AND ABOUT RETINA CAUSED BY LASERS
// CLASS = H, PRTY = 5, MSGCLASS = A, MSGLEVEL = (2, 0)
// HBZ67RE2 JOB (3H01, 8020, 6, 9, 200, ,, Y, 61), HBM360@10R ANDERSON ,

Figure C-1. Sample card deck for running retinal program RE2.

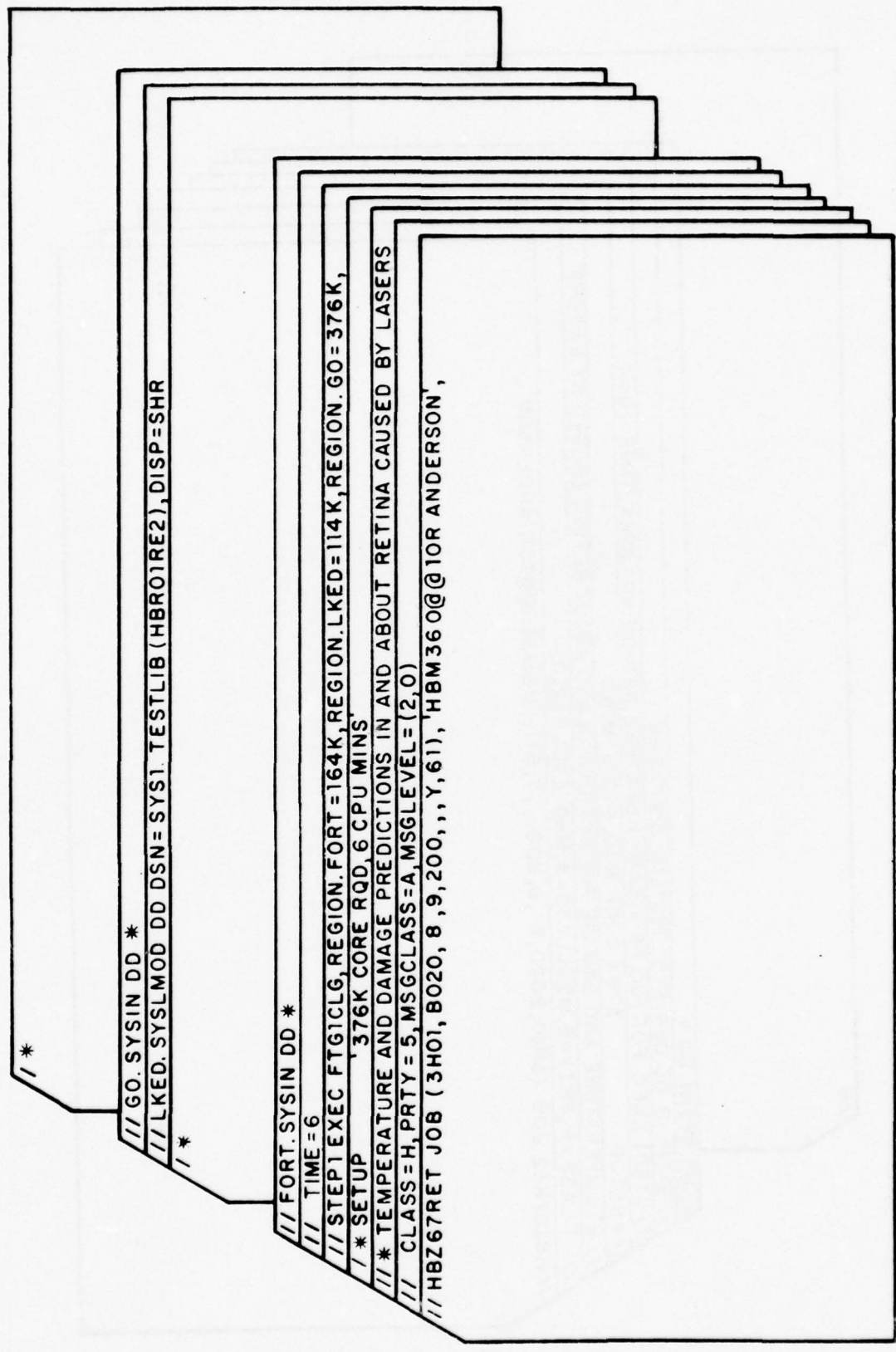


Figure C-2. Sample card deck for compiling and running retinal program RE2.

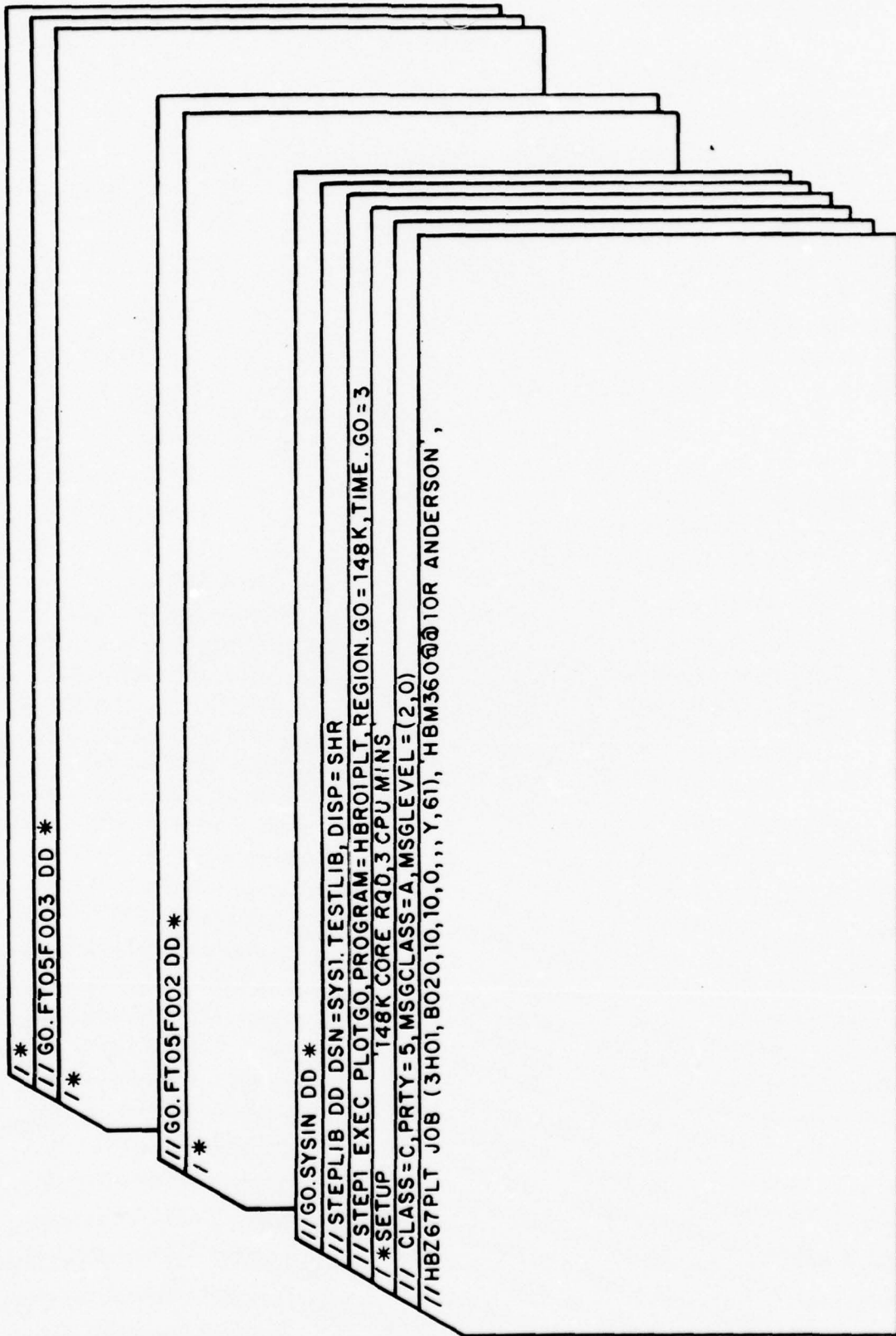


Figure C-3. Sample card deck for running the plot routine.

PRECEDING PAGE, BLANK, NOT FILMED

APPENDIX D
PROGRAM LISTING


```

C                               RETINAL MODEL IITRI                               RE100001
C                               VERSION 1 14 NOV 1975                               RE100002
C TEMPERATURE AND DAMAGE PREDICTIONS IN AND ABOUT RETINA CAUSED BY LASER RE100003
C NEW PRINTOUT TITLES AND GROUPINGS OF INFORMATION                               RE100004
C UTILIZES SUBROUTINE MXGRAN                                                    RE100005
C                                                                              RE100006
COMMON A(29,3),AP,AAV,ACH,APE,ASC,ATS,AVL,B(14,3),BB,BV(14,3),
1CONX(6),CON(29),CUT,DFLOW(6),DPULSE,DR,DT,DTX,DZ,FL,HR(14),
2IAB(29,14),IBLOOD(10),IFIL,IG,IGX,IHT,IPA,IPC,IPE,IPOF,IPS,IPT,
3IPV,IV(29),JVL,LIM,LPA,LPC,LPE,LPS,LPV,LPX,LTMAX,K,KM,KT,M,M1,M2,
4M3,N,N1,N3,N4,NVL,POX,PE(14),PTIME,QP,R(14),RCC,FIM,RN,PPE,RPT,
5RVL,RSC,S(29,14),SHB,TAV,TCH,TOM,TPE,TVL,TS(2200),TSC,TTS,V(29,14)
6,VC(29,14,120),VSH(29),VSHX(6),WAVEL,XC,XFLOW,XFLOWI(6),XFLOWO(6),
7XPD(120),XT(120),Z(29),ZD(8),ZM,FLOWI(14),FLOWX(14),PUPIL,SIGMA,
8IPRT(10),APE1,APE2,RINT,ZO,FLO,CABER,CABER2,PP,PC,NB,NC,FC
DIMENSION CXC(14),CXP(29),DAMAGE(2,2),DXC(14),DXF(29),FTIME(38),
1FXC(14),FXR(29),ID(230),JD(230),KTT(38),NPT(38),NPULSE(7),NFUN(7),
2QD(29,14),REPET(7),TIMEX(10),XCT(38),XQB(29,14),VE(27,120,2),
3VXX(29,14),VZ(27,42,8,2),ZT(8),ZTT(8),ZTX(8),SAVRGV(10)
REAL LESION
2 FORMAT(10F7.3)
3 FORMAT(F7.4,3I7)
4 FORMAT(11F7.2)
5 FORMAT(10I7)
6 FORMAT(F7.2,I7,2F7.2)
7 FORMAT(10E7.2)
8 FORMAT(I7,3E7.2)
9 FORMAT(F7.2,2I7,F7.2)
300 READ(5,4,END=200)(FTIME(L),L=1,38)
READ(5,5) IPRT
READ(5,3) FIM,LIM,IFIL,IGX
READ(5,9) RMAX,LIMAX,MAXPRT,LESION
C *** SET VALUES FOR MTEST,N,N1,N3,N4, AND DR
MTEST=0
N1=4
N=N1+9
N3=N+1
N4=N1+1
READ(5,8) IPOF,POW,CUT
DR=LESION/LIM
IF(IPOF.EQ.0) DR=FIM/(LIM-.5)
READ(5,7) DPULSE
READ(5,5) NTEST,(NFUN(L),L=1,NTEST)
READ(5,7) (REPET(L),L=1,NTEST)
READ(5,5) (NPULSE(L),L=1,NTEST)
READ(5,5) ID1,ID2,JD1,JD2,ITYPE
LPX=1
IF(NTEST.EQ.1.AND.NPULSE(1).EQ.1) LPX=0
XDPULS=DPULSE
XXQ=1.
IF(DPULSE.GT..3E-8) GO TO 10
C *** ADJUST POWER AND PULSE WIDTH FOR EXPOSURES WITH PULSES LESS THAN
C *** .3E-8 SEC
XXQ=.3E-8/DPULSE
POW=POW*DPULSE/.3E-8
DPULSE=.3E-8
10 READ(5,4) TO,EDT1,EDT2

```

READ (5,4) TOM, APE, AVL, ACH, ASC, ATS, RCO, RRT, RSC, RPE, WAVEL	RE100058
READ (5,4) TAV, TPF, TVL, TCH, TSC, RVL	RE100059
AAV=-ALOG (TOM)/TAV	RE100060
READ (5, 4) (CONX (L), L=1,6)	RE100061
READ (5,4) (VSHX (L), L=1,6)	RE100062
READ (5,5) (NPT (L), L=1,38)	RE100063
READ (5,2) (XCT (L), L=1,38)	RE100064
READ (5,5) (KTT (L), L=1,38)	RE100065
C *** COMPUTE DT, KM, KT, NP, PTIME, TIME, AND XC	RE100066
L1=ALOG (DPULSE)/.69315+29.	RE100067
IF (L1.LT.1) L1=1	RE100068
IF (L1.GT.38) L1=38	RE100069
IF (LPX.EQ.1) GO TO 11	RE100070
C *** --- SINGLE PULSED EXPOSURES	RE100071
XC=XCT (L1)	RE100072
NP=NPT (L1)	RE100073
KT=KTT (L1)	RE100074
DT=DPULSE* (XC-1.)/(XC**NP-1.)	RE100075
TIME=DT* (XC**KT-1.)/(XC-1.)	RE100076
GO TO 13	RE100077
C *** --- MULTIPLE PULSED EXPOSURES	RE100078
11 XC=1.4	RE100079
NP=5	RE100080
X1=0.	RE100081
DO 12 L=1, NTEST	RE100082
IF (X1.LT.NPULSE (L)/REPET (L)) X1=NPULSE (L)/REPET (L)	RE100083
12 CONTINUE	RE100084
TIME=FTIME (L1)*X1	RE100085
DT=DPULSE* (XC-1.)/(XC**NP-1.)	RE100086
KT=ALOG (1.+TIME* (XC-1.)/DT)/ALOG (XC)+1.	RE100087
PTIME=DPULSE/NP	RE100088
13 KT=KT+1	RE100089
KM=NP+1	RE100090
IF (KT.GT.119) WRITE (6,14) KT	RE100091
14 FORMAT (1H0, 3HKT=, I3, 2X, 22HTIME DIMENSION TOO LOW)	RE100092
IF (KT.GT.119) STOP	RE100093
C *** CALC. DZ AND I INDICES	RE100094
M1=6	RE100095
M=2*M1+16	RE100096
M2=M/2	RE100097
M3=M+1	RE100098
IPE=M2-M1+2	RE100099
DZ=TPE/M1-1.E-25	RE100100
IPA=2	RE100101
C *** STORE AXIAL DISTANCES TO INTERFACES OF EYE	RE100102
ZD (1)=1.E-25	RE100103
ZD (2)=TAV	RE100104
ZD (3)=ZD (2)+RPE*TPE	RE100105
ZD (4)=ZD (3)+(1.-RPE)*TPE	RE100106
ZD (5)=ZD (4)+TVL	RE100107
ZD (6)=ZD (5)+TCH	RE100108
ZD (7)=ZD (6)+TSC	RE100109
ZD (8)=ZD (7)+10.	RE100110
CALL GRID	RE100111
NVL=LPV-IPV+1	RE100112
C *** CALCULATE AND STORE I, J INDICES AT WHICH TEMPERATURES ARE PRINTED	RE100113
ID1=ID1+IPE	RE100114

```

ID2=ID2+IPE
IF (ID1.LT.IPA) ID1=IPA
IF (ID2.GT.M) ID2=M
IF (JD2.GT.N) JD2=N
IF (IPET (1).EQ.C) GO TO 23
WRITE (6,15) ID1, ID2, JD1, JD2
15 FORMAT (1H0,5X,4HID1=,I3,3X,4HID2=,I3,3X,4HJD1=,I2,3X,4HJD2=,I2)
WRITE (6,16) DR, DZ
16 FORMAT (1H0,5X,3HDR=,E11.4,2X,3HDZ=,E11.4)
WRITE (6,17) IPA, IPC, IPE, IPS, IPT, IPV, LPA, LPC, LPE, LPS, LPV
17 FORMAT (1H0,5X,4HIPA=,I3,2X,4HIPC=,I3,2X,4HIPE=,I3,2X,4HIPS=,I3,2X,
14HIPT=,I3,2X,4HIPV=,I3/1H,5X,4HLPA=,I3,2X,4HLPC=,I3,2X,4HLPE=,I3,
22X,4HLPS=,I3,2X,4HLPV=,I3)
WRITE (6,22) M, M1, N, N1
22 FORMAT (1H0,5X,2HM=,I2,2X,3HM1=,I2,2X,2HN=,I2,2X,3HN1=,I2)
WRITE (6,18) (R (J), J=1, N3)
18 FORMAT (1H0,5X,2HR=/(1H,5X,10P8.4))
WRITE (6,19) (Z (I), I=1, M3)
19 FORMAT (1H0,5X,2HZ=/(1H,5X,10P8.4))
23 DO 20 L1=1, NVL
20 IBLOOD (L1)=IPV+L1-1
C *** CALC. NORMALIZED LASER PROFILES---
DO 21 L=1, N3
21 HR (L)=0.
POX=POW
CALL IMAGE
DO 27 J=1, N3
DO 27 I=1, M3
V (J, J)=1.E-10
27 S (I, J)=0.
READ (5,2) SHB, XFLOW, CFLOW
C *** SET BLOOD FLOW RATES ENTERING AND LEAVING VASCULAR LAYER AS
C *** FUNCTION OF RADIAL DISTANCE
X2=CFLOW/(3.1416*RVL*RVL)
DFLOW (1)=0.
X4=0.
DO 30 L1=2,6
X4=X4+.1
30 DFLOW (L1)=X4
DO 31 L1=1,6
XFLOWI (L1)=X2
31 XFLOWO (L1)=X2
DO 34 I=1, M3
DO 34 J=1, N3
34 VC (I, J, 1)=1.E-10
XPOW=XXQ+POW
READ (5,3) KTYPEO
READ (5,8) KTYPE
L1=KTYPE
IF (KTYPE.EQ.C) L1=1
READ (5,7) (TIMEX (K), K=1, L1)
READ (5,5) II1, II2, II3, JJ1, JJ2
C *** START OF TEMPERATURE CALCULATIONS FOR ONE PULSE. TO BE USED EITHER
C *** FOR MULTIPLE OR SINGLE PULSED EXPOSURES
C -----
XT (1)=0.
DTX=DT

```

```

RE100115
RE100116
RE100117
RE100118
RE100119
RE100120
RE100121
RE100122
RE100123
RE100124
RE100125
RE100126
RE100127
RE100128
RE100129
RE100130
RE100131
RE100132
RE100133
RE100134
RE100135
RE100136
RE100137
RE100138
RE100139
RE100140
RE100141
RE100142
RE100143
RE100144
RE100145
RE100146
RE100147
RE100148
RE100149
RE100150
RE100151
RE100152
RE100153
RE100154
RE100155
RE100156
RE100157
RE100158
RE100159
RE100160
RE100161
RE100162
RE100163
RE100164
RE100165
RE100166
RE100167
RE100168
RE100169
RE100170
RE100171

```

	KTX=KT+1	RE100172
	DO 36 K=2,KTX	RE100173
	XT(K)=XT(K-1)+DT	RE100174
36	DT=XC*DT	RE100175
	IKX=TIME**EDT1+EDT2	RE100176
	IF(IKX.LT.1)IKX=1	RE100177
	XX=2*IKX	RE100178
	K=2	RE100179
	IHT=2	RE100180
	ITYPEX=ITYPE	RE100181
	CALL BLOOD	RE100182
38	DT=XT(K)-XT(K-1)	RE100183
	IF(K.GT.KM)QP=0.	RE100184
	CALL HTXDEP	RE100185
	IF(K.GT.2)GO TO 41	RE100186
	IF(IPRT(2).EQ.0)GO TO 335	RE100187
	WRITE(6,301)	RE100188
301	FORMAT(1H0,13HLASER PROFILE)	RE100189
	IF(IPROP.EQ.0)WRITE(6,302)RIM	RE100190
302	FORMAT(1H0,5X,4HRIM=,E10.3)	RE100191
	IF(IPROP.EQ.1)WRITE(6,303)SIGMA,RIM,CUT	RE100192
303	FORMAT(1H0,5X,6HSIGMA=,E10.3,5X,4HRIM=,E10.3,5X,4HCUT=,E10.3)	RE100193
	IF(IFIL.EQ.1)WRITE(6,304)RINT,ZO,PLO,CABER,CABER2,PP,PC,NB,NC,FC,	RE100194
	1WAVEL	RE100195
304	FORMAT(1H0,5X,5HRINT=,E10.3,3X,3HZO=,E10.3,3X,4HFLO=,F6.3/1H,5X,	RE100196
	16HCABER=,E10.3,3X,7HCABER2=,F7.0,3X,3HPP=,F6.3/1H,5X,3HPC=,F6.3,	RE100197
	23X,3HNB=,F7.3,3X,3HNC=,F7.3/1H,5X,3HPC=,F6.3,3X,6HWAVEL=,F7.1)	RE100198
	IF(IFIL.EQ.1)GO TO 306	RE100199
	IF(IPROP.EQ.2)WRITE(6,305)RINT	RE100200
305	FORMAT(1H0,5X,5HRINT=,E10.3)	RE100201
306	WRITE(6,307)QP	RE100202
307	FORMAT(1H0,5X,3HQP=,E10.3)	RE100203
	WRITE(6,308)(HP(J),J=1,N)	RE100204
308	FORMAT(1H0,5X,3HHR=/(1H,10X,10E10.3))	RE100205
335	IF(IPRT(3).EQ.0)GO TO 340	RE100206
	WRITE(6,309)	RE100207
309	FORMAT(1H0,19HDATA IDENTIFICATION)	RE100208
	WRITE(6,310)(REFET(L),L=1,NTEST)	RE100209
310	FORMAT(1H0,5X,6HREPET=/(1H,5X,10E10.3))	RE100210
	WRITE(6,311)(NPULSE(L),L=1,NTEST)	RE100211
311	FORMAT(1H0,5X,7HNPULSE=/(1H,5X,10I8))	RE100212
	WRITE(6,312)AAV,ACH,APE,ASC,ATS,RCO,RRT,RPE,TOM,AVL,TAV,TCH,TPE,	RE100213
	1TSC,TVL,IGX,IFIL,IPROP,LIM,NTEST,POW,XDPULS,RIM,RMAX,TIME,CFLOW,	RE100214
	2XFLOW,SHB,EDT1,EDT2,DT,KM,KT,PTIME,XC,IKX,AP,APE1,APE2,IG,RVL,	RE100215
	3PUPIL,TO,LIMAX,MAXPRT	RE100216
312	FORMAT(1H0,5X,4HAAV=,F7.1,2X,4HACH=,F7.0,2X,4HAPE=,F7.0,2X,4HASC=	RE100217
	1,F7.0,2X,4HATS=,F7.0/1H,5X,4HRCO=,F7.4,2X,4HRRT=,F7.4,2X,4HRPE=,	RE100218
	2F7.4,2X,4HTOM=,F7.4,2X,4HAVL=,F7.0/1H,5X,4HTAV=,E9.3,2X,4HTCH=,	RE100219
	3E9.3,2X,4HTPE=,E9.3,2X,4HTSC=,E9.3,2X,4HTVL=,E9.3/1H,5X,4HIGX=,I2	RE100220
	4,2X,5HIPIL=,I2,2X,6HIPROF=,I2,2X,4HLIM=,I2,2X,6HNTEST=,I2/1H,5X,	RE100221
	54HPOW=,E9.3,2X,7HDPULSE=,E9.3,2X,4HRIM=,F7.4,2X,5HRMAX=,F7.4,2X,	RE100222
	65HTIME=,E9.3/1H,5X,6HCFLOW=,F7.4,2X,6HXFLOW=,F7.4,2X,4HSHB=,F7.2,	RE100223
	72X,5HEDT1=,F7.4,2X,5HEDT2=,F7.4/1H,5X,3HDT=,E9.3,2X,3HKM=,I3,2X,	RE100224
	83HKT=,I3,2X,6HPTIME=,E9.3,2X,3HXC=,F5.1/1H,5X,4HIKX=,I2,2X,3HAP=,	RE100225
	9F7.4,2X,5HAPE1=,F8.2,2X,5HAPE2=,F8.2,2X,3HIG=,I3/1H,5X,4HRVL=,	RE100226
	1F6.3,2X,6HPUPIL=,F6.3,2X,3HTO=,F5.1,2X,6HLIMAX=,I2,2X,7HMAXPRT=,	RE100227
	2I2)	RE100228

```

340 IF (IPRT(4) .EQ. 0) GO TO 355                                RE100229
WRITE(6,313)                                                    RE100230
313 FORMAT(1H0,30HBLOOD FLOW AND HEAT DEPOSITION)            RE100231
WRITE(6,314) (FLOWI(J),J=1,JVL)                               RE100232
314 FORMAT(1H0,5X,6HFLOWI=/(1H,5X,10E10.3))                 RE100233
WRITE(6,315) (FLOWX(J),J=1,JVL)                               RE100234
315 FORMAT(1H0,5X,6HFLOWX=/(1H,5X,10E10.3))                 RE100235
WRITE(6,316)                                                    RE100236
316 FORMAT(1H )                                                RE100237
DO 318 I=IPA,M                                                  RE100238
WRITE(6,317) (S(I,J),J=1,N)                                    RE100239
317 FORMAT(1H,5X,2HS=,10E8.3)                                  RE100240
318 CONTINUE                                                    RE100241
355 IF (IPRT(5) .EQ. 0) GO TO 41                               RE100242
WRITE(6,319)                                                    RE100243
319 FORMAT(1H0,17HTEMPERATURE RISES)                          RE100244
JCNT=JD2-JD1+1                                                 RE100245
IF (JCNT.GT.9) GO TO 40                                        RE100246
GO TO 41                                                         RE100247
40 JJCNT=JCNT-9                                                RE100248
JJD2=JD2-JJCNT                                                 RE100249
JJD2P1=JJD2+1                                                 RE100250
41 IF (IPRT(5) .EQ. 0) GO TO 356                               RE100251
WRITE(6,42) XT(K),K                                            RE100252
42 FORMAT(1H0,5X,5HTIME=,E11.4,3X,2HK=,I3)                  RE100253
C *** CALCULATE TEMPERATURE RISE (MATRIX REDUCTION ALGORITHM) RE100254
C *** COLUMNS (NORMAL) -----                                RE100255
356 IK=1                                                         RE100256
43 DO 45 I=IPA,M                                               RE100257
W=XX*VSH(I)/DT                                                 RE100258
DO 44 J=1,N                                                     RE100259
FXC(J)=W+CON(I)*B(J,2)-BV(J,2)*IV(I)-BB*IAB(I,J)            RE100260
IF (J.GT.1) FXC(J)=FXC(J)+(CON(I)*B(J,1)+BV(J,1)*IV(I))-CXC(J-1) RE100261
CXC(J)=- (CON(I)*B(J,3)+BV(J,3)*IV(I))/FXC(J)                RE100262
SUM=(W-(A(I,2)-BV(J,2)*IV(I)-BB*IAB(I,J))*V(I,J)+A(I,1)*V(I-1,J)+ RE100263
1A(I,3)*V(I+1,J)+S(I,J))
DXC(J)=SUM/FXC(J)                                              RE100265
IF (J.GT.1) DXC(J)=(SUM+(CON(I)*B(J,1)+BV(J,1)*IV(I))*DXC(J-1))/FXC RE100266
1J)
44 CONTINUE                                                    RE100267
VX=0.                                                           RE100268
DO 45 L=1,N                                                     RE100269
J=N+1-L                                                         RE100270
VX=DXC(J)-CXC(J)*VX                                           RE100271
45 VXX(I,J)=VX                                                 RE100272
DO 46 I=IPA,M                                                  RE100273
DO 46 J=1,N                                                     RE100274
46 V(I,J)=VXX(I,J)                                             RE100275
C *** ROWS (NORMAL) -----                                    RE100276
CXR(IPA-1)=0.                                                  RE100277
DO 50 J=1,N                                                     RE100278
DO 48 I=IPA,M                                                  RE100279
W=XX*VSH(I)/DT                                                 RE100280
FXF(I)=W+A(I,2)-BV(J,2)*IV(I)-BB*IAB(I,J)+A(I,1)*CXR(I-1)  RE100281
CXR(I)=-A(I,3)/FXF(I)                                          RE100282
SUM=(W-(CON(I)*B(J,2)-BV(J,2)*IV(I)-BB*IAB(I,J))*V(I,J)+(CON(I)* RE100283
1B(J,3)+BV(J,3)*IV(I))*V(I,J+1)+S(I,J))                       RE100284

```

	IF (J.GT.1) SUM=SUM+(CON(I)*B(J,1)+BV(J,1)*IV(I))*V(I,J-1)	RE100286
	DXR(I)=SUM/FXR(I)	RE100287
	IF (I.GT.IPA) DXR(I)=(SUM+A(I,1)*DXR(I-1))/FXR(I)	RE100288
48	CONTINUE	RE100289
	VX=0.	RE100290
	DO 50 L=IPA,M	RE100291
	I=M+IPA-L	RE100292
	VX=DXR(I)-CXR(I)*VX	RE100293
	VC(I,J,K)=VX	RE100294
50	VXX(I,J)=VX	RE100295
	DO 51 I=IPA,M	RE100296
	DO 51 J=1,N	RE100297
51	V(I,J)=VXX(I,J)	RE100298
	IK=IK+1	RE100299
C ***	RECYCLE TEMPERATURE CALCULATIONS	RE100300
	IF (IK.LE.IKX) GO TO 43	RE100301
	IF (K.EQ.KM) GO TO 62	RE100302
	IF (ITYPEX.LT.ITYPE.AND.K.LT.KT) GO TO 66	RE100303
62	IF (IPRT(5).EQ.0) GO TO 357	RE100304
	WRITE(6,63) (R(J),J=JD1,JD2)	RE100305
63	FORMAT(1H,13X,2HZ=,9F13.5/1H,15X,30H-----	RE100306
	1--)	RE100307
	DO 65 I=ID1,JD2	RE100308
	X1=Z(I)-Z(IPE)+DZ/2.	RE100309
	IF (JCNT.GT.9) GO TO 57	RE100310
	WRITE(6,64) X1,(VC(I,J,K),J=JD1,JD2)	RE100311
	GO TO 65	RE100312
57	WRITE(6,64) X1,(VC(I,J,K),J=JD1,JJD2)	RE100313
	WRITE(6,64) X1,(VC(I,J,K),J=JJD2P1,JD2)	RE100314
64	FORMAT(1H,3X,2HZ=,F8.5,2X,1P9E13.6)	RE100315
65	CONTINUE	RE100316
357	ITYPEX=0	RE100317
66	K=K+1	RE100318
	ITYPEX=ITYPEX+1	RE100319
	IF (K.LE.KT) GO TO 38	RE100320
	ITYPEX=ITYPE	RE100321
	IF (IPRT(6).EQ.0) GO TO 365	RE100322
	WRITE(6,320)	RE100323
320	FORMAT(1H0,28HNORMALIZED TEMPERATURE RISES)	RE100324
	DO 70 K=2,KT	RE100325
	IF (K.EQ.KM) GO TO 67	RE100326
	IF (ITYPEX.LT.ITYPE.AND.K.LT.KT) GO TO 70	RE100327
67	X1=1.	RE100328
	WRITE(6,321) XT(K),K,X1	RE100329
321	FORMAT(1H0,5X,5HTIME=,E11.4,3X,2HK=,I3,3X,6HPOWER=,E11.4,5HWATTS)	RE100330
	WRITE(6,63) (R(J),J=JD1,JD2)	RE100331
	JCNT=JD2-JD1+1	RE100332
	IF (JCNT.GT.9) GO TO 380	RE100333
	GO TO 381	RE100334
380	JJCNT=JCNT-9	RE100335
	JJD2=JD2-JJCNT	RE100336
	JJD2P1=JJD2+1	RE100337
381	DO 69 I=ID1,JD2	RE100338
	DO 68 J=JD1,JD2	RE100339
68	V(I,J)=VC(I,J,K)/POW	RE100340
	X1=Z(I)-Z(IPE)+DZ/2.	RE100341
	IF (JCNT.GT.9) GO TO 382	RE100342

	WRITE (6,64) X1, (V (I,J) ,J=JD1,JD2)	RE100343
	GO TO 69	RE100344
382	WRITE (6,64) X1, (V (I,J) ,J=JD1,JJD2)	RE100345
	WRITE (6,64) X1, (V (I,J) ,J=JJD2P1,JD2)	RE100346
69	CONTINUE	RE100347
	ITYPEX=0	RE100348
70	ITYPEX=ITYPEX+1	RE100349
C ***	READ NORMALIZED TEMPERATURE RISES TS OF GRANULES FOR .3E-8 PULSE	RE100350
C ***	CALCULATE NORMALIZED RISES XPD FOR ACTUAL PULSE	RE100351
330	FORMAT (1H0,61HDIMENSION OF ARRAYS ASSOCIATED WITH ARGUMENT LIJ IS	RE100352
	1TOO SMALL)	RE100353
365	READ (5,8) LTMAX	RE100354
	DO 71 L1=1,LTMAX	RE100355
71	TS (L1)=1.	RE100356
	READ (5,2) (TS (L) ,L=1,LTMAX,10)	RE100357
	CALL MXGRAN	RE100358
	DO 72 L=1,KT	RE100359
72	XPD (L)=AP*XPD (L)+1.-AP	RE100360
	READ (5,4) (DAMAGE (L2,1) ,DAMAGE (L2,2) ,L2=1,2) ,TSTEAM,DTSTM	RE100361
	WRITE (6,73) WAVEL,TSTEAM,DAMAGE (1,1) ,DAMAGE (1,2) ,DAMAGE (2,1) ,	RE100362
	1DAMAGE (2,2)	RE100363
73	FORMAT (1H0,5X,11HWAVELENGTH=,F7.1,2HNM,3X,7HTSTEAM=,F6.0,3X,7HDAMAR	RE100364
	1GE=,4F9.0)	RE100365
C ***	CALCULATE I,J INDICES AT WHICH DAMAGE CALCULATIONS ARE TO BE MADE	RE100366
	JM=0	RE100367
	DO 74 J=1,N	RE100368
	IF (R (J) .LT. RMAX+.000001) JM=J+1	RE100369
74	CONTINUE	RE100370
	X1=0.	RE100371
	DO 75 I=IPA,M	RE100372
	IF (VC (I,1,KM) .GT. X1) IMAX=I	RE100373
	IF (VC (I,1,KM) .GT. X1) X1=VC (I,1,KM)	RE100374
75	CONTINUE	RE100375
	L=0	RE100376
	GO TO (366,367,368) ,MAXPRT	RE100377
366	LIMAX1=2*LIMAX	RE100378
	LIMAX2=0	RE100379
	GO TO 369	RE100380
367	LIMAX1=LIMAX	RE100381
	LIMAX2=LIMAX	RE100382
	GO TO 369	RE100383
368	LIMAX1=0	RE100384
	LIMAX2=2*LIMAX	RE100385
369	ID1=IMAX-LIMAX1	RE100386
	ID2=IMAX+LIMAX2	RE100387
	IF (ID2.GT.28) ID2=28	RE100388
	DO 76 I=ID1,ID2	RE100389
	DO 76 J=1,JM	RE100390
	L=L+1	RE100391
	ID (L)=I	RE100392
76	JD (L)=J	RE100393
	LIJ= (ID2-ID1+1)*JM	RE100394
	DO 385 LL15=1,10	RE100395
385	SAVRGV (LL15)=0.	RE100396
	IF (LPX.EQ.0) GO TO 125	RE100397
	IF (LIJ.GT.27) WRITE (6,330)	RE100398
	IF (LIJ.GT.27) GO TO 300	RE100399

	IF (IPRT(8).EQ.0) GO TO 370	RE100400
C ***	TEMPERATURE AND DAMAGE EVALUATIONS FOR MULTIPLE PULSES	RE100401
C	-----	RE100402
C ***	EVALUATE TEMPERATURE RISES WITH AND WITHOUT GRANULES	RE100403
	DO 77 L=1,L1J	RE100404
	I=ID(L)	RE100405
	J=JD(L)	RE100406
	VE(L,1,1)=0.	RE100407
	VE(L,1,2)=0.	RE100408
	DO 77 K=2,KT	RE100409
	VE(L,K,1)=VC(I,J,K)	RE100410
	VE(L,K,2)=VC(I,J,K)	RE100411
	IF (I.NE.IG) GO TO 77	RE100412
	VF(L,K,2)=XPD(K)*VC(I,J,K)	RE100413
	IF (VE(L,K,1).LT..C) VE(L,K,1)=0.	RE100414
	IF (VE(L,K,2).LT..C) VE(L,K,2)=0.	RE100415
77	CONTINUE	RE100416
	X60=(XC-1.)/DEX	RE100417
	X61=ALOG(XC)	RE100418
	XSTEAM=TSIFAM	RE100419
370	L13=0	RE100420
371	L13=L13+1	RE100421
	X3=DPULSE+(NPULSE(L13)-1)/REPET(L13)	RE100422
	WRITE(6,78) NRUN(L13),X3,XDPULS,NPULSE(L13),REPET(L13)	RE100423
78	FORMAT(1H0,5X,5HNFUN=,I3,2X,13HTRAIN LENGTH=,E10.3,3HSEC,2X,12HPULS	RE100424
	1SE WIDTH=,E10.3,3HSEC/1H,5X,17HNUMBER OF PULSES=,I5,3X,16HREPETITRE	RE100425
	2ION RATE=,E10.3,10HPULSES/SEC)	RE100426
	IF (IPIL.EQ.0) GO TO 80	RE100427
	WRITE(6,79) RIM,LESION	RE100428
79	FORMAT(1H,5X,12HBFAM RADIUS=,E10.3,2HCM,5X,14HLESION RADIUS=,E10.	RE100429
	13,2HCM)	RE100430
	GO TO 82	RE100431
80	WRITE(6,81) RIM,LESION	RE100432
81	FORMAT(1H,5X,13HIMAGE RADIUS=,E10.3,2HCM,5X,14HLESION RADIUS=,E10	RE100433
	1.3,2HCM)	RE100434
82	IF (IPRT(8).EQ.0) GO TO 108	RE100435
	TC=1./REPET(L13)	RE100436
	NPL=NPULSE(L13)	RE100437
	KX=NP+3	RE100438
	IN=1	RE100439
83	IF (NPL/IN.LT.20) GO TO 84	RE100440
	IN=IN+2	RE100441
	GO TO 83	RE100442
84	X1=NPL	RE100443
	INX=.5+X1/IN	RE100444
	L1=ALOG(DPULSE)/.69315+29.	RE100445
	IF (L1.LT.1) L1=1	RE100446
	INXX=PTIME(L1)*INX	RE100447
C ***	STORE TIME INTERVALS AND LOGS OF INTERVALS FOR DAMAGE CALCULATIONS	RE100448
	ZTX(1)=PTIME	RE100449
	ZT(1)=PTIME/2.	RE100450
	ZTT(1)=ALOG(IN*PTIME)	RE100451
	DO 85 L3=2,NP	RE100452
	ZTT(L3)=ALOG(IN*PTIME)	RE100453
	ZTX(L3)=ZTX(L3-1)+PTIME	RE100454
85	ZT(L3)=ZT(L3-1)+PTIME	RE100455
	L1=NP+1	RE100456

X3=(TC-DPULSE)/(KX-NP)	RE100457
ZTX(L1)=DPULSE+X3	RE100458
ZT(L1)=DPULSE+X3/2.	RE100459
ZTT(L1)=ALOG(IN*X3)	RE100460
L1=L1+1	RE100461
DO 86 L3=L1,KX	RE100462
ZTT(L3)=ALOG(IN*X3)	RE100463
ZTX(L3)=ZTX(L3-1)+X3	RE100464
86 ZT(L3)=ZT(L3-1)+X3	RE100465
C *** CALCULATE TEMPERATURE RISES ASSOCIATED WITH L3-TH TIME INTERVAL	RE100466
C *** FOLLOWING (L6-.5)*IN-.5 PULSE	RE100467
DO 95 L=1,LIJ	RE100468
DO 95 L3=1,KX	RE100469
X1=0.	RE100470
X2=0.	RE100471
L1=1+IN/2	RE100472
L7=1	RE100473
87 X3=(L7-1)*TC+ZT(L3)	RE100474
K=ALOG(X3*X60+1.)/X61+1.	RE100475
X5=VE(L,K,1)+(X3-XT(K))*(VE(L,K+1,1)-VE(L,K,1))/(XT(K+1)-XT(K))	RE100476
X1=X1+X5	RE100477
X3=(L7-1)*TC+ZTX(L3)	RE100478
K=ALOG(X3*X60+1.)/X61+1.	RE100479
X2=X2+VE(L,K,2)+(X3-XT(K))*(VE(L,K+1,2)-VE(L,K,2))/(XT(K+1)-XT(K))	RE100480
IF(X5.LT..0001*X1)GO TO 88	RE100481
L7=L7+1	RE100482
IF(L7.LE.L1)GO TO 87	RE100483
88 VZ(L,1,L3,1)=X1	RE100484
VZ(L,1,L3,2)=X2	RE100485
DO 93 L6=2,INXX	RE100486
IF(X5.LT..0001*X1)GO TO 91	RE100487
X1=VZ(L,L6-1,L3,1)	RE100488
X2=VZ(L,L6-1,L3,2)	RE100489
L2=L1+1	RE100490
L1=L1+IN	RE100491
L7=L2	RE100492
90 X3=(L7-1)*TC+ZT(L3)	RE100493
K=ALOG(X3*X60+1.)/X61+1.	RE100494
X5=VE(L,K,1)+(X3-XT(K))*(VE(L,K+1,1)-VE(L,K,1))/(XT(K+1)-XT(K))	RE100495
X1=X1+X5	RE100496
X3=(L7-1)*TC+ZTX(L3)	RE100497
K=ALOG(X3*X60+1.)/X61+1.	RE100498
X2=X2+VE(L,K,2)+(X3-XT(K))*(VE(L,K+1,2)-VE(L,K,2))/(XT(K+1)-XT(K))	RE100499
IF(X5.LT..0001*X1)GO TO 91	RE100500
L7=L7+1	RE100501
IF(L7.LE.L1)GO TO 90	RE100502
91 VZ(L,L6,L3,1)=X1	RE100503
93 VZ(L,L6,L3,2)=X2	RE100504
L1=INX+1	RE100505
DO 94 L6=L1,INXX	RE100506
L8=L6-INX	RE100507
VZ(L,L6,L3,1)=VZ(L,L6,L3,1)-VZ(L,L8,L3,1)	RE100508
94 VZ(L,L6,L3,2)=VZ(L,L6,L3,2)-VZ(L,L8,L3,2)	RE100509
95 CONTINUE	RE100510
C *** DAMAGE CALCULATIONS -----	RE100511
WRITE(6,375)	RE100512
375 FORMAT('HO, 31HPREDICTED THRESHOLD LASER POWER)	RE100513

TSTEAM=XSTEAM	RE100514
XQ=0.	RE100515
96 WRITE(6,130)TSTEAM	RE100516
DO 104 L=1,LIJ	RE100517
I=ID(L)	RE100518
J=JD(L)	RE100519
IF(VZ(L,INX,NP,1).LT..001)QD(I,J)=1.E+20	RE100520
IF(VZ(L,INX,NP,1).LT..001)GO TO 104	RE100521
L9=10.*(.4+EXP(-.0014*DPULSE))/VZ(L,INX,NP,1)	RE100522
CQ=L9+1.	RE100523
X10=70.*(.4+EXP(-.0014*DPULSE))/VZ(L,INX,NP,1)	RE100524
IF(L9.EQ.0)CQ=X10	RE100525
LLT=0	RE100526
LGT=0	RE100527
99 DAMC=0.	RE100528
L6=1	RE100529
100 DO 101 L3=1,KX	RE100530
X3=0.	RE100531
IF(VZ(L,L6,L3,2)*CQ.GT.TSTEAM-T0)X3=1.E+30	RE100532
IF(VZ(L,L6,L3,2)*CQ.GT.TSTEAM-T0)GO TO 101	RE100533
X50=VZ(L,L6,L3,1)*CQ+273.+T0	RE100534
IF(X50.LT.317.)GO TO 101	RE100535
X1=ZTT(L3)+DAMAGE(1,1)-DAMAGE(1,2)/X50	RE100536
IF(X50.GT.323.)X1=ZTT(L3)+DAMAGE(2,1)-DAMAGE(2,2)/X50	RE100537
IF(X1.GT.0.)X3=1.01	RE100538
IF(X1.GT.0.)GO TO 101	RE100539
X3=EXP(X1)	RE100540
101 DAMC=DAMC+X3	RE100541
IF(DAMC.GT.1.)GO TO 102	RE100542
C *** INCREASE TIME INDICES AND CONTINUE	RE100543
L6=L6+1	RE100544
IF(L6.LE.INXX)GO TO 100	RE100545
C *** ADJUST LASER POWER TO YIELD THRESHOLD DAMAGE AT GIVEN POINT	RE100546
IF(LGT.EQ.1)CQ=1.02*CQ	RE100547
IF(LGT.EQ.1)GO TO 103	RE100548
LLT=1	RE100549
CQ=1.04*CQ	RE100550
GO TO 99	RE100551
102 IF(LLT.EQ.1)CQ=.98*CQ	RE100552
IF(LLT.EQ.1)GO TO 103	RE100553
LGT=1	RE100554
CQ=.96*CQ	RE100555
GO TO 99	RE100556
103 QD(I,J)=CQ*POX	RE100557
104 CONTINUE	RE100558
WRITE(6,63)(F(J),J=1,JM)	RE100559
DO 97 I=ID1,ID2	RE100560
DO 97 J=1,JM	RE100561
97 XQD(I,J)=QD(I,J)*XXQ	RE100562
DO 106 I=ID1,ID2	RE100563
X1=Z(I)-Z(IPE)+DZ/2.	RE100564
IF(JM.GT.9)GO TO 98	RE100565
WRITE(6,105)X1,(XQD(I,J),J=1,JM)	RE100566
GO TO 106	RE100567
98 WRITE(6,105)X1,(XQD(I,J),J=1,9)	RE100568
WRITE(6,105)X1,(XQD(I,J),J=10,JM)	RE100569
105 FORMAT(1H ,2X,2HZ=,F7.5,1X,3HQD=,1P9E13.6)	RE100570

106	CONTINUE	RE100571
	X2=(XQ-QD(IMAX,1))/QD(IMAX,1)	RE100572
	X3=X2*X2	RE100573
	IF(X3.LT..0001)GO TO 108	RE100574
	TSTEAM=TSTEAM+DTSTM	RE100575
	XQ=QD(IMAX,1)	RE100576
	GO TO 96	RE100577
108	IF(KTYPE.EQ.0)GO TO 174	RE100578
C ***	CALCULATE AND STORE (MULTIPLE PULSE EXPOSURE) TEMPERATURES FOR	RE100579
C ***	PLOTTING PROFILES	RE100580
	TC=1./REPET(L13)	RE100581
	NPL=NPULSE(L13)	RE100582
	WRITE(6,139)	RE100583
	DO 123 L15=1,KTYPE	RE100584
	IF(TIMEX(L15).GT.XT(KT))GO TO 123	RE100585
	RGV=0.	RE100586
	L2=TIMEX(L15)/TC	RE100587
	DTIME=TIMEX(L15)-L2*TC	RE100588
	L2=L2+1	RE100589
	DO 116 I=II1,II2	RE100590
	DO 116 J=JJ1,JJ2	RE100591
	X1=0.	RE100592
	DO 113 L6=1,L2	RE100593
	K=ALOG((DTIME+(L6-1)*TC)*X60+1.)/X61+1.	RE100594
	X2=(DTIME+(L6-1)*TC-XT(K))/(XT(K+1)-XT(K))	RE100595
113	X1=X1+VC(I,J,K)+X2*(VC(I,J,K+1)-VC(I,J,K))	RE100596
	V(I,J)=X1	RE100597
	L3=L2-NPL	RE100598
	IF(L3.LE.0)GO TO 115	RE100599
	X1=0.	RE100600
	DO 114 L6=1,L3	RE100601
	K=ALOG((DTIME+(L6-1)*TC)*X60+1.)/X61+1.	RE100602
	X2=(DTIME+(L6-1)*TC-XT(K))/(XT(K+1)-XT(K))	RE100603
114	X1=X1+VC(I,J,K)+X2*(VC(I,J,K+1)-VC(I,J,K))	RE100604
	V(I,J)=V(I,J)+X1	RE100605
115	IF(V(I,J).GT.RGV)RGV=V(I,J)	RE100606
116	CONTINUE	RE100607
	SAVEGV(L15)=RGV	RE100608
	IF(KTYPE.EQ.1)GO TO 121	RE100609
	WRITE(7,117)NPUN(L13),NPULSE(L13),REPET(L13)	RE100610
117	FORMAT(2I7,F10.4)	RE100611
	WRITE(7,118)XDPULS,WAVEL,RIM	RE100612
118	FORMAT(7E11.4)	RE100613
	WRITE(7,119)II1,II2,II3,JJ1,JJ2	RE100614
119	FORMAT(5I7)	RE100615
	WRITE(7,119)N3,M3	RE100616
	WRITE(7,120)(F(J),J=1,N3)	RE100617
120	FORMAT(10F8.4)	RE100618
	WRITE(7,120)(Z(I),I=1,M3)	RE100619
	WRITE(7,118)TIMEX(L15)	RE100620
121	WRITE(6,141)TIMEX(L15)	RE100621
	WRITE(6,63)(R(J),J=JJ1,JJ2)	RE100622
	JCNT=JJ2-JJ1+1	RE100623
	IF(JCNT.GT.9)GO TO 390	RE100624
	GO TO 391	RE100625
390	JJCNT=JCNT-9	RE100626
	JJJ2=JJ2-JJCNT	RE100627

	JJJ2P1=JJJ2+1	RE100628
391	DO 122 I=II1,II2	RE100629
	X1=Z(I)-Z(IPE)+DZ/2.	RE100630
	IF(JCNT.GT.9)GO TO 392	RE100631
	WRITE(6,64)X1,(V(I,J),J=JJ1,JJ2)	RE100632
	GO TO 393	RE100633
392	WRITE(6,64)X1,(V(I,J),J=JJ1,JJJ2)	RE100634
	WRITE(6,64)X1,(V(I,J),J=JJJ2P1,JJ2)	RE100635
393	IF(KTYPE.EQ.1)GO TO 122	RE100636
	WRITE(7,137)(V(I,J),J=JJ1,JJ2)	RE100637
122	CONTINUE	RE100638
123	CONTINUE	RE100639
	RGV=0.	RE100640
	DO 395 LL15=1,KTYPE	RE100641
	IF(SAVRGV(LL15).GT.RGV)RGV=SAVRGV(LL15)	RE100642
395	CONTINUE	RE100643
	WRITE(7,396)	RE100644
396	FORMAT(22HMAX RGV CARD(S) FOLLOW)	RE100645
	DO 397 LL15=1,KTYPE	RE100646
397	WRITE(7,137)RGV	RE100647
	GO TO 174	RE100648
124	FORMAT(1H ,5X,1P9F13.6)	RE100649
137	FORMAT(6F13.6)	RE100650
139	FORMAT(1H0,35HTEMPERATURE RISES AT SELECTED TIMES)	RE100651
141	FORMAT(1H0,5X,5HTIME=,E11.4)	RE100652
145	IF(L13.EQ.NTEST)GO TO 300	RE100653
	GO TO 371	RE100654
C	*** DAMAGE CALCULATIONS FOR SINGLE PULSE	RE100655
C	-----	RE100656
125	WRITE(6,126)NFUN(1),XDPULS,NPULSE(1)	RE100657
126	FORMAT(1H0,5X,5HNFUN=,I3,2X,12HPULSE WIDTH=,E10.3,2X,17HNUMBER OF	RE100658
	1PULSES=,I5)	RE100659
	IF(IPIL.EQ.0)GO TO 127	RE100660
	WRITE(6,79)PIM,LESION	RE100661
	GO TO 128	RE100662
127	WRITE(6,81)PIM,LESION	RE100663
128	IF(IPRT(8).EQ.0)GO TO 150	RE100664
	WRITE(6,375)	RE100665
	XQ=0.	RE100666
129	WRITE(6,130)TSTEAM	RE100667
130	FORMAT(1H0,5X,7HTSTEAM=,F7.0/1H ,5X,10H-----)	RE100668
	DO 138 I=ID1,ID2	RE100669
	DO 138 J=1,JM	RE100670
	IF(VC(I,J,KM).LT..001)QD(I,J)=1.0E+20	RE100671
	IF(VC(I,J,KM).LT..001)GO TO 138	RE100672
	L9=10.*(.4+EXP(-.0014*DPULSE))/VC(I,J,KM)	RE100673
	CQ=L9+1.	RE100674
	X10=70.*(.4+EXP(-.0014*DPULSE))/VC(I,J,KM)	RE100675
	IF(L9.EQ.0)CQ=X10	RE100676
	LLT=0	RE100677
	LGT=0	RE100678
131	DAMC=0.	RE100679
	K=2	RE100680
132	X13=ALOG(XT(K)-XT(K-1))	RE100681
	VPX=(VC(I,J,K)+VC(I,J,K-1))/2.	RE100682
	X3=0.	RE100683
	IF(I.NE.IG)GO TO 133	RE100684

	IF (VPX*XPD(K)*CQ.GT.TSTEAM-T0) X3=1.E+30	RE100685
	IF (VPX*XPD(K)*CQ.GT.TSTEAM-T0) GO TO 134	RE100686
133	X50=VPX*CQ+273.+T0	RE100687
	IF (X50.LT.317.) GO TO 134	RE100688
	X1=X13+DAMAGE(1,1)-DAMAGE(1,2)/X50	RE100689
	IF (X50.GT.323.) X1=X13+DAMAGE(2,1)-DAMAGE(2,2)/X50	RE100690
	IF (X1.GT.0.) X3=1.01	RE100691
	IF (X1.GT.0.) GO TO 134	RE100692
	X3=EXP(X1)	RE100693
134	DAMC=DAMC+X3	RE100694
	IF (DAMC.GE.1.) GO TO 135	RE100695
	K=K+1	RE100696
	IF (K.LT.KI) GO TO 132	RE100697
C ***	ADJUST LASER POWER TO YIELD THRESHOLD DAMAGE AT GIVEN POINT	RE100698
	IF (LGT.EQ.1) CQ=1.02*CQ	RE100699
	IF (LGT.EQ.1) GO TO 136	RE100700
	LLT=1	RE100701
	CQ=1.04*CQ	RE100702
	GO TO 131	RE100703
135	IF (LLT.EQ.1) CQ=.98*CQ	RE100704
	IF (LLT.EQ.1) GO TO 136	RE100705
	LGT=1	RE100706
	CQ=.96*CQ	RE100707
	GO TO 131	RE100708
136	QD(I,J)=CQ*POX	RE100709
138	CONTINUE	RE100710
	WRITE(6,63) (R(J),J=1,JM)	RE100711
	DO 140 I=ID1,ID2	RE100712
	DO 140 J=1,JM	RE100713
140	XQD(I,J)=QD(I,J)*XXQ	RE100714
	DO 143 I=ID1,ID2	RE100715
	X1=Z(I)-Z(IPE)+DZ/2.	RE100716
	IF (JM.GT.9) GO TO 142	RE100717
	WRITE(6,105) X1, (XQD(I,J),J=1,JM)	RE100718
	GO TO 143	RE100719
142	WRITE(6,105) X1, (XQD(I,J),J=1,9)	RE100720
	WRITE(6,105) X1, (XQD(I,J),J=10,JM)	RE100721
143	CONTINUE	RE100722
	X2=(XQ-QD(IMAX,1))/QD(IMAX,1)	RE100723
	X3=X2*X2	RE100724
	IF (X3.LT..0001) GO TO 150	RE100725
	TSTEAM=TSTEAM+DTSTM	RE100726
	XQ=QD(IMAX,1)	RE100727
	GO TO 129	RE100728
150	IF (KTYPE.EQ.0) GO TO 174	RE100729
C ***	CALCULATE AND STORE (SINGLE PULSE EXPOSURE) TEMPERATURES FOR	RE100730
C ***	PLOTTING PROFILES	RE100731
	WRITE(6,139)	RE100732
	DO 170 L15=1,KTYPE	RE100733
	RGV=0.	RE100734
	DTIME=TIMEX(L15)	RE100735
	K=ALOG(DTIME*(XC-1.)/DTX+1.)/ALOG(XC)+1.	RE100736
	IF (K+1.GT.KT) GO TO 170	RE100737
	X1=(DTIME-XT(K))/(XT(K+1)-XT(K))	RE100738
	DO 166 I=II1,II2	RE100739
	DO 166 J=JJ1,JJ2	RE100740
	V(I,J)=VC(I,J,K)+X1*(VC(I,J,K+1)-VC(I,J,K))	RE100741

	IF (V (I, J) .GT. RGV) RGV=V (I, J)	RE100742
166	CONTINUE	RE100743
	SAVRGV (L15) =RGV	RE100744
	IF (KTYPE.EQ.1) GO TO 167	RE100745
	WRITE (7, 117) NRUN (1), NPULSE(1), REPET (1)	RE100746
	WRITE (7, 118) XDPULS, WAVEL, RIM	RE100747
	WRITE (7, 119) II1, II2, II3, JJ1, JJ2	RE100748
	WRITE (7, 119) N3, M3	RE100749
	WRITE (7, 120) (F (J), J=1, N3)	RE100750
	WRITE (7, 120) (Z (I), I=1, M3)	RE100751
	WRITE (7, 118) TIMEX (L15)	RE100752
167	WRITE (6, 141) TIMEX (L15)	RE100753
	WRITE (6, 63) (R (J), J=JJ1, JJ2)	RE100754
	JCNT=JJ2-JJ1+1	RE100755
	IF (JCNT.GT.9) GO TO 400	RE100756
	GO TO 401	RE100757
400	JJCNT=JCNT-9	RE100758
	JJJ2=JJ2-JJCNT	RE100759
	JJJ2P1=JJJ2+1	RE100760
401	DO 168 I=II1, II2	RE100761
	X1=Z (I) -Z (IPE) +DZ/2.	RE100762
	IF (JCNT.GT.9) GO TO 402	RE100763
	WRITE (6, 64) X1, (V (I, J), J=JJ1, JJ2)	RE100764
	GO TO 403	RE100765
402	WRITE (6, 64) X1, (V (I, J), J=JJ1, JJJ2)	RE100766
	WRITE (6, 64) X1, (V (I, J), J=JJJ2P1, JJ2)	RE100767
403	IF (KTYPE.EQ.1) GO TO 168	RE100768
	WRITE (7, 137) (V (I, J), J=JJ1, JJ2)	RE100769
168	CONTINUE	RE100770
170	CONTINUE	RE100771
	RGV=0.	RE100772
	DO 405 LL15=1, KTYPE	RE100773
	IF (SAVRGV (LL15) .GT. RGV) RGV=SAVRGV (LL15)	RE100774
405	CONTINUE	RE100775
	WRITE (7, 396)	RE100776
	DO 406 LL15=1, KTYPE	RE100777
406	WRITE (7, 137) RGV	RE100778
C ***	INTERPOLATE AXIAL EXTENT OF DAMAGE	RE100779
174	I5=0	RE100780
	I6=0	RE100781
	IF (ID1.EQ.ID2) GO TO 182	RE100782
	DO 175 I=ID1, ID2	RE100783
	L1=ID1+ID2-I	RE100784
	IF (QD (L1, 1) .GT. POX) I5=L1	RE100785
	IF (QD (L1, 1) .LT. POX) I6=L1	RE100786
	IF (QD (I, 1) .GT. POX) I7=I	RE100787
	IF (QD (I, 1) .LT. POX) I8=I	RE100788
175	CONTINUE	RE100789
	IF (IPRT(9) .EQ.0) GO TO 182	RE100790
	WRITE (6, 350)	RE100791
350	FORMAT (1H0, 22HAXIAL EXTENT OF DAMAGE)	RE100792
	IF (I5.EQ.0) WRITE (6, 176)	RE100793
176	FORMAT (1H0, 5X, 45HDEPTHS OF DAMAGE BEYOND BOTH SPECIFIED DEPTHS)	RE100794
	IF (I5.EQ.0) GO TO 182	RE100795
	IF (I6.EQ.0) GO TO 190	RE100796
	IF (I5.GE.I6) GO TO 178	RE100797
	X2=ALOG (QD (I6, 1) /QD (I5, 1)) / (Z (I6) -Z (I5))	RE100798

```

X1=QD (I5, 1)
X3=ALOG (POX/X1)/X2+Z (I5) -Z (IPE) +DZ/2.
WRITE (6, 177) X3
177 FORMAT (1H0, 5X, 24HMINIMUM DEPTH OF DAMAGE=, E10.3, 2HCM)
178 IF (I8.GE. I7) GO TO 182
X2=ALOG (QD (I8, 1)/QD (I7, 1))/(Z (I8) -Z (I7))
X1=QD (I7, 1)
X3=ALOG (POX/X1)/X2+Z (I7) -Z (IPE) +DZ/2.
180 WRITE (6, 181) X3
181 FORMAT (1H0, 5X, 24HMAXIMUM DEPTH OF DAMAGE=, E10.3, 2HCM)
C *** INTERPOLATE RADIAL EXTENT OF IRREVERSIBLE DAMAGE AT SPECIFIED
C *** DEPTHS
182 IF (IPRT (10).EQ.0) GO TO 192
WRITE (6, 360)
360 FORMAT (1H0, 23H RADIAL EXTENT OF DAMAGE)
DO 189 I=ID1, ID2
J1=0
X3=Z (I) -Z (IPE) +DZ/2.
DO 183 J=1, JM
IF (POX.GT. QD (I, J)) J1=J
183 CONTINUE
X20=0.
IF (J1.EQ.0) GO TO 187
IF (J1.EQ. JM) WRITE (6, 185) X3, R (JM)
185 FORMAT (1H0, 5X, 2HZ=, E9.3, 2HCM, 5X, 36H RADIAL EXTENT OF DAMAGE GREATER
1 TRAN, F10.3, 2HCM)
IF (J1.EQ. JM) GO TO 189
X2=ALOG (QD (I, J1+1)/QD (I, J1))/(R (J1+1) -R (J1))
X1=QD (I, J1)
X20=ALOG (POX/X1)/X2+R (J1)
187 WRITE (6, 188) X3, X20
188 FORMAT (1H0, 5X, 2HZ=, E9.3, 2HCM, 5X, 37H RADIAL EXTENT OF IRREVERSIBLE D
1 AMAGE=, E10.3, 2HCM)
189 CONTINUE
IF (LPX.EQ.0) GO TO 300
GO TO 145
190 WRITE (6, 191)
191 FORMAT (1H0, 5X, 31HNO DAMAGE---LASER POWER TOO LOW)
192 IF (LPX.EQ.0) GO TO 300
GO TO 145
200 STOP
END
SUBROUTINE GRID
C *** GRID COMPUTES THE COEFFICIENTS IN PARTIAL DIFFERENTIAL EQUATIONS ARE
C *** RADIAL AND AXIAL COORDINATES, R AND Z, AND ASSIGNS CONDUCTIVITY ANRE
C *** VOLUMETRIC SPECIFIC HEAT TO GRID
C *** CALCULATE B (CM** -2) AND F (CM)
COMMON A (29, 3), AP, AAV, ACH, APE, ASC, ATS, AVL, B (14, 3), BB, BV (14, 3),
1CONX (6), CON (29), CUT, DFLOW (6), DPULSE, DR, DT, DTX, DZ, FL, HR (14),
2IAB (29, 14), IBLOOD (10), IFIL, IG, IGX, IHT, IPA, IPC, IPE, IPROF, IPS, IPT,
3IPV, IV (29), JVL, LIM, LPA, LPC, LPE, LPS, LPV, LPX, LTHAX, K, KM, KT, M, M1, M2,
4M3, N, N1, N3, N4, NVL, POX, PR (14), PTIME, QP, R (14), RCO, FIM, RN, RPE, RRT,
5RVL, RSC, S (29, 14), SHB, TAV, TCH, TOM, TPE, TVL, TS (2200), TSC, TTS, V (29, 14)
6, VC (29, 14, 120), VSH (29), VSHX (6), WAVEL, XC, XFLOW, XFLOWI (6), XFLOWO (6),
7XPD (120), XT (120), Z (29), ZD (8), ZM, FLOWX (14), FLOWY (14), PUPIL, SIGMA,
8IPRT (10), APE1, APE2, RINT, ZO, FLO, CABER, CABER2, PP, PC, NB, NC, FC
DIMENSION IX (7), LX (7)

```

C *** CALCULATE B(CM**-2) AND R (CM)	RE100856
WRITE(6,170)	RE100857
170 FORMAT(1H1)	RE100858
R(1)=0.	RE100859
CK=N-N1	RE100860
CP=RVL/DR-N1+1.	RE100861
X1=2.	RE100862
180 R2=EXP(ALOG(2.*(CP*(X1-1.)+1.)/(X1+1.))/(CK-1.))	RE100863
IF(R2/X1.GT..99999.AND.R2/X1.LT.1.00001)GO TO 181	RE100864
X1=R2	RE100865
GO TO 180	RE100866
181 IF(IPRT(1).EQ.0)GO TO 220	RE100867
WRITE(6,182)	RE100868
182 FORMAT(1H0,16HGRID INFORMATION)	RE100869
WRITE(6,184)R2	RE100870
184 FORMAT(1H0,5X,3HR2=,F8.4)	RE100871
220 RN=DR*(N1-1.+(R2**(CK+1.)-1.)/(R2-1.))	RE100872
C *** CALCULATE RADIAL SPACE STEPS R(J)	RE100873
DO 185 J=2,N4	RE100874
185 R(J)=DR*(J-1)	RE100875
X1=R2*DR	RE100876
DO 186 J=N4,N	RE100877
R(J+1)=R(J)+X1	RE100878
186 X1=R2*X1	RE100879
C *** CALCULATE COEFFICIENTS B OF FINITE DIFFERENCE EQNS.	RE100880
X1=2./(DR*DR)	RE100881
DO 187 J=2,N4	RE100882
B(J,1)=.25*(2*J-3)*X1/(J-1)	RE100883
B(J,2)=X1	RE100884
187 B(J,3)=X1-B(J,1)	RE100885
X2=DR	RE100886
X1=R2*DR	RE100887
DO 188 J=N4,N	RE100888
B(J,2)=2./(X1*X2)	RE100889
B(J,1)=(2./X2-1./R(J))/(X1+X2)	RE100890
B(J,3)=B(J,2)-B(J,1)	RE100891
X2=R2*X2	RE100892
188 X1=R2*X1	RE100893
B(1,1)=0.	RE100894
B(1,2)=2./(DR*DR)	RE100895
B(1,3)=B(1,2)	RE100896
DO 189 J=1,N	RE100897
IF(R(J).LT.RVL)JVL=J	RE100898
189 CONTINUE	RE100899
C *** CALCULATE AXIAL SPACE STEPS Z(I)	RE100900
CK=M2-M1+1	RE100901
X1=2.	RE100902
190 CP=2.*TAV/DZ+1.-(X1*(CK-1.)-1.)/(X1-1.)	RE100903
R1=EXP(ALOG(CP*X1-CP+1.)/CK)	RE100904
IF(R1/X1.GT..99999.AND.R1/X1.LT.1.00001)GO TO 192	RE100905
X1=R1	RE100906
GO TO 190	RE100907
192 ZM=((R1**CK-1.)/(R1-1.)*M1-1.)*DZ	RE100908
IF(IPRT(1).EQ.0)GO TO 230	RE100909
WRITE(6,194)R1,ZM	RE100910
194 FORMAT(1H ,5X,3HR1=,F8.4,2X,3HZM=,F8.4)	RE100911
230 X1=DZ	RE100912

X2=X1	RE100913
DO 195 I=2,M2	RE100914
Z(M2+I)=ZM+X2	RE100915
Z(M2+2-I)=ZM-X2	RE100916
IF(I.GT.M1)X1=F1*X1	RE100917
195 X2=X2+X1	RE100918
Z(1)=0.	RE100919
Z(M2+1)=ZM	RE100920
Z(M+1)=2.*ZM	RE100921
X1=Z(IPM)-DZ/2.-ZD(2)	RE100922
DO 196 I=1,M3	RE100923
196 Z(I)=Z(I)-X1	RE100924
L3=IPA	RE100925
DO 200 L=1,7	RE100926
L1=0	RE100927
DO 197 I=IPA,M3	RE100928
IF(Z(I).LT.ZD(L+1))L3=I	RE100929
IF(Z(I).LT.ZD(L).OR.Z(I).GE.ZD(L+1))GO TO 197	RE100930
L2=I	RE100931
L1=L1+1	RE100932
197 CONTINUE	RE100933
IF(L1.EQ.0)IX(L)=L3	RE100934
IF(L1.EQ.0)LX(L)=L3	RE100935
IF(L1.GT.0)IX(L)=L2+1-L1	RE100936
IF(L1.GT.0)LX(L)=L2	RE100937
200 CONTINUE	RE100938
IPV=IX(4)	RE100939
IPC=IX(5)	RE100940
IPS=IX(6)	RE100941
IPT=IX(7)	RE100942
LPA=LX(1)	RE100943
LPE=LX(3)	RE100944
LPV=LX(4)	RE100945
LPC=LX(5)	RE100946
LPS=LX(6)	RE100947
LPT=M3	RE100948
C *** SET CONDUCTIVITY CON AND HEAT CAPACITY VSH FOR VARIOUS EYE MEDIA	RE100949
DO 203 I=1,LPA	RE100950
CON(I)=CONX(1)	RE100951
203 VSH(I)=VSHX(1)	RE100952
DO 204 I=IPE,LPE	RE100953
CON(I)=CONX(2)	RE100954
204 VSH(I)=VSHX(2)	RE100955
DO 205 I=IPV,LPV	RE100956
CON(I)=CONX(3)	RE100957
205 VSH(I)=VSHX(3)	RE100958
DO 206 I=IPC,LPC	RE100959
CON(I)=CONX(4)	RE100960
206 VSH(I)=VSHX(4)	RE100961
DO 207 I=IPS,LPS	RE100962
CON(I)=CONX(5)	RE100963
207 VSH(I)=VSHX(5)	RE100964
DO 208 I=IPT,M3	RE100965
CON(I)=CONX(6)	RE100966
208 VSH(I)=VSHX(6)	RE100967
C *** CALCULATE COEFFICIENTS A OF FINITE DIFFERENCE EQNS.	RE100968
DO 210 I=IPA,M	RE100969

```

X1=Z(I+1)-Z(I-1)
X2=(CON(I-1)-CON(I+1))/(X1*X1)
X3=2.*CON(I)/X1
A(I,1)=X2+X3/(Z(I)-Z(I-1))
IF(I.EQ.IPA)A(I,1)=0.
A(I,3)=-X2+X3/(Z(I+1)-Z(I))
210 A(I,2)=A(I,1)+A(I,3)
RETURN
END
SUBROUTINE IMAGE
C *** IMAGE COMPUTES THE RETINAL IRRADIANCE PROFILE
COMMON A(29,3),AP,AAV,ACH,APE,ASC,ATS,AVL,B(14,3),BB,BV(14,3),
1CONX(6),CON(29),CUT,DFLOW(6),DPULSE,DE,DT,DTX,DZ,FL,HR(14),
2IAB(29,14),IBLOOD(10),IFIL,IG,IGX,IHT,IPA,IPC,IPE,IPOF,IPS,IPT,
3IPV,IV(29),JVL,LIM,LPA,LPC,LPE,LPS,LPV,LPX,LTMAX,K,KM,KT,M,M1,M2,
4M3,N,N1,N3,N4,NVL,POX,PP(14),PTIME,QP,R(14),PCO,RIM,RN,RPE,RRT,
5RVL,RSC,S(29,14),SHB,TAV,TCH,TOM,TPE,TVL,TS(2200),TSC,TTS,V(29,14)
6,VC(29,14,120),VSH(29),VSHX(6),WAVEL,XC,XFLOW,XFLOWI(6),XFLOWO(6),
7XPD(120),XT(120),Z(24),ZD(8),ZM,FLOWI(14),FLOWX(14),PUPIL,SIGMA,
8IPRT(10),APE1,APE2,RINT,ZO,FLO,CABER,CABEP2,PP,PC,NB,NC,FC
DIMENSION FA(2001),FP(2001),FX(2001),FY(2001),JO(32),NA(22),PX(30)
1,RX(30),XF1(2001),XF2(2001)
REAL JO,NA,NB,NC
DO 200 J=1,N
200 PR(J)=0.
LI=500
LII=LI
DO 201 L=1,LI
201 FX(L)=0.
READ(5,202)PUPIL
202 FORMAT(10E8.3)
RINT=PUPIL/(LI-1)
IF(IPROP.EQ.1)GO TO 214
IF(IPROP.EQ.0)GO TO 219
C *** INTERPOLATE IRREGULAR LASER PROFILE (SYMMETRIC IN R) AT INTERVALS
C *** OF RINT STARTING AT P=0
READ(5,205)LF
205 FORMAT(I7)
READ(5,206)(FX(L),L=1,LR)
206 FORMAT(10E7.3)
READ(5,206)(PX(L),L=1,LF)
X1=PX(1)
DO 207 L=1,LR
207 PX(L)=PX(L)/X1
X5=0.
X6=0.
DO 208 L=2,LR
X2=(PX(L)-PX(L-1))/(FX(L)-FX(L-1))
X1=PX(L-1)-X2*FX(L-1)
X3=X1*(RX(L)*FX(L)-RX(L-1)*FX(L-1))/2.
X4=X2*(RX(L)*FX(L)-RX(L-1)*FX(L-1))/3.
IF(RX(L).GT.PUPIL)X6=X6+6.2832*(X3+X4)
208 X5=X5+6.2832*(X3+X4)
QP=POX*.23906*(1.-RCO)/X5
XX=(X5-X6)/X5
IF(RX(LR).LT.PUPIL)LII=RX(LR)/RINT+1
L2=2

```

	X1=0.	RE101027
	DO 213 L=1,LII	RE101028
210	IF (RX (L2) .GT. X1) GO TO 212	RE101029
	L2=L2+1	RE101030
	IF (L2.LE. LR) GO TO 210	RE101031
	GO TO 213	RE101032
212	X2=(X1-RX (L2-1)) / (RX (L2) -RX (L2-1))	RE101033
	FX (L) =PX (L2-1) +X2*(PX (L2) -PX (L2-1))	RE101034
213	X1=X1+RINT	RE101035
	GO TO 223	RE101036
C ***	CALCULATE GAUSSIAN LASER PROFILE AT INTERVALS OF RINT STARTING AT	RE101037
214	SIGMA=RIM*SQRT (-2./ALOG (CUT))	RE101038
	QP=2.*POX*.23906*(1.-RCO)/(3.1416*SIGMA*SIGMA)	RE101039
	XX=1.-EXP (-2.*PUPIL*PUPIL/(SIGMA*SIGMA))	RE101040
	IF (IFIL.EQ.1) GO TO 217	RE101041
	DO 216 J=1,N	RE101042
	X3=2.*R (J) *R (J) / (SIGMA*SIGMA)	RE101043
	IF (X3.GT.80.) GO TO 216	RE101044
	PR (J) =EXP (-X3)	RE101045
216	CONTINUE	RE101046
	GO TO 276	RE101047
217	X1=0.	RE101048
	DO 218 L=1,LII	RE101049
	X3=2.*X1*X1/(SIGMA*SIGMA)	RE101050
	FX (L) =0.	RE101051
	IF (X3.GT.80.) GO TO 218	RE101052
	FX (L) =EXP (-X3)	RE101053
218	X1=X1+RINT	RE101054
	GO TO 227	RE101055
C ***	SPECIFY UNIFORM LASER PROFILE FROM R (1) TO R (LIM)	RE101056
219	QP=POX*.23906*(1.-PCO)/(3.1416*RIM*RIM)	RE101057
	XX=1.	RE101058
	IF (RIM.GT.PUPIL) XX=PUPIL*PUPIL/(RIM*RIM)	RE101059
	IF (IFIL.EQ.1) GO TO 221	RE101060
	DO 220 J=1,LIM	RE101061
220	PR (J) =1.	RE101062
	GO TO 276	RE101063
221	L1=RIM/RINT	RE101064
	RINT=RIM/L1	RE101065
	LII=RIM/RINT+1	RE101066
	DO 222 L=1,LII	RE101067
222	FX (L) =1.	RE101068
	GO TO 227	RE101069
C ***	CALCULATE TOTAL AREA FA (L) AND PORTION OF LASERS POWER BETWEEN R=0	RE101070
C ***	AND (L-.5)*RINT	RE101071
223	IF (IFIL.EQ.1) GO TO 227	RE101072
	FP (1) =3.1416*FX (1) *RINT*RINT/4.	RE101073
	FA (1) =3.1416*RINT*RINT/4.	RE101074
	DO 224 L=2,LII	RE101075
	X1=(L-.5)*RINT	RE101076
	X2=(L-1.5)*RINT	RE101077
	FP (L) =FP (L-1) +FX (L) *3.1416*(X1*X1-X2*X2)	RE101078
224	FA (L) =FA (L-1) +3.1416*(X1*X1-X2*X2)	RE101079
C ***	CALCULATE PROPILE PF (J)	RE101080
	X1=0.	RE101081
	X2=0.	RE101082
	DO 225 J=1,N	RE101083

	X3=(R(J)+R(J+1))/(2.*RINT)+.5000001	RE101084
	IF(X3.LT.1.)X3=1.000001	RE101085
	L2=X3	RE101086
	IF(L2.GE.LII)GO TO 225	RE101087
	X4=X3-L2	RE101088
	X5=FP(L2)+X4*(FP(L2+1)-FP(L2))	RE101089
	X6=FA(L2)+X4*(FA(L2+1)-FA(L2))	RE101090
	PF(J)=(X5-X1)/(X6-X2)	RE101091
	X1=X5	RE101092
	X2=X6	RE101093
225	CONTINUE	RE101094
	GO TO 276	RE101095
C ***	SPREAD FUNCTION CALCULATIONS	RE101096
227	READ(5,202)ZO,FLO,FC,NB,CABER,PP,PC	RE101097
	CABER2=CABER/WAVEL	RE101098
	READ(5,228)(JO(L),L=1,32)	RE101099
228	FORMAT(10F8.5)	RE101100
	RFAD(5,228)(NA(L),L=1,22)	RE101101
	X1=(WAVEL-350.)/50.+1.	RE101102
	L1=X1	RE101103
	X2=X1-L1	RE101104
	NC=NA(L1)+X2*(NA(L1+1)-NA(L1))	RE101105
	X1=(NB-1.)*NC/(NB*(NC-1.))	RE101106
	FL=FLO*X1	RE101107
	X2=ZO/FLO	RE101108
	X0=NC*ZO*X1/(NC*X2-X1)-FLO	RE101109
	X3=1.-PC*(NC*ZO-FC)/(NC*ZO*FC)	RE101110
	DO 230 L=1,L1	RE101111
	IF(L.GT.LII)GO TO 230	RE101112
	X1=(L-1)/X3+1.000001	RE101113
	L1=X1	RE101114
	X2=X1-L1	RE101115
	IF(L1+1.GT.LI)FY(L)=0.	RE101116
	IF(L1+1.GT.LI)LII=L	RE101117
	IF(L1+1.GT.LI)GO TO 230	RE101118
	FY(L)=(FX(L1)+X2*(FX(L1+1)-FX(L1)))/(X3*X3)	RE101119
230	CONTINUE	RE101120
	DO 231 L=1,LII	RE101121
231	FX(L)=FY(L)	RE101122
	X5=ATAN(PUPIL/(FLO-PP+X0))	RE101123
	X6=1.-COS(X5)	RE101124
	X7=SIN(X5)*SIN(X5)	RE101125
	FF=FLO-PP	RE101126
	DO 234 L=1,LII	RE101127
	X4=(L-1)*RINT	RE101128
	X1=6.2832*NC*(-FF-X6*X0+SQRT(FF*FF-X7*X0*X0))*X4*X4/(WAVEL*1.E-7*	RE101129
	1PUPIL*PUPIL)	RE101130
	X2=CABER2*X4*X4*X4*X4	RE101131
	XF1(L)=SQRT(FX(L))*COS(X1+X2)	RE101132
234	XF2(L)=SQRT(FX(L))*SIN(X1+X2)	RE101133
	DO 260 J=1,N	RE101134
	X1=6.2832*R(J)/(WAVEL*1.E-7*FF)	RE101135
	X2=0.	RE101136
	X3=0.	RE101137
	DO 255 L=1,LII	RE101138
	X4=X1*(L-1)*RINT	RE101139
	IF(L.EQ.1)X4=X1*.25*RINT	RE101140

```

IF (X4.GT.3.) GO TO 250
X5=X4/.1+1.000001
L1=X5
X5=X5-L1
X7=JO (L1) +X5* (JO (L1+1) -JO (L1))
GO TO 251
250 X6=3./X4
X8=.79788456-.00000077*X6-.00552740*X6*X6-.00009512*X6*X6*X6+
1.00137237*X6*X6*X6*X6-.00072805*X6*X6*X6*X6+.00014476*X6*X6*X6*X6
2X6*X6*X6
X9=X4-.78539816-.04166397*X6-.00003954*X6*X6+.00262573*X6*X6*X6-
1.00054125*X6*X6*X6*X6-.00029333*X6*X6*X6*X6+.00013558*X6*X6*X6*X6
2X6*X6*X6
X7=X8*COS (X9)/SQRT (X4)
251 IF (L.GT.1) GO TO 252
X2=X2+X7*.25* (3.*XF1 (1) +XF1 (2)) *.25*RINT*.5*RINT
X3=X3+X7*.25* (3.*XF2 (1) +XF2 (2)) *.25*RINT*.5*RINT
GO TO 255
252 X2=X2+X7*XF1 (L) * (L-1) *RINT*RINT
X3=X3+X7*XF2 (L) * (L-1) *RINT*RINT
255 CONTINUE
260 HF (J) =X2*X2+X3*X3
X1=HF (1)
DO 270 J=1,N
270 HF (J) =HF (J) /X1
X1=.0002
X2=3.1416*X1*X1/4
J=2
X4=HF (1) *X2
L1=2
271 IF (X1.LT.R (J) +.0000001) GO TO 272
J=J+1
GO TO 271
272 X5= (X1-R (J-1)) / (R (J) -R (J-1))
X6=HF (J-1) +X5* (HF (J) -HF (J-1))
X7=8.* (L1-1) *X2
X4=X4+X6*X7
L1=L1+1
X1=X1+.0002
IF (X1.LE..1) GO TO 271
QP=.23906*XX*POX* (1.-RCO) /X4
RETURN
276 DO 280 J=1,N
280 HF (J) =PF (J)
RETURN
END
SUBROUTINE HTXDEF
C *** HTXDEF COMPUTES RATE OF HEAT DEPOSITON AT VARIOUS POINTS I,J
COMMON A (29,3) ,AP,AAV,ACH,APE,ASC,ATS,AVL,B (14,3) ,BB,BV (14,3) ,
1CONX (6) ,CON (29) ,CUT,DFLOW (6) ,DPULSE,DP,DT,DTX,DZ,FL,HR (14) ,
2IAB (29,14) ,IBLOOD (10) ,IFIL,IG,IGX,IHT,IPA,IPC,IPE,IPOF,IPS,IPT,
3IPV,IV (29) ,JVL,LIM,LPA,LPC,LPE,LPS,LPV,LPX,LTMAX,K,KM,KT,M,M1,M2,
4M3,N,N1,N3,N4,NVL,POX,PR (14) ,PTIME,QP,R (14) ,RCO,RIM,RN,RPE,RRT,
5RVL,RSC,S (29,14) ,SHB,TAV,TCH,TOM,TPE,TVL,TS (2200) ,TSC,TTS,V (29,14)
6,VC (29,14,120) ,VSH (29) ,VSHX (6) ,WAVEL,XC,XFLOW,XFLOWI (6) ,XFLOWO (6) ,
7XPD (120) ,XT (120) ,Z (29) ,ZD (8) ,ZM,FLOWI (14) ,FLOWX (14) ,PUPIL,SIGMA,
8IPPT (10) ,APE1,APE2,RINT,ZO,FLO,CABER,CABER2,PP,PC,NB,NC,FC
RE101141
RE101142
RE101143
RE101144
RE101145
RE101146
RE101147
RE101148
RE101149
RE101150
RE101151
RE101152
RE101153
RE101154
RE101155
RE101156
RE101157
RE101158
RE101159
RE101160
RE101161
RE101162
RE101163
RE101164
RE101165
RE101166
RE101167
RE101168
RE101169
RE101170
RE101171
RE101172
RE101173
RE101174
RE101175
RE101176
RE101177
RE101178
RE101179
RE101180
RE101181
RE101182
RE101183
RE101184
RE101185
RE101186
RE101187
RE101188
RE101189
RE101190
RE101191
RE101192
RE101193
RE101194
RE101195
RE101196
RE101197

```

	DIMENSION AB(29,3), ABR(29,7), ABS(7), II(29), IZ(29), REF(8), REFL(8),	RE101198
	IZH(29)	RE101199
	IF(IHT.EQ.0) RETURN	RE101200
	IF(QP.LT.1.E-25) GO TO 340	RE101201
	IF(IHT.EQ.1) RETURN	RE101202
	LZ=7	RE101203
	LZ0=LZ-1	RE101204
	LZ1=LZ+1	RE101205
	DO 280 I=1,M	RE101206
	II(I)=0	RE101207
	IZ(I)=0	RE101208
	ZH(I)=(Z(I)+Z(I+1))/2.	RE101209
	DO 279 L1=1,3	RE101210
279	AB(I,L1)=0.	RE101211
	DO 280 L1=1,LZ	RE101212
280	ABR(I,L1)=0.	RE101213
	DO 282 L1=1,LZ	RE101214
	REF(L1)=0.	RE101215
282	REFL(L1)=0.	RE101216
	REF(2)=RPT	RE101217
	REF(6)=RSC	RE101218
	REF(LZ1)=0.	RE101219
	IF(IPRT(1).EQ.0) GO TO 350	RE101220
	WRITE(6,283)(ZH(I),I=1,M)	RE101221
283	FORMAT(1H0,5X,3HZH=/(1H,5X,10E10.3))	RE101222
C ***	EVALUATE ABSORPTION CONSTANTS APE1 AND APE2 FOR FRONT AND REAR OF	RE101223
C ***	PE AS WELL AS IG INDICATING I INDEX WHERE GRANULES ARE LOCATED	RE101224
350	IF(IGX.EQ.1) GO TO 284	RE101225
	APE1=(APE-ACH*(1.-RPE))/RPE	RE101226
	APE2=ACH	RE101227
	AP=(EXP(-ACH*RPE*TPE)-EXP(-APE1*RPE*TPE))/(1.-EXP(-APE1*RPE*TPE))	RE101228
	IG=IPE	RE101229
	GO TO 285	RE101230
284	APE1=ACH	RE101231
	APE2=(APE-ACH*RPE)/(1.-RPE)	RE101232
	AP=(EXP(-ACH*(1.-RPE)*TPE)-EXP(-APE2*(1.-RPE)*TPE))/(1.-EXP(-APE2*	RE101233
	1(1.-RPE)*TPE))	RE101234
	IG=LPE-(1.001-RPE)*(LPE-IPE+1)+.5	RE101235
285	ABS(1)=AAV	RE101236
	ABS(2)=APE1	RE101237
	ABS(3)=APE2	RE101238
	ABS(4)=AVL	RE101239
	ABS(5)=ACH	RE101240
	ABS(6)=ASC	RE101241
	ABS(7)=ATS	RE101242
	L1=2	RE101243
	DO 306 I=IPA,M	RE101244
295	IF(ZH(I-1).LT.ZD(L1)) GO TO 296	RE101245
	L1=L1+1	RE101246
	GO TO 295	RE101247
296	IF(ZH(I).GE.ZD(L1)) GO TO 299	RE101248
C ***	NO ZD BETWEEN ZH(I-1) AND ZH(I)	RE101249
	AB(I,1)=ABS(L1-1)*(ZH(I)-ZH(I-1))	RE101250
	II(I)=1	RE101251
	IZ(I)=L1	RE101252
	IF(L1.GT.LZ) GO TO 306	RE101253
	DO 297 L2=L1,LZ	RE101254

297	ABR(I,L2)=AB(I,1)	RE101255
	GO TO 306	RE101256
299	IF(ZH(I).GE.ZD(L1+1))GO TO 303	RE101257
C ***	ONLY ZD(L1) BETWEEN ZH(I-1) AND ZH(I)	RE101258
	AB(I,1)=ABS(L1-1)*(ZD(L1)-ZH(I-1))	RE101259
	AB(I,2)=ABS(L1)*(ZH(I)-ZD(L1))	RE101260
	ABR(I,L1)=AB(I,1)	RE101261
	II(I)=2	RE101262
	IZ(I)=L1	RE101263
	L3=L1+1	RE101264
	IF(L3.GT.LZ)GO TO 306	RE101265
	DO 300 L2=L3,LZ	RE101266
300	ABR(I,L2)=AB(I,1)+AB(I,2)	RE101267
	GO TO 306	RE101268
C ***	ZD(L1) AND ZD(L1+1) BETWEEN ZH(I-1) AND ZH(I)	RE101269
303	AB(I,1)=ABS(L1-1)*(ZD(L1)-ZH(I-1))	RE101270
	AB(I,2)=ABS(L1)*(ZD(L1+1)-ZD(L1))	RE101271
	AB(I,3)=ABS(L1+1)*(ZH(I)-ZD(L1+1))	RE101272
	ABR(I,L1)=AB(I,1)	RE101273
	ABR(I,L1+1)=AB(I,1)+AB(I,2)	RE101274
	II(I)=3	RE101275
	IZ(I)=L1	RE101276
	L3=L1+2	RE101277
	IF(L3.GT.LZ)GO TO 306	RE101278
	DO 304 L2=L3,LZ	RE101279
304	ABR(I,L2)=AB(I,1)+AB(I,2)+AB(I,3)	RE101280
306	CONTINUE	RE101281
	DO 314 I=IPA,M	RE101282
	IF(AB(I,1).GT.10.)AB(I,1)=10.	RE101283
	IF(AB(I,2).GT.10.)AB(I,2)=10.	RE101284
	IF(AB(I,3).GT.10.)AB(I,3)=10.	RE101285
	DO 314 L=2,LZ	RE101286
	IF(ABR(I,L).GT.10.)ABR(I,L)=10.	RE101287
314	CONTINUE	RE101288
C ***	DEPOSITION BY INCOMING BEAM	RE101289
	X2=QP	RE101290
	L1=2	RE101291
	DO 317 I=IPA,M	RE101292
	L2=II(I)	RE101293
	X3=X2	RE101294
	X2=X2*EXP(-AB(I,1))	RE101295
	X4=0.	RE101296
	IF(L2.EQ.1)GO TO 315	RE101297
	L3=IZ(I)	RE101298
	X4=X2*REF(L3)	RE101299
	X2=X2*(1.-REF(L3))*EXP(-AB(I,2))	RE101300
	IF(L2.EQ.2)GO TO 315	RE101301
	X4=X4+X2*REF(L3+1)	RE101302
	X2=X2*(1.-REF(L3+1))*EXP(-AB(I,3))	RE101303
315	IF(X2.LT.1.E-10)X2=0.	RE101304
	DO 317 J=1,JVL	RE101305
	S(I,J)=(X3-X2-X4)*HP(J)/(ZH(I)-ZH(I-1))	RE101306
	IF(S(I,J).LT.1.E-10/DPULSE)S(I,J)=0.	RE101307
317	CONTINUE	RE101308
C ***	CALCULATION OF REFLECTED INTENSITIES BY VARIOUS INTERFACES	RE101309
C ***	STARTING WITH FIRST INTERNAL INTERFACE	RE101310
	X2=QP	RE101311

```

DO 322 L1=1,LZ0
X3=ABS(L1)* (ZD(L1+1)-ZD(L1))
IF(X3.GT.10.) X3=10.
X2=X2*EXP(-X3)
REFL(L1+1)=X2*REF(L1+1)
322 X2=X2*(1.-REF(L1+1))
DO 327 L1=2,LZ
I=IPA
324 IF(ZH(I).GT.ZD(L1))GO TO 325
I=I+1
IF(I.LE.M)GO TO 324
GO TO 327
325 X2=REFL(L1)
DO 326 L3=IPA,I
X3=X2
L4=I+IPA-L3
X2=X2*EXP(-ABR(L4,L1))
DO 326 J=1,JVL
S(L4,J)=S(L4,J)+(X3-X2)*HR(J)/(ZH(L4)-ZH(L4-1))
IF(S(L4,J).LT.1.E-10/DPULSE)S(L4,J)=0.
326 CONTINUE
327 CONTINUE
IHT=1
RETURN
C *** NO HEAT DEPOSITION, BEAM OFF
340 DO 342 I=1,M3
DO 342 J=1,N3
342 S(I,J)=0.
IHT=0
RETURN
END
SUBROUTINE MXGRAN
C *** THIS ROUTINE COMPUTES CONSEQUENCE OF GRANULAR ABSORPTION ON
C *** TEMPERATURE VARIATIONS IN PE. (USED ONLY ONCE.)
COMMON A(29,3),AP,AAV,ACH,APE,ASC,ATS,AVL,B(14,3),BB,BV(14,3),
1CONX(6),CON(29),CUT,DFLOW(6),DPULSE,DR,DT,DTX,DZ,FL,HR(14),
2IAB(29,14),IBLOOD(10),IFIL,IG,IGX,IHT,IPA,IPC,IPE,I PROF,IPS,IPT,
3IPV,IV(29),JVL,LIM,LPA,LPC,LPE,LPS,LPV,LPX,LTMAX,K,KM,KT,M,M1,M2,
4M3,N,N1,N3,N4,NVL,POX,PR(14),PTIME,QP,R(14),RCO,RIM,RN,RPE,RRT,
5RVL,RSC,S(29,14),SHB,TAV,TCH,TOM,TPE,TVL,TS(2200),TSC,TTS,V(29,14)
6,VC(29,14,120),VSH(29),VSHX(6),WAVEL,XC,XFLOW,XFLOWI(6),XFLOWO(6),
7XPD(120),XT(120),Z(29),ZD(8),ZM,FLOWI(14),FLOWX(14),PUPIL,SIGMA,
8IPRT(10),APE1,APE2,RINT,ZO,FLO,CABER,CABER2,PP,PC,NB,NC,FC
L5=1
BT=.3E-8
IF(IPRT(7).EQ.0)GO TO 407
WRITE(6,403)
403 FOFMAT(1H0,48HNORMALIZED TEMPERATURE RISES OF MELANIN GRANULES)
WRITE(6,405)LTMAX,BT
405 FOFMAT(1H0,5X,6H1TMAX=,I4,2X,3HBT=,E8.3)
407 IF(DPULSE.GT.1.0E-5)GO TO 494
LPT=DPULSE/.3E-8
L7=LTMAX-10
DO 410 L=1,L7,10
L1=L+1
L2=L+9
X1=TS(L)

```


X2=TS (L+10) -X1	RE101369
X3=0.	RE101370
DO 410 L3=L1,L2	RE101371
X3=X3+.1	RE101372
410 TS (L3)=X1+X3*X2	RE101373
LTT=2	RE101374
XPD (1)=1.0	RE101375
IF (LPT.GE.LTMAX) GO TO 472	RE101376
C *** CASE FOR LPT LFSS THAN LTMAX-----	RE101377
440 IF (XT (LTT) .GE. .3E-8) GO TO 442	RE101378
XPD (LTT)=TS (1)	RE101379
LTT=LTT+1	RE101380
GO TO 440	RE101381
442 TIMEX=XT (LTT)	RE101382
XX=TIMEX/BT+.000001	RE101383
LX=XX	RE101384
PT=LX	RE101385
IF (LX.GT.LPT) PT=LPT	RE101386
PO=0.	RE101387
IF (LX.GT.LTMAX) PO=LX-LTMAX	RE101388
IF (LX.GT.LPT) GO TO 443	RE101389
L1=1	RE101390
L2=LX	RE101391
GO TO 446	RE101392
443 IF (LX.GT.LTMAX) GO TO 444	RE101393
L1=LX+1-LPT	RE101394
L2=LX	RE101395
GO TO 446	RE101396
444 IF (LX.LT.LTMAX+LPT) GO TO 445	RE101397
L5=LTT	RE101398
GO TO 494	RE101399
445 L1=LX-LPT+1	RE101400
L2=LPTMAX	RE101401
446 X2=PO	RE101402
DO 448 L3=L1,L2	RE101403
448 X2=X2+TS (L3)	RE101404
XPD (LTT)=X2/PT	RE101405
LTT=LTT+1	RE101406
IF (LTT.LE.KT) GO TO 442	RE101407
GO TO 496	RE101408
C *** CASE FOR LTMAX LESS THAN LPT-----	RE101409
472 TIMEX=XT (LTT)	RE101410
XX=TIMEX/BT+.000001	RE101411
LX=XX	RE101412
PT=LX	RE101413
IF (LX.GT.LPT) PT=LPT	RE101414
PO=0.	RE101415
IF (LX.GT.LTMAX) PO=LX-LTMAX	RE101416
IF (LX.GT.LPT) GO TO 473	RE101417
L1=1	RE101418
L2=LX	RE101419
IF (LX.GT.LTMAX) L2=LPTMAX	RE101420
GO TO 475	RE101421
473 IF (LX.LT.LTMAX+LPT) GO TO 474	RE101422
L5=LTT	RE101423
GO TO 494	RE101424
474 L1=LX-LPT+1	RE101425

L2=LTMAX	RE101426
475 X2=PO	RE101427
DO 476 L3=L1,L2	RE101428
476 X2=X2+TS(L3)	RE101429
XPD(LTT)=X2/PT	RE101430
LTT=LTT+1	RE101431
IF(LTT.LE.KT)GO TO 472	RE101432
GO TO 496	RE101433
C *** END CALCULATION IF TEMPERATURES VERY UNIFORM	RE101434
494 IF(L5.GT.KT)GO TO 496	RE101435
DO 495 L1=L5,KT	RE101436
495 XPD(L1)=1.	RE101437
496 IF(IPRT(7).FQ.0)RETURN	RE101438
WRITE(6,497)(XPD(L1),L1=1,KT)	RE101439
497 FORMAT(1H0,5X,4HXPD=/(1H,5X,10F8.2))	RE101440
RETURN	RE101441
END	RE101442
SUBFOUNTINE BLOOD	RE101443
C SUBFOUNTINE BLOOD COMPUTES CHANGES IN MATRIX ELEMENTS A AND B DUE	RE101444
C TO BLOOD FLOW	RE101445
COMMON A(29,3),AF,AAV,ACH,APE,ASC,ATS,AVL,B(14,3),BB,BV(14,3),	RE101446
1CCNX(6),CON(29),CUT,DFLOW(6),DPULSE,DR,DT,DTX,DZ,FL,HR(14),	RE101447
2IAB(29,14),IBLOOD(10),IFIL,IG,IGX,IHT,IPA,IPC,IPE,IPROP,IPS,IPT,	RE101448
3IPV,IV(29),JVL,LIM,LPA,LPC,LPE,LPS,LPV,LPX,LTMAX,K,KM,KT,M,M1,M2,	RE101449
4M3,N,N1,N3,N4,NVL,POX,PR(14),PTIME,QP,R(14),RCO,FIM,RN,RPE,RRT,	RE101450
5RVL,PSC,S(29,14),SHB,TAV,TCH,TOM,TPE,TVL,TS(2200),TSC,TTS,V(29,14)	RE101451
6,VC(29,14,120),VSH(29),VSHX(6),WAVEL,XC,XFLOW,XFLOWI(6),XFLOWO(6),	RE101452
7XPD(120),XT(120),Z(29),ZD(8),ZM,FLOWI(14),FLOWX(14),PUPIL,SIGMA,	RE101453
8IPRT(10),APE1,APE2,RINT,ZO,FLO,CABEF,CABER2,PP,PC,NB,NC,FC	RE101454
DIMENSION ED(14),EH(14),XI(14),XO(14)	RE101455
C *** INITIAL EVALUATION OF PARAMETERS AND ARRAYS	RE101456
DO 800 J=1,N3	RE101457
BV(J,1)=0.	RE101458
BV(J,2)=0.	RE101459
BV(J,3)=0.	RE101460
FLOWI(J)=0.	RE101461
800 FLOWX(J)=0.	RE101462
RH(1)=R(2)/2.	RE101463
DO 803 J=2,JVL	RE101464
803 RH(J)=(R(J)+R(J+1))/2.	RE101465
L2=2	RE101466
DO 810 J=1,JVL	RE101467
805 IF(DFLOW(L2).GT.RH(J))GO TO 806	RE101468
L2=L2+1	RE101469
GO TO 805	RE101470
806 X1=DFLOW(L2)-DFLOW(L2-1)	RE101471
X2=RH(J)-DFLOW(L2-1)	RE101472
X3=X2/X1	RE101473
XI(J)=XFLOWI(L2-1)+X3*(XFLOWI(L2)-XFLOWI(L2-1))	RE101474
810 XO(J)=XFLOWO(L2-1)+X3*(XFLOWO(L2)-XFLOWO(L2-1))	RE101475
FLOWX(1)=0.	RE101476
DO 812 J=2,JVL	RE101477
812 FLOWX(J)=FLOWX(J-1)+(XI(J-1)-XO(J-1))*(R(J)*R(J)-R(J-1)*R(J-1))/	RE101478
1(2.*TVL)	RE101479
FLOWX(JVL+1)=FLOWX(JVL)	RE101480
L2=2	RE101481
FLOWI(1)=XFLOWI(1)/TVL	RE101482

DO 820 J=2,JVL	RE101483
814 IF (DFLOW(L2).GT.R(J)) GO TO 816	RE101484
L2=L2+1	RE101485
GO TO 814	RE101486
816 X4=DFLOW(L2)-DFLOW(L2-1)	RE101487
X5=R(J)-DFLOW(L2-1)	RE101488
X6=X5/X4	RE101489
820 FLOWI(J) = (XFLOWI(L2-1)+X6*(XFLOWI(L2)-XFLOWI(L2-1)))/TVL	RE101490
DO 823 J=2,JVL	RE101491
823 RD(J) = 1./ (R(J)*(R(J+1)-R(J-1)))	RE101492
C *** CALCULATE CHANGES IN MATRIX ELEMENTS A AND B DUE TO BLOOD FLOW	RE101493
BV(1,1)=0.	RE101494
BV(1,2)=-SHB*FLOWI(1)/2.	RE101495
BV(1,3)=0.	RE101496
BB=-SHB*XFLOW/2.	RE101497
DO 825 J=2,JVL	RE101498
BV(J,1)=SHB*RD(J)*FLOWX(J)	RE101499
BV(J,2)=SHB*RD(J)*(FLOWX(J-1)-FLOWX(J+1))/2.-SHB*FLOWI(J)/2.	RE101500
825 BV(J,3)=-SHB*RD(J)*FLOWX(J)	RE101501
DO 835 I=IPA,M	RE101502
835 IV(I)=0	RE101503
DO 840 L3=1,NVL	RE101504
L4=IBLOOD(L3)	RE101505
840 IV(L4)=1	RE101506
DO 845 I=IPA,LPS	RE101507
DO 842 J=1,JVL	RE101508
842 IAB(I,J)=0	RE101509
IF (JVL.EQ.N) GO TO 845	RE101510
L1=JVL+1	RE101511
DO 843 J=L1,N	RE101512
843 IAB(I,J)=1	RE101513
845 CONTINUE	RE101514
DO 850 I=IPT,M	RE101515
DO 850 J=1,N	RE101516
850 IAB(I,J)=1	RE101517
RETURN	RE101518
END	RE101519

1519 RECORDS PRINTED

```

C                               RETINAL MODEL IITRI                               RE200001
C                               VERSION 2 14 NOV 1975                             RE200002
C TEMPERATURE AND DAMAGE PREDICTIONS IN AND ABOUT RET CAUSED BY LASERS RE200003
C EFFECTS OF MELANIN GRANULES NOT INCORPORATED IN PROGRAM RE200004
C                               RE200005
COMMON A (29,3),AAV,ACH,APE,ASC,ATS,AVL,B(14,3),BB,BV(14,3), RE200006
1CONX(6),CON(29),CUT,DFLOW(6),DPULSE,DR,DT,DTX,DZ,FL,HP(14), RE200007
2IAB(29,14),IBLOOD(10),IFIL,IGX,IHT,IPA,IPC,IPE,IPOF,IPS,IPT, RE200008
3IPV,IV(29),JVL,LIM,LPA,LPC,LPE,LPS,LPV,LPX,K,KH,KT,H,M1,M2, RE200009
4M3,N,N1,N3,N4,NVL,POX,PR(14),PTIME,QP,R(14),RCO,RIM,RN,RPE,RRT, RE200010
5RVL,RSC,S(29,14),SHB,TAV,TCH,TOM,TPE,TVL,TSC,TTS,V(29,14) RE200011
6,VC(29,14,120),VSH(29),VSHX(6),WAVEL,XC,XFLOW,XFLOWI(6),XFLOWO(6), RE200012
7XT(120),Z(29),ZD(8),ZM,FLOWI(14),FLOWX(14),PUPIL,SIGMA, RE200013
8IPRT(10),APE1,APE2,RINT,ZO,FLO,CABER,CABER2,PP,PC,NB,NC,PC RE200014
DIMENSION CXC(14),CXR(29),DAMAGE(2,2),DXC(14),DXR(29),FTIME(38), RE200015
1FXC(14),FXR(29),ID(230),JD(230),KTT(38),NPT(38),NPULSE(7),NRUN(7), RE200016
2QD(29,14),REPFT(7),TIMEX(10),XCT(38),XQD(29,14),VE(27,120,1), RE200017
3VXX(29,14),VZ(27,42,8,1),ZT(8),ZTT(8),SAVRGV(10) RE200018
REAL LESION RE200019
2 FORMAT(10F7.3) RE200020
3 FORMAT(F7.4,3I7) RE200021
4 FORMAT(11F7.2) RE200022
5 FORMAT(10I7) RE200023
6 FORMAT(F7.2,I7,2F7.2) RE200024
7 FORMAT(10E7.2) RE200025
8 FORMAT(I7,3E7.2) RE200026
9 FORMAT(F7.2,2I7,F7.2) RE200027
300 READ(5,4,END=200)(FTIME(L),L=1,38) RE200028
READ(5,5)IPRT RE200029
READ(5,3)RIM,LIM,IFIL,IGX RE200030
READ(5,9)RMAX,LIMAX,MAXPRT,LESION RE200031
C *** SET VALUES FOR MTEST,N,N1,N3,N4, AND DR RE200032
MTEST=0 RE200033
N1=4 RE200034
N=N1+9 RE200035
N3=N+1 RE200036
N4=N1+1 RE200037
READ(5,8)IPROF,POW,CUT RE200038
DR=LESION/LIM RE200039
IF(IPROF.EQ.0)DR=RIM/(LIM-.5) RE200040
READ(5,7)DPULSE RE200041
READ(5,5)NTEST,(NRUN(L),L=1,NTEST) RE200042
READ(5,7)(REPET(L),L=1,NTEST) RE200043
READ(5,5)(NPULSE(L),L=1,NTEST) RE200044
READ(5,5)ID1,ID2,JD1,JD2,ITYPE RE200045
LPX=1 RE200046
IF(NTEST.EQ.1.AND.NPULSE(1).EQ.1)LPX=0 RE200047
XDPULS=DPULSE RE200048
XXQ=1. RE200049
IF(DPULSE.GT..3E-8)GO TO 10 RE200050
C *** ADJUST POWER AND PULSE WIDTH FOR EXPOSURES WITH PULSES LESS THAN RE200051
C *** .3E-8 SEC RE200052
XXQ=.3E-8/DPULSE RE200053
POW=POW*DPULSE/.3E-8 RE200054
DPULSE=.3E-8 RE200055
10 READ(5,4)TO,EDT1,EDT2 RE200056
READ(5,4)TOM,APE,AVL,ACH,ASC,ATS,RCO,RRT,RSC,RPE,WAVEL RE200057

```

READ (5,4) TAV,TPE,TVL,TCH,TSC,RVL	RE200058
AAV=-ALOG (TOM)/TAV	RE200059
READ (5, 4) (CONX (L) ,L=1,6)	RE200060
READ (5,4) (VSHX (L) ,L=1,6)	RE200061
READ (5,5) (NPT (L) ,L=1,38)	RE200062
READ (5,2) (XCT (L) ,L=1,38)	RE200063
READ (5,5) (KTT (L) ,L=1,38)	RE200064
C *** COMPUTE DT,KM,KT,NP,PTIME,TIME, AND XC	RE200065
L1=ALOG (DPULSE)/.69315+29.	RE200066
IF (L1.LT.1) L1=1	RE200067
IF (L1.GT.38) L1=38	RE200068
IF (LPX.EQ.1) GO TO 11	RE200069
C *** ---SINGLE PULSED EXPOSURES	RE200070
XC=XCT (L1)	RE200071
NP=NPT (L1)	RE200072
KT=KTT (L1)	RE200073
DT=DPULSE*(XC-1.)/(XC**NP-1.)	RE200074
TIME=DT*(XC**KT-1.)/(XC-1.)	RE200075
GO TO 13	RE200076
C *** ---MULTIPLE PULSED EXPOSURES	RE200077
11 XC=1.4	RE200078
NP=5	RE200079
X1=0.	RE200080
DO 12 L=1,NTEST	RE200081
IF (X1.LT.NPULSE (L)/REPET (L)) X1=NPULSE (L)/REPET (L)	RE200082
12 CONTINUE	RE200083
TIME=PTIME (L1)*X1	RE200084
DT=DPULSE*(XC-1.)/(XC**NP-1.)	RE200085
KT=ALOG (1.+TIME*(XC-1.)/DT)/ALOG (XC)+1.	RE200086
PTIME=DPULSE/NP	RE200087
13 KT=KT+1	RE200088
KM=NP+1	RE200089
IF (KT.GT.119) WRITE (6,14) KT	RE200090
14 FORMAT (1H0,3HKT=,I3,2X,22HTIME DIMENSION TOO LOW)	RE200091
IF (KT.GT.119) STOP	RE200092
C *** CALC. DZ AND I INDICES	RE200093
M1=6	RE200094
M=2*M1+16	RE200095
M2=M/2	RE200096
M3=M+1	RE200097
IPE=M2-M1+2	RE200098
DZ=TPE/M1-1.E-25	RE200099
IPA=2	RE200100
C *** STORE AXIAL DISTANCES TO INTERFACES OF EYE	RE200101
ZD (1) =1.E-25	RE200102
ZD (2) =TAV	RE200103
ZD (3) =ZD (2) +RPE*TPE	RE200104
ZD (4) =ZD (3) + (1.-RPE)*TPE	RE200105
ZD (5) =ZD (4) +TVL	RE200106
ZD (6) =ZD (5) +TCH	RE200107
ZD (7) =ZD (6) +TSC	RE200108
ZD (8) =ZD (7) +10.	RE200109
CALL GRID	RE200110
NVL=LPV-IPV+1	RE200111
C *** CALCULATE AND STORE I,J INDICES AT WHICH TEMPERATURES ARE PRINTED	RE200112
ID1=ID1+IPE	RE200113
ID2=ID2+IPE	RE200114

IF (ID1.LT.IPA) ID1=IPA	RE200115
IF (ID2.GT.M) ID2=M	RE200116
IF (JD2.GT.N) JD2=N	RE200117
IF (IPRT (1) .EQ.0) GO TO 23	RE200118
WRITE (6, 15) ID1, ID2, JD1, JD2	RE200119
15 FORMAT (1H0, 5X, 4HID1=, I3, 3X, 4HID2=, I3, 3X, 4HJD1=, I2, 3X, 4HJD2=, I2)	RE200120
WRITE (6, 16) DR, DZ	RE200121
16 FORMAT (1H0, 5X, 3HDR=, E11.4, 2X, 3HDZ=, E11.4)	RE200122
WRITE (6, 17) IPA, IPC, IPE, IPS, IPT, IPV, LPA, LPC, LPE, LPS, LPV	RE200123
17 FORMAT (1H0, 5X, 4HIPA=, I3, 2X, 4HIPC=, I3, 2X, 4HIPE=, I3, 2X, 4HIPS=, I3, 2X,	RE200124
14HIPT=, I3, 2X, 4HIPV=, I3/1H, 5X, 4HLPA=, I3, 2X, 4HLPC=, I3, 2X, 4HLPE=, I3,	RE200125
22X, 4HLPS=, I3, 2X, 4HLPV=, I3)	RE200126
WRITE (6, 22) M, M1, N, N1	RE200127
22 FORMAT (1H0, 5X, 2HM=, I2, 2X, 3HM1=, I2, 2X, 2HN=, I2, 2X, 3HN1=, I2)	RE200128
WRITE (6, 18) (R (J), J=1, N3)	RE200129
18 FORMAT (1H0, 5X, 2HR=/(1H, 5X, 10F8.4))	RE200130
WRITE (6, 19) (Z (I), I=1, M3)	RE200131
19 FORMAT (1H0, 5X, 2HZ=/(1H, 5X, 10F8.4))	RE200132
23 DO 20 L1=1, NVL	RE200133
20 IBLOOD (L1)=IPV+I1-1	RE200134
C *** CALC. NORMALIZED LASER PROFILES---	RE200135
DO 21 L=1, N3	RE200136
21 HR (L) =0.	RE200137
POX=POW	RE200138
CALL IMAGE	RE200139
DO 27 J=1, N3	RE200140
DO 27 I=1, M3	RE200141
V (I, J) =1.E-10	RE200142
27 S (I, J) =0.	RE200143
READ (5, 2) SHB, XFLOW, CFLOW	RE200144
C *** SET BLOOD FLOW RATES ENTERING AND LEAVING VASCULAR LAYER AS	RE200145
C *** FUNCTION OF RADIAL DISTANCE	RE200146
X2=CFLOW/(3.1416*EVL*RVL)	RE200147
DFLOW (1) =0.	RE200148
X4=0.	RE200149
DO 30 L1=2, 6	RE200150
X4=X4+.1	RE200151
30 DFLOW (L1) =X4	RE200152
DO 31 L1=1, 6	RE200153
XFLOWI (L1) =X2	RE200154
31 XFLOWO (L1) =X2	RE200155
DO 34 I=1, M3	RE200156
DO 34 J=1, N3	RE200157
34 VC (I, J, 1) =1.E-10	RE200158
XPOW=XXQ*POW	RE200159
READ (5, 8) KTYPEO	RE200160
READ (5, 8) KTYPE	RE200161
L1=KTYPE	RE200162
IF (KTYPE.EQ.0) L1=1	RE200163
READ (5, 7) (TIMEX (K), K=1, L1)	RE200164
READ (5, 5) II1, II2, II3, JJ1, JJ2	RE200165
C *** START OF TEMPERATURE CALCULATIONS FOR ONE PULSE. TO BE USED EITHER	RE200166
C *** FOR MULTIPLE OR SINGLE PULSED EXPOSURES	RE200167
C -----	RE200168
XT (1) =0.	RE200169
DTX=DT	RE200170
KTX=KT+1	RE200171

DO 36 K=2,KTX	RE200172
XT(K)=XT(K-1)+DT	RE200173
36 DT=XC*DT	RE200174
IKX=TIME**EDT1+EDT2	RE200175
IF(IKX.LT.1)IKX=1	RE200176
XX=2*IKX	RE200177
K=2	RE200178
IHT=2	RE200179
ITYPEX=ITYPE	RE200180
CALL BLOOD	RE200181
38 DT=XT(K)-XT(K-1)	RE200182
IF(K.GT.KM)QP=0.	RE200183
CALL HTXDEP	RE200184
IF(K.GT.2)GO TO 41	RE200185
IF(IPRT(2).EQ.0)GO TO 335	RE200186
WRITE(6,301)	RE200187
301 FORMAT(1H0,13HLASER PROFILE)	RE200188
IF(IPROF.EQ.0)WRITE(6,302)RIM	RE200189
302 FORMAT(1H0,5X,4HRIM=,E10.3)	RE200190
IF(IPROF.EQ.1)WRITE(6,303)SIGMA,RIM,CUT	RE200191
303 FORMAT(1H0,5X,6HSIGMA=,E10.3,5X,4HRIM=,E10.3,5X,4HCUT=,E10.3)	RE200192
IF(IFIL.EQ.1)WRITE(6,304)RINT,ZO,FLO,CABER,CABER2,PP,PC,NB,NC,FC,	RE200193
1WAVEL	RE200194
304 FORMAT(1H0,5X,5HRINT=,E10.3,3X,3HZO=,E10.3,3X,4HFLO=,F6.3/1H,5X,	RE200195
16HCABER=,E10.3,3X,7HCABER2=,F7.0,3X,3HPP=,F6.3/1H,5X,3HPC=,F6.3,	RE200196
23X,3HNB=,F7.3,3X,3HNC=,F7.3/1H,5X,3HFC=,F6.3,3X,6HWAVEL=,F7.1)	RE200197
IF(IFIL.EQ.1)GO TO 306	RE200198
IF(IPROF.EQ.2)WRITE(6,305)RINT	RE200199
305 FORMAT(1H0,5X,5HRINT=,E10.3)	RE200200
306 WRITE(6,307)QP	RE200201
307 FORMAT(1H0,5X,3HQP=,E10.3)	RE200202
WRITE(6,308)(HR(J),J=1,N)	RE200203
308 FORMAT(1H0,5X,3HHR=/(1H,10X,10E10.3))	RE200204
335 IF(IPRT(3).EQ.0)GO TO 340	RE200205
WRITE(6,309)	RE200206
309 FORMAT(1H0,19HDATA IDENTIFICATION)	RE200207
WRITE(6,310)(REPET(L),L=1,NTEST)	RE200208
310 FORMAT(1H0,5X,6HREPET=/(1H,5X,10E10.3))	RE200209
WRITE(6,311)(NPULSE(L),L=1,NTEST)	RE200210
311 FORMAT(1H0,5X,7HNPULSE=/(1H,5X,10I8))	RE200211
WRITE(6,312)AAV,ACH,APE,ASC,ATS,RCO,RRT,RPE,TOM,AVL,TAV,TCH,TPE,	RE200212
1TSC,TVL,IGX,IFIL,IPROF,LIM,NTEST,POW,XDPULS,RIM,RMAX,TIME,CFLOW,	RE200213
2XPLOW,SHB,EDT1,EDT2,DT,KM,KT,PTIME,XC,IKX,APE1,APE2,RVL,	RE200214
3PUPIL,T0,LIMAX,MAXPPT	RE200215
312 FORMAT(1H0,5X,4HAAV=,F7.1,2X,4HACH=,F7.0,2X,4HAPE=,F7.0,2X,4HASC=	RE200216
1,F7.0,2X,4HATS=,F7.0/1H,5X,4HRCO=,F7.4,2X,4HRRT=,F7.4,2X,4HRPE=,	RE200217
2F7.4,2X,4HTOM=,F7.4,2X,4HAVL=,F7.0/1H,5X,4HTAV=,E9.3,2X,4HTCH=,	RE200218
3E9.3,2X,4HTPL=,E9.3,2X,4HTSC=,E9.3,2X,4HTVL=,E9.3/1H,5X,4HIGX=,I2	RE200219
4,2X,5HIPIL=,I2,2X,6HIPROF=,I2,2X,4HLIM=,I2,2X,6HNTEST=,I2/1H,5X,	RE200220
54HPOW=,E9.3,2X,7HDPULSE=,E9.3,2X,4HFIM=,F7.4,2X,5HRMAX=,F7.4,2X,	RE200221
65HTIME=,E9.3/1H,5X,6HCFLOW=,F7.4,2X,6HXFLOW=,F7.4,2X,4HSHB=,F7.2,	RE200222
72X,5HEDT1=,F7.4,2X,5HEDT2=,F7.4/1H,5X,3HDT=,E9.3,2X,3HKM=,I3,2X,	RE200223
83HKT=,I3,2X,6HPTIME=,E9.3,2X,3HXC=,F5.1/1H,5X,4HIRX=,I2,2X,	RE200224
9	
5SHAPE1=,F8.2,2X,5SHAPE2=,F8.2	RE200225
1F6.3,2X,6HPUPIL=,F6.3,2X,3HT0=,F5.1,2X,6HLIMAX=,I2,2X,7HMAXPRT=,	RE200226
2I2)	RE200227
340 IF(IPRT(4).EQ.0)GO TO 355	RE200228

	WRITE (6, 313)	RE200229
313	FORMAT (1H0, 30HBLOOD FLOW AND HEAT DEPOSITION)	RE200230
	WRITE (6, 314) (FLOWI (J), J=1, JVL)	RE200231
314	FORMAT (1H0, 5X, 6HFLOWI=/(1H, 5X, 10E10.3))	RE200232
	WRITE (6, 315) (FLOWX (J), J=1, JVL)	RE200233
315	FORMAT (1H0, 5X, 6HFLOWX=/(1H, 5X, 10E10.3))	RE200234
	WRITE (6, 316)	RE200235
316	FORMAT (1H)	RE200236
	DO 318 I=IPA, M	RE200237
	WRITE (6, 317) (S (I, J), J=1, N)	RE200238
317	FORMAT (1H, 5X, 2HS=, 10E8.3)	RE200239
318	CONTINUE	RE200240
355	IF (IPRT (5) .EQ. 0) GO TO 41	RE200241
	WRITE (6, 319)	RE200242
319	FORMAT (1H0, 17HTEMPERATURE RISES)	RE200243
	JCNT=JD2-JD1+1	RE200244
	IF (JCNT.GT. 9) GO TO 40	RE200245
	GO TO 41	RE200246
40	JJCNT=JCNT-9	RE200247
	JJD2=JD2-JJCNT	RE200248
	JJD2P1=JJD2+1	RE200249
41	IF (IPRT (5) .EQ. 0) GO TO 356	RE200250
	WRITE (6, 42) XT (K), K	RE200251
42	FORMAT (1H0, 5X, 5HTIME=, E11.4, 3X, 2HK=, I3)	RE200252
C ***	CALCULATE TEMPERATURE RISE (MATRIX REDUCTION ALGORITHM)	RE200253
C ***	COLUMNS (NORMAL) -----	RE200254
356	IK=1	RE200255
43	DO 45 I=IPA, M	RE200256
	W=XX*VSH (I) /DT	RE200257
	DO 44 J=1, N	RE200258
	FXC (J)=W+CON (I) *B (J, 2)-BV (J, 2) *IV (I)-BB*IAB (I, J)	RE200259
	IF (J.GT. 1) FXC (J)=FXC (J)+(CON (I) *B (J, 1)+BV (J, 1) *IV (I)) *CXC (J-1)	RE200260
	CXC (J)=- (CON (I) *B (J, 3)+BV (J, 3) *IV (I)) /FXC (J)	RE200261
	SUM=(W-(A (I, 2)-BV (J, 2) *IV (I)-BB*IAB (I, J))) *V (I, J)+A (I, 1) *V (I-1, J)+	RE200262
1A (I, 3) *V (I+1, J)+S (I, J)		RE200263
	DXC (J)=SUM/FXC (J)	RE200264
	IF (J.GT. 1) DXC (J)=(SUM+(CON (I) *B (J, 1)+BV (J, 1) *IV (I)) *DXC (J-1)) /FXC (RE200265
	1J)	RE200266
44	CONTINUE	RE200267
	VX=0.	RE200268
	DO 45 L=1, N	RE200269
	J=N+1-L	RE200270
	VX=DXC (J)-CXC (J) *VX	RE200271
45	VXX (I, J)=VX	RE200272
	DO 46 I=IPA, M	RE200273
	DO 46 J=1, N	RE200274
46	V (I, J)=VXX (I, J)	RE200275
C ***	ROWS (NORMAL) -----	RE200276
	CXR (IPA-1)=0.	RE200277
	DO 50 J=1, N	RE200278
	DO 48 I=IPA, M	RE200279
	W=XX*VSH (I) /DT	RE200280
	FXR (I)=W+A (I, 2)-BV (J, 2) *IV (I)-BB*IAB (I, J)+A (I, 1) *CXR (I-1)	RE200281
	CXR (I)=-A (I, 3) /FXR (I)	RE200282
	SUM=(W-(CON (I) *B (J, 2)-BV (J, 2) *IV (I)-BB*IAB (I, J))) *V (I, J)+(CON (I) *	RE200283
1B (J, 3) +BV (J, 3) *IV (I)) *V (I, J+1)+S (I, J)		RE200284
	IF (J.GT. 1) SUM=SUM+(CON (I) *B (J, 1)+BV (J, 1) *IV (I)) *V (I, J-1)	RE200285

	DXR (I) =SUM/FXR (I)	RE200286
	IF (I.GT.IPA) DXR (I) = (SUM+A (I, 1) *DXR (I-1)) /FXR (I)	RE200287
48	CONTINUE	RE200288
	VX=0.	RE200289
	DO 50 L=IPA, M	RE200290
	I=M+IPA-L	RE200291
	VX=DXR (I) -CXR (I) *VX	RE200292
	VC (I, J, K) =VX	RE200293
50	VXX (I, J) =VX	RE200294
	DO 51 I=IPA, M	RE200295
	DO 51 J=1, N	RE200296
51	V (I, J) =VXX (I, J)	RE200297
	IK=IK+1	RE200298
C ***	RECYCLE TEMPERATURE CALCULATIONS	RE200299
	IF (IK.LE.IKX) GO TO 43	RE200300
	IF (K.EQ.KM) GO TO 62	RE200301
	IF (ITYPEX.LT.ITYPE.AND.K.LT.KT) GO TO 66	RE200302
62	IF (IPRT (5) .EQ.0) GO TO 357	RE200303
	WRITE (6, 63) (R (J) , J=JD1, JD2)	RE200304
63	FORMAT (1H , 13X, 2HF=, 9F13.5/1H , 15X, 30H-----	RE200305
	1--)	RE200306
	DO 65 I=ID1, ID2	RE200307
	X1=Z (I) -Z (IPE) +DZ/2.	RE200308
	IF (JCNT.GT.9) GO TO 57	RE200309
	WRITE (6, 64) X1, (VC (I, J, K) , J=JD1, JD2)	RE200310
	GO TO 65	RE200311
57	WRITE (6, 64) X1, (VC (I, J, K) , J=JD1, JJD2)	RE200312
	WRITE (6, 64) X1, (VC (I, J, K) , J=JJD2P1, JD2)	RE200313
64	FORMAT (1H , 3X, 2HZ=, F8.5, 2X, 1P9E13.6)	RE200314
65	CONTINUE	RE200315
357	ITYPEX=0	RE200316
66	K=K+1	RE200317
	ITYPEX=ITYPEX+1	RE200318
	IF (K.LE.KT) GO TO 38	RE200319
	ITYPEX=ITYPE	RE200320
	IF (IPRT (6) .EQ.0) GO TO 365	RE200321
	WRITE (6, 320)	RE200322
320	FORMAT (1H0, 28HNORMALIZED TEMPERATURE RISES)	RE200323
	DO 70 K=2, KT	RE200324
	IF (K.EQ.KM) GO TO 67	RE200325
	IF (ITYPEX.LT.ITYPE.AND.K.LT.KT) GO TO 70	RE200326
67	X1=1.	RE200327
	WRITE (6, 321) XT (K) , K, X1	RE200328
321	FORMAT (1H0, 5X, 5HTIME=, E11.4, 3X, 2HK=, I3, 3X, 6HPower=, E11.4, 5HWATTS)	RE200329
	WRITE (6, 63) (R (J) , J=JD1, JD2)	RE200330
	JCNT=JD2-JD1+1	RE200331
	IF (JCNT.GT.9) GO TO 380	RE200332
	GO TO 381	RE200333
380	JJCNT=JCNT-9	RE200334
	JJD2=JD2-JJCNT	RE200335
	JJD2P1=JJD2+1	RE200336
381	DO 69 I=ID1, ID2	RE200337
	DO 68 J=JD1, JD2	RE200338
68	V (I, J) =VC (I, J, K) /POW	RE200339
	X1=Z (I) -Z (IPE) +DZ/2.	RE200340
	IF (JCNT.GT.9) GO TO 382	RE200341
	WRITE (6, 64) X1, (V (I, J) , J=JD1, JD2)	RE200342

	GO TO 69	RE200343
382	WRITE (6,64) X1, (V(I,J), J=JD1, JJD2)	RE200344
	WRITE (6,64) X1, (V(I,J), J=JJD2P1, JD2)	RE200345
69	CONTINUE	RE200346
	ITYPEX=0	RE200347
70	ITYPEX=ITYPEX+1	RE200348
330	FORMAT (1H0, 61HDIMENSION OF ARRAYS ASSOCIATED WITH ARGUMENT LIJ IS	RE200349
	1TOO SMALL)	RE200350
365	CONTINUE	RE200351
	READ (5,4) (DAMAGE (L2,1), DAMAGE (L2,2), L2=1,2)	RE200352
	WRITE (6,73) WAVEL, DAMAGE (1,1), DAMAGE (1,2), DAMAGE (2,1),	RE200353
	1DAMAGE (2,2)	RE200354
73	FORMAT (1H0, 5X, 11HWAVELENGTH=, F7.1, 2HNM, 3X, 7HDAMAGE=, 4F9.0)	RE200355
C ***	CALCULATE I, J INDICES AT WHICH DAMAGE CALCULATIONS ARE TO BE MADE	RE200356
	JM=0	RE200357
	DO 74 J=1, N	RE200358
	IF (R(J) .LT. RMAX+.000001) JM=J+1	RE200359
74	CONTINUE	RE200360
	X1=0.	RE200361
	DO 75 I=IPA, M	RE200362
	IF (VC (I, 1, KM) .GT. X1) IMAX=I	RE200363
	IF (VC (I, 1, KM) .GT. X1) X1=VC (I, 1, KM)	RE200364
75	CONTINUE	RE200365
	L=0	RE200366
	GO TO (366, 367, 368), MAXPRT	RE200367
366	LIMAX1=2*LIMAX	RE200368
	LIMAX2=0	RE200369
	GO TO 369	RE200370
367	LIMAX1=LIMAX	RE200371
	LIMAX2=LIMAX	RE200372
	GO TO 369	RE200373
368	LIMAX1=0	RE200374
	LIMAX2=2*LIMAX	RE200375
369	ID1=IMAX-LIMAX1	RE200376
	ID2=IMAX+LIMAX2	RE200377
	IF (ID2.GT.28) ID2=28	RE200378
	DO 76 I=ID1, ID2	RE200379
	DO 76 J=1, JM	RE200380
	L=L+1	RE200381
	ID (L) =I	RE200382
76	JD (L) =J	RE200383
	LIJ=(ID2-ID1+1)*JM	RE200384
	DO 505 LL15=1, 10	RE200385
505	SAVRGV (LL15)=0.	RE200386
	IF (LPX.EQ.0) GO TO 125	RE200387
	IF (LIJ.GT.27) WRITE (6, 330)	RE200388
	IF (LIJ.GT.27) GO TO 300	RE200389
	IF (IPRT (8) .EQ.0) GO TO 370	RE200390
C ***	TEMPERATURE AND DAMAGE EVALUATIONS FOR MULTIPLE PULSES	RE200391
C	-----	RE200392
C ***	EVALUATE TEMPERATURE RISES WITHOUT GRANULES	RE200393
	DO 77 L=1, LIJ	RE200394
	I=ID (L)	RE200395
	J=JD (L)	RE200396
	VE (L, 1, 1)=0.	RE200397
	DO 77 K=2, KT	RE200398
	VE (L, K, 1)=VC (I, J, K)	RE200399

AD-A034 239

SCHOOL OF AEROSPACE MEDICINE BROOKS AFB TEX
RETINAL THERMAL MODEL OF LASER-INDUCED EYE DAMAGE: COMPUTER OPE--ETC(U)
NOV 76 A R MERTZ, B R ANDERSON, E L BELL

F/G 9/2

UNCLASSIFIED

SAM-TR-76-33

NL

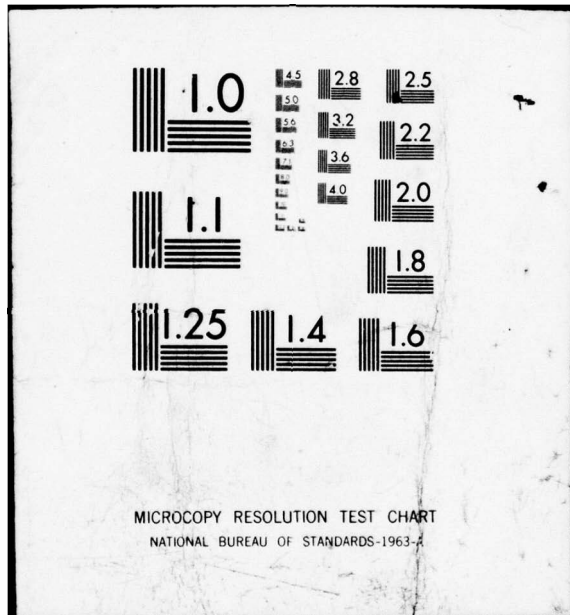
2 OF 2

AD
A034239



END

DATE
FILMED
2-77



```

77 CONTINUE RE200400
   X60=(XC-1.)/DTX RE200401
   X61=ALOG(XC) RE200402
370 L13=0 RE200403
371 L13=L13+1 RE200404
   X3=DPULSE+(NPULSE(L13)-1)/REPET(L13) RE200405
   WRITE(6,78)NRUN(L13),X3,XDPULS, NPULSE(L13), REPET(L13) RE200406
78 FORMAT(1H0,5X,5HNRUN=,I3,2X,13HTRAIN LENGTH=,E10.3,3HSEC,2X,12HPUL RE200407
1SE WIDTH=,E10.3,3HSEC/1H,5X,17HNUMBER OF PULSES=,I5,3X,16HREPETITRE200408
2ION RATE=,E10.3,10HPULSES/SEC) RE200409
   IF(IFIL.EQ.0)GO TO 80 RE200410
   WRITE(6,79)RIM,LESION RE200411
79 FORMAT(1H,5X,12HBEAM RADIUS=,E10.3,2HCM,5X,14HLESION RADIUS=,E10. RE200412
13,2HCM) RE200413
   GO TO 82 RE200414
80 WRITE(6,81)RIM,LESION RE200415
81 FORMAT(1H,5X,13HIMAGE RADIUS=,E10.3,2HCM,5X,14HLESION RADIUS=,E10 RE200416
1.3,2HCM) RE200417
82 IF(IPRT(8).EQ.0)GO TO 108 RE200418
   TC=1./REPET(L13) RE200419
   NPL=NPULSE(L13) RE200420
   KX=NP+3 RE200421
   IN=1 RE200422
83 IF(NPL/IN.LT.20)GO TO 84 RE200423
   IN=IN+2 RE200424
   GO TO 83 RE200425
84 X1=NPL RE200426
   INX=.5+X1/IN RE200427
   L1=ALOG(DPULSE)/.69315+29. RE200428
   IF(L1.LT.1)L1=1 RE200429
   INXX=FTIME(L1)*INX RE200430
C *** STORE TIME INTERVALS AND LOGS OF INTERVALS FOR DAMAGE CALCULATIONS RE200431
   ZT(1)=PTIME/2. RE200432
   ZTT(1)=ALOG(IN*PTIME) RE200433
   DO 85 L3=2,NP RE200434
   ZTT(L3)=ALOG(IN*PTIME) RE200435
85 ZT(L3)=ZT(L3-1)+PTIME RE200436
   L1=NP+1 RE200437
   X3=(TC-DPULSE)/(KX-NP) RE200438
   ZT(L1)=DPULSE+X3/2. RE200439
   ZTT(L1)=ALOG(IN*X3) RE200440
   L1=L1+1 RE200441
   DO 86 L3=L1,KX RE200442
   ZTT(L3)=ALOG(IN*X3) RE200443
86 ZT(L3)=ZT(L3-1)+X3 RE200444
C *** CALCULATE TEMPERATURE RISES ASSOCIATED WITH L3-TH TIME INTERVAL RE200445
C *** FOLLOWING (L6-.5)*IN-.5 PULSE RE200446
   DO 95 L=1,LIJ RE200447
   DO 95 L3=1,KX RE200448
   X1=0. RE200449
   L1=1+IN/2 RE200450
   L7=1 RE200451
87 X3=(L7-1)*TC+ZT(L3) RE200452
   K=ALOG(X3*X60+1.)/X61+1. RE200453
   X5=VE(L,K,1)+(X3-XT(K))*(VE(L,K+1,1)-VE(L,K,1))/(XT(K+1)-XT(K)) RE200454
   X1=X1+X5 RE200455
   IF(X5.LT..0001*X1)GO TO 88 RE200456

```

L7=L7+1	RE200457
IF (L7.LE.L1) GO TO 87	RE200458
88 VZ (L, 1, L3, 1) =X1	RE200459
DO 93 L6=2, INXX	RE200460
IF (X5.LT..0001*X1) GO TO 93	RE200461
X1=VZ (L, L6-1, L3, 1)	RE200462
L2=L1+1	RE200463
L1=L1+IN	RE200464
L7=L2	RE200465
90 X3= (L7-1)*TC+ZT (L3)	RE200466
K=ALOG (X3*X60+1.)/X61+1.	RE200467
X5=VE (L, K, 1) + (X3-YT (K)) * (VE (L, K+1, 1) -VE (L, K, 1)) / (YT (K+1) -YT (K))	RE200468
X1=X1+X5	RE200469
IF (X5.LT..0001*X1) GO TO 93	RE200470
L7=L7+1	RE200471
IF (L7.LE.L1) GO TO 90	RE200472
93 VZ (L, L6, L3, 1) =X1	RE200473
L1=INX+1	RE200474
DO 94 L6=L1, INXX	RE200475
L8=L6-INX	RE200476
94 VZ (L, L6, L3, 1) =VZ (L, L6, L3, 1) -VZ (L, L8, L3, 1)	RE200477
95 CONTINUE	RE200478
C *** DAMAGE CALCULATIONS -----	RE200479
WRITE (6, 375)	RE200480
375 FORMAT (1H0, 31HPREDICTED THRESHOLD LASER POWER)	RE200481
DO 104 L=1, LIJ	RE200482
I=ID (L)	RE200483
J=JD (L)	RE200484
IF (VZ (L, INX, NP, 1) .LT..001) QD (I, J) =1.E+20	RE200485
IF (VZ (L, INX, NP, 1) .LT..001) GO TO 104	RE200486
L9=10. * (.4+EXP (-.0014*DPULSE)) /VZ (L, INX, NP, 1)	RE200487
CQ=L9+1.	RE200488
X10=70. * (.4+EXP (-.0014*DPULSE)) /VZ (L, INX, NP, 1)	RE200489
IF (L9.EQ.0) CQ=X10	RE200490
LLT=0	RE200491
LGT=0	RE200492
99 DAMC=0.	RE200493
L6=1	RE200494
100 DO 101 L3=1, KX	RE200495
X3=0.	RE200496
X50=VZ (L, L6, L3, 1) *CQ+273.+T0	RE200497
IF (X50.LT.317.) GO TO 101	RE200498
X1=ZTT (L3) +DAMAGE (1, 1) -DAMAGE (1, 2) /X50	RE200499
IF (X50.GT.323.) X1=ZTT (L3) +DAMAGE (2, 1) -DAMAGE (2, 2) /X50	RE200500
IF (X1.GT.0.) X3=1.01	RE200501
IF (X1.GT.0.) GO TO 101	RE200502
X3=EXP (X1)	RE200503
101 DAMC=DAMC+X3	RE200504
IF (DAMC.GT.1.) GO TO 102	RE200505
C *** INCREASE TIME INDICES AND CONTINUE	RE200506
L6=L6+1	RE200507
IF (L6.LE.INXX) GO TO 100	RE200508
C *** ADJUST LASER POWER TO YIELD THRESHOLD DAMAGE AT GIVEN POINT	RE200509
IF (LGT.EQ.1) CQ=1.02*CQ	RE200510
IF (LGT.EQ.1) GO TO 103	RE200511
LLT=1	RE200512
CQ=1.04*CQ	RE200513

GO TO 99	RE200514
102 IF (LLT.EQ.1) CQ=.98*CQ	RE200515
IF (LLT.EQ.1) GO TO 103	RE200516
LGT=1	RE200517
CQ=.96*CQ	RE200518
GO TO 99	RE200519
103 QD(I,J)=CQ*POX	RE200520
104 CONTINUE	RE200521
WRITE(6,63) (R(J),J=1,JM)	RE200522
DO 97 I=ID1,ID2	RE200523
DO 97 J=1,JM	RE200524
97 XQD(I,J)=QD(I,J)*XXQ	RE200525
DO 106 I=ID1,ID2	RE200526
X1=Z(I)-Z(IPE)+DZ/2.	RE200527
IF(JM.GT.9) GO TO 385	RE200528
WRITE(6,105) X1, (XQD(I,J),J=1,JM)	RE200529
GO TO 106	RE200530
385 WRITE(6,105) X1, (XQD(I,J),J=1,9)	RE200531
WRITE(6,105) X1, (XQD(I,J),J=10,JM)	RE200532
105 FORMAT(1H,2X,2HZ=,F7.5,1X,3HQD=,1P9E13.6)	RE200533
106 CONTINUE	RE200534
108 IF(KTYPE.EQ.0) GO TO 174	RE200535
C *** CALCULATE AND STORE (MULTIPLE PULSE EXPOSURE) TEMPERATURES FOR	RE200536
C *** PLOTTING PROFILES	RE200537
TC=1./REPET(L13)	RE200538
NPL=NPULSE(L13)	RE200539
WRITE(6,139)	RE200540
DO 123 L15=1,KTYPE	RE200541
IF(TIMEX(L15).GT.XT(KT)) GO TO 123	RE200542
RGV=0.	RE200543
L2=TIMEX(L15)/TC	RE200544
DTIME=TIMEX(L15)-L2*TC	RE200545
L2=L2+1	RE200546
DO 116 I=II1,II2	RE200547
DO 116 J=JJ1,JJ2	RE200548
X1=0.	RE200549
DO 113 L6=1,L2	RE200550
K=ALOG((DTIME+(L6-1)*TC)*X60+1.)/X61+1.	RE200551
X2=(DTIME+(L6-1)*TC-XT(K))/(XT(K+1)-XT(K))	RE200552
113 X1=X1+VC(I,J,K)+X2*(VC(I,J,K+1)-VC(I,J,K))	RE200553
V(I,J)=X1	RE200554
L3=L2-NPL	RE200555
IF(L3.LE.0) GO TO 115	RE200556
X1=0.	RE200557
DO 114 L6=1,L3	RE200558
K=ALOG((DTIME+(L6-1)*TC)*X60+1.)/X61+1.	RE200559
X2=(DTIME+(L6-1)*TC-XT(K))/(XT(K+1)-XT(K))	RE200560
114 X1=X1+VC(I,J,K)+X2*(VC(I,J,K+1)-VC(I,J,K))	RE200561
V(I,J)=V(I,J)-X1	RE200562
115 IF(V(I,J).GT.RGV) RGV=V(I,J)	RE200563
116 CONTINUE	RE200564
SAVRGV(L15)=RGV	RE200565
IF(KTYPE.EQ.1) GO TO 121	RE200566
WRITE(7,117) NRUN(L13), NPULSE(L13), REPET(L13)	RE200567
117 FORMAT(2I7,E10.4)	RE200568
WRITE(7,118) XDPULS,WAVEL,RIM	RE200569
118 FORMAT(7E11.4)	RE200570

	WRITE (7, 119) II1, II2, II3, JJ1, JJ2	RE200571
119	FORMAT (5I7)	RE200572
	WRITE (7, 119) N3, M3	RE200573
	WRITE (7, 120) (R (J) , J=1, N3)	RE200574
120	FORMAT (10F8.4)	RE200575
	WRITE (7, 120) (Z (I) , I=1, M3)	RE200576
	WRITE (7, 118) TIMEX (L15)	RE200577
121	WRITE (6, 141) TIMEX (L15)	RE200578
	WRITE (6, 63) (R (J) , J=JJ1, JJ2)	RE200579
	JCNT=JJ2-JJ1+1	RE200580
	IF (JCNT.GT.9) GO TO 390	RE200581
	GO TO 391	RE200582
390	JJCNT=JCNT-9	RE200583
	JJJ2=JJ2-JJCNT	RE200584
	JJJ2P1=JJJ2+1	RE200585
391	DO 122 I=II1, II2	RE200586
	X1=Z (I) -Z (IPE) +DZ/2.	RE200587
	IF (JCNT.GT.9) GO TO 392	RE200588
	WRITE (6, 64) X1, (V (I, J) , J=JJ1, JJ2)	RE200589
	GO TO 393	RE200590
392	WRITE (6, 64) X1, (V (I, J) , J=JJ1, JJJ2)	RE200591
	WRITE (6, 64) X1, (V (I, J) , J=JJJ2P1, JJ2)	RE200592
393	IF (KTYPEO.EQ.1) GO TO 122	RE200593
	WRITE (7, 137) (V (I, J) , J=JJ1, JJ2)	RE200594
122	CONTINUE	RE200595
123	CONTINUE	RE200596
	RGV=0.	RE200597
	DO 395 LL15=1, KTYPE	RE200598
	IF (SAVRGV (LL15) .GT. RGV) RGV=SAVRGV (LL15)	RE200599
395	CONTINUE	RE200600
	WRITE (7, 396)	RE200601
396	FORMAT (22HMAX RGV CARD(S) FOLLOW)	RE200602
	DO 397 LL15=1, KTYPE	RE200603
397	WRITE (7, 137) RGV	RE200604
	GO TO 174	RE200605
124	FORMAT (1H , 5X, 1P9F13.6)	RE200606
137	FORMAT (6E13.6)	RE200607
139	FORMAT (1H0, 35HTEMPERATURE RISES AT SELECTED TIMES)	RE200608
141	FORMAT (1H0, 5X, 5HTIME=, E11.4)	RE200609
145	IF (L13.EQ.NTEST) GO TO 300	RE200610
	GO TO 371	RE200611
C ***	DAMAGE CALCULATIONS FOR SINGLE PULSE	RE200612
C	-----	RE200613
125	WRITE (6, 126) NRUN (1) , XDPULS, NPULSE (1)	RE200614
126	FORMAT (1H0, 5X, 5HNRUN=, I3, 2X, 12HPULSE WIDTH=, E10.3, 2X, 17HNUMBER OF	RE200615
	1PULSES=, I5)	RE200616
	IF (IPIL.EQ.0) GO TO 127	RE200617
	WRITE (6, 79) RIM, LESION	RE200618
	GO TO 128	RE200619
127	WRITE (6, 81) RIM, LESION	RE200620
128	IF (IPRT (8) .EQ.0) GO TO 150	RE200621
	WRITE (6, 375)	RE200622
	XQ=0.	RE200623
	DO 138 I=ID1, ID2	RE200624
	DO 138 J=1, JM	RE200625
	IF (VC (I, J, KM) .IT..001) QD (I, J) =1.0E+20	RE200626
	IF (VC (I, J, KM) .LT..001) GO TO 138	RE200627

L9=10.*(.4+EXP(-.0014*DPULSE))/VC(I,J,KM)	RE200628
CQ=L9+1.	RE200629
X10=70.*(.4+EXP(-.0014*DPULSE))/VC(I,J,KM)	RE200630
IF(L9.EQ.0)CQ=X10	RE200631
LLT=0	RE200632
LGT=0	RE200633
131 DAMC=0.	RE200634
K=2	RE200635
132 X13=ALOG(XT(K)-XT(K-1))	RE200636
VPX=(VC(I,J,K)+VC(I,J,K-1))/2.	RE200637
X3=0.	RE200638
X50=VPX*CQ+273.+T0	RE200639
IF(X50.LT.317.)GO TO 134	RE200640
X1=X13+DAMAGE(1,1)-DAMAGE(1,2)/X50	RE200641
IF(X50.GT.323.)X1=X13+DAMAGE(2,1)-DAMAGE(2,2)/X50	RE200642
IF(X1.GT.0.)X3=1.01	RE200643
IF(X1.GT.0.)GO TO 134	RE200644
X3=EXP(X1)	RE200645
134 DAMC=DAMC+X3	RE200646
IF(DAMC.GE.1.)GO TO 135	RE200647
K=K+1	RE200648
IF(K.LT.KT)GO TO 132	RE200649
C *** ADJUST LASEP POWER TO YIELD THRESHOLD DAMAGE AT GIVEN POINT	RE200650
IF(LGT.EQ.1)CQ=1.02*CQ	RE200651
IF(LGT.EQ.1)GO TO 136	RE200652
LLT=1	RE200653
CQ=1.04*CQ	RE200654
GO TO 131	RE200655
135 IF(LLT.EQ.1)CQ=.98*CQ	RE200656
IF(LLT.EQ.1)GO TO 136	RE200657
LGT=1	RE200658
CQ=.96*CQ	RE200659
GO TO 131	RE200660
136 QD(I,J)=CQ*POX	RE200661
138 CONTINUE	RE200662
WRITE(6,63)(R(J),J=1,JM)	RE200663
DO 140 I=ID1,ID2	RE200664
DO 140 J=1,JM	RE200665
140 XQD(I,J)=QD(I,J)*XXQ	RE200666
DO 143 I=ID1,ID2	RE200667
X1=Z(I)-Z(IPE)+DZ/2.	RE200668
IF(JM.GT.9)GO TO 142	RE200669
WRITE(6,105)X1,(XQD(I,J),J=1,JM)	RE200670
GO TO 143	RE200671
142 WRITE(6,105)X1,(XQD(I,J),J=1,9)	RE200672
WRITE(6,105)X1,(XQD(I,J),J=10,JM)	RE200673
143 CONTINUE	RE200674
150 IF(KTYPE.EQ.0)GO TO 174	RE200675
C *** CALCULATE AND STORE (SINGLE PULSE EXPOSURE) TEMPERATURES FOR	RE200676
C *** PLOTTING PROFILES	RE200677
WRITE(6,139)	RE200678
DO 170 L15=1,KTYPE	RE200679
RGV=0.	RE200680
DTIME=TIMEX(L15)	RE200681
K=ALOG(DTIME*(XC-1.)/DTX+1.)/ALOG(XC)+1.	RE200682
IF(K+1.GT.KT)GO TO 170	RE200683
X1=(DTIME-XT(K))/(XT(K+1)-XT(K))	RE200684

DO 166 I=II1,II2	RE200685
DO 166 J=JJ1,JJ2	RE200686
V(I,J)=VC(I,J,K)+X1*(VC(I,J,K+1)-VC(I,J,K))	RE200687
IF(V(I,J).GT.RGV)RGV=V(I,J)	RE200688
166 CONTINUE	RE200689
SAVRGV(LL15)=RGV	RE200690
IF(KTYPEO.EQ.1)GO TO 167	RE200691
WRITE(7,117)NRUN(1),NPULSE(1),FEPET(1)	RE200692
WRITE(7,118)XDPULS,WAVEL,RIM	RE200693
WRITE(7,119)II1,II2,II3,JJ1,JJ2	RE200694
WRITE(7,119)N3,M3	RE200695
WRITE(7,120)(R(J),J=1,N3)	RE200696
WRITE(7,120)(Z(I),I=1,M3)	RE200697
WRITE(7,118)TIMEX(LL15)	RE200698
167 WRITE(6,141)TIMEX(LL15)	RE200699
WRITE(6,63)(R(J),J=JJ1,JJ2)	RE200700
JCNT=JJ2-JJ1+1	RE200701
IF(JCNT.GT.9)GO TO 400	RE200702
GO TO 401	RE200703
400 JJCNT=JCNT-9	RE200704
JJJ2=JJ2-JJCNT	RE200705
JJJ2P1=JJJ2+1	RE200706
401 DO 168 I=II1,II2	RE200707
X1=Z(I)-Z(IPE)+DZ/2.	RE200708
IF(JCNT.GT.9)GO TO 402	RE200709
WRITE(6,64)X1,(V(I,J),J=JJ1,JJ2)	RE200710
GO TO 403	RE200711
402 WRITE(6,64)X1,(V(I,J),J=JJ1,JJJ2)	RE200712
WRITE(6,64)X1,(V(I,J),J=JJJ2P1,JJ2)	RE200713
403 IF(KTYPEO.EQ.1)GO TO 168	RE200714
WRITE(7,137)(V(I,J),J=JJ1,JJ2)	RE200715
168 CONTINUE	RE200716
170 CONTINUE	RE200717
RGV=0.	RE200718
DO 405 LL15=1,KTYPE	RE200719
IF(SAVRGV(LL15).GT.RGV)RGV=SAVRGV(LL15)	RE200720
405 CONTINUE	RE200721
WRITE(7,396)	RE200722
DO 406 LL15=1,KTYPE	RE200723
406 WRITE(7,137)RGV	RE200724
C *** INTERPOLATE AXIAL EXTENT OF DAMAGE	RE200725
174 I5=0	RE200726
I6=0	RE200727
IF(ID1.EQ.ID2)GO TO 182	RE200728
DO 175 I=ID1,ID2	RE200729
L1=ID1+ID2-I	RE200730
IF(QD(L1,1).GT.POX)I5=L1	RE200731
IF(QD(L1,1).LT.POX)I6=L1	RE200732
IF(QD(I,1).GT.POX)I7=I	RE200733
IF(QD(I,1).LT.POX)I8=I	RE200734
175 CONTINUE	RE200735
IF(IPRT(9).EQ.0)GO TO 182	RE200736
WRITE(6,350)	RE200737
350 FORMAT(1H0,22HAXIAL EXTENT OF DAMAGE)	RE200738
IF(I5.EQ.0)WRITE(6,176)	RE200739
176 FORMAT(1H0,5X,45HDEPTHS OF DAMAGE BEYOND BOTH SPECIFIED DEPTHS)	RE200740
IF(I5.EQ.0)GO TO 182	RE200741

```

IF (I6.EQ.0) GO TO 190
IF (I5.GE.I6) GO TO 178
X2=ALOG(QD(I6,1)/QD(I5,1))/(Z(I6)-Z(I5))
X1=QD(I5,1)
X3=ALOG(POX/X1)/X2+Z(I5)-Z(IPE)+DZ/2.
WRITE(6,177) X3
177 FORMAT(1H0,5X,24HMINIMUM DEPTH OF DAMAGE=,E10.3,2HCM)
178 IF (I8.GE.I7) GO TO 182
X2=ALOG(QD(I8,1)/QD(I7,1))/(Z(I8)-Z(I7))
X1=QD(I7,1)
X3=ALOG(POX/X1)/X2+Z(I7)-Z(IPE)+DZ/2.
180 WRITE(6,181) X3
181 FORMAT(1H0,5X,24HMAXIMUM DEPTH OF DAMAGE=,E10.3,2HCM)
C *** INTERPOLATE RADIAL EXTENT OF IRREVERSIBLE DAMAGE AT SPECIFIED
C *** DEPTHS
182 IF (IPRT(10).EQ.0) GO TO 192
WRITE(6,360)
360 FORMAT(1H0,23HRADIAL EXTENT OF DAMAGE)
DO 189 I=ID1,ID2
J1=0
X3=Z(I)-Z(IPE)+DZ/2.
DO 183 J=1,JM
IF (POX.GT.QD(I,J)) J1=J
183 CONTINUE
X20=0.
IF (J1.EQ.0) GO TO 187
IF (J1.EQ.JM) WRITE(6,185) X3,R(JM)
185 FORMAT(1H0,5X,2HZ=,E9.3,2HCM,5X,36HRADIAL EXTENT OF DAMAGE GREATER
1 THAN,E10.3,2HCM)
IF (J1.EQ.JM) GO TO 189
X2=ALOG(QD(I,J1+1)/QD(I,J1))/(R(J1+1)-R(J1))
X1=QD(I,J1)
X20=ALOG(POX/X1)/X2+R(J1)
187 WRITE(6,188) X3,X20
188 FORMAT(1H0,5X,2HZ=,E9.3,2HCM,5X,37HRADIAL EXTENT OF IRREVERSIBLE D
1 AMAGE=,E10.3,2HCM)
189 CONTINUE
IF (LPX.EQ.0) GO TO 300
GO TO 145
190 WRITE(6,191)
191 FORMAT(1H0,5X,31HNO DAMAGE---LASER POWER TOO LOW)
192 IF (LPX.EQ.0) GO TO 300
GO TO 145
200 STOP
END
SUBROUTINE GRID
C *** GRID COMPUTES THE COEFFICIENTS IN PARTIAL DIFFERENTIAL EQUATIONS ARE
C *** RADIAL AND AXIAL COORDINATES, R AND Z, AND ASSIGNS CONDUCTIVITY AN
C *** VOLUMETRIC SPECIFIC HEAT TO GRID
C *** CALCULATE B (CM**2) AND R (CM)
COMMON A(29,3),AAV,ACH,APE,ASC,ATS,AVL,B(14,3),BB,BV(14,3),
1CONX(6),CON(29),CUT,DFLOW(6),DPULSE,DR,DT,DTX,DZ,FL,HR(14),
2IAB(29,14),IBLOOD(10),IFIL,IGX,IHT,IPA,IPC,IPE,IPROP,IPS,IPT,
3IPV,IV(29),JVL,LIH,LPA,LPC,LPE,LPS,LPV,LPX,K,KH,KT,H,M1,M2,
4H3,N,N1,N3,N4,NVL,POX,PR(14),PTIME,QP,R(14),RCO,PIH,RN,RPE,RRT,
5RVL,RSC,S(29,14),SHB,TAV,TCH,TON,TPE,TVL,TSC,TTS,V(29,14)
6,VC(29,14,120),VSH(29),VSHX(6),WAVEL,XC,XFLOW,XFLOWI(6),XFLOWO(6),

```

```

7XT (120), Z (29), ZD (8), ZM, FLOWI (14), FLOWX (14), PUPIL, SIGMA,
8IPRT (10), APE1, APE2, RINT, ZO, FLO, CABER, CABER2, PP, PC, NB, NC, FC
DIMENSION IX (7), LX (7)
C *** CALCULATE B (CM**-2) AND R (CM)
WRITE (6, 170)
170 FORMAT (1H1)
R (1) = 0.
CK = N - N1
CP = RVL / DR - N1 + 1.
X1 = 2.
180 R2 = EXP (ALOG (2. * (CP * (X1 - 1.) + 1.) / (X1 + 1.)) / (CK - 1.))
IF (R2 / X1 .GT. .99999 .AND. R2 / X1 .LT. 1.00001) GO TO 181
X1 = R2
GO TO 180
181 IF (IPRT (1) .EQ. 0) GO TO 220
WRITE (6, 182)
182 FORMAT (1H0, 16HGRID INFORMATION)
WRITE (6, 184) R2
184 FORMAT (1H0, 5X, 3HR2 =, F8.4)
220 RN = DR * (N1 - 1. + (R2 ** (CK + 1.) - 1.) / (R2 - 1.))
C *** CALCULATE RADIAL SPACE STEPS R (J)
DO 185 J = 2, N4
185 R (J) = DR * (J - 1)
X1 = R2 * DR
DO 186 J = N4, N
R (J + 1) = R (J) + X1
186 X1 = R2 * X1
C *** CALCULATE COEFFICIENTS B OF FINITE DIFFERENCE EQNS.
X1 = 2. / (DR * DR)
DO 187 J = 2, N1
B (J, 1) = .25 * (2 * J - 3) * X1 / (J - 1)
B (J, 2) = X1
187 B (J, 3) = X1 - B (J, 1)
X2 = DR
X1 = R2 * DR
DO 188 J = N4, N
B (J, 2) = 2. / (X1 * X2)
B (J, 1) = (2. / X2 - 1. / R (J)) / (X1 + X2)
B (J, 3) = B (J, 2) - B (J, 1)
X2 = R2 * X2
188 X1 = R2 * X1
B (1, 1) = 0.
B (1, 2) = 2. / (DR * DR)
B (1, 3) = B (1, 2)
DO 189 J = 1, N
IF (R (J) .LT. RVL) JVL = J
189 CONTINUE
C *** CALCULATE AXIAL SPACE STEPS Z (I)
CK = M2 - M1 + 1
X1 = 2.
190 CP = 2. * TAV / DZ + 1. - (X1 ** (CK - 1.) - 1.) / (X1 - 1.)
R1 = EXP (ALOG (CP * X1 - CP + 1.) / CK)
IF (R1 / X1 .GT. .99999 .AND. R1 / X1 .LT. 1.00001) GO TO 192
X1 = R1
GO TO 190
192 ZM = ((R1 ** CK - 1.) / (R1 - 1.) + M1 - 1.) * DZ
IF (IPRT (1) .EQ. 0) GO TO 230

```

```

RE200799
RE200800
RE200801
RE200802
RE200803
RE200804
RE200805
RE200806
RE200807
RE200808
RE200809
RE200810
RE200811
RE200812
RE200813
RE200814
RE200815
RE200816
RE200817
RE200818
RE200819
RE200820
RE200821
RE200822
RE200823
RE200824
RE200825
RE200826
RE200827
RE200828
RE200829
RE200830
RE200831
RE200832
RE200833
RE200834
RE200835
RE200836
RE200837
RE200838
RE200839
RE200840
RE200841
RE200842
RE200843
RE200844
RE200845
RE200846
RE200847
RE200848
RE200849
RE200850
RE200851
RE200852
RE200853
RE200854
RE200855

```

WRITE (6,194) R1,ZM	RE200856
194 FORMAT (1H ,5X,3HR1=,F8.4,2X,3HZM=,F8.4)	RE200857
230 X1=DZ	RE200858
X2=X1	RE200859
DO 195 I=2,M2	RE200860
Z (M2+I)=ZM+X2	RE200861
Z (M2+2-I)=ZM-X2	RE200862
IF (I.GT.M1) X1=R1*X1	RE200863
195 X2=X2+X1	RE200864
Z (1)=0.	RE200865
Z (M2+1)=ZM	RE200866
Z (M+1)=2.*ZM	RE200867
X1=Z (IPE)-DZ/2.-ZD (2)	RE200868
DO 196 I=1,M3	RE200869
196 Z (I)=Z (I)-X1	RE200870
L3=IPA	RE200871
DO 200 L=1,7	RE200872
L1=0	RE200873
DO 197 I=IPA,M3	RE200874
IF (Z (I) .LT. ZD (L+1)) L3=I	RE200875
IF (Z (I) .LT. ZD (L) .OR. Z (I) .GE. ZD (L+1)) GO TO 197	RE200876
L2=I	RE200877
L1=L1+1	RE200878
197 CONTINUE	RE200879
IF (L1.EQ.0) IX (L) =L3	RE200880
IF (L1.EQ.0) LX (L) =L3	RE200881
IF (L1.GT.0) IX (L) =L2+1-L1	RE200882
IF (L1.GT.0) LX (L) =L2	RE200883
200 CONTINUE	RE200884
IPV=IX (4)	RE200885
IPC=IX (5)	RE200886
IPS=IX (6)	RE200887
IPT=IX (7)	RE200888
LPA=LX (1)	RE200889
LPE=LX (3)	RE200890
LPV=LX (4)	RE200891
LPC=LX (5)	RE200892
LPS=LX (6)	RE200893
LPT=M3	RE200894
C *** SET CONDUCTIVITY COM AND HEAT CAPACITY VSH FOR VARIOUS EYE MEDIA	RE200895
DO 203 I=1,LPA	RE200896
CON (I) =CONX (1)	RE200897
203 VSH (I) =VSHI (1)	RE200898
DO 204 I=IPE,LPE	RE200899
CON (I) =CONX (2)	RE200900
204 VSH (I) =VSHI (2)	RE200901
DO 205 I=IPV,LPV	RE200902
CON (I) =CONX (3)	RE200903
205 VSH (I) =VSHI (3)	RE200904
DO 206 I=IPC,LPC	RE200905
CON (I) =CONX (4)	RE200906
206 VSH (I) =VSHI (4)	RE200907
DO 207 I=IPS,LPS	RE200908
CON (I) =CONX (5)	RE200909
207 VSH (I) =VSHI (5)	RE200910
DO 208 I=IPT,M3	RE200911
CON (I) =CONX (6)	RE200912

208	VSH(I)=VSHX(6)	RE200913
C ***	CALCULATE COEFFICIENTS A OF FINITE DIFFERENCE EQNS.	RE200914
	DO 210 I=IPA,M	RE200915
	X1=Z(I+1)-Z(I-1)	RE200916
	X2=(CON(I-1)-CON(I+1))/(X1*X1)	RE200917
	X3=2.*CON(I)/X1	RE200918
	A(I,1)=X2+X3/(Z(I)-Z(I-1))	RE200919
	IF(I.EQ.IPA)A(I,1)=0.	RE200920
	A(I,3)=-X2+X3/(Z(I+1)-Z(I))	RE200921
210	A(I,2)=A(I,1)+A(I,3)	RE200922
	RETURN	RE200923
	END	RE200924
	SUBROUTINE IMAGE	RE200925
C ***	IMAGE COMPUTES THE RETINAL IRRADIANCE PROFILE	RE200926
	COMMON A(29,3),AAV,ACH,APE,ASC,ATS,AVL,B(14,3),BB,BV(14,3),	RE200927
	1CONX(6),CON(29),CUT,DFLOW(6),DPULSE,DR,DT,DTX,DZ,FL,HR(14),	RE200928
	2IAB(29,14),IBLOOD(10),IFIL,IGX,IHT,IPA,IPC,IPE,IPROP,IPS,IPT,	RE200929
	3IPV,IV(29),JVL,LIM,LPA,LPC,LPE,LPS,LPV,LPX,K,KH,KT,M,M1,M2,	RE200930
	4M3,N,N1,N3,N4,NVL,POX,PR(14),PTIME,QP,R(14),RCO,RIM,RN,RPE,RRT,	RE200931
	5RVL,RSC,S(29,14),SHB,TAV,TCH,TOM,TPE,TVL,TSC,TTS,V(29,14)	RE200932
	6,VC(29,14,120),VSH(29),VSHX(6),WAVEL,XC,XFLOW,XFLOWI(6),XFLOWO(6),	RE200933
	7XT(10),Z(29),ZD(8),ZH,FLOWI(14),FLOWX(14),PUPIL,SIGMA,	RE200934
	8IPRT(10),APE1,APE2,RINT,ZO,FLO,CABER,CABER2,PP,PC,NB,NC,FC	RE200935
	DIMENSION FA(2001),FP(2001),FX(2001),FY(2001),JO(32),NA(22),PX(30)	RE200936
	1,RX(30),XF1(2001),XF2(2001)	RE200937
	REAL JO,NA,NB,NC	RE200938
	DO 200 J=1,N	RE200939
200	PR(J)=0.	RE200940
	LI=500	RE200941
	LII=LI	RE200942
	DO 201 L=1,LI	RE200943
201	FX(L)=0.	RE200944
	READ(5,202)PUPIL	RE200945
202	FORMAT(10E8.3)	RE200946
	RINT=PUPIL/(LI-1)	RE200947
	IF(IPROP.EQ.1)GO TO 214	RE200948
	IF(IPROP.EQ.0)GO TO 219	RE200949
C ***	INTERPOLATE IRREGULAR LASER PROFILE (SYMMETRIC IN R) AT INTERVALS	RE200950
C ***	OF RINT STARTING AT R=0	RE200951
	READ(5,205)LF	RE200952
205	FORMAT(I7)	RE200953
	READ(5,206)(RX(L),L=1,LR)	RE200954
206	FORMAT(10E7.3)	RE200955
	READ(5,206)(PX(L),L=1,LR)	RE200956
	X1=PX(1)	RE200957
	DO 207 L=1,LR	RE200958
207	PX(L)=PX(L)/X1	RE200959
	X5=0.	RE200960
	X6=0.	RE200961
	DO 208 L=2,LR	RE200962
	X2=(PX(L)-PX(L-1))/(RX(L)-RX(L-1))	RE200963
	X1=PX(L-1)-X2*RX(L-1)	RE200964
	X3=X1*(RX(L)*RX(L)-RX(L-1)*RX(L-1))/2.	RE200965
	X4=X2*(RX(L)*RX(L)*RX(L)-RX(L-1)*RX(L-1)*RX(L-1))/3.	RE200966
	IF(RX(L).GT.PUPIL)X6=X6+6.2832*(X3+X4)	RE200967
208	X5=X5+6.2832*(X3+X4)	RE200968
	QP=POX*.23906*(1.-RCO)/X5	RE200969

```

XX=(X5-X6)/X5
IF (RX (LR) .LT. PUPIL) LII=RX (LR) /RINT+1
L2=2
X1=0.
DO 213 L=1,LII
210 IF (RX (L2) .GT. X1) GO TO 212
L2=L2+1
IF (L2.LE. LR) GO TO 210
GO TO 213
212 X2= (X1-RX (L2-1)) / (RX (L2) -RX (L2-1))
FX (L) =PX (L2-1) +X2*(PX (L2) -PX (L2-1))
213 X1=X1+RINT
GO TO 223
C *** CALCULATE GAUSSIAN LASER PROFILE AT INTERVALS OF RINT STARTING AT
214 SIGMA=RIM*SQRT (-2./ALOG (CUT))
QP=2.*POX*.23906*(1.-RCO)/(3.1416*SIGMA*SIGMA)
XX=1.-EXP (-2.*PUPIL*PUPIL/(SIGMA*SIGMA))
IF (IFIL.EQ.1) GO TO 217
DO 216 J=1,N
X3=2.*R (J)*R (J) / (SIGMA*SIGMA)
IF (X3.GT.80.) GO TO 216
PR (J) =EXP (-X3)
216 CONTINUE
GO TO 276
217 X1=0.
DO 218 L=1,LII
X3=2.*X1*X1/(SIGMA*SIGMA)
FX (L) =0.
IF (X3.GT.80.) GO TO 218
FX (L) =EXP (-X3)
218 X1=X1+RINT
GO TO 227
C *** SPECIFY UNIFORM LASER PROFILE FROM R (1) TO R (LIM)
219 QP=POX*.23906*(1.-RCO)/(3.1416*RIM*RIM)
XX=1.
IF (RIM.GT. PUPIL) XX=PUPIL*PUPIL/(RIM*RIM)
IF (IFIL.EQ.1) GO TO 221
DO 220 J=1,LIM
220 PR (J) =1.
GO TO 276
221 L1=RIM/RINT
RINT=RIM/L1
LII=RIM/RINT+1
DO 222 L=1,LII
222 FX (L) =1.
GO TO 227
C *** CALCULATE TOTAL AREA FA (L) AND PORTION OF LASERS POWER BETWEEN R=0
C *** AND (L-.5)*RINT
223 IF (IFIL.EQ.1) GO TO 227
FP (1) =3.1416*FX (1)*RINT*RINT/4.
FA (1) =3.1416*RINT*RINT/4.
DO 224 L=2,LII
X1=(L-.5)*RINT
X2=(L-1.5)*RINT
FP (L) =FP (L-1) +FX (L) *3.1416*(X1*X1-X2*X2)
224 FA (L) =FA (L-1) +3.1416*(X1*X1-X2*X2)
C *** CALCULATE PROFILE PR (J)

```

```

RE200970
RE200971
RE200972
RE200973
RE200974
RE200975
RE200976
RE200977
RE200978
RE200979
RE200980
RE200981
RE200982
RE200983
RE200984
RE200985
RE200986
RE200987
RE200988
RE200989
RE200990
RE200991
RE200992
RE200993
RE200994
RE200995
RE200996
RE200997
RE200998
RE200999
RE201000
RE201001
RE201002
RE201003
RE201004
RE201005
RE201006
RE201007
RE201008
RE201009
RE201010
RE201011
RE201012
RE201013
RE201014
RE201015
RE201016
RE201017
RE201018
RE201019
RE201020
RE201021
RE201022
RE201023
RE201024
RE201025
RE201026

```

X1=0.	RE201027
X2=0.	RE201028
DO 225 J=1,N	RE201029
X3=(R(J)+R(J+1))/(2.*RINT)+.5000001	RE201030
IF(X3.LT.1.)X3=1.000001	RE201031
L2=X3	RE201032
IF(L2.GE.LII)GO TO 225	RE201033
X4=X3-L2	RE201034
X5=FP(L2)+X4*(FP(L2+1)-FP(L2))	RE201035
X6=FA(L2)+X4*(FA(L2+1)-FA(L2))	RE201036
PR(J)=(X5-X1)/(X6-X2)	RE201037
X1=X5	RE201038
X2=X6	RE201039
225 CONTINUE	RE201040
GO TO 276	RE201041
C *** SPREAD FUNCTION CALCULATIONS	RE201042
227 READ(5,202)ZO,FLO,PC,NB,CABER,PP,PC	RE201043
CABER2=CABER/WAVEL	RE201044
READ(5,228)(JO(L),L=1,32)	RE201045
228 FORMAT(10F8.5)	RE201046
READ(5,228)(NA(L),L=1,22)	RE201047
X1=(WAVEL-350.)/50.+1.	RE201048
L1=X1	RE201049
X2=X1-L1	RE201050
NC=NA(L1)+X2*(NA(L1+1)-NA(L1))	RE201051
X1=(NB-1.)*NC/(NB*(NC-1.))	RE201052
FL=FLO*X1	RE201053
X2=ZO/FLO	RE201054
X0=NC*ZO*X1/(NC*X2-X1)-FLO	RE201055
X3=1.-PC*(NC*ZO-PC)/(NC*ZO*PC)	RE201056
DO 230 L=1,LI	RE201057
IF(L.GT.LII)GO TO 230	RE201058
X1=(L-1)/X3+1.000001	RE201059
L1=X1	RE201060
X2=X1-L1	RE201061
IF(L1+1.GT.LI)FY(L)=0.	RE201062
IF(L1+1.GT.LI)LII=L	RE201063
IF(L1+1.GT.LI)GO TO 230	RE201064
FX(L)=(FX(L1)+X2*(FX(L1+1)-FX(L1)))/(X3*X3)	RE201065
230 CONTINUE	RE201066
DO 231 L=1,LII	RE201067
231 FX(L)=FY(L)	RE201068
X5=ATAN(PUPIL/(FLO-PP+X0))	RE201069
X6=1.-COS(X5)	RE201070
X7=SIN(X5)*SIN(X5)	RE201071
FF=FLO-PP	RE201072
DO 234 L=1,LII	RE201073
X4=(L-1)*RINT	RE201074
X1=6.2832*NC*(-FF-X6*X0+SQRT(FF*FF-X7*X0*X0))*X4*X4/(WAVEL*1.E-7*	RE201075
1PUPIL*PUPIL)	RE201076
X2=CABER2*X4*X4*X4*X4	RE201077
XF1(L)=SQRT(FX(L))*COS(X1+X2)	RE201078
234 XF2(L)=SQRT(FX(L))*SIN(X1+X2)	RE201079
DO 260 J=1,N	RE201080
X1=6.2832*R(J)/(WAVEL*1.E-7*FF)	RE201081
X2=0.	RE201082
X3=0.	RE201083


```

DO 255 L=1,LII
X4=X1*(L-1)*RINT
IF (L.EQ.1) X4=X1*.25*RINT
IF (X4.GT.3.) GO TO 250
X5=X4/.1+1.000001
L1=X5
X5=X5-L1
X7=JO (L1) +X5*(JO (L1+1)-JO (L1))
GO TO 251
250 X6=3./X4
X8=.79788456-.0000077*X6-.00552740*X6*X6-.00009512*X6*X6*X6+
1.00137237*X6*X6*X6*X6-.00072805*X6*X6*X6*X6*X6+.00014476*X6*X6*X6*X6*
2X6*X6*X6
X9=X4-.78539816-.04166397*X6-.00003954*X6*X6+.00262573*X6*X6*X6-
1.00054125*X6*X6*X6*X6-.00029333*X6*X6*X6*X6*X6+.00013558*X6*X6*X6*X6*
2X6*X6*X6
X7=X8*COS (X9)/SQRT (X4)
251 IF (L.GT.1) GO TO 252
X2=X2+X7*.25*(3.*XF1 (1) +XF1 (2))*.25*RINT*.5*RINT
X3=X3+X7*.25*(3.*XF2 (1) +XF2 (2))*.25*RINT*.5*RINT
GO TO 255
252 X2=X2+X7*XF1 (L) * (L-1) *RINT*RINT
X3=X3+X7*XF2 (L) * (L-1) *RINT*RINT
255 CONTINUE
260 HR (J) =X2*X2+X3*X3
X1=HR (1)
DO 270 J=1,N
270 HR (J) =HR (J) /X1
X1=.0002
X2=3.1416*X1*X1/4
J=2
X4=HR (1) *X2
L1=2
271 IF (X1.LT.R (J) +.0000001) GO TO 272
J=J+1
GO TO 271
272 X5=(X1-R (J-1)) / (R (J) -R (J-1))
X6=HR (J-1) +X5*(HR (J) -HR (J-1))
X7=8.*(L1-1)*X2
X4=X4+X6*X7
L1=L1+1
X1=X1+.0002
IF (X1.LE..1) GO TO 271
QP=.23906*XX*POX*(1.-RCO) /X4
RETURN
276 DO 280 J=1,N
280 HR (J) =PR (J)
RETURN
END
SUBROUTINE HTXDEP
C *** HTXDEP COMPUTES RATE OF HEAT DEPOSITON AT VARIOUS POINTS I,J
COMMON A (29,3),AAV,ACH,APE,ASC,ATS,AVL,B (14,3),BB,BV (14,3),
1CONX (6),CON (29),CUT,DFLOW (6),DPULSE,DR,DT,DTX,DZ,FL,HR (14),
2IAB (29,14),IBLOOD (10),IFIL,IGX,IHT,IPA,IPC,IPE,IPOCF,IPS,IPT,
3IPV,IV (29),JVL,LIM,LPA,LPC,LPE,LPS,LPV,LPX,K,KM,KT,M,M1,M2,
4M3,N,N1,N3,N4,NVL,POX,PR (14),PTIME,QP,R (14),RCO,RIM,RN,RPE,RRT,
5RVL,RSC,S (29,14),SHB,TAV,TCH,TOM,TPE,TVL,TSC,TTS,V (29,14)

```

6, VC (29, 14, 120), VSH (29), VSHX (6), WAVEL, XC, XFLOW, XFLOWI (6), XFLOWO (6),	RE201141
7XT (120), Z (29), ZD (8), ZH, FLOWI (14), FLOWY (14), PUPIL, SIGMA,	RE201142
8IPRT (10), APE1, APE2, RINT, ZO, FLO, CABER, CABER2, PP, PC, NB, NC, FC	RE201143
DIMENSION AB (29, 3), ABR (29, 7), ABS (7), II (29), IZ (29), REF (8), REFL (8),	RE201144
1ZH (29)	RE201145
IF (IHT.EQ.0) RETURN	RE201146
IF (QP.LT.1.E-25) GO TO 340	RE201147
IF (IHT.EQ.1) RETURN	RE201148
LZ=7	RE201149
LZ0=LZ-1	RE201150
LZ1=LZ+1	RE201151
DO 280 I=1, M	RE201152
II (I)=0	RE201153
IZ (I)=0	RE201154
ZH (I) = (Z (I) + Z (I+1)) / 2.	RE201155
DO 279 L1=1, 3	RE201156
279 AB (I, L1) = 0.	RE201157
DO 280 L1=1, LZ	RE201158
280 ABR (I, L1) = 0.	RE201159
DO 282 L1=1, LZ	RE201160
REF (L1) = 0.	RE201161
282 REFL (L1) = 0.	RE201162
REF (2) = RRT	RE201163
REF (6) = RSC	RE201164
REF (LZ1) = 0.	RE201165
IF (IPRT (1).EQ.0) GO TO 350	RE201166
WRITE (6, 283) (ZH (I), I=1, M)	RE201167
283 FORMAT (1H0, 5X, 3HZH=/(1H, 5X, 10E10.3))	RE201168
C *** EVALUATE ABSORPTION CONSTANTS APE1 AND APE2 FOR FRONT AND REAR OF	RE201169
C *** PE	RE201170
350 IF (IGX.EQ.1) GO TO 284	RE201171
APE1 = (APE-ACH* (1.-RPE)) / RPE	RE201172
APE2 = ACH	RE201173
GO TO 285	RE201174
284 APE1 = ACH	RE201175
APE2 = (APE-ACH*RPE) / (1.-RPE)	RE201176
285 ABS (1) = AAV	RE201177
ABS (2) = APE1	RE201178
ABS (3) = APE2	RE201179
ABS (4) = AVL	RE201180
ABS (5) = ACH	RE201181
ABS (6) = ASC	RE201182
ABS (7) = ATS	RE201183
L1=2	RE201184
DO 306 I=IPA, M	RE201185
295 IF (ZH (I-1).LT.ZD (L1)) GO TO 296	RE201186
L1=L1+1	RE201187
GO TO 295	RE201188
296 IF (ZH (I).GE.ZD (L1)) GO TO 299	RE201189
C *** NO ZD BETWEEN ZH (I-1) AND ZH (I)	RE201190
AB (I, 1) = ABS (L1-1) * (ZH (I) - ZH (I-1))	RE201191
II (I) = 1	RE201192
IZ (I) = L1	RE201193
IF (L1.GT.LZ) GO TO 306	RE201194
DO 297 L2=L1, LZ	RE201195
297 ABR (I, L2) = AB (I, 1)	RE201196
GO TO 306	RE201197

299	IF (ZH (I) .GE. ZD (L1+1)) GO TO 303	RE201198
C ***	ONLY ZD (L1) BETWEEN ZH (I-1) AND ZH (I)	RE201199
	AB (I, 1) = ABS (L1-1) * (ZD (L1) - ZH (I-1))	RE201200
	AB (I, 2) = ABS (L1) * (ZH (I) - ZD (L1))	RE201201
	ABR (I, L1) = AB (I, 1)	RE201202
	II (I) = 2	RE201203
	IZ (I) = L1	RE201204
	L3 = L1 + 1	RE201205
	IF (L3 .GT. LZ) GO TO 306	RE201206
	DO 300 L2 = L3, LZ	RE201207
300	ABR (I, L2) = AB (I, 1) + AB (I, 2)	RE201208
	GO TO 306	RE201209
C ***	ZD (L1) AND ZD (L1+1) BETWEEN ZH (I-1) AND ZH (I)	RE201210
303	AB (I, 1) = ABS (L1-1) * (ZD (L1) - ZH (I-1))	RE201211
	AB (I, 2) = ABS (L1) * (ZD (L1+1) - ZD (L1))	RE201212
	AB (I, 3) = ABS (L1+1) * (ZH (I) - ZD (L1+1))	RE201213
	ABR (I, L1) = AB (I, 1)	RE201214
	ABR (I, L1+1) = AB (I, 1) + AB (I, 2)	RE201215
	II (I) = 3	RE201216
	IZ (I) = L1	RE201217
	L3 = L1 + 2	RE201218
	IF (L3 .GT. LZ) GO TO 306	RE201219
	DO 304 L2 = L3, LZ	RE201220
304	ABR (I, L2) = AB (I, 1) + AB (I, 2) + AB (I, 3)	RE201221
306	CONTINUE	RE201222
	DO 314 I = IPA, M	RE201223
	IF (AB (I, 1) .GT. 10.) AB (I, 1) = 10.	RE201224
	IF (AB (I, 2) .GT. 10.) AB (I, 2) = 10.	RE201225
	IF (AB (I, 3) .GT. 10.) AB (I, 3) = 10.	RE201226
	DO 314 L = 2, LZ	RE201227
	IF (ABR (I, L) .GT. 10.) ABR (I, L) = 10.	RE201228
314	CONTINUE	RE201229
C ***	DEPOSITION BY INCOMING BEAM	RE201230
	X2 = QP	RE201231
	L1 = 2	RE201232
	DO 317 I = IPA, M	RE201233
	L2 = II (I)	RE201234
	X3 = X2	RE201235
	X2 = X2 * EXP (-AB (I, 1))	RE201236
	X4 = 0.	RE201237
	IF (L2 .EQ. 1) GO TO 315	RE201238
	L3 = IZ (I)	RE201239
	X4 = X2 * REF (L3)	RE201240
	X2 = X2 * (1. - REF (L3)) * EXP (-AB (I, 2))	RE201241
	IF (L2 .EQ. 2) GO TO 315	RE201242
	X4 = X4 + X2 * REF (L3+1)	RE201243
	X2 = X2 * (1. - REF (L3+1)) * EXP (-AB (I, 3))	RE201244
315	IF (X2 .LT. 1.E-10) X2 = 0.	RE201245
	DO 317 J = 1, JVL	RE201246
	S (I, J) = (X3 - X2 - X4) * HR (J) / (ZH (I) - ZH (I-1))	RE201247
	IF (S (I, J) .LT. 1.E-10 / DPULSE) S (I, J) = 0.	RE201248
317	CONTINUE	RE201249
C ***	CALCULATION OF REFLECTED INTENSITIES BY VARIOUS INTERFACES	RE201250
C ***	STARTING WITH FIRST INTERNAL INTERFACE	RE201251
	X2 = QP	RE201252
	DO 322 L1 = 1, LZ0	RE201253
	X3 = ABS (L1) * (ZD (L1+1) - ZD (L1))	RE201254

	IF (X3.GT.10.) X3=10.	RE201255
	X2=X2*EXP (-X3)	RE201256
	REFL (L1+1)=X2*REF (L1+1)	RE201257
322	X2=X2*(1.-REF (L1+1))	RE201258
	DO 327 L1=2,LZ	RE201259
	I=IPA	RE201260
324	IF (ZH (I) .GT.ZD (L1)) GO TO 325	RE201261
	I=I+1	RE201262
	IF (I.LE.M) GO TO 324	RE201263
	GO TO 327	RE201264
325	X2=REFL (L1)	RE201265
	DO 326 L3=IPA,I	RE201266
	X3=X2	RE201267
	L4=I+IPA-L3	RE201268
	X2=X2*EXP (-ABR (L4,L1))	RE201269
	DO 326 J=1,JVL	RE201270
	S (L4,J)=S (L4,J) + (X3-X2)*HR (J) / (ZH (L4) -ZH (L4-1))	RE201271
	IF (S (L4,J) .LT.1.E-10/DPULSE) S (L4,J)=0.	RE201272
326	CONTINUE	RE201273
327	CONTINUE	RE201274
	IHT=1	RE201275
	RETURN	RE201276
C ***	NO HEAT DEPOSITION, BEAM OFF	RE201277
340	DO 342 I=1,M3	RE201278
	DO 342 J=1,N3	RE201279
342	S (I,J)=0.	RE201280
	IHT=0	RE201281
	RETURN	RE201282
	END	RE201283
	SUBROUTINE BLOOD	RE201284
C	SUBROUTINE BLOOD COMPUTES CHANGES IN MATRIX ELEMENTS A AND B DUE	RE201285
C	TO BLOOD FLOW	RE201286
	COMMON A (29,3), AAV, ACH, APE, ASC, ATS, AVL, B (14,3), BB, BV (14,3),	RE201287
	1CONX (6), CON (29), CUT, DFLOW (6), DPULSE, DR, DT, DTX, DZ, FL, HR (14),	RE201288
	2IAB (29,14), IBLOOD (10), IFIL, IGX, IHT, IPA, IPC, IPE, IPROP, IPS, IPT,	RE201289
	3IPV, IV (29), JVL, LIM, LPA, LPC, LPE, LPS, LPV, LPX, K, KM, KT, M, M1, M2,	RE201290
	4M3, N, N1, N3, N4, NVL, POX, PR (14), PTIME, QP, R (14), RCO, RIM, RN, RPE, RRT,	RE201291
	5RVL, RSC, S (29,14), SHB, TAV, TCH, TOM, TPE, TVL, TSC, TTS, V (29,14)	RE201292
	6, VC (29,14,120), VSH (29), VSHX (6), WAVE, XC, XFLOW, XFLOWI (6), XFLOWO (6),	RE201293
	7XT (120), Z (29), ZD (8), ZH, FLOWI (14), FLOWX (14), PUPIL, SIGMA,	RE201294
	8IPRT (10), APE1, APE2, RINT, ZO, FLO, CABER, CABER2, PP, PC, NB, NC, FC	RE201295
	DIMENSION RD (14), RH (14), XI (14), XO (14)	RE201296
C ***	INITIAL EVALUATION OF PARAMETERS AND ARRAYS	RE201297
	DO 800 J=1,N3	RE201298
	BV (J,1)=0.	RE201299
	BV (J,2)=0.	RE201300
	BV (J,3)=0.	RE201301
	FLOWI (J)=0.	RE201302
800	FLOWX (J)=0.	RE201303
	RH (1)=R (2)/2.	RE201304
	DO 803 J=2,JVL	RE201305
803	RH (J) = (R (J) +R (J+1)) /2.	RE201306
	L2=2	RE201307
	DO 810 J=1,JVL	RE201308
805	IF (DFLOW (L2) .GT.RH (J)) GO TO 806	RE201309
	L2=L2+1	RE201310
	GO TO 805	RE201311

806	X1=DFLOW(L2)-DFLOW(L2-1)	RE201312
	X2=RH(J)-DFLOW(L2-1)	RE201313
	X3=X2/X1	RE201314
	XI(J)=XFLOWI(L2-1)+X3*(XFLOWI(L2)-XFLOWI(L2-1))	RE201315
810	XO(J)=XFLOWO(L2-1)+X3*(XFLOWO(L2)-XFLOWO(L2-1))	RE201316
	FLOWX(1)=0.	RE201317
	DO 812 J=2,JVL	RE201318
812	FLOWX(J)=FLOWX(J-1)+(XI(J-1)-XO(J-1))*(R(J)*R(J)-R(J-1)*R(J-1))/	RE201319
	1(2.*TVL)	RE201320
	FLOWX(JVL+1)=FLOWX(JVL)	RE201321
	L2=2	RE201322
	FLOWI(1)=XFLOWI(1)/TVL	RE201323
	DO 820 J=2,JVL	RE201324
814	IF(DFLOW(L2).GT.R(J))GO TO 816	RE201325
	L2=L2+1	RE201326
	GO TO 814	RE201327
816	X4=DFLOW(L2)-DFLOW(L2-1)	RE201328
	X5=R(J)-DFLOW(L2-1)	RE201329
	X6=X5/X4	RE201330
820	FLOWI(J)=(XFLOWI(L2-1)+X6*(XFLOWI(L2)-XFLOWI(L2-1)))/TVL	RE201331
	DO 823 J=2,JVL	RE201332
823	RD(J)=1./(R(J)*(R(J+1)-R(J-1)))	RE201333
C ***	CALCULATE CHANGES IN MATRIX ELEMENTS A AND B DUE TO BLOOD FLOW	RE201334
	BV(1,1)=0.	RE201335
	BV(1,2)=-SHB*FLOWI(1)/2.	RE201336
	BV(1,3)=0.	RE201337
	BB=-SHB*XFLOW/2.	RE201338
	DO 825 J=2,JVL	RE201339
	BV(J,1)=SHB*RD(J)*FLOWX(J)	RE201340
	BV(J,2)=SHB*RD(J)*(FLOWX(J-1)-FLOWX(J+1))/2.-SHB*FLOWI(J)/2.	RE201341
825	BV(J,3)=-SHB*RD(J)*FLOWX(J)	RE201342
	DO 835 I=IPA,M	RE201343
835	IV(I)=0	RE201344
	DO 840 L3=1,NVL	RE201345
	L4=IBLOOD(L3)	RE201346
840	IV(L4)=1	RE201347
	DO 845 I=IPA,LPS	RE201348
	DO 842 J=1,JVL	RE201349
842	IAB(I,J)=0	RE201350
	IF(JVL.EQ.N)GO TO 845	RE201351
	L1=JVL+1	RE201352
	DO 843 J=L1,N	RE201353
843	IAB(I,J)=1	RE201354
845	CONTINUE	RE201355
	DO 850 I=IPT,M	RE201356
	DO 850 J=1,N	RE201357
850	IAB(I,J)=1	RE201358
	RETURN	RE201359
	END	RE201360

2694 RECORDS PRINTED

C		PLOTTING ROUTINE IITRI	PLT00001
C		VERSION 14 NOV 1975	PLT00002
C	TWO AND THREE DIMENSIONAL PLOTS		PLT00003
C			PLT00004
C	II1,II2	I VALUES DESIGNATING RANGE OF Z(I) VALUES FOR	PLT00005
C		PLOTTING RANGE=Z(II1) TO Z(II2)	PLT00006
C	II3	DESIGNATED PLANE OR SURFACE CURVE MARKED WITH	PLT00007
C		AN ASTERISK SYMBOL	PLT00008
C	JJ1,JJ2	J VALUES DESIGNATING RANGE OF R(J) VALUES FOR	PLT00009
C		PLOTTING RANGE=R(JJ1) TO R(JJ2)	PLT00010
C	R(J)	ORDINATE,CM	PLT00011
C	RGR	RANGE OF R VALUES TO BE PLOTTED,CM	PLT00012
C	RGV	RANGE OF TEMPERATURE VALUES TO BE PLOTTED,C	PLT00013
C	RGZ	RANGE OF Z VALUES TO BE PLOTTED,CM	PLT00014
C	TIMEX	TIME AT WHICH TEMPERATURE RISE VALUES ARE PLOTTED,SEC	PLT00015
C	V(I,J)	TEMPERATURE RISE AT TIME TIMEX(K),C	PLT00016
C	Z(I)	ABSCISSA,CM	PLT00017
C	REAL LA		PLT00018
	COMMON/PLBAS1/ P(4,3001),ICON(3001),NUM,NUMAX,IPLTX		PLT00019
	COMMON/PLBAS2/AP(16),AV(16),CP(16),DAT(8)		PLT00020
	DIMENSION LA(4)		PLT00021
	DIMENSION RR(100),PT(3),RP(100)		PLT00022
	DIMENSION R(14),V(29,14),Z(29)		PLT00023
	DATA LA/4HZ,CM,4HR,CM,4H T,C,4HRUN=/		PLT00024
	IPLTX=0		PLT00025
	5 DAT(1)=1.0		PLT00026
	IRR=0		PLT00027
	CALL SSPLIT		PLT00028
	READ(5,9,END=50)NRUN,NPULSE,REPET		PLT00029
	9 FORMAT(2I7,E10.4)		PLT00030
	READ(5,10)DPULSE,WAVEL,RIM		PLT00031
	10 FORMAT(3E11.4)		PLT00032
	READ(5,11)II1,II2,II3,JJ1,JJ2		PLT00033
	11 FORMAT(5I7)		PLT00034
	READ(5,11)N3,M3		PLT00035
	READ(5,12)(R(J),J=1,N3)		PLT00036
	12 FORMAT(10F8.4)		PLT00037
	READ(5,12)(Z(I),I=1,M3)		PLT00038
	READ(5,10)TIMEX		PLT00039
	DO 15 I=II1,II2		PLT00040
	READ(5,16)(V(I,J),J=JJ1,JJ2)		PLT00041
	15 CONTINUE		PLT00042
	16 FORMAT(6E13.6)		PLT00043
	READ(5,16)RGV		PLT00044
C	*** START OF PROGRAM FOR PLOTTING		PLT00045
	RGR=R(JJ2)-R(JJ1)		PLT00046
	RGZ=Z(II2)-Z(II1)		PLT00047
	SFLAG=0.		PLT00048
	SPAC=0.		PLT00049
	IF(RGV.LT.1.)GO TO 25		PLT00050
	SFLAG=1.		PLT00051
	IF((RGV.GE.12.).AND.(RGV.LT.112.))SPAC=10.		PLT00052
	IF((RGV.GE.112.).AND.(RGV.LT.1120.))SPAC=100.		PLT00053
	IF((RGV.GE.1120.).AND.(RGV.LT.11200.))SPAC=1000.		PLT00054
	IF(RGV.GE.11200.)SPAC=10000.		PLT00055
	IF(SFAC.EQ.0.)GO TO 26		PLT00056
	DO 14 I=II1,II2		PLT00057

DO 13 J=JJ1, JJ2	PLT00058
13 V(I, J)=V(I, J)/SPAC	PLT00059
14 CONTINUE	PLT00060
RGV=RGV/SPAC	PLT00061
GO TO 26	PLT00062
25 IF((RGV.LT.1.) .AND. (RGV.GE..1)) SPAC=10.	PLT00063
IF((RGV.LT..1) .AND. (RGV.GE..01)) SPAC=100.	PLT00064
IF((RGV.LT..01) .AND. (RGV.GE..001)) SPAC=1000.	PLT00065
IF((RGV.LT..001) .AND. (RGV.GE..0001)) SPAC=10000.	PLT00066
IF(RGV.LT..0001) SPAC=100000.	PLT00067
DO 18 I=II1, II2	PLT00068
DO 17 J=JJ1, JJ2	PLT00069
17 V(I, J)=V(I, J)*SPAC	PLT00070
18 CONTINUE	PLT00071
RGV=SPAC*RGV	PLT00072
26 WRITE(6, 19)	PLT00073
19 FORMAT(1H1, 3X, 21HSCIENTIFIC INPUT DATA)	PLT00074
WRITE(6, 21) RGZ, RGR, RGV	PLT00075
21 FORMAT(1H0, 4HRGZ=, E8.3, 2X, 4HRGR=, E8.3, 2X, 4HRGV=, E8.3)	PLT00076
IF(SFAC.EQ.0.) GO TO 28	PLT00077
IF(SFLAG.NE.0.) GO TO 8	PLT00078
WRITE(6, 7) SFAC	PLT00079
7 FORMAT(1H0, 30HTEMPERATURE RISES SCALED UP BY, F9.1)	PLT00080
GO TO 28	PLT00081
8 WRITE(6, 27) SFAC	PLT00082
27 FORMAT(1H0, 32HTEMPERATURE RISES SCALED DOWN BY, F9.1)	PLT00083
28 DO 23 I=II1, II2	PLT00084
WRITE(6, 22) I, (V(I, J), J=JJ1, JJ2)	PLT00085
22 FORMAT(1H0, 2HI=, I3/(1X, 10F10.5))	PLT00086
23 CONTINUE	PLT00087
WRITE(6, 24)	PLT00088
24 FORMAT(1H0, 3X, 35HAXIS INFORMATION (SYSTEM GENERATED)/)	PLT00089
C *** PLOT ROUTINE	PLT00090
30 CONTINUE	PLT00091
C	PLT00092
C----- SET UP FOR PLOT	PLT00093
C	PLT00094
IDIF=II2-II1+1	PLT00095
JDIF=JJ2-JJ1+1	PLT00096
NM=1	PLT00097
DO 100 N=1, IDIF	PLT00098
DO 100 M=1, JDIF	PLT00099
I1=II1+N-1	PLT00100
J1=JJ1+M-1	PLT00101
P(1, NM)=R(J1)	PLT00102
P(2, NM)=Z(I1)	PLT00103
P(3, NM)=V(I1, J1)	PLT00104
ICON(NM)=10	PLT00105
IF(M.NE.1) ICON(NM)=0	PLT00106
NM=N+1	PLT00107
100 CONTINUE	PLT00108
DO 200 M=1, JDIF	PLT00109
DO 200 N=1, IDIF	PLT00110
J1=JJ1+M-1	PLT00111
I1=II1+N-1	PLT00112
P(1, NM)=R(J1)	PLT00113
P(2, NM)=Z(I1)	PLT00114

```

P(3,NM)=V(I1,J1)
ICON(NM)=10
IF(N.NE.1) ICON(NM)=0
NM=NM+1
200 CONTINUE
NUMAX=3000
NUM=NM-1
CALL POLSUR(JDIF, IDIF)
DO 150 MM=1, JDIF
M=JJ1+MM-1
NUM=NUM+1
P(1,NUM)=R(M)
P(2,NUM)=Z(II1)
P(3,NUM)=0.0
ICON(NUM)=10
NUM=NUM+1
P(1,NUM)=R(M)
P(2,NUM)=Z(II1)
P(3,NUM)=V(II1,M)
ICON(NUM)=0
150 CONTINUE
DO 160 MM=1, JDIF
M=JJ1+MM-1
NUM=NUM+1
P(1,NUM)=R(M)
P(2,NUM)=Z(II2)
P(3,NUM)=0.0
ICON(NUM)=10
NUM=NUM+1
P(1,NUM)=R(M)
P(2,NUM)=Z(II2)
P(3,NUM)=V(II2,M)
ICON(NUM)=0
160 CONTINUE
DO 170 NN=1, IDIF
NUM=NUM+1
N=NN+II1-1
P(1,NUM)=R(JJ2)
P(2,NUM)=Z(N)
P(3,NUM)=0.0
ICON(NUM)=10
NUM=NUM+1
P(1,NUM)=R(JJ2)
P(2,NUM)=Z(N)
P(3,NUM)=V(N, JJ2)
ICON(NUM)=0
170 CONTINUE
NUM=NUM+1
P(1,NUM)=R(JJ2)
P(2,NUM)=Z(II3)
P(3,NUM)=V(II3, JJ2)
P(4,NUM)=11.
ICON(NUM)=31
NUM=NUM+1
P(1,NUM)=R(JJ1)
P(2,NUM)=Z(II1)-RGZ*0.25
P(3,NUM)=RGV*0.5

```

```

PLT00115
PLT00116
PLT00117
PLT00118
PLT00119
PLT00120
PLT00121
PLT00122
PLT00123
PLT00124
PLT00125
PLT00126
PLT00127
PLT00128
PLT00129
PLT00130
PLT00131
PLT00132
PLT00133
PLT00134
PLT00135
PLT00136
PLT00137
PLT00138
PLT00139
PLT00140
PLT00141
PLT00142
PLT00143
PLT00144
PLT00145
PLT00146
PLT00147
PLT00148
PLT00149
PLT00150
PLT00151
PLT00152
PLT00153
PLT00154
PLT00155
PLT00156
PLT00157
PLT00158
PLT00159
PLT00160
PLT00161
PLT00162
PLT00163
PLT00164
PLT00165
PLT00166
PLT00167
PLT00168
PLT00169
PLT00170
PLT00171

```



```

P (4, NUM) = LA (3)
ICON (NUM) = 32
NUM = NUM + 1
P (1, NUM) = R (JJ2) + RGR * 0.1
P (2, NUM) = Z (II1) + RGZ * 0.5
P (3, NUM) = 0.0
P (4, NUM) = LA (1)
ICON (NUM) = 32
NUM = NUM + 1
P (1, NUM) = R (JJ1) + RGR * 0.5
P (2, NUM) = Z (II1) - RGZ * 0.1
P (3, NUM) = 0.0
P (4, NUM) = LA (2)
ICON (NUM) = 32
CALL SYMCON (.07, 4, -1.1, -1.2)
NUM = NUM + 1
P (1, NUM) = R (JJ2)
P (2, NUM) = Z (II1)
P (3, NUM) = 0.0
P (4, NUM) = R (JJ2)
ICON (NUM) = 33
C----- X-AXIS AT Y=Z (II1)
RP (1) = JDIF
DO 300 KK = 1, JDIF
IJ = KK * 2
JK = JJ1 + KK - 1
RP (IJ) = R (JK)
IJ = IJ + 1
RP (IJ) = -1
300 CONTINUE
PRINT 398
398 FORMAT (10X, ' R-AXIS')
PRINT 399, (RP (LL), LL = 1, IJ)
399 FORMAT (5X, 10F10.4)
PT (1) = R (JJ1)
PT (2) = Z (II1)
PT (3) = 0
LAB = 1
CALL AXES (RP, PT, LAB, 2, 1)
C----- X-AXIS AT Y=Z (II2)
PT (1) = R (JJ1)
PT (2) = Z (II2)
PT (3) = 0
LAB = 1
CALL AXES (RP, PT, LAB, 2, 2)
CALL SYMCON (0.07, 4, 1.1, -1.2)
C----- Y-AXIS AT X=R (JJ1)
RP (1) = IDIF
DO 301 KK = 1, IDIF
IJ = KK * 2
JK = II1 + KK - 1
RP (IJ) = Z (JK)
IJ = IJ + 1
RP (IJ) = -1
301 CONTINUE
PRINT 397
397 FORMAT (10X, ' Z-AXIS')

```

```

PLT00172
PLT00173
PLT00174
PLT00175
PLT00176
PLT00177
PLT00178
PLT00179
PLT00180
PLT00181
PLT00182
PLT00183
PLT00184
PLT00185
PLT00186
PLT00187
PLT00188
PLT00189
PLT00190
PLT00191
PLT00192
PLT00193
PLT00194
PLT00195
PLT00196
PLT00197
PLT00198
PLT00199
PLT00200
PLT00201
PLT00202
PLT00203
PLT00204
PLT00205
PLT00206
PLT00207
PLT00208
PLT00209
PLT00210
PLT00211
PLT00212
PLT00213
PLT00214
PLT00215
PLT00216
PLT00217
PLT00218
PLT00219
PLT00220
PLT00221
PLT00222
PLT00223
PLT00224
PLT00225
PLT00226
PLT00227
PLT00228

```

```

PRINT 399, (RP(LL),LL=1,IJ)
PT(1)=R(JJ1)
PT(2)=Z(II1)
PT(3)=0
LAB=2
CALL AXES(RP,PT,LAB,2,2)
C----- Y-AXIS AT TOP OF V
PT(1)=R(JJ1)
PT(2)=Z(II1)
PT(3)=RGV
LAB=2
CALL AXES(RP,PT,LAB,2,1)
C----- Y-AXIS AT X=R(JJ2)
PT(1)=R(JJ2)
PT(2)=Z(II1)
PT(3)=0
RP(3)=1
RP(IJ)=1
LAB=2
CALL AXES(RP,PT,LAB,2,1)
C----- Z-AXIS AT X=R(JJ1), Y=Z(II1)
CALL SYMCON(0.07,1,-1.1,1.2)
RP(1)=RGV+1.
RR(1)=RGV+1
II=RR(1)+1
DO 302 KK=1,II
IJ=KK*2
RP(IJ)=KK-1
RR(IJ)=KK-1
IJ=IJ+1
RR(IJ)=(-1)**(KK+1)
RP(IJ)=-1.
302 CONTINUE
PRINT 396
396 FORMAT(10X,' V-AXIS')
PRINT 399, (RR(LL),LL=1,IJ)
PT(1)=R(JJ1)
PT(2)=Z(II1)
PT(3)=0
LAB=3
CALL AXES(RR,PT,LAB,2,1)
PT(1)=R(JJ1)
PT(2)=Z(II2)
PT(3)=0.
LAB=3
CALL AXES(RP,PT,LAB,2,1)
PRINT 400
400 FORMAT(1H0,3X,37HTHREE DIMENSIONAL POINTS IN PLOT FILE/1H0,6X,
15HPOINT,23X,1HR,14X,1HZ,14X,1HV)
DO 299 LL=1,NUM
PRINT 199,LL,ICON(LL),P(1,LL),P(2,LL),P(3,LL)
199 FORMAT(5X,I5,5X,I5,5X,3F15.4)
299 CONTINUE
C
C----- END OF PLOT SETUP
C
WRITE(6,34)

```

```

PLT00229
PLT00230
PLT00231
PLT00232
PLT00233
PLT00234
PLT00235
PLT00236
PLT00237
PLT00238
PLT00239
PLT00240
PLT00241
PLT00242
PLT00243
PLT00244
PLT00245
PLT00246
PLT00247
PLT00248
PLT00249
PLT00250
PLT00251
PLT00252
PLT00253
PLT00254
PLT00255
PLT00256
PLT00257
PLT00258
PLT00259
PLT00260
PLT00261
PLT00262
PLT00263
PLT00264
PLT00265
PLT00266
PLT00267
PLT00268
PLT00269
PLT00270
PLT00271
PLT00272
PLT00273
PLT00274
PLT00275
PLT00276
PLT00277
PLT00278
PLT00279
PLT00280
PLT00281
PLT00282
PLT00283
PLT00284
PLT00285

```

34	FORMAT(1H0,3X,43HSUMMARY OF ADDITIONAL SCIENTIFIC INPUT DATA)	PLT00286
	WRITE(6,35) WAVEL,NPULSE	PLT00287
35	FORMAT(1H0,11HWAVELENGTH=,E9.4,2HNM,8X,17HNUMBER OF PULSES=,I5)	PLT00288
	WRITE(6,36) DPULSE,RIM	PLT00289
36	FORMAT(1H0,12HPULSE WIDTH=,E9.4,3HSEC,10X,13HIMAGE RADIUS=,E9.4,	PLT00290
	12HCM)	PLT00291
	WRITE(6,37) REPET	PLT00292
37	FORMAT(1H0,16HREPETITION RATE=,E9.4,10HPULSES/SEC)	PLT00293
	WRITE(6,40)	PLT00294
40	FORMAT(1H0,17HAXIAL DISTANCE,CM)	PLT00295
	WRITE(6,41)	PLT00296
41	FORMAT(1HC,18HRADIAL DISTANCE,CM)	PLT00297
	WRITE(6,42)	PLT00298
42	FORMAT(1H0,25HTEMPERATURE RISE,DEGREE C)	PLT00299
	WRITE(6,43) TIMEX,NRUN	PLT00300
43	FORMAT(1H0,27HTEMPERATURE RISE PROFILE AT,E9.4,9HSEC (RUN=,I4,1H))	PLT00301
	WRITE(6,44)	PLT00302
44	FORMAT(1H0,3X,17HPLOT COMMAND LIST/)	PLT00303
	CALL PLOT(12.,-11.,-3)	PLT00304
	CALL PLOT(0.,.5,-3)	PLT00305
	HT=.14	PLT00306
	A=TIMEX	PLT00307
	B=NRUN	PLT00308
	CALL SYMBOL(0.,1.,HT,29H TEMPERATURE RISE PROFILE AT ,0.,29)	PLT00309
	XX=29*HT	PLT00310
	CALL FNUM(XX,1.,A,12)	PLT00311
	XX=XX+16*HT	PLT00312
	CALL SYMBOL(XX,1.,HT,13HSEC -- RUN = ,0.0,13)	PLT00313
	XX=XX+13*HT	PLT00314
	CALL NUMBER(XX,1.,HT,B,0.0,0)	PLT00315
	IF(SFAC.EQ.0.)GO TO 45	PLT00316
	FPN=SFAC	PLT00317
	IF(SFLAG.EQ.0.)FPN=1./SFAC	PLT00318
	CALL SYMBOL(0.,.75,.1,29H ORIGINAL T,C = PLOTTED T,C* ,0.,29)	PLT00319
	CALL NUMBER(2.9,.75,.1,FPN,0.,5)	PLT00320
45	CALL READIN(IRR)	PLT00321
	IF(IRR.EQ.1)GO TO 50	PLT00322
	GO TO 5	PLT00323
50	CALL PLOT(12.,0.,999)	PLT00324
	STOP	PLT00325
	END	PLT00326
	SUBROUTINE POLSUR(M,N)	PLT00327
	COMMON/PLBAS1/ P(4,3001),ICON(3001),NUM,NUMAX,IPLTX	PLT00328
	DIMENSION W(3,500)	PLT00329
	NCT=0	PLT00330
	DO 10 I=1,N	PLT00331
	DO 10 J=1,M	PLT00332
	NCT=NCT+1	PLT00333
	DO 10 L=1,3	PLT00334
	W(L,NCT)=P(L,NCT)	PLT00335
10	CONTINUE	PLT00336
	NUM=0	PLT00337
	DO 20 N1=1,N	PLT00338
	NLO=N1	PLT00339
	MM=M-1	PLT00340
	DO 20 M1=1,MM	PLT00341
		PLT00342

MLO=M1	PLT00343
NUM=NUM+1	PLT00344
NA=M1+N1*M-M	PLT00345
CALL EQUIV(P(1,NUM),W(1,NA))	PLT00346
ICON(NUM)=0	PLT00347
IF(M1.EQ.1) ICON(NUM)=10	PLT00348
NUM=NUM+1	PLT00349
NA=M1+1+(N1-1)*M	PLT00350
CALL EQUIV(P(1,NUM),W(1,NA))	PLT00351
ICON(NUM)=0	PLT00352
NUM=NUM+1	PLT00353
ISIGN=1	PLT00354
IF(N1.NE.1) ISIGN=-1	PLT00355
NA=MLO+(NLO-1)*M+1	PLT00356
NB=NA-1	PLT00357
NC=NA+ISIGN*M	PLT00358
ISIGN=-ISIGN	PLT00359
CALL PCROSS(W(1,NA),W(1,NB),W(1,NC),P(1,NUM),ISIGN)	PLT00360
ICON(NUM)=50	PLT00361
20 CONTINUE	PLT00362
DO 30 M1=1,M	PLT00363
MLO=M1	PLT00364
NN=N-1	PLT00365
DO 30 N1=1,NN	PLT00366
NLO=N1	PLT00367
NUM=NUM+1	PLT00368
NA=M1+(N1-1)*M	PLT00369
CALL EQUIV(P(1,NUM),W(1,NA))	PLT00370
ICON(NUM)=0	PLT00371
IF(N1.EQ.1) ICON(NUM)=10	PLT00372
NUM=NUM+1	PLT00373
NA=M1+(N1-1)*M+M	PLT00374
CALL EQUIV(P(1,NUM),W(1,NA))	PLT00375
ICON(NUM)=0	PLT00376
NUM=NUM+1	PLT00377
ISIGN=1	PLT00378
IF(M1.EQ.M) ISIGN=-1	PLT00379
NA=MLO+(NLO-1)*M+M	PLT00380
NB=NA+ISIGN	PLT00381
NC=NA-M	PLT00382
ISIGN=-ISIGN	PLT00383
CALL PCROSS(W(1,NA),W(1,NB),W(1,NC),P(1,NUM),ISIGN)	PLT00384
ICON(NUM)=50	PLT00385
30 CONTINUE	PLT00386
RETURN	PLT00387
END	PLT00388
SUBROUTINE PCROSS(PA,PB,PC,V,IS)	PLT00389
DIMENSION PA(3),PB(3),PC(3),V(3)	PLT00390
DIMENSION VX(3),VY(3)	PLT00391
DO 10 I=1,3	PLT00392
VX(I)=PB(I)-PA(I)	PLT00393
VY(I)=PC(I)-PA(I)	PLT00394
10 CONTINUE	PLT00395
V(1)=VX(2)*VY(3)-VX(3)*VY(2)	PLT00396
V(2)=-VX(1)*VY(3)-VX(3)*VY(1)	PLT00397
V(3)=VX(1)*VY(2)-VX(2)*VY(1)	PLT00398
SUM=0.0	PLT00399

DO 20 I=1,3	PLT00400
20 SUM=SUM+V(I)*V(I)	PLT00401
SUM=SQRT(SUM)+1.0E-20	PLT00402
DO 30 I=1,3	PLT00403
30 V(I)=IS*V(I)/SUM	PLT00404
RETURN	PLT00405
END	PLT00406
SUBROUTINE EQUIV(PA,PB)	PLT00407
DIMENSION PA(3),PB(3)	PLT00408
DO 10 I=1,3	PLT00409
10 PA(I)=PB(I)	PLT00410
RETURN	PLT00411
END	PLT00412
SUBROUTINE SYMCON(HH,NN,XX,YY)	PLT00413
COMMON/PLBAS1/ P(4,3001),ICON(3001),NUM,NUMAX,IPLTX	PLT00414
NUM=NUM+1	PLT00415
DO 10 I=1,3	PLT00416
10 P(I,NUM)=0.0	PLT00417
P(4,NUM)=HH	PLT00418
ICON(NUM)=71	PLT00419
NUM=NUM+1	PLT00420
DO 20 I=1,3	PLT00421
20 P(I,NUM)=0.0	PLT00422
P(4,NUM)=NN	PLT00423
ICON(NUM)=72	PLT00424
NUM=NUM+1	PLT00425
DO 30 I=1,3	PLT00426
30 P(I,NUM)=0.0	PLT00427
P(4,NUM)=XX	PLT00428
ICON(NUM)=73	PLT00429
NUM=NUM+1	PLT00430
DO 40 I=1,3	PLT00431
40 P(I,NUM)=0.0	PLT00432
P(4,NUM)=YY	PLT00433
ICON(NUM)=74	PLT00434
RETURN	PLT00435
END	PLT00436
SUBROUTINE READIN(IRR)	PLT00437
COMMON/PLBAS1/ P(4,3001),ICON(3001),NUM,NUMAX,IPLTX	PLT00438
COMMON/PLBAS2/ AP(16),AV(16),CP(16),DAT(8)	PLT00439
COMMON/PLBAS3/ WINXL,WINYL,WINXW,WINYW,IWIN	PLT00440
COMMON/PLBAS4/ SCRNXL,SCRNYL,SCRNXW,SCRNYW,ISCRN	PLT00441
COMMON/PLBAS5/ SIGNOR,SNPLOT,IH	PLT00442
DIMENSION NAM(21)	PLT00443
DATA NAM/ 4HP ,4HINIT,4HROLL,4HPITC,4HYAW ,	PLT00444
X 4HSCAL,4HTRAN,4HDIST,4HREIN,4HHIDE,	PLT00445
X 4HSIGN,4HWIND,4HSCRN,4HBOX ,4HFACT,	PLT00446
X 4HPLOT,4HUSER,4HPRIN,4HEND ,4HDUM ,	PLT00447
X 4HAXIS /	PLT00448
DATA NONAM/21/	PLT00449
EQUIVALENCE (DAT(1),RDAR(1))	PLT00450
DIMENSION RDAR(8)	PLT00451
DIMENSION P(4),RMN(3),RMX(3),PT(3)	PLT00452
IPRIN=0	PLT00453
1 READ(5,10,END=999) NAMM,(RDAR(L),L=2,8)	PLT00454
10 FORMAT(A4,6X,7F10.4)	PLT00455
IF(IPRIN.GT.0) GO TO 41	PLT00456

	WRITE(6,40) NAMM, (RDAR(L),L=2,8)	PLT00457
40	FORMAT(1X,A4,6X,7F10.4)	PLT00458
41	CONTINUE	PLT00459
C---	COMPARE TO PRESTORED NAMES IN ORDER TO DETERMINE THE ACTION CODE	PLT00460
	DO 20 I=1,NONAM	PLT00461
	IF(NAMM.EQ.NAM(I)) GO TO 30	PLT00462
20	CONTINUE	PLT00463
C---	ERROR PATH -- INPUT WORD WAS NOT VALID	PLT00464
	IRR=1	PLT00465
	WRITE(6,50) NAMM,NAM	PLT00466
50	FORMAT(/,' ERROR -- THE CODE NAME ',A5,1X,' WAS NOT VALID, VALID	PLT00467
	X NAMES ARE AS FOLLOWS',/,20(1X,A4))	PLT00468
	GO TO 999	PLT00469
30	CONTINUE	PLT00470
	IF(I.EQ.1) GO TO 100	PLT00471
	IF(I.GT.1.AND.I.LT.17) GO TO 120	PLT00472
	IK=I-16	PLT00473
	GO TO (170,180,190,200,210),IK	PLT00474
100	CONTINUE	PLT00475
	IF(RDAR(2).LT.-0.1.OR.RDAR(2).GT.99.) GO TO 110	PLT00476
	NUM=NUM+1	PLT00477
	DO 111 L=1,4	PLT00478
111	P(L,NUM)=RDAR(L+2)	PLT00479
	ICON(NUM)=RDAR(2)	PLT00480
	GO TO 1	PLT00481
110	CONTINUE	PLT00482
	NUM=RDAR(3)	PLT00483
	GO TO 1	PLT00484
120	CONTINUE	PLT00485
	RDAR(1)=I-1	PLT00486
	CALL SSPLIT	PLT00487
	GO TO 1	PLT00488
170	CONTINUE	PLT00489
	RDAR(1)=17	PLT00490
	CALL USER	PLT00491
	GO TO 1	PLT00492
180	CONTINUE	PLT00493
	IPRIN=RDAR(2)	PLT00494
	GO TO 1	PLT00495
190	CONTINUE	PLT00496
	GO TO 999	PLT00497
200	CONTINUE	PLT00498
	WRITE(6,201) NUM,NUMAX	PLT00499
201	FORMAT(5X,'CURRENT NUMBER OF POINTS= ',I6,' AND MAXIMUM ALLOWED= '	PLT00500
	X,I6)	PLT00501
	NUM1=MIN0(NUMAX,NUM)	PLT00502
	IF(NUM1.LE.0) GO TO 1	PLT00503
	WRITE(6,205)	PLT00504
205	FORMAT(1X,10HCOORDINATE,10H LOW VAL ,10H HI VAL ,	PLT00505
	X 10H MEAN VAL ,10H WIDTH)	PLT00506
	DO 202 J=1,3	PLT00507
	RMIN=1.0E+20	PLT00508
	RMAX=-1.0E+20	PLT00509
	DO 203 L=1,NUM	PLT00510
	IF(ICON(L).GE.49) GO TO 203	PLT00511
	RMIN=AMIN1(RMIN,P(J,L))	PLT00512
	RMAX=AMAX1(RMAX,P(J,L))	PLT00513

203	CONTINUE	PLT00514
	RMEAN=(RMAX+RMIN)/2.0	PLT00515
	DIF=RMAX-RMIN	PLT00516
	WRITE(6,204) J,RMIN,RMAX,RMEAN,DIF	PLT00517
204	FORMAT(1X,'COORD ',I2,1X,4F10.3)	PLT00518
202	CONTINUE	PLT00519
	GO TO 1	PLT00520
210	CONTINUE	PLT00521
	IF(NUM.LE.0) GO TO 999	PLT00522
	DO 211 J=1,3	PLT00523
	RMIN=1.0E+20	PLT00524
	RMAX=-RMIN	PLT00525
	DO 212 I=1,NUM	PLT00526
	IF(ICON(I).GT.49) GO TO 212	PLT00527
	RMIN=AMIN1(RMIN,P(J,I))	PLT00528
	RMAX=AMAX1(RMAX,P(J,I))	PLT00529
212	CONTINUE	PLT00530
	RMX(J)=RMAX	PLT00531
	RMN(J)=RMIN	PLT00532
	PT(J)=(RMIN+RMAX)/2.0	PLT00533
	IF(RDAR(2).GT.0.1) PT(J)=RMAX	PLT00534
	IF(RDAR(2).LT.-0.1) PT(J)=RMIN	PLT00535
211	CONTINUE	PLT00536
	DO 213 J=1,3	PLT00537
	IF(RMX(J)-RMN(J).LT.0.0001) GO TO 213	PLT00538
	IF(RMX(J)-RMN(J).GT.1.0E+20) GO TO 213	PLT00539
	R(1)=RMN(J)	PLT00540
	R(2)=(RMX(J)-RMN(J))/5.0	PLT00541
	R(3)=6.0	PLT00542
	R(4)=5.0	PLT00543
	LAB=J	PLT00544
	CALL AXES(R,PT,LAB,1)	PLT00545
213	CONTINUE	PLT00546
	GO TO 1	PLT00547
999	CONTINUE	PLT00548
	RETURN	PLT00549
	END	PLT00550
	SUBROUTINE AXES(R,PT,LAB,MODE,NCON)	PLT00551
	COMMON/PLBAS1/ P(4,3001),ICON(3001),NUM,NUMAX,IPLTX	PLT00552
	DIMENSION R(1),T(102)	PLT00553
	DATA NT/100/	PLT00554
	DIMENSION PT(3)	PLT00555
	DATA BIG/1.0E+20/	PLT00556
C---	OBJECTIVE OF ROUTINE IS TO GENERATE AXIS DATA IN THE THREE	PLT00557
C---	DIMENSIONAL POINT DATA BASE	PLT00558
C---	INPUT IS THRU CALLING ARGUMENTS AS FOLLOWS	PLT00559
C---	LAB SHOULD BE 1 2 OR 3 DENOTING X, Y OR Z AXIS INFORMATION	PLT00560
C---	IF MODE IS 1 THEN R(1,2,3 AND 4) DENOTE RESPECTIVELY THE START,	PLT00561
C---	INCREMENT,NUMBER OF INCREMENTS AND INCREMENT FOR NUMBERING	PLT00562
C---	MODE=2 MEANS THAT THE TICK DATA IS STORED IN THE ARRAY R SO THAT	PLT00563
C---	R(1) IS THE NUMBER OF POINTS, R(2) IS THE VALUE FOR THE FIRST,	PLT00564
C---	MARK, R(3) IS POSITIVE IF A NUMBER SHOULD BE PLOTTED, AND NEGATIVE	PLT00565
C---	OTHERWISE AND SO ON	PLT00566
C---	IN THE CASE OF EACH MODE, TICK DATA IS BUILT INTO THE LOCAL ARRAY	PLT00567
C---	T AS A BUFFER, AND THEN TRANSFERRED TO THE POINT ARRAY	PLT00568
	GO TO (10,20),MODE	PLT00569
10	CONTINUE	PLT00570

START=R(1)	PLT00571
AINC=R(2)	PLT00572
NO=R(3)	PLT00573
IVINC=R(4)	PLT00574
IRR=1	PLT00575
IF(NO.LE.0) GO TO 998	PLT00576
IRR=2	PLT00577
IF(NO.GT.NT/2) GO TO 998	PLT00578
T(1)=NO	PLT00579
SMIN=BIG	PLT00580
SMAX=-BIG	PLT00581
DO 11 I=1,NO	PLT00582
T(2*I)=START+(I-1)*AINC	PLT00583
T(2*I+1)=-1	PLT00584
SMIN=AMIN1(T(2*I),SMIN)	PLT00585
SMAX=AMAX1(T(2*I),SMAX)	PLT00586
IF(IVINC.LE.0) GO TO 10	PLT00587
IF(MOD(I,IVINC).EQ.1) T(2*I+1)=1.0	PLT00588
11 CONTINUE	PLT00589
GO TO 100	PLT00590
20 CONTINUE	PLT00591
NO=R(1)	PLT00592
IRR=3	PLT00593
IF(NO.LE.0) GO TO 998	PLT00594
IRR=4	PLT00595
IF(NO.GT.NT/2) GO TO 998	PLT00596
SMIN=BIG	PLT00597
SMAX=-BIG	PLT00598
DO 21 I=1,NO	PLT00599
T(2*I)=R(2*I)	PLT00600
T(2*I+1)=R(2*I+1)	PLT00601
21 CONTINUE	PLT00602
100 CONTINUE	PLT00603
JTEM=NUM	PLT00604
DO 110 I=1,NO	PLT00605
JTEM=JTEM+1	PLT00606
DO 120 J=1,3	PLT00607
120 P(J,JTEM)=PT(J)	PLT00608
P(4,JTEM)=LAB	PLT00609
P(LAB,JTEM)=T(2*I)	PLT00610
IF(I.EQ.1) ICON(JTEM)=NCON*10+1	PLT00611
IF(I.NE.1) ICON(JTEM)=1	PLT00612
110 CONTINUE	PLT00613
NUM=NUM+NO	PLT00614
JTEM=NUM	PLT00615
DO 130 I=1,NO	PLT00616
IF(T(2*I+1).LT.0.0) GO TO 130	PLT00617
NUM=NUM+1	PLT00618
JTEM=JTEM+1	PLT00619
DO 140 J=1,3	PLT00620
140 P(J,JTEM)=PT(J)	PLT00621
P(LAB,JTEM)=T(2*I)	PLT00622
ICON(JTEM)=33.	PLT00623
P(4,JTEM)=T(2*I)	PLT00624
130 CONTINUE	PLT00625
999 IRR=0	PLT00626
RETURN	PLT00627


```

998 WRITE(6,997) IRR                                PLT00628
997 FORMAT(/,'  ERROR IN AXES ROUTINE, IRR= ',I6,/) PLT00629
RETURN                                               PLT00630
END                                                  PLT00631
SUBROUTINE SSPLIT                                  PLT00632
COMMON/PLBAS1/ P(4,3001),ICON(3001),NUM,NUMAX,IPLTX PLT00633
COMMON/PLBAS2/ AP(16),AV(16),CP(16),DAT(8)         PLT00634
COMMON/PLBAS3/ WINXL,WINYL,WINXW,WINYW,IWIN        PLT00635
COMMON/PLBAS4/ SCRNXL,SCRNYL,SCRNXW,SCRNYW,SCRNZW,ISCRN PLT00636
COMMON/PLBAS5/ SIGNOR,SNPLOT,IH                   PLT00637
COMMON/PLBAS6/DIMAGE,DORIG,DOBY,DOBY              PLT00638
COMMON/PLBAS7/HT,NDECFX,XLATE,YLATE               PLT00639
C--- AP,AV ARE PROJECTIVE NON SINGULAR MATRICES WHICH RECORD THE PLT00640
C--- CURRENT POSITION OF THE POINT SET              PLT00641
C--- IH THE HIDDEN LINE FLAG                       PLT00642
C--- ZVIEW IS DISTANCE OF VIEWERS EYE FROM PROJECTION(XY) PLANE PLT00643
C--- DAT CONTAINS THE COMMAND DATA FOR EXECUTING PIECES OF THIS ROUTINE PLT00644
C--- SIGNOR THE SIGN APPLIED TO THE SURFACE NORMALS PLT00645
C--- P CONTAINS XYZ DATA OF POINTS,VECTORS AND SYMBOL DATA IN 4TH PLC PLT00646
C--- ICON CONTAINS TWO PACKED DIGITS AB WITH THE FOLLOWING MEANING PLT00647
C--- A=0, CONTINUE PRESENT MODE OF PLOTTING, A=1 START CONNECTING POINTS PLT00648
C--- BY STRAIGHT LINES, A=2 CONNECT PTS BY DASHED LINES, A=4 PLOT POINTS PLT00649
C--- S ONLY, A=4 PLOT DASHED POINTS               PLT00650
C--- B=0 PLOT NO SYMBOL, B=1 PLOT CENTERED SYMBOL WHOSE VALUE IS P(4,) PLT00651
C--- PLOT LITERAL STRING IN FIELD P(4,) B=3 PLOT NUMBER IN FIELD P(4,) PLT00652
C--- SET UP WINDOW PARAMETERS                     PLT00653
DATA SMALL/1.0E-10/,SMAL/1.0E-8/                 PLT00654
DIMENSION AID(16),TP(16),BP(16)                  PLT00655
DIMENSION RWID(3),RCEN(3),RMIN(3),RMAX(3)         PLT00656
DIMENSION PP(3),VV(3)                             PLT00657
DATA AID/1.0,4*0.0,1.0,4*0.0,1.0,4*0.0,1.0/     PLT00658
IT=DAT(1)                                          PLT00659
GO TO (10,20,30,40,50,60,70,80,90,100,110,120,130,140,150),IT PLT00660
C--- IT=1 INITIALIZE KEY VARIABLES WITH DEFAULT VALUES PLT00661
10 SIGNOR=1.0                                      PLT00662
NUM=0                                              PLT00663
IPRIN=0                                           PLT00664
HT=0.07                                           PLT00665
SWIDTH=8.25                                       PLT00666
SHEIGHT=6.5                                       PLT00667
ISCRN=-1                                          PLT00668
IWIN=-1                                           PLT00669
SCRNXL=0.0                                        PLT00670
SCRNYL=0.0                                        PLT00671
SCRNXW=8.5                                        PLT00672
SCRNYW=6.25                                       PLT00673
SCRNZW=SCRNXW                                     PLT00674
SXUNIT=1024.                                       PLT00675
SYUNIT=760.0                                       PLT00676
IH=0                                              PLT00677
ZVIEW=0.0                                         PLT00678
NERASE=0                                          PLT00679
NDECFX=-1                                         PLT00680
XLATE=-1.1                                        PLT00681
YLATE=-1.1                                        PLT00682
IF(IPLTX.GT.0)GO TO 12                            PLT00683
CALL PLOTS(0,0,8)                                  PLT00684

```

IPLTX=1	PLT00685
12 DO 11 I=1,16	PLT00686
BP(I)=AID(I)	PLT00687
AP(I)=AID(I)	PLT00688
11 AV(I)=AID(I)	PLT00689
BP(11)=0.0	PLT00690
C--- REPLACE INCREMENTAL VALUES WITH ABSOLUTE VALUES	PLT00691
NUMAX=3000	PLT00692
DO 13 L=1,NUMAX	PLT00693
DO 14 K=1,4	PLT00694
14 P(K,L)=0.0	PLT00695
ICON(L)=0	PLT00696
13 CONTINUE	PLT00697
DOBX=0.0	PLT00698
DOBY=0.0	PLT00699
GO TO 999	PLT00700
C--- 20,30 AND 40 ARE ROTATION COMMANDS	PLT00701
C--- IT=2 XYROT OR ROLL	PLT00702
20 DAT(1)=1.0	PLT00703
CALL PERSPT(DAT,TP)	PLT00704
CALL MMULT(AP,TP,CP,1)	PLT00705
CALL MMULT(AV,TP,CP,1)	PLT00706
GO TO 999	PLT00707
C--- IT=3 YZROT OR PITCH	PLT00708
30 DAT(1)=2.0	PLT00709
CALL PERSPT(DAT,TP)	PLT00710
CALL MMULT(AP,TP,CP,1)	PLT00711
CALL MMULT(AV,TP,CP,1)	PLT00712
GO TO 999	PLT00713
C--- IT=4 ZXROT OR YAW	PLT00714
40 DAT(1)=3	PLT00715
CALL PERSPT(DAT,TP)	PLT00716
CALL MMULT(AP,TP,CP,1)	PLT00717
CALL MMULT(AV,TP,CP,1)	PLT00718
GO TO 999	PLT00719
C--- IT=5 SCALE	PLT00720
50 DAT(1)=4	PLT00721
CALL PERSPT(DAT,TP)	PLT00722
CALL MMULT(AP,TP,CP,1)	PLT00723
GO TO 999	PLT00724
C--- IT=6 TRANSLATION	PLT00725
60 DAT(1)=5	PLT00726
CALL PERSPT(DAT,TP)	PLT00727
CALL MMULT(AP,TP,CP,1)	PLT00728
GO TO 999	PLT00729
C--- IT=7 SETUP PROJECTION ONTO XYPLAN FROM VIEWERS POSITION	PLT00730
70 DAT(1)=6	PLT00731
ZVIEW=DAT(2)	PLT00732
DIMAGE=DAT(2)	PLT00733
DORIG=DAT(3)	PLT00734
DOBX=DAT(4)	PLT00735
DOBY=DAT(5)	PLT00736
CALL PERSPT(DAT,BP)	PLT00737
GO TO 999	PLT00738
C--- REIDENTIFY THE TRANSFORMATION MATRICES	PLT00739
80 DO 81 I=1,16	PLT00740
AP(I)=AID(I)	PLT00741

81	AV(I)=AID(I)	PLT00742
	GO TO 999	PLT00743
C---	SETUP THE HIDDEN LINE FLAG	PLT00744
90	IH=DAT(2)	PLT00745
	GO TO 999	PLT00746
100	SIGNOR=DAT(2)	PLT00747
	GO TO 999	PLT00748
110	CONTINUE	PLT00749
	IWIN=-1	PLT00750
	IF(DAT(2)**2+DAT(3)**2+DAT(4)**2+DAT(5)**2.LT.SMAL) GO TO 999	PLT00751
	IWIN=1	PLT00752
	WINXL=DAT(2)	PLT00753
	WINYL=DAT(3)	PLT00754
	WINXW=DAT(4)	PLT00755
	WINYW=DAT(5)	PLT00756
	GO TO 999	PLT00757
C---	SCREEN PARAMETERS INTRODUCED	PLT00758
120	CONTINUE	PLT00759
	ISCRN=-ISCRN	PLT00760
	IF(DAT(2)**2+DAT(3)**2+DAT(4)**2+DAT(5)**2.LT.SMAL) GO TO 999	PLT00761
	SCRNXL=DAT(2)	PLT00762
	SCRNYL=DAT(3)	PLT00763
	SCRNXW=DAT(4)	PLT00764
	SCRNYW=DAT(5)	PLT00765
	SCRNZW=DAT(6)	PLT00766
	ISCRN=1	PLT00767
	GO TO 999	PLT00768
C---	BOX COMMAND, SCALE THE OBJECT TO FILL THE SCREEN	PLT00769
130	CONTINUE	PLT00770
	IF(ISCRN.LT.0) GO TO 999	PLT00771
	PROA=DAT(2)	PLT00772
	PROB=DAT(3)	PLT00773
	PROC=DAT(4)	PLT00774
C---	DETERMINE THE XYZ EXTENT OF THE TRANSFORMED OBJECT	PLT00775
	DO 131 L=1,3	PLT00776
	RMIN(L)=1.0E+20	PLT00777
131	RMAX(L)=-1.0E+20	PLT00778
	I=0	PLT00779
137	I=I+1	PLT00780
	IF(I.GT.NUM) GO TO 138	PLT00781
	CALL DECOD(PP,VV,AA,JCON,ISYM,IVEC,I)	PLT00782
	IF(IVEC.EQ.999) GO TO 137	PLT00783
	IF(I.LT.0) GO TO 999	PLT00784
	WW=PP(1)*AP(13)+PP(2)*AP(14)+PP(3)*AP(15)+AP(16)+SMALL	PLT00785
	DO 132 L=1,3	PLT00786
	L4=L*4	PLT00787
	PPP=(PP(1)*AP(L4-3)+PP(2)*AP(L4-2)+PP(3)*AP(L4-1)+AP(L4))/WW	PLT00788
	RMIN(L)=AMIN1(PPP,RMIN(L))	PLT00789
	RMAX(L)=AMAX1(PPP,RMAX(L))	PLT00790
132	CONTINUE	PLT00791
	GO TO 137	PLT00792
138	CONTINUE	PLT00793
	DO 133 L=1,3	PLT00794
	RCEN(L)=(RMIN(L)+RMAX(L))/2.0	PLT00795
	RWID(L)=RMAX(L)-RMIN(L)+SMALL	PLT00796
	DAT(L+1)=-RCEN(L)	PLT00797
133	CONTINUE	PLT00798

C---	CENTERISE THE OBJECT AROUND THE ORIGIN	PLT00799
	DAT(1)=5	PLT00800
	CALL PERSPT(DAT,TP)	PLT00801
	CALL MMULT(AP,TP,CP,1)	PLT00802
C---	SCALE THE OBJECT INTO THE SCREEN AREA OR WINDOW AREA IF REQUESTED	PLT00803
	IF (ISCRN.LE.0) GO TO 999	PLT00804
	A=1.0E+20	PLT00805
	SX=SCRNXW/RWID(1)*PROA	PLT00806
	IF (PROB.GT.0.0) GO TO 135	PLT00807
	A=SCRNYW/RWID(2)*PROA	PLT00808
	SX=AMIN1(SX,A)	PLT00809
	DAT(2)=SX	PLT00810
	DAT(3)=SX	PLT00811
	DAT(4)=SX	PLT00812
	GO TO 136	PLT00813
135	CONTINUE	PLT00814
	SY=SCRNYW/RWID(2)*PROB	PLT00815
	DAT(2)=SX	PLT00816
	DAT(3)=SY	PLT00817
	DAT(4)=1.0	PLT00818
	IF (PROC.GT.0.0) DAT(4)=SCRNZW/RWID(3)*PROC	PLT00819
136	CONTINUE	PLT00820
	DAT(1)=4	PLT00821
	CALL PERSPT(DAT,TP)	PLT00822
	CALL MMULT(AP,TP,CP,1)	PLT00823
	CALL MMULT(AV,TP,CP,1)	PLT00824
	IF (IWIN.LE.0) GO TO 999	PLT00825
C---	APPLY A FURTHER TRANSLATION AND SCALE IF WINDOW IS IN EFFECT	PLT00826
	DAT(1)=5.	PLT00827
	DAT(2)=- (WINXL+WINXW/2.0)	PLT00828
	DAT(3)=- (WINYL+WINYW/2.0)	PLT00829
	DAT(4)=0.0	PLT00830
	CALL PERSPT(DAT,TP)	PLT00831
	CALL MMULT(AP,TP,CP,1)	PLT00832
	DAT(2)=SCRNXW/WINXW	PLT00833
	DAT(3)=SCRNYW/WINYW	PLT00834
	DAT(2)=AMIN1(DAT(2),DAT(3))	PLT00835
	DAT(3)=DAT(2)	PLT00836
	DAT(4)=DAT(2)	PLT00837
	DAT(4)=1.0	PLT00838
	DAT(1)=4.0	PLT00839
	CALL PERSPT(DAT,TP)	PLT00840
	CALL MMULT(AP,TP,CP,1)	PLT00841
	CALL MMULT(AV,TP,CP,1)	PLT00842
	WINXW=SCRNXW	PLT00843
	WINYW=SCRNYW	PLT00844
	WINXL=SCRNXL	PLT00845
	WINYL=SCRNYL	PLT00846
	GO TO 999	PLT00847
C---	APPLY A STRAIGHT FACTOR TO ALL SUBSEQUENT PLTS	PLT00848
140	CONTINUE	PLT00849
	IF (DAT(2).LE.SMAL) GO TO 999	PLT00850
	CALL FACTOR(DAT(2))	PLT00851
	GO TO 999	PLT00852
C---	MAIN PLOT PROCESSING IS HERE	PLT00853
150	CONTINUE	PLT00854
	IF (DAT(4).LT.0.0) CALL PLOT(0.,0.,-3)	PLT00855

IF (DAT (4) .LT.0.0) CALL PLOT (DAT (2) ,DAT (3) ,999)	PLT00856
IF (DAT (4) .LT.0.0) GO TO 999	PLT00857
CALL PLOT (DAT (2) ,DAT (3) ,-3)	PLT00858
151 CONTINUE	PLT00859
CALL MMULT (AP,BP,CP,3)	PLT00860
C--- SETUP THE WINDOW,SCREEN AND PLOT BOUNDARIES	PLT00861
IF (IWIN.LE.0.AND.ISCRN.LE.0) GO TO 154	PLT00862
IF (ISCRN.GT.0) GO TO 153	PLT00863
IF (IWIN.LE.0) GO TO 154	PLT00864
XL=WINXL	PLT00865
YL=WINYL	PLT00866
XW=WINXW	PLT00867
YW=WINYW	PLT00868
GO TO 152	PLT00869
153 XL=SCRNXL	PLT00870
YL=SCRNYL	PLT00871
XW=SCRNXW	PLT00872
YW=SCRNYW	PLT00873
152 CONTINUE	PLT00874
IF (DAT (2) **2+DAT (3) **2.GT.SMAL) CALL PLOT (XL+XW/2.0,YL+YW/2.0,3)	PLT00875
CALL PLOT (XL,YL,3)	PLT00876
CALL PLOT (XL+XW,YL,2)	PLT00877
CALL PLOT (XL+XW,YL+YW,2)	PLT00878
CALL PLOT (XL,YL+YW,2)	PLT00879
CALL PLOT (XL,YL,2)	PLT00880
154 CONTINUE	PLT00881
MOVNOW=0	PLT00882
IF (ISCRN.GT.0.OR.IWIN.GT.0) CALL WINDOW (XL,YL,XW,YW,MOVNOW)	PLT00883
XLAS=0.0	PLT00884
YLAS=0.0	PLT00885
IPERM=0	PLT00886
NPLT=0	PLT00887
I=0	PLT00888
301 I=I+1	PLT00889
IF (I.GT.NUM) GO TO 302	PLT00890
C--- MAIN PLOTTING LOOP	PLT00891
X1=XLAS	PLT00892
Y1=YLAS	PLT00893
C--- DECODE THE NECESSARY POINT AND AUXILIARY DATA	PLT00894
IA=I	PLT00895
CALL DECOD (PP,VV,AA,JCON,ISYM,IVEC,IA)	PLT00896
IF (IVEC.EQ.999) GO TO 301	PLT00897
IF (IA.LE.0) GO TO 300	PLT00898
I=IA	PLT00899
IF (JCON*(5-JCON) .NE.0) IPERM=JCON	PLT00900
IF (IPERM.EQ.0) GO TO 300	PLT00901
WNOW=PP (1) *CP (13) +PP (2) *CP (14) +PP (3) *CP (15) +CP (16) +SMALL	PLT00902
XNOW= (PP (1) *CP (1) +PP (2) *CP (2) +PP (3) *CP (3) +CP (4)) /WNOW	PLT00903
YNOW= (PP (1) *CP (5) +PP (2) *CP (6) +PP (3) *CP (7) +CP (8)) /WNOW	PLT00904
X2=XNOW	PLT00905
Y2=YNOW	PLT00906
MOVNOW=2	PLT00907
IF (IWIN.LT.0) GO TO 310	PLT00908
C--- MAKE THE REQUIRED WINDOW CHECK	PLT00909
MOVNOW=1	PLT00910
CALL WINDOW (X1,Y1,X2,Y2,MOVNOW)	PLT00911
310 CONTINUE	PLT00912

	IF (MOVNOW.LT.0) GO TO 600	PLT00913
	IF (IH.EQ.0.OR.IVEC.LT.1) GO TO 320	PLT00914
C---	MAKE THE HIDDEN LINE/SURFACE NORMAL CHECK	PLT00915
	VXNOW=VV (1)*AV (1)+VV (2)*AV (2)+VV (3)*AV (3)	PLT00916
	VYNOW=VV (1)*AV (5)+VV (2)*AV (6)+VV (3)*AV (7)	PLT00917
	VZNOW=VV (1)*AV (9)+VV (2)*AV (10)+VV (3)*AV (11)	PLT00918
	PXNOW=PP (1)*AP (1)+PP (2)*AP (2)+PP (3)*AP (3)+AP (4)	PLT00919
	PYNOW=PP (1)*AP (5)+PP (2)*AP (6)+PP (3)*AP (7)+AP (8)	PLT00920
	PZNOW=PP (1)*AP (9)+PP (2)*AP (10)+PP (3)*AP (11)+AP (12)	PLT00921
	IF (ABS (BP (15)).LT.0.0001) GO TO 330	PLT00922
	ZVIEW=-BP (16)/BP (15)	PLT00923
	D=(PXNOW-DOBX)*VXNOW+(PYNOW-DOBY)*VYNOW+(PZNOW-ZVIEW)*VZNOW	PLT00924
	D=D*SIGNOR	PLT00925
	IHCUP=0	PLT00926
	PRINT 311	PLT00927
311	FORMAT (' PXNOW,PYNOW,PZNOW,VXNOW,VYNOW,VZNOW,DOBX,DOBY,ZVIEW,D')	PLT00928
	WRITE (6,312) I,	PLT00929
X	PXNOW,PYNOW,PZNOW,VXNOW,VYNOW,VZNOW,DOBX,DOBY,ZVIEW,D	PLT00930
312	FORMAT (1X,I4,3 (3 (1X,F9.3)),2X,F9.3)	PLT00931
	IF (D.GT.0.0) IHCUR=1	PLT00932
	GO TO 340	PLT00933
330	IHCUR=0	PLT00934
	D=VZNOW*SIGNOR	PLT00935
	IF (D.LT.0.0) IHCUR=1	PLT00936
340	CONTINUE	PLT00937
320	CONTINUE	PLT00938
	IPERMN =IPERM	PLT00939
	IF (IH.EQ.0.OR.IVEC.LT.1) GO TO 350	PLT00940
	IF (IHCUR.EQ.0) GO TO 350	PLT00941
	IF (IH.EQ.2) GO TO 360	PLT00942
C---	TOTALLY HIDDEN LINE	PLT00943
	IPERMN=0	PLT00944
	GO TO 350	PLT00945
360	CONTINUE	PLT00946
	IF (IPERM.EQ.1) IPERMN =2	PLT00947
	IF (IPERM.EQ.2) IPERMN =4	PLT00948
350	CONTINUE	PLT00949
	IF (IPERMN .EQ.0) GO TO 600	PLT00950
	IF ((IPERMN-2)*(IPERMN-4).EQ.0.AND.JCON.EQ.0) GO TO 370	PLT00951
	NDASH=1	PLT00952
	UX=X2-X1	PLT00953
	UY=Y2-Y1	PLT00954
	GO TO 380	PLT00955
370	CONTINUE	PLT00956
	D=SQRT ((X2-X1)**2+(Y2-Y1)**2)	PLT00957
	NDASH=D/0.25	PLT00958
	NDASH=MAX0 (3,NDASH)	PLT00959
	D1=D/NDASH	PLT00960
	UX=(X2-X1)/(D+SMALL)*D1	PLT00961
	UY=(Y2-Y1)/(D+SMALL)*D1	PLT00962
C---	POSITION POINT AT START OF SEGMENT	PLT00963
	IF (MOVNOW.EQ.3.OR.MOVNOW.EQ.5) CALL PLOT (X1,Y1,3)	PLT00964
	IF (MOVNOW.EQ.3.OR.MOVNOW.EQ.5) NPLT=NPLT+1	PLT00965
380	CONTINUE	PLT00966
	IF (IPERMN .GT.2) GO TO 420	PLT00967
	MODO=-1	PLT00968
	DO 410 J=1,NDASH	PLT00969

XX=X1+UX*J	PLT00970
YY=Y1+UY*J	PLT00971
MODO=-MODO	PLT00972
IPLT=2	PLT00973
IF(MODO.LT.0) IPLT=3	PLT00974
IF(JCON.NE.0) IPLT=3	PLT00975
NPLT=NPLT+1	PLT00976
CALL PLOT(XX,YY,IPLT)	PLT00977
410 CONTINUE	PLT00978
GO TO 500	PLT00979
420 DO 430 J=1,NDASH	PLT00980
XX=X1+UX*J	PLT00981
YY=Y1+UY*J	PLT00982
CALL PLOT(XX,YY,3)	PLT00983
CALL PLOT(XX,YY,2)	PLT00984
NPLT=NPLT+1	PLT00985
430 CONTINUE	PLT00986
GO TO 500	PLT00987
500 CONTINUE	PLT00988
IF(MOVNOW.EQ.4.OR.MOVNOW.EQ.5) GO TO 590	PLT00989
IF(ISYM.EQ.0) GO TO 590	PLT00990
GO TO (510,520,530),ISYM	PLT00991
510 CONTINUE	PLT00992
INT=AA	PLT00993
CALL SYMBOL(X2,Y2,HT,INT,0.0,-2)	PLT00994
GO TO 590	PLT00995
520 CONTINUE	PLT00996
NCHAR=4.0	PLT00997
XLEFT=(XLATE-1.0)*0.5*NCHAR*HT	PLT00998
YLEFT=(YLATE-1.0)*0.5*NCHAR*HT	PLT00999
CALL SYMBOL(X2+XLEFT,Y2+YLEFT,HT,AA,0.0,4)	PLT01000
GO TO 590	PLT01001
530 CONTINUE	PLT01002
SZ=2	PLT01003
S1=ABS(AA)	PLT01004
IF(S1.GT.SMAL) SZ=ALOG10(S1)	PLT01005
IF(S1.LT.0.0001) GO TO 591	PLT01006
SZ=ALOG10(S1)	PLT01007
IF(SZ.GE.0.0) NDEC=1	PLT01008
IF(SZ.LT.0.0) NN=SZ	PLT01009
IF(SZ.LT.0.0) NDEC=NN+2	PLT01010
IF(NDECFX.GE.0) NDEC=NDECFX	PLT01011
IF(SZ.GE.0.0) NSIG=SZ+1.0+2.0	PLT01012
IF(SZ.LT.0.0) NSIG=NDEC+2.0	PLT01013
GO TO 592	PLT01014
591 CONTINUE	PLT01015
NSIG=3	PLT01016
NDEC=1	PLT01017
592 CONTINUE	PLT01018
IF(AA.LT.0.0) NSIG=NSIG+1	PLT01019
XLEFT=HT*NSIG*(XLATE-1.0)*0.5	PLT01020
YLEFT=HT*NSIG*(YLATE-1.0)*0.5	PLT01021
CALL NUMBER(X2+XLEFT,Y2+YLEFT,HT,AA,0.0,NDEC)	PLT01022
590 CONTINUE	PLT01023
XLAS=XNOW	PLT01024
YLAS=YNOW	PLT01025
GO TO 300	PLT01026

600	CONTINUE	PLT01027
300	CONTINUE	PLT01028
	GO TO 301	PLT01029
302	CONTINUE	PLT01030
	WRITE (6,390) NPLT	PLT01031
390	FORMAT (6X, 'PLOT COMPLETED, TOTAL POINTS PLOTTED= ',I6)	PLT01032
	GO TO 999	PLT01033
999	RETURN	PLT01034
	END	PLT01035
	SUBROUTINE WINDOW(XA,YA,XB,YB,MOD)	PLT01036
C---	ROUTINE TO EXAMINE THE CURRENT SEGMENT RELATIVE TO THE CURRENT	PLT01037
C---	WINDOW	PLT01038
C---	INPUT IF MOD IS 0 THEN XA,YA ARE LOWER LEFT CORNER OF NEW WINDOW	PLT01039
C---	AND XB AND YB ARE THE WIDTH AND HEIGHT OF THE WINDOW	PLT01040
C---	OTHER PARAMETERS ARE ALSO INITIALIZED IN THIS CASE	PLT01041
C---	THE RETURN VALUE OF MOD IS -1	PLT01042
C---	IF MOD IS 1 THEN XA,YA AND XB,YB REPRESENT END POINTS OF A LINE	PLT01043
C---	SEGMENT WHICH SHOULD BE WINDOWED. IF MOD=-1 ON RETURN THE SEGMENT	PLT01044
C---	DOES NOT INTERSECT THE WINDOW, IF MOD=2 THE INTERSECTION OCCURS	PLT01045
C---	AND THE FIRST POINT DOES NOT CHANGE, WHILE IF MOD=3 THE FIRST P	PLT01046
C---	HAS CHANGED. XA,YA,XB,YB MAY BE MODIFIED ON OUTPUT TO HOLD	PLT01047
C---	CHANGED VALUES OF THE END POINTS	PLT01048
C---	IF MOD IS LESS THAN -1, AN ERROR HAS OCCURRED	PLT01049
	DIMENSION PX(2),PY(2),PD(5),X(5),Y(5),IND(2,2)	PLT01050
	DATA IND/1,2,4,3/	PLT01051
	DATA SMAL/1.0E-20/	PLT01052
	LOGICAL AIN,BIN	PLT01053
	BET(A,B,C)=(B-A)*(C-B)	PLT01054
	IF(MOD)20,10,20	PLT01055
C---	INITIALIZATION OF WINDOW PARAMETERS	PLT01056
10	CONTINUE	PLT01057
	XL=XA	PLT01058
	YL=YA	PLT01059
	XW=XB	PLT01060
	YW=YB	PLT01061
	XU=XL+XW	PLT01062
	YU=YL+YW	PLT01063
	X(1)=XL	PLT01064
	X(2)=XL+XW	PLT01065
	X(3)=X(2)	PLT01066
	X(4)=XL	PLT01067
	X(5)=XL	PLT01068
	Y(1)=YL	PLT01069
	Y(2)=YL	PLT01070
	Y(3)=YL+YW	PLT01071
	Y(4)=Y(3)	PLT01072
	Y(5)=YL	PLT01073
	HXW=XW/2.0	PLT01074
	HYW=YW/2.0	PLT01075
	KC=XL+HXW	PLT01076
	KC=YL+HYW	PLT01077
	DC=HXW*HXW+HYW*HYW	PLT01078
	MOD=-1	PLT01079
	GO TO 999	PLT01080
C---	BEGIN WINDOW CUTTING ACTION ON SEGMENT	PLT01081
20	CONTINUE	PLT01082
	AX=BET(XL,XA,XU)	PLT01083


```

AY=BET (YL, YA, YU)
AIN=.TRUE.
IF (AX.LT.0.0.OR.AY.LT.0.0) AIN=.FALSE.
BX=BET (XL, XB, XU)
BY=BET (YL, YB, YU)
BIN=.TRUE.
IF (BX.LT.0.0.OR.BY.LT.0.0) BIN=.FALSE.
IF (AIN.AND.BIN) GO TO 100
IF (AIN.OR.BIN) GO TO 200
GO TO 300
C--- BOTH INSIDE
100 CONTINUE
MOD=2
GO TO 999
C--- ONE INSIDE/ ONE OUTSIDE
200 CONTINUE
IF (AIN) GO TO 210
XX=XA
YY=YA
GO TO 220
210 XX=XB
YY=YB
220 CONTINUE
C--- CHOOSE THE MAIN CORNER REFERENCE POINT
SX=XX-XC
SY=YY-YC
I=2
J=2
IF (SX.LT.0.0) I=1
IF (SY.LT.0.0) J=1
IS=IND (I, J)
C--- SET UP THE EQN OF THE LINE SEGMENT
A=YB-YA
B=XA-XB
C=XB*YA-XA*YB
ISA=IS-1
IF (ISA.LT.1) ISA=4
D1=A*X (IS) +B*Y (IS) +C
D2=A*X (ISA) +B*Y (ISA) +C
IF (D1*D2.GT.0.0) ISA=IS+1
IF (ISA.GT.4) ISA=1
ICUM=ISA+IS
IF (ICUM.NE.5) GO TO 240
XX=X (IS)
YY=- (C+A*X (IS)) / (B+SMAL)
GO TO 250
240 XX=- (C+B*Y (IS)) / (A+SMAL)
YY=Y (IS)
250 CONTINUE
IF (AIN) GO TO 260
XA=XX
YA=YY
MOD=3
GO TO 999
260 CONTINUE
XB=XX
YB=YY

```

```

PLT01084
PLT01085
PLT01086
PLT01087
PLT01088
PLT01089
PLT01090
PLT01091
PLT01092
PLT01093
PLT01094
PLT01095
PLT01096
PLT01097
PLT01098
PLT01099
PLT01100
PLT01101
PLT01102
PLT01103
PLT01104
PLT01105
PLT01106
PLT01107
PLT01108
PLT01109
PLT01110
PLT01111
PLT01112
PLT01113
PLT01114
PLT01115
PLT01116
PLT01117
PLT01118
PLT01119
PLT01120
PLT01121
PLT01122
PLT01123
PLT01124
PLT01125
PLT01126
PLT01127
PLT01128
PLT01129
PLT01130
PLT01131
PLT01132
PLT01133
PLT01134
PLT01135
PLT01136
PLT01137
PLT01138
PLT01139
PLT01140

```

MOD=4	PLT01141
GO TO 999	PLT01142
C--- THE CASE OF TWO POINTS OUTSIDE THE WINDOW	PLT01143
300 CONTINUE	PLT01144
IF (XA-XL.LT.0.0.AND.XB-XL.LT.0.0) GO TO 390	PLT01145
IF (XA-XU.GT.0.0.AND.XB-XU.GT.0.0) GO TO 390	PLT01146
IF (YA-YL.LT.C.0.AND.YB-YL.LT.0.0) GO TO 390	PLT01147
IF (YA-YU.GT.0.0.AND.YB-YU.GT.0.0) GO TO 390	PLT01148
A=YB-YA	PLT01149
B=XA-XB	PLT01150
C=XB*YA-XA*YB	PLT01151
ICUM=0	PLT01152
PD (1)=A*X (1)+B*Y (1)+C	PLT01153
DO 310 I=2,5	PLT01154
PD (I)=A*X (I)+B*Y (I)+C	PLT01155
IF (PD (I)*PD (I-1).LT.0.0) ICUM=ICUM+1	PLT01156
310 CONTINUE	PLT01157
IF (ICUM.EQ.0) GO TO 390	PLT01158
NUM=0	PLT01159
DO 340 I=1,4	PLT01160
IF (PD (I)*PD (I+1).GT.0.0) GO TO 340	PLT01161
NUM=NUM+1	PLT01162
IF (NUM.GT.2) GO TO 340	PLT01163
ICUM=I+I+1	PLT01164
IF (ICUM.EQ.3.OP.ICUM.EQ.7) GO TO 350	PLT01165
PY (NUM)=- (C+A*X (I))/(B+SMAL)	PLT01166
PX (NUM)=X (I)	PLT01167
GO TO 340	PLT01168
350 PX (NUM)=- (C+B*Y (I))/(A+SMAL)	PLT01169
PY (NUM)=Y (I)	PLT01170
340 CONTINUE	PLT01171
IF (NUM.LT.2) GO TO 998	PLT01172
D1= (PX (1)-XA)**2+ (PY (1)-YA)**2	PLT01173
D2= (PX (2)-XA)**2+ (PY (2)-YA)**2	PLT01174
NUM1=1	PLT01175
IF (D2.LT.D1) NUM1=2	PLT01176
XA=PX (NUM1)	PLT01177
YA=PY (NUM1)	PLT01178
NUM2=2	PLT01179
IF (NUM1.EQ.2) NUM2=1	PLT01180
XB=PX (NUM2)	PLT01181
YB=PY (NUM2)	PLT01182
MOD=5	PLT01183
GO TO 999	PLT01184
390 CONTINUE	PLT01185
MOD=-1	PLT01186
999 CONTINUE	PLT01187
RETURN	PLT01188
998 MOD=-2	PLT01189
GO TO 999	PLT01190
END	PLT01191
SUBROUTINE DECOD (PP,VV,AA,JCON,ISYM,IVEC,I)	PLT01192
COMMON/PLBAS1/ P (4,3001),ICON (3001),NUM,NUMAX,IPLTX	PLT01193
COMMON/PLBAS2/ AP (16),AV (16),CP (16),DAT (8)	PLT01194
COMMON/PLBAS3/ WINXL,WINYL,WINXW,WINYW,IWIN	PLT01195
COMMON/PLBAS4/ SCRNXL,SCRNYL,SCRNXW,SCRNYW,ISCRN	PLT01196
COMMON/PLBAS5/ SIGNOR,SNPLOT,IH	PLT01197

```

COMMON/PLBAS7/HT,NDECFX,XLATE,YLATE
DIMENSION PP(3),VV(3)
IVEC=0
IF(I.GE.NUMAX) GO TO 999
IF(I.GT.NUM) GO TO 999
DO 10 L=1,3
10 PP(L)=P(L,I)
AA=P(4,I)
JCON=ICON(I)/10
ISYM=ICON(I)-10*JCON
IF(JCON.GE.5) GO TO 997
IF(ISYM.GT.3) GO TO 999
INEX=ICON(I+1)/10
IVEC=0
IF(INEX.NE.5) GO TO 998
I=I+1
DO 20 L=1,3
20 VV(L)=P(L,I)
IVEC=1
998 CONTINUE
RETURN
999 CONTINUE
I=-1
RETURN
997 CONTINUE
IVEC=999
IF(JCON.NE.7) RETURN
IF(ISYM.EQ.1) HT=P(4,I)
IF(ISYM.EQ.2) NDECFX=P(4,I)
IF(ISYM.EQ.3) XLATE=P(4,I)
IF(ISYM.EQ.4) YLATE=P(4,I)
RETURN
END
SUBROUTINE MMULT(A,B,C,M)
C--- CONSTRUCT C=A*B AND STORE THE RESULT IN A OR B
DIMENSION A(16),B(16),C(16)
DIMENSION ITEMP(4)
DATA ITEMP/1,5,9,13/
DO 10 IROW=1,4
DO 10 ICOL=1,4
KK=ITEMP(ICOL)
SUM=0.0
DO 11 K=1,4
SUM=SUM+A(IROW+K*4-4)*B(KK+K-1)
11 CONTINUE
C(4*ICOL-4+IROW)=SUM
10 CONTINUE
IDEBUG=0
IF(IDEBUG.EQ.0) GO TO 20
WRITE(6,50)
50 FORMAT(//)
DO 30 I=1,4
IL=I+12
WRITE(6,40) (A(L),L=I,IL,4), (B(L),L=I,IL,4), (C(L),L=I,IL,4)
40 FORMAT(' MMULT',4(1X,F8.3),3X,4(1X,F8.3),3X,4(1X,F8.3))
30 CONTINUE
20 CONTINUE

```

```

PLT01198
PLT01199
PLT01200
PLT01201
PLT01202
PLT01203
PLT01204
PLT01205
PLTC1206
PLT01207
PLT01208
PLT01209
PLT01210
PLT01211
PLT01212
PLT01213
PLT01214
PLT01215
PLT01216
PLT01217
PLT01218
PLT01219
PLT01220
PLT01221
PLT01222
PLT01223
PLT01224
PLT01225
PLT01226
PLT01227
PLT01228
PLT01229
PLT01230
PLT01231
PLT01232
PLT01233
PLT01234
PLT01235
PLT01236
PLT01237
PLT01238
PLT01239
PLT01240
PLT01241
PLT01242
PLT01243
PLT01244
PLT01245
PLT01246
PLT01247
PLT01248
PLT01249
PLT01250
PLT01251
PLT01252
PLT01253
PLT01254

```

IF (M.EQ.3) RETURN	PLT01255
DO 12 I=1,16	PLT01256
IF (M.EQ.1) A(I)=C(I)	PLT01257
IF (M.EQ.2) B(I)=C(I)	PLT01258
12 CONTINUE	PLT01259
RETURN	PLT01260
END	PLT01261
SUBROUTINE PERSPT (DAT,B)	PLT01262
C--- GENERATE A PROJECTIVE MATRIX B FROM A SIMPLE COMMAND DAT	PLT01263
DIMENSION DAT (1),B (1),AID (16)	PLT01264
DATA AID/1.0,4*0.0,1.0,4*0.0,1.0,4*0.0,1.0/	PLT01265
DATA CDR/0.01745329251994/	PLT01266
C--- DAT (1) CONTAINS THE COMMAND FLAG =1=XYROT, 2=YZROT,	PLT01267
C--- 3=ZXROT, 4=VARIABLE SCALE, 5=TRANS, 6=CENTER	PLT01268
DO 10 I=1,16	PLT01269
10 B (I)=AID (I)	PLT01270
IFLAG=DAT (1)	PLT01271
IF (IFLAG.GT.3) GO TO 50	PLT01272
A=DAT (2)*CDR	PLT01273
C=COS (A)	PLT01274
S=SIN (A)	PLT01275
GO TO (20,30,40),IFLAG	PLT01276
20 B (1)=C	PLT01277
B (2)=-S	PLT01278
B (5)=S	PLT01279
B (6)=C	PLT01280
GO TO 100	PLT01281
30 B (6)=C	PLT01282
B (7)=-S	PLT01283
B (10)=S	PLT01284
B (11)=C	PLT01285
GO TO 100	PLT01286
40 B (1)=C	PLT01287
B (3)=S	PLT01288
B (9)=-S	PLT01289
B (11)=C	PLT01290
GO TO 100	PLT01291
50 IFLAG=IFLAG-3	PLT01292
GO TO (60,70,80),IFLAG	PLT01293
60 W=DAT (3)**2+DAT (4)**2	PLT01294
IF (W.LT.0.000001) GO TO 65	PLT01295
B (1)=DAT (2)	PLT01296
B (6)=DAT (3)	PLT01297
B (11)=DAT (4)	PLT01298
GO TO 100	PLT01299
65 B (1)=DAT (2)	PLT01300
B (6)=DAT (2)	PLT01301
B (11)=DAT (2)	PLT01302
GO TO 100	PLT01303
70 B (4)=DAT (2)	PLT01304
B (8)=DAT (3)	PLT01305
B (12)=DAT (4)	PLT01306
GO TO 100	PLT01307
80 D=ABS (DAT (2))	PLT01308
B (11)=0.0	PLT01309
IF (D.GT.0.0001) B (15)=-1./DAT (2)	PLT01310
D1=ABS (DAT (3))	PLT01311

IF (D1.GT.0.0001.AND.D.GT.0.0001) B(16) = DAT(3)/DAT(2)	PLT01312
B(4) = -DAT(4)	PLT01313
B(8) = -DAT(5)	PLT01314
100 CONTINUE	PLT01315
IDEBUG=0	PLT01316
IF (IDEBUG.EQ.0) RETURN	PLT01317
WRITE(6,140)	PLT01318
140 FORMAT(//)	PLT01319
DO 110 I=1,4	PLT01320
IL=I+12	PLT01321
WRITE(6,120) (B(J),J=I,IL,4)	PLT01322
120 FORMAT(10X,'PERSPT',4(2X,F12.5))	PLT01323
110 CONTINUE	PLT01324
RETURN	PLT01325
END	PLT01326
SUBROUTINE USER	PLT01327
COMMON/PLBAS1/ P(4,3001),ICON(3001),NUM,NUMAX,IPLTX	PLT01328
COMMON/PLBAS2/ AP(16),AV(16),CP(16),DAT(8)	PLT01329
COMMON/PLBAS3/ WINXL,WINYL,WINXW,WINYW,IWIN	PLT01330
COMMON/PLBAS4/ SCRNL,SCRNYL,SCRNXW,SCRNYW,ISCRN	PLT01331
COMMON/PLBAS5/ SIGNOR,SNPLOT,IH	PLT01332
RETURN	PLT01333
END	PLT01334