LA-5587-MS
Informal Report

c. 3

UC-34c
Reporting Date: March 1974
Issued: April 1974

A Method to Estimate the Specific Fissions in ^{235}U Using a NaI Crystal

by

Silvio Balestrini

los alamos
scientific laboratory
of the University of California
LOS ALAMOS, NEW MEXICO 87544

UNITED STATES
ATOMIC ENERGY COMMISSION
CONTRACT W-7405-ENG. 36

In the interest of prompt distribution, this LAMS report was not edited by the Technical Information staff.

Printed in the United States of America. Available from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road
Springfield, Virginia 22151
Price: Printed Copy $4.00 Microfiche $1.45

A METHOD TO ESTIMATE THE SPECIFIC FISSIONS IN ^{235}U USING A NaI CRYSTAL

by

Silvio Balestrini

ABSTRACT

A technique is described to assay both the fissions and ^{235}U content in certain test samples used for critical assembly studies. The method essentially compares a pulse height analysis from a NaI well crystal scintillator with an empirical mathematical model. In the case of fissions, the analysis can be made from 6 hours to 50 days after irradiation. The listings and description are included for the computer program ASSAY that was developed to do the computations. The listings and brochure are also included for GPLOT, a general subroutine to plot on film and which is used extensively by ASSAY.

I. INTRODUCTION

This report describes a simple method to estimate the absolute number of fissions in a small sample of ^{235}U. It compares the shape and intensity of the gamma-radiation spectrum from the fission products in the sample by scaling to those predicted by an empirical mathematical model. Any time from 6 hours to 50 days after irradiation can be chosen. The ^{235}U in the sample is similarly assayed by comparing its gamma spectrum with that from a known mass.

The gamma-radiation energy spectra are obtained with a NaI well crystal scintillating counter and stored in a pulse height analyzer. This method takes advantage of the relatively poorer resolution of a NaI crystal compared to that of a Ge(Li) crystal. Fission-product spectrum detail from the NaI crystal is washed out into a simpler, gross picture that changes more slowly in character with time and also which is not too sensitive to the energy spectrum of the fissioning neutron flux.

The specifics of the ASSAY computer program that was developed to solve the problem are treated in a series of Appendices. Appendix A describes how to structure the problem deck for the computer from data cards. Appendix B describes the ASSAY routine itself and its associated subroutines and contains all but one of the FORTRAN listings. Subroutine GPLOT is treated separately in Appendix C. This is a generally useful subroutine to plot on film. It was developed by the author for his own use, and it found its way naturally into the ASSAY program. It is included in this report for the convenience of the reader; and because of its more general applicability, it is given this special treatment.

II. EXPERIMENTAL

It has been convenient at times in the critical-assembly studies group of this laboratory, when studying neutron activation profiles, to prepare samples of ^{235}U by cutting suitable lengths of standard alloy wire from a supply especially manufactured for the purpose. The wire alloy consists by weight of 90% Al and 10% U that is highly enriched (93%) in ^{235}U. The wire diameter is 0.51 mm.

A sample 381 mm (15 in.) long was spiraled to fit into a small space near the center of the Flattop critical assembly[1] and then irradiated with a high-intensity neutron flux. The sample was afterwards cut into three lengths in approximate proportion to 0.3, 1.2, and 13.5 to provide a range of radiating sources from 1 to 50 in strength.

The analyses were performed with a Harshaw type 12SW12-W4 NaI well crystal and photomultiplier unit. The crystal was enclosed by lead shielding more than 10 cm thick. The output was amplified and stored in a Hewlett-Packard 1024 channel pulse height analyzer set to count on real time. After each analysis, the data was read onto paper tape and later converted to computer cards. The channels were calibrated for energy using the 0.662 MeV γ-ray from ^{137}Cs.

The shortest sample piece was analyzed periodically, beginning a few hours after irradiation. The sample position inside the crystal well was fixed by scotchtaping the sample to the flat bottom of a plastic crucible of a size and shape to fit snugly in the bottom of the well. When the intensities had decreased sufficiently, the shortest sample piece was analyzed several times alternately by itself and together with the next longer piece to normalize the two spectrum sources. About a week later, a similar normalization was possible also between the two sample pieces and all three. Nearly 50 spectral analyses were made, subtracting the background each time, covering a period of 53 days after irradiation. The time interval between analyses was increased as time went on. The entire set of analyses normalized to the intensity of the complete sample is shown in Fig. 1.

III. THE FISSION SPECTRUM MODEL

In structuring the model, the time dependence of the radiation intensity at a given energy was examined first at each of all values 0.02 MeV apart from 0.1 to 2.0 MeV.

Because of the scatter of counts in those channels near any selected energy, the intensity for each analysis was estimated as follows: Since the gain was inadvertedly changed in the course of the work, the counts per channel in some analyses had to be adjusted first to reflect the same intensity in counts per minute per MeV interval for the gain originally set. Then the intensity was estimated as the value of a second degree function adjusted for best fit to the data from the seven adjacent channels centered nearest to that energy.

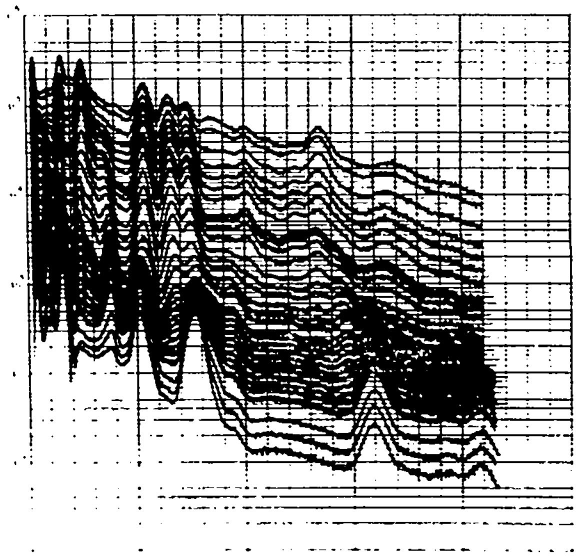

Fig. 1. Synthesis of all the normalized pulse height analyses of the irradiated sample, taken from a few hours to over 50 days after irradiation.

The results at 0.5 MeV are typical and appear in Fig. 2. The intensity estimates are shown as points. The curve results when a smoothing spline is adjusted[2-3] with ten knots for best fit to the points. It was more convenient for this work to fit the spline to the logarithms of the rates and time lapses after irradiation rather than to the values themselves. The knots are equally spaced on the log-time scale from 5 to 1300 hours.

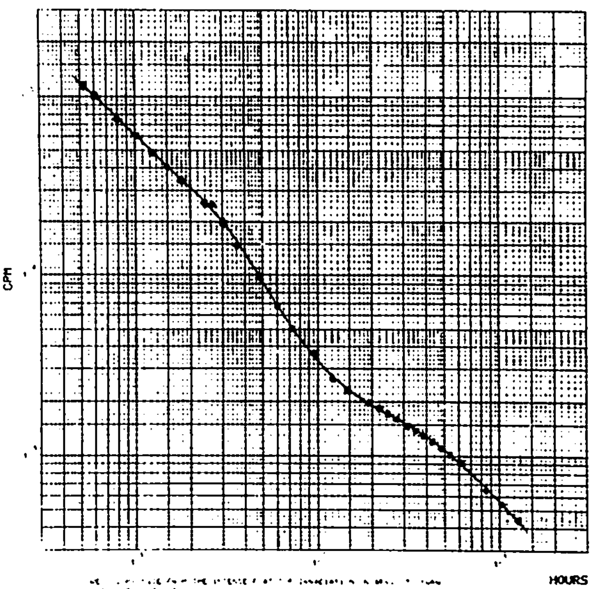

Fig. 2. Time dependence of the radiation intensity at 0.5 MeV.

The shape of the radiation energy spectrum can now be constructed for any time lapse within this time span: A value for the activity is computed from each adjusted spline for the time lapse, and then a new spline is evaluated as a function of energy which passes through these activity values. Figure 3 is a set of 50 spectrum constructions for time lapses increasing in geometric progression from 5 to 1500 hours after irradiation.

The model is formed by choosing only the adjusted splines included by the energy range from 0.45 to 0.9 MeV. When a spectrum construction is to be compared with data for assay, three practical considerations are met in this energy range: The structure is relatively simple and yet distinctive (useful in curve fitting), the intensity is reasonably high, and it is also comparatively free of background from cosmic rays, from the ^{235}U in the sample, and from aluminum activation.

The model was finally calibrated by comparison with radiochemical results: A ^{235}U sample prepared from the supply of standard Al-U(93) wire was irradiated and then analyzed inside the well crystal using an adopted standard geometry. It was afterwards submitted for radiochemical analysis for the number of fissions via a determination of the ^{99}Mo content.

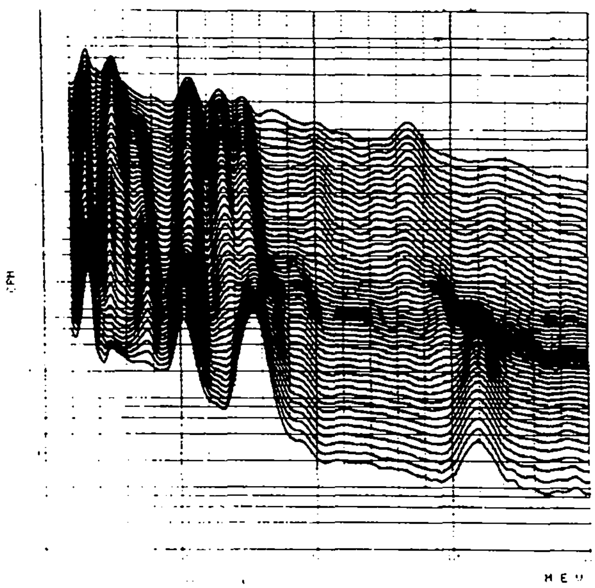

Fig. 3. Fission-product spectrum constructions for time lapses increasing in geometric progression from 5 to 1500 hours after irradiation.

IV. THE ASSAY FOR FISSIONS

A sample is analyzed in the well crystal at some convenient lapsed time T after irradiation, when the radioactivity level has decreased to some satisfactory point. Generally, T is long compared to the durations of both the irradiation and the analysis, so that it can be measured from the time center of one duration to that of the other.

A fission-product spectrum constructed from the model for the time lapse T is a spline function of energy, S(E), which must be related to the data. Let Y_i represent the counts in the i-th channel. The energy is linear with channel number; i.e., for channel i,

$$E_i = A + B \cdot i \tag{1}$$

where A and B are constants. The spline is fit to the data[4] by adjusting three parameters A, B, and C so as to minimize the summation

$$\Sigma P_i [Y_i - C \cdot S(A + B \cdot i)]^2 \tag{2}$$

over all the data. P_i is a weight associated with Y_i and which is set equal either to Y_i if the channel falls within the energy range of the model or to zero if it does not. The number of fissions in the sample is

$$f = K_f \frac{C}{B \cdot t} \tag{3}$$

where t is the duration of the analysis and K_f is the fission calibration factor for the crystal.

An example of a spline from the model when fit to the data from an actual analysis appears in Fig. 4. In this assay, the fissions were evaluated at $(1.541 \pm 0.016) \times 10^9$. The assay was repeated from the same analysis data, but this time using only the counts in those channels between 0.475 and 0.59 MeV (about one-fourth as many channels), and the result was $1.587 \pm 0.019 \times 10^9$ fissions. It is important to understand the reason for this discrepancy. In both cases, the adjusted values for the amplitude parameter C agreed to within less than 0.2%. The discrepancy between the assays resulted almost entirely from the two estimates for B, which is the scaling factor for the energy interval per channel. In the former case, the evaluation for B was

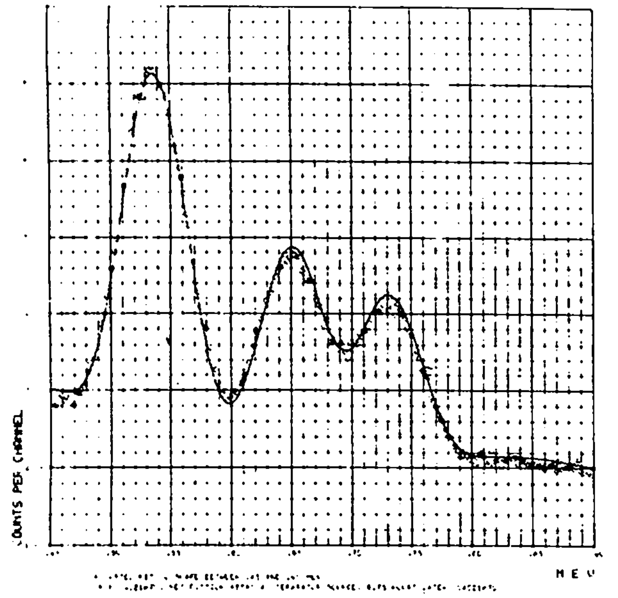

Fig. 4. A fission-product spectrum constructed from the model and then fit by scaling to the data from an analysis.

influenced mostly by the channel locations of the three major spectral peaks. In the latter case, the evaluation was forced to rely only on the width of the one peak, which is less reliable. The two assay error estimates do not reflect the discrepancy since they are based in part on how well the spline function can be scaled to fit both coordinates of the data. The more trustworthy assay is the former, when all the data was used, as will been seen in the discussion below.

V. THE ASSAY FOR ^{235}U

A spectrum model for the radiation from ^{235}U decay was constructed more simply. Since this spectrum is invariant with time, one well crystal analysis of a previously unirradiated sample followed later by a radiochemical assay for the ^{235}U content completed the necessary experimental work. The model was formed directly by a smoothing spline through the analysis data and selecting knots equally spaced 0.01 MeV apart. The energy range of this model spans the values from 0.135 to 0.25 MeV in order to describe the 0.19-MeV line and neighboring lesser lines. The model consists of a value for the intensity and the second derivative for each of a set of energy values.

The mass of ^{235}U in a sample is assayed by matching the model to the data from the well crystal analysis of the sample before irradiation. As for the case with fissions, see equations (1) to (3); the mass of ^{235}U is

$$U = K_U \frac{C}{B \cdot t} \qquad (4)$$

where K_U is the mass calibration factor for the crystal.

VI. STANDARD SAMPLE GEOMETRIES

Since the efficiency of the crystal depends to some extent upon the location of the sample in the well of the crystal, a standard reproducible sample geometry is important.

As a primary standard geometry, the wire sample was first wound in the form of a tight helix roughly 2.3 cm in diameter, using a mandrel, and then inserted into a plastic crucible so as to fit firmly against the crucible sides by spring action. A lid fitting tightly inside the crucible helped to hold the sample fixed and also reduced the risk of contamination in the well. An advantage with this geometry is that the length of a sample up to 40 cm is not critical.

A secondary standard geometry was adopted for convenience. The wire sample was wound in a compact spiral resembling a disc and held firmly against the bottom of the crucible by the crucible lid. Although the crystal is less sensitive to this geometry, the difference is less than 2% for 10 mg and smaller samples of ^{235}U, and sometimes worth the convenience since the equipment to spiral the wire already exists, and at times a sample in the form of a spiraled disk is desirable.

When an assay for ^{235}U was attempted on two 1.27-cm-diameter U(93) foils 25 μm thick and weighing about 60 mg each, the results fell short by about 14% due in part to a bad state of oxidation that affected the apparent weight and in part to self-absorption. The latter was shown to be a contributing cause by assaying the foils flat against each other and then separately. The sum of the assays from each was 7% greater than from the two together. Although the 0.19-MeV radiation from ^{235}U is more likely to be self-absorbed than the higher energy radiation from the fission products, the risk of error by departing to other forms of samples should be borne in mind.

VII. DISCUSSION

A group of 20 samples from a neutron activation profile study of a critical assembly were spiraled and then analyzed for relative activation in the programable, sample cycling beta counter known as the "jukebox."[5] Each sample was afterwards assayed for fissions using the well crystal, and then they were all assayed for fissions in two groups of ten each by the Los Alamos Scientific Laboratory radiochemistry group CNC-11.

Table I compares the fission assays from the well crystal with the relative activations from the beta-counter results. These samples were prepared from four identical, long wires that had been placed side by side during the irradiation and afterwards cut into five lengths each. All the samples with the same letter were of the same length and exposed to the same neutron flux and should therefore have the same activation except for possible small variations in ^{235}U content. In general, the like-letter groups of results show standard deviations that are precentagewise smaller for the well crystal assays than for the activation ratios from the beta-counter measurements. However, the individual error estimates for the latter, which are based on reproducibility from repeated observations, are smaller. A likely explanation for the discrepancy, although the discrepancy is small, is that the well counter is less sensitive to sample form for small samples than the beta counter of the jukebox. In support of this, one notes from the normalized ratios in the last column of Table I that these tend to be lower for the "A" and "E" groups. And indeed, the length of these wires was 30% shorter than the uniform, 19-cm length for the others.

TABLE I

COMPARISON OF TWENTY RELATIVE ACTIVATIONS DETERMINED BY THE CYCLING BETA COUNTER WITH THE NUMBER OF FISSIONS ESTIMATED USING THE NaI-CRYSTAL WELL COUNTER

Wire Sample	NaI Crystal Assay, Fissions x 10^{-9}		Relative Activation, Beta Counter Results		Normalized Ratios[a]	
1-A	0.8954	± .0093	0.6550	± .0029	0.963	± .011
B	1.441	.015	1.0000	.0054	1.015	.012
C	1.236	.013	0.8602	.0035	1.012	.011
D	0.8185	.0086	0.5750	.0029	1.002	.012
E	0.2214	.0024	0.1559	.0011	1.000	.013
2-A	0.8938	± .0093	0.6403	± .0020	0.983	± .011
B	1.420	.015	0.9916	.0037	1.008	.011
C	1.249	.013	0.8829	.0039	0.996	.011
D	0.8270	.0086	0.5810	.0033	1.002	.012
E	0.2240	.0025	0.1588	.0012	0.993	.013
3-A	0.8849	± .0092	0.6415	± .0022	0.971	± .011
B	1.404	.014	0.9860	.0043	1.003	.011
C	1.228	.013	0.8694	.0041	0.995	.012
D	0.8091	.0084	0.5610	.0032	1.016	.012
E	0.2202	.0025	0.1526	.0012	1.016	.014
4-A	0.8943	± .0093	0.6385	± .0024	0.986	± .011
B	1.404	.015	0.9876	.0044	1.001	.012
C	1.254	.013	0.8594	.0041	1.028	.011
D	0.8207	.0085	0.5569	.0033	1.038	.012
E	0.2254	.0024	0.1579	.0012	1.005	.013

[a] Ratio of fission assay to beta counter results multiplied by a factor to normalize the average to unity.

TABLE II

COMPARISON OF FISSION ASSAY RESULTS USING THE NaI-CRYSTAL ASSAY METHOD WITH THE RADIOCHEMICAL RESULTS FROM CNC-11.

Wire Samples	Sum of NaI Crystal Assays, Fissions	CNC-11 Assay, Fissions
All of Wires 1 & 2	$(9.226 \pm .033) \times 10^9$	9.225×10^9
All of Wires 3 & 4	$(9.143 \pm .033) \times 10^9$	9.065×10^9

The results of the fission assays from the well crystal and radiochemistry are compared in Table II. The elapsed time between irradiation and well crystal assay varied from 6.8 to 11.8 hours. This degree of good agreement has been obtained in other instances when elapsed times were as long as 42 hours.

VIII. ACKNOWLEDGMENTS

The author expresses his gratitude to E. A. Bryant and G. W. Knobeloch of Group CNC-11 for the several radiochemical assays used to calibrate and test the models and also to J. C. Hoogterp of Group N-2 for making available his irradiated samples and the results from the jukebox so that a comparison of methods could be made.

IX. REFERENCES

1. W. U. Geer, P. G. Koontz, J. D. Orndoff, and H. C. Paxton, "Safety Analysis for the Los Alamos Critical-Assembly Facility," Los Alamos Scientific Laboratory report LA-4273 (November 1969), p. 7.

2. J. L. Walsh, J. H. Ahlberg, E. N. Nilson, "Best Approximation Properties of the Spline Fit," J. Math. Mech. 11, 225 (1962).

3. T. L. Jordan, "Smoothing and Multivariable Interpolation with Splines," Los Alamos Scientific Laboratory report LA-3137 (March 1965).

4. R. H. Moore and R. K. Zeigler, "The Solution of the General Least Squares Problem with Special Reference to High-Speed Computers," Los Alamos Scientific Laboratory report LA-2367 (March 1960).

5. C. G. Chezem and H. H. Helmick, "100-Foil Sample Changer with Tape-Controlled Sequencing," Nucleonics 21, 3, 83 (1963).

APPENDIX A

STRUCTURING THE PROBLEM DECK FOR THE ASSAY ROUTINE

Generally, the experimenter may ask the following three questions concerning a sample of ^{235}U:

1. What is the ^{235}U content?
2. How many fissions occurred in it?
3. What are the specific fissions (fissions per gram of ^{235}U)?

The computer can be directed to answer all or part of these questions by means of entries in a control card, which is read first. An A-deck will refer to a deck of computer cards that contains all the information concerning a well crystal pulse height analysis. One or two A-decks follow the control card to complete the problem deck. When finished, the computer will read the next control card for the next problem deck. A blank card causes the computer to return.

To solve only for the ^{235}U content or the fissions, the appropriate A-deck follows the control card. But if the specific fissions are desired, three choices exist:

1. The ^{235}U pre-irradiation A-deck is read first followed by the fission A-deck. The solution produces two assay results and the specific fissions together with an estimate of the propagated error in this ratio.

2. Only the fission A-deck is read, and the assay is related to the results of a previous ^{235}U assay of a representative sample that are still in computer memory. After once made, the results from a ^{235}U assay remain available in memory from problem to problem until replaced with new values from a subsequent assay.

3. Only the fission A-deck is read, but the assay is related to an assumed mass of ^{235}U.

A-deck Structure

The A-deck is structured as follows:

1. First Card: Title card containing the analysis description, 10A8 format,

2. A series of data cards follow the title card. The format is 1X, I4, 1X, 10F6.0, 6X, A8. The first entry is the number of a channel and is mandatory. The subsequent entries are the contents of that channel and of the next nine, followed by an identification (ID) for the analysis; entries here are not mandatory. If the card for channel zero is included, it must be the first card. The others can follow in any order.

3. Last Card: Termination for the A-deck, must be blank in spaces 2-5. The remaining 10F6.0 fields are used to record the following information:

Spaces 7-12, count duration in minutes.

Spaces 13-18, analyzer dead time in percent.

Spaces 19-24, estimated channel number for zero energy; the iterating routine requires this estimate to start with.

Spaces 25-30, estimated channel number where the ^{137}Cs, 0.662-MeV gamma-ray peak would be centered for this gain setting; the iterating routine also requires this estimate to start with.

Spaces 31-36, minimum channel to be used.

Spaces 37-42, maximum channel to be used. If no entry, the value 1000 will be substituted by the program.

Control Card

The control card is prepared as follows (obviously, not all entries are always mandatory):

Space 1 entries:

- 0 = No ^{235}U assay is to be made.
- 1 = The deck that follows is from a ^{235}U analysis.
- 2 = Also list the ^{235}U analysis data in the output.
- 3 = Use the ^{235}U results in memory from the earlier assay.

Space 2 entries:

- 0 = No fission assay is to be made. (If both spaces 1 and 2 are zero, the computer returns.)
- 1 = A fission assay is to be made.
- 2 = Also list the fission analysis data in the output.

Space 3 entries:

- 0 = The fission spectrum analysis has not had the background subtracted from it. (The program will correct for it.)
- 1 = The fission spectrum analysis data is corrected for background.

Spaces 7-10, time when irradiation started; 2:38 p.m. is written 1438, and midnight is 2400; I4 format.

Spaces 13-16, duration of the irradiation in minutes; I4 format.

Spaces 19-26, irradiation date, i.e., 10/18/74; A8 format.

Spaces 31-34, time when the well crystal fission analysis started; I4 format.

Spaces 37-44, date when well crystal analysis started; A8 format.

Spaces 46-48, number of calendar days between irradiation date and analysis date, regardless of the times of day; I3 format. (If the dates are 10/18/74 and 10/19/74, this entry is 1.) The computer uses this information and the times of day to compute the time elapsed after irradiation.

Spaces 49-54, optional, default entry for an assumed mass in mg of ^{235}U for the irradiated sample; F6.0 format. (This entry will not affect the value in memory from a previous assay.)

APPENDIX B

ASSAY PROGRAM AND ASSOCIATED SUBROUTINES

The main PROGRAM reads the control card and proceeds according to the entries for the major controls (MC) in spaces 1-3. The main function is to understand the MC directives, conduct the computations through subroutines and print the final results. The principal variables used throughout are defined in this listing.

Subroutine ASSAYZ is a block of data that is structured as a subroutine and called once early from the main program. (As a subroutine, it can be included in one's USERLIB file.) It contains the information for the ^{235}U model and the cosmic-ray background model. Each model is described first by a listing of the model identification, the number of spline points, and several essential parameters. This is followed by listings of the abscissa, the ordinate, and the second derivative for each point for use by spline routines.

Subroutine FSHAPE contains the fission model and constructs the shape of the fission spectrum from it for any time lapse after irradiation. The block of data in this subroutine is headed by a listing of the model identification, six parameters essential to it, the number of knots on the time axis, and the number of points on the energy axis. This is followed by data arrays for 24 splines with 10 knots each, describing the logarithms of the intensity (CPMK) and times (CPMT) and the associated second derivatives (CPMW) for each of the 24 energy values (FX). The subroutine solves for the intensity of the model at the lapsed time for each energy value and then passes a spline through these values.

Subroutine READ HP reads and stores the A-deck contents, lists them if directed, and plots the data on film.

Subroutine BPART. If the fission analysis data has not had a background subtracted from it, this subroutine corrects for it.

Subroutine MATCH adjusts the scaling of the spline function for best fit to the data. It computes an assay from the scaling parameters P and also an estimate of precision from the goodness of fit and the estimated calibration accuracy for the model. It plots the data and the fitted spline on film. This subroutine makes repeated use of the subroutine ASSAYP.

Subroutine ASSAYP is an iterative program that adjusts the three scaling parameters P of a spline function to fit a set of data by the least squares criterion. The arguments IX1 and IX2 are optional entries for indices of parameters to be kept constant during adjustment of the others. This version was adapted for the problem from a more general routine developed for the least squares treatment discussed in Ref. 4 of this report.

Subroutines SPLIN1 and SPLIN2 are old subroutines no longer available except from the archives, and therefore they are listed here for the convenience of the user. The identification label is E202. SPLIN1 solves for the second derivatives of the spline given the coordinates of the points, and SPLIN2 solves for the value of the function for a given argument.

Subroutine GPLOT: See Appendix C.

```
      PROGRAM     ASSAY       ( INP, OUT, PUN, FILM,                        ASSAY001
     P      FSET9=OUT, FSET10=INP, FSET11=PUN, FSET12=FILM )                ASSAY002
C- VERSN = CURRENT VERSION OF THE ASSAYING CODE                             ASSAY003
C-UMODEL = LABEL OF THE URANIUM SPECTRUM MODEL USED                         ASSAY004
C-BMODEL = LABEL OF THE BACKGROUND SPECTRUM MODEL USED (I.E. COSMIC RAYSASSAY005
C-FMODEL = LABEL OF THE FISSION SPECTRUM MODEL USED                         ASSAY006
C-    MC = MAJOR CONTROL NUMBERS                                            ASSAY007
C- ID235, IDFISS = A-DECK LABEL CARD CONTENTS                               ASSAY008
C-  ASMG, ASSD = WT AND SD OF THE URANIUM-235 ASSAY                         ASSAY009
C-   UDT = COUNT DURRATION FOR THE U-235 ASSAY                              ASSAY010
C-   UWT = WT OF U-235 IF ENTERED WITHOUT ASSAY                             ASSAY011
C- RATIO = AMPLITUDE RATIO OF DATA TO THE CURVE SHAPE MODEL                 ASSAY012
C-IRDATE, IRT1, IRT2 = IRRADIATION DATE, START TIME AND DURATION            ASSAY013
C-WLDATE, IWLT = WELL COUNT DATE AND START TIME                             ASSAY014
C- NDAYS = NUMBER OF CALENDAR DAYS WLDATE FOLLOWS THE IRRADIATION           ASSAY015
C-   UCS = CS-137 CHANNEL NUMBER FOR THE U-235 ASSAY                        ASSAY016
C-   FCS = CS-137 CHANNEL NUMBER FOR THE FISSION ASSAY                      ASSAY017
C-MINUCH, MAXUCH, MINFCH, MAXFCH = MIN AND MAX CHANNELS FOR ASSAYS          ASSAY018
C-  NUCH, NFCH = NUMBER OF CHANNELS QUALIFYING FOR U AND FISSION ASSAYS ASSAY019
C-  FISS, FISSSD = NUMBER AND SD OF FISSIONS                                ASSAY020
C-   NUK, NFK, NBK - UX, UY, UW - FX, FY, FW - BX, BY, BW -                 ASSAY021
C        = NUMBER OF SPLINE POINTS, ABSCISSAE, ORDINATES, AND SECOND        ASSAY022
C          DERIVATIVES, RESPECTIVELY, FOR U-235, FISSION, AND BACKGROUNDASSAY023
C-   FPG, FPGSD = FISSIONS PER GRAM AND SD OF THE ASSAY                     ASSAY024
C-  UPAR, FPAR, I=1,2 = ENERGY LIMITS FOR U AND FISSION SHAPE COMPARISONASSAY025
C-                  3 = MDEL COUNT TIME                                     ASSAY026
C-                  4 = MODEL QUANTITY, I.E., MG OF U-235                   ASSAY027
C-                  5 = CS-137 CHANNEL FOR THE MODEL                        ASSAY028
C-                  6 = UNCERTAINTY PLACED ON THE MODEL QUANTITY            ASSAY029
C   BPAR, I=1, MODEL COUNT TIME                                             ASSAY030
C           2, CS-137 CHANNEL FOR THE MODEL                                 ASSAY031
      DIMENSION  ID235(10), IDFISS(10), MC(3), JOB(4)                       ASSAY032
      COMMON  /DATA/  N, X(1030), Y(1030), DUM(10)                          ASSAY033
      COMMON  /MODEL/  UMODEL,FMODEL,BMODEL                                 ASSAY034
      COMMON  /USPLIN/  NUK, UPAR(6), UX(50), UY(50), UW(50)                ASSAY035
      COMMON  /FSPLIN/  NFK, FPAR(6), FX(50), FY(50), FW(50)                ASSAY036
      LOGICAL  ASSAYU, ASSAYF, OLDWT, ARBWT, NET                            ASSAY037
      DATA  VERSN/8HFEB 1974/                                               ASSAY038
      DATA  ZERO/1.E-6/                                                     ASSAY039
    2 FORMAT( 3I1,3X,I4, 2X, I4, 2X, A8, 4X, I4, 2X, A8, 1X, I3, F6.0 ) ASSAY040
    3 FORMAT(1H1////* WELL COUNTER ASSAY SUMMARY, ASSAY ROUTINE VERSION ASSAY041
     X*A8/*  RUN ON *  A8   * AT * A8 *, JOB   * A10 *    SERIAL* I3 / )  ASSAY042
    4 FORMAT( *0ORALLOY SAMPLE -*/ 1X10A8 )                                 ASSAY043
    5 FORMAT( *0ORALLOY SAMPLE WEIGHT IS CHOSEN AT *1PE10.3*  MG U-235*)ASSAY044
    6 FORMAT( *0IRRADIATION -*/ 1X10A8 )                                    ASSAY045
    7 FORMAT(/6X44(1H* )/  6X1H*, 19X,  14HVALUE      SD , 9X 1H* )         ASSAY046
    8 FORMAT(6X1H*,42X1H*/6X16H*     FISSIONS = , 1PE12.4, E10.2, 5X1H*)ASSAY047
    9 FORMAT(6X1H*,42X1H*/6X16H*     MG U-235 = , 1PE12.4, E10.2, 5X1H*)ASSAY048
   10 FORMAT(6X1H*,42X1H*/6X16H*      FISS/GM = , 1PE12.4, E10.2, 5X1H*)ASSAY049
   11 FORMAT( /  * IRRADIATION STARTED ON  *A8*  AT  *I4*, LASTED*I3      ASSAY050
     X* MINUTES*/* WELL COUNT STARTED ON  *A8*  AT  *I4*, LASTED*I4* MINASSAY051
     XUTES*/13X*WITH  *I2*  PERCENT DEAD TIME,* F7.1*  HOURS LATER*/     ASSAY052
     X12X,I4* CHANNELS WERE USED BETWEEN  *I3*  AND  *I3/                   ASSAY053
     X 13X*WITH THE CS-137 LINE FALLING ON  * F7.1 )                        ASSAY054
   12 FORMAT( * FISSION MODEL  *A8*  WAS USED ON A NET SPECTRUM*)           ASSAY055
   13 FORMAT( * FISSION MODEL  *A8*  WAS USED ON A GROSS SPECTRUM*/         ASSAY056
     X         13X*AFTER CORRECTING IT WITH BACKGROUND MODEL  * A8 )        ASSAY057
   14 FORMAT( /  * U-235 WELL COUNT LASTED  *I3*  MINUTES*/                 ASSAY058
     X12X,I4* CHANNELS WERE USED BETWEEN  *I3*  AND  *I3/                   ASSAY059
     X 13X*WITH THE CS-137 LINE FALLING ON  * F7.1 /                        ASSAY060
```

```
  X    * URANIUN MODEL   * A8 *   WAS USED* )
 15 FORMAT( 6X1H*, 42X1H*/ 6X, 44(1H* ) )
    CALL         ASSAYZ
    CALL  DATE1   ( JOB(1) )
    CALL  CLOCK1   ( JOB(2) )
    CALL  JOBNAME( JOB(3) )
    JOB(4) = 0
 99 ASSAYU = .FALSE.
    ASSAYF = .FALSE.
    OLDWT = .FALSE.
    ARBWT = .FALSE.
    NET = .FALSE.
    JOB(4) = JOB(4) + 1
    READ(10,2) MC, IRT1, IRDT, IRDATE, IWLT, WLDATE, NDAYS, UWT
    IF   ( MC(1) .EQ. 0 .AND. MC(2) .EQ. 0 )   RETURN
    IF   ( MC(1) .GT. 0 )   ASSAYU = .TRUE.
    IF   ( MC(1) .EQ. 3 )   OLDWT = .TRUE.
    IF   ( MC(2) .GT. 0 )   ASSAYF = .TRUE.
    IF   ( MC(3) .GT. 0 )   NET = .TRUE.
    IF   ( OLDWT )   GO TO  200
    IF   ( ASSAYU )   GO TO  110
    IF   ( UWT .GT. ZERO )   ARBWT = .TRUE.
    GO TO  200
110 CALL  READ HP   ( ID235, MC(1), JOB )
    UDT = DUM(1)
    CALL MATCH ( NUCH, NUK, UX, UY, UW, UPAR, ASMG, ASSD, ID235, JOB )
    UCS = DUM(4)
    MINUCH = DUM(5) + 0.5
    MAXUCH = DUM(6) + 0.5
    IF   ( .NOT. ASSAYF )   GO TO   300
200 CALL  READ HP   ( IDFISS, MC(2), JOB )
    IF   ( NET )   GO TO 205
    CALL  BPART
205 NH = IRT1 / 100
    M = IRT1 - NH * 100
    T = FLOAT(NH) + ( FLOAT(M)+0.5*FLOAT(IRDT) ) / 60.0
    NH = IWLT / 100
    M = IWLT - NH * 100
    DECAYT = FLOAT(NDAYS)*24.0-T+FLOAT(NH)+FLOAT(M)/60.0+
  X         DUM(1)/(1.0-DUM(2) *0.01)/120.0
210 CALL  FSHAPE   ( DECAYT )
    CALL MATCH ( NFCH,NFK,FX,FY,FW,FPAR,FISS,FISSSD,IDFISS,JOB )
    FCS = DUM(4)
    MINFCH = DUM(5) + 0.5
    MAXFCH = DUM(6) + 0.5
    IF   ( ARBWT )   GO TO   220
    IF   ( .NOT. ASSAYU )   GO TO   300
    FPG = FISS /   ASMG * 1000.0
    FPGSD = FPG*SQRT( (FISSSD/FISS)**2 + (ASSD/ ASMG)**2 )
    GO TO  300
220 FPG = FISS / UWT * 1000.0
    FPGSD = FISSSD/ UWT * 1000.0
300 WRITE(9,3) VERSN, JOB
    IF   ( ASSAYU .OR. OLDWT )   WRITE(9,4) ID235
    IF   ( ARBWT )   WRITE(9,5)   UWT
    IF   ( ASSAYF )   WRITE(9,6)   IDFISS
    WRITE(9,7)
    IF   ( ASSAYF )   WRITE(9,8)   FISS, FISS SD
    IF   ( ASSAYU )   WRITE(9,9)    ASMG, ASSD
    IF   ( ASSAYF .AND. (ASSAYU.OR.ARBWT) )   WRITE(9,10) FPG, FPGSD
```

```
      WRITE(9,15)                                                       ASSAY121
      IF  ( .NOT.  ASSAYF )  GO TO  310                                 ASSAY122
      IT = DUM(1) + 0.5                                                 ASSAY123
      IP = DUM(2)+ 0.5                                                  ASSAY124
      WRITE(9,11) IRDATE, IRT1, IRDT, WLDATE, IWLT, IT, IP, DECAYT,     ASSAY125
     X            NFCH, MINFCH, MAXFCH, FCS                             ASSAY126
      IF  ( NET )  WRITE(9,12)  FMODEL                                  ASSAY127
      IF  ( .NOT. NET )  WRITE(9,13)  FMODEL, BMODEL                    ASSAY128
  310 IF  ( .NOT. ASSAYU )  GO TO  99                                   ASSAY129
      IT = UDT + 0.5                                                    ASSAY130
      WRITE(9,14) IT, NUCH, MINUCH, MAXUCH, UCS, UMODEL                 ASSAY131
      GO TO  99                                                         ASSAY132
      END                                                               ASSAY133
```

```
      SUBROUTINE  ASSAYZ                                                ASAYZ 01
C     BLOCK DATA FOR THE ASSAY CODE ON USERLIB TAPE                     ASAYZ 02
      COMMON  /MODEL/  UMODEL,FMODEL,BMODEL                             ASAYZ 03
      COMMON  /USPLIN/  NUK, UPAR(6), UX(50), UY(50), UW(50)            ASAYZ 04
      COMMON  /BSPLIN/  NBK, BPAR(2), BX(50), BY(50), BW(50)            ASAYZ 05
C     URANIUM SPECTRUM SHAPE                                            ASAYZ 06
      DATA  UMODEL/8HMAY 1974/,NUK/15/,UPAR/.135,.25,1.,9.8,662.,0.1/   ASAYZ 07
      DATA  (UX(I),I=1,15) / .12,.13,.14,.15,.16,.17,.18,.19,.2,.21,.22,ASAYZ 08
     X                       .23,.24,.25,.26/                           ASAYZ 09
      DATA  ( UY(I), I = 1, 15 )  /       1.4564E2,  1.0975E2,  1.2200E2,ASAYZ 10
     X  1.9492E2,  2.1683E2,  2.1895E2,  3.7618E2,  6.5999E2,  7.2401E2,ASAYZ 11
     X  4.6409E2,  1.9829E2,  6.9214E1,  2.5461E1,  1.3728E1,  1.3728E1/ASAYZ 12
      DATA  ( UW(I), I = 1, 15 )  /       3.2945E2,  4.6906E5,  1.0121E6,ASAYZ 13
     X -8.7748E5, -5.6306E5,  1.9427E6,  2.0988E6, -2.7432E6, -4.3136E6,ASAYZ 14
     X  5.6139E5,  1.7150E6,  7.8270E5,  2.6750E5,  8.0106E4,  0.0     /ASAYZ 15
C     COSMIC RAY BACKGROUND                                             ASAYZ 16
      DATA  BMODEL/8H031270  /, NBK/6/, BPAR/1.,931./                   ASAYZ 17
      DATA  ( BX(I),I = 1, 6 ) / 0.25, 0.5, 0.75, 1.0, 1.5, 2.0 /       ASAYZ 18
      DATA  (BY(I),I=1,6)/1.1679,.47823,.26427,.17725,.073422,.024707/  ASAYZ 19
      DATA  (BW(I),I=1,6)/8.9926,8.9926,.70303,.3815,.18919,.18919/     ASAYZ 20
      RETURN   $   END                                                  ASAYZ 21
```

```
      SUBROUTINE  FSHAPE  ( T )                                         FSHPE 01
      COMMON  /FSPLIN/  NFK, FPAR(6), FX(50), FY(50), FW(50)            FSHPE 02
      COMMON  /MODEL/  UMODEL,FMODEL,BMODEL                             FSHPE 03
      DIMENSION  CPMK(10, 24), CPMW(10, 24), CPMT(10), TEMP(24)         FSHPE 04
      DATA FMODEL/8H040171  /,FPAR/.45,.9,1.,1.499E11,315.5,1.499E9/    FSHPE 05
      DATA  KNOTS /10/, NFK /24/                                        FSHPE 06
      DATA  ( CPMT(I), I = 1, 10 )  /                          1.609,   FSHPE 07
     X 2.227, 2.845, 3.463, 4.081, 4.699, 5.317, 5.934, 6.552, 7.170/   FSHPE 08
      DATA  (FX(I), I = 1, 24 )  / .44,.46,.48,.50,.52,.54,.56,.58,.60, FSHPE 09
     X      .62,.64,.66,.68,.70,.72,.74,.76,.78,.80,.82,.84,.86,.88,.90/ FSHPE 10
      DATA  ( CPMK(I), I =   1, 172 )  /                       11.480,  FSHPE 11
     X10.664, 9.899, 9.031, 8.109, 7.261, 6.710, 6.156, 5.527, 4.826,   FSHPE 12
     X11.417,10.620, 9.857, 9.005, 8.126, 7.328, 6.801, 6.280, 5.677,   FSHPE 13
     X 4.973,11.458,10.702, 9.996, 9.255, 8.361, 7.635, 7.173, 6.747,   FSHPE 14
     X 6.189, 5.520,11.698,11.100,10.487, 9.828, 8.843, 8.004, 7.541,   FSHPE 15
     X 7.191, 6.682, 6.048,12.031,11.531,10.999,10.332, 9.325, 8.197,   FSHPE 16
     X 7.575, 7.218, 6.715, 6.070,12.109,11.652,11.148,10.394, 9.447,   FSHPE 17
     X 8.223, 7.418, 6.959, 6.371, 5.589,11.882,11.395,10.852, 9.981,   FSHPE 18
```

```
     X 9.036, 7.907, 7.155, 6.638, 5.946, 4.955,11.508,10.882,10.269,
     X 9.322, 8.286, 7.271, 6.676, 6.129, 5.411, 4.425,11.373,10.577,
     X 9.879, 9.033, 7.885, 6.852, 6.254, 5.636, 4.928, 4.087,11.594,
     X10.819,10.164, 9.435, 8.231, 6.994, 6.215, 5.515, 4.770, 3.989,
     X11.801,11.155,10.598, 9.910, 8.778, 7.400, 6.447, 5.598, 4.728,
     X 3.895,11.769,11.214,10.719,10.028, 9.072, 7.724, 6.680, 5.678,
     X 4.660, 3.815,11.560,10.998,10.480, 9.774, 8.949, 7.782, 6.792,
     X 5.734, 4.713, 4.037,11.494,10.892,10.294, 9.567, 8.650, 7.631,
     X 6.749, 5.839, 5.104, 4.677,11.604,11.054,10.466, 9.759, 8.689,
     X 7.554, 6.725, 6.098, 5.661, 5.379,11.612,11.084,10.567, 9.850,
     X 8.870, 7.654, 6.861, 6.395, 6.103, 5.905,11.422,10.844,10.334,
     X 9.598, 8.789, 7.730, 6.996, 6.547, 6.316, 6.173,11.228,10.464/
      DATA  ( CPMK(I), I = 173, 240 )  /
     X 9.840, 9.075, 8.437, 7.638, 7.015, 6.499, 6.259, 6.127,11.188,
     X10.233, 9.400, 8.574, 7.979, 7.430, 6.895, 6.304, 5.945, 5.719,
     X11.259,10.153, 9.218, 8.322, 7.646, 7.200, 6.752, 6.130, 5.562,
     X 5.006,11.302,10.119, 9.177, 8.273, 7.533, 7.037, 6.633, 6.031,
     X 5.282, 4.313,11.280,10.107, 9.148, 8.270, 7.510, 6.948, 6.485,
     X 5.862, 5.028, 3.874,11.209,10.110, 9.161, 8.282, 7.527, 6.878,
     X 6.328, 5.591, 4.702, 3.472,11.132,10.125, 9.197, 8.265, 7.536,
     X 6.872, 6.235, 5.376, 4.425, 3.238/
      DATA  ( CPMW(I), I =   1, 172 )  /                          .246,
     X  .246, -.440, -.105,  .012, 1.228, -.273, -.175, -.196, -.196,
     X  .185,  .185, -.393, -.020,  .048, 1.106, -.202, -.207, -.274,
     X -.274,  .155,  .155,  .017, -.767,  .635,  .870,  .036, -.457,
     X -.262, -.262, -.089, -.089,  .214,-1.494,  .638, 1.242,  .297,
     X -.638, -.265, -.265, -.053, -.053, -.242,-1.100, -.699, 1.987,
     X  .713, -.673, -.311, -.311,  .045,  .045, -.959, -.142,-1.495,
     X 1.753, 1.085, -.659, -.480, -.480,  .101,  .101,-1.404,  .368,
     X-1.221, 1.613,  .686, -.654, -.812, -.812,  .322,  .322,-1.417,
     X  .103, -.394, 1.791, -.150, -.448, -.751, -.751,  .386,  .386,
     X -.386,-1.168,  .308, 1.749, -.480, -.134, -.394, -.394,  .359,
     X  .359,  .087,-1.859, -.129, 1.861, -.123, -.123, -.090, -.090,
     X  .329,  .329, -.246,-1.404,-1.111, 1.968, -.063, -.097,  .134,
     X  .134,  .345,  .345, -.762, -.383,-1.875, 1.721, -.231, -.148,
     X  .577,  .577,  .307,  .307, -.859,  .175,-1.702, 1.263, -.586,
     X  .024, 1.078, 1.078,  .091,  .091, -.396, -.532, -.462,  .771,
     X -.458,  .598,  .844,  .844, -.101, -.101, -.104,-1.348, -.208,
     X 1.163,  .359,  .567,  .374,  .374,  .176,  .176, -.696, -.549,
     X-1.233, 1.785,  .726,  .454,  .202,  .202,  .429,  .429,-1.079,
     X  .344,-1.446, 1.498,  .575,  .682,  .136,  .136,  .633,  .633/
      DATA  ( CPMW(I), I = 173, 240 )  /
     X -.974, 1.033,-1.142,  .999, -.094, 1.072,  .128,  .128,  .456,
     X  .456, -.349, 1.034, -.139,  .222, -.518,  .983,  .221,  .221,
     X  .572,  .572, -.168,  .720,  .725,  .006, -.788,  .422, -.050,
     X -.050,  .790,  .790, -.172,  .501,  .740,  .379, -.815, -.217,
     X -.646, -.646,  .658,  .658,  .082,  .290,  .610,  .389, -.613,
     X -.448, -.918, -.918,  .465,  .465,  .046,  .437,  .165,  .564,
     X -.852, -.116,-1.051,-1.051,  .314,  .314, -.319,  .890, -.054,
     X  .361, -.977,  .066, -.751, -.751/
  125 AA = ALOG( T )
      DO  130  J = 1, NFK
      CALL  SPLIN2  ( CPMT, CPMK(1,J), KNOTS, CPMW(1,J), AA,  TEMP )
  130 FY(J) = EXP( TEMP(1) )
      CALL  SPLIN1  ( FX, FY, NFK, FW, TEMP, 2, 1.0, 1.0 )
      RETURN
      END
```

```
      SUBROUTINE  READ HP  ( ID, LST, JOB )
      COMMON  /DATA/  N, X(1030), Y(1030), DUM(10)
      DIMENSION  ID(1), IH(10), IC(10), JOB(1)
      DATA  ( IH(I),I=1,10)/1H0,1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9/
      DATA  ZERO/1.E-6/, SPACE/1.E6/, FOLD/9.85E5/
      LOGICAL  LIST
    1 FORMAT( 10A8 )
    2 FORMAT( 1X, I4, 1X, 10F6.0, 6X, A8 )
    3 FORMAT( 1H1,10A8 // 3X, 4H IND, 8X, 10A10 )
    4 FORMAT( 1H  )
    5 FORMAT( 1X, I6, 10I10, 4X, A8 )
    6 FORMAT( 16X, 10A8 / 86X3A10 * SERIAL* I3 )
    7 FORMAT(/* COUNT TIME         * F7.0,6X* DEAD TIME          * F7.0
     X /* ZERO ENERGY AT      * F7.0,6X  * CHANNEL FOR CS-137 * F7.0
     X /* MINIMUM CHANNEL     * F7.0,6X * MAXIMUM CHANNEL    * F7.0 )
      LIST = .FALSE.
      IF  ( LST .GT. 1 )  LIST = .TRUE.
      READ(10,1) ( ID(I), I = 1, 10 )
      IF  ( LIST )  WRITE (9,3) ( ID(I), I=1,10 ), IH
      N = 0
      NG = 0
      IL = 0
      READ(10,2) II, DUM, IE
      GO TO  120
  110 READ(10,2) II, DUM, IE
      IF  ( II .LT. 1 )  GO TO  180
  120 K2 = ( II/10 + 1 ) * 10 - II
      DO  130  I = 1, K2
      JJ = II - 1 + I
      IF  ( JJ .GT. 1023 .OR. N .GT. 1024 )  GO TO  135
      N = N + 1
      X(N) = JJ
      Y(N) = DUM(I)
  130 IF  ( DUM(I) .GT. FOLD )  Y(N) = DUM(I) - SPACE
  135 IF  ( .NOT. LIST )  GO TO  110
      DO  140  I = 1, 10
  140 IC(I) = DUM(I) + ZERO
      IF  ( NG .NE. 5 )  GO TO  150
      WRITE(9,3) ( ID(I), I=1,10 ), IH
      NG = 6
  150 IF  ( IL .LT. 10 )  GO TO  160
      WRITE(9,4)
      IL = 0
  160 WRITE(9,5) II, ( IC(I), I = 1, 10 ), IE
      IL = IL + 1
      IF  ( IL .EQ. 10 )  NG = NG + 1
      GO TO  110
  180 IF  ( DUM(6) .LE. ZERO )  DUM(6) = 1000.0
      IF ( DUM(1) .LE. 0.0 )  DUM(1) = 1.0
      IF ( LIST ) WRITE(9, 7) (DUM(I), I=1,6 )
      NN = N
      N = 0
      DO  190  I = 1, NN
      IF  ( X(I).LT.DUM(5) .OR. X(I).GT.DUM(6) )  GO TO  190
      N = N + 1
      X(N) = X(I)
```

```
      Y(N) = Y(I)                                                       READ  57
  190 CONTINUE                                                          READ  58
      CALL  GPLOT  ( 1, N, X, Y, 38, 0 )                                READ  59
      CALL  WLCV  ( 60, 910,18,18HCOUNTS PER CHANNEL, 1 )               READ  60
      CALL  WLCH  ( 896, 945, 7, 7HCHANNEL, 1 )                         READ  61
      CALL  LINCNT  ( 60 )                                              READ  62
      WRITE(12,6) ( ID(I), I = 1, 10 ), ( JOB(I),I=1,4 )                READ  63
      RETURN                                                            READ  64
      END                                                               READ  65
```

```
      SUBROUTINE   BPART                                                BPART 01
      COMMON  /DATA/  N, X(1030), Y(1030), DUM(10)                      BPART 02
      COMMON  /BSPLIN/  NBK, BPAR(2), BX(50), BY(50), BW(50)            BPART 03
      DIMENSION  TEMP(3)                                                BPART 04
      SPA = DUM(1) * BPAR(2) / BPAR(1) / (DUM(4)-DUM(3))                BPART 05
      SPB = 0.662 / DUM(4)                                              BPART 06
      DO  110  I = 1, N                                                 BPART 07
      EE = (X(I) - DUM(3) ) * SPB                                       BPART 08
      CALL  SPLIN2  ( BX, BY, NBK, BW,  EE , TEMP(2) )                  BPART 09
  110 Y(I) = Y(I) - TEMP(1) * SPA                                       BPART 10
      RETURN                                                            BPART 11
      END                                                               BPART 12
```

```
      SUBROUTINE MATCH ( NCH, NK, XK, YK, WK, PAR, ASSAY, SDEV, ID,JOB )MATCH 01
      DIMENSION  XX(1032),YY(1032),PAR(1),XK(1),YK(1),WK(1),ID(1),JOB(1)MATCH 02
      COMMON /PAUX/ P(3),SP(3),BM(3,4),NP,EP(2),XP(50),YP(50),WP(50)    MATCH 03
      COMMON  /DATA/  N, X(1030), Y(1030), DUM(10)                      MATCH 04
      DIMENSION  IND(2)                                                 MATCH 05
      DATA  (IND(I),I=1,2) / 9HCONVERGED ,  6HFAILED /                  MATCH 06
    1 FORMAT( 16X*ADJUSTED FIT*/ 16X10A8/ 81X2A10,A12*  SERIAL* I3 )    MATCH 07
    2 FORMAT(1H1////1X10A8///4X4HP(1),8X4HP(2),8X4HP(3),5X10HITERATIONS)MATCH 08
    3 FORMAT( / 2X,1PE10.2, 2E12.4, I7, 2X, A9 )                        MATCH 09
    4 FORMAT( /4X 39HP(3) * SPLINE(  P(2) * ( X - P(1) )   ) )          MATCH 10
      NP = NK                                                           MATCH 11
      EP(1) = PAR(1)                                                    MATCH 12
      EP(2) = PAR(2)                                                    MATCH 13
      DO  105  I = 1, NK                                                MATCH 14
      XP(I) = XK(I)                                                     MATCH 15
      YP(I) = YK(I)                                                     MATCH 16
  105 WP(I) = WK(I)                                                     MATCH 17
      P(1) = DUM(3)                                                     MATCH 18
      P(2) = 0.662 / ( DUM(4) - P(1) )                                  MATCH 19
      P(3) = 1.0                                                        MATCH 20
      WRITE(9,2) ( ID(I), I = 1, 10 )                                   MATCH 21
      WRITE(9,3) P                                                      MATCH 22
      CALL  ASSAYP  ( 1, 2, IT, II )                                    MATCH 23
      WRITE(9,3) P, IT, IND(II)                                         MATCH 24
      CALL  ASSAYP  ( 1, 3, IT, II )                                    MATCH 25
      WRITE(9,3) P, IT, IND(II)                                         MATCH 26
      CALL  ASSAYP  ( 2, 3, IT, II )                                    MATCH 27
```

```
      WRITE(9,3) P, IT, IND(II)
      CALL  ASSAYP  ( 1, 0, IT, II )
      WRITE(9,3) P, IT, IND(II)
      CALL  ASSAYP  ( 0, 0, IT, II )
      WRITE(9,3) P, IT, IND(II)
      WRITE(9,4)
      DUM(4) = 0.662 / P(2) + P(1)
      NCH = 0
      ASSAY = PAR(4)*PAR(3)/DUM(1)*0.662/P(2)*P(3)/PAR(5)
      SDEV = ASSAY*SQRT( (1.0/P(2))**2*BM(2,3) + (1.0/P(3))**2*BM(3,4)
     X                    +1.0/P(2)/P(3)*BM(2,4) + (PAR(6)/PAR(4))**2 )
      DO  110  I = 1, N
      X(I) = ( X(I) - P(1)) * P(2)
      IF  ( X(I) .LT. PAR(1) .OR. X(I) .GT. PAR(2) )  GO TO  110
      NCH = NCH + 1
      XX(NCH) = X(I)
      YY(NCH) = Y(I)
  110 CONTINUE
      CALL  GPLOT  ( 1, NCH, XX, YY, 38, 0 )
      XMIN = XX(1)
      XMAX = XX(1)
      DO 120 I = 1, NCH
  120 IF  ( XX(I) .GT. XMAX )  XMAX = XX(I)
      IF  ( XX(I) .LT. XMIN )  XMIN = XX(I)
      DD = (XMAX - XMIN ) / 101.0
      XX(1) = XMIN
      DO 125 I = 1, 101
      CALL  SPLIN2  ( XK, YK, NK, WK, XX(I), YY(I) )
      YY(I) = YY(I) * P(3)
  125 XX(I+1) = XX(I) + DD
      CALL  GPLOT  ( 20, 101, XX, YY, 48, 1 )
      CALL  WLCH  ( 896, 945, 7, 7H  M E V, 1 )
      CALL  WLCV  ( 60, 910,18,18HCOUNTS PER CHANNEL, 1 )
      CALL  LINCNT  ( 60 )
      WRITE(12,1) ( ID(I),I=1,10),(JOB(I),I=1,4)
      RETURN
      END




      SUBROUTINE  ASSAYP ( IX1, IX2, IT, IND )
C  PARAMETER ADJUSTING SUBROUTINE MODIFIED ESPECIALLY FOR ASSAY
      COMMON  /DATA/  N, X(1030), Y(1030), DUM(10)
      COMMON /PAUX/ P(3),SP(3),BM(3,4),NP,EP(2),XP(50),YP(50),WP(50)
      DIMENSION DYDP(3),AN(3),TMP(100),IX(2),W(1030),AM(3,3),DY(1030)
      LOGICAL  NOFIX, DONE, KLT2
      DATA  H/1.0/, TEST/0.000001/
C
  100 DONE=.FALSE.
      IM = 0
      IF  ( IX1 .LT. 1 )  GO TO  105
      IM = 1
      IX(1) = IX1
  105 IF  ( IX2 .LT. 1 )  GO TO  110
      IM = IM + 1
      IX(IM) = IX2
  110 K = 3 - IM
      NOFIX=IM.LT.1
```

```
      KLT2=K.LT.2
      IT=0
      IND = 1
      KP=K+1
  150 IT=IT+1
      DO160I=1,3
      DO160J=1,4
      IF(J.LT.KP)AM(I,J)=0.0
      IF(I+1.EQ.J)GOTO155
      BM(I,J)=0.0
      GOTO160
  155 BM(I,J)=1.0
  160 CONTINUE
      DUM(5) = EP(1) / P(2)  + P(1)
      DUM(6) = EP(2) / P(2)  + P(1)
      DO  180  I = 1, N
      W(I) = Y(I)
  180 IF  ( X(I).LT.DUM(5) .OR. X(I).GT.DUM(6) )  W(I) = 0.0
  200 DO230L=1,N
      ARG = ( X(L) - P(1) ) * P(2)
      CALL  SPLIN2  ( XP, YP, NP, WP, ARG, TMP )
      DYDP(1) = - TMP(2) * P(2) * P(3)
      DYDP(2) =  TMP(2) * (X(L)-P(1)) * P(3)
      DYDP(3) = TMP(1)
      DY(L) = Y(L) - TMP(1) * P(3)
      J = 0
      DO215JUK=1,3
      IF(NOFIX)GOTO 210
      DO205JOKE=1,IM
  205 IF(JUK.EQ.IX(JOKE))GOTO215
  210 J = J + 1
      AN(J) = DYDP(JUK)
  215 CONTINUE
      DO225I=1,K
      DO 225 J=I,KP
      IF(J.EQ.KP)GOTO 220
      AM(I,J)=AM(I,J)+AN(I)*AN(J)*W(L)
      GO TO 225
  220 BM(I,1)=BM(I,1)+AN(I)*DY(L)*W(L)
  225 CONTINUE
  230 CONTINUE
      AM(2,1) = AM(1,2)
      AM(3,1) = AM(1,3)
      AM(3,2) = AM(2,3)
  300 IF  ( KLT2 )  GO TO  305
      CALL  LSS  ( K, KP, 3, AM, BM, TMP, DET )
      GOTO310
  305 BM(1,1) = BM(1,1)/AM(1,1)
      BM(1,2) = 1.0/AM(1,1)
  310 J = 0
      DO 330 I=1,3
      IF(NOFIX)GOTO 320
      DO315JOKE=1,IM
  315 IF(I.EQ.IX(JOKE))GOTO330
  320 J = J + 1
      IF  ( ABS ( TEST * P(I) ) .LT. ABS( BM(J,1) ) )  GO TO  325
      DONE = .TRUE.
  325 P(I) = P(I) + H * BM(J,1)
  330 CONTINUE
  400 IF  ( .NOT. DONE .AND. IT .LT. 25 )  GO TO  150
```

```
      IF  ( .NOT. DONE )  IND = 2
      IF  (  IM .GT. 1 )  GO TO  425
      VAR = 0.0
      WVAR = 0.0
      IF  ( N .LT. 4 )  GO TO  410
      DO  405  L = 1, N
  405 VAR = VAR + W(L) * DY(L) ** 2
      WVAR = VAR / FLOAT(N-3)
  410 DO 420 I = 1,3
      DO  415  J = 2, 4
  415 BM(I,J) =BM(I,J) * WVAR
  420 SP(I) = SQRT (BM(I,I+1) )
  425 RETURN
      END
```

```
      SUBROUTINE SPLIN1(X,F,N,W,G,IND,XM1,XMN)
      DIMENSION X(1),F(1),W(1),G(1)
      GO TO (15,16,200),IND
   15 W(1) =(X(2)-X(1))/3.
      G(1) =((F(2)-F(1))/(X(2)-X(1))-XM1)/W(1)
      NDIC =1
      GO TO 17
   16 W(2) =((X(2)-X(1))*(1.+XM1/2.)+X(3)-X(2))/3.
      G(2) =((F(3)-F(2))/(X(3)-X(2))-(F(2)-F(1))/(X(2)-X(1)))/W(2)
      NDIC =2
      GO TO 17
  200 G(1)=XM1
      W(1)=1.
      W(2) =(X(3)-X(1))/3.
      G(2) =((F(3)-F(2))/(X(3)-X(2))-(F(2)-F(1))/(X(2)-X(1))
     1-(X(2)-X(1))/6.*G(1))/W(2)
      NDIC =2
   17 K =N-NDIC
      J =NDIC+1
   30 DO 165 I=J,K
      W(I) =(X(I+1)-X(I-1))/3.-((X(I)-X(I-1))**2.)/36./W(I-1)
  165 G(I) =((F(I+1)-F(I))/(X(I+1)-X(I))-(F(I)-F(I-1))/(X(I)-X(I-1))
     1-(X(I)-X(I-1))/6.*G(I-1))/W(I)
      GO TO (53,54,55),IND
   53 W(N) =(X(N)-X(N-1))/3.*(1.-(X(N)-X(N-1))/12./W(N-1))
      W(N) =(XMN-(F(N)-F(N-1))/(X(N)-X(N-1))-(X(N)-X(N-1))*G(N-1)/6.
     1)/W(N)
      GO TO 18
   54 W(N-1) =((X(N)-X(N-1))*(1.+XMN/2.)+(X(N-1)-X(N-2)))/3.
     1-(X(N-1)-X(N-2))/36.*(X(N-1)-X(N-2))/W(N-2)
      W(N-1) =((F(N)-F(N-1))/(X(N)-X(N-1))-(F(N-1)-F(N-2))/(X(N-1)-X(N-2
     1))-(X(N-1)-X(N-2))/6.*G(N-2))/W(N-1)
      GO TO 18
   55 IF(NDIC-1)60,60,56
   56 J =N-1
      K =N-1
      NDIC =1
      GO TO 30
   60 G(N)=1.
      W(N)=XMN
   18 I =N-NDIC
```

```
 19 K=2*NDIC                                              SPLN1 42
 20 DO 156 J=K,N                                          SPLN1 43
    W(I) =G(I)-(X(I+1)-X(I))/W(I)* W(I+1)/6.              SPLN1 44
156 I=I-1                                                 SPLN1 45
    GO TO(22,52,57),IND                                   SPLN1 46
 57 W(1)=XM1                                              SPLN1 47
    GO TO 22                                              SPLN1 48
 52 W(1) =XM1*W(2)                                        SPLN1 49
    W(N) =XMN*W(N-1)                                      SPLN1 50
 22 RETURN                                                SPLN1 51
    END                                                   SPLN1 52
```

```
      SUBROUTINE SPLIN2(X,F,N,W,Y,TAB)                                  SPLN2 01
      DIMENSION X(N),F(N),W(N),TAB(3)                                   SPLN2 02
      DATA  T/0.33333333/                                               SPLN2 03
      IF  ( N .LT. 2 )  GO TO  80                                       SPLN2 04
      I = 1                                                             SPLN2 05
      K = N                                                             SPLN2 06
10    J = ( I + K ) / 2                                                 SPLN2 07
      IF  ( I .EQ. J )  GO TO  70                                       SPLN2 08
      IF  ( X(J) .LT. Y )  GO TO  20                                    SPLN2 09
      K = J                                                             SPLN2 10
      GO TO  10                                                         SPLN2 11
20    I = J                                                             SPLN2 12
      GO TO  10                                                         SPLN2 13
70    EL = X(K) - X(I)                                                  SPLN2 14
      DX1 = X(K) - Y                                                    SPLN2 15
      REL = 1.0 / EL                                                    SPLN2 16
      EL = EL * 0.16666666666667                                        SPLN2 17
      DX2 = Y - X(I)                                                    SPLN2 18
      F1 = DX1 * W(I) * REL                                             SPLN2 19
      F2 = DX2 * W(K) * REL                                             SPLN2 20
      TAB(3) =F1+F2                                                     SPLN2 21
      F3 = -F1 * 0.5 * DX1                                              SPLN2 22
      F4 = F2 * 0.5 * DX2                                               SPLN2 23
      F5 =  F(I)  *REL - EL * W(I)                                      SPLN2 24
      F6 = F(K) *REL - EL * W(K)                                        SPLN2 25
      TAB(2)=F3+F4-F5+F6                                                SPLN2 26
      TAB(1) = ( F5 - F3 * T ) * DX1 + ( F6 + F4 * T ) * DX2            SPLN2 27
      RETURN                                                            SPLN2 28
   80 TAB(1) = F(1)                                                     SPLN2 29
      TAB(2) = 0.0                                                      SPLN2 30
      TAB(3) = 0.0                                                      SPLN2 31
      RETURN                                                            SPLN2 32
      END                                                               SPLN2 33
```

APPENDIX C
SUBROUTINE GPLOT

This is a general subroutine developed by the author to plot on film. It became incorporated naturally into the ASSAY program, and therefore a listing and brochure are included for convenience to the reader.

Purpose

The subroutine will draw a suitably scaled rectangular grid and plot N points on film to any combination of log and linear scales. The controlling program can either assign axial boundary values or let the subroutine determine them for optimum use of the plotting space. Several sets of points can be plotted on one grid.

Usage

CALL GPLOT (IOP, N, X, Y, ICHAR, ICON)

where IOP ≤ 1 plots linear-linear,
- = 2 plots linear-log,
- = 3 plots log-linear,
- = 4 to 9 plots log-log;
- add 10 for new plot with previously used boundaries,
- 20 to plot another set of points on the previous plot;
- = 30 means to find the ranges of coordinate values;

N = number of points to be plotted;

X and Y are the single-indexed coordinates of the points;

ICHAR is the decimal integer code for the plotting symbol;

ICON = 0 means do not connect the points,
= 1 means connect the points.

General Information

By making appropriate use of IOP, the programmer can also plot several sets of points on one grid, or on separate but identical grids, or plot points without a grid. A simplified flow chart is included as a quick guide in Fig. C-1.

The four interdependent task assignments (TA) for which the subroutine can be employed are indicated by the tens digit of IOP.

TA-1 (0 ≤ IOP ≤ 9) will advance the film one frame, adjust boundary values for best use of the plotting area, draw and scale a grid, and plot N points. The frame is left in readiness for additional plots via TA-3, writing on film, etc. See the template in Fig. C-2. The boundary values will be determined by the results from TA-4 only if TA-1 is assigned next after TA-4.

TA-2 (10 ≤ IOP ≤ 19) differs from TA-1 only in that the boundary values previously stored in the subroutine memory will be used. The programmer can plot various sets of points on several identical grids by successive re-entry with TA-2 and choosing the same plotting mode each time.

TA-3 (20 ≤ IOP ≤ 29) will plot an additional set of points on the existing frame if it follows TA-1, -2, or -3, and the units digit of IOP will be ignored. On entry next after TA-4, the points will be plotted on a new frame with a grid and scaling subject to the results from TA-4, and the units digit of IOP will be observed, as if entered with TA-1.

TA-4 (30 ≤ IOP) will select and retain maximum and minimum coordinate values without plotting. An unbroken succession of entries with TA-4 for several sets of points will adjust the values to accomodate all the sets. These values will be used on first entry with any other TA and then forgotten.

Whenever a point with a zero or negative coordinate value appears, the subroutine will generally veto a log scale request and program a linear scale for that coordinate. An exception can occur with TA-3 after TA-1, -2, or -3 without the benefit of results from TA-4, and which would lead to a job abort.

Points falling outside the outer boundary will be plotted on that boundary.

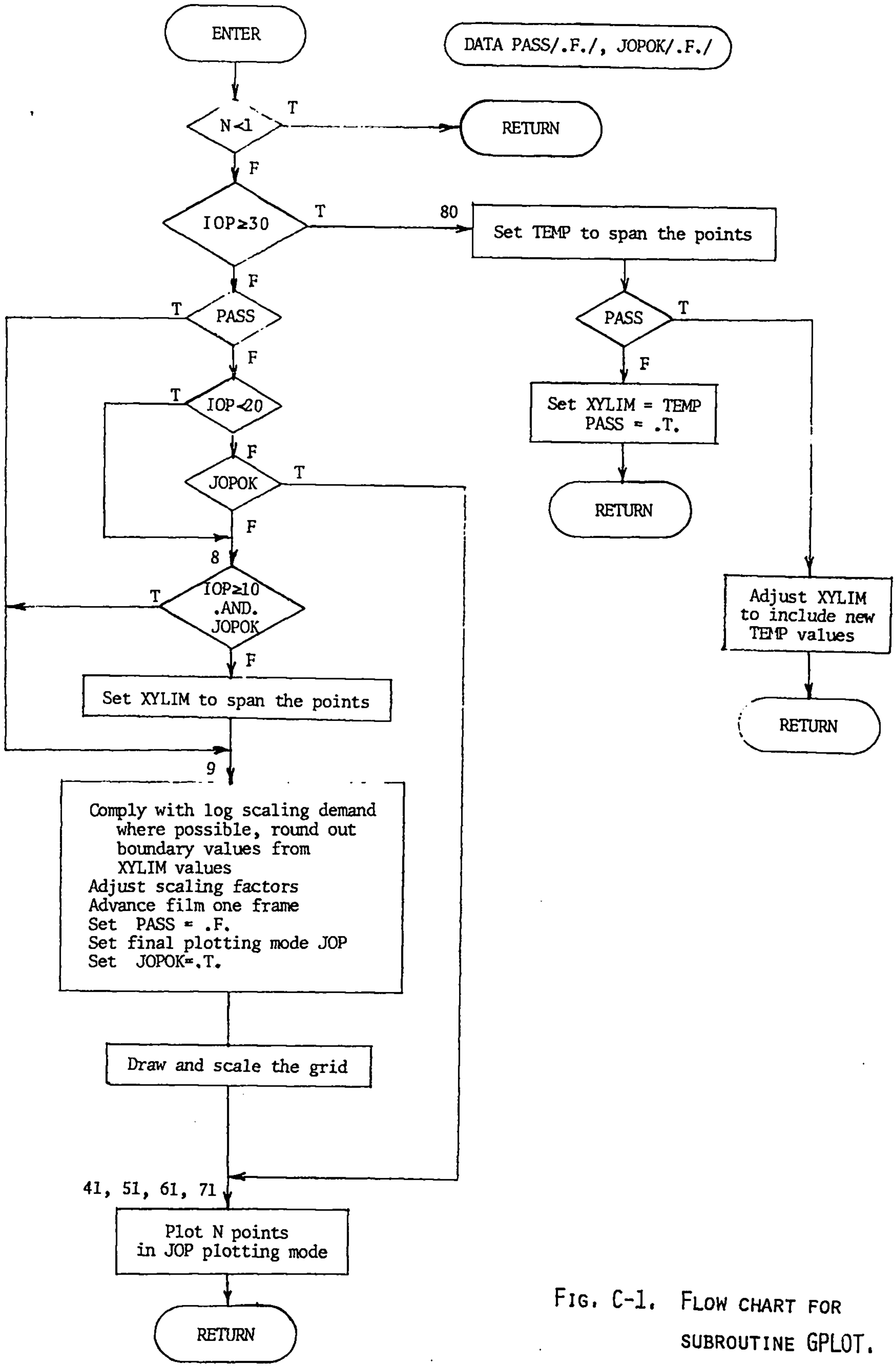

FIG. C-1. FLOW CHART FOR SUBROUTINE GPLOT.

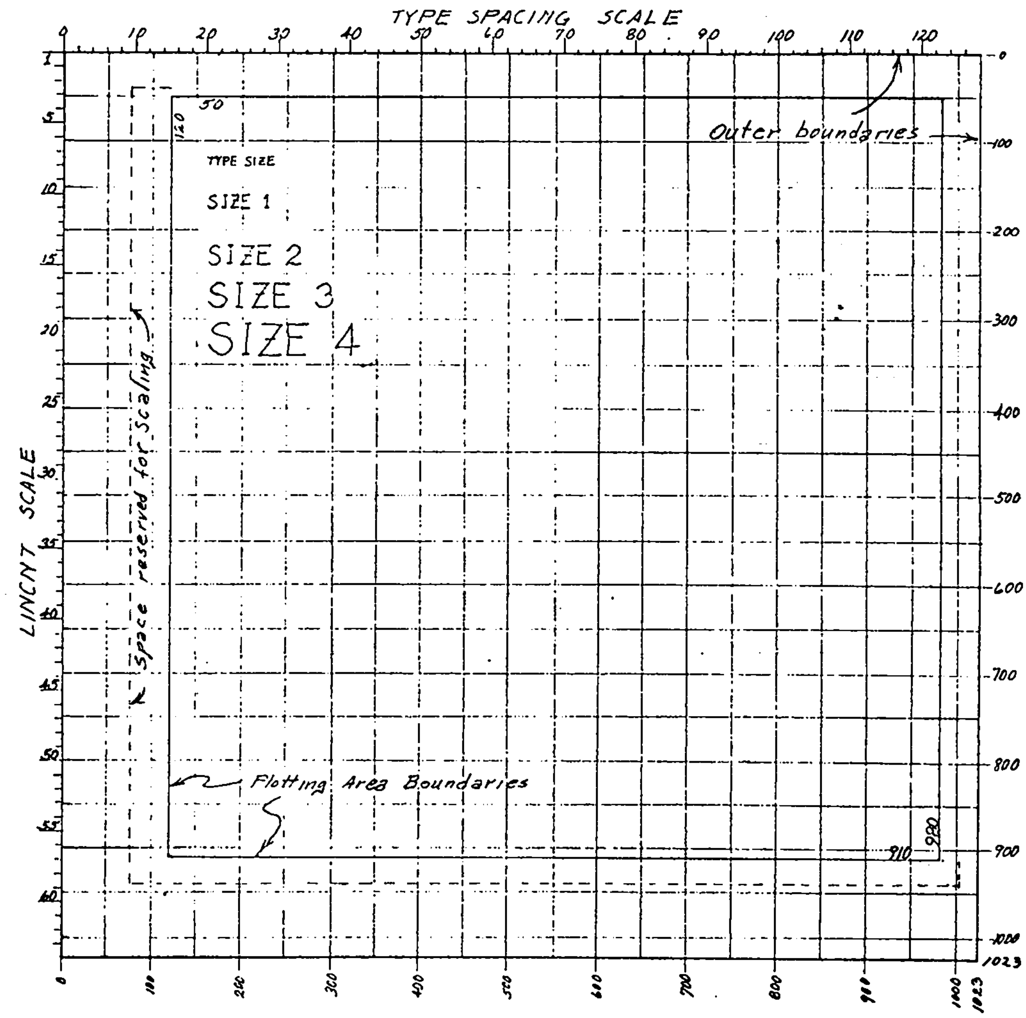

Fig. C-2. Template for additions
to a film plot.

```
      SUBROUTINE  GPLOT  ( IOP, N, X, Y, ICHAR, ICON )                  GPLOT001
C     GPLOT -- GENERAL PLOTTING SUBROUTINE  F4  VERSION  DEC. 1, 1966   GPLOT002
C     THE SIX ASSOCIATED SUBROUTINES ARE                                GPLOT003
C           1.  GPLOTA, FOR MAXIMUM AND MINIMUM VALUES OF COORDINATE    GPLOT004
C           2.  GPLOTB, TO ROUND OUT LOGARITHMS OF THE PLOTTING LIMITS  GPLOT005
C           3.  GPLOTC, TO ROUND OUT THE LINEAR PLOTTING LIMITS         GPLOT006
C           4.  GPLOTD, TO DRAW AND LABEL THE GRID                      GPLOT007
C           5.  GPLOTE, TO KEEP THE PLOTTING POINT WITHIN THE OUTER FRAMEGPLOT008
C           6.  GPLOTF, TO INSCRIBE A LOGARITHMIC SCALING               GPLOT009
C     ALL THE OTHER SUBROUTINES ARE CONTAINED IN THE LA-FLM1 PACKEAGE.  GPLOT010
      DIMENSION  X(N), Y(N), XYLIM(4), TEMP(4)                          GPLOT011
      LOGICAL  PXLIN, PYLIN, PASS, JOPOK                                GPLOT012
      DATA  PASS/.FALSE./, JOPOK/.FALSE./                               GPLOT013
      DATA  IXL/120/, IXR/980/, IYT/50/, IYB/910/                       GPLOT014
      IF  ( N .LT. 1 )  RETURN                                          GPLOT015
      IF  ( IOP .GE. 30 )  GO TO  80                                    GPLOT016
      IF  ( PASS )  GO TO  9                                            GPLOT017
      IF  ( IOP .LT. 20 ) GO TO  8                                      GPLOT018
      IF  ( JOPOK )  GO TO  ( 41, 51, 61, 71 ),JOP                      GPLOT019
    8 IF  ( IOP .GE. 10 .AND. JOPOK )  GO TO  9                         GPLOT020
      CALLGPLOTA(N,X,XYLIM(1),XYLIM(2))                                 GPLOT021
      CALLGPLOTA(N,Y,XYLIM(3),XYLIM(4))                                 GPLOT022
    9 JOP = IOP - (IOP/10) * 10                                         GPLOT023
   10 PXLIN=.TRUE.                                                      GPLOT024
      PYLIN=.TRUE.                                                      GPLOT025
      IF(JOP.GT.2)PXLIN=.FALSE.                                         GPLOT026
      IF(JOP.EQ.2.OR.JOP.GT.3)PYLIN=.FALSE.                             GPLOT027
   12 IF(PXLIN)GOTO14                                                   GPLOT028
      IF(XYLIM(1).LE.0.0)GOTO13                                         GPLOT029
      CALLGPLOTB(XYLIM(2),XYLIM(1),XR,XL,JX)                            GPLOT030
      GOTO22                                                            GPLOT031
   13 PXLIN=.TRUE.                                                      GPLOT032
   14 CALLGPLOTC(XYLIM(2),XYLIM(1),XR,XL,NX,KX,JX)                      GPLOT033
   22 IF(PYLIN)GOTO24                                                   GPLOT034
      IF(XYLIM(3).LE.0.0)GOTO23                                         GPLOT035
      CALLGPLOTB(XYLIM(4),XYLIM(3),YT,YB,JY)                            GPLOT036
      GOTO30                                                            GPLOT037
   23 PYLIN=.TRUE.                                                      GPLOT038
   24 CALLGPLOTC(XYLIM(4),XYLIM(3),YT,YB,NY,KY,JY)                      GPLOT039
 30   IF  ( JOPOK )  GO TO  32                                          GPLOT040
      SPANX = FLOAT( IXR - IXL )                                        GPLOT041
      SPANY = FLOAT( IYB - IYT )                                        GPLOT042
      BIASX = FLOAT( IXL ) + 0.4995                                     GPLOT043
      BIASY = FLOAT( IYT ) + 0.4995                                     GPLOT044
 32   AX = SPANX / ( XR - XL )                                          GPLOT045
      AY = SPANY / ( YT - YB )                                          GPLOT046
      CALLADV(1)                                                        GPLOT047
      PASS = .FALSE.                                                    GPLOT048
      JOP=4                                                             GPLOT049
      IF(PXLIN)JOP=2                                                    GPLOT050
      IF(PYLIN)JOP=JOP-1                                                GPLOT051
      JOPOK = .TRUE.                                                    GPLOT052
      CALL        DGA (IXL,IXR,IYT,IYB,XL,XR,YT,YB )                    GPLOT053
      CALL     GPLOTD (IXL,IXR,IYT,IYB,XL,XR,YT,YB,JOP,NX,KX,NY,KY,JX,JY)GPLOT054
      GOTO(41,51,61,71),JOP                                             GPLOT055
   41 NN = N                                                            GPLOT056
      IF(ICON.GT.0)NN=1                                                 GPLOT057
      DO  42  I = 1, NN                                                 GPLOT058
      IX = BIASX + AX * ( X(I) - XL )                                   GPLOT059
      IY = BIASY + AY * ( YT - Y(I) )                                   GPLOT060
```

```
      CALL GPLOTE  ( IX, IY )
   42 CALL  PLT( IX, IY, ICHAR )
      IF  ( NN .EQ. N )  RETURN
      DO  43  I = 2, N
      IIX = IX
      IIY = IY
      IX = BIASX + AX * ( X(I) - XL )
      IY = BIASY + AY * ( YT - Y(I) )
      CALL GPLOTE  ( IX, IY )
      CALL  PLT( IX, IY, ICHAR )
   43 CALL  DRV( IIX, IIY, IX, IY )
      RETURN
   51 NN=N
      IF(ICON.GT.0)NN=1
      DO52I=1,NN
      IX = BIASX + AX * ( X(I) - XL )
      IY = BIASY + AY * ( YT - ALOG10( Y(I) ) )
      CALL GPLOTE  ( IX, IY )
   52 CALLPLT(IX,IY,ICHAR)
      IF  ( NN .EQ. N )  RETURN
      DO53I=2,N
      IIX=IX
      IIY=IY
      IX = BIASX + AX * ( X(I) - XL )
      IY = BIASY + AY * ( YT - ALOG10( Y(I) ) )
      CALL GPLOTE  ( IX, IY )
      CALLPLT(IX,IY,ICHAR)
   53 CALLDRV(IIX,IIY,IX,IY)
      RETURN
   61 NN=N
      IF(ICON.GT.0)NN=1
      DO62I=1,NN
      IX = BIASX + AX * ( ALOG10( X(I) ) - XL )
      IY = BIASY + AY * ( YT - Y(I) )
      CALL GPLOTE  ( IX, IY )
   62 CALLPLT(IX,IY,ICHAR)
      IF  ( NN .EQ. N )  RETURN
      DO63I=2,N
      IIX=IX
      IIY=IY
      IX = BIASX + AX * ( ALOG10( X(I) ) - XL )
      IY = BIASY + AY * ( YT - Y(I) )
      CALL GPLOTE  ( IX, IY )
      CALLPLT(IX,IY,ICHAR)
   63 CALLDRV(IIX,IIY,IX,IY)
      RETURN
   71 NN=N
      IF(ICON.GT.0)NN=1
      DO72I=1,NN
      IX = BIASX + AX * ( ALOG10( X(I) ) - XL )
      IY = BIASY + AY * ( YT - ALOG10( Y(I) ) )
      CALL GPLOTE  ( IX, IY )
   72 CALLPLT(IX,IY,ICHAR)
      IF  ( NN .EQ. N )  RETURN
      DO73I=2,N
      IIX=IX
      IIY=IY
      IX = BIASX + AX * ( ALOG10( X(I) ) - XL )
      IY = BIASY + AY * ( YT - ALOG10( Y(I) ) )
      CALL GPLOTE  ( IX, IY )
```

```
      CALLPLT(IX,IY,ICHAR)                                              GPLOT121
   73 CALLDRV(IIX,IIY,IX,IY)                                            GPLOT122
      RETURN                                                            GPLOT123
80    CALL   GPLOTA   ( N, X, TEMP(1), TEMP(2) )                        GPLOT124
      CALL   GPLOTA   ( N, Y, TEMP(3), TEMP(4) )                        GPLOT125
      IF   ( PASS )   GO TO  82                                         GPLOT126
      DO   81   I = 1, 4                                                GPLOT127
   81 XYLIM(I) = TEMP(I)                                                GPLOT128
      PASS = .TRUE.                                                     GPLOT129
      RETURN                                                            GPLOT130
   82 DO   83   I = 1, 3, 2                                             GPLOT131
      IF   ( TEMP(I) .LT. XYLIM(I) )   XYLIM(I) = TEMP(I)               GPLOT132
   83 IF   ( TEMP(I+1) .GT. XYLIM(I+1) )  XYLIM(I+1) = TEMP(I+1)        GPLOT133
      RETURN                                                            GPLOT134
      END                                                               GPLOT135


      SUBROUTINE  GPLOTA ( N, X, XMIN, XMAX )                           GPLTA001
      DIMENSION  X(N)                                                   GPLTA002
      XMIN=X(1)                                                         GPLTA003
      XMAX=X(1)                                                         GPLTA004
      DO2I=1,N                                                          GPLTA005
      IF(X(I).GE.XMIN)GOTO1                                             GPLTA006
      XMIN=X(I)                                                         GPLTA007
      GOTO2                                                             GPLTA008
    1 IF(X(I).GT.XMAX)XMAX=X(I)                                         GPLTA009
    2 CONTINUE                                                          GPLTA010
      RETURN                                                            GPLTA011
      END                                                               GPLTA012


      SUBROUTINE  GPLOTB   ( XRR, XLL, XR, XL, J )                      GPLTB001
      IF   ( XRR .LT. XLL )   GO TO   4                                 GPLTB002
    1 C = 0.9999                                                        GPLTB003
      IF   ( XRR .LT. 1.0) C = -0.0001                                  GPLTB004
      TR= ALOG10( XRR)                                                  GPLTB005
      J = TR+ C                                                         GPLTB006
      XR=J                                                              GPLTB007
      C = 0.0001                                                        GPLTB008
      IF   ( XLL .LT. 1.0) C = -0.9999                                  GPLTB009
      T = ALOG10( XLL)                                                  GPLTB010
      J = T + C                                                         GPLTB011
      XL=J                                                              GPLTB012
      IF   ( XR - XL .GT. 4.0 )   GO TO   3                             GPLTB013
      IF   ( XR - TR .LE. 0.522878 )   GO TO 2                          GPLTB014
      XR=XR-0.522878                                                    GPLTB015
    2 IF(T-XL.LE.0.477121)GOTO3                                         GPLTB016
      XL=XL+0.477121                                                    GPLTB017
    3 IF(XR-XL.GT.25.0)XL=XR-25.0                                       GPLTB018
      IF   ( XR .LT. XL + 0.0001 )   XR = XL + 0.477122                 GPLTB019
      RETURN                                                            GPLTB020
    4 T=XRR                                                             GPLTB021
      XRR=XLL                                                           GPLTB022
      XLL=T                                                             GPLTB023
      GOTO1                                                             GPLTB024
      END                                                               GPLTB025
```

```
      SUBROUTINE  GPLOTC  ( XRR, XLL, XR, XL, NX, K, J )               GPLTC001
      IF  ( XRR .LT. XLL*1.001 )  GO TO  7                              GPLTC002
      XR = XRR                                                          GPLTC003
    1 XL = XLL                                                          GPLTC004
      KADD = 0                                                          GPLTC005
      T=ALOG(XR-XL)*0.434294                                            GPLTC006
      IF(T.LE.0.0) KADD = 1                                             GPLTC007
      J=T-FLOAT( KADD )                                                 GPLTC008
      IF  ( T .LT. 1.0 )  KADD = 1                                      GPLTC009
      TEN=10.0**(-J+1)                                                  GPLTC010
      IR = 0.999999 + XR * TEN                                          GPLTC011
      IL=XL*TEN+ 0.0000001                                              GPLTC012
      IF(XL.LT.0.0)GOTO5                                                GPLTC013
      IL=IL/5                                                           GPLTC014
    2 IR=(IR+4)/5                                                       GPLTC015
    3 NX=(IR-IL)                                                        GPLTC016
      IF(NX.GT.10)GOTO6                                                 GPLTC017
    4 AR=IR*5                                                           GPLTC018
      AL=IL*5                                                           GPLTC019
      XR=AR/TEN                                                         GPLTC020
      XL=AL/TEN                                                         GPLTC021
      A=XR                                                              GPLTC022
      IF(XL+XR.LT.0.0)A=-XL                                             GPLTC023
      K = KADD                                                          GPLTC024
      IF(J.LE.0)K=-J+KADD                                               GPLTC025
      C = 0.00001                                                       GPLTC026
      IF  ( A .LT. 1.0 )  C = -0.99999                                  GPLTC027
      J = ALOG10( A ) + C                                               GPLTC028
      IF(J.LT.5.AND.K.LT.5) RETURN                                      GPLTC029
      K = 10 + K                                                        GPLTC030
      IF  ( J .LT. 0 )  K = K + J                                       GPLTC031
      RETURN                                                            GPLTC032
    5 IL=(IL-5)/5                                                       GPLTC033
      IF(XR.GT.0.0)GOTO2                                                GPLTC034
      IR=IR/5                                                           GPLTC035
      GOTO3                                                             GPLTC036
6     IF  ( (IL/2)*2  .NE. IL )  IL = IL - 1                            GPLTC037
      IF  ( (IR/2)*2  .NE. IR )  IR = IR + 1                            GPLTC038
      NX = ( IR - IL ) / 2                                              GPLTC039
      KADD = 0                                                          GPLTC040
      GOTO4                                                             GPLTC041
7     IF  ( XRR.LT. XLL)  GO TO  9                                      GPLTC042
8     XR = XLL * 1.001                                                  GPLTC043
      GOTO1                                                             GPLTC044
9     T = XRR                                                           GPLTC045
      XRR = XLL                                                         GPLTC046
      XLL = T                                                           GPLTC047
      IF  ( XRR .LT. XLL*1.001 )  GO TO  8                              GPLTC048
      GO TW  1                                                          GPLTC049
      END                                                               GPLTC050
```

```
      SUBROUTINE  GPLOTD  ( IXL, IXR, IYT, IYB, XL, XR, YT, YB, JOP,     GPLTD001
     X                      NX, KX, NY, KY, JX, JY )                     GPLTD002
      DIMENSION  G(80), IG(171), NG1(5), NG2(5)                          GPLTD003
      DATA  ( G(I), I = 1, 16 ) /                                        GPLTD004
     X  .00000, .07918, .14613, .20412, .25527, .30103, .39794, .47712,  GPLTD005
     X  .54407, .60206, .65321, .69897, .77815, .84510, .90309, .95424/  GPLTD006
      DATA  ( IG(I), I = 1, 27 ) /                                       GPLTD007
     X 1,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,                         GPLTD008
     X 1,1,6,8,10,12,13,14,15,16/                                        GPLTD009
      DATA  ( NG1(I), I = 1, 3 )  /  1,  18, 1 /                         GPLTD010
      DATA  ( NG2(I), I = 1, 3 )  / 17,  27, 1 /                         GPLTD011
      IF  ( JOP .GT. 2 )  GO TO  100                                     GPLTD012
      CALL  EXL                                                          GPLTD013
      R = FLOAT( IXR - IXL ) / FLOAT (NX)                                GPLTD014
      K2 = NX + 1                                                        GPLTD015
      DO 20 LAP=1,2                                                      GPLTD016
      X = IXL                                                            GPLTD017
      DO  10  I = 1, K2                                                  GPLTD018
      IX = X + 0.4999                                                    GPLTD019
      X = X + R                                                          GPLTD020
   10 CALL  GYA  ( IYT, IYB, IX )                                        GPLTD021
      IF ( NX.GE.5 ) GOTO 30                                             GPLTD022
      K2 = NX*5  + 1                                                     GPLTD023
   20 R = R*0.2                                                          GPLTD024
   30 CALL  EXH                                                          GPLTD025
       CALL  SBLIN  ( NX, KX )                                           GPLTD026
      GO TO  200                                                         GPLTD027
  100 CYC = XR - XL                                                      GPLTD028
      T = JX                                                             GPLTD029
      R = FLOAT( IXR - IXL ) / CYC                                       GPLTD030
      NL = CYC + 1.9999                                                  GPLTD031
      ING = 1                                                            GPLTD032
      J = 1                                                              GPLTD033
      IY = IYB + 20                                                      GPLTD034
      IF  ( CYC .GT. 1.9 )  GO TO  115                                   GPLTD035
      K1 = NG1(2) + 2                                                    GPLTD036
      K2 = NG2(2)                                                        GPLTD037
      CALL  EXH                                                          GPLTD038
      TT = T                                                             GPLTD039
      DO  111  LAP = 1, NL                                               GPLTD040
      DO  110  I = K1, K2                                                GPLTD041
      J = J + 1                                                          GPLTD042
      K = IG(I)                                                          GPLTD043
      X = TT+ G(K)                                                       GPLTD044
      IF  ( X .LT. XL - 0.0001 )  GO TO  110                             GPLTD045
      IF  ( X .GT. XR + 0.0001 )  GO TO  115                             GPLTD046
      IX = ( X - XL ) * R + FLOAT( IXL ) + 0.4999                        GPLTD047
      CALL  PLT( IX, IY, J )                                             GPLTD048
  110 CONTINUE                                                           GPLTD049
      J = 1                                                              GPLTD050
  111 TT = TT + 1.0                                                      GPLTD051
  115 IF  ( CYC .GT. 1.9 ) ING = 2                                       GPLTD052
      IF  ( CYC .GT. 4.1 ) ING = 3                                       GPLTD053
      K1 = NG1(ING)                                                      GPLTD054
      K2 = NG2(ING)                                                      GPLTD055
      DO  135  I = 1, NL                                                 GPLTD056
      IF  ( T .LT. XL - 0.0001 )  GO TO  122                             GPLTD057
      CALL  EXH                                                          GPLTD058
      IX = ( T - XL ) * R + FLOAT( IXL) + 0.4999                         GPLTD059
      CALL  GPLOTF  ( IX, IY, JX, 2 )                                    GPLTD060
```

```
      CALL  EXL
122   DO  125  J = K1, K2
      K = IG(J)
      X = T + G(K)
      IF  ( X .LT. XL - 0.0001 )  GO TO  125
      IF  ( X .GT. XR + 0.0001 )  GO TO  200
      IX = ( X - XL ) * R + FLOAT(IXL) + 0.4999
      CALL  GYA  ( IYT, IYB, IX )
125   CONTINUE
130   JX = JX + 1
135   T = JX
200   IF  ( JOP .EQ. 2 .OR. JOP .GT. 3 )  GO TO  300
      CALL  EXL
      R = FLOAT( IYB - IYT ) / FLOAT( NY )
      K2 = NY + 1
      DO 220 LAP=1,2
      X = IYB
      DO  210  I = 1, K2
      IY = X + 0.4999
      X = X - R
210   CALL  GXA  ( IXR, IXL, IY )
      IF ( NY.GE.5 ) GOTO 230
      K2 = NY*5  + 1
  220 R = R*0.2
  230 CALL EXH
      CALL  SLLIN  ( NY, KY )
      RETURN
  300 CYC = YT - YB
      T = JY
      R = FLOAT( IYB - IYT ) / CYC
      NL = CYC + 1.9999
      ING = 1
      J = 1
      IX = IXL - 10
      IF  ( CYC .GT. 1.9 )  GO TO  315
      K1 = NG1(2) + 2
      K2 = NG2(2)
      CALL  EXH
      TT = T
      DO  311  LAP = 1, NL
      DO  310  I = K1, K2
      J = J + 1
      K = IG(I)
      X = TT+ G(K)
      IF  ( X .LT. YB - 0.0001 )  GO TO  310
      IF  ( X .GT. YT + 0.0001 )  GO TO  315
      IY = FLOAT( IYB ) + 0.4999 - ( X - YB ) * R
      CALL  PLT( IX, IY, J )
310   CONTINUE
      J = 1
311   TT = TT + 1.0
  315 IF  ( CYC .GT. 1.9 )  ING = 2
      IF  ( CYC .GT. 4.1 )  ING = 3
      K1 = NG1(ING)
      K2 = NG2(ING)
      DO  335  I = 1, NL
      IF  ( T .LT. YB - 0.0001 )  GO TO  322
      CALL  EXH
      IY = FLOAT(IYB ) + 0.4999 - ( T - YB ) * R
      CALL  GPLOTF  ( IX, IY, JY, 3 )
```

```
      CALL  EXL                                                        GPLTD121
  322 DO  325  J = K1, K2                                              GPLTD122
      K = IG(J)                                                        GPLTD123
      X = T + G(K)                                                     GPLTD124
      IF  ( X .LT. YB - 0.0001 )  GO TO  325                           GPLTD125
      IF  ( X .GT. YT + 0.0001 )  GO TO  400                           GPLTD126
      IY = FLOAT( IYB ) + 0.4999 - ( X - YB ) * R                      GPLTD127
      CALL  GXA ( IXL, IXR, IY )                                       GPLTD128
  325 CONTINUE                                                         GPLTD129
  330 JY = JY + 1                                                      GPLTD130
  335 T = JY                                                           GPLTD131
  400 CALL  EXH                                                        GPLTD132
      RETURN                                                           GPLTD133
      END                                                              GPLTD134


      SUBROUTINE  GPLOTE ( IX, IY )                                    GPLTE001
      IF  ( IX .LT. 0 )  IX = 0                                        GPLTE002
      IF  ( IX .GT. 1023 ) IX = 1023                                   GPLTE003
      IF  ( IY .LT. 0 )  IY = 0                                        GPLTE004
      IF  ( IY .GT. 1023 )  IY = 1023                                  GPLTE005
      RETURN                                                           GPLTE006
      END                                                              GPLTF001


      SUBROUTINE  GPLOTF  ( IX, IY, J, IS )                            GPLTF002
      IYY = IY - 12                                                    GPLTF003
      JJ = J                                                           GPLTF004
      M = 0                                                            GPLTF005
      IF  ( J .GE. 0 )  GO TO  1                                       GPLTF006
      JJ = - J                                                         GPLTF007
      M = 1                                                            GPLTF008
    1 K = 1                                                            GPLTF009
      L = 10                                                           GPLTF010
    2 IF  ( JJ .LT. L )  GO TO  3                                      GPLTF011
      L = L * 10                                                       GPLTF012
      K = K + 1                                                        GPLTF013
      GO TO  2                                                         GPLTF014
    3 IF  ( IS - 2 )  4, 5, 6                                          GPLTF015
    4 IXX = IX                                                         GPLTF016
      GO TW  7                                                         GPLTF017
    5 IXX = IX - ( 8 + 7 * ( K+M) ) / 2                                GPLTF018
      GO TO  7                                                         GPLTF019
    6 IXX = IX - 8 - 7 * (K+M)                                         GPLTF020
    7 CALL  TSP  ( IXX, IY, 2, 2H10 )                                  GPLTF021
      IXX = IXX + 14                                                   GPLTF022
      IF  ( M .EQ. 0 )  GO TO  8                                       GPLTF023
      CALL  TSP  ( IXX, IYY, 1, 1H- )                                  GPLTF024
      IXX = IXX + 7                                                    GPLTF025
    8 DO  9  I = 1, K                                                  GPLTF026
      L = L / 10                                                       GPLTF027
      M = JJ / L                                                       GPLTF028
      CALL  PLT  ( IXX, IYY, M )                                       GPLTF029
      JJ = JJ - M * L                                                  GPLTF030
    9 IXX = IXX + 7                                                    GPLTF031
      RETURN                                                           GPLTF032
      END                                                              GPLTF033
```

JJ:209(60)