IN-1/116

REPUBLICA ARGENTINA
COMISION NACIONAL DE ENERGIA ATOMICA
GERENCIA DE INVESTIGACIONES
DEPARTAMENTO DE FISICA NUCLEAR

AUTOMATIC ANALYSIS OF GAMMA-RAY SPECTRA ON SMALL COMPUTERS

E.Achterberg, F.C.Iglesias, A.E.Jech, J.A.Moragues,
M.Pérez, J.J.Rossi, W.Scheuer, and J.F.Suarez

Buenos Aires
1973

IN-1/116

REPUBLICA ARGENTINA
COMISION NACIONAL DE ENERGIA ATOMICA
GERENCIA DE INVESTIGACIONES
DEPARTAMENTO DE FISICA NUCLEAR

AUTOMATIC ANALYSIS OF GAMMA-RAY SPECTRA ON SMALL COMPUTERS

E. Achterberg, F. C. Iglesias, A. E. Jech, J. A. Moragues, M. Pérez, J. J. Rossi, W. Scheuer, and J. F. Suarez

Buenos Aires
1973

AUTOMATIC ANALYSIS OF GAMMA-RAY SPECTRA ON SMALL COMPUTERS.*

E.Achterberg, F.C.Iglesias, A.E.Jech†, J.A.Moragues†, M.Pérez†, J.J.Rossi, W.Scheuer and J.F.Suárez.

Departamento Física Nuclear. Comisión Nacional de Energía Atómica. Buenos Aires. Argentina.

ABSTRACT

A program for the automatic analysis of Ge(Li) gamma-ray spectra is described. Owing to its reduced size it can be used on small (at least 16 K memory words of 16 bits) computers. For the design of the program, the steps of manual methods of analysis have been followed as closely as possible. Most of the subroutines are written in Fortran II language.

* Work supported in part by the Argentine Scientific and Technical Research Council. Published in condensed form in IEEE Trans. Nucl.Sc. NS-19 (1972)3.

† Member of the Scientific Research Career of the Argentine Scientific and Technical Research Council.

CONTENTS

1. Introduction.

The automatic analysis of high-resolution gamma-ray spectra has been intensively studied in the years subsequent to the advent of Ge(Li) detectors. As a consequence, a number of theoretical approaches and computer programs are available today [1-5]. However, no automatic solution has yet been proposed for the analysis on small computers frequently included in data-acquisition systems.

In this paper, a computer code is described which was developed to be used on a computer of at least 16 K memory words of 16 bits (in our case, Hewlett-Packard 2116B). The spectrum to be analysed may have up to 4096 channels; the peaks must be not less than 4 channels wide at FWHM (see section 2.3).

The code is subdivided into two parts. The first one (ANAGAMMA) does the search for photopeaks and determines their centroids, areas and widths. The second part (FALEN) is used for energy determinations.

The principal program steps are:

a) Data Smoothing.
b) Background calculation.
c) Data analysis.
d) Energy calibration.

The smoothing of the raw data and the calculation of the background are carried out in a working area of 300 channels, but the analysis of the data is performed over only the central 100-channel zone. When the analysis of these 100 channels is completed, the working area moves upward 100 positions and the whole process starts again. In this way no boundary problems appear when the peaks fulfill the restriction on the FWHM mentioned above.

Drawings may be obtained at different stages. The authors wish to point out the usefulness of the peripheral plotting device not only to see howw well the computer fits welldefined single peaks, but also to look at multiplets and interpret the significance of the errors quoted on the output forms.

2. Description of the Program Steps.

2.1. Manual and Raw Data Inputs.

Figure 1 is a general flow chart of the code. The RAW DATA is the original spectrum punched on a binary tape, with data of up to 20 bits per channel. This is the only input which is not entered manually. The rest of the data are given through a teletype as answers to the following questions made by the computer: (1) DATA BLOCK: asks for the number of channels punched on the tape; (2) LOWER BOUNDARY CHANNEL: asks for the channel below which the spectrum is not analysed; (3) INITIAL AND FINAL CHANNEL: asks for the limits of the zone being analysed; (4) PLOTTING STEP and (5) MINIMUM AND COUNTS/INCH should be answered for the drawings; (6) BACKGROUND CRITERION (see section 2.4.), must be established for the background calculation; (7) NUMBER OF FWHM DATA: asks for the number of data pairs to be introduced when answering the following question; (8) CHANNEL-FWHM: asks for the pairs (channel-FWHM at that channel) to be entered for the calculation of FWHM.

The subroutines used to perform this step of the program are: @FC19,ENTRO, RBT and FC13. Appendix I gives an example of the ANAGAMMA input form. Appendix II contains a table of the options for the computer switches regarding questions (4), (5) and (6).

2.2. 20 to 16 Bits Conversion.

In order to save memory positions, the channel contents originally stored in up to 20 bits per channel are converted to 16 bits, preserving accuracy within one fourth of the statistical fluctuation. The following algorithm is used: a) if the channel content to be converted is equal or less than 32 767 (2^{15}-1), it is stored as a negative 15 bit number; b) if the channel content is greater than 32 767, it is replaced by the remainder of its division by 32 767, whose last two digits are replaced by 50 plus the integer part of the quotient of the division.

Reconversion is accomplished as follows: a') The sign of the stored number is changed; b') adding to the stored number the product of 32 767 times the difference between the last two digits of the stored number and 50. The error in the reconversion of numbers equal or less than 32 767 is zero; for larger numbers it never exceeds 50.

The following examples are intended for clarification of the procedure:

Content	Quotient	Remainder	Stored Number	Reconverted Number	Error
4 369	0	4 369	- 4 369	4 369	0
110 947	3	12 646	12 653	110 954	7
32 768	1	1	51	32 818	50

The subroutines used in this step of the program are: FC13, CONV, and COP.

2.3. Data Smoothing.

For smoothing the raw data we use a procedure strongly recommended in Ref. 6, which was found to be suitable for our purpose. Its first step consists in fitting to every five adjacent points a third-degree polynomial by least-squares methods. Since the procedure requires that there be at most one change of curvature in the fitting region, this imposes the condition that the FWHM must be not less than 4 channels. The smoothed spectrum thus obtained is subjected to the same procedure three times more.

This step is performed by subroutine ALIS.

2.4. Background Calculation.

The background is calculated by an approximation using a slowly varying function which connects all the minima in the smoothed spectrum. This is done by using only the parameter BACKGROUND CRITERION, as proposed in Ref.3. The optimum value for this parameter depends on the characteristics of the spectrum to be analized (the value we normally use is about 0.05) and noticeable changes in background are obtained only by varying it by orders of magnitude.

The subroutine FONDO is used in this step of the program.

2.5. Background Smoothing.

The obtained background is itself smoothed by applying subroutine ALIS to fit a third-degree polynomial to the data of every 21 adjacent channels. This procedure gives a background (BG)', lower than the "actual" one by a fraction of its square root which depends on the smoothing performed on the original spectrum. As in our case this fraction turns out to be[6] about $(1/17)^{1/2}$, the final expression for the background is

$$BG = (BG)' + 0.24 \ \{(BG)'\}^{1/2} \ .$$

Once the background is computed, it is subtracted from the smoothed spectrum, equating to zero those channel contents which are less than or equal to one fourth the statistical fluctuation of the background.

This step of the program is performed by subroutine ZEROS.

2.6. Data Analysis.

The background subtraction transforms the original spectrum into a number of "group of channels" with positive contents, delimited by regions containing zeros. In order to proceed with the analysis, a "zonal FWHM" is calculated for each region of interest by linear interpolation between the input FWHM data fed to the computer (section 2.1., questions 7 and 8). The initial channel of the first region to be analysed is required to be higher than the first channel of the input FWHM data. If a zone of interest has channels above the last one corresponding to the input FWHM data, the program will use the FWHM for the highest input for those channels. It is not required that input FWHM data be equispaced.

Any of the mentioned "groups of channels", whose base width is less than the FWHM, or whose maximum channel content is smaller than the statistical fluctuation of the background, is considered as not spectroscopically significant. Every remaining group is fitted by iterative methods, assuming from the outset that it is formed by the superposition of two gaussian peaks having the zonal FWHM (see below). The conditions imposed on these peaks are that their total area and combined first-order moments must equal the total area and the first-order moment corresponding to the data in the "group of channels" being considered, taken as a whole. These two condition are formulated as follows:

$$A_1 + A_2 = \frac{1}{R} \Sigma Y_i = \mathrm{I} \tag{1}$$

$$A_1 X_1 + A_2 X_2 = \frac{1}{R} \Sigma Y_i X_i = \mathrm{II} \tag{2}$$

where A_1 and A_2 are the heights of the gaussians and X_1 and X_2 the abscissae of their centroides; Y_i is the content of channel X_i; $R=2.505\sigma$; and 2.3548σ is the "zonal FWHM". Furthermore, the maximum content Y_M and the corresponding abscissa X_M of a given "group of channels" is obtained by determining the vertex of a parabola adjusted to the highest points of the group. Then, the ordinate for an arbitrary channel will be:

$$Y_i = A_1 e^{-\frac{1}{2\sigma^2}(X_1-X_i)^2} + A_2 e^{-\frac{1}{2\sigma^2}(X_2-X_i)^2}$$

Requiring that the condition $dY/dX=0$ is fulfilled at the point (Y_M, X_M), one obtains

$$e^{-\frac{1}{2\sigma^2}(X_1-X_M)^2} = \frac{(X_2 - X_M)}{(X_2 - X_1)} \frac{Y_M}{A_1}$$

which, combined with equations (1) and (2), gives:

$$e^{-\frac{1}{2\sigma^2}(X_M-X_1)^2} = \frac{(X_2-X_M)\ Y_M}{\mathrm{I}.X_2 - \mathrm{II}} \; ; \tag{3}$$

likewise:

$$e^{-\frac{1}{2\sigma^2}(X_M-X_2)^2} = \frac{(X_1-X_M)\ Y_M}{\mathrm{I}.X_1 - \mathrm{II}} \; . \tag{4}$$

The system of equations (1) to (4) is solved by an iterative process starting from the initial values $X_1 = \frac{\text{II}}{\text{I}} - \frac{\sigma}{2}$ and $X_2 = \frac{\text{II}}{\text{I}} + \frac{\sigma}{2}$. Care was taken to ensure the convergence of the iterative procedure. The subroutine which determines X_1, X_2, A_1 and A_2 is called DOBLE.

Next, the smaller gaussian is subtracted from the initial group of channels and a gaussian for the bigger peak is fitted by least-squares procedures to the resultant bulk. In this way the position, area and FWHM of the principal component of a possible multiplet are obtained. This component is subtracted from the original group and, if the remaining difference passes the significance test, the analysis is repeated for this new bulk. If not, the next group is considered. Hence, the significance test is the only limit for the number of components obtainable from the data contained in a "group of channels". At this stage a new condition is imposed: if during the fitting process the vertex of a gaussian peak falls inside the boundary of another one, both peaks are considered as being a single one, adding their areas and averaging their positions and FWHM.

Figure 2 exemplifies the procedure described above for the triplet shown in Fig. 5; the components are called A,B, and C. The subroutine that performs the least-squares fits is called PICO; the one which performs the subtractions is named RESTO, the control of the whole analysis process being made by the MMRI subroutine.

2.7. Error Determinations.

The error Δx in the peak position x is estimated in an unconventional way illustrated in Fig.3. The gaussian distribution fitted to the peak will be characterized by its "standard deviation" σ_G, its heigth H above the background BG on which it sits, and its central position m_o. The statistical errors for points B and C are then respectively $b=(BG + 0.325H)^{1/2}$ and

$c = (BG + 0.882H)^{1/2}$. Next, point m is obtained as shown on the figure. Finally, we take $\Delta x = 0.25\ \overline{mm}_o$, based on experience with well defined peaks. This procedure, which is not strictly correct, has been adopted since it yields reasonable errors despite its simplicity.

The standard error of the photopeak areas depends not only on the statistical fluctuation of the value itself but on the error of the background as well. As the latter in turn depends on the number of channels used to compute it, the error of the areas is expressed as:

$$\Delta_A = (A + \phi . 8\sigma_G . BG)^{1/2} \quad ,$$

where $8\sigma_G$ is twice the width of the peak base and $\phi \leq 1$ is an empirical factor that takes into account that usually more than $8\sigma_G$ points are available for computing BG. We have chosen $\phi = 0.75$.

One should not forget that the formulae for computing errors were obtained for single peaks. Hence, the errors quoted in the output forms are valid only for such peaks. Those quoted for the components of multiplets must be examined critically, taking into account the number of components and their relative intensities, appealing to the experience of the experimenter.

This whole step of the program is performed by the MMRI subroutine.

2.8. Output Selection.

According to the position of computer switch 9 (see Apendix II), the results obtained by the ANAGAMMA program have two alternative outputs: through a teletype, as shown by Table 1, or through a punched tape which may be used as an input for the FALEN program.

2.9. Manual and Raw Data Inputs.

The FALEN program requires as inputs the manual entry of the energies and channel positions of the calibration peaks as well as the corresponding standard errors for these values; besides, it has to be fed with the output of the ANAGAMMA program, either manually or through a punched tape. Appendix I gives an example of the FALEN input form.

2.10. Energy Calibration.

The energy calibration is performed with the FALEN code by adjusting a third-degree polynomial to a number of standard peaks distributed over the entire energy range. The adjustment is made by weighted least-squares methods. The energy for a given channel X is then $E=\Sigma_0^3\ C_i . X^i$ and its error is defined as

$$\Delta E = \{\sigma_c^2 + (\Delta X.C_1)^2\}^{1/2} ,$$

where σ_c stands for the error in the calibration curve. Due to memory-size limitations, σ_c is not computed using the correct error propagation method but is defined as $\sigma_c = \frac{1}{n} \Sigma_1^n\ \varepsilon_i$, where n is the number of calibration points and ε_i are the differences between energy input values and computed values. Although this value for σ_c is an overestimation, it usually is significantly smaller than the term $\Delta X.C_1$.

The energy calibration step requires the use of the FALEN, ALAN, and BINOM subroutines. Table 2 shows a partial output of the FALEN program.

2.11. Plots 1, 2, 3 and 4.

The circles on Fig.1 indicate the drawing options available at different stages of the analysis. PLOT 1 draws the initial spectrum; PLOT 2 draws the smoothed one; PLOT 3 draws the calculated background and PLOT 4 the adjusted gaussians. The switch positions to be used for selecting a given option are tabulated in Apendix II.

The subroutines used in these steps of the program are DIBU and EJES.

3. Examples of Analysis.

The partial spectra shown in Figs. 4 and 5 are typical. In particular, Fig. 4 corresponds to a smoothed spectrum and its calculated background.

It shows that drawings are invaluable aids to reject spurious peaks which are in fact the tails of intense peaks, based on their relative intensity and shape. The tail peaks should be eliminated from the output forms.

Fig. 6, finally, exemplifies a spectral zone of poor statistics. The original spectrum as well as the calculated background are shown in the upper part of the figure. For the sake of clarity the smoothed spectrum is not shown. In the lower part four single peaks can be seen.

4. Concluding Remarks.

Extended use of these programs at the Buenos Aires isotope separator on-line facility (IALE Project) with satisfactory results moves us to recommend their use where only small computers are available. One of the most remarkable features to be pointed out is the small number of external parameters involved, which implies simplicity of management. Once again, the use of an automatic plotter is emphatically recommended for a better evaluation of the output data.

The time required for the analysis of a 4096 channel spectrum with approximatelly 250 peaks is about 40 min including plotting time, using a computer cycle of 1.6 μsec.

Fifteen out of eighteen subroutines are written in Fortran II language, the rest in Assembler language. All of them are listed in Apendix III.

References.

1. M.A.J.Mariscotti, "A Method for Automatic Identification of Peaks in the Presence of Background and Its Application to Spectrum Analysis", Nuclear Instruments and Methods, Vol. 50 (1967) 309-320.

2. R.G.Helmer, R.L.Heath, M.Putnam, and D.H.Gipson, "Photopeak Analysis Program for Photon Energy and Intensity Determinations.Ge(Li) and NaI (Tl) Spectrometers", Nuclear Instruments and Methods, Vol.57 (1967) 46-57.

3. T.Inouye, T.Harper, and N.C.Rasmussen, "Application of Fourier Transforms to the Analysis of Spectral Data", Nuclear Instruments and Methods, Vol. 67 (1969) 125-132.

4. J.T.Routt and S.G.Prussin, "Photopeak Methods for the Computer Analysis of Gamma-Ray Spectra From Semiconductor Detector", Nuclear Instruments and Methods, Vol.72 (1969) 125-142.

5. E.Junod,"Elements de Programmes en Spectrometrie Gamma et Analyse par Activation", Centre d'Etudes Nucléaires de Grenoble, Rapport CEA-R-3955 (1970).

6. A.Savitzky and M.Golay, "Smoothing and Differentiation of Data by Simplified Least Squares Procedures", Analytical Chemistry, Vol. 36 (1964) 1627-1639.

TABLE 1

ANAGAMMA OUTPUT FORM

CHANNEL	ERROR	AREA	ERROR	FWHM
840.13	.46	.209255E+04	.201195E+03	6.562
847.97	.17	.665938E+04	.214075E+03	7.003
858.32	.01	.757646E+06	.891368E+03	7.108
884.91	.04	.383575E+05	.264476E+03	7.000

TABLE 2

FALEN OUTPUT FORM

PN	CHANNEL	ENERGY	ERROR	AREA	ERROR	FWHM
1	840.13	829.66	.59	2092.55	201.19	6.56
2	847.97	837.40	.22	6659.38	214.07	7.00
3	858.32	847.62	.01	757646.87	891.36	7.11
4	884.91	873.88	.05	38357.56	264.48	7.00

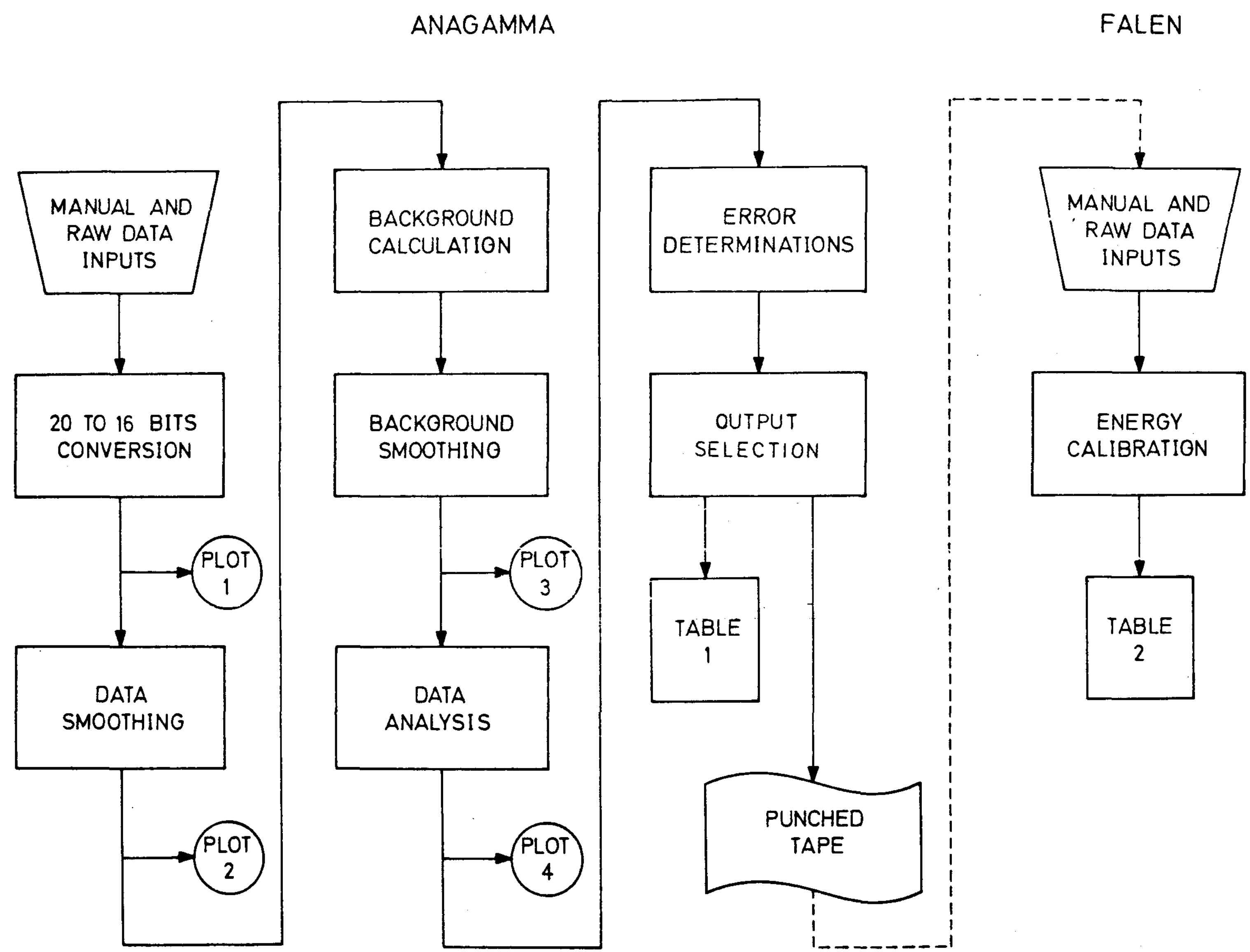

Fig. 1. Flow chart of the ANAGAMMA and FALEN codes.

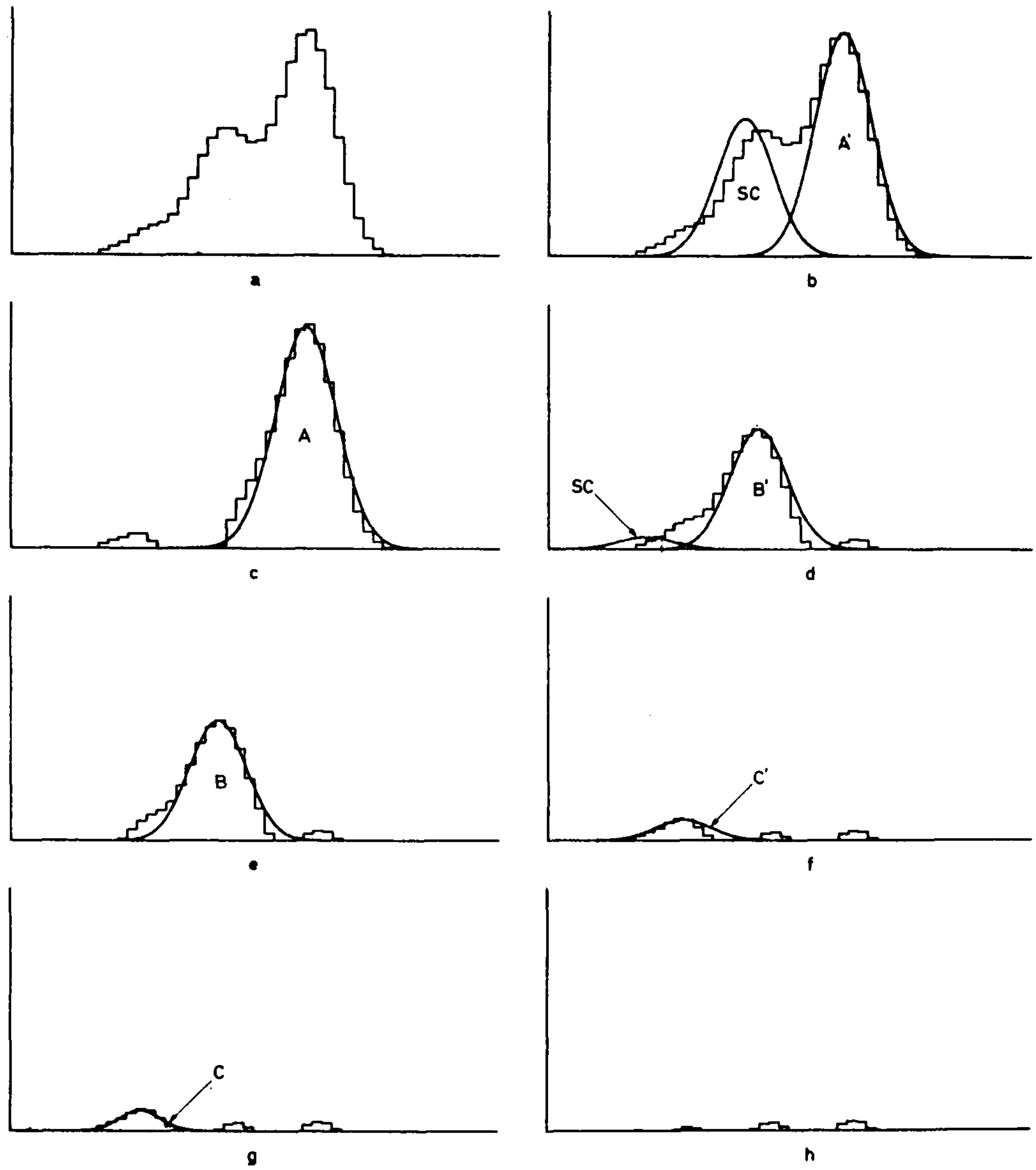

Fig. 2. Schematic analysis of a triplet (SC, smaller component): a) initial bulk, b) tentative fit with two gaussian peaks, c) least-squares fitting of the principal component, d) second tentative fit, e) least-squares fitting of the second component, f) third tentative fit, g) least squares fitting of the third component, h) no significant peaks remain.

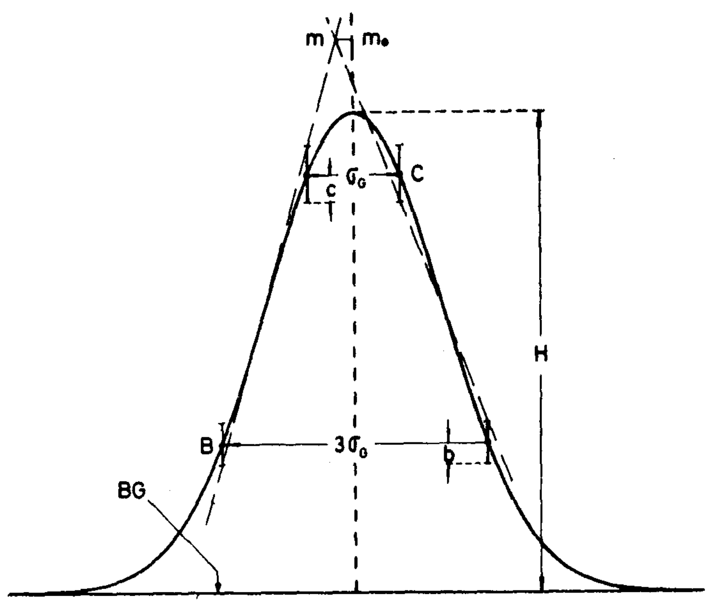

Fig. 3. Diagram showing the determination of position error: $\Delta x = 0.25\ \overline{mm_0}$.

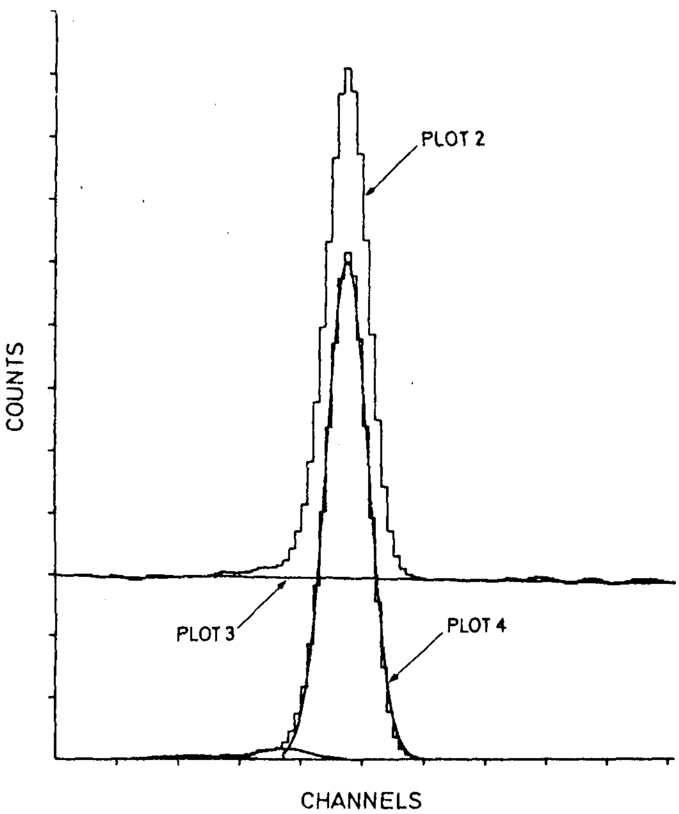

Fig. 4. Single peak fitting. The smaller component is a tail peak.

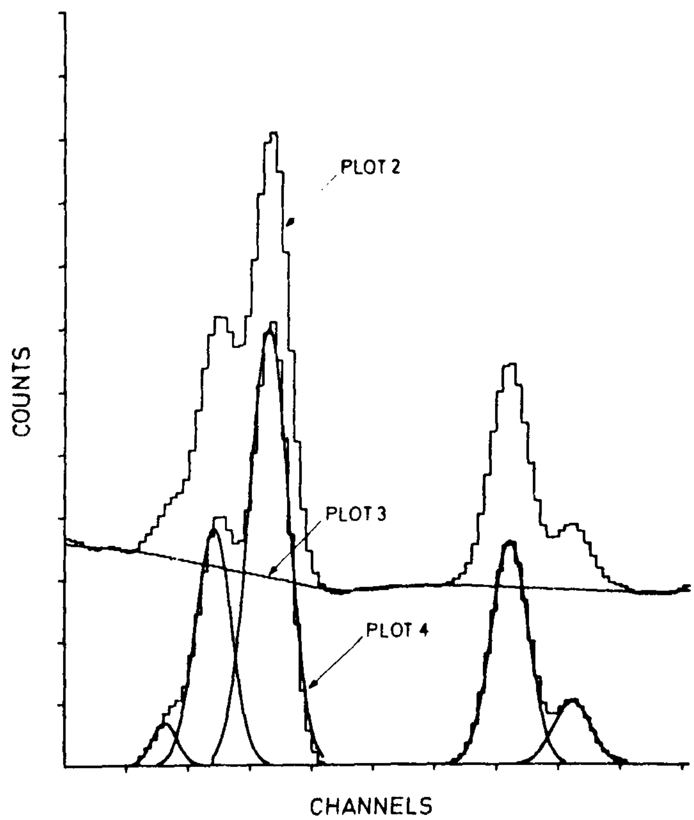

Fig. 5. Decomposition of multiplets.

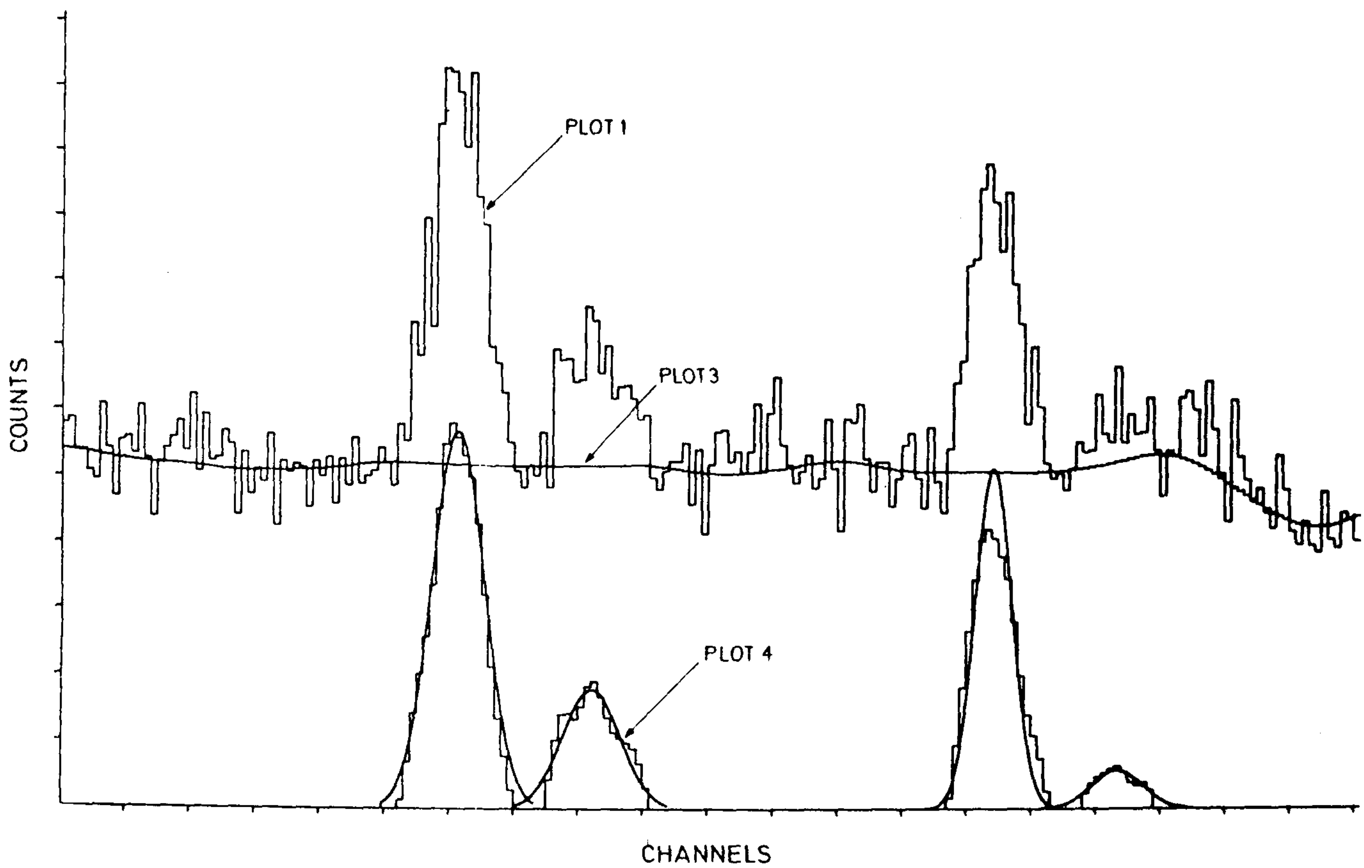

Fig. 6. Poor-statistics partial spectrum. The upper part of the drawing shows the original data and the computed background.

APPENDIX I: EXAMPLES OF INPUT FORMS.

Each question made by the computer is followed, within brackets, by an input example.

1.1. ANAGAMMA Code.

Case 1: Switches Ø and 1Ø in ON position (see Appendix II).

```
DATA BLOCK (4Ø96)

2Ø48

LOWER BOUNDARY CHANNEL (1ØØ)

8Ø

INITIAL AND FINAL CHANNEL (176Ø  28ØØ)

256Ø      286Ø

PLOTTING STEP  (.Ø2)

.Ø3

MINIMUM (1ØØØ.Ø) , COUNTS/INCH (5ØØ.Ø)

2ØØ.Ø      6ØØ.Ø

BACKGROUND CRITERION  (.Ø5)

.ØØØ5

NUMBER OF FWHM DATA (5)

3

CHANNEL        FWHM

2ØØØ           4.9

25ØØ           5.1

3ØØØ           5.3
```

Case 2: Switches Ø and 1Ø in OFF position (see Appendix II).

```
DATA BLOCK (4096)
1024
LOWER BOUNDARY CHANNEL (100)
75
INITIAL AND FINAL CHANNEL (1760    2800)
450      1750
NUMBER OF FWHM DATA (5)
5
CHANNEL        FWHM
0              4.0
500            4.1
1000           4.2
1500           4.3
2000           4.5
```

1.2 FALEN Code.

RUN NUMBER

145

NUMBER OF CALIBRATION DATA

6

ENERGY	ERROR	CHANNEL	ERROR	(511.01	.02	89.7	.32)
402.70	.30	255.70	.30				
511.01	.01	325.50	.10				
1118.60	.20	716.87	.20				
2517.70	.50	1617.12	.20				
5107.59	.40	3280.67	.20				
6129.60	.40	3937.26	.40				

(Here the computer will type the calibration energy values according to the third-degree polynomial adjusted to the input calibration data by weighted least-squares methods).

NORMALIZATION FACTOR

284167.5

(Here the program will provide the Falen Output Form as given in Table 2, where, in order to make the area of the selected peak equal to 1000, all the other ones have been divided by the stated "Normalization Factor").

APPENDIX II: OPTIONS FOR THE COMPUTER SWITCHES.

II.1 ANAGAMMA Code.

SWITCH	ON	OFF	FUNCTION
Ø	X		Asks for background criterion.
		X	Makes background criterion equal to .05 (normal).
9	X		Punched-tape output
		X	Teletype output.
1Ø	X		Drawing of fitted gaussians (Plot 4)
		X	No drawings at all
12	X		(Switch 1Ø ON) Draws smoothed spectrum and background (Plots 2 and 3).
		X	—
13	X		(Switch 1Ø ON) Draws original spectrum (Plot 1).
		X	—
14	X		Sends control to the first step.
		X	—
15	X		Stops the current plot.
		X	—

II.2. FALEN Code.

SWITCH	ON	OFF	FUNCTION
1Ø	X		Peak positions entered by teletype.
		X	Peak positions entered by punched tape.

APPENDIX III.

III.1. ANAGAMMA Code.

```
        NAM @FCI9
        ENT STA,@MSDB,@LSDB
        EXT RBT,FCI3,ENTRO
TBK     BSS 4196
TBL     BSS 600
TBM     BSS 600
LONG    OCT 0
STA     NOP
        JSB ENTRO
        DEF *+3
        DEF TBK
        DEF LONG
        LDA LONG
        STA LONG1
        LDA F0
        STA F1
        JSB RBT
        DEF *+3
        DEF LONG1
        DEF F1
        JSB FCI3
        DEF *+5
        DEF TBK
        DEF TBL
        DEF TBM
        DEF LONG
        JMP STA +1
@LSDB   OCT 2000
@MSDB   OCT 12000
F0      OCT 0
LONG1   OCT 0
F1      OCT 0
        END STA
```

```
      SUBROUTINE CONV (IP,TBK,I1,I2)
      DIMENSION IP(1),TBK(1)
      K=1
      DO 1 I=I1,I2
      IF(IP(I)) 2,2,3
    2 TBK(K)=-IP(I)
      GO TO 1
    3 L=IP(I)/100*100
      F=IP(I)-L-50
      F=32767.0*F
      TBK(K)=FLOAT(IP(I))+F
    1 K=K+1
      RETURN
      END
      END$
```

```
        NAM COP
        ENT COP
        EXT .ENTR,@MSDB
IY      BSS 1
Y       BSS 1
COP     NOP
        JSB .ENTR
        DEF IY
        LDB IY
        ADB =B176000
        SLB
        JMP LS1.3
        LDA =B170000
        BRS,SLB
        LDA MASK2
        JMP SIGA
LS1.3   LDA MASK1
        BRS,SLB
        LDA MASK3
SIGA    STA PASO
        BRS
        ADB @MSDB
        LDA 1,I
        AND PASO
        LDB PASO
        CPB MASK3
        JMP PACK3
        CPB MASK2
        JMP PACK2
        CPB MASK1
         JMP PACK1
        JMP PACK0
PACK2   ALF
PACK1   ALF
PACK0   ALF
PACK3   LDB EXP23
        STB M23
        LDB IY,I
        SSA
        JMP *+8
        CLE,ELB
        ELA
        ISZ M23
        JMP *-5
        CLA
        CLB
        JMP FIN
        CLE,ERA
        ERB
        STA PASO
        LDA M23
        CMA,INA
        ALS
```

```
           STA M23
           LDA 1
           IOR M23
           LDB 0
           LDA PASO
     FIN   STA Y,I
           ISZ Y
           STB Y,I
           JMP COP,I
     EXP23 DEC -32
     M23   OCT 0
     PASO  OCT 0
     MASK3 OCT 17
     MASK2 OCT 360
     MASK1 OCT 7400
           END
```

```
      SUBROUTINE ENTRO(IP,LONG)
      DIMENSION IP(5120)
      DO 1 I=1,5120
    1 IP(I)=0
      WRITE (2,100)
  100 FORMAT("DATA BLOCK  (4096)")
      READ (1,*) LONG
      RETURN
      END
      END$
```

```
      NAM RBT
      ENT RBT
      EXT .ENTR,.IOC.,@LSDB,@MSDB
PAR   BSS 1
FLAG  BSS 1
*
RBT   NOP
      JSB .ENTR
      DEF PAR
*
*
*
*
BGN1  NOP
      LDA =B10105
      STA UNIT1
      LDA =B40005
      STA UNIT2
*
      LDA PAR
      LDB 0,I
      STB LGTH
      INB
      STB BFRLN
      INA
      LDA 0,I
      STA SA2
      ADA @LSDB
      STA STAD
      ADA =D-1
      STA DBFR
      JSB STMK
      BLR
      SZB,RSS
      JMP *-3
*
      LIA 1
      CPA =B170000
      JMP OVER
      LDA STAD,I
      SZA,RSS
      JMP B16
*
OVER  LDA SA2
      LDB LGTH
B20   ARS,ARS
      BRS,BRS
      ADA @MSDB
      ADA =D-1
      INB
      STA DBFR
      STB BFRLN
      JSB STMK
```

```
          JMP RBT,I
B16       JMP RBT,I
*
STMK      NOP
          LDB DBFR,I
          STB CNTR
          LDB =B177777
          STB DBFR,I
          JSB IO
          LDA CNTR
          STA DBFR,I
          JMP STMK,I
*
IO        NOP
          JSB .IOC.
          OCT 1
          JSB .IOC.
UNIT1     OCT 20104
          JMP *-2
DBFR      BSS 1
BFRLN     BSS 1
          JSB .IOC.
UNIT2     OCT 40004
          SSA
          JMP *-3
          JMP IO,I
SA2       OCT 0
LGTH      BSS 1
STAD      BSS 1
CNTR      BSS 1
          END
```

```
      SUBROUTINE FCI3 (IP,TBK,TBL,LONG)
      DIMENSION IP(1),TBK(300),TBL(300)
      DIMENSION NAN(5),TAC(5)
      T=0
      WRITE(2,999)
999   FORMAT("LOWER BOUNDARY CHANNEL (100)")
      READ(1,*) KT
      I=0
      ALA=10000.0
  8   I=I+1
      IF(I-4096) 1,1,82
  1   N=0
      CALL COP (IP(I),T)
      T0=T
  3   A=T0-32767.0
      IF(A) 4,5,5
  5   N=N+1
      T0=A
      GO TO 3
  4   IF(N) 6,6,7
  6   A=-T
      IP(I)=A
      GO TO 8
  7   A=T0
      K=A
      IP(I)=K/100*100+N+50
      GO TO 8
 82   DO 84 I=1,KT
 84   IP(I)=IP(KT)
  2   WRITE (2,101)
101   FORMAT(//,"INITIAL AND FINAL CHANNEL","  (1760   2800)")
      READ(1,*) I,IFA
      FAC=.25
      IF(I) 19,19,20
 20   IF(ISSW(10)) 36,40
 36   WRITE (2,110)
110   FORMAT("PLOTTING STEP  (.02)")
      READ(1,*) P
      WRITE (2,104)
104   FORMAT("IS THE PLOTTER READY?")
 30   WRITE (2,114)
114   FORMAT("MINIMUM  (1000.0), COUNTS/INCH (500.0)")
      READ (1,*) FMIN,ESC
 40   IF(ISSW(0)) 70,71
 71   DCR=.05
      GO TO 73
 70   WRITE (2,111)
111   FORMAT("BACKGROUND CRITERION (.05)")
      READ (1,*) DCR
 73   WRITE (2,112)
```

```
112 FORMAT("NUMBER OF FWHM DATA (5)")
    READ(1,*) KHT
    WRITE (2,708)
708 FORMAT("CHANNEL    FWHM")
    DO 709 KTH=1,KHT
709 READ(1,*) NAN(KTH),TAC(KTH)
    A2=TAC(1)
 72 IW=5
 74 ER=.2*A2
    JW=A2
    IF(ISSW(14)) 2,802
802 IF(ISSW(10)) 800,801
800 CALL WHERE(Y0,X0)
    CALL PLOT (Y0,X0,-1)
801 P=-P
    XS=0.0
    XP=0.0
    IF(ISSW(9)) 635,637
637 IF(ISSW(14)) 2,820
820 WRITE (2,634)
634 FORMAT(///,13X,"CHANNEL",2X,"ERROR",5X,"AREA",9X,"ERROR",
   16X,"FWHM",/)
635 IF(I-100) 11,11,12
 11 I1=2
    KC=I
    GO TO 13
 12 I1=I-100
    KC=101
 13 IF(IFA-100-I) 14,14,15
 14 L=0
    IF(IFA+100-LONG) 17,17,18
 17 I2=IFA+100
    GO TO 16
 18 I2=LONG
    GO TO 16
 15 L=1
    IF(I+200-LONG) 23,18,18
 23 I2=I+200
 16 DO 803 I23=1,600
803 TBK(I23)=0.0
    CALL CONV (IP,TBK,I1,I2)
    KF=I2-I1-99
    KF2=I2-I1
    XP=XS
    IF(ISSW(13)) 50,51
 50 CALL DIBU(TBK,XP,KC,KF,P,FMIN,ESC)
    IF(ISSW(14)) 2,51
 51 XP=XS
    CALL ALIS (IW,TBK,1,KF2,4)
    IF(ISSW(12)) 804,805
804 CALL DIBU(TBK,XP,KC,KF,P,FMIN,ESC)
805 XP=XS
    IF(ISSW(14)) 2,821
821 CALL FONDO (TBK,TBL,1,KF2,DCH)
    CALL ALIS (21,TBL,1,KF2,1)
    DO 63 L1=2,299
    IF(TBL(L1)) 60,63,63
```

```
 60 TBL(L1)=0.0
 63 TBL(L1)=TBL(L1)+.24*SQRT(TBL(L1))
    IF(ISSW(12)) 806,807
806 CALL DIBU(TBL,XP,KC,KF,P,FMIN,ESC)
807 XP=XS
    IF(ISSW(14)) 2,822
822 DO 808 L1=50,250
 86 TBK(L1)=TBK(L1)-TBL(L1)-.25*SQRT(TBL(L1))
    IF(TBK(L1) ) 809,809,808
809 TBK(L1)=0.0
808 CONTINUE
    CALL ZEROS(TBK,KF2,JW,TBL,FAC)
    IF(ISSW(10)) 810,811
810 CALL DIBU(TBK,XP,KC,KF,P,0.0,ESC)
811 XM=XP
    IF(ISSW(14)) 2,823
823 XP=XS
    II1=I1-2
    DO 740 KTH=1,KHT
    IF(II1-NAN(KTH)) 720,740,740
740 CONTINUE
    A2=TAC(KHT)
    GO TO 730
720 A2=(TAC(KTH)-TAC(KTH-1))/FLOAT(NAN(KTH)-NAN(KTH-1))*FLOAT(
   1II1-NAN(KTH-1))+TAC(KTH-1)
730 YP=0.0
    CALL MMRI(TBK,TBL,II1,KC,KF2,P,YP,ESC,XP,A2,ER)
    XS=XM
    IF(ISSW(9)) 75,76
 76 WRITE(2,650) A2
650 FORMAT(/,53X,F7.3)
 75 IF(L) 21,21,22
 21 IF(ISSW(10)) 33,32
 33 CALL PLOT(0.0,XM,3)
    CALL EJES (P)
 32 IF(ISSW(9)) 173,174
173 WRITE(4) ALA,ALA,ALA,ALA,ALA
174 GO TO 2
 22 I=I2-99
    GO TO 12
 19 RETURN
    END
    END$
```

```
      SUBROUTINE DIBU (Y,X,II,IF,P,T,A)
      DIMENSION Y(1)
      IF(ISSW(10)) 25,20
   25 II=II-1
    7 YN=(Y(II)-T)/A
      B=12.0
      IF(YN-B) 100,100,11
   11 YN=B
  100 CALL PLOT (YN,X,3)
      II=II+1
      DO 5 I=II,IF
      IF(ISSW(15)) 20,2
    2 YN=(Y(I)-T)/A
  102 IF(YN-B) 4,4,3
    3 YN=B
    4 CALL PLOT (YN,X,2)
      X=X+P
    5 CALL PLOT (YN,X,2)
      CALL PLOT(YN,X,3)
   20 RETURN
      END
      END$
```

```
      SUBROUTINE EJES (P)
      CALL WHERE (YN,X)
      CALL PLOT (0.0,X,3)
      CALL PLOT (0.0,0.0,2)
      A=0.0
    1 A=A+P*10.0
      IF(A-X) 3,3,2
    2 CALL PLOT (0.0,A,3)
      CALL PLOT (-.1,A,2)
      GO TO 1
    3 CALL PLOT (0.0,0.0,3)
    5 DO 7 I=1,12
      Y=I
      CALL PLOT (Y,0.0,2)
      CALL PLOT (Y,.1,3)
    7 CALL PLOT (Y,0.0,2)
      CALL PLOT(Y,0.0,3)
      RETURN
      END
      END$
```

```
      SUBROUTINE FONDO (TBK,TBL,IMAX,NCH,DCR)
      DIMENSION TBK(1),TBL(1)
      JJ=NCH-1
      LA =IMAX+1
      I=IMAX+2
      IMIN=0
      IX=0
      ILOOP=0
  333 DTSI=TBK(I+1)-TBK(I)
      DTS1=TBK(I)-TBK(I-1)
      XM=DTSI*DTS1
      DB=DTSI-DTS1
      IF(XM) 305,305,303
  305 IF(DB) 303,303,304
  304 IF(I+15-JJ) 310,310,510
  310 GRAD=(TBK(I)-TBK(LA))/FLOAT(I-LA)
      IX=IX+1
      TBL(IX)=GRAD
      TBL(IX+2)=FLOAT(I)
      IF(IX-2) 303,564,564
  564 CONTINUE
      GRAD1=TBL(1)
      GRAD2=TBL(2)
      M1=TBL(3)
      M2=TBL(4)
      TB1=TBK(LA)+GRAD1*FLOAT(M2-LA)
      TB2=TBK(M2)
      TB12=(TB1+TB2)/2.0
      IF(TB12) 663,663,664
  663 CRT=DCR*FLOAT(M2-M1)
      GO TO 665
  664 CRT=DCR*FLOAT(M2-M1)*SQRT(TB12)
  665 IF(TB1-TB2) 565,565,666
  666 IF(TB1-TB2-CRT) 565,565,566
  566 ILOOP=ILOOP+1
      IF(ILOOP-5) 567,565,565
  567 I=M1
       IX=0
      GO TO 303
  565 ILOOP=0
      IX=0
      I=M1
  308 LL=I
      LOOP=0
  507 LOOP=LOOP+1
      IF(LOOP-16) 525,510,510
  525 TB=(TBK(LL-1)+TBK(LL)+TBK(LL+1))/3.0
      IF(TB-TBK(LL-2)) 603,603,602
  602 IF(TB-TBK(LL-1)) 603,603,502
  603 IF(TB-TBK(LL+2)) 503,503,604
  604 IF(TB-TBK(LL+1)) 503,503,504
```

```
502 LL=LL-1
    IF(LL-I) 523,503,523
523 IF(LL-LA) 303,303,507
504 LL=LL+1
    IF(LL-I) 524,503,524
524 IF(LL-I-6) 507,510,510
510 LL=I
    GO TO 533
503 TBK(LL)=(TBK(LL-1)+TBK(LL)+TBK(LL+1))/3.0
533 LB=LA+1
    LX=LL-1
    LBASE=LL-LA
    TBL(LA)=TBK(LA)
    IF(LBASE-8) 550,551,551
551 IMIN=IMIN+1
550 DO 306 IK=LB,LX
306 TBL(IK)=(TBK(LA)*FLOAT(LL-IK)+TBK(LL)*FLOAT(IK-LA))/FLOAT(LL-LA)
    LA=LL
    IF(I-LL) 520,303,303
520 I=LL
303 IF(I-JJ) 334,335,335
334 I=I+1
    GO TO 333
335 TBL(LA)=TBK(LA)
    LB=LA+1
    LX=JJ
    LL=NCH
    DO 307 IK=LB,LX
307 TBL(IK)=(TBK(LA)*FLOAT(LL-IK)+TBK(LL)*FLOAT(IK-LA))/
   1FLOAT(LL-LA)
    TBL(NCH)=TBK(NCH)
    TBL(IMAX)=TBK(IMAX)
    RETURN
    END
    END$
```

```
      SUBROUTINE ALIS (IW,Y,II,IF,IH)
      DIMENSION Y(1),B(22),V(21)
      X2=0.0
      X4=0.0
      KW=IW/2
      DO 41 I=1,KW
      T1=I
      X2=X2+T1**2
   41 X4=X4+T1**4
      X2=2.0*X2
      X4=2.0*X4
      Z=FLOAT(IW)*X4-X2**2
      DO 42 I=1,KW
   42 V(I)=X4-X2*FLOAT(I)**2
      L=KW+2
      DO 44 I=L,IW
      K=I-L+1
   44 V(I)=V(K)
      V(KW+1)=X4
      KKW=KW/2
      DO 43 I=1,KKW
      T1=V(I)
      L=KW-I+1
      V(I)=V(L)
   43 V(L)=T1
      DO 50 IJT=1,IH
      N=0
      IT=KW+1
      I2=IF-IW+1
      I=II-1
    7 I=I+1
      IF(I2-I) 4,5,5
    5 N=N+1
      B(N)=0.0
      DO 6 M=1,IW
      J=I+M-1
    6 B(N)=B(N)+V(M)*Y(J)
      B(N)=B(N)/Z
      IF(N-KW) 7,7,8
    8 Y(I)=B(1)
      N=N-1
      DO 9 M=2,IT
    9 B(M-1)=B(M)
      GO TO 7
    4 DO 11 M=1,KW
      Y(I)=B(M)
   11 I=I+1
   50 CONTINUE
      RETURN
      END
      END$
```

```
      SUBROUTINE ZEROS (Y,IJ,IW,X,FAC)
      DIMENSION Y(1),X(1)
      N=0
      K=1
      J=IJ-1
    1 K=K+1
      IF(K-J) 2,2,3
    3 RETURN
    2 IF(Y(K)-FAC*SQRT(X(K))) 4,4,5
    4 Y(K)=0.0
      IF(N) 1,1,7
    5 IF(N) 6,6,8
    6 L=K
    8 N=N+1
      GO TO 1
    7 IF(N-IW) 9,9,10
    9 DO 33 M=L,K
   33 Y(M)=0.0
   10 N=0
      GO TO 1
      END
      END$
```

```
      SUBROUTINE RESTO (TBK,K1,K2,X,A,SIG)
      DIMENSION TBK(1)
      D=1.0/(2.0*SIG**2)
      DO 1 I=K1,K2
      T=I
      TBK(I)=TBK(I)-A*EXP(-(X-T)**2*D)
      IF(TBK(I)) 2,1,1
    2 TBK(I)=0.0
    1 CONTINUE
      RETURN
      END
      END$
```

```
      SUBROUTINE MMRI(TBK,TBL,I,KC,KF,P,T,AT,XE1,AA,ER)
      DIMENSION TBK(1),TBL(1)
      IF(ISSW(10)) 75,76
   75 CALL PLOT(0.0,XE1,3)
   76 FI=I
      ITL=0
      JW=AA
      K=KC+1
    5 K=K-1
      IF(K+I-5) 9,9,8
    8 IF(TBK(K)) 9,9,5
    9 K=K-2
      FKC=KC
      XE0=XE1
   10 N=-1
      NT=0
   11 K=K+1
      IF(K-KF) 12,1,1
  111 RETURN
   12 IF(N) 13,14,15
   13 IF(TBK(K)) 16,16,11
   16 IF(K-(KF-80)) 60,1,1
   60 N=0
      GO TO 11
   14 IF(TBK(K)) 11,11,17
   17 K1=K-1
      FK1=K1
      N=1
      GO TO 11
   15 IF(TBK(K)) 18,18,11
   18 K2=K
   91 IF(K2-K1-1-JW) 20,20,42
   42 FK2=K2
      SIG=AA/2.3548
      D=1.0/(2.0*SIG**2)
      FN2=1.5*SIG
   93 CALL DOBLE (TBK,TBL,K1,K2,SIG,D,X1,X2,A1,A2,NP)
   97 B1=2.505*SIG
      IF(NP-1) 20,21,22
   20 IF(K2-(KF-80)) 23,1,1
   23 K=K2-1
      GO TO 10
   21 K11=X1-FN2
      K12=X1+FN2
      CALL PICO (TBK,K11,K12,SIG0,X0,A0,ITT)
      IF(ITT) 80,80,81
   81 A0=A1/(2.505*SIG)
      SIG0=SIG
      X0=X1
      GO TO 80
   22 DO 25 J=K1,K2
```

```
 25 TBK(J-100)=TBK(J)
101 KR1=K1-100
    KR2=K2-100
    IF(A1-A2) 26,26,27
 26 AR=A1/(2.505*SIG)
    XR=X1-100.0
    GO TO 28
 27 AR=A2/(2.505*SIG)
    XR=X2-100.0
 28 CALL RESTO (TBK,KR1,KR2,XR,AR,SIG)
    W=0.0
    DO 70 J=KR1,KR2
    IF(TBK(J)-W) 70,71,71
 71 W=TBK(J)
    FT1=J
 70 CONTINUE
    K11=FT1-FN2
    K12=FT1+FN2
    CALL PICO (TBK,K11,K12,SIG0,X0,A0,ITT)
    IF(ITT) 29,29,21
 29 X0=X0+100.0
 80 CALL RESTO (TBK,K1,K2,X0,A0,SIG0)
    K=K1-1
    A1=A0*2.505*SIG0
    X1=X0
    SIG=SIG0
500 X1=X1+FI
    A120=SIG*2.3548
 38 ITL=ITL+1
    TBL(ITL)=X1
    TBL(ITL+50)=A1
    TBL(ITL+250)=A120
    GO TO 10
 84 WRITE (2,200) X1,ERX,A1,ERA,A120
    GO TO 174
173 WRITE (4) X1,ERX,A1,ERA,A120
174 CONTINUE
200 FORMAT(10X,F10.2,F7.2,2E13.6,F7.3)
    IF(ISSW(10)) 77,59
 77 X0=X1-FI
    FK1=X0-3.0*AA/2.3548
    FK2=X0+3.0*AA/2.3548
    D=1.0/(2.0*SIG**2)
    A0=A1/(2.505*SIG)
    T=FK1
    IF(ISSW(14)) 111,50
 50 B=12.0
 52 XE1=XE0+(FK1-FKC)*P+P/2.0
    CALL PLOT (0.0,XE1,3)
 58 CONTINUE
 53 H=A0*EXP(-(X0-T)**2*D)/AT
 55 IF(H-B) 56,56,57
 57 H=B
 56 CALL PLOT (H,XE1,2)
    XE1=XE1-.01
    T=T-.01/P
```

```
      CALL PLOT (H,XE1,2)
      IF(T-FK2) 58,591,591
  591 CALL PLOT(H,XE1,3)
   59 IF(M-ITL) 85,83,111
    1 M=1
      IF(ITL-1) 111,83,61
   61 N=M
  620 W=TBL(M)
   62 N=N+1
      IF(N-ITL) 63,63,66
   63 IF(W-TBL(N)) 62,62,64
   64 DO 65 I3=1,3
      IF(I3-3) 32,33,33
   33 I33=150
      GO TO 34
   32 I33=0
   34 I4=(I3-1)*50+I33
      I5=I4+N
      I4=I4+M
      TRN=TBL(I4)
      TBL(I4)=TBL(I5)
   65 TBL(I5)=TRN
      GO TO 620
   66 M=M+1
      IF(M-ITL) 61,67,67
   67 M=1
   85 IF(TBL(M+50)-TBL(M+51)) 68,68,69
   68 D8=1.0/(2.0*(TBL(M+251)/2.3548)**2)
      IF((TBL(M)-TBL(M+1)-SQRT(ALOG(TBL(M+51)/TBL(M+50))/D8))*
     1(TBL(M)-TBL(M+1)+SQRT(ALOG(TBL(M+51)/TBL(M+50))/D8))) 82,82,83
   69 D8=1.0/(2.0*(TBL(M+250)/2.3548)**2)
      IF((TBL(M+1)-TBL(M)-SQRT(ALOG(TBL(M+50)/TBL(M+51))/D8))*
     1(TBL(M+1)-TBL(M)+SQRT(ALOG(TBL(M+50)/TBL(M+51))/D8))) 82,82,83
   82 W=TBL(M+50)
      W1=TBL(M+51)
      TBL(M+51)=W+W1
      TBL(M+1)=(TBL(M)*W+TBL(M+1)*W1)/TBL(M+51)
      TBL(M+251)=(TBL(M+250)*W+TBL(M+251)*W1)/TBL(M+51)
      M=M+1
      GO TO 85
   83 X1=TBL(M)
      A1=TBL(M+50)
      A120=TBL(M+250)
      M=M+1
  840 SIG=A120/2.3548
      IF(X1-FI-FKC) 85,150,150
  150 IF(X1-FI-(FKC+100.5)) 841,111,111
  841 IKI=X1-FI
      TF=TBL(IKI)*4.0
      SIGT=A120/2.3548
      ART=A1/(2.505*SIGT)
      ERX=(3.0*SQRT(TF+3.0*ART)+SQRT(TF+ART))*SIGT/(8.0*ART)
      ERA=SQRT(A1+1.5*SIGT*TF)
      IF(ISSW(9)) 173,84
      END
      END$
```

```
      SUBROUTINE DOBLE (TBK,TBL,K1,K2,SIG,D,X1,X2,A1,A2,NP)
      DIMENSION TBK(1),TBL(1)
      A=0.0
      B=0.0
      FK1=K1
      FK2=K2
      T2=TBK(K1)
      DO 2 L=K1,K2
      T1=L
      IF(T2-TBK(L)) 8,8,7
    8 T2=TBK(L)
      L1=L
    7 A=A+TBK(L)
    2 B=B+TBK(L)*T1
      B1=1.0/(2.505*SIG)
      A=A*B1
      B=B*B1
      T1=TBK(L1)-.5*(TBK(L1-1)+TBK(L1+1))
      X=FLOAT(L1)-.25*(TBK(L1-1)-TBK(L1+1))/T1
      Y=TBK(L1)+T1*(FLOAT(L1)-X)**2
      IF(Y-SQRT(TBL(L1))) 150,150,3
    3 S=B/A
      X1=S-.5*SIG
      X2=S+.5*SIG
      IL=0
  120 IL=IL+1
      IF(IL-30) 19,19,26
   19 T1=EXP(-(X-X1)**2*D)
      T2=EXP(-(X-X2)**2*D)
      T3=(X-X1)*Y/(B-X1*A)
      T4=(X-X2)*Y/(B-X2*A)
      T6=(X-X2)*T2/SIG**2
      T8=-Y*(X*A-B)/(B-X2*A)**2
      T5=-Y*(X*A-B)/(B-X1*A)**2
      T7=(X-X1)*T1/SIG**2
      T10=T3-T2
      T11=T4-T1
      T12=T6*T7-T8*T5
      IF(ABS(T12)-1.E-19) 26,26,20
   20 T14=(T10*T7-T11*T5)/T12
      T13=(T6*T11-T8*T10)/T12
   23 X1=X1+T13
      X2=X2+T14
      IF(X2-S) 29,29,72
   72 IF(S-X1) 29,29,71
   71 IF((FK2-X1)*(FK1-X1)) 25,35,35
   26 X1=X
      A1=A/B1
      NP=1
      GO TO 100
   25 IF((FK2-X2)*(FK1-X2)) 27,37,37
```

```
 27 IF(ABS(T13)-.01) 28,28,29
 28 IF(ABS(T14)-.01) 30,30,29
 30 IF(ABS(X2-X1)-.25*SIG) 31,31,32
 31 X1=(X2+X1)/2.0
    A1=A/B1
    NP=1
    GO TO 100
 32 A1=(X2*A-B)/(X2-X1)
    IKK=X1
    IF(A1-.5*SQRT(TBL(IKK))) 26,26,861
861 IF(A-A1-.5*SQRT(TBL(IKK))) 26,26,862
862 IF(ABS(A1-A/2.0)-.01*A1) 33,33,34
 33 IF(ABS(X2-X1)-.5*SIG) 31,31,34
 34 A2=A-A1
    IF(A1-A2) 863,863,864
863 IF((X1-X2-SQRT(ALOG(A2/A1)/D))*(X1-X2+SQRT(ALOG(A2/A1)/
   1D)))26,26,865
864 IF((X2-X1-SQRT(ALOG(A1/A2)/D))*(X2-X1+SQRT(ALOG(A1/A2)/
   1D)))26,26,865
865 NP=2
    A1=A1/B1
    A2=A2/B1
    IF(A1*A2) 26,26,100
 29 IF(S-X1-.05*SIG) 35,35,36
 35 X1=S-.05*SIG
    IF((FK2-X2)*(FK1-X2)) 36,37,37
 36 IF(X2-S-.05*SIG) 37,37,120
 37 X2=S+.05*SIG
    GO TO 120
150 NP=0
100 RETURN
    END
    END$
```

```
      SUBROUTINE PICO (ZA,K1,K2,SIG,X,A,ITT)
       DIMENSION ZA(50)
       C1=0.0
       C2=0.0
       C3=0.0
       C4=0.0
       C5=0.0
       C6=0.0
       C7=0.0
       C8=0.0
       DO 150 I=K1,K2
       FI=I-(K1-1)
       IF(ZA(I)-1.0) 150,150,7
     7 C1=C1+FI**2
       C2=C2+FI
       C3=C3+1.0
       AZA=ALOG(ZA(I))
       C4=C4+AZA
       C5=C5+FI**3
       C6=C6+FI*AZA
       C7=C7+FI**4
       C8=C8+FI**2*AZA
   150 CONTINUE
       D=C1**3+C2**2*C7+C3*C5**2-C1*C3*C7-C1*C2*C5*2.0
       IF(ABS(D)-1.0E-15) 151,151,152
   151 ITT=1
       GO TO 153
   152 X1=C1**2*C4+C2**2*C8+C3*C5*C6-C1*C3*C8-C1*C2*C6-C2*C4*C5
       X1=X1/D
       IF(X1) 8,151,151
     8 CONTINUE
       X2=C1**2*C6+C2*C4*C7+C3*C5*C8-C3*C6*C7-C1*C4*C5-C1*C2*C8
       X2=X2/D
       X3=C1**2*C8+C2*C6*C7+C4*C5**2-C1*C4*C7-C1*C5*C6-C2*C5*C8
       X3=X3/D
       SIG=SQRT(-1.0/(2.0*X1))
       X=-X2/(2.0*X1)+FLOAT(K1-1)
       A=EXP(X3-X2**2/(4.0*X1))
       ITT=0
   153 RETURN
       END
       END$
```

III.2. FALEN Code.

```
      NAM 0FC18
      ENT START
      EXT FALEN
START NOP
      JSB FALEN
      DEF *+1
      JMP START+1
      END START
```

```
      SUBROUTINE BINOM (A1,C,N)
      J=A1/100.0
      B=J*100
      A=A1-B
      B=J
      CALL ALAN (A,Z,N)
      CALL ALAN (B,Y,N)
      T=Z+Y*100.0**N
      T0=1
      FAC=1
      M=N-1
      DO 1 J=1,M
      T0=T0*FLOAT(J)
      J0=J-1
      FAC=FAC*FLOAT(N-J0)
      I=N-J
      CALL ALAN (A,Z,I)
      CALL ALAN (B,Y,J)
    1 T=T+FAC*Z*Y*100.0**J/T0
      C=T
      RETURN
      END
      END$
```

```
      SUBROUTINE FALEN
      DIMENSION E(50),X(50),Y(50),ENE(50),PESO(50)
    5 WRITE (2,110)
  110 FORMAT("PUT THE ROW DATA IN THE PHOTOREADER",///)
      WRITE (2,130)
  130 FORMAT("RUN NUMBER")
      SPESO=0.0
      READ (1,*) NRUN
      WRITE (2,101)
  101 FORMAT("NUMBER OF ENERGY CALIBRATION DATA  (20)")
      READ (1,*) N
      WRITE (2,102)
  102 FORMAT("ENERGY   ERROR   CHANNEL  ERROR (511.01  .02  89.7  .32)")
      DO 6 I=1,N
      READ (1,*) E(I),ER,X(I),XR
      PESO(I)=1.0/(ER**2+XR**2)
    6 SPESO=SPESO+PESO(I)
      PEN=(E(N)-E(1))/(X(N)-X(1))
      DO 1 I=1,N
      K=X(I)
      F=X(I)-FLOAT(K)
      ENE(I)=E(I)
      Y(I)=X(I)
      E(I)=E(I)-F*PEN
    1 X(I)=K
      B0=1.0
      B1=0.0
      B2=0.0
      B3=0.0
      B4=0.0
      B5=0.0
      B6=0.0
      E0=0.0
      E1=0.0
      E2=0.0
      E3=0.0
      DO 2 I=1,N
      RT=PESO(I)/SPESO
      Z=E(I)
      H=X(I)
      B1=B1+H*RT
      E1=E1+Z*H*RT
      CALL BINOM (H,G,2)
      E2=E2+E(I)*G*RT
      B2=B2+G*RT
      CALL BINOM (H,G,3)
      E3=E3+Z*G*RT
```

```
      B3=B3+G*RT
      CALL BINOM (H,G,4)
      B4=B4+G*RT
      CALL BINOM (H,G,5)
      B5=B5+G*RT
      CALL BINOM (H,G,6)
      B6=B6+G*RT
    2 E0=E0+Z*RT
      J=.07*(ALOG(B0)+ALOG(B1)+ALOG(B2)+ALOG(B3)+ALOG(B4)+ALOG(B5)+
     1ALOG(B6))
      FJ=10.0**J
      B0=B0/FJ
      B1=B1/FJ
      B2=B2/FJ
      B3=B3/FJ
      B4=B4/FJ
      B5=B5/FJ
      B6=B6/FJ
      E0=E0/FJ
      E1=E1/FJ
      E2=E2/FJ
      E3=E3/FJ
      D44=B4*B4-B3*B5
      D33=B3*B3-B2*B4
      D55=B5*B5-B4*B6
      D34=B3*B4-B2*B5
      D24=B2*B4-B1*B5
      DE42=B4*E2-B5*E1
      H1=D34*D33-D44*(B2*B3-B1*B4)
      H2=(B4*B5-B3*B6)*D44-D55*D34
      H3=D24*D33-D44*(B1*B3-B0*B4)
      H4=(B3*B5-B2*B6)*D44-D55*D24
      H5=(B5*E3-B6*E2)*D44-DE42*D55
      H6=DE42*D33-(B3*E1-B4*E0)*D44
      C0=(H1*H5-H2*H6)/(H1*H4-H2*H3)
      C1=(H4*H6-H3*H5)/(H1*H4-H2*H3)
      Q=E0-B0*C0-B1*C1
      P=E1-B1*C0-B2*C1
      C2=(Q*B4-P*B3)/(B2*B4-B3*B3)
      C3=(P*B2-Q*B3)/(B2*B4-B3*B3)
      ET=0.0
      WRITE(2,200)
      WRITE (2,510) C0,C1,C2,C3
  510 FORMAT(" A=",E14.8," B=",E14.8," C=",E14.8," D=",E14.8,/)
      DO 3 I=1,N
      H=X(I)
      CALL BINOM (H,H2,2)
      CALL BINOM (H,H3,3)
      ENER=C0+C1*H+C2*H2+C3*H3+(Y(I)-X(I))*(C1+2.0*H*C2+3.0*H2*C3)
      ERR=ENER-ENE(I)
      ET=ET+ABS(ERR)*PESO(I)
    3 WRITE (2,100) Y(I),ENER,ERR
      ET=ET/SPESO
      WRITE (2,100) ET
```

```
100 FORMAT(3F11.3)
    ET=ET**2
    WRITE(2,800)
800 FORMAT("NORMALIZATION FACTOR")
    READ (1,*) FAC
    IF(FAC) 5,5,80
 80 NP=0
 10 NP=NP+1
    WRITE (2,200)
200 FORMAT(////)
    NL=(NP-1)*60
    WRITE (2,300)
300 FORMAT("-----------------------------------------------------
   1----------------------")
    WRITE (2,200)
    WRITE (2,400) NRUN,NP
400 FORMAT(5X,"RUN",I5,45X,"PAGE",I2,/)
    WRITE (2,500)
500 FORMAT(5X," PN",3X,"CHANNEL",8X,"ENERGY",15X,"AREA",11X,"
   1FWHM",/)
 30 IF(ISSW(10)) 35,36
 35 READ(1,*) C
    A=0.0
    NL=NL+1
    A12=0.0
    GO TO 56
 36 READ(5) C,ERC,A,ERA,A12
 56 IF(C-10000.0) 20,99,99
 20 K=C
    Y1=C-FLOAT(K)
    H=K
    NL=NL+1
    CALL BINOM (H,H2,2)
    CALL BINOM (H,H3,3)
    E1=C0+C1*H+C2*H2+C3*H3+Y1*(C1+2.0*H*C2+3.0*H2*C3)
    ERC=SQRT(ET+(ERC*C1)**2)
    ERA=ERA/FAC*1000.0
    A=A/FAC*1000.0
    WRITE(2,700) NL,C,E1,ERC,A,ERA,A12
700 FORMAT(4X,I3,F10.2,F11.2,F8.2,F11.2,F10.2,F9.2)
    IF(NL-NP*60) 30,10,10
 99 WRITE (2,200)
    WRITE (2,300)
    WRITE (2,200)
    RETURN
    END
    END$
```

```
      SUBROUTINE ALAN (A,B,N)
      J=A/10.0
      FJ=J*10
      FA=A-FJ
      FJ=J
      T=FA**N+FJ**N*10.0**N
      T0=1
      FAC=1
      M=N-1
      IF(M) 3,3,2
    2 DO 1 J=1,M
      T0=T0*FLOAT(J)
      J0=J-1
      FAC=FAC*FLOAT(N-J0)
      I=N-J
      Z=FA**I
      Y=FJ**J
    1 T=T+FAC*Z*Y*10.0**J/T0
    3 B=T
      RETURN
      END
      END$
```