KfK 4184
Februar 1987

Computer Code FIT

D. Rohmann, T. Köhler
Institut für Kernphysik

Kernforschungszentrum Karlsruhe

KERNFORSCHUNGSZENTRUM KARLSRUHE

Institut für Kernphysik

KfK 4184

Computer Code FIT

Dorothee Rohmann and Thomas Köhler

Kernforschungszentrum Karlsruhe GmbH, Karlsruhe

Kernforschungszentrum Karlsruhe GmbH
Postfach 3640, 7500 Karlsruhe 1

ISSN 0303-4003

Abstract:

This is a description of the computer code FIT, written in FORTRAN-77 for a PDP 11/34. FIT is an interactive program to decude position, width and intensity of lines of X-ray spectra (max. length of 4K channels). The lines (max. 30 lines per fit) may have Gauss- or Voigt-profile, as well as exponential tails. Spectrum and fit can be displayed on a Tektronix terminal.

Zusammenfassung:

Es wird der Computer Code FIT beschrieben, der in FORTRAN-77 für eine PDP 11/34 geschrieben wurde. FIT ist ein interaktives Programm zur Bestimmung der Position, Breite und Intensität der Linien von X-ray Spektren (max. 4K Kanäle lang). Die zu bestimmenden Linien (max. 30 Linien pro Durchlauf) können Gauss- oder Voigt-Profil haben, sowie exponentielle Füße besitzen. Spektrum und Fit können auf einem Tektronix Terminal angesehen werden.

Contents

Figures

1. Introduction

FIT is an interactive spectrum fit program. It was – in its present form – built up for a LSI and/or a PDP computer on the basis of EVAL (and later FITOS) during the years 1983/84 by D.Rohmann and Th.Köhler.

The program-language is FORTRAN-77 except for some library-routines written in MACRO. The program is menu driven (MNPACK in RSXLIB [1]). FIT needs a Tektronix 4010 (or compatible) terminal to display spectra and fits.

The major part of the program is either handling Input/Output (routines FITIO, FITPFL, FITSPR, FITENE, FITOUT, FITCUR, FITST1, FITST2, FITPRT, FITSFP, FITROU) or controlling the various possibilities of the code (routines FITMEN, FITDIS, FITSET) in a dialogue. The fitting itself is done in the routines FITFIT, FITBAC, and FITPEA. The routines FITPAR and PEKDEF do some preparation for the fit procedure. The main control is handled by the main program FIT. (The routines SETUP and FITEND are subroutines only because of memory space problems. Logically they are part of the main program FIT.) A detailed description of the single program-parts is given in chapter 4.

The high segmentation of the FIT-program (see chapter 2) is not only due to modularity but because of the relatively small address space of 64 kbytes of a LSI/PDP computer with 16 bit addressing.

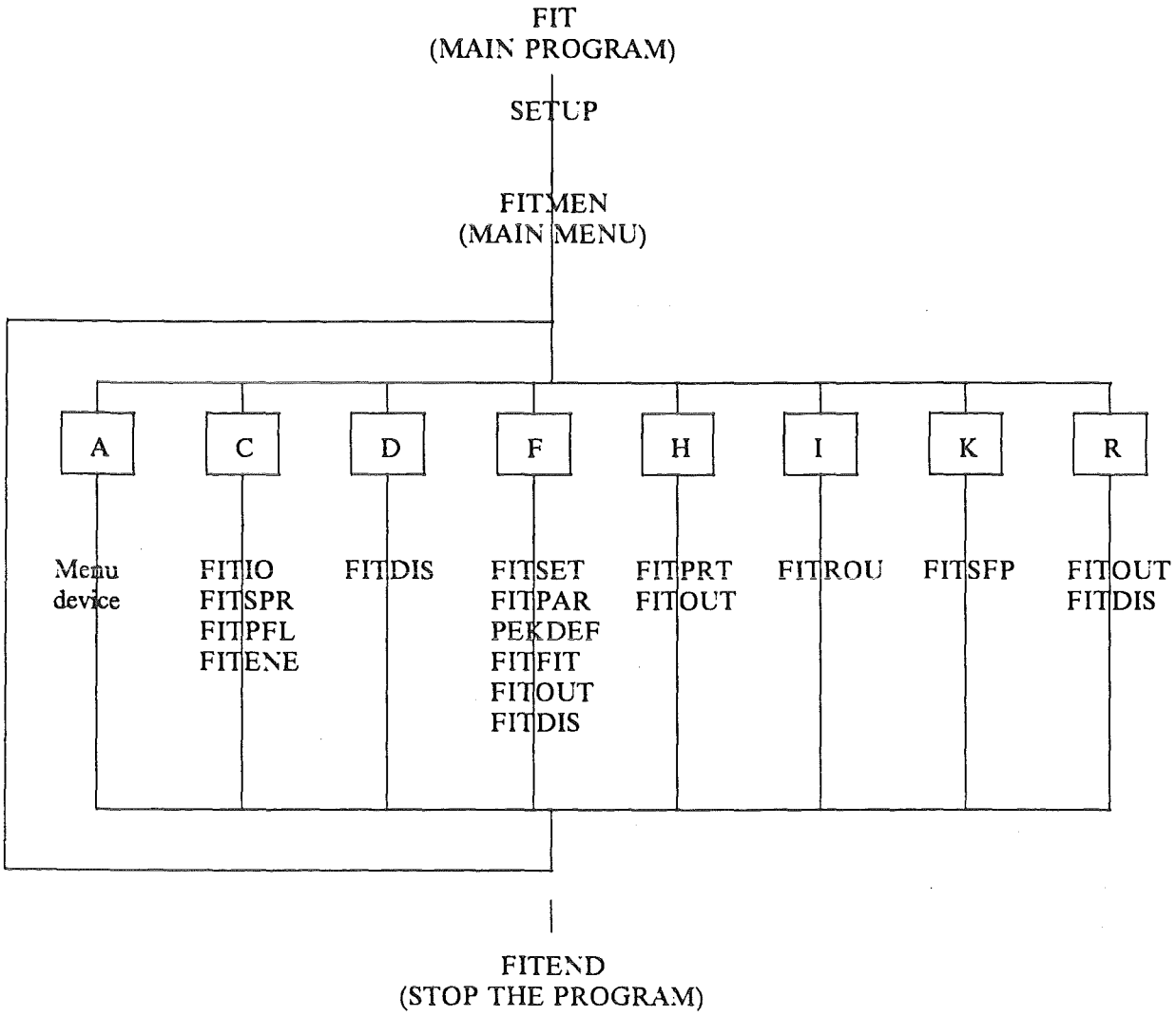
The FIT-program can handle a maximum of 30 peaks in a time. (This limit is given only by the space restrictions mentioned above.) The peaks can either have pure Gaussian form or be the convolution of a Lorentzian curve with a detector resolution (Gaussian line shape), or a pure Gaussian with an exponential low or high tail. For the mathematical formalism see chapter 6 and for the description of the control of these possibilities see chapter 4.

In chapter 5 the different Input/Output files of FIT are described.

Chapter 7 describes other tasks which deal with the FIT. Such as FITSPY, a task which reads in a temporary parameter file written by FITFIT before each iteration, and prints the peak-parameters on the line-printer. This is very useful to have a look to the developement of the fit.

2. Program structure

2.1 Flow chart



2.2 F77-Compiler file (FITF77.CMD)

This file is used to compile all modules of FIT.

```
FIT,FIT = FIT
SETUP,SETUP = SETUP
FITMEN,FITMEN = FITMEN
FITIO,FITIO = FITIO
FITDIS,FITDIS = FITDIS
FITROU,FITROU = FITROU
FITBAC,FITBAC = FITBAC
FITPEA,FITPEA = FITPEA
FITSET,FITSET = FITSET
FITCUR,FITCUR = FITCUR
FITST1,FITST1 = FITST1
FITST2,FITST2 = FITST2
FITPAR,FITPAR = FITPAR
PEKDEF,PEKDEF = PEKDEF
FITFIT,FITFIT = FITFIT
FITOUT,FITOUT = FITOUT
FITPRT,FITPRT = FITPRT
FITEND,FITEND = FITEND
FITSFP,FITSFP = FITSFP
```

Some routines of the program not shown here are in the following program libraries:

GENLIB.OLB (modules of general interest)
DISLIB.OLB (modules concerning the display)
RSXLIB.OLB (system modules, written at CERN)

or they are standard FORTRAN run time modules.

2.3 Task-builder files

The program FIT is so large that it can only be run by segmentating the various program parts. Therefore the task is overlayed. The following describes the two task-builder files needed for this overlay.

2.3.1 TKB command-file (FIT.CMD)

The following TKB command-file is used to build the task FIT :

```
FIT/FP,FIT = FIT/MP
TASK = FIT
ASG = TI:6
//
```

2.3.2 TKB overlay-description-file (FIT.ODL)

The TKB overlay-description-file looks like the following :

```
            .PSECT  PARFIC
MNPACK:     .FCTR   LB:[1,1]RSXLIB.OLB/LB:MNPACK:MNEXPK
LIBR:       .FCTR   GLIB-DLIB-RLIB-FLIB
  GLIB:     .FCTR   US1:[374,101]GENLIB.OLB/LB
  DLIB:     .FCTR   US1:[374,101]DISLIB.OLB/LB
  RLIB:     .FCTR   LB:[1,1]RSXLIB.OLB/LB
  FLIB:     .FCTR   US1:[1,1]F77DSKIO.OLB/LB
SEG1:       .FCTR   SETUP-FITMEN-FITEND-FITROU-FITSFP-LIBR
SEG2:       .FCTR   FITIO-LIBR-*(SEG21,SEG22,SEG23)
  SEG21:    .FCTR   FITPFL-LIBR
  SEG22:    .FCTR   FITSPR-LIBR
  SEG23:    .FCTR   FITENE-LIBR
SEG3:       .FCTR   FITDIS-FITBAC-FITPEA-LIBR
SEG4:       .FCTR   FITSET-LIBR-*(SEG41,SEG42,SEG43,SEG44)
  SEG41:    .FCTR   FITCUR-LIBR
  SEG42:    .FCTR   FITST1-LIBR
  SEG43:    .FCTR   FITST2-LIBR
  SEG44:    .FCTR   FITPAR-FITBAC-PEKDEF-LIBR
SEG5:       .FCTR   FITFIT-RLIB-FLIB-*(SEG51,SEG52)
  SEG51:    .FCTR   FITPEA-FITBAC-LIBR
  SEG52:    .FCTR   GLIB:SPXINV:BEEP:VDZERO-LIBR
SEG6:       .FCTR   FITOUT-FITPRT-LIBR
            .END
```

3. Description of PARAMETERs and COMMON-blocks (FIT.COM)

In general all REAL-variables are REAL*8 (Double Precision) variables : (IMPLICIT REAL*8 (A-H,O-Z)).

3.1 PARAMETERs

LUNSPE = 1	Logical unit used for spectrum input
LUNCAL = 2	Logical unit used for calibration input or energy table
LUNPRT = 3	Logical unit used for printer
LUNPAR = 4	Logical unit used for parameter file (PARFIL)
LUNTRM = 5	Logical unit used for terminal I/O (incl. DLPACK and MNPACK)
MAXCHN = 4096	Maximum number of channels in spectrum
MAXPK = 30	Maximum number of peaks possible
MAXCO = 6	Maximum number of coefficients for the background polynomial
MAXREG = 7	Maximum number of fit regions
MAXPAR = 40	Maximum dimension of Matrix
NPPARM = 6	Number of peak-parameters
MXCALF = 3	Maximum nr.of coeff. for the calib. polynomial
MPOINT = 500	Number of points for display-buffer
MXLINK = 20	Maximum number of linked peak-parameters

3.2 FILEs – Byte arrays

SPESPC(30)	Spectrum file name
PARSPC(30)	Parameter file name
CALSPC(30)	Calibration file name
FILPRT(30)	Print-output file name
ENESPC(30)	Energy table file name
SAVFIL(30)	Plotter-output file name

3.3 COMMON-blocks

/SPECTR/ JCOUNT(MAXCHN)

JCOUNT()	[I*4] Spectrum buffer (maximum dimension MAXCHN)

/SPDESC/ NXCHAN, NYCHAN, IDATIM(9), NRUN, ISPILL, ITARG, ITEXT(18), LABEL

NXCHAN	[I*2] Number of X-channels
NYCHAN	[I*2] Number of Y-channels
IDATIM()	[I*2] Date of data saving (FORMAT : dd-mmm-yy_hh:mm:ss)
NRUN	[I*2] Run number
ISPILL	[I*2] Spill number
ITARG	[I*2] Target number
ITEXT()	[I*2] Text (comment)
LABEL	[I*2] Label number (i.e. detector number)

/OPPARS/ MDEV, SPESPC, PARSPC, MXITER, LBFIX, INREG, LIXLO, LIXHI, LIYLO, LIYHI, IYPWR, LIFLO, LIFHI, IPEAK, LBACK, LPEAK, LPEBA, NCOEFF, COEFF(MAXCO), ITERA, NFREE, PEAK(NPPARM,MAXPK), IFLAG(NPPARM,MAXPK), INDEX(NPPARM,MAXPK), ISELEC, IGNOR(MAXPK), NREG, KREGLO(MAXREG), KREGHI(MAXREG), IMODE, INTCOR, NCALF, CALFAC(MXCALF), SIGMA(MAXPK), GAMMA(MAXPK), CALSPC, FILPRT, ENESPC

MDEV	[I*2] Menu device type
SPESPC()	[L*1] Spectrum file name
PARSPC()	[L*1] Parameter file name
MXITER	[I*2] Maximum number of iterations
LBFIX	[I*2] Flag controlling the fitting of the background
INREG	[I*2] Index of fit region
LIXLO	[I*2] Lower X limit for display
LIXHI	[I*2] Higher X limit for display
LIYLO	[I*2] Lower Y limit for display
LIYHI	[I*2] Higher Y limit for display
IYPWR	[I*2] Power of 10 for Y display
LIFLO	[I*2] Lower X limit for fit region
LIFHI	[I*2] Higher X limit for fit region
IPEAK	[I*2] Peak index
LBACK	[I*2] Flag controlling the display of the background
LPEAK	[I*2] Flag controlling the display of the peak
LPEBA	[I*2] Flag controlling the display of the fit
NCOEFF	[I*2] Number of background coeff.

COEFF()	[R*8] Background coefficients
ITERA	[I*2] Number of iterations
NFREE	[I*2] Number of degrees of freedom
PEAK(,)	[R*8] Peak parameters
IFLAG(,)	[I*2] Flags for peak-parameters
INDEX(,)	[I*2] Peak indices (flags)
ISELEC	[I*2] Selected spectrum
IGNOR()	[I*2] Ignore-flags
NREG	[I*2] Number of fit regions
KREGLO()	[I*2] Low end of fit regions
KREGHI()	[I*2] High end of fit regions
IMODE	[I*2] Display mode
INTCOR	[I*2] Status of internal start-value correction
NCALF	[I*2] Number of calibration coefficients
CALFAC()	[R*8] Calibration coefficients
SIGMA()	[R*8] Internal Gaussian widths
GAMMA()	[R*8] Internal Lorentzian widths
CALSPC()	[L*1] Calibration file name
FILPRT()	[L*1] Print-output file name
ENESPC()	[L*1] Energy table file name

/OPPART/ IOPARS

IOPARS	[I*2] Dummy variable

/RESULT/ CHINEW, CHIOLD, FVBACK, FVPEAK

CHINEW	[R*8] χ^2 of fit
CHIOLD	[R*8] χ^2_{old} of previous iteration
FVBACK	[R*8] Contribution of background
FVPEAK	[R*8] Contribution of peaks

/ERRCOM/ IERROR

IERROR	[I*2] RSXLIB error codes

/PARAM/ ILINK(NPPARM,MAXPK), NPAR, NDIMM, KUSE, KSKIP, TOLO(6), TOHI(6)

ILINK(,)	[I*2] Flags for linked parameters
NPAR	[I*2] Number of fit-parameters
NDIMM	[I*2] Dimension of matrix
KUSE	[I*2] Number of used parameters
KSKIP	[I*2] Number of skipped parameters
TOLO()	[R*8] Lower limits for fit
TOHI()	[R*8] Higher limits for fit

/ARRAYS/ VMAT(MAXPAR*(MAXPAR + 1)/2), DERIV(MAXPAR), RVEC(MAXPAR), HDERIV(MAXPAR)

VMAT()	[R*8] Matrix
DERIV()	[R*8] Derivatives
RVEC()	[R*8] R-Vector
HDERIV()	[R*8] Parameter increments

/CONST/ PICON, EPSILN, SQRTPI, SRLN2

PICON	[R*8] Constant π
EPSILN	[R*8] Radius of convergence for $\chi^2 - \chi^2_{old} := 1$
SQRTPI	[R*8] $\sqrt{\pi}$
SRLN2	[R*8] $\sqrt{\ln(2)}$

/GLOBAL/ IOP, T0, IFIT, FILOPN

IOP	[I*2] Meun control parameter
T0	[R*4] Start-time of fit
IFIT	[I*2] Fit-status control parameter
FILOPN	[L*2] = .TRUE. : Print-output file is open

4. Short description of the functions and subroutines

In this chapter we introduce the FUNCTIONs and SUBROUTINEs in alphabetical order. For each program section a more or less brief description of the function and/or the method of solution is given. The calling sequence is shown and the parameters – if any – are explained. The routines and library-programs called are listed and the library indicated. If the module is menu driven, the menu is shown and explained – if necessary –. In order to give a better overview the "Master"-program(s) are mentioned also.

4.1 CALVAL – Calibration of peak-parameters

Purpose : This function returns the calibrated value of a "channel-value" (if there is any calibration). The function CALVAL is part of the FITOUT.FTN file. The calibration is done via the polynomial :

$$E = a_0 + a_1 \times chn + a_2 \times chn^2$$

The calibration coefficients are read in via option -K- of the input-menu of FIT (FITIO).

Calling sequence : Result = CALVAL (DCHAN)

where DCHAN is the channel-value

Subroutines and library-programs needed : – – – none – – –

Menu-description : – – – none – – –

Called from :

FITOUT – Output of results

4.2 FIT – Main program

Purpose : This is the main-program for the interactive fit of spectra. Together with SETUP and FITEND the main-program controls the program-flow.

Calling sequence : – – – none – – –

Subroutines and library-programs needed :

DIINI – Initialize display (DIPACK in DISLIB)
FITDIS – Display spectrum (and fit)
FITEND – Terminate program
FITFIT – Fit the spectrum
FITIO – I/O of spectrum and parameter file
FITMEN – Master menu of program FIT
FITOUT – Output of results
FITPRT – Write output to print-output file
FITROU – Route print-output file to spooler
FITSET – Submenu for fit (option -F- of main-menu)
FITSFP – Write output to plotter-output file
SETUP – Set up program parameters

Menu-description : – – – none – – –

Called from : – – – none – – –

4.3 FITBAC – Contribution of background to fit

Purpose : This subroutines calculates at a given channel –X– the contribution of the background and –if fit processing– its derivative. The background is taken as a polynomial:

$$Y = a_0 + a_1 \times chn^1 + \ldots + a_5 \times chn^5$$

The background coefficients are either set by option -E- of the fit-menu (FITSET) of program FIT or determined automatically by the program itself (FITPAR).

Calling sequence : CALL FITBAC (X,IBACK)

where	X	is the actual channel
and		< 0 : Only one peak |IBACK| for display and output
	IBACK	= 0 : All peaks for fit processing
		> 0 : All peaks for display and output

Subroutines and library-programs needed : – – – none – – –

Menu-description : – – – none – – –

Called from :

FITDIS	– Display spectrum (and fit)
FITFIT	– Fit the spectrum
PEKDEF	– Set default values for peak

4.4 FITCUR – Peak parameters via crosshair cursor

Purpose : This subroutine handles the option -N- of the fit-menu (FITSET) of program FIT. It sets the fit-parameters in cursor mode.

Before displaying data, all peak-parameters and flags are set to zero, all peaks are disabled, and the fit-region(s) are reset. Thus you have to define the low and the high end(s) of the fit-region(s). With X and Y you have the possibility to change the display region. At the end of FITCUR the option -L- (Start value correction) of the fit-menu (FITSET) is set to YES to determine the start-values for the Gauss-FWHM and the height of each peak as well as to set some background start-parameters (FITPAR, and PEKDEF).

Cursor handling characters :

H	Define high end of fit region
L	Define low end of fit region
P	Define peak position
X	Define new X-display
Y	Define new Y-display
blank	Return to calling program

Each character entered ("H", "L", "P") is displayed on the screen to have a control of positions already marked (MARKER).

Calling sequence : CALL FITCUR

Subroutines and library-programs needed :

DIANU	– Set alpha mode (DIPACK in DISLIB)
DIHOME	– Set alpha mode and clear home (DIPACK in DISLIB)
DIPAGE	– Erase screen, set alpha mode, and clear home (DIPACK in DISLIB)
DIPNT	– Set point mode (DIPACK in DISLIB)
DJCURI	– Display the cursor and wait for response (DISLIB)
DL....	– Terminal I/O routines (DLPACK in RSXLIB)
MARKER	– Mark the cursor position on screen
PREPDI	– Prepare display (DISLIB)
SDISPL	– Display spectrum (DISLIB)

Menu-description : – – – none – – –

Called from :

FITSET	– Submenu for fit (option -F- of main-menu)

4.5 FITDIS – Display spectrum (and fit)

Purpose : This subroutine handles the option -D- of the main-menu (FITMEN) of program FIT. It shows the menu for display and displays the spectrum and optionally the contribution of the peaks and/or the background. The setting of the limits in X = channel (options -A- and -B-) and Y = countrate (options -D- and -E-) contains an automatic scaling: If the "lower...limit" is greater than the "higher...limit" the program searches for both the smallest/greatest values and takes these as lower/higher limit. The "power of 10" (option -F-) is needed for a Y higher than $2^{15} - 1 = = 32767$ (normal Integer*2 limit). Options -H- (Display of background) and -J- (Display of peak and background) are automatically set to YES after fit. Option -I- (Display of peaks separately) gives the display of each individual peak as well as the background in the display region. The display mode can be changed in option -N-. The display is started by option -R-.

Calling sequence : CALL FITDIS

Subroutines and library-programs needed :

DJDRWA – Draw absolute light vector in user space (DJABS in DISLIB)
DJMOVA – Draw absolute dark vector in user space (DJABS in DISLIB)
DJWNDW – Define screen window in user coordinates (DISLIB)
DL.... – Terminal I/O routines (DLPACK in RSXLIB)
FITBAC – Contribution of background to fit
FITPEA – Contribution of peaks to fit
MN.... – Menu routines (MNPACK in RSXLIB)
PREPDI – Prepare display (DISLIB)
PRTHDR – Display header (GENLIB)
SDISPL – Display spectrum (DISLIB)

Menu-description :

Display spectrum, peaks and background

A Lower X limit []
B Higher X limit []
D Lower Y-limit []
E Higher Y limit []
F Power of 10 for Y – low, Y – high []
H Display of background []
I Display peaks separately []
J Display of peak + background []
N Display mode: 0 = Histo, 1 = Err, 2 = Spec, 3 = Point []
R Redisplay

Called from :

FIT – Main program

4.6 FITEND – Terminate program

Purpose : This subroutine saves the parameters in COMMON-block /OPPARS/ to US1:[250,2]FIT.LAS and stops the FIT program. Since it often happened that the user didn't really want to stop but hit the "1" or "2" (= stop) in the main-menu by chance, a question was introduced to confirm the exit with the default NO:

—— > Do you really want to stop ? [NO]

Calling sequence : CALL FITEND

Subroutines and library-programs needed :

DL.... – Terminal I/O routines (DLPACK in RSXLIB)
MN.... – Menu routines (MNPACK in RSXLIB)
PAR... – Disk I/O routines (PARFIL in RSXLIB)

Menu-description : – – – none – – –

Called from :

FIT – Main program

4.7 FITENE – Read energy table

Purpose : This subroutine handles the option -M- of the input-menu (FITIO) of program FIT. In order to help the user with the setting up of fit-parameters you have the possibility to read an energy table (see chapter 5.5.) containing reference energies in keV. These are calculated into channel-numbers by the program. Thus the positions are set according to the internal calibration.

Calling sequence : CALL FITENE

Subroutines and library-programs needed : – – – none – – –

Menu-description : – – – none – – –

Called from :

FITIO – I/O of spectrum and parameter file

4.8 FITFIT – Fit the spectrum

Purpose : This subroutine handles the option -S- of the fit-menu (FITSET) of program FIT. It fits the spectrum by doing the following for each iteration (maximum MXITER) :

- Set widths to formula-parameters, e.g. the FWHM (PEAK(2/6,ipeak)) is set to :
  Gauss : SIGMA = σ_0 = PEAK(2,ipeak) / SRLN2
  Lorentz : GAMMA = Γ = PEAK(6,ipeak) / 2.
  Save old fit-parameters for this iteration to a temporary file FIT.TMP. This file is used by FITSPY (see chapter 7.) to print the fit-parameters on the line-printer for each iteration (format like in FITOUT).
- Set up arrays for next interation. CHIOLD holds old CHI. Reset R and V
- Start the loop over fit regions
- Start the loop over the channels
- Reset YFIT, DERIV. Define COUNT = actual countrate
- Calculate contribution of background (FITBAC)
- Calculate contribution of peak(s) (FITPEA)
- Calculate V and R
- End of loop over channels
- End of loop over fit regions
- Calculate parameter increments H
- Update background coefficients
- Update peak-parameters (Special treatment for linked peak height parameters)
- Reset widths to physics values
- Compare old and new χ^2

If $\chi^2_{old.} - \chi^2_{new}$ is greater than 1 the next iteration is started, if this difference is less than 1, convergence is reached. Then the fit procedure is stopped and terminal "beep" notifies the user of the end of the procedure.

After each iteration (including the 0. = start-values) a message is given to the screen containing the χ^2/NFREE (NFREE is the number of degrees of freedom) and the elapsed (not CPU) time used so far.

If the update of a peak-parameter would exceed its boundaries (FITPAR) the program gives the message:

—— > Parm. n of Peak m out

and skips the update. The parameters are: 1 = position, 2 = Gauss-FWHM, 3 = height, 4 = low tail, 5 = high tail, 6 = Lorentz-FWHM. If the parameter is linked to other peaks the increment is globally determined and updated for all peaks with that linking parameter. Is the Gaussian width of a Lorentzian peak linked to the non-Lorentzian Gauss width(s) the update of the Gaussian width will be done in the same way as the non-Lorentzian Gaussian width(s), but not used for the determination. This is done to change the energy resolution of the Lorentzian peak according to the "fitted" resolution in the neighbourhood from iteration to iteration.

The message

—— > *** Bad matrix ***

means that the determinant of the inverse matrix is zero. This (error)-message is mainly caused by a Gauss-FWHM less than 1. Other reasons may be a zero background value.

Calling sequence : CALL FITFIT

Subroutines and library-programs needed :

BEEP – Beep on terminal (GENLIB)
DL.... – Terminal I/O routines (DLPACK in RSXLIB)
FITBAC – Contribution of background to fit
FITPEA – Contribution of peaks to fit
PAR... – Disk I/O routines (PARFIL in RSXLIB)
SPXINV – Matrix–inversion (GENLIB)
VDZERO – Reset R*8 array (GENLIB)

Menu-description : – – – none – – –

Called from :

FIT – Main program

4.9 FITIJ – Look up the values of the inverse matrix

Purpose : This function returns the value of element (I,J) of the packed symmetric matrix V. The function FITIJ is part of the FITOUT.FTN file. It calculates the index in the packed matrix and checks for "Out of range".

Calling sequence : Result = FITIJ (VMAT,NDIMM,I,J)

where	VMAT	is the packed upper symmetric matrix
	NDIMM	is the dimension of the matrix
and	I and J	are the indices of the matrix element

Subroutines and library-programs needed : – – – none – – –

Menu-description : – – – none – – –

Called from :

FITOUT – Output of results

4.10 FITIO – I/O of spectrum and parameter file

Purpose : This subroutine handles the option -C- of the main-menu (FITMEN) of program FIT. It displays the menu for Input/Output. The names of both the spectrum and the fit-parameter files have to be declared (options -A- and -B-) before the reading/writing. The label (i.e. detector) of the spectrum is defined by option -G- and the spectrum read in (FITSPR). The saved fit-parameters (option -E-) (FITPFL) are of no other use than to be read in (option -D-) by FIT again, there is no possibility to handle these files in a different way. Option -I- scans through the spectrum file (given by option -A-) and displays the header information for each label on the screen. At the end of option -I- the spectrum buffer contains the spectrum of the last label. The calibration file name (option -K-) is asked for and the calibration read in. It is absolutely necessary to have the correct label number already declared in option -G- since this number is the control for reading the corresponding calibration (see also chapter 5.3). In order to help the user with the setup of new parameter files, the option -M- (FITENE) allows you to read in a energy table (see also chapter 5.$myene.).

Calling sequence : CALL FITIO

Subroutines and library-programs needed :

DL.... – Terminal I/O routines (DLPACK in RSXLIB)
FITENE – Read energy table
FITPFL – Read/write fit-parameter file
FITSPR – Read spectrum file
MN.... – Menu routines (MNPACK in RSXLIB)

Menu-description :

Get spectrum / save fit-parameters

A Spectrum file []
B Fit parameter file []
D Save fit-parameters (FITPFL)
E Read fit-parameters (FITPFL)
G Get spectrum from disk (FITSPR) []
I Show spectra available in spectrum file
K Enter calibration polynomial []
M Enter energy table (FITENE) []

Called from :

FIT – Main program

4.11 FITMEN – Master menu of program FIT

Purpose : This subroutine displays the master menu and returns the option number to the calling program.

Calling sequence : CALL FITMEN

Subroutines and library-programs needed :

MN.... – Menu routines (MNPACK in RSXLIB)

Menu-description :

FIT 0.0 – – Interactive fit of spectra

A Menu device: 4 = T2, 6 = NB, 9 = D1 []
C Get spectrum / Save fit-parameters (FITIO).
D Display (FITDIS)
F Fit the spectrum (FITSET and co.)
H Write output to print-output file (FITPRT).
I Route print-output file to spooler (FITROU).
K Save spectrum and fit-parameters for plotting (FITSFP).
R Redisplay results (FITOUT).

Called from :

FIT – Main program

4.12 FITOUT – Output of results

Purpose : This subroutine prints the results on either the terminal screen (LUN = 5) or the printer-output file (LUN = 3). It calculates the errors for the background parameters and uses the calibration (if any) to print the results not only in channels but also in keV (energies) or eV (Gauss and Lorentzian FWHM). The output contains the background, the calibration polynomial, and the peak-parameters.

Calling sequence : CALL FITOUT (LUN)

where LUN is the logical unit number for the printout
= 3 : printer-output file
= 5 : terminal screen

Subroutines and library-programs needed :

CALVAL – Calibration of peak-parameters
DATE – Get actual date from CPU (GENLIB)
DL.... – Terminal I/O routines (DLPACK in RSXLIB)
FITLI – Look up the values of the inverse matrix
MN.... – Menu routines (MNPACK in RSXLIB)
TIME – Get actual time from CPU (GENLIB)

Menu-description : – – – none – – –

Called from :

FIT – Main program
FITPRT – Write output to print-output file

4.13 FITPAR – Prepare data for fit

Purpose : This subroutine handles the option -L- of the fit-menu (FITSET) of program FIT. It defines the boundaries for the fitted quantities, prepares data for fit and determines the number of free parameters. The boundaries are :

position	low end < – > high end
Gauss-FWHM	0 < – > high – low end
heigth	0 < – > 10^9
low tail	0 < – > 1
high tail	0 < – > 1
Lorentz-FWHM	0 < – > high – low end

If option -L- of the fit-menu (FITSET) is set to YES, it defines the fit-parameters HEIGHT (= parameter 3) and FWHM-GAUSS (= parameter 2) and it determines some background start coefficients by putting a straight line through the very low end (average of ± 5 channels) and the very high end (average of ± 5 channels) of the fit-region. If the background is a higher order (> 2) function the following coefficients (> 2) are set like :

$$c(i+1) = -c(i) / (\text{high end})^i$$

just to give it some non zero values (no magic behind!!)**The option -L- is obligatory for previously done option -N- (define peaks by crosshair cursor).**

Calling sequence : CALL FITPAR

Subroutines and library-programs needed :

DL....	– Terminal I/O routines (DLPACK in RSXLIB)
MN....	– Menu routines (MNPACK in RSXLIB)
PEKDEF	– Set default values for peak
VIZERO	– Reset I*2 array (GENLIB)

Menu-description : – – – none – – –

Called from :

FITSET	– Submenu for fit (option -F- of main-menu)

4.14 FITPEA – Contribution of peaks to fit

Purpose : This subroutines calculate at a given channel – X – the contribution of the peaks and (if fit) their derivatives (for the mathematics see chapter 6.) :

1. Lorentzian peak or

2. Gaussian peak and
   - exponential tail to the left (if any) or
   - exponential tail to the right (if any)

Calling sequence : CALL FITPEA (X,IBACK)

where	X	is the actual channel
and		< 0 : Only one peak |IBACK| for display and output
	IBACK	= 0 : All peaks for fit processing
		> 0 : All peaks for display and output

Subroutines and library-programs needed :

CWERF – Complex error function (GENLIB)

Menu-description : – – – none – – –

Called from :

FITDIS – Display spectrum (and fit)
FITFIT – Fit the spectrum

4.15 FITPFL – Read/write fit-parameter file

Purpose : This subroutine handles the options -D- and -E- of the input-menu (FITIO) of program FIT. It reads or writes the actual fit-parameter file whose name has to be declared (FITIO) before action. The saved fit-parameters are of no other use than to be read in by FIT again, there is no possibility to handle these files in a different way. For instance, to be able to print the fit-parameters on the line-printer you must use option -H- of the main-menu (FITMEN, and FITPRT).

Calling sequence : CALL FITPFL

Subroutines and library-programs needed :

PAR... – Disk I/O routines (PARFIL in RSXLIB)

Menu-description : – – – none – – –

Called from :

FITIO – I/O of spectrum and parameter file

4.16 FITPRT – Write output to print-output file

Purpose : This subroutine handles the option -H- of the main-menu (FITMEN) of program FIT. It opens the print-output file (if not already open) and sends the output of the fit results to the print-output file (see FITOUT). It asks you :

—— > Do you want a new output file ? [NO]

so that you have the possibility to send your output to different files.

Calling sequence : CALL FITPRT

Subroutines and library-programs needed :

DL....	– Terminal I/O routines (DLPACK in RSXLIB)
FITOUT	– Output of results
MN....	– Menu routines (MNPACK in RSXLIB)

Menu-description : – – – none – – –

Called from :

FIT	– Main program

4.17 FITROU – Route print-output file to spooler

Purpose : This subroutine handles the option -I- of the main-menu (FITMEN) of program FIT. It checks wether a print-output file is open or not. If yes, closes it and opens a new one. It routes the old file to the printer.

> This option is not yet implemented.

Calling sequence : CALL FITROU

Subroutines and library-programs needed :

DL.... – Terminal I/O routines (DLPACK in RSXLIB)
MN.... – Menu routines (MNPACK in RSXLIB)

Menu-description : – – – none – – –

Called from :

FIT – Main program

4.18 FITSET – Submenu for fit (option -F- of main-menu)

Purpose : This subroutine handles the option -F- of the main-menu (FITMEN) of program FIT. It displays the menu for the parameter definition. The maximum iteration count (option -A-) is 15. If you need more just start the fit a second time having option -L- set to NO if not already. Then the fit-procedure will continue at the conditions of the previous stop. The background is given by a polynomial of the form:

$$Y = a_0 + a_1 \times chn^1 + \ldots + a_5 \times chn^5$$

The maximum number of background coefficients (option -C-) that makes sense is 4 (allowed is 6). The higher the order of the background polynomial the greater the risk of fitting to nonsense. For a normal X-ray spectrum a background of the order 2 or 3 is absolutely sufficient. The background can be fixed (option -D-) to what ever values you want. The option -E- (to set the coefficients to some definite values) only shows up if you set the background fixed (YES in option -D-). The maximum number of fit regions (option -F-) is 7. For each fit region (option -G-) you have to define the low (option -H-) and the high end (option -I-). Since 30 peaks are allowed it sometimes is useful to have all peaks globally reset (option -K-). All peak-parameters (parameters 1 to 6, see FITPAR) are set to zero and simultaniously disabled (the peaks are all set to "ignor"). The internal start-value correction (option -L-) is automatically set to YES after the use of the cursor (option -N-) and to NO after the parameter setting (option -O-). For detailed informations concerning option -L- please see FITPAR, PEKDEF. Options -N- (FITCUR), -O- (FITST1), and -Q- (FITST2) give you the possibility to define the peaks. With option -S- you really start the fit. The peak-parameters will be prepared for the fit (FITPAR) and shown on the screen to ask for confirmation. If everything is correct the fit is started (FITFIT and co.), otherwise the setting up of parameters is again made available (FITSET and co.).

Calling sequence : CALL FITSET

Subroutines and library-programs needed :

DIPAGE – Erase screen, set alpha mode, and clear home (DIPACK in DISLIB)
DL.... – Terminal I/O routines (DLPACK in RSXLIB)
FITCUR – Peak parameters via crosshair cursor
FITPAR – Prepare data for fit
FITST1 – Peak parameters via menu
FITST2 – Gobal flag definition
MN.... – Menu routines (MNPACK in RSXLIB)
VDZERO – Reset R*8 array (GENLIB)
VIZERO – Reset I*2 array (GENLIB)

Menu-description

Set parameters for fit

A	Maximum iteration count []	
C	No of background coeffs. []	
D	Fixed background []	
(E	Set background coeff. [])	only if D is "YES"
F	No of fit regions []	
G	Define fit region []	
H	Lower end of fit region []	
I	Upper end of fit region []	
K	Reset peak-parameters and disable peaks	
L	Do internal start-value correction []	
N	Define peak by crosshair cursor	
O	Define peaks by setting parameters	
Q	Define flags globally	
S	Start the fit	

Called from :

FIT – Main program

4.19 FITSFP – Write output to plotter-output file

Purpose : This subroutine handles the option -K- of the main-menu (FITMEN) of program FIT. It saves spectrum and parameters for plotting on a main frame computer.

Calling sequence : CALL FITSFP

Subroutines and library-programs needed :

DL.... – Terminal I/O routines (DLPACK in RSXLIB)
MN.... – Menu routines (MNPACK in RSXLIB)

Menu-description : – – – none – – –

Called from :

FIT – Main program

4.20 FITSPR – Read spectrum file

Purpose : This subroutine handles the option -G- of the input-menu (FITIO) of program FIT. It reads in the spectrum. The spectrum file name has to be declared (FITIO) before action. The spectrum buffer will be reset (VJZERO), the label of the spectrum wanted is searched for and read in. A control message is given on the screen.

Calling sequence : CALL FITSPR

Subroutines and library-programs needed :

DL.... – Terminal I/O routines (DLPACK in RSXLIB)
MN.... – Menu routines (MNPACK in RSXLIB)
VJZERO – Reset I*4 array (GENLIB)

Menu-description : – – – none – – –

Called from :

FITIO – I/O of spectrum and parameter file

4.21 FITST1 – Peak parameters via menu

Purpose : This subroutine handles the option -O- of the fit-menu (FITSET) of program FIT. It modifies the fit-parameters if necessary. With option -A- you choose the peak whose parameters and/or flags you want to change or define. Options -C- to -H- give you the actual values of the peak-parameters for control and wait for the new values. If you don't want to enter a new value just give a slash (/). The (link-) flags are set and changes by options -K- to -O-. The maximum group identifier possible is 20. Option -R- tells you whether this peak will be at all taken into account during the fit or not. Thus you have the possibility to turn on and off peaks (for the fit and/or the display). After the end of the setting the option -L- of the fit-menu is set to NO (see FITSET, PEKDEF) since a correction of the values is normally not required.

Calling sequence : CALL FITST1

Subroutines and library-programs needed :

MN.... – Menu routines (MNPACK in RSXLIB)

Menu-description :

Changes to peak-parameters

A Peak Nr.
C Set position
D Set Gauss-FWHM
E Set height
F Set low tail
G Set high tail
H Set Lorentz-FWHM
J Position: -1 = fix, 0 = fit, >0 = group-id []
K FWHM-Gauss: -1 = fix, 0 = fit, >0 = group-id []
L Height: -1 = fix, 0 = fit, >0 = group-id []
M Low tail: -1 = fix, 0 = fit, >0 = group-id []
N High tail: -1 = fix, 0 = fit, >0 = group-id []
O FWHM-Lorentz: -1 = fix, 0 = fit, >0 = group-id []
R Ignore this peak (Do not fit) []

Called from :

FITSET – Submenu for fit (option -F- of main-menu)

4.22 FITST2 – Global flag definition

Purpose : This subroutine handles the option -Q- of the fit-menu (FITSET) of program FIT. It globally sets the flags for the parameters (if necessary). Sometimes it is very useful for time- and type-conservation to have this possibility. This subroutine asks you for the flag-number you would like to set or change :

1 Position	-1 = fix, 0 = fit, >0 = group-id
2 FWHM-Gauss	-1 = fix, 0 = fit, >0 = group-id
3 Height	-1 = fix, 0 = fit, >0 = group-id
4 Low tail	-1 = fix, 0 = fit, >0 = group-id
5 High tail	-1 = fix, 0 = fit, >0 = group-id
6 FWHM-Lorentz	-1 = fix, 0 = fit, >0 = group-id
8 Ignor-flag	YES / NO

Which flag do you want to set
(0 = No setting wanting) []

and then gives you this flag for each peak (which is not to be ignored) in order to change or confirm it. It will ask you for the flag number again after each global setting. If you want to go back to the fit-menu give it the "flag number" 0.

Calling sequence : CALL FITST2

Subroutines and library-programs needed :

DIPAGE – Erase screen, set alpha mode, and clear home (DIPACK in DISLIB)
DL.... – Terminal I/O routines (DLPACK in RSXLIB)

Menu-description : – – – none – – –

Called from :

FITSET – Submenu for fit (option -F- of main-menu)

4.23 MARKER – Mark the cursor position on screen

Purpose : This subroutine writes the hitted character when using the cursor to the graphic screen. Thus it gives the user a control, where he did set which mark to the spectrum. The subroutine MARKER is part of the FITCUR.FTN file.

Calling sequence : CALL MARKER (IX,IY,CHAR)

where	IX and IY	are the coordinates on the screen
and	CHAR	is the character to be written at position (IX,IY)

Subroutines and library-programs needed :

DIANU	– Set alpha mode (DIPACK in DISLIB)
DIDRWA	– Draw absolute light vector in screen space (DIABS in DISLIB)
DIMOVA	– Draw absolute dark vector in screen space (DIABS in DISLIB)
DIVEC	– Set vector mode (DIPACK in DISLIB)

Menu-description : – – – none – – –

Called from :

FITCUR	– Peak parameters via crosshair cursor

4.24 PEKDEF – Set default values for peak

Purpose : This subroutine sets default values for a selected peak :

- First do some adjustement of the position (parameter 1). Find out in which fit region the peak is situated. The countrate in the two neighbouring channels is compared with the one of the actual position. The position start-value will be set to the channel with the highest countrate of these three channels unless there is again a peak defined within the range of two channels.

- Then set height (parameter 3). The height is set to the countrate of the position (channel) subtracted by the contribution of the background to this channel.

- Now try to estimate a Gauss-FWHM (parameter 2). If the distance to the nearest peak is less then 1.5 channels the position of the two peaks are set to a distances of 1.5 channels. The Gauss-FWHM is set to the channel difference of the locations where the background subtracted countrate is in the order of half the height of this peak. If within this difference there is another peak the Gauss-FWHM is set to half this value. In the case that this difference cannot be determined because the spectrum has no half-height countrate within the range of 15 channels the Gauss-FWHM is set to 15 channels.

The message

—— > Merde n

(where n is the peak number) notifies that the determined Gauss-FWHM was negative and now set to 1.5 channels.

Calling sequence : CALL PEKDEF (IPK)

where IPK is the peak number

Subroutines and library-programs needed :

DL.... – Terminal I/O routines (DLPACK in RSXLIB)
FITBAC – Contribution of background to fit

Menu-description : – – – none – – –

Called from :

FITPAR – Prepare data for fit

4.25 SETUP – Set up program parameters

Purpose : This subroutine sets up the fit program.

Calling sequence : CALL SETUP

Subroutines and library-programs needed :

DIWNDW – Define absolute display window of screen (DISLIB)
ERRSET – Check for error 29 : "No such file" (F77LIB)
MN.... – Menu routines (MNPACK in RSXLIB)
PAR... – Disk I/O routines (PARFIL in RSXLIB)
VJZERO – Reset I*4 array (GENLIB)

Menu-description : – – – none – – –

Called from :

FIT – Main program

5. General aspects of input/output files used by FIT

5.1 Spectrum file SPESPC

Normally a spectrum file (options -A- and -G- of the input-menu (FITIO) and FITSPR) has the extention .BIN or .SUM. (Of course it can have every other extension.) The general form of the spectrum file is :

1. Label = first detector

   FORTRAN record 1 contains the header-information (all INTEGER*2):

   NXCHAN, NYCHAN, IDATIM, NRUN, ISPILL, ITARG, ITEXT, LABEL

where	NXCHAN	is the length of the spectrum in x (energy)
	NYCHAN	is the length of the spectrum in y (time)
	IDATIM(9)	contains date and time of data – saving
	NRUN	is the run number
	ISPILL	is the spill number
	ITARG	is the target number
	ITEXT(18)	contains the spectrum description
	LABEL	is the label number (= diode number)

   FORTRAN record 2 contains the spectrum-buffer (INTEGER*4):

   JBUFF(I),I = 1,NXCHAN

   where JBUFF(4096) is the spectrum buffer

2. Label same as 1. Label

The highest label number used so far is 8, i.e. one spectrum file contains the spectra of maximum 8 detectors.

5.2 Fit-parameter file PARSPC

The fit-parameter file (options -B-, -D- and -E- of the input-menu (FITIO) and FITPFL) is read (written) via PARFIL (RSXLIB) with the COMMON-blocks : /OPPARS/ and /OPPART/. This means the whole COMMON-block /OPPARS/ is fetched from (dumped on) disk. The COMMON-block /OPPART/ is only necessary because this procedure is done in a subroutine and not in the main program [1].

5.3 Calibration file CALSPC

The calibration file (option -K- of the input-menu (FITIO)) should have the following format : (It is the same as needed for the calibration file for the LINPRE-program).

n, number of channels, 'text', a_0, a_1, a_2
.....
– 1/

where "n" gives the label number (detector number). "n" = – 1/ means the end of the file. (Don't forget the slash.)

5.4 Print-output file FILPRT

The print-output file (option -H- of the main-menu (FITMEN)) will contain all the informations that are written on the terminal after the fit was done, except for the picture, of course. It will contain also some printer control characters (such as "1" for "top of form").

The program RESULT takes this print-output file as an input file in order to do some modifications to the output-form or to produce a CALIB input file.

5.5 Energy table file ENESPC

The energy table file (option -M- of the input-menu (FITIO) and FITENE) has the following general structure : (It is the same as needed for the energy table file for the LINPRE-program).

' Z', A, 'trans', E_{weight}, E^{-+}, E^{++}, E^{--}

where	' Z'	is the two letter code of the element,
	A	is the atomic weight number,
	'trans'	is the atomic transition for exotic atoms or a description for nuclear γ-lines,
	E_{weight}	is the weighted energy for $\bar{p}$-atoms or the total energy for single lines,
	E^{ij}	are the fine structure components for $\bar{p}$-atomic lines or zero for single lines.

The energy table file will contain in most cases more than one of the above described input lines. The program will identify E_{weight} of the first line with the peak nr. 1 if the E^{ij} are zero or the E^{ij} with peaks nr. 1 (E^{-+}) to 3 (E^{--}) if those are none-zero no matter whether peak nr. 1 is ignored or not (etc.). So you have to be very careful in editing your energy tables.

5.6 Plotter-output file SAVFIL

The plotter-output file (option -K- of the main-menu (FITMEN)) saves some general informations concerning the part of the actual spectrum, the background coefficients, the peak-parameters, and a termination record:

LIXLO, LIXHI, LIYLO, LIYHI, IYPWR, LIFLO, LIFHI, IMODE, MAXREG, NREG, MAXCO, NCOEFF	(8I7,4I6)
KREGLO, KREGHI	(14I5)
COEFF	(4(1PE20.13))
J, (PEAK(K,J),K = 1,NPPARM)	(I2,6(1X,1PE12.6))
.....	MAXPK times
−1/	

6. Mathematics of the fit

This chapter gives you a – brief – description of the mathematical formula used for the fit.

6.1 Pure Gaussian peak

The pure Gaussian peak is fitted by the F(x) [2] as:

$$F(x) = H_0 \exp\left(- \left[(x - x_0) / \sigma_0 \right]^2 \right)$$

where : H_0 = Height
x_0 = Position in channels (center of gravity)
σ_0 = FWHM / ($2\sqrt{\ln 2}$)

and the intensity of the line is given by :

$$\text{Area} = \sqrt{\pi}\ \sigma_0\ H_0$$

The parameters x_0 (= parameter 1), FWHM (= parameter 2), and H_0 (= parameter 3) are given to the program either via the cursor (see FITCUR and co.) or by hand (see FITSET and co.).

6.2 Gaussian peak with exponential tail(s)

In addition to the pure Gaussian peak (see there) you have the possibility to modify the peak shape(s) by exponential tail(s). This is done by the definition (only possible in the parameter setting mode (FITSET and co.)) of a characteristic value t_0 whose sign determines the direction of the tail:

negative sign => low energy tail (parameter 4)
positive sign => high energy tail (parameter 5)

The value t_0 itself is the quotient of two parameters (a) and (b) which define the strength and form of the "anomaly". (a) gives the half width of the normal Gaussian peak at 1/e of the total height, (b) is the half width of the normal Gaussian peak at the height where the exponential tail should start:

$$t_0 = \pm\, b / a$$

The total peak is then fitted by G(x) [2] as :

$$G(x) = \begin{cases} H_0 \exp\left(\left[t_0 - 2\,(x - x_0) / \sigma_0 \right] t_0 \right) & \text{in the region of the tail} \\ H_0 \exp\left(- \left[(x - x_0) / \sigma_0 \right]^2 \right) & \text{elsewhere} \end{cases}$$

where : H_0 = Height
x_0 = Position in channels (center of gravity)
σ_0 = FWHM / ($2\sqrt{\ln 2}$)
t_0 = Exponential tail (including the sign)

and the intensity of the line is given by :

$$\text{Area} = \sqrt{\pi}\ \sigma_0\ H_0$$

The parameter t_0 (=parameter 4 or 5) can only be set by hand (see FITSET and co.). For all other parameters (meanings as described in chapter 6.1) both cursor and manual setting is possible.

6.3 Lorentzian peak folded with Gaussian energy resolution

The problem of fitting a Lorentzian line which has been measured with a detector of a non zero resolution (normally Gaussian line shape) was solved in the old FITOS/EVAL program [2] by a Voigt-formula. This means a folding mechanism using Hermite polynomials for the integration. Unfortunately this method is only valid [3] if the ratio Γ/σ_0 is greater than 0.5. If this ratio is smaller the Voigt-formula will lead to oscillations above the original line shape. These oscillations can be avoided using the complex error function as discussed in [3]:

The shape of a line with a natural width is of Lorentzian form :

$$I_0(E) = \frac{\Gamma}{2\pi} \cdot \frac{1}{E^2 + {}^1\!/_4 \Gamma^2}$$

be aproximately of Gaussian form :

$$D(E) = \exp\left(- [E/\sigma_0]^2 \right) / \left(\sigma_0 \sqrt{\pi} \right)$$

Thus the signal will be the folding of the two functions :

$$I(E) = \int_{-\infty}^{\infty} I_0(E') \, D(E - E') \, dE'$$

$$= \mathrm{Re}\, w(z) \times \left[E/\sigma_0 + i\,\Gamma/2\sigma_0 \right] / \left(\sigma_0 \sqrt{\pi} \right)$$

Here w(z) is the complex error function :

$$w(z) = \exp\left(- z^2 \right) \mathrm{erfc}\,(-iz)$$

$$\mathrm{erfc}(z) = 1 - \mathrm{erf}(z)$$

$$\mathrm{erf}(z) = 2/\sqrt{\pi} \times \int_0^z \exp(-t^2)\,dt$$

Since the first derivations of the complex error function are combinations of the same function and its arguments :

$$\mathrm{Re}\left(\frac{dw(z)}{dy}\right) = -2/\sqrt{\pi} + 2 \times x \times \mathrm{Im}\, w(z) + 2 \times y \times \mathrm{Re}\, w(z)$$

$$\mathrm{Re}\left(\frac{dw(z)}{dx}\right) = -2 \times x \times \mathrm{Re}\, w(z) + 2 \times y \times \mathrm{Im}\, w(z)$$

with $z = x + iy$

a single calculation of the complex error function not only gives the value of the Voigt integral but also gives its derivations with respect to x and y.

The Lorentzian peak is fitted by the L(x) as :

$$L(x) = H_\ell \ \mathrm{Re}\, w(z) \ \sqrt{\pi}\, \Gamma/\sigma_0$$

where : $w(z) = (x - x_0)/\sigma_0 + i\,\Gamma/\sigma_0$

H_ℓ = Height of the Lorentzian line
x_0 = Position in channels (center of gravity)
σ_0 = Gauss-FWHM / ($2\sqrt{\ln 2}$)
Γ = Lorentz-FWHM / 2

and the intensity of the line is given by :

$$\text{Area} = \pi\ \Gamma\ H_\ell$$

The parameters x_0 (= parameter 1) and Gauss-FWHM (= parameter 2) can be either set by cursor (see FITCUR and co.) or by hand (see FITSET and co.), the Lorentz-FWHM (= parameter 6) only by hand (see FITSET and co.).

The height of the Lorentzian line H_ℓ (= parameter 3) is connected to the "pure Gaussian height H_0" :

$$H_\ell = \frac{\sigma_0}{\Gamma\sqrt{\pi}} \times H_0$$

It can be set by cursor, but then it has to be corrected by hand.

7. FITSPY – A spy to the fitting procedure

FITSPY is a separate task which can be started by FIT. It has to be installed to get active. It uses the temporary fit-parameter file FIT.TMP written by FITFIT. This file is read in via PARFIL and printed on the line-printer. Thus the user has the possibility to control each iteration.

Note If the fitting time is smaler than the printing time the FITSPY will print only the parameters saved last.

References

[1] CERN RSXLIB library manual, Version 1.8 of 11 January 1983.
[2] R. Guigas, Thesis, Kernforschungszentrum Karlsruhe, KfK 3208, (Appendix), 1981
[3] C.J. Batty, S.D. Hoath, and B.L. Roberts, Nucl. Instrum. Meth.137 (1976) 179.

APPENDIX A

EXAMPLE

As an example for the output of FIT we present in figure 1 a part of the $\bar{p}$ ^{17}O spectrum containing the shifted and broadened atomic 4→3 transition. A print-output file is given on the next page.

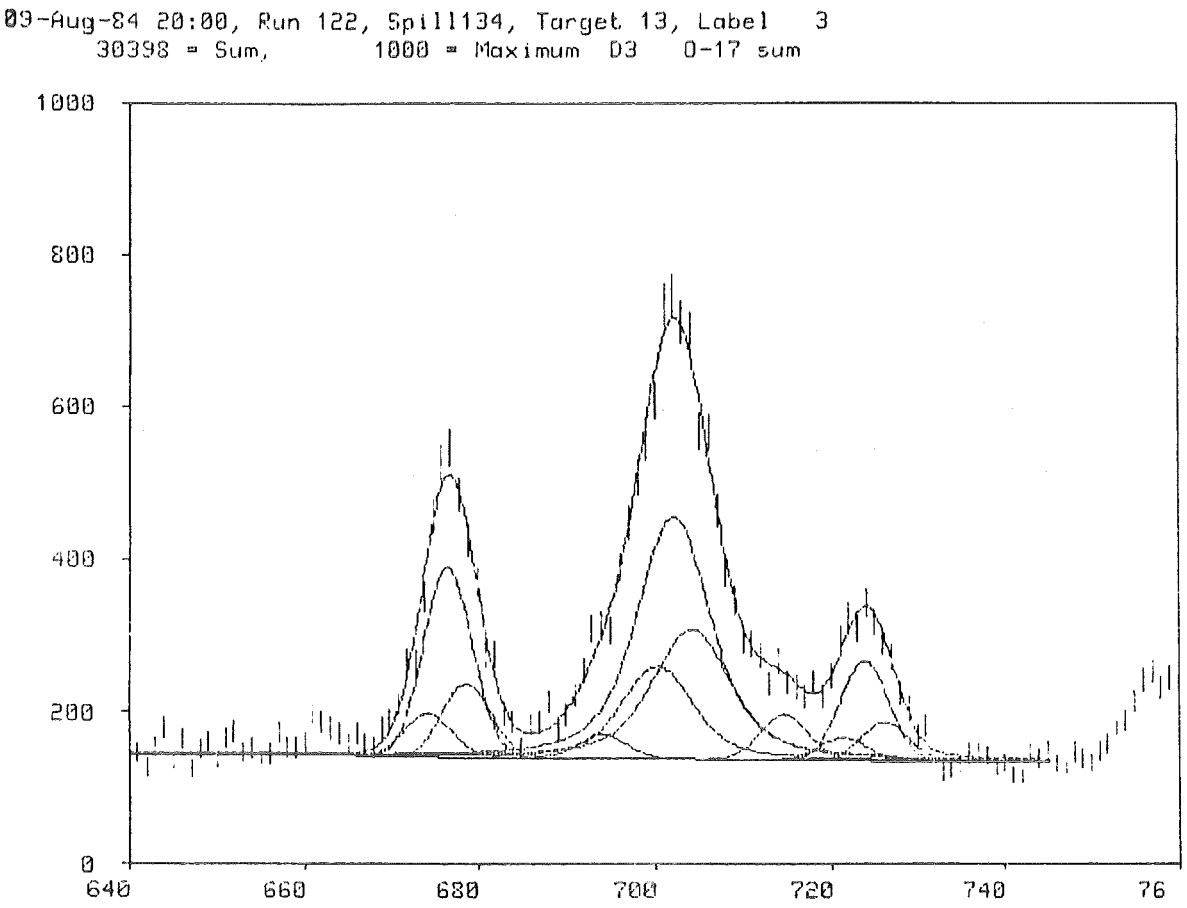


Figure 1: Fit to the 4→3 region of $\bar{p}$-atomic ^{17}O.

```
Output from 14-AUG-85  22:02:49                    DM2:[3122S7.017

08-AUG-84 23:03:41  Run 122, Split   S, Target 13 Label  3      GE I ENERGY

 3. Iteration. CHIsq2:    136.0871  NFREE:  119  CHIsq2/NFREE:   1.1436


Fit regions used :    620- 745


Background with  2 coefficients:

Coeffs:  3.88319607E+01 -2.54994220E-02
Errors:  8.91122148E+00  1.33337566E-02


Calibration polynomial:

Coeffs: -2.10750600E-01  1.05227300E-01

Position  Gauss-FWHM   Height Low-tail High-tail Lorentz-FWHM Intensity

Peak  1          (Flags:-1,-1, 4,-2,-2,-2)
 674.113   6.614          0.13    0.000    0.000    0.000          0.92
   0.000   0.000          0.01    0.000    0.000    0.000          0.05
E:  70.7243(  0.0000), G:   696.0(     0.0), L:     0.0(    0.0)


Peak  2          (Flags:-1,-1, 4,-2,-2,-2)
 674.134   6.614          4.57    0.000    0.000    0.000         32.16
   0.000   0.000          0.30    0.000    0.000    0.000          2.14
E:  70.7265(  0.0000), G:   696.0(     0.0), L:     0.0(    0.0)


Peak  3          (Flags:-1,-1, 4,-2,-2,-2)
 674.393   6.614          3.52    0.000    0.000    0.000         24.81
   0.000   0.000          0.23    0.000    0.000    0.000          1.65
E:  70.7538(  0.0000), G:   696.0(     0.0), L:     0.0(    0.0)


Peak  4          (Flags:-1,-1, 4,-2,-2,-2)
 676.478   6.614          0.58    0.000    0.000    0.000          4.07
   0.000   0.000          0.04    0.000    0.000    0.000          0.27
E:  70.9732(  0.0000), G:   696.0(     0.0), L:     0.0(    0.0)
1

Position  Gauss-FWHM   Height Low-tail High-tail Lorentz-FWHM Intensity

Peak  5          (Flags:-1,-1, 4,-2,-2,-2)
 676.489   6.614         20.23    0.000    0.000    0.000        142.44
   0.000   0.000          1.35    0.000    0.000    0.000          9.48
E:  70.9744(  0.0000), G:   696.0(     0.0), L:     0.0(    0.0)


Peak  6          (Flags:-1,-1, 4,-2,-2,-2)
 676.758   6.614         15.61    0.000    0.000    0.000        109.83
   0.000   0.000          1.04    0.000    0.000    0.000          7.32
E:  71.0019(  0.0000), G:   696.0(     0.0), L:     0.0(    0.0)


Peak  7          (Flags:-1,-1, 4,-2,-2,-2)
 678.576   6.614          0.22    0.000    0.000    0.000          1.58
   0.000   0.000          0.01    0.000    0.000    0.000          0.12
E:  71.1939(  0.0000), G:   696.0(     0.0), L:     0.0(    0.0)


Peak  8          (Flags:-1,-1, 4,-2,-2,-2)
 678.597   6.614          7.83    0.000    0.000    0.000         55.14
   0.000   0.000          0.52    0.000    0.000    0.000          3.67
E:  71.1962(  0.0000), G:   696.0(     0.0), L:     0.0(    0.0)


Peak  9          (Flags:-1,-1, 4,-2,-2,-2)
 678.859   6.614          6.04    0.000    0.000    0.000         42.53
   0.000   0.000          0.40    0.000    0.000    0.000          2.83
E:  71.2238(  0.0000), G:   696.0(     0.0), L:     0.0(    0.0)


Peak 10          (Flags:-1,-1, 4,-2,-2,-2)
 693.876   6.614          4.57    0.000    0.000    0.000         32.16
   0.000   0.000          0.30    0.000    0.000    0.000          2.14
E:  72.8040(  0.0000), G:   696.0(     0.0), L:     0.0(    0.0)


Peak 11          (Flags:-1,-1, 4,-2,-2,-1)
 699.418   6.614          0.84    0.000    0.000    4.752          6.28
   0.000   0.000          0.06    0.000    0.000    0.000          0.42
E:  73.3871(  0.0000), G:   696.0(     0.0), L:   500.0(    0.0)


Peak 12          (Flags:-1,-1, 4,-2,-2,-1)
 699.699   6.614         16.82    0.000    0.000    4.752        125.52
   0.000   0.000          1.12    0.000    0.000    0.000          8.36
E:  73.4167(  0.0000), G:   696.0(     0.0), L:   500.0(    0.0)


Peak 13          (Flags:-1,-1, 4,-2,-2,-1)
 700.751   6.614         11.77    0.000    0.000    4.752         87.86
   0.000   0.000          0.78    0.000    0.000    0.000          5.85
E:  73.5274(  0.0000), G:   696.0(     0.0), L:   500.0(    0.0)
1

Position  Gauss-FWHM   Height Low-tail High-tail Lorentz-FWHM Intensity

Peak 14          (Flags:-1,-1, 7,-2,-2,11)
 701.211   6.614          2.35    0.000    0.000    4.752         21.21
   0.348   0.000          0.49    0.000    0.000    1.264          5.84
E:  73.5758(  0.0366), G:   696.0(     0.0), L:   500.0(  111.9)


Peak 15          (Flags:-1,-1, 7,-2,-2,11)
 701.494   6.614         37.25    0.000    0.000    4.752        425.84
   0.348   0.000          3.84    0.000    0.000    1.264        119.45
E:  73.6055(  0.0366), G:   696.0(     0.0), L:   500.0(  111.9)


Peak 16          (Flags:-1,-1, 7,-2,-2,11)
 702.553   6.614         33.93    0.000    0.000    4.752        298.86
   0.248   0.000          6.89    0.000    0.000    1.264         83.53
E:  73.7170(  0.0366), G:   696.0(     0.0), L:   500.0(  111.9)


Peak 17          (Flags:-1,-1, 4,-2,-2,-1)
 703.295   6.614          1.23    0.000    0.000    6.177         10.81
   0.000   0.000          0.07    0.000    0.000    0.000          0.67
E:  73.7951(  0.0000), G:   696.0(     0.0), L:   650.0(    0.0)


Peak 18          (Flags:-1,-1, 4,-2,-2,-1)
 703.573   6.614         28.64    0.000    0.000    6.177        280.24
   0.000   0.000          1.77    0.000    0.000    0.000         19.33
E:  73.8249(  0.0000), G:   696.0(     0.0), L:   650.0(    0.0)


Peak 19          (Flags:-1,-1, 4,-2,-2,-1)
 704.643   6.614         14.45    0.000    0.000    6.177        140.17
   0.000   0.000          0.95    0.000    0.000    0.000          9.23
E:  73.9378(  0.0000), G:   696.0(     0.0), L:   650.0(    0.0)


Peak 20          (Flags:-1,-1, 4,-2,-2,-2)
 714.451   6.614          8.82    0.000    0.000    0.000         68.69
   0.000   0.000          0.57    0.000    0.000    0.000          4.04
E:  74.9698(  0.0000), G:   696.0(     0.0), L:     0.0(    0.0)


Peak 21          (Flags:-1,-1, 8,-2,-2,-2)
 721.055   6.614          2.03    0.000    0.000    0.000         0.58
   0.000   0.000          0.21    0.000    0.000    0.000         0.06
E:  75.6640(  0.0000), G:   696.0(     0.0), L:     0.0(    0.0)


Peak 22          (Flags:-1,-1, 8,-2,-2,-2)
 721.078   6.614          2.89    0.000    0.000    0.000         28.33
   0.000   0.000          0.29    0.000    0.000    0.000          2.81
E:  75.6655(  0.0000), G:   696.0(     0.0), L:     0.0(    0.0)
1

Position  Gauss-FWHM   Height Low-tail High-tail Lorentz-FWHM Intensity

Peak 23          (Flags:-1,-1, 8,-2,-2,-2)
 721.335   6.614          2.03    0.000    0.000    0.000         15.68
   0.000   0.000          0.22    0.000    0.000    0.000          1.65
E:  75.6935(  0.0000), G:   696.0(     0.0), L:     0.0(    0.0)


Peak 24          (Flags:-1,-1, 8,-2,-2,-2)
 723.375   6.614          0.37    0.000    0.000    0.000          2.87
   0.222   0.000          0.04    0.000    0.000    0.000          0.25
E:  75.9291(  0.0000), G:   696.0(     0.0), L:     0.0(    0.0)


Peak 25          (Flags:-1,-1, 8,-2,-2,-2)
 723.552   6.614          1.78    0.000    0.000    0.000         58.87
   0.222   0.000          0.22    0.000    0.000    0.000          8.33
E:  75.9307(  0.0000), G:   696.0(     0.0), L:     0.0(    0.0)


Peak 26          (Flags:-1,-1, 8,-2,-2,-2)
 723.852   6.614          0.85    0.000    0.000    0.000         69.44
   0.222   0.000          0.07    0.000    0.000    0.000          6.26
E:  75.9588(  0.0000), G:   696.0(     0.0), L:     0.0(    0.0)


Peak 27          (Flags:-1,-1, 8,-2,-2,-2)
 725.838   6.614          0.14    0.000    0.000    0.000          1.22
   0.222   0.000          0.01    0.000    0.000    0.000          0.12
E:  76.1664(  0.0000), G:   696.0(     0.0), L:     0.0(    0.0)


Peak 28          (Flags:-1,-1, 8,-2,-2,-2)
 725.946   6.614          4.95    0.000    0.000    0.000         34.33
   0.222   0.000          0.43    0.000    0.000    0.000          3.44
E:  76.1679(  0.0000), G:   696.0(     0.0), L:     0.0(    0.0)


Peak 29          (Flags:-1,-1, 8,-2,-2,-2)
 726.114   6.614          3.82    0.000    0.000    0.000         26.58

   0.000   0.000          0.33    0.000    0.000    0.000          2.56
E:  76.1362(  0.0000), G:   696.0(     0.0), L:     0.0(    0.0)
```

APPENDIX B

PROGRAM LISTINGS

The following pages give a listing of the FIT program with all its subroutines and functions, of the group-written library routines (GENLIB and DISLIB), and of the FITSPY program.

```
PDP-11 FORTRAN-77 V5.0-0          09:34:02     26-Sep-86          Page 1
FIT.FTN;1                  /F77/OP/TR:BLOCKS/WR

0001            PROGRAM FIT
          C*    **********
          C*
          C*    Interactive fit of spectra
          C*
          C*
0002            INCLUDE 'FIT.COM'
0003  *         IMPLICIT REAL*8 (A-H,O-Z)
      *   C*
      *   C*    Logical unit numbers used:
      *   C*
      *   C*    LUNSPE: Lun to read spectrum
      *   C*    LUNPAR: Lun for PARFIL
      *   C*    LUNPRT: Lun for print output
      *   C*    LUNCAL: Lun for calibration input, and energy table
      *   C*    LUNTRM: Lun for terminal I/o (incl. DLPACK, MNPACK)
      *   C*
0004  *         PARAMETER LUNSPE=1, LUNCAL=2, LUNPRT=3, LUNPAR=4, LUNTRM = 5
      *   C*
0005  *         PARAMETER MAXCHN=4096, MAXPK=30, MAXCO=6, MAXREG=7, MAXPAR=40,
      *  +                  NPPARM=6, MXCALF=3, MPOINT=500, MXLINK=20
      *   C*
0006  *         BYTE SPESPC(30),PARSPC(30),CALSPC(30),FILPRT(30),ENESPC(30)
      *   C*
0007  *         CHARACTER*20 HDGLIN
      *   C*
0008  *         INTEGER*4 JCOUNT
      *   C*
0009  *         REAL*4 T0,SECNDS
      *   C*
0010  *         COMMON /SPECTR/ JCOUNT(MAXCHN)
0011  *         COMMON /SPDESC/ NXCHAN,NYCHAN,IDATIM(9),NRUN,ISPILL,ITARG,
      *  +                      ITEXT(18),LABEL
0012  *         COMMON /OPPARS/ MDEV,SPESPC,PARSPC,MXITER,LBFIX,INREG,LIXLO,LIXHI,
      *  +                      LIYLO,LIYHI,IYPWR,LIFLO,LIFHI,IPEAK,LBACK,LPEAK,
      *  +                      LPEBA,NCOEFF,COEFF(MAXCO),ITERA,NFREE,
      *  +                      PEAK(NPPARM,MAXPK),IFLAG(NPPARM,MAXPK),
      *  +                      INDEX(NPPARM,MAXPK),ISELEC,IGNOR(MAXPK),NREG,
      *  +                      KREGLO(MAXREG),KREGHI(MAXREG),IMODE,INTCOR,NCALF,
      *  +                      CALFAC(MXCALF),SIGMA(MAXPK),GAMMA(MAXPK),CALSPC,
      *  +                      FILPRT,ENESPC
0013  *         COMMON /OPPART/ IOPARS
0014  *         COMMON /RESULT/ CHINEW,CHIOLD,FVBACK,FVPEAK
0015  *         COMMON /LRHCOM/ ILERROR
0016  *         COMMON /PARAM/  ILINK(NPPARM,MXLINK),NPAR,NDIMM,KUSE,KSKIP,
      *  +                      TOLO(6),TOHI(6)
0017  *         COMMON /ARRAYS/ VMAT(MAXPAR*(MAXPAR+1)/2),DERIV(MAXPAR),
      *  +                      RVEC(MAXPAR),HDERIV(MAXPAR)
0018  *         COMMON /CONST/  PICON,EPSILN,SQRTPI,SRLN2
      *   C*
0019  *         LOGICAL FILOPN
0020  *         COMMON /GLOBAL/ IOP,T0,IFIT,FILOPN
      *   C*
0021  *         DATA IOPARS /0/
          C**

PDP-11 FORTRAN-77 V5.0-0          09:34:02     26-Sep-86          Page 2
FIT.FTN;1                  /F77/OP/TR:BLOCKS/WR

          C*    SQRTPI  = sqrt(pi)
          C*    SRLN2 = 2. * sqrt(ln2)
          C*
0022            DATA SPESPC /30*0./, ENESPC /30*0./, PARSPC /30*0./,
               +     EPSILN /1.D0/, PICON  /3.141592654D0/,
               +     SQRTPI /1.772453851D0/, SRLN2  /1.6651092D0/,
               +     FILOPN /.FALSE./
          C**
          C*    Set up program
          C*
0023            CALL DIINI (6D0)
0024            CALL SETUP
          C*
          C*    Display master menu and get option
          C*
0025       1000 CALL FITMEN
0026            IF (IOP) 9000,1000,1005
          C*
          C*    The big option switch
          C*
0027       1005 GOTO (1000, 1000, 1010, 1015, 1000, 1020, 1000, 1025,
               +      1030, 1000, 1035, 1000, 1000, 1000, 1000, 1000,
               +      1000, 1040), IOP
          C*
          C*    Opt C(3) : Store/Restore spectrum
          C*
0028       1010 CALL FITIO
0029            GOTO 1000
          C*
          C*    Opt D(4) : Display spectrum and fit data
          C*
0030       1015 CALL FITDIS
0031            IFIT = 0
0032            GOTO 1000
          C*
          C*    Opt F(6) : Fit
          C*
0033       1020 CALL FITSET
0034            IF (IFIT .EQ. 0)  GOTO 1000
0035            IF (IERROR .LT. 0)  GOTO 1000
0036            CALL FITFIT
0037            GOTO 1040
          C*
          C*    Opt H(8) : Output results to print file
          C*
0038       1025 CALL FITPRT
0039            GOTO 1000
          C*
          C*    Opt I(9) : Route print file to printer
          C*
0040       1030 CALL FITROU
0041            GOTO 1000
          C*
          C*    Opt K(11) : Save spectrum and fit parameters for plotting
          C*

PDP-11 FORTRAN-77 V5.0-0          09:34:02     26-Sep-86          Page 3
FIT.FTN;1                  /F77/OP/TR:BLOCKS/WR

0042       1035 CALL FITSFP
0043            GOTO 1000
          C*
          C*    Opt R(18) : (Re)display results
          C*
0044       1040 CALL FITOUT (5)
0045            IF (IFIT .EQ. 1)  GOTO 1015
0046            GOTO 1000
          C*
          C*    Terminate program
          C*
0047       9000 CALL FITEND
0048            IF (IOP .NE. 1)  GOTO 1000
0049            STOP
          C*    ****
0050            END
```

```
PDP-11 FORTRAN-77 V5.0-0          09:37:56     26-Sep-86          Page 1
FITBAC.FTN;1              /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE FITBAC (X,IBACK)
                C*  ************************
                C*
                C*  This subroutine calculates at a given -X- the contribution
                C*  of the background
                C*
                C*
                C*  Ver. 1.0  12-OCT-83 (DR)    !!Original version!!
                C*
0002            INCLUDE 'FIT.COM'
0003  *         IMPLICIT REAL*8 (A-H,O-Z)
      *   C*
      *   C*    Logical unit numbers used:
      *   C*
      *   C*    LUNSPE:  Lun to read spectrum
      *   C*    LUNPAR:  Lun for PARFIL
      *   C*    LUNPRT:  Lun for print output
      *   C*    LUNCAL:  Lun for calibration input, and energy table
      *   C*    LUNTRM:  Lun for terminal I/o (incl. DLPACK, MNPACK)
      *   C*
0004  *   C*    PARAMETER LUNSPE=1, LUNCAL=2, LUNPRT=3, LUNPAR=4, LUNTRM = 5
      *   C*
0005  *         PARAMETER MAXCHN=4096, MAXPK=30, MAXCO=6, MAXREG=7, MAXPAR=40,
      *  +                  NPPARM=6, MXCALF=3, MPOINT=500, MXLINK=20
      *   C*
0006  *   C*    BYTE SPESPC(30),PARSPC(30),CALSPC(30),FILPRT(30),ENESPC(30)
      *   C*
0007  *   C*    CHARACTER*20 HDGLIN
      *   C*
0008  *   C*    INTEGER*4 JCOUNT
      *   C*
0009  *   C*    REAL*4 T0,SECNDS
      *   C*
0010  *         COMMON /SPECTR/ JCOUNT(MAXCHN)
0011  *         COMMON /SPDESC/ NXCHAN,NYCHAN,IDATIM(9),NRUN,ISPILL,ITARG,
      *  +                      ITEXT(18),LABEL
0012  *         COMMON /OPPARS/ MDEV,SPESPC,PARSPC,MXITER,LBFIX,INREG,LIXLO,LIXHI,
      *  +                      LIYLO,LIYHI,IYPWR,LIFLO,LIFHI,IPEAK,LBACK,LPEAK,
      *  +                      LPEBA,NCOEFF,COEFF(MAXCO),ITERA,NFREE,
      *  +                      PEAK(NPPARM,MAXPK),IFLAG(NPPARM,MAXPK),
      *  +                      INDEX(NPPARM,MAXPK),ISELEC,IGNOR(MAXPK),NREG,
      *  +                      KREGLO(MAXREG),KREGHI(MAXREG),IMODE,INTCOR,NCALF,
      *  +                      CALFAC(MXCALF),SIGMA(MAXPK),GAMMA(MAXPK),CALSPC,
      *  +                      FILPRT,ENESPC
0013  *         COMMON /OPPART/ IOPARS
0014  *         COMMON /RESULT/ CHINEW,CHIOLD,FVBACK,FVPEAK
0015  *         COMMON /ERRCOM/ IERROR
0016  *         COMMON /PARAM/  ILINK(NPPARM,MXLINK),NPAR,NDIMM,KUSE,KSKIP,
      *  +                      TOLO(6),TOHI(6)
0017  *         COMMON /ARRAYS/ VMAT(MAXPAR*(MAXPAR+1)/2),DERIV(MAXPAR),
      *  +                      RVEC(MAXPAR),HDERIV(MAXPAR)
0018  *         COMMON /CONST/  PICON,EPSILN,SQRTPI,SRLN2
      *   C*
0019  *         LOGICAL FILOPN
0020  *         COMMON /GLOBAL/ IOP,T0,IFIT,FILOPN
```

```
PDP-11 FORTRAN-77 V5.0-0          09:37:56     26-Sep-86          Page 2
FITBAC.FTN;1              /F77/OP/TR:BLOCKS/WR

      *   C*
0021  *         DATA IOPARS /0/
                C*
                C**
                C*    Prepare values
                C*
0022            FVBACK = 0.0D0
                C*
                C*    Calculate contribution of background
                C*
0023            XT = 1.0D0
0024            DO 1000 J=1,NCOEFF,1
0025               FVBACK = FVBACK + (COEFF(J) * XT)
0026               IF (COEFF(J) .NE. 0.D0)  THEN
0027                  IF ((LBFIX+IBACK) .EQ. 0)  DERIV(J) = XT
0028               ENDIF
0029               XT = XT*X
0030         1000 CONTINUE
                C*
                C*    Return to calling program
                C*
0031         9000 RETURN
                C*    ******
0032            END
```

```
PDP-11 FORTRAN-77 V5.0-0          09:38:51     26-Sep-86          Page 1
FITCUR.FTN;1              /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE FITCUR
                C*  ****************
                C*
                C*  This subroutine sets the fit parameters in cursor mode
                C*
0002            INCLUDE 'FIT.COM'
0003  *         IMPLICIT REAL*8 (A-H,O-Z)
      *   C*
      *   C*    Logical unit numbers used:
      *   C*
      *   C*    LUNSPE:  Lun to read spectrum
      *   C*    LUNPAR:  Lun for PARFIL
      *   C*    LUNPRT:  Lun for print output
      *   C*    LUNCAL:  Lun for calibration input, and energy table
      *   C*    LUNTRM:  Lun for terminal I/o (incl. DLPACK, MNPACK)
      *   C*
0004  *   C*    PARAMETER LUNSPE=1, LUNCAL=2, LUNPRT=3, LUNPAR=4, LUNTRM = 5
      *   C*
0005  *         PARAMETER MAXCHN=4096, MAXPK=30, MAXCO=6, MAXREG=7, MAXPAR=40,
      *  +                  NPPARM=6, MXCALF=3, MPOINT=500, MXLINK=20
      *   C*
0006  *   C*    BYTE SPESPC(30),PARSPC(30),CALSPC(30),FILPRT(30),ENESPC(30)
      *   C*
0007  *   C*    CHARACTER*20 HDGLIN
      *   C*
0008  *   C*    INTEGER*4 JCOUNT
      *   C*
0009  *   C*    REAL*4 T0,SECNDS
      *   C*
0010  *         COMMON /SPECTR/ JCOUNT(MAXCHN)
0011  *         COMMON /SPDESC/ NXCHAN,NYCHAN,IDATIM(9),NRUN,ISPILL,ITARG,
      *  +                      ITEXT(18),LABEL
0012  *         COMMON /OPPARS/ MDEV,SPESPC,PARSPC,MXITER,LBFIX,INREG,LIXLO,LIXHI,
      *  +                      LIYLO,LIYHI,IYPWR,LIFLO,LIFHI,IPEAK,LBACK,LPEAK,
      *  +                      LPEBA,NCOEFF,COEFF(MAXCO),ITERA,NFREE,
      *  +                      PEAK(NPPARM,MAXPK),IFLAG(NPPARM,MAXPK),
      *  +                      INDEX(NPPARM,MAXPK),ISELEC,IGNOR(MAXPK),NREG,
      *  +                      KREGLO(MAXREG),KREGHI(MAXREG),IMODE,INTCOR,NCALF,
      *  +                      CALFAC(MXCALF),SIGMA(MAXPK),GAMMA(MAXPK),CALSPC,
      *  +                      FILPRT,ENESPC
0013  *         COMMON /OPPART/ IOPARS
0014  *         COMMON /RESULT/ CHINEW,CHIOLD,FVBACK,FVPEAK
0015  *         COMMON /ERRCOM/ IERROR
0016  *         COMMON /PARAM/  ILINK(NPPARM,MXLINK),NPAR,NDIMM,KUSE,KSKIP,
      *  +                      TOLO(6),TOHI(6)
0017  *         COMMON /ARRAYS/ VMAT(MAXPAR*(MAXPAR+1)/2),DERIV(MAXPAR),
      *  +                      RVEC(MAXPAR),HDERIV(MAXPAR)
0018  *         COMMON /CONST/  PICON,EPSILN,SQRTPI,SRLN2
      *   C*
0019  *         LOGICAL FILOPN
0020  *         COMMON /GLOBAL/ IOP,T0,IFIT,FILOPN
      *   C*
0021  *         DATA IOPARS /0/
                C*
0022            BYTE JCHAR,XCHAR,YCHAR,LCHAR,HCHAR,PCHAR,BLANK
```

```
PDP-11 FORTRAN-77 V5.0-0          09:38:51     26-Sep-86          Page 2
FITCUR.FTN;1              /F77/OP/TR:BLOCKS/WR

                C*
0023            REAL*4 YL,YH
                C*
0024            INTEGER*4 JMIN,JMAX,JSUM,JY
                C*
0025            DATA LCHAR /'L'/, HCHAR /'H'/, PCHAR /'P'/, XCHAR /'X'/,
                +     YCHAR /'Y'/, BLANK /' '/, ICHAR /' '/
                C*
0026            EQUIVALENCE (ICHAR,JCHAR)
                C**
                C*
0027            IREGL = 0
0028            IREGH = 0
0029            IPEAK = 0
0030            IDISX = 0
0031            IDISY = 0
                C*
                C*    If no spectrum present, Go home
                C*
0032            IF (NXCHAN .LE. 0)  THEN
0033               CALL DIPAGE
0034               CALL DLOUT ('0No data to display')
0035               CALL DLKEY
0036               GOTO  9000
0037            ENDIF
```

```
C*
C*     Redisplay spectrum
C*
0038    1000 CALL PREPDI (JCOUNT,LIXLO,LIXHI,LIYLO,LIYHI,IYPWR,JMIN,JMAX,JSUM)
0039         IF (JSUM .LE. 0)  THEN
0040            CALL DIPAGE
0041            CALL DLOUT ('0No counts in spectrum')
0042            CALL DLKEY
0043            GOTO  9000
0044         ENDIF
0045         CALL SDISPL (JCOUNT,LIXLO,LIXHI,JMIN,JMAX,IMODE)
0046         CALL DIHOME
0047         CALL DIANU
0048         WRITE (5,1005)
0049    1005 FORMAT ('+',' Hit    P   to define peak position',10X,
            +        'blank  to finish input'/
            +        ' ',' '          X   new X-display          ',10X,
            +        'Y       new Y-display'/
            +        ' ',' '          L   fit regions (low end)  ',10X,
            +        'H       (high end)')
C*
0050    1010 JCHAR = BLANK
0051         CALL DIPNT
0052         CALL DJCURI (ICHAR,IX,JY)
0053         IF (JCHAR .EQ. BLANK)  GOTO 4000
C*
C*     Get absolute x and y value for MARKER
C*
0054         KX = 105 + IFIX (FLOAT (IX-LIXLO) / FLOAT (LIXHI-LIXLO) * 918.)
0055         KY = 650
```



```
C*
C*     Hit H : Define high end of fit region
C*
0056         IF (JCHAR .EQ. HCHAR)  THEN
0057            IREGH = IREGH + 1
0058            IF (IREGH .NE. IREGL)  GOTO 3000
0059            KREGHI(IREGH) = IX
0060            CALL MARKER (KX,KY,HCHAR)
0061            GOTO 1010
0062         ENDIF
C*
C*     Hit L : Define low end of fit region
C*
0063         IF (JCHAR .EQ. LCHAR)  THEN
0064            IREGL = IREGL + 1
0065            IF (IREGL .GT. MAXREG)  GOTO 3005
0066            KREGLO(IREGL) = IX
0067            CALL MARKER (KX,KY,LCHAR)
0068            GOTO 1010
0069         ENDIF
C*
C*     Hit P : Define peak positions
C*
0070         IF (JCHAR .EQ. PCHAR)  THEN
0071            IPEAK = IPEAK + 1
0072            IF (IPEAK .GT. MAXPK)  GOTO 3010
0073            PEAK(1,IPEAK) = DBLE(IX)
0074            PEAK(3,IPEAK) = JCOUNT(INT(PEAK(1,IPEAK)))
0075            IGNOR(IPEAK) = 0
0076            CALL MARKER (KX,KY,PCHAR)
0077            GOTO 1010
0078         ENDIF
C*
C*     Hit X : Define new X-display
C*
0079         IF (JCHAR .EQ. XCHAR)  THEN
0080            IDISX = IDISX + 1
0081            IF (IDISX .EQ. 1)  THEN
0082               IF (IX .LT. LIXLO)  THEN
0083                  LIXLO = 1
0084               ELSE
0085                  LIXLO = IX
0086               ENDIF
0087               GOTO 1010
0088            ENDIF
0089            IF (IDISX .EQ. 2)  THEN
0090               IF (IX .LT. LIXHI)  THEN
0091                  LIXHI = IX
0092                  LIYHI = LIYLO - 10
0093               ELSE
0094                  LIXHI = NXCHAN
0095               ENDIF
0096               IDISX = 0
0097               GOTO 1000
0098            ENDIF
```



```
0099         ENDIF
C*
C*     Hit Y : Define new Y-display
C*
0100         IF (JCHAR .EQ. YCHAR)  THEN
0101            IYPWR = 0
0102            IDISY = IDISY + 1
0103            IF (IDISY .EQ. 1)  THEN
0104               IF (JY .LT. LIYLO)  THEN
0105                  LIYLO = 0
0106               ELSE
0107                  LIYLO = JY
0108               ENDIF
0109               GOTO 1010
0110            ENDIF
0111            IF (IDISY .EQ. 2)  THEN
0112               IF (JY .LT. LIYHI)  THEN
0113                  LIYHI = JMAX
0114               ELSE
0115                  LIYHI = JY
0116               ENDIF
0117               GOTO 1000
0118            ENDIF
0119         ENDIF
C*
C*     Error message for wrong character
C*
0120         WRITE (5,2000)
0121    2000 FORMAT (' ','Undefined symbol')
0122         GOTO 3100
C*
C*     Error message for fit region failure
C*
0123    3000 CALL DLOUT ('1Low end of this region is missing')
0124         GOTO 3100
C*
C*
C*     Error message for max number of fit regions reached
C*
0125    3005 CALL DLOUT ('1Max nr. of fit regions got')
0126         GOTO 3100
C*
C*
C*     Error message for max peak nr.
C*
0127    3010 CALL DLOUT ('1Max nr. of peaks got')
0128    3100 CALL DLKEY
0129         GOTO 1000
C*
C*     Define other parameters for fit
C*
0130    4000 IF (IREGL .NE. 0)  THEN
0131            NREG = IREGL
0132            INREG = 1
0133         ENDIF
```



```
C*
C*     Return to calling program
C*
0134    9000 RETURN
C*     ******
0135         END
```



```
0001         SUBROUTINE MARKER (IX,IY,CHAR)
C*           ******************************
C*
C*
0002         BYTE CHAR
C*
C**
C*
0003         CALL DIMOVA (IX-15,IY)
0004         CALL DIANU
0005         WRITE (5,1000) CHAR
0006    1000 FORMAT (2X,A1)
0007         CALL DIVEC
0008         CALL DIMOVA (IX,IY-50)
0009         CALL DIDRWA (IX,IY-75)
C*
C*     Return to calling program
C*
0010         RETURN
C*     ******
0011         END
```

```
PDP-11 FORTRAN-77 V5.0-0        09:37:11    26-Sep-86          Page 1
FITDIS FTN,1              /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE FITDIS
                C*  ****************
                C*
                C*  This subroutine displays the spectrum and optionally the
                C*  contribution of the peaks and/or the background
                C*
                C*
                C*  Ver. 1.0/07-Sep-83(TK)  Original version
                C*  Ver. 1.1/14-Dec-83(TK)  Some enhancements
                C*  Ver. 1.2/17-Jan-84(TK)  Some enhancements
                C*  Ver. 1.3/25-Jan-84(TK)  Separated peak display
                C*
0002            INCLUDE 'FIT.COM'
0003            IMPLICIT REAL*8 (A-H,O-Z)
                C*
                C*  Logical unit numbers used:
                C*
                C*  LUNSPE: Lun to read spectrum
                C*  LUNPAR: Lun for PARFIL
                C*  LUNPRT: Lun for print output
                C*  LUNCAL: Lun for calibration input, and energy table
                C*  LUNTRM: Lun for terminal I/O (incl. DLPACK, MNPACK)
                C*
0004            PARAMETER LUNSPE=1, LUNCAL=2, LUNPRT=3, LUNPAR=4, LUNTRM = 5
                C*
0005            PARAMETER MAXCHN=4096, MAXPK=30, MAXCO=6, MAXREG=7, MAXPAR=40,
                +          NPPARM=6, MXCALF=3, MPOINT=500, MXLINK=20
                C*
0006            BYTE SPESPC(30),PARSPC(30),CALSPC(30),FILPRT(30),ENESPC(30)
                C*
0007            CHARACTER*20 HOGLIN
                C*
0008            INTEGER*4 JCOUNT
                C*
0009            REAL*4 TO,SECNDS
                C*
0010            COMMON /SPECTR/ JCOUNT(MAXCHN)
0011            COMMON /SPDESC/ NXCHAN,NYCHAN,IDATIM(9),NRUN,ISPILL,ITARG,
                +               ITEXT(18),LABEL
0012            COMMON /OPPARS/ MDEV,SPESPC,PARSPC,MXITER,LBFIX,INREG,LIXLO,LIXHI
                +               LIYLO,LIYHI,IYPWR,LIFLO,LIFHI,IPEAK,LBACK,LPEAK,
                +               LPEBA,NCOEFF,COEFF(MAXCO),ITERA,NFREE,
                +               PEAK(NPPARM,MAXPK),IFLAG(NPPARM,MAXPK),
                +               INDEX(NPPARM,MAXPK),ISELEC,IGNOR(MAXPK),NREG,
                +               KREGLO(MAXREG),KREGHI(MAXREG),IMODE,INTCOR,NCALF
                +               CALFAC(MXCALF),SIGMA(MAXPK),GAMMA(MAXPK),CALSPC,
                +               FILPRT,ENESPC
0013            COMMON /OPPART/ IOPARS
0014            COMMON /RESULT/ CHINEW,CHIOLD,FVBACK,FVPEAK
0015            COMMON /ERRCOM/ IERROR
0016            COMMON /PARAM/  ILINK(NPPARM,MXLINK),NPAR,NDIMM,KUSE,KSKIP,
                +               TOLO(6),TOHI(6)
0017            COMMON /ARRAYS/ VMAT(MAXPAR*(MAXPAR+1)/2),DERIV(MAXPAR),
                +               RVEC(MAXPAR),HDERIV(MAXPAR)
0018            COMMON /CONST/  PICON,EPSILN,SQRTPI,SRLN2


PDP-11 FORTRAN-77 V5.0-0        09:37:11    26-Sep-86          Page 2
FITDIS FTN,1              /F77/OP/TR:BLOCKS/WR

                C*
0019            LOGICAL FILOPN
0020            COMMON /GLOBAL/ IOP,TO,IFIT,FILOPN
                C*
0021            DATA IOPARS /0/
                C*
0022            LOGICAL NODRAW
                C*
0023            INTEGER*4 JMIN,JMAX,JSUM,JYVAL
                C*
0024            COMMON /BUFMEM/ MBUFF(600)
                C*
                C*  If no spectrum present: Go home
                C*
0025            IF (NXCHAN.LE.0) THEN
0026               CALL MNERAS
0027               CALL DLOUT ('0No data to display')
0028               CALL DLKEY
0029               GOTO 9000
0030            ENDIF
                C*
0031            IF (IFIT .EQ. 1)  GOTO 1010
                C*
                C*  Display sub menu
                C*
0032     1000   CALL MNCLR
0033            CALL MNBUFF (MBUFF,600)
0034            CALL MNHEAD ('Display spectrum, peaks and background')
0035            CALL MNDEC  (1,'Lower X limit',LIXLO,LIXLO,1,MAXCHN)
0036            CALL MNDEC  (2,'Higher X limit',LIXHI,LIXHI,1,MAXCHN)
0037            CALL MNDEC  (4,'Lower Y limit',LIYLO,LIYLO,0)
0038            CALL MNDEC  (5,'Higher Y limit',LIYHI,LIYHI,1)
0039            CALL MNDEC  (6,'Power of 10 for Y-low, Y-high',IYPWR,IYPWR,0,35)
0040            CALL MNONOF (8,'Display of background',LBACK,LBACK)
0041            CALL MNONOF (9,'Display peaks separately',LPEAK,LPEAK)
0042            CALL MNONOF (10,'Display of peak+background',LPEBA,LPEBA)
0043            CALL MNDEC  (14,'Display mode: 0=Histo, 1=Err, 2=Spec, 3=Point',
                +                IMODE,IMODE,0,3)
0044            CALL MNOPT  (18,'Redisplay')
0045            CALL MNDISP
0046            CALL MNIN (IOP)
0047            IF (IOP) 9000,1000,1005
                C*
                C*  The big option switch
                C*
0048     1005   IF (IOP .NE. 18)  GOTO 1000
                C*
                C*  Option R(18): Redisplay spectrum
                C*
0049     1010   LIXHI = MIN (LIXHI,NXCHAN)
0050            IF (LIXLO .GE. LIXHI)  THEN
0051               LIXLO = 1
0052               LIXHI = NXCHAN
0053            ENDIF
0054            CALL PREPDI (JCOUNT,LIXLO,LIXHI,LIYLO,LIYHI,IYPWR,JMIN,JMAX,JSUM)


PDP-11 FORTRAN-77 V5.0-0        09:37:11    26-Sep-86          Page 3
FITDIS.FTN,1              /F77/OP/TR:BLOCKS/WR

0055            IF (JSUM .LE. 0) THEN
0056               CALL MNERAS
0057               CALL DLOUT ('0No counts in spectrum')
0058               CALL DLKEY
0059               GOTO 1000
0060            ENDIF
0061            CALL SDISPL (JCOUNT,LIXLO,LIXHI,JMIN,JMAX,IMODE)
                C*
                C*  Display fit data (if requested)
                C*
0062            IF ((LPEAK + LBACK + LPEBA) .NE. 0)  THEN
                C*
                C*     Find limits for display
                C*
0063               LIFLO = KREGLO(1)
0064               LIFHI = KREGHI(1)
0065               DO 1020 J=2,NREG,1
0066                  LIFLO = MIN (LIFLO,KREGLO(J))
0067                  LIFHI = MAX (LIFHI,KREGHI(J))
0068    1020       CONTINUE
0069               LIFLO = MAX (LIFLO,LIXLO)
0070               LIFHI = MIN (LIFHI,LIXHI)
                C*
                C*     Prepare new scalings for background and fit
                C*
0071               IF (LIFLO .LT. LIFHI)  THEN
0072                  IDIFF = LIXHI - LIXLO + 1
0073                  RDIV = DFLOAT(MPOINT) / IDIFF
0074                  IPLO = IFIX (SNGL(RDIV) * LIXLO + 0.5)
0075                  IPHI = IFIX (SNGL(RDIV) * LIXHI + 0.5)
0076                  CALL DJWNOW (IPLO,IPHI,JMIN,JMAX)
0077                  IPLO = IFIX (SNGL(RDIV) * LIFLO + 0.5)
0078                  IPHI = IFIX (SNGL(RDIV) * LIFHI + 0.5)
                C*
                C*     Display background (if requested)
                C*
0079                  IF (LBACK.NE.0) THEN
0080                     NODRAW = .TRUE.
0081                     DO 1025 J=IPLO,IPHI,1
0082                        CALL FITBAC (DFLOAT(J)/RDIV,1)
0083                        JYVAL = JIDINT(FVBACK+0.5)
0084                        JYVAL = MAX(JYVAL,JMIN)
0085                        JYVAL = MIN(JYVAL,JMAX)
0086                        IF (NODRAW) THEN
0087                           CALL DJMOVA (J,JYVAL)
0088                           NODRAW = .FALSE.
0089                        ELSE
0090                           CALL DJDRWA (J,JYVAL)
0091                        ENDIF
0092    1025                CONTINUE
0093                  ENDIF
```

```
C=
C=              Display peak(s) (if requested)
C=
0094                    IF (LPEAK.NE.0) THEN


PDP-11 FORTRAN-77 V5.0-0        09:37:11    26-Sep-86           Page 4
FITDIS.FTN;1              /F77/OP/TR:BLOCKS/WR

0095                        DO 1035 I=1,MAXPK,1
0096                           IF (IGNOR(I) .EQ. 1)  GOTO 1035
0097                           NODRAW = .TRUE.
0098                           DO 1030 J=IPLO,IPHI,1
0099                              CALL FITPEA (DFLOAT(J)/RDIV,-I)
0100                              IF (FVPEAK.GT.0.400) THEN
0101                                 CALL FITBAC (DFLOAT(J)/RDIV,1)
0102                                 FVBPK = FVBACK
0103                                 JYVAL = JIDINT(FVBPK+FVPEAK+0.5)
0104                                 JYVAL = MAX(JYVAL,JMIN)
0105                                 JYVAL = MIN(JYVAL,JMAX)
0106                                 IF (NODRAW) THEN
0107                                    CALL DJMOVA (J,JYVAL)
0108                                    NODRAW = .FALSE.
0109                                 ELSE
0110                                    CALL DJDRWA (J,JYVAL)
0111                                 ENDIF
0112                              ENDIF
0113     1030                     CONTINUE
0114     1035                  CONTINUE
0115                        ENDIF
C=
C=              Display peak(s) and background (if requested)
C=
0116                        IF (LPEBA .NE. 0)  THEN
0117                           NODRAW = .TRUE.
0118                           DO 1040 J=IPLO,IPHI,1
0119                              CALL FITBAC (DFLOAT(J)/RDIV,1)
0120                              FVBPK = FVBACK
0121                              CALL FITPEA (DFLOAT(J)/RDIV,1)
0122                              JYVAL = JIDINT(FVBPK+FVPEAK+0.5)
0123                              JYVAL = MAX(JYVAL,JMIN)
0124                              JYVAL = MIN(JYVAL,JMAX)
0125                              IF (NODRAW) THEN
0126                                 CALL DJMOVA (J,JYVAL)
0127                                 NODRAW = .FALSE.
0128                              ELSE
0129                                 CALL DJDRWA (J,JYVAL)
0130                              ENDIF
0131     1040                  CONTINUE
0132                        ENDIF
0133                     ENDIF
0134                  ENDIF
C=
C=     Put some headerlines onto the pictures
C=
0135              CALL PRTHDR (JMAX,JSUM)
0136              READ (5,1045) J
0137      1045 FORMAT (A1)
0138              IF (IFIT .EQ. 0)  GOTO 1000
C=
C=     Return to calling program
C=
0139      9000 RETURN
C=     ******


PDP-11 FORTRAN-77 V5.0-0        09:37:11    26-Sep-86           Page 5
FITDIS.FTN;1              /F77/OP/TR:BLOCKS/WR

0140              END
```

```
PDP-11 FORTRAN-77 V5.0-0        09:42:10    26-Sep-86           Page 1
FITEND.FTN;1              /F77/OP/TR:BLOCKS/WR

0001              SUBROUTINE FITEND
C=                **********
C=
C=     This subroutine does some cleanup at the end of
C=     fit program
C=
C=
C=     Ver. 1.0/22-Nov-83(TK)  Original version
C=
0002              INCLUDE 'FIT.COM'
0003  *           IMPLICIT REAL*8 (A-H,O-Z)
      *  C=
      *  C=     Logical unit numbers used:
      *  C=
      *  C=     LUNSPE: Lun to read spectrum
      *  C=     LUNPAR: Lun for PARFIL
      *  C=     LUNPRT: Lun for print output
      *  C=     LUNCAL: Lun for calibration input, and energy table
      *  C=     LUNTRM: Lun for terminal I/o (incl. DLPACK, MNPACK)
      *  C=
0004  *           PARAMETER LUNSPE=1, LUNCAL=2, LUNPRT=3, LUNPAR=4, LUNTRM = 5
      *  C=
0005  *           PARAMETER MAXCHN=4096, MAXPK=30, MAXCO=6, MAXREG=7, MAXPAR=40,
      *  +                  NPPARM=6, MXCALF=3, MPOINT=500, MXLINK=20
      *  C=
0006  *           BYTE SPESPC(30),PARSPC(30),CALSPC(30),FILPRT(30),ENESPC(30)
      *  C=
0007  *           CHARACTER*20 HDGLIN
      *  C=
0008  *           INTEGER*4 JCOUNT
      *  C=
0009  *           REAL*4 T0,SECNDS
      *  C=
0010  *           COMMON /SPECTR/ JCOUNT(MAXCHN)
0011  *           COMMON /SPDESC/ NXCHAN,NYCHAN,IDATIM(9),NRUN,ISPILL,ITARG,
      *  +                        ITEXT(18),LABEL
0012  *           COMMON /OPPARS/ MDEV,SPESPC,PARSPC,MXITER,LBFIX,INREG,LIXLO,LIXHI,
      *  +                        LIYLO,LIYHI,IYPWR,LIFLO,LIFHI,IPEAK,LBACK,LPEAK,
      *  +                        LPEBA,NCOEFF,COEFF(MAXCO),ITERA,NFREE,
      *  +                        PEAK(NPPARM,MAXPK),IFLAG(NPPARM,MAXPK),
      *  +                        INDEX(NPPARM,MAXPK),ISELEC,IGNOR(MAXPK),NREG,
      *  +                        KREGLO(MAXREG),KREGHI(MAXREG),IMODE,INTCOR,NCALF,
      *  +                        CALFAC(MXCALF),SIGMA(MAXPK),GAMMA(MAXPK),CALSPC,
      *  +                        FILPRT,ENESPC
0013  *           COMMON /OPPART/ IOPARS
0014  *           COMMON /RESULT/ CHINEW,CHIOLD,FVBACK,FVPEAK
0015  *           COMMON /LRRCOM/ IERROR
0016  *           COMMON /PARAM/  ILINK(NPPARM,MXLINK),NPAR,NDIMM,KUSE,KSKIP,
      *  +                        TOLO(6),TOHI(6)
0017  *           COMMON /ARRAYS/ VMAT(MAXPAR*(MAXPAR+1)/2),DERIV(MAXPAR),
      *  +                        RVEC(MAXPAR),HDERIV(MAXPAR)
0018  *           COMMON /CONST/  PICON,EPSILN,SQRTPI,SRLN2
      *  C=
0019  *           LOGICAL FILOPN
0020  *           COMMON /GLOBAL/ IOP,T0,IFIT,FILOPN


PDP-11 FORTRAN-77 V5.0-0        09:42:10    26-Sep-86           Page 2
FITEND.FTN;1              /F77/OP/TR:BLOCKS/WR

      *  C=
0021  *           DATA IOPARS /0/
      *  C=
0022              CALL PARWT
0023              CALL MNCRAS
0024              CALL DLYENQ ('Do you really want to stop',0,IOP)
C=
C=     Return to calling program
C=
0025              RETURN
C=     ******
0026              END
```

```
PDP-11 FORTRAN-77 V5.0-0          09:37:00     26-Sep-86          Page 1
FITENE.FTN;1                /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE FITENE
                C*  ***************
                C*
                C*  This subroutine handles the option -M- of program FITIO
                C*
                C*
                C*
                C*
                C**
0002            PARAMETER MAXNUC=10
                C*
0003            INCLUDE 'FIT.COM'
0004 m          IMPLICIT REAL*8 (A-H,O-Z)
     m  C*
     m  C*      Logical unit numbers used:
     m  C*
     m  C*      LUNSPE:  Lun to read spectrum
     m  C*      LUNPAR:  Lun for PARFIL
     m  C*      LUNPRT:  Lun for print output
     m  C*      LUNCAL:  Lun for calibration input, and energy table
     m  C*      LUNTRM:  Lun for terminal I/o (incl. DLPACK, MNPACK)
     m  C*
0005 m          PARAMETER LUNSPE=1, LUNCAL=2, LUNPRT=3, LUNPAR=4, LUNTRM = 5
     m  C*
0006 m          PARAMETER MAXCHN=4096, MAXPK=30, MAXCO=6, MAXREG=7, MAXPAR=40,
     m         +          NPPARM=6, MXCALF=3, MPOINT=500, MXLINK=20
     m  C*
0007 m          BYTE SPESPC(30),PARSPC(30),CALSPC(30),FILPRT(30),ENESPC(30)
     m  C*
0008 m          CHARACTER*20 HDGLIN
     m  C*
0009 m          INTEGER*4 JCOUNT
     m  C*
0010 m          REAL*4 T0,SECNDS
     m  C*
0011 m          COMMON /SPECTR/ JCOUNT(MAXCHN)
0012 m          COMMON /SPDESC/ NXCHAN,NYCHAN,IDATIM(9),NRUN,ISPILL,ITARG,
     m         +                ITEXT(18),LABEL
0013 m          COMMON /OPPARS/ MDEV,SPESPC,PARSPC,MXITER,LBFIX,INREG,LIXLO,LIXHI,
     m         +                LIYLO,LIYHI,IYPWR,LIFLO,LIFHI,IPEAK,LBACK,LPEAK,
     m         +                LPEBA,NCOEFF,COEFF(MAXCO),ITERA,NFREE,
     m         +                PEAK(NPPARM,MAXPK),IFLAG(NPPARM,MAXPK),
     m         +                INDEX(NPPARM,MAXPK),ISELEC,IGNOR(MAXPK),NREG,
     m         +                KREGLO(MAXREG),KREGHI(MAXREG),IMODE,INTCOR,NCALF,
     m         +                CALFAC(MXCALF),SIGMA(MAXPK),GAMMA(MAXPK),CALSPC,
     m         +                FILPRT,ENESPC
0014 m          COMMON /OPPART/ IOPARS
0015 m          COMMON /RESULT/ CHINEW,CHIOLD,FVBACK,FVPEAK
0016 m          COMMON /ERRCOM/ IERROR
0017 m          COMMON /PARAM/  ILINK(NPPARM,MXLINK),NPAR,NDIMM,KUSE,KSKIP,
     m         +                TOLO(6),TOHI(6)
0018 m          COMMON /ARRAYS/ VMAT(MAXPAR*(MAXPAR+1)/2),DERIV(MAXPAR),
     m         +                RVEC(MAXPAR),HDERIV(MAXPAR)
0019 m          COMMON /CONST/  PICON,EPSILN,SQRTPI,SRLN2
     m  C*
0020 m          LOGICAL FILOPN

PDP-11 FORTRAN-77 V5.0-0          09:37:00     26-Sep-86          Page 2
FITENE.FTN;1                /F77/OP/TR:BLOCKS/WR

0021 m          COMMON /GLOBAL/ IOP,T0,IFIT,FILOPN
     m  C*
0022 m          DATA IOPARS /0/
                C*
0023            CHARACTER*8 TRANS
0024            CHARACTER*2 NUCLUS,NUCL(MAXNUC)
                C*
0025            REAL*8 ESPLIT(3)
                C**
0026            CALL ASSIGN (LUNCAL,ENESPC,LEN)
0027            NPEAK = 0
0028     1000   READ (LUNCAL,*,END=1010)  NUCLUS,IAA,TRANS,WTENER,ESPLIT
0029            IF (ESPLIT(1) .EQ. 0.D0)  THEN
0030                NPEAK = NPEAK + 1
0031                PEAK(1,NPEAK) = (WTENER - CALFAC(1)) / CALFAC(2)
0032            ELSE
0033                DO 1005 I=1,3,1
0034                    NPEAK = NPEAK + 1
0035                    IF (IGNOR(NPEAK) .EQ. 1)  GOTO 1005
0036                    PEAK(1,NPEAK) = (ESPLIT(I) - CALFAC(1)) / CALFAC(2)
0037     1005       CONTINUE
0038            ENDIF
0039            GOTO 1000
0040     1010   CLOSE (UNIT=LUNCAL)
                C*
                C*  Return to calling program
                C*
0041     9000   RETURN
                C*  ******
0042            END
```

```
PDP-11 FORTRAN-77 V5.0-0          09:40:41     26-Sep-86          Page 1
FITFIT.FTN;1                /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE FITFIT
                C*  ***************
                C*
                C*  This routine fits a spectrum
                C*
                C*
                C*  Ver.  0.0/09-SEP-83(TK)  Original version
                C*  Ver.  1.0/06-OCT-83(DR)
                C*  Ver.  2.0/31-OCT-83(DR)
                C*  Ver.  2.1/03-Nov-83(TK)  Beep implemented
                C*
0002            INCLUDE 'FIT.COM'
0003 m          IMPLICIT REAL*8 (A-H,O-Z)
     m  C*
     m  C*      Logical unit numbers used:
     m  C*
     m  C*      LUNSPE:  Lun to read spectrum
     m  C*      LUNPAR:  Lun for PARFIL
     m  C*      LUNPRT:  Lun for print output
     m  C*      LUNCAL:  Lun for calibration input, and energy table
     m  C*      LUNTRM:  Lun for terminal I/o (incl. DLPACK, MNPACK)
     m  C*
0004 m          PARAMETER LUNSPE=1, LUNCAL=2, LUNPRT=3, LUNPAR=4, LUNTRM = 5
     m  C*
0005 m          PARAMETER MAXCHN=4096, MAXPK=30, MAXCO=6, MAXREG=7, MAXPAR=40,
     m         +          NPPARM=6, MXCALF=3, MPOINT=500, MXLINK=20
     m  C*
0006 m          BYTE SPESPC(30),PARSPC(30),CALSPC(30),FILPRT(30),ENESPC(30)
     m  C*
0007 m          CHARACTER*20 HDGLIN
     m  C*
0008 m          INTEGER*4 JCOUNT
     m  C*
0009 m          REAL*4 T0,SECNDS
     m  C*
0010 m          COMMON /SPECTR/ JCOUNT(MAXCHN)
0011 m          COMMON /SPDESC/ NXCHAN,NYCHAN,IDATIM(9),NRUN,ISPILL,ITARG,
     m         +                ITEXT(18),LABEL
0012 m          COMMON /OPPARS/ MDEV,SPESPC,PARSPC,MXITER,LBFIX,INREG,LIXLO,LIXHI,
     m         +                LIYLO,LIYHI,IYPWR,LIFLO,LIFHI,IPEAK,LBACK,LPEAK,
     m         +                LPEBA,NCOEFF,COEFF(MAXCO),ITERA,NFREE,
     m         +                PEAK(NPPARM,MAXPK),IFLAG(NPPARM,MAXPK),
     m         +                INDEX(NPPARM,MAXPK),ISELEC,IGNOR(MAXPK),NREG,
     m         +                KREGLO(MAXREG),KREGHI(MAXREG),IMODE,INTCOR,NCALF,
     m         +                CALFAC(MXCALF),SIGMA(MAXPK),GAMMA(MAXPK),CALSPC,
     m         +                FILPRT,ENESPC
0013 m          COMMON /OPPART/ IOPARS
0014 m          COMMON /RESULT/ CHINEW,CHIOLD,FVBACK,FVPEAK
0015 m          COMMON /ERRCOM/ IERROR
0016 m          COMMON /PARAM/  ILINK(NPPARM,MXLINK),NPAR,NDIMM,KUSE,KSKIP,
     m         +                TOLO(6),TOHI(6)
0017 m          COMMON /ARRAYS/ VMAT(MAXPAR*(MAXPAR+1)/2),DERIV(MAXPAR),
     m         +                RVEC(MAXPAR),HDERIV(MAXPAR)
0018 m          COMMON /CONST/  PICON,EPSILN,SQRTPI,SRLN2
     m  C*

PDP-11 FORTRAN-77 V5.0-0          09:40:41     26-Sep-86          Page 2
FITFIT.FTN;1                /F77/OP/TR:BLOCKS/WR

0019 m          LOGICAL FILOPN
0020 m          COMMON /GLOBAL/ IOP,T0,IFIT,FILOPN
     m  C*
0021 m          DATA IOPARS /0/
                C*
0022            LOGICAL NOUPDT
                C**
0023            IERROR = 0
0024            CHINEW = 0.0D0
0025            NOUPDT = .TRUE.
                C*
                C*  Set widths to formula-parameters
                C*
0026            DO 1000 IP=1,MAXPK,1
0027               SIGMA(IP) = PEAK(2,IP) / SRLN2
0028               GAMMA(IP) = PEAK(6,IP) / 2.0D0
0029     1000   CONTINUE
                C*
                C*  Loop up to label 1060 dowt the iterations
                C*
                C*
0030            DO 1060 ITERA=0,MXITER,1
                C*
0031               CALL PARDEF
0032               CALL PARWT
0033               CALL PARDEF (VMAT,(MAXPAR*(MAXPAR+1)/2)*8)
0034               CALL PARWT ('IMP')
0035               CALL RLGUES (RAD,%DESC('FITSPY'),,LREG)
```

```
C*
C*      Set up for next interation. CHIOLD holds old CHINEW
C*
0036            CHIOLD = CHINEW
0037            CHINEW = 0.0D0
0038            IF (NOUPDT)  GOTO 1051
C*
C*      Update background coefficients
C*
0039            IF (LBFIX.LE.0) THEN
0040               DO 1035 J=1,NCOEFF,1
0041                  COEFF(J) = COEFF(J) + HDERIV(J)
0042  1035         CONTINUE
0043            ENDIF
C*
C*      Update peak parameters
C*
0044            DO 1050 IP=1,MAXPK,1
0045               IF (IGNOR(IP) .EQ. 1)  GOTO 1050
0046               DO 1045 IV=1,NPPARM,1
0047                  I = INDEX(IV,IP)
0048                  IF (I .LE. 0)  GOTO 1045
C*
C*                   Special treatment for linked peak height parameters
C*
0049                  IF (IV .EQ. 3)  THEN
0050                     IF (IFLAG(3,IP) .GT. 0)  THEN
0051                        XX = PEAK(3,IP) * (1.0D0 + HDERIV(I))


PDP-11 FORTRAN-77 V5.0-0        09:40:41    26-Sep-86            Page 3
FITFIT.FTN;1                /F77/OP/TR:BLOCKS/WR

0052                        ELSE
0053                           XX = PEAK(3,IP) + HDERIV(I)
0054                        ENDIF
0055                     ELSE
0056                        XX = PEAK(IV,IP) + HDERIV(I)
0057                        IF (IV .EQ. 2)  XX = SRLN2 * (SIGMA(IP) + HDERIV(I))
0058                        IF (IV .EQ. 6)  XX = 2.0D0 * (GAMMA(IP) + HDERIV(I))
0059                     ENDIF
C*
0060                     IF (XX .LE. TOLO(IV)  .OR.  XX .GE. TOHI(IV))  THEN
0061                        WRITE (5,1040) IV,IP
0062  1040                  FORMAT (' Parm.',I2,' of Peak',I3,' out')
0063                     ELSE
0064                        PEAK(IV,IP) = XX
0065                     ENDIF
0066  1045            CONTINUE
0067               SIGMA(IP) = PEAK(2,IP) / SRLN2
0068               GAMMA(IP) = PEAK(6,IP) / 2.D0
0069  1050       CONTINUE
C*
C*      Reset temporary variables
C*
0070  1051       NOUPDT = .FALSE.
0071            CALL VDZERO (RVEC,MAXPAR)
0072            CALL VDZERO (VMAT,(MAXPAR*(MAXPAR+1)/2))
C*
C*      Run through the regions
C*
0073            DO 1020 LREG=1,NREG,1
C*
C*         Run through the channels
C*
0074               DO 1015 ICHAN=KREGLO(LREG),KREGHI(LREG),1
0075                  YFIT = 0.0D0
0076                  CALL VDZERO (DERIV,MAXPAR)
0077                  COUNT = JCOUNT(ICHAN)
0078                  IF (COUNT.LT.1.0D0) COUNT = 0.45D0
C*
C*                   Calculate contribution of background
C*
0079                  CALL FITBAC (DFLOAT(ICHAN),0)
0080                  YFIT = YFIT + FVBACK
C*
C*                   Calculate contribution of peaks
C*
0081                  CALL FITPEA (DFLOAT(ICHAN),0)
0082                  YFIT = YFIT + FVPEAK
C*
C*                   The 'fit value' YFIT is now computed  Update CHINEW
C*
0083                  YFIT = COUNT - YFIT
0084                  IF (DABS(CHINEW) .LT. 1.0015)
     +                   CHINEW = CHINEW + (YFIT * YFIT / COUNT)
C*
C*                   Calculate:
```

```
PDP-11 FORTRAN-77 V5.0-0        09:40:41    26-Sep-86            Page 4
FITFIT.FTN;1                /F77/OP/TR:BLOCKS/WR

C*                   T: packed upper triangular normal matrix
C*                   R: right side of normal equation
C*
0085                 IJ = 0
0086                 DO 1010 I=1,NPAR,1
0087                    XX = DERIV(I)/COUNT
0088                    RVEC(I) = RVEC(I) + XX * YFIT
0089                    DO 1005 J=I,NPAR,1
0090                       IJ = IJ+1
0091                       VMAT(IJ) = VMAT(IJ) + (DERIV(J) * XX)
0092  1005              CONTINUE
0093  1010           CONTINUE
C*
0094  1015        CONTINUE
0095  1020     CONTINUE
C*
C*      Chi**2 is calculated  Print message.
C*
0096            WRITE (5,1055) ITERA,CHINEW/DBLE(NFREE),SECNDS(T0)
0097  1055      FORMAT (' ',I2,'. Iteration done. CHI**2/NFREE ',F12.2,
     +                  3X,'(',F6.1,' secs fitting)')
C*
C*      If desired accuracy reached, terminate iteration
C*
0098            IF (ABS(CHINEW-CHIOLD) .LT. EPSILN)  GOTO 1065
C*
C*      We ran through all channels. Do a matrix inversion
C*
0099            CALL SPXINV (VMAT,NPAR)
0100            IF (IERROR .LT. 0)  THEN
0101               CALL DLOUT (' ** Bad matrix **')
0102               GOTO 1065
0103            ENDIF
C*
C*      Calculate parameter increments
C*
0104            CALL VDZERO (HDERIV,MAXPAR)
0105            IJ = 0
0106            DO 1030 I=1,NPAR,1
0107               DO 1025 J=I,NPAR,1
0108                  IJ = IJ + 1
0109                  HDERIV(I) = HDERIV(I) + VMAT(IJ) * RVEC(J)
0110                  IF (I .NE. J) HDERIV(J) = HDERIV(J) + VMAT(IJ) * RVEC(I)
0111  1025            CONTINUE
0112  1030        CONTINUE
0113  1060  CONTINUE
C*
C*      Max. iteration count exhausted
C*
0114            ITERA = ITERA - 1
0115  1065  CALL BEEP
0116        CALL PARRD ('TMP')
0117        CALL PARDEF
C*


PDP-11 FORTRAN-77 V5.0-0        09:40:41    26-Sep-86            Page 5
FITFIT.FTN;1                /F77/OP/TR:BLOCKS/WR

C*      Set flags for display
C*
0118        LBACK = 1
0119        LPEGA = 1
0120        LPEAK = 0
C*
C*      Wait until user looked at the data
C*
0121        CALL DLKEY
C*
C*      Return to calling program
C*
0122        RETURN
C*      ******
0123        END
```

```
PDP-11 FORTRAN-77 V5.0-0          09:36:18     26-Sep-86          Page 1
FITIO.FTN;1                /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE FITIO
                C*************
                C*
                C*      This subroutine handles the option -C- of program FIT
                C*
                C*
                C*      Ver. 1.0  29-Sep-83  (TK)  Original version
                C*      Ver. 1.1  14-Dec-83  (TK)  Clear entire spectrum before reading new one
                C*      Ver. 2.0  01-Feb-84  (TK)  Implementation of calibration polynomial
                C*      Ver. 3.0  28-May-85  (DR)  Bug in calibration-read removed
                C*      Ver. 4.0  21-Jun-85  (DR)  Implementation of energy table
                C*
                C**
0002            INCLUDE 'FIT.COM'
0003 *          IMPLICIT REAL*8 (A-H,O-Z)
     *  C*
     *  C*      Logical unit numbers used:
     *  C*
     *  C*      LUNSPE: Lun to read spectrum
     *  C*      LUNPAR: Lun for PARFIL
     *  C*      LUNPRT: Lun for print output
     *  C*      LUNCAL: Lun for calibration input, and energy table
     *  C*      LUNTRM: Lun for terminal I/o (incl. DLPACK, MNPACK)
     *  C*
0004 *          PARAMETER LUNSPE=1, LUNCAL=2, LUNPRT=3, LUNPAR=4, LUNTRM = 5
     *  C*
0005 *          PARAMETER MAXCHN=4096, MAXPK=30, MAXCO=6, MAXREG=7, MAXPAR=40,
     *         +          NPPARM=6, MXCALF=3, MPOINT=500, MXLINK=20
     *  C*
0006 *          BYTE SPESPC(30),PARSPC(30),CALSPC(30),FILPRT(30),ENESPC(30)
     *  C*
0007 *          CHARACTER*20 HDGLIN
     *  C*
0008 *          INTEGER*4 JCOUNT
     *  C*
0009 *          REAL*4 T0,SECNDS
     *  C*
0010 *          COMMON /SPECTR/ JCOUNT(MAXCHN)
0011 *          COMMON /SPDESC/ NXCHAN,NYCHAN,IDATIM(9),NRUN,ISPILL,ITARG,
     *         +                ITEXT(18),LABEL
0012 *          COMMON /OPPARS/ MDEV,SPESPC,PARSPC,MXITER,LBFIX,INREG,LIXLO,LIXHI,
     *         +                LIYLO,LIYHI,IYPWR,LIFLO,LIFHI,IPEAK,LBACK,LPEAK,
     *         +                LPEBA,NCOEFF,COEFF(MAXCO),ITERA,NFREE,
     *         +                PEAK(NPPARM,MAXPK),IFLAG(NPPARM,MAXPK),
     *         +                INDEX(NPPARM,MAXPK),ISELEC,IGNOR(MAXPK),NREG,
     *         +                KREGLO(MAXREG),KREGHI(MAXREG),IMODE,INTCOR,NCALF,
     *         +                CALFAC(MXCALF),SIGMA(MAXPK),GAMMA(MAXPK),CALSPC,
     *         +                FILPRT,ENESPC
0013 *          COMMON /OPPART/ IOPARS
0014 *          COMMON /RESULT/ CHINEW,CHIOLD,FVBACK,FVPEAK
0015 *          COMMON /ERRCOM/ IERROR
0016 *          COMMON /PARAM/  ILINK(NPPARM,MXLINK),NPAR,NDIMM,KUSE,KSKIP,
     *         +                TOLO(6),TOHI(6)
0017 *          COMMON /ARRAYS/ VMAT(MAXPAR*(MAXPAR+1)/2),DERIV(MAXPAR),
     *         +                RVEC(MAXPAR),HDERIV(MAXPAR)

PDP-11 FORTRAN-77 V5.0-0          09:36:18     26-Sep-86          Page 2
FITIO.FTN;1                /F77/OP/TR:BLOCKS/WR

0018 *          COMMON /CONST/  PICON,EPSILN,SQRTPI,SRLN2
     *  C*
0019 *          LOGICAL FILOPN
0020 *          COMMON /GLOBAL/ IOP,T0,IFIT,FILOPN
     *  C*
0021 *          DATA IOPARS /0/
                C*
0022            COMMON /BUFMEN/ MBUFF(600)
                C*
                C*      Display sub menu
                C*
0023       1000 CALL MNCLR
0024            CALL MNBUFF (MBUFF,600)
0025            CALL MNHEAD ('Get spectrum / save fit parameters')
0026            CALL MNTXT (1,'Spectrum file',SPESPC,SPESPC,29)
0027            CALL MNTXT (2,'Fit parameter file',PARSPC,PARSPC,29)
0028            CALL MNOPT (4,'Save fit parameters')
0029            CALL MNOPT (5,'Read fit parameters')
0030            CALL MNDEC (7,'Get spectrum from disk',ISELEC,ISELEC)
0031            CALL MNOPT (9,'Show spectra available in spectrum file')
0032            CALL MNTXT (11,'Enter calibration polynomial',CALSPC,CALSPC,29)
0033            CALL MNTXT (13,'Enter energy table',ENESPC,ENESPC,29)
0034            CALL MNDISP
0035            CALL MNIN (IOP)
0036            IF (IOP) 9000,1000,1005
                C*
                C*      The big option switch
                C*
0037       1005 GOTO (1000, 1000, 1000, 1010, 1010, 1000, 1020, 1000,
                +       1040, 1010, 1065, 1000, 1080), IOP
                C*
                C*      Option D(4): Save fit parameters
                C*      Option E(5): Read fit parameters
                C*
0038       1010 CALL FITPFL
0039            GOTO 1000
                C*
                C*      Option G(7): Get spectrum from disk
                C*
0040       1020 CALL FITSPR
0041            GOTO 1000
                C*
                C*      Option I(9): Show spectra available
                C*
0042       1040 CALL MNERAS
0043            NXCHAN = 0
0044            CALL ASSIGN (LUNSPE,SPESPC)
0045       1045 DO 1055 J=1,6,1
0046            READ (LUNSPE,ERR=1060,END=1060) NXCHAN,NYCHAN,IDATIM,NRUN,
     *         +                                  ISPILL,ITARG,ITEXT,LABEL
0047            WRITE (5,1035) LABEL,IDATIM,NRUN,ISPILL,ITARG,ITEXT,NXCHAN
0048       1035 FORMAT ('0','Spectrum with label',I6/
     *         +       ' ',9A2,' Run',I4,', Spill',I4,', Target',I3/
     *         +       ' ',18A2,3x,'No. of channels:',I5)
0049            DO 1050 K=1,NYCHAN,1

PDP-11 FORTRAN-77 V5.0-0          09:36:18     26-Sep-86          Page 3
FITIO.FTN;1                /F77/OP/TR:BLOCKS/WR

0050            READ (LUNSPE,ERR=1060,END=1060) (JCOUNT(I),I=1,NXCHAN)
0051       1050 CONTINUE
0052       1055 CONTINUE
0053            CALL DLKEY
0054            CALL MNERAS
0055            GOTO 1045
                C*
0056       1060 CLOSE (UNIT=LUNSPE)
0057            CALL DLKEY
0058            GOTO 1000
                C*
                C*      Option K(11): Input of calibration polynomial
                C*
0059       1065 CALL MNERAS
0060            CALL ASSIGN (LUNCAL,CALSPC)
0061       1070 READ (LUNCAL,*,END=2005)  ID,LIMIT,HDGLIN,CALFAC
0062            IF (ID .NE. ISELEC)  GOTO 1070
0063            NCALF = 3
0064            CALL DLOUT (' ')
0065            WRITE (5,1075)  CALFAC
0066       1075 FORMAT (' Calibration polynomial :'/'  a0 = ',E15.7/
     *         +       '  a1 = ',E15.7/'  a2 = ',E15.7)
0067            CALL DLKEY
0068            CLOSE (UNIT=LUNCAL)
0069            GOTO 1000
                C*
                C*      Option M(13): Input of energy table
                C*
0070       1080 CALL MNERAS
0071            CALL FITENE
0072            GOTO 1000
                C*
                C*      Disk i/o error
                C*
0073       2000 CALL DLOUT ('0No such spectrum')
0074            NXCHAN = 0
0075            CLOSE (UNIT=LUNSPE)
0076            CALL DLKEY
0077            GOTO 1000
                C*
0078       2005 CALL DLOUT ('0No such calibration')
0079            CALL DLKEY
0080            GOTO 1000
                C*
                C*      Return to calling program
                C*
0081       9000 RETURN
                C*      ******
0082            END
```

```
PDP-11 FORTRAN-77 V5.0-0        09:34:28    26-Sep-86          Page 1
FITMEN.FTN,1                /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE FITMEN
                C*  *****************
                C*
                C*  This subroutine displays the master menu and
                C*  transmits the option number to the calling program
                C*
                C*  VER.  0.0/07-SEP-83(TK)  Original version
                C*
0002            INCLUDE 'FIT.COM'
0003 *          IMPLICIT REAL*8 (A-H,O-Z)
     *  C*
     *  C*  Logical unit numbers used:
     *  C*
     *  C*  LUNSPE: Lun to read spectrum
     *  C*  LUNPAR: Lun for PARFIL
     *  C*  LUNPRT: Lun for print output
     *  C*  LUNCAL: Lun for calibration input, and energy table
     *  C*  LUNTRM: Lun for terminal I/o (incl. DLPACK, MNPACK)
     *  C*
0004 *          PARAMETER LUNSPE=1, LUNCAL=2, LUNPRT=3, LUNPAR=4, LUNTRM = 5
     *  C*
0005 *          PARAMETER MAXCHN=4096, MAXPK=30, MAXCO=6, MAXREG=7, MAXPAR=40,
     *         +          NPPARM=6, MXCALF=3, MPOINT=500, MXLINK=20
     *  C*
0006 *          BYTE SPESPC(30),PARSPC(30),CALSPC(30),FILPRT(30),ENESPC(30)
     *  C*
0007 *          CHARACTER*20 HDGLIN
     *  C*
0008 *          INTEGER*4 JCOUNT
     *  C*
0009 *          REAL*4 T0,SECNDS
     *  C*
0010 *          COMMON /SPECTR/ JCOUNT(MAXCHN)
0011 *          COMMON /SPDESC/ NXCHAN,NYCHAN,IDATIM(9),NRUN,ISPILL,ITARG,
     *         +                ITEXT(18),LABEL
0012 *          COMMON /OPPARS/ MDEV,SPESPC,PARSPC,MXITER,LBFIX,INREG,LIXLO,LIXHI,
     *         +                LIYLO,LIYHI,IYPWR,LIFLO,LIFHI,IPEAK,LBACK,LPEAK,
     *         +                LPEBA,NCOEFF,COEFF(MAXCO),ITERA,NFREE,
     *         +                PEAK(NPPARM,MAXPK),IFLAG(NPPARM,MAXPK),
     *         +                INDEX(NPPARM,MAXPK),ISELEC,IGNOR(MAXPK),NREG,
     *         +                KREGLO(MAXREG),KREGHI(MAXREG),IMODE,INTCOR,NCALF,
     *         +                CALFAC(MXCALF),SIGMA(MAXPK),GAMMA(MAXPK),CALSPC,
     *         +                FILPRT,ENESPC
0013 *          COMMON /OPPART/ IOPARS
0014 *          COMMON /RESULT/ CHINEW,CHIOLD,FVBACK,FVPEAK
0015 *          COMMON /ERRCOM/ IERROR
0016 *          COMMON /PARAM/  ILINK(NPPARM,MXLINK),NPAR,NDIMM,KUSE,KSKIP,
     *         +                TOLO(6),TOHI(6)
0017 *          COMMON /ARRAYS/ VMAT(MAXPAR*(MAXPAR+1)/2),DERIV(MAXPAR),
     *         +                RVEC(MAXPAR),HDERIV(MAXPAR)
0018 *          COMMON /CONST/  PICON,EPSILN,SQRTPI,SRLN2
     *  C*
0019 *          LOGICAL FILOPN
0020 *          COMMON /GLOBAL/ IOP,T0,IFIT,FILOPN
     *  C*

PDP-11 FORTRAN-77 V5.0-0        09:34:28    26-Sep-86          Page 2
FITMEN.FTN,1                /F77/OP/TR:BLOCKS/WR

0021 *          DATA IOPARS /0/
     *  C*
0022            COMMON /BUFMEN/ MBUFF(600)
                C**
                C*
                C*  Set up menu
                C*
0023             1000 CALL MNDEV (MDEV)
0024                  CALL MNCLR
0025                  CALL MNBUFF (MBUFF,600)
                C*
0026                  CALL MNHEAD ('FIT 0.0 -- Interactive fit of spectra')
0027                  CALL MNDEVC (1,'Menu device: 4=T2, 6=NU, 9=D1',MDEV,MDEV)
0028                  CALL MNOPT (3,'Get spectrum / Save fit parameters')
0029                  CALL MNOPT (4,'Display')
0030                  CALL MNOPT (6,'Fit spectrum')
0031                  CALL MNOPT (8,'Output results to print file')
0032                  CALL MNOPT (9,'Route print file to printer')
0033                  CALL MNOPT (11,'Save spectrum and fit parameters for plotting')
0034                  CALL MNOPT (10,'Redisplay results')
0035                  CALL MNDISP
0036                  CALL MNIN (IOP)
0037                  IF (IOP) 9000,1000,1005
                C*
0038             1005 GOTO (1000, 1000, 9000, 9000, 1000, 9000, 1000, 9000,
                +       9000, 1000, 9000, 1000, 1000, 1000, 1000, 1000,
                +       1000, 9000), IOP
                C*
                C*  Return to calling program
                C*
0039             9000 RETURN
                C*  ******
0040                 END
```

```
PDP-11 FORTRAN-77 V5.0-0        09:41:11    26-Sep-86          Page 1
FITOUT.FTN,1                /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE FITOUT (LUN)
                C*  ********************
                C*
                C*  This routine prints the results (LUN=LUNTRM -> screen)
                C*
                C*
                C*  Ver.  1.0/09-SEP-83(TK)  Original version
                C*  Ver.  1.1/11-OCT-83(DR)  Error included version
                C*  Ver.  1.2/19-JAN-84(TK)  Output format slightly changed
                C*  Ver.  2.0/02-FEB-84(TK)  Output of flags and calibration
                C*  Ver.  2.1/10-FEB-84(TK)  Output of FIT regions included
                C*  Ver.  2.2/10-JAN-85(DR)  Output of FIT date and time included
                C*
0002            PARAMETER MAXLIN=6
                C*
0003            INCLUDE 'FIT.COM'
0004 *          IMPLICIT REAL*8 (A-H,O-Z)
     *  C*
     *  C*  Logical unit numbers used:
     *  C*
     *  C*  LUNSPE: Lun to read spectrum
     *  C*  LUNPAR: Lun for PARFIL
     *  C*  LUNPRT: Lun for print output
     *  C*  LUNCAL: Lun for calibration input, and energy table
     *  C*  LUNTRM: Lun for terminal I/o (incl. DLPACK, MNPACK)
     *  C*
0005 *          PARAMETER LUNSPE=1, LUNCAL=2, LUNPRT=3, LUNPAR=4, LUNTRM = 5
     *  C*
0006 *          PARAMETER MAXCHN=4096, MAXPK=30, MAXCO=6, MAXREG=7, MAXPAR=40,
     *         +          NPPARM=6, MXCALF=3, MPOINT=500, MXLINK=20
     *  C*
0007 *          BYTE SPESPC(30),PARSPC(30),CALSPC(30),FILPRT(30),ENESPC(30)
     *  C*
0008 *          CHARACTER*20 HDGLIN
     *  C*
0009 *          INTEGER*4 JCOUNT
     *  C*
0010 *          REAL*4 T0,SECNDS
     *  C*
0011 *          COMMON /SPECTR/ JCOUNT(MAXCHN)
0012 *          COMMON /SPDESC/ NXCHAN,NYCHAN,IDATIM(9),NRUN,ISPILL,ITARG,
     *         +                ITEXT(18),LABEL
0013 *          COMMON /OPPARS/ MDEV,SPESPC,PARSPC,MXITER,LBFIX,INREG,LIXLO,LIXHI,
     *         +                LIYLO,LIYHI,IYPWR,LIFLO,LIFHI,IPEAK,LBACK,LPEAK,
     *         +                LPEBA,NCOEFF,COEFF(MAXCO),ITERA,NFREE,
     *         +                PEAK(NPPARM,MAXPK),IFLAG(NPPARM,MAXPK),
     *         +                INDEX(NPPARM,MAXPK),ISELEC,IGNOR(MAXPK),NREG,
     *         +                KREGLO(MAXREG),KREGHI(MAXREG),INODE,INTCOR,NCALF,
     *         +                CALFAC(MXCALF),SIGMA(MAXPK),GAMMA(MAXPK),CALSPC,
     *         +                FILPRT,ENESPC
0014 *          COMMON /OPPART/ IOPARS
0015 *          COMMON /RESULT/ CHINEW,CHIOLD,FVBACK,FVPEAK
0016 *          COMMON /ERRCOM/ IERROR
0017 *          COMMON /PARAM/  ILINK(NPPARM,MXLINK),NPAR,NDIMM,KUSE,KSKIP,
     *         +                TOLO(6),TOHI(6)

PDP-11 FORTRAN-77 V5.0-0        09:41:11    26-Sep-86          Page 2
FITOUT.FTN,1                /F77/OP/TR:BLOCKS/WR

0018 *          COMMON /ARRAYS/ VMAT(MAXPAR*(MAXPAR+1)/2),DERIV(MAXPAR),
     *         +                RVEC(MAXPAR),HDERIV(MAXPAR)
0019 *          COMMON /CONST/  PICON,EPSILN,SQRTPI,SRLN2
     *  C*
0020 *          LOGICAL FILOPN
0021 *          COMMON /GLOBAL/ IOP,T0,IFIT,FILOPN
     *  C*
0022 *          DATA IOPARS /0/
     *  C*
0023            LOGICAL NODLKY
                C*
0024            INTEGER*2 FTDAT(9),KDATE(5),KTIME(4)
                C*
0025            REAL*8 DCOEFF(MAXCO),DH(MAXPAR),DPEAK(NPPARM)
                C*
0026            DATA DT /1000.D0/, FTDAT /9*'  '/
                C*
0027            EQUIVALENCE (DPEAK(1),HDERIV(1)),(DCOEFF(1),HDERIV(NPPARM+1))
                C**
                C*  Get actual date and time
                C*
0028            CALL DATE (KDATE)
0029            CALL TIME (KTIME)
```

```
C*
C*    Start printout with header lines
C*
      IF (LUN .EQ. LUNTRM)  THEN
         CALL MNERAS
      ELSE
         WRITE (LUN,1000)
 1000    FORMAT ('1')
      ENDIF
      WRITE (LUN,1001) KDATE,KTIME,PARSPC
 1001 FORMAT ('+','Output from ',5A2,4A2,20X,30A1)
      WRITE (LUN,1005) IDATIM,NRUN,ISPILL,ITARG,ITEXT
 1005 FORMAT ('0',9A2,'  Run',I4,', Spill',I3,', Target',I3,/
     +        ' ',18A2)
      WRITE (LUN,1010) ITERA,CHINEW,NFREE,CHINEW/DBLE(NFREE)
 1010 FORMAT ('0',I2,'. Iteration.  CHI**2: ',F12.4,'  NFREE: ',I5/
     +        ' ','                 CHI**2/NFREE: ',F9.4)
C*
C*    Printout of FIT regions
C*
      WRITE (LUN,1015) ((KREGLO(J),KREGHI(J)),J=1,NREG)
 1015 FORMAT ('0','FIT regions used: ',/,
     +        ' ',6(I5,'-',I4))
C*
C*    Calculate error for background
C*
      DO 1020 I=1,NPAR,1
         DH(I) = SQRT (ABS (FITIJ (VMAT,NPAR,I,I)))
 1020 CONTINUE
C*
C*    Store errors for background
C*
```



```
      DO 1025 J=1,NCOEFF,1
         DCOEFF(J) = 0.0D0
         IF (LOFIX .EQ. 0)  DCOEFF(J) = DH(J)
 1025 CONTINUE
C*
C*    Printout of background
C*
      WRITE (LUN,1030) NCOEFF
 1030 FORMAT ('0','Background with',I3,' coefficients:')
      WRITE (LUN,1035) (COEFF(J),J=1,NCOEFF)
 1035 FORMAT (' ',' ','Coeffs: ',3(1PE16.8)/
     +        ' ',' ',         ',3(1PE16.8))
      WRITE (LUN,1040) (DCOEFF(J),J=1,NCOEFF)
 1040 FORMAT (' ',' ','Errors: ',3(1PE16.8)/
     +        ' ',' ',         ',3(1PE16.8))
C*
C*    Printout of calibration polynomial
C*
      IF (NCALF.GT.0) THEN
         WRITE (LUN,1045)
 1045    FORMAT ('0','Calibration polynomial:')
         WRITE (LUN,1035) (CALFAC(J),J=1,NCALF)
      ENDIF
C*
C*    Finish header page
C*
      IF (LUN .NE. LUNPRT)  THEN
         CALL DLOUT (' ')
         CALL DLYENO (' Do you want to see the results',1,IOP)
         IF (IOP .NE. 1)  GOTO 2000
         CALL MNERAS
      ENDIF
      NODLKY = .FALSE.
C*
C*    Printout of peaks
C*
      NPLINE = 0
      DO 1080  IPK=1,MAXPK,1
         IF (IGNOR(IPK) .EQ. 1) GOTO 1080
         NODLKY = .TRUE.
         DO 1050  IPARM=1,NPPARM,1
            IN = INDEX(IPARM,IPK)
            DPEAK(IPARM) = 0.0D0
            IF (IN .GT. 0)  DPEAK(IPARM) = DH(IN)
 1050    CONTINUE
C*
C*       Convert width to FWHM
C*
         IF (IFLAG(3,IPK) .GT. 0)  DPEAK(3) = DPEAK(3) * PEAK(3,IPK)
         IWEIG = INDEX(3,IPK)
C*
C*       Gaussian
C*
         IPARM = 2
         XPI = SORTPI
```



```
         WIDTH = SIGMA(IPK)
C*
C*       Lorentzian
C*
         IF (IFLAG(6,IPK) .NE. -2)  THEN
            IPARM = 6
            XPI = PICON
            WIDTH = GAMMA(IPK)
         ENDIF
         IWIDTH = INDEX(IPARM,IPK)
         DWIDTH = DPEAK(IPARM)
C*
C*       Area
C*
         AREA = PEAK(3,IPK) * WIDTH
         DAREA = (PEAK(3,IPK) * DWIDTH) **2 + (WIDTH * DPEAK(3)) **2
     +           * 2. * AREA * FITIJ(VMAT,NPAR,IWIDTH,IWEIG)
         DAREA = XPI * SQRT (ABS(DAREA))
         AREA  = XPI * AREA
         DPEAK(2) = DPEAK(2) * SRLN2
         DPEAK(6) = DPEAK(6) * 2.0D0
C*
C*       Start output
C*
         IF (NPLINE .EQ. 0)  THEN
            WRITE (LUN,1055)
 1055       FORMAT (' ')
            WRITE (LUN,1060)
 1060       FORMAT (' ',' ','Position',2X,'Gauss-FWHM',3X,'Height',1X,
     +                      'Low-tail',1X,'High-tail',1X,'Lorentz-FWHM',
     +                      1X,'Intensity')
         ENDIF
         WRITE (LUN,1065) IPK,(IFLAG(I,IPK),I=1,NPPARM)
 1065    FORMAT ('0','Peak',I3,10X,'(Flags: ',5(I2,','),I2,')')
         WRITE (LUN,1070) (PEAK(I,IPK),I=1,NPPARM),AREA
 1070    FORMAT (' ',F8.3,1X,F7.3,1X,F13.2,2X,0PF7.3,2X,F7.3,2X,F7.3,
     +           1X,F13.2)
         WRITE (LUN,1070) (DPEAK(I),I=1,NPPARM),DAREA
C*
C*       Print calibration (if any)
C*
         IF (NCALF.GT.0) THEN
            XPI = PEAK(1,IPK)
            WIDTH = 0.5D0 * PEAK(2,IPK)
            DWIDTH = 0.5D0 * PEAK(6,IPK)
            DPEAK(1) = 0.5D0 * DPEAK(1)
            DPEAK(2) = 0.5D0 * DPEAK(2)
            DPEAK(6) = 0.5D0 * DPEAK(6)
            WRITE (LUN,1075) CALVAL(XPI),
     +           CALVAL(XPI+DPLAN(1))-CALVAL(XPI-DPEAK(1)),
     +           DI*(CALVAL(XPI+WIDTH)-CALVAL(XPI-WIDTH)),
     +           DI*(CALVAL(XPI+DPEAK(2))-CALVAL(XPI-DPEAK(2))),
     +           DI*(CALVAL(XPI+DWIDTH)-CALVAL(XPI-DWIDTH)),
     +           DI*(CALVAL(XPI+DPEAK(6))-CALVAL(XPI-DPEAK(6)))
 1075       FORMAT (' ','E ',F9.4,'(',F8.4,'), ',
```



```
     +                    'G ',F8.4,'(',F8.4,'), ',
     +                    'L ',F8.4,'(',F8.4,')')
         ENDIF
C*
C*       Make a nice paging
C*
         NPLINE = NPLINE+1
         IF (NPLINE .GE. MAXLIN) THEN
            IF (LUN .EQ. LUNTRM) THEN
               NPLINE = 0
               CALL DLOUT (' ')
               CALL DLYENO (' Continue',1,IOP)
               IF (IOP .NE. 1)  GOTO 2000
               CALL MNERAS
               NODLKY = .FALSE.
            ENDIF
         ENDIF
 1080 CONTINUE
      IFII = 0
      IF (LUN .NE. LUNTRM)  GOTO 9000
      IF (NODLKY)  CALL DLKEY
C*
 2000 CALL MNERAS
      CALL DLYENO ('Do you want the picture',1,IFII)
C*
C*    Return to calling program
C*
 9000 RETURN
C*    ******
      END
```

```
PDP-11 FORTRAN-77 V5.0-0        09:41:49    26-Sep-86          Page 8
FITOUT.FTN,1              /F77/OP/TR:BLOCKS/WR

0001            DOUBLE PRECISION FUNCTION FITIJ (VMAT,NDIMM,I,J)
          C*    ************************************************
          C*
          C*    This function returns the value of element (I,J) of
          C*    packed symmetric matrix
          C*
          C*
          C*    Ver. 0.0/12-OCT-83 (DR)   Original version
          C*
0002            REAL*8 VMAT(1)
          C**
0003            IF (I .LE. 0  .OR.  I .GT. NDIMM)  GOTO 1000
0004            IF (J .LE. 0  .OR.  J .GT. NDIMM)  GOTO 1000
          C*
0005            IF (I .LE. J)  THEN
          C*
          C*       Upper triangular matrix
          C*
0006               II = I
0007               JJ = J
0008            ELSE
          C*
          C*       Lower triangular matrix
          C*
0009               II = J
0010               JJ = I
0011            ENDIF
          C*
          C*    Calculate index in packed matrix
          C*
0012            IJ = (2 * NDIMM - II) * (II - 1) / 2 + JJ
0013            FITIJ = VMAT(IJ)
0014            GOTO 9000
          C*
          C*    Out of range return
          C*
0015         1000 FITIJ = 0.0D0
          C*
          C*    Return to calling program
          C*
0016         9000 RETURN
0017            END
```

```
PDP-11 FORTRAN-77 V5.0-0        09:41:52    26-Sep-86          Page 10
FITOUT.FTN,1              /F77/OP/TR:BLOCKS/WR

0001            DOUBLE PRECISION FUNCTION CALVAL (DCHAN)
          C*    ****************************************
          C*
          C*    This function returns the ...
          C*
          C*
0002            INCLUDE 'FIT.COM'
0003 *          IMPLICIT REAL*8 (A-H,O-Z)
     *    C*
     *    C*    Logical unit numbers used:
     *    C*
     *    C*    LUNSPE: Lun to read spectrum
     *    C*    LUNPAR: Lun for PARFIL
     *    C*    LUNPRT: Lun for print output
     *    C*    LUNCAL: Lun for calibration input, and energy table
     *    C*    LUNTRM: Lun for terminal I/O (incl. DLPACK, MNPACK)
     *    C*
0004 *          PARAMETER LUNSPE=1, LUNCAL=2, LUNPRT=3, LUNPAR=4, LUNTRM = 5
     *    C*
0005 *          PARAMETER MAXCHN=4096, MAXPK=30, MAXCO=6, MAXREG=7, MAXPAR=40,
     *    +                 NPPARM=6, MXCALF=3, MPOINT=500, MXLINK=20
     *    C*
0006 *          BYTE SPESPC(30),PARSPC(30),CALSPC(30),FILPRT(30),ENESPC(30)
     *    C*
0007 *          CHARACTER*20 HOGLIN
     *    C*
0008 *          INTEGER*4 JCOUNT
     *    C*
0009 *          REAL*4 T0,SECNDS
     *    C*
0010 *          COMMON /SPECTR/ JCOUNT(MAXCHN)
0011 *          COMMON /SPDESC/ NXCHAN,NYCHAN,IDATIM(9),NRUN,ISPILL,ITARG,
     *    +                     ITEXT(18),LABEL
0012 *          COMMON /OPPARS/ MDEV,SPESPC,PARSPC,MXITER,LBFIX,INREG,LIXLO,LIXHI,
     *    +                     LIYLO,LIYHI,IYPWR,LIFLO,LIFHI,IPEAK,LBACK,LPEAK,
     *    +                     LPEGA,NCOEFF,COEFF(MAXCO),ITERA,NFREE,
     *    +                     PEAK(NPPARM,MAXPK),IFLAG(NPPARM,MAXPK),
     *    +                     INDEX(NPPARM,MAXPK),ISELEC,IGNOR(MAXPK),NREG,
     *    +                     KREGLO(MAXREG),KREGHI(MAXREG),IMODE,INTCOR,NCALF,
     *    +                     CALFAC(MXCALF),SIGMA(MAXPK),GAMMA(MAXPK),CALSPC,
     *    +                     FILPRT,ENESPC
0013 *          COMMON /OPPART/ IOPARS
0014 *          COMMON /RESULT/ CHINEW,CHIOLD,FVBACK,FVPEAK
0015 *          COMMON /ERRCOM/ IERROR
0016 *          COMMON /PARAM/  ILINK(NPPARM,MXLINK),NPAR,NDIMM,KUSE,KSKIP,
     *    +                     TOLO(6),TOHI(6)
0017 *          COMMON /ARRAYS/ VMAT(MAXPAR*(MAXPAR+1)/2),DERIV(MAXPAR),
     *    +                     RVEC(MAXPAR),HDERIV(MAXPAR)
0018 *          COMMON /CONST/  PICON,EPSILN,SQRTPI,SRLN2
     *    C*
0019 *          LOGICAL FILOPN
0020 *          COMMON /GLOBAL/ IOP,T0,IFIT,FILOPN
     *    C*
0021 *          DATA IOPARS /0/
          C*
```

```
PDP-11 FORTRAN-77 V5.0-0        09:41:52    26-Sep-86          Page 11
FITOUT.FTN,1              /F77/OP/TR:BLOCKS/WR

          C*    Apply calibration (if any)
          C*
0022            IF (NCALF.GT.0) THEN
0023               DTEMP = DCHAN
0024               CALVAL = CALFAC(1)
0025               DO 1000 J=2,NCALF,1
0026                  CALVAL = CALVAL+(DTEMP*CALFAC(J))
0027                  DTEMP = DTEMP*DTEMP
0028    1000       CONTINUE
0029            ELSE
0030               CALVAL = 0.0D0
0031            ENDIF
          C*
          C*    Return to calling program
          C*
0032            RETURN
          C*    ******
0033            END
```

```
PDP-11 FORTRAN-77 V5.0-0          09:39:51     26-Sep-86          Page 1
FITPAR.FTN;1                /F77/OP/TR:BLOCKS/WR

0001                SUBROUTINE FITPAR (IANW)
                C*  ************************
                C*
                C*  This routine defines parameters for fit
                C*
                C*
                C*  Ver.  0.0/09-SEP-83(TK)  Original version
                C*  Ver.  1.0/26-SEP-83(DR)
                C*  Ver.  1.0/02-DEC-83(DR)
                C*
0002                INCLUDE 'FIT.COM'
0003    *           IMPLICIT REAL*8 (A-H,O-Z)
        *       C*
        *       C*  Logical unit numbers used:
        *       C*
        *       C*  LUNSPE:  Lun to read spectrum
        *       C*  LUNPAR:  Lun for PARFIL
        *       C*  LUNPRT:  Lun for print output
        *       C*  LUNCAL:  Lun for calibration input, and energy table
        *       C*  LUNTRM:  Lun for terminal I/O (incl. DLPACK, MNPACK)
        *       C*
0004    *           PARAMETER LUNSPE=1, LUNCAL=2, LUNPRT=3, LUNPAR=4, LUNTRM = 5
        *       C*
0005    *           PARAMETER MAXCHN=4096, MAXPK=30, MAXCO=6, MAXREG=7, MAXPAR=40,
        *      +              NPPARM=6, MXCALF=3, MPOINT=500, MXLINK=20
        *       C*
0006    *           BYTE SPESPC(30),PARSPC(30),CALSPC(30),FILPRT(30),ENESPC(30)
        *       C*
0007    *           CHARACTER*20 HDGLIN
        *       C*
0008    *           INTEGER*4 JCOUNT
        *       C*
0009    *           REAL*4 T0,SECNDS
        *       C*
0010    *           COMMON /SPECTR/  JCOUNT(MAXCHN)
0011    *           COMMON /SPDESC/  NXCHAN,NYCHAN,IDATIM(9),NRUN,ISPILL,ITARG,
        *      +                     ITEXT(10),LABEL
0012    *           COMMON /OPPARS/  MDEV,SPESPC,PARSPC,MXITER,LBFIX,INREG,LIXLO,LIXHI,
        *      +                     LIYLO,LIYHI,IYPWR,LIFLO,LIFHI,IPEAK,LBACK,LPEAK,
        *      +                     LPEBA,NCOEFF,COEFF(MAXCO),ITERA,NFREE,
        *      +                     PEAK(NPPARM,MAXPK),IFLAG(NPPARM,MAXPK),
        *      +                     INDEX(NPPARM,MAXPK),ISELEC,IGNOR(MAXPK),NREG,
        *      +                     KREGLO(MAXREG),KREGHI(MAXREG),IMODE,INTCOR,NCALF,
        *      +                     CALFAC(MXCALF),SIGMA(MAXPK),GAMMA(MAXPK),CALSPC,
        *      +                     FILPRT,ENESPC
0013    *           COMMON /OPPART/  IOPARS
0014    *           COMMON /RESULT/  CHINEW,CHIOLD,FVBACK,FVPEAK
0015    *           COMMON /ERRCOM/  IERROR
0016    *           COMMON /PARAM/   ILINK(NPPARM,MXLINK),NPAR,NOIMM,KUSE,KSKIP,
        *      +                     TOLO(6),TOHI(6)
0017    *           COMMON /ARRAYS/  VMAT(MAXPAR*(MAXPAR+1)/2),DERIV(MAXPAR),
        *      +                     RVEC(MAXPAR),HDERIV(MAXPAR)
0018    *           COMMON /CONST/   PICON,EPSILN,SQRTPI,SRLN2
        *       C*
0019    *           LOGICAL FILOPN


PDP-11 FORTRAN-77 V5.0-0          09:39:51     26-Sep-86          Page 2
FITPAR.FTN;1                /F77/OP/TR:BLOCKS/WR

0020    *           COMMON /GLOBAL/ IOP,T0,IFIT,FILOPN
        *       C*
0021    *           DATA IOPARS /0/
                C*
0022                DATA TOLO /MAXCHN,   0.0D0,  0.0D0,  0.0D0,  0.0D0,  0.0D0/
0023                DATA TOHI / 0.0D0, MAXCHN, 1.0D9, 1.0D0, 1.0D0, 0.0D0/
                C**
                C*
0024                IERROR = 0
                C*
                C*  Clear screen
                C*
0025                CALL MNIRAS
0026                T0 = SECNDS(0.0)
                C*
                C*  Define limits for fitted quantities
                C*
0027                DO 1000 J=1,NREG,1
0028                   TOLO(1) = MIN (TOLO(1),DBLE(KREGLO(J)))
0029                   TOHI(1) = MAX (TOHI(1),DBLE(KREGHI(J)))
0030     1000    CONTINUE
0031                TOHI(2) = TOHI(1) - TOLO(1)
0032                TOHI(6) = TOHI(2)
                C*
                C*  Internal start-value corrections
                C*
0033                IF (INTCOR .NE. 1)  GOTO 1015
                C*
                C*  Setup start background parameters
                C*
0034                IF (LBFIX .EQ. 0)  THEN
                C*
                C*     Determine lower average
                C*
0035                   IL = 9999
0036                   IH = 0
0037                   DO 1005 J=1,NREG,1
0038                      IL = MIN (IL,KREGLO(J))
0039                      IH = MAX (IH,KREGHI(J))
0040     1005          CONTINUE
0041                   IL = MAX (2,IL)
0042                   IH = MIN (NXCHAN-1,IH)
0043                   SL = (JCOUNT(IL-1) + JCOUNT(IL) + JCOUNT(IL+1)) / 3
0044                   SH = (JCOUNT(IH-1) + JCOUNT(IH) + JCOUNT(IH+1)) / 3
                C*
0045                   COEFF(2) = (SH - SL) / DBLE(IH-IL)
0046                   IF (COEFF(2) .EQ. 0.0D0)  COEFF(2) = 5.0D-5
0047                   COEFF(1) = SL - COEFF(2) * IL
0048                   DO 1010 I=1,NCOEFF-1,1
0049                      COEFF(I) = SIGN (MAX (ABS(COEFF(I)),1.0D-15),COEFF(I))
0050                      IF (I .GE. 2)  COEFF(I+1) = - COEFF(I) / TOHI(1)**I
0051     1010          CONTINUE
0052                ENDIF
                C*
                C*  Prepare data for fit


PDP-11 FORTRAN-77 V5.0-0          09:39:51     26-Sep-86          Page 3
FITPAR.FTN;1                /F77/OP/TR:BLOCKS/WR

                C*
0053     1015    CALL VIZERO (INDEX,MAXPK*NPPARM)
0054             CALL VIZERO (ILINK,MXLINK*NPPARM)
                C*
                C*  Run through the fit regions
                C*
0055             KUSE = 0
0056             DO 1020 J=1,NREG,1
0057                KUSE = KUSE + KREGHI(J) - KREGLO(J) + 1
0058     1020    CONTINUE
0059             KSKIP = NXCHAN - KUSE
                C*
                C*  Reserve parameters for background
                C*
0060             NPAR = 0
0061             IF (LBFIX .EQ. 0)  NPAR = NCOEFF
                C*
                C*  Run through the peaks
                C*
0062             DO 1030 IPK=1,MAXPK,1
0063                IF (IGNOR(IPK) .EQ. 1)  GOTO 1030
0064                IF (INTCOR .EQ. 1)  CALL PEKDEF (IPK)
                C*
                C*  Set "forgotten" flags
                C*
0065                IF (PEAK(4,IPK) .EQ. 0.0D0)  IFLAG(4,IPK) = -2
0066                IF (PEAK(5,IPK) .EQ. 0.0D0)  IFLAG(5,IPK) = -2
0067                IF (PEAK(6,IPK) .EQ. 0.0D0)  THEN
0068                   IFLAG(6,IPK) = -2
                C     ELSE
                C        IFLAG(2,IPK) = -1
0069                ENDIF
                C*
                C*  Interpret flags
                C*
0070                DO 1025 IPARM=1,NPPARM,1
                C*
                C*     IFLAG = 0: Free parameter
                C*
0071                   IF (IFLAG(IPARM,IPK) .LE. 0)  THEN
0072                      NPAR = NPAR+1
0073                      IF (NPAR .GT. MAXPAR)  GOTO 7000
0074                      INDEX(IPARM,IPK) = NPAR
0075                   ENDIF
                C*
                C*     Linked parameter
                C*
0076                   IF (IFLAG(IPARM,IPK) .GT. 0)  THEN
0077                      K = IFLAG(IPARM,IPK)
0078                      IF (K .GT. MXLINK)  GOTO 7005
0079                      IF (ILINK(IPARM,K) .LE. 0) THEN
0080                         NPAR = NPAR+1
0081                         IF (NPAR .GT. MAXPAR)  GOTO 7000
0082                         ILINK(IPARM,K) = NPAR
0083                      ENDIF
```

```
PDP-11 FORTRAN-77 V5.0-0        09:39:51    26-Sep-86          Page 4
FITPAR.FTN,1                /F77/OP/TR:BLOCKS/WR

0084                      INDEX(IPARM,IPK) = ILINK(IPARM,K)
0085                    ENDIF
0086      1025      CONTINUE
0087      1030  CONTINUE
          C*
          C*    Calculate number of degrees of freedom
          C*
0088            NFREE = KUSE - NPAR
0089            NDIM = NPAR*(NPAR+1) / 2
0090            GOTO 9000
          C*
          C*    Error message: Too many parameters
          C*
0091       7000 CALL MNERAS
0092            CALL DLOUT ('0** Too many parameters **')
0093            CALL DLOUT (' NPAR = ', NPAR)
0094            CALL DLOUT (' KUSE = ',KUSE)
0095            CALL DLOUT (' NDIM = ',NDIM)
0096            GOTO 8000
          C*
          C*    Error message: Linkage flag too big
          C*
0097       7005 CALL MNERAS
0098            CALL DLOUT ('0** Linkage flag too big **')
0099       8000 CALL DLKEY
0100            IERROR = -1
          C*
          C*    Return to calling program
          C*
0101       9000 RETURN
          C*    ******
0102            END
```

```
PDP-11 FORTRAN-77 V5.0-0        09:36:38    26-Sep-86          Page 1
FITPFL.FTN,1                /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE FITPFL
          C*    ****************
          C*
          C*
0002            INCLUDE 'FIT.COM'
0003  *         IMPLICIT REAL*8 (A-H,O-Z)
      *   C*
      *   C*    Logical unit numbers used:
      *   C*
      *   C*    LUNSPE:  Lun to read spectrum
      *   C*    LUNPAR:  Lun for PARFIL
      *   C*    LUNPRT:  Lun for print output
      *   C*    LUNCAL:  Lun for calibration input, and energy table
      *   C*    LUNTRM:  Lun for terminal i/o (incl. DLPACK, MNPACK)
      *   C*
0004  *         PARAMETER LUNSPE=1, LUNCAL=2, LUNPRT=3, LUNPAR=4, LUNTRM = 5
      *   C*
0005  *         PARAMETER MAXCHN=4096, MAXPK=30, MAXCO=6, MAXREG=7, MAXPAR=40,
      *  +                  NPPARM=6, MXCALF=3, MPOINT=500, MXLINK=20
      *   C*
0006  *         BYTE SPESPC(30),PARSPC(30),CALSPC(30),FILPRT(30),ENCSPC(30)
      *   C*
0007  *         CHARACTER*20 HDGLIN
      *   C*
0008  *         INTEGER*4 JCOUNT
      *   C*
0009  *         REAL*4 T0,SECNDS
      *   C*
0010  *         COMMON /SPECTR/ JCOUNT(MAXCHN)
0011  *         COMMON /SPDESC/ NXCHAN,NYCHAN,IDATIM(9),NRUN,ISPILL,ITARG,
      *  +                      ITEXT(18),LABEL
0012  *         COMMON /OPPARS/ MDEV,SPESPC,PARSPC,MXITER,LBFIX,INREG,LIXLO,LIXHI,
      *  +                      LIYLO,LIYHI,IVPWR,LIFLO,LIFHI,IPEAK,LBACK,LPEAK,
      *  +                      LPEAK,NCOEFF,COEFF(MAXCO),ITERA,NFREE,
      *  +                      PEAK(NPPARM,MAXPK),IFLAG(NPPARM,MAXPK),
      *  +                      INDEX(NPPARM,MAXPK),ISELEC,IGNOR(MAXPK),NREG,
      *  +                      KREGLO(MAXREG),KREGHI(MAXREG),IMODE,INTCOR,NCALF,
      *  +                      CALFAC(MXCALF),SIGMA(MAXPK),GAMMA(MAXPK),CALSPC,
      *  +                      FILPRT,ENESPC
0013  *         COMMON /OPPART/ IOPARS
0014  *         COMMON /RESULT/ CHINEW,CHIOLD,FVBACK,FVPEAK
0015  *         COMMON /ERRCOM/ IERROR
0016  *         COMMON /PARAM/  ILINK(NPPARM,MXLINK),NPAR,NDIMM,KUSE,KSKIP,
      *  +                      TOLO(6),TOHI(6)
0017  *         COMMON /ARRAYS/ VMAT(MAXPAR*(MAXPAR+1)/2),DERIV(MAXPAR),
      *  +                      RVEC(MAXPAR),HDERIV(MAXPAR)
0018  *         COMMON /CONST/  PICON,EPSILN,SQRTPI,SRLN2
      *   C*
0019  *         LOGICAL FILOPN
0020  *         COMMON /GLOBAL/ IOP,T0,IFIT,FILOPN
      *   C*
0021  *         DATA IOPARS /0/
          C*
          C*    Option D(4):  Save fit parameters
          C*
```

```
PDP-11 FORTRAN-77 V5.0-0        09:36:38    26-Sep-86          Page 2
FITPFL.FTN,1                /F77/OP/TR:BLOCKS/WR

0022            IF (IOP .EQ. 4) THEN
0023               CALL PARLUN (LUNCAL)
0024               CALL PARFIL (PARSPC)
0025               CALL PARWT
0026               GOTO 1000
0027            ENDIF
          C*
          C*    Option E(5):  Read fit parameters
          C*
0028            IF (IOP .EQ. 5) THEN
0029               CALL PARLUN (LUNCAL)
0030               CALL PARFIL (PARSPC)
0031               CALL PARRD
0032               GOTO 1000
0033            ENDIF
          C*
0034       1000 CALL PARLUN
0035            CALL PARFIL
          C*
          C*    Return to calling program
          C*
0036       9000 RETURN
          C*    ******
0037            END
```

```
PDP-11 FORTRAN-77 V5.0-0        09:38:05    26-Sep-86          Page 1
FITPEA.FTN;1                /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE FITPEA (X,IBACK)
                C*************************
                C*
                C*      This subroutine calculates at a given -X- the contribution
                C*      of the peaks
                C*
                C*              < 0 :   Only one peak (-IBACK) for display and output
                C*      IBACK   = 0 :   All peaks for fit processing
                C*              > 0 :   All peaks for display and output
                C*
                C*
                C*      Ver. 1.0/02-DEC-83 (DR)    Original version
                C*      Ver. 1.1/25-JAN-84 (TK)    Implemetation of peak selection
                C*      Ver. 1.2/06-FEB-84 (DR)    Implemetation of folding with complex
                C*                                 error function
                C*
0002            INCLUDE 'FIT.COM'
0003 =          IMPLICIT REAL*8 (A-H,O-Z)
     =  C*
     =  C*      Logical unit numbers used:
     =  C*
     =  C*      LUNSPE: Lun to read spectrum
     =  C*      LUNPAR: Lun for PARFIL
     =  C*      LUNPRT: Lun for print output
     =  C*      LUNCAL: Lun for calibration input, and energy table
     =  C*      LUNTRM: Lun for terminal i/o (incl. DLPACK, MNPACK)
     =  C*
0004 =          PARAMETER LUNSPE=1, LUNCAL=2, LUNPRT=3, LUNPAR=4, LUNTRM = 5
     =  C*
0005 =          PARAMETER MAXCHN=4096, MAXPK=30, MAXCO=6, MAXREG=7, MAXPAR=40,
     =     +            NPPARM=6, MXCALF=3, MPOINT=500, MXLINK=20
     =  C*
0006 =          BYTE SPESPC(30),PARSPC(30),CALSPC(30),FILPRT(30),ENESPC(30)
     =  C*
0007 =          CHARACTER*20 HDGLIN
     =  C*
0008 =          INTEGER*4 JCOUNT
     =  C*
0009 =          REAL*4 T0,SECNDS
     =  C*
0010 =          COMMON /SPECTR/ JCOUNT(MAXCHN)
0011 =          COMMON /SPDESC/ NXCHAN,NYCHAN,IDATIM(9),NRUN,ISPILL,ITARG,
     =     +                    ITEXT(18),LABEL
0012 =          COMMON /OPPARS/ MDEV,SPESPC,PARSPC,MXITER,LBFIX,INREG,LIXLO,LIXHI,
     =     +                    LIYLO,LIYHI,IYPWR,LIFLO,LIFHI,IPEAK,LBACK,LPEAK,
     =     +                    LPEBA,NCOEFF,COEFF(MAXCO),ITERA,NFREE,
     =     +                    PEAK(NPPARM,MAXPK),IFLAG(NPPARM,MAXPK),
     =     +                    INDEX(NPPARM,MAXPK),ISELEC,IGNOR(MAXPK),NREG,
     =     +                    KREGLO(MAXREG),KREGHI(MAXREG),INODI,INICOR,NCALF,
     =     +                    CALFAC(MXCALF),SIGMA(MAXPK),GAMMA(MAXPK),CALSPC,
     =     +                    FILPRT,ENESPC
0013 =          COMMON /OPPART/ IOPARS
0014 =          COMMON /RESULT/ CHINEW,CHIOLD,FVBACK,FVPEAK
0015 =          COMMON /ERRCOM/ IERROR
0016 =          COMMON /PARAM/  ILINK(NPPARM,MXLINK),NPAR,NDIMM,KUSE,KSKIP,


PDP-11 FORTRAN-77 V5.0-0        09:38:05    26-Sep-86          Page 2
FITPEA.FTN;1                /F77/OP/TR:BLOCKS/WR

     =     +                    TOLO(6),TOHI(6)
0017 =          COMMON /ARRAYS/ VMAT(MAXPAR*(MAXPAR+1)/2),DERIV(MAXPAR),
     =     +                    RVEC(MAXPAR),HDERIV(MAXPAR)
0018 =          COMMON /CONST/  PICON,EPSILN,SQRTPI,SRLN2
     =  C*
0019 =          LOGICAL FILOPN
0020 =          COMMON /GLOBAL/ IOP,T0,IFIT,FILOPN
     =  C*
0021 =          DATA IOPARS /0/
                C*
0022            COMPLEX CWERF,C
                C**
                C*      Prepare values
                C*
0023            FVPEAK = 0.0D0
                C*
                C*      Determine 'peak range'
                C*
0024            IF (IBACK .GE. 0)  THEN
0025               IL = 1
0026               IH = MAXPK
0027            ELSE
0028               IL = -IBACK
0029               IH = -IBACK
0030            ENDIF
                C*
                C*      Calculate contribution of peaks
                C*
0031            DO 1015 IP=IL,IH,1
0032               IF (IGNOR(IP) .EQ. 1)  GOTO 1015
0033               IF (IBACK .EQ. 0)  THEN
0034                  I1 = INDEX(1,IP)
0035                  I2 = INDEX(2,IP)
0036                  I3 = INDEX(3,IP)
0037                  I6 = INDEX(6,IP)
0038               ENDIF
0039               XT = X - PEAK(1,IP)
0040               HEIGHT = PEAK(3,IP)
0041               IF (IFLAG(6,IP) .NE. -2)  THEN
                   C*
                   C*      A. Lorentzian peak
                   C*
0042                  ZREAL = XT / SIGMA(IP)
0043                  ZIMAG = GAMMA(IP) / SIGMA(IP)
0044                  C = CWERF(CMPLX(ZREAL,ZIMAG))
0045                  DHEIG = DBLE(REAL(C)) * SQRTPI * ZIMAG
0046                  IF (IBACK .EQ. 0)  THEN
0047                     CDERIV = 2.D0 * HEIGHT * ZIMAG * SQRTPI / SIGMA(IP)
0048                     DGAML = CDERIV * (ZREAL * DBLE(AIMAG(C)) -
     +                                 (ZIMAG + 0.5D0 / ZIMAG) * DBLE(REAL(C))
     +                                 - 1.D0 / SQRTPI)
0049                     DPOSI = - CDERIV * (ZIMAG * DBLE(AIMAG(C)) -
     +                                 ZREAL * DBLE(REAL(C)))
0050                  ENDIF
0051               ELSE


FITPEA.FTN;1                /F77/OP/TR:BLOCKS/WR

0052                  U = XT / SIGMA(IP)
                   C*
                   C*  or      B. Exponential tail to the left
                   C*
0053                  IF (IFLAG(4,IP) .NE. -2)  THEN
0054                     F = PEAK(4,IP)
0055                     IF (U .LT. F)  GOTO 1000
0056                  ENDIF
                   C*
                   C*  or      C. Exponential tail to the right
                   C*
0057                  IF (IFLAG(5,IP) .NE. -2)  THEN
0058                     F = PEAK(5,IP)
0059                     IF (U .GT. F)  GOTO 1000
0060                  ENDIF
0061                  GOTO 1005
0062  1000            DHEIG = (F - 2.D0 * U) * F
0063                  GOTO 1010
                   C*
                   C*  or      D. Pure Gaussian peak
                   C*
0064  1005            IF (ABS(U) .GT. 2.6D0)  GOTO 1015
0065                  DHEIG = - U * U
                   C*
                   C*      Determine derivatives
                   C*
0066  1010            DHEIG = EXP(DHEIG)
0067                  IF (IBACK .EQ. 0)  THEN
0068                     DPOSI = 2.D0 * U / SIGMA(IP) * HEIGHT * DHEIG
0069                     DSIGM = U * DPOSI
0070                  ENDIF
0071               ENDIF
                   C*
                   C*      Determine results and update derivatives (if needed)
                   C*
0072               FUNCT  = HEIGHT * DHEIG
0073               FVPEAK = FVPEAK + FUNCT
0074               IF (IBACK .EQ. 0)  THEN
0075                  IF (IFLAG(3,IP)  .GT.  0)  DHEIG = FUNCT
0076                  IF (I1 .GT. 0)  DERIV(I1) = DERIV(I1) + DPOSI
0077                  IF (I2 .GT. 0)  DERIV(I2) = DERIV(I2) + DSIGM
0078                  IF (I3 .GT. 0)  DERIV(I3) = DERIV(I3) + DHEIG
0079                  IF (I6 .GT. 0)  DERIV(I6) = DERIV(I6) + DGAML
0080               ENDIF
0081  1015 CONTINUE
                C*
                C*      Return to calling program
                C*
0082            RETURN
                C*      ******
0083            END
```

```
PDP-11 FORTRAN-77 V5.0-0        09:42:01    26-Sep-86          Page 1
FITPRT.FTN,1                /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE FITPRT
                C*  ****************
                C*
                C*  This subroutine handles the printout of the fit results
                C*  to a print file
                C*
                C*
                C*  Ver. 0.0/12-Oct-83(TK)  Preliminary version
                C*
0002            INCLUDE 'FIT.COM'
0003 *          IMPLICIT REAL*8 (A-H,O-Z)
     *      C*
     *      C*  Logical unit numbers used:
     *      C*
     *      C*  LUNSPE: Lun to read spectrum
     *      C*  LUNPAR: Lun for PARFIL
     *      C*  LUNPRT: Lun for print output
     *      C*  LUNCAL: Lun for calibration input, and energy table
     *      C*  LUNTRM: Lun for terminal I/o (incl. DLPACK, MNPACK)
     *      C*
0004 *          PARAMETER LUNSPE=1, LUNCAL=2, LUNPRT=3, LUNPAR=4, LUNTRM = 5
     *      C*
0005 *          PARAMETER MAXCHN=4096, MAXPK=30, MAXCO=6, MAXREG=7, MAXPAR=40,
     *     +            NPPARM=6, MXCALF=3, MPOINT=500, MXLINK=20
     *      C*
0006 *          BYTE SPESPC(30),PARSPC(30),CALSPC(30),FILPRT(30),ENESPC(30)
     *      C*
0007 *          CHARACTER*20 HDGLIN
     *      C*
0008 *          INTEGER*4 JCOUNT
     *      C*
0009 *          REAL*4 T0,SECNDS
     *      C*
0010 *          COMMON /SPECTR/ JCOUNT(MAXCHN)
0011 *          COMMON /SPDESC/ NXCHAN,NYCHAN,IDATIM(9),NRUN,ISPILL,ITARG,
     *     +                    ITEXT(18),LABEL
0012 *          COMMON /OPPARS/ MDEV,SPESPC,PARSPC,MXITER,LBFIX,INREG,LIXLO,LIXHI,
     *     +                    LIYLO,LIYHI,IYPWR,LIFLO,LIFHI,IPEAK,LBACK,LPEAK,
     *     +                    LPEBA,NCOEFF,COEFF(MAXCO),ITERA,NFREE,
     *     +                    PEAK(NPPARM,MAXPK),IFLAG(NPPARM,MAXPK),
     *     +                    INDEX(NPPARM,MAXPK),ISELEC,IGNOR(MAXPK),NREG,
     *     +                    KREGLO(MAXREG),KREGHI(MAXREG),IMODE,INTCOR,NCALF,
     *     +                    CALFAC(MXCALF),SIGMA(MAXPK),GAMMA(MAXPK),CALSPC,
     *     +                    FILPRT,ENESPC
0013 *          COMMON /OPPART/ IOPARS
0014 *          COMMON /RESULT/ CHINEW,CHIOLD,FVBACK,FVPEAK
0015 *          COMMON /ERRCOM/ IERROR
0016 *          COMMON /PARAM/  ILINK(NPPARM,MXLINK),NPAR,NDIMM,KUSE,KSKIP,
     *     +                    TOLO(6),TOHI(6)
0017 *          COMMON /ARRAYS/ VMAT(MAXPAR*(MAXPAR+1)/2),DERIV(MAXPAR),
     *     +                    RVEC(MAXPAR),HDERIV(MAXPAR)
0018 *          COMMON /CONST/  PICON,EPSILN,SQRTPI,SRLN2
     *      C*
0019 *          LOGICAL FILOPN
0020 *          COMMON /GLOBAL/ IOP,T0,IFIT,FILOPN

PDP-11 FORTRAN-77 V5.0-0        09:42:01    26-Sep-86          Page 2
FITPRT.FTN,1                /F77/OP/TR:BLOCKS/WR

     *      C*
0021 *          DATA IOPARS /0/
                C**
                C*  Ask for file-name
                C*
0022            CALL MNERAS
0023            CALL DLYLNO (' Do you want a new print output-file',0,IAN)
0024            IF (.NOT.FILOPN .OR. IAN.EQ.1) THEN
0025               IF (FILOPN) CLOSE (UNIT=LUNPRT)
0026               FILOPN = .FALSE.
0027               CALL DLTXT (' Print output-file name',FILPRT,FILPRT,30,LEN)
0028            ENDIF
                C*
                C*  Open print file (if not allready open)
                C*
0029            IF (.NOT.FILOPN) THEN
0030               CALL ASSIGN (LUNPRT,FILPRT)
0031               FILOPN = .TRUE.
0032            ENDIF
                C*
                C*  Print results (using FITOUT)
                C*
0033            CALL FITOUT (LUNPRT)
                C*
                C*  Return to calling program
                C*
0034            RETURN
                C*  ******
0035            END
```

```
PDP-11 FORTRAN-77 V5.0-0        09:37:47    26-Sep-86          Page 1
FITROU.FTN,1                /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE FITROU
                C*  ****************
                C*
                C*  This subroutine routes a print file to the printer
                C*
                C*
                C*  Ver. 0.0/12-Oct-83(TK)  Preliminary version
                C*
0002            INCLUDE 'FIT.COM'
0003 *          IMPLICIT REAL*8 (A-H,O-Z)
     *      C*
     *      C*  Logical unit numbers used:
     *      C*
     *      C*  LUNSPE: Lun to read spectrum
     *      C*  LUNPAR: Lun for PARFIL
     *      C*  LUNPRT: Lun for print output
     *      C*  LUNCAL: Lun for calibration input, and energy table
     *      C*  LUNTRM: Lun for terminal I/o (incl. DLPACK, MNPACK)
     *      C*
0004 *          PARAMETER LUNSPE=1, LUNCAL=2, LUNPRT=3, LUNPAR=4, LUNTRM = 5
     *      C*
0005 *          PARAMETER MAXCHN=4096, MAXPK=30, MAXCO=6, MAXREG=7, MAXPAR=40,
     *     +            NPPARM=6, MXCALF=3, MPOINT=500, MXLINK=20
     *      C*
0006 *          BYTE SPESPC(30),PARSPC(30),CALSPC(30),FILPRT(30),ENESPC(30)
     *      C*
0007 *          CHARACTER*20 HDGLIN
     *      C*
0008 *          INTEGER*4 JCOUNT
     *      C*
0009 *          REAL*4 T0,SECNDS
     *      C*
0010 *          COMMON /SPECTR/ JCOUNT(MAXCHN)
0011 *          COMMON /SPDESC/ NXCHAN,NYCHAN,IDATIM(9),NRUN,ISPILL,ITARG,
     *     +                    ITEXT(18),LABEL
0012 *          COMMON /OPPARS/ MDEV,SPESPC,PARSPC,MXITER,LBFIX,INREG,LIXLO,LIXHI,
     *     +                    LIYLO,LIYHI,IYPWR,LIFLO,LIFHI,IPEAK,LBACK,LPEAK,
     *     +                    LPEBA,NCOEFF,COEFF(MAXCO),ITERA,NFREE,
     *     +                    PEAK(NPPARM,MAXPK),IFLAG(NPPARM,MAXPK),
     *     +                    INDEX(NPPARM,MAXPK),ISELEC,IGNOR(MAXPK),NREG,
     *     +                    KREGLO(MAXREG),KREGHI(MAXREG),IMODE,INTCOR,NCALF,
     *     +                    CALFAC(MXCALF),SIGMA(MAXPK),GAMMA(MAXPK),CALSPC,
     *     +                    FILPRT,ENESPC
0013 *          COMMON /OPPART/ IOPARS
0014 *          COMMON /RESULT/ CHINEW,CHIOLD,FVBACK,FVPEAK
0015 *          COMMON /ERRCOM/ IERROR
0016 *          COMMON /PARAM/  ILINK(NPPARM,MXLINK),NPAR,NDIMM,KUSE,KSKIP,
     *     +                    TOLO(6),TOHI(6)
0017 *          COMMON /ARRAYS/ VMAT(MAXPAR*(MAXPAR+1)/2),DERIV(MAXPAR),
     *     +                    RVEC(MAXPAR),HDERIV(MAXPAR)
0018 *          COMMON /CONST/  PICON,EPSILN,SQRTPI,SRLN2
     *      C*
0019 *          LOGICAL FILOPN
0020 *          COMMON /GLOBAL/ IOP,T0,IFIT,FILOPN
     *      C*

PDP-11 FORTRAN-77 V5.0-0        09:37:47    26-Sep-86          Page 2
FITROU.FTN,1                /F77/OP/TR:BLOCKS/WR

0021 *          DATA IOPARS /0/
                C*
                C*
                C*  Check wether a printfile is open or not
                C*
0022            CALL MNERAS
0023            IF (.NOT.FILOPN) THEN
0024               CALL DLOUT ('No print file open.')
0025               CALL DLKEY
0026            ELSE
0027               CLOSE (UNIT=LUNPRT)
                C     CALL ASSIGN (LUNPRT,'FITRES.DMP')
0028            ENDIF
                C*
                C*  Return to calling program
                C*
0029            RETURN
                C*  ******
0030            END
```

```
PDP-11 FORTRAN-77 V5.0-0        09:38:27    26-Sep-86          Page 1
FITSET.FTN;1              /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE FITSET
          C*    ******************
          C*
          C*    This subroutine sets the fit parameters in a dialoge mode
          C*
0002            INCLUDE 'FIT.COM'
0003  *         IMPLICIT REAL*8 (A-H,O-Z)
      *   C*
      *   C*    Logical unit numbers used:
      *   C*
      *   C*    LUNSPE:  Lun to read spectrum
      *   C*    LUNPAR:  Lun for PARFIL
      *   C*    LUNPRT:  Lun for print output
      *   C*    LUNCAL:  Lun for calibration input, and energy table
      *   C*    LUNTRM:  Lun for terminal i/o (incl. DLPACK, MNPACK)
      *   C*
0004  *         PARAMETER LUNSPE=1, LUNCAL=2, LUNPRT=3, LUNPAR=4, LUNTRM = 5
      *   C*
0005  *         PARAMETER MAXCHN=4096, MAXPK=30, MAXCO=6, MAXREG=7, MAXPAR=40,
      *        +          NPPARM=6, MXCALF=3, MPOINT=500, MXLINK=20
      *   C*
0006  *         BYTE SPESPC(30),PARSPC(30),CALSPC(30),FILPRT(30),ENESPC(30)
      *   C*
0007  *         CHARACTER*20 HDGLIN
      *   C*
0008  *         INTEGER*4 JCOUNT
      *   C*
0009  *         REAL*4 T0,SECNDS
      *   C*
0010  *         COMMON /SPECTR/ JCOUNT(MAXCHN)
0011  *         COMMON /SPDESC/ NXCHAN,NYCHAN,IDATIM(9),NRUN,ISPILL,ITARG,
      *        +                ITEXT(18),LABEL
0012  *         COMMON /OPPARS/ MDEV,SPESPC,PARSPC,MXITER,LBFIX,INREG,LIXLO,LIXHI,
      *        +                LIYLO,LIYHI,IYPWR,LIFLO,LIFHI,IPEAK,LBACK,LPEAK,
      *        +                LPEBA,NCOEFF,COEFF(MAXCO),ITERA,NFREE,
      *        +                PEAK(NPPARM,MAXPK),IFLAG(NPPARM,MAXPK),
      *        +                INDEX(NPPARM,MAXPK),ISELEC,IGNOR(MAXPK),NREG,
      *        +                KREGLO(MAXREG),KREGHI(MAXREG),IMODE,INTCOR,NCALF,
      *        +                CALFAC(MXCALF),SIGMA(MAXPK),GAMMA(MAXPK),CALSPC,
      *        +                FILPRT,ENESPC
0013  *         COMMON /OPPART/ IOPARS
0014  *         COMMON /RESULT/ CHINEW,CHIOLD,FVBACK,FVPEAK
0015  *         COMMON /ERRCOM/ IERROR
0016  *         COMMON /PARAM/  ILINK(NPPARM,MXLINK),NPAR,NDIMM,KUSE,KSKIP,
      *        +                TOLO(6),TOHI(6)
0017  *         COMMON /ARRAYS/ VMAT(MAXPAR*(MAXPAR+1)/2),DERIV(MAXPAR),
      *        +                RVEC(MAXPAR),HDERIV(MAXPAR)
0018  *         COMMON /CONST/  PICON,EPSILN,SQRTPI,SRLN2
      *   C*
0019  *         LOGICAL FILOPN
0020  *         COMMON /GLOBAL/ IOP,T0,IFIT,FILOPN
      *   C*
0021  *         DATA IOPARS /0/
          C*
0022            COMMON /BUFMEN/ MBUFF(600)

PDP-11 FORTRAN-77 V5.0-0        09:38:27    26-Sep-86          Page 2
FITSET.FTN;1              /F77/OP/TR:BLOCKS/WR

          C**
          C*
          C*    If no spectrum present: Go home
          C*
0023            IF (NXCHAN .LE. 0)  THEN
0024               CALL MNERAS
0025               CALL DLOUT ('0No spectrum to fit')
0026               CALL DLKEY
0027               IERROR = -1
0028               GOTO 9000
0029            ENDIF
          C*
          C*    Display sub menu
          C*
0030            IFIT = 0
0031     1000   CALL MNCLR
0032            CALL MNBUFF (MBUFF,600)
0033            CALL MNHEAD ('Set parameters for fit')
0034            CALL MNDEC  ( 1,'Maximum iteration count',MXITER,MXITER,0,32)
0035            CALL MNOPT  ( 2,'Set convergence radius for fit')
0036            CALL MNDEC  ( 3,'No of background coeffs.',NCOEFF,NCOEFF,1,MAXCO)
0037            CALL MNYENO ( 4,'Fixed background',LBFIX,LBFIX)
0038            IF (LBFIX.EQ.1) CALL MNDEC (5,'Set background coeff.',
     +                                      ICOEFF,ICOEFF,1,NCOEFF)
0039            CALL MNDEC  ( 6,'No of fit regions',NREG,NREG,1,MAXREG)
0040            CALL MNDEC  ( 7,'Define fit region',INREG,INREG,1,NREG)
0041            CALL MNDEC  ( 8,'Lower end of fit region',
     +                        KREGLO(INREG),KREGLO(INREG),1,MAXCHN)
0042            CALL MNDEC  ( 9,'Upper end of fit region',
     +                        KREGHI(INREG),KREGHI(INREG),1,MAXCHN)
0043            CALL MNOPT  (11,'Reset peak parameters and disable peaks')
0044            CALL MNYENO (12,'Do internal start-value corrections',
     +                        INTCOR,INTCOR)
0045            CALL MNOPT  (14,'Define peaks by crosshair cursor')
0046            CALL MNOPT  (15,'Define peaks by setting parameters')
0047            CALL MNOPT  (17,'Define flags globally')
0048            CALL MNOPT  (19,'Start the fit')
0049            CALL MNDISP
0050            CALL MNIN (IOP)
0051            IF (IOP) 1065,1000,1005
          C*
          C*    The big option switch
          C*
0052     1005   GOTO (1000, 1006, 1000, 1000, 1010, 1000, 1000, 1000,
     +                1000, 1000, 1020, 1000, 1000, 1020, 1035, 1000,
     +                1040, 1000, 1045), IOP
          C*
          C*    Option 8(2): Set convergence radius for fit
          C*
0053     1006   CALL DIPAGE
0054            WRITE (5,1007) EPSILN
0055     1007   FORMAT (' ',' Old convergence radius  = ',E15.7,/,
     +              ' ',' Enter new value')
0056            READ (5,*) EPSILN
0057            GOTO 1000

PDP-11 FORTRAN-77 V5.0-0        09:38:27    26-Sep-86          Page 3
FITSET.FTN;1              /F77/OP/TR:BLOCKS/WR

          C*
          C*    Option E(5): Set background coefficient
          C*
0058     1010   WRITE (5,1015) ICOEFF,COEFF(ICOEFF)
0059     1015   FORMAT (' ','Background coefficient',I3,' = ',E15.7,/,
     +              ' ','Enter new value')
0060            READ (5,*) COEFF(ICOEFF)
0061            GOTO 1000
          C*
          C*    Option K(11): Reset peak parameters and disable peaks
          C*
0062     1020   DO 1025 J=1,MAXPK,1
0063               IGNOR(J) = 1
0064     1025   CONTINUE
0065            CALL VDZERO (PEAK,MAXPK*NPPARM)
0066            CALL VIZERO (IFLAG,MAXPK*NPPARM)
0067            IF (IOP .EQ. 14)  GOTO 1030
0068            GOTO 1000
          C*
          C*    Option N(14): Define peaks by crosshair cursor
          C*
0069     1030   CALL DIPAGE
0070            CALL FITCUR
0071            INTCOR = 1
0072            GOTO 1000
          C*
          C*    Option O(15): Define peaks by setting parameters
          C*
0073     1035   CALL DIPAGE
0074            CALL FITST1
0075            GOTO 1000
          C*
          C*    Option G(17): Define flags globally
          C*
0076     1040   CALL DIPAGE
0077            CALL FITST2
0078            GOTO 1000
          C*
          C*    Option S(19): Start the fit
          C*    Set parameters for fit
          C*
0079     1045   IANW = 1
0080            CALL FITPAR (IANW)
0081            IF (IERROR .LT. 0)  GOTO 9000
0082            IF (IANW .NE. 1)  GOTO 1000
```

```
      C*
      C*     Display peak parameters and ask for confirmation
      C*
             CALL DIPAGE
             WRITE (5.1050)
        1050 FORMAT (' ','The following peaks will be fitted:',//,
            +        ' ','No',4X,'Position',4X,'Gauss-FWHM',7X,'Height',5X,
            +        'Lorentz-FWHM')
             DO 1060 J=1.MAXPK,1
                IF (IGNOR(J) .EQ. 0)  THEN
```

```
PDP-11 FORTRAN-77 V5.0-0          09:38:27      26-Sep-86             Page 4
FITSET.FTN;1                 /F77/OP/TR:BLOCKS/WR

                   WRITE (5.1055) J.PEAK(1.J).IFLAG(1.J).PEAK(2.J).IFLAG(2.J).
            +                      PEAK(3.J).IFLAG(3.J).PEAK(6.J).IFLAG(6.J)
        1055       FORMAT (' ',I2,': ',F8.3,'(',I2,')',F8.3,'(',I2,')',
            +                      F13.2,'(',I2,')',F8.3,'(',I2,')')
                ENDIF
        1060 CONTINUE
             CALL DLOUT (' ')
             CALL DLYENO ('Correct',1,J)
             IF (J .LE. 0)  GOTO 1000
             CALL DIPAGE
             IFIT = 1
             GOTO 9000
      C*
      C*     Error return :   no fit
      C*
        1065 IF (IOP .LT. 0)  IERROR = -1
      C*
      C*     Return to calling program
      C*
        9000 RETURN
      C*     ******
             END
```

```
PDP-11 FORTRAN-77 V5.0-0          09.36:47      26-Sep-86             Page 1
FITSPR.FTN;1                 /F77/OP/TR:BLOCKS/WR

                   SUBROUTINE FITSPR
             C*    *****************
             C*
             C*
                   INCLUDE 'FIT.COM'
                   IMPLICIT REAL*8 (A-H,O-Z)
             C*
             C*    Logical unit numbers used:
             C*
             C*    LUNSPE:  Lun to read spectrum
             C*    LUNPAR:  Lun for PARFIL
             C*    LUNPRT:  Lun for print output
             C*    LUNCAL:  Lun for calibration input, and energy table
             C*    LUNTRM:  Lun for terminal I/o (incl. DLPACK, MNPACK)
             C*
                   PARAMETER LUNSPE=1, LUNCAL=2, LUNPRT=3, LUNPAR=4, LUNTRM = 5
             C*
                   PARAMETER MAXCHN=4096, MAXPK=30, MAXCO=6, MAXREG=7, MAXPAR=40,
                  +          NPPARM=6, MXCALF=3, MPOINT=500, MXLINK=20
             C*
                   BYTE SPESPC(30),PARSPC(30),CALSPC(30),FILPRT(30),ENESPC(30)
             C*
                   CHARACTER*20 HDGLIN
             C*
                   INTEGER*4 JCOUNT
             C*
                   REAL*4 T0,SECNDS
             C*
                   COMMON /SPECTR/ JCOUNT(MAXCHN)
                   COMMON /SPDESC/ NXCHAN,NYCHAN,IDATIM(9),NRUN,ISPILL,ITARG,
                  +                ITEXT(18),LABEL
                   COMMON /OPPARS/ MDEV,SPESPC,PARSPC,MXITER,LBFIX,INREG,LIXLO,LIXHI,
                  +                LIYLO,LIYHI,IYPWR,LIFLO,LIFHI,IPEAK,LBACK,LPEAK,
                  +                LPEBA,NCOEFF,COEFF(MAXCO),ITERA,NFREE,
                  +                PEAK(NPPARM,MAXPK),IFLAG(NPPARM,MAXPK),
                  +                INDEX(NPPARM,MAXPK),ISELEC,IGNOR(MAXPK),NREG,
                  +                KREGLO(MAXREG),KREGHI(MAXREG),IMODE,INTCOR,NCALF,
                  +                CALFAC(MXCALF),SIGMA(MAXPK),GAMMA(MAXPK),CALSPC,
                  +                FILPRT,ENESPC
                   COMMON /OPPART/ IOPARS
                   COMMON /RESULT/ CHINEW,CHIOLD,FVBACK,FVPEAK
                   COMMON /ERRCOM/ IERROR
                   COMMON /PARAM/  ILINK(NPPARM,MXLINK),NPAR,NDIMM,KUSE,KSKIP,
                  +                TOLO(6),TOHI(6)
                   COMMON /ARRAYS/ VMAT(MAXPAR*(MAXPAR+1)/2),DERIV(MAXPAR),
                  +                RVEC(MAXPAR),HDERIV(MAXPAR)
                   COMMON /CONST/  PICON,EPSILN,SQRTPI,SRLN2
             C*
                   LOGICAL FILOPN
                   COMMON /GLOBAL/ IOP,T0,IFIT,FILOPN
             C*
                   DATA IOPARS /0/
             C*
```

```
PDP-11 FORTRAN-77 V5.0-0          09:36:47      26-Sep-86             Page 2
FITSPR.FTN;1                 /F77/OP/TR:BLOCKS/WR

      C*
        1020 CALL MNERAS
             CALL VJZERO (JCOUNT,MAXCHN)
             CALL ASSIGN (LUNSPE,SPESPC)
        1025 READ (LUNSPE,ERR=2000,END=2000) NXCHAN,NYCHAN,IDATIM,
            +                                NRUN,ISPILL,ITARG,ITEXT,LABEL
             IF (LABEL.NE.ISELEC) THEN
                DO 1030 J=1,NYCHAN,1
                   READ (LUNSPE,ERR=2000,END=2000) (JCOUNT(I),I=1,NXCHAN)
        1030    CONTINUE
                GOTO 1025
             ENDIF
      C*
      C*     Correct spectrum found
      C*
             IF (NYCHAN.NE.1) THEN
                CALL DLOUT ('0Can work only with one dimensional spectra')
                NXCHAN = 0
             ELSE
                READ (LUNSPE,ERR=2000,END=2000) (JCOUNT(J),J=1,NXCHAN)
      C*
                WRITE (5,1035) LABEL,IDATIM,NRUN,ISPILL,ITARG,ITEXT,NXCHAN
        1035    FORMAT ('0','Spectrum with label',I5,/,
            +           ' ',9A2,' Run',I4,' Spill',I3,' Target',I3,/
            +           ' ',18A2,3X,'No. of channels: ',I5)
             ENDIF
             CLOSE (UNIT=LUNSPE)
             CALL DLKEY
             GOTO 9000
      C*
      C*     Disk i/o error
      C*
        2000 CALL DLOUT ('0No such spectrum')
             NXCHAN = 0
             CLOSE (UNIT=LUNSPE)
             CALL DLKEY
      C*
      C*     Return to calling program
      C*
        9000 RETURN
      C*     ******
             END
```

```
PDP-11 FORTRAN-77 V5.0-0        09:42:18    26-Sep-86          Page 1
FITSFP.FTN;1              /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE FITSFP
                C*  ****************
                C*
                C*  This subroutine saves and parameters for plotting
                C*  on a big machine.
                C*
                C*
                C*  Ver. 1.0/08-Feb-84(TK)  Original version
                C*
0002            INCLUDE 'FIT.COM'
0003 *          IMPLICIT REAL*8 (A-H,O-Z)
     *  C*
     *  C*  Logical unit numbers used:
     *  C*
     *  C*  LUNSPE: Lun to read spectrum
     *  C*  LUNPAR: Lun for PARFIL
     *  C*  LUNPRT: Lun for print output
     *  C*  LUNCAL: Lun for calibration input, and energy table
     *  C*  LUNTRM: Lun for terminal i/o (incl. DLPACK, MNPACK)
     *  C*
0004 *          PARAMETER LUNSPE=1, LUNCAL=2, LUNPRT=3, LUNPAR=4, LUNTRM = 5
     *  C*
0005 *          PARAMETER MAXCHN=4096, MAXPK=30, MAXCO=6, MAXREG=7, MAXPAR=40,
     *    +               NPPARM=6, MXCALF=3, MPOINT=500, MXLINK=20
     *  C*
0006 *          BYTE SPESPC(30),PARSPC(30),CALSPC(30),FILPRT(30),ENESPC(30)
     *  C*
0007 *          CHARACTER*20 HDGLIN
     *  C*
0008 *          INTEGER*4 JCOUNT
     *  C*
0009 *          REAL*4 T0,SECNDS
     *  C*
0010 *          COMMON /SPECTR/ JCOUNT(MAXCHN)
0011 *          COMMON /SPDESC/ NXCHAN,NYCHAN,IDATIM(9),NRUN,ISPILL,ITARG,
     *    +                     ITEXT(18),LABEL
0012 *          COMMON /OPPARS/ MDEV,SPESPC,PARSPC,MXITER,LBFIX,INREG,LIXLO,LIXHI,
     *    +                     LIYLO,LIYHI,IYPWR,LIFLO,LIFHI,IPEAK,LBACK,LPEAK,
     *    +                     LPEBA,NCOEFF,COEFF(MAXCO),ITERA,NFREE,
     *    +                     PEAK(NPPARM,MAXPK),IFLAG(NPPARM,MAXPK),
     *    +                     INDEX(NPPARM,MAXPK),ISELEC,IGNOR(MAXPK),NREG,
     *    +                     KREGLO(MAXREG),KREGHI(MAXREG),IMODE,INTCOR,NCALF,
     *    +                     CALFAC(MXCALF),SIGMA(MAXPK),GAMMA(MAXPK),CALSPC,
     *    +                     FILPRT,ENESPC
0013 *          COMMON /OPPART/ IOPARS
0014 *          COMMON /RESULT/ CHINEW,CHIOLD,FVBACK,FVPEAK
0015 *          COMMON /ERRCOM/ IERROR
0016 *          COMMON /PARAM/  ILINK(NPPARM,MXLINK),NPAR,NDIMM,KUSE,KSKIP,
     *    +                     TOLO(6),TOHI(6)
0017 *          COMMON /ARRAYS/ VMAT(MAXPAR*(MAXPAR+1)/2),DERIV(MAXPAR),
     *    +                     RVEC(MAXPAR),HDERIV(MAXPAR)
0018 *          COMMON /CONST/  PICON,EPSILN,SQRTPI,SRLN2
     *  C*
0019 *          LOGICAL FILOPN
0020 *          COMMON /GLOBAL/ IOP,T0,IFIT,FILOPN


PDP-11 FORTRAN-77 V5.0-0        09:42:18    26-Sep-86          Page 2
FITSFP.FTN;1              /F77/OP/TR:BLOCKS/WR

     *  C*
0021 *          DATA IOPARS /0/
                C*
0022            BYTE SAVFIL(30)
                C*
                C*  Ask for file
                C*
0023            CALL MNERAS
0024            CALL GTXT ('Specify file: ',SAVFIL,30,LEN)
0025            CALL ASSIGN (LUNSPE,SAVFIL)
                C*
                C*  Save some general information
                C*
0026            WRITE (LUNSPE,1000) LIXLO,LIXHI,LIYLO,LIYHI,IYPWR,LIFLO,LIFHI,
                +                   IMODE,MAXREG,NREG,MAXCO,NCOEFF
0027       1000 FORMAT (8I7,4IG)
0028            WRITE (LUNSPE,1005) KREGLO,KREGHI
0029       1005 FORMAT (14I5)
                C*
                C*  Save background
                C*
0030            WRITE (LUNSPE,1010) COEFF
0031       1010 FORMAT (4(1PE20.13))
                C*
                C*  Save peak parameters
                C*
0032            DO 1020 J=1,MAXPK,1
0033              IF (IGNOR(J) NE.1) THEN
0034                WRITE (LUNSPE,1015) J,(PEAK(K,J),K=1,NPPARM)
0035       1015     FORMAT (I2.6(1X,1PE12.6))
0036              ENDIF
0037       1020 CONTINUE
                C*
                C*  Write a 'termination record' and close output file
                C*
0038            WRITE (LUNSPE,1025)
0039       1025 FORMAT ('-1',78X)
0040            CLOSE (UNIT=LUNSPE)
                C*
                C*  Return to calling program
                C*
0041            RETURN
                C*  ******
0042            END
```

```
PDP-11 FORTRAN-77 V5.0-0        09:39:17    26-Sep-86          Page 1
FITST1.FTN;1              /F77/OP/TR:BLOCKS/WR

0001          SUBROUTINE FITST1
          C*  ++++++++++++++++
          C*
          C*
          C*    This subroutine changes the fit parameters if neccessary
          C*
          C*
          C*    Ver. 1.0  25-Feb-84   (DR)  New menu structure
          C*
          C*
0002          INCLUDE 'FIT.COM'
0003      *   IMPLICIT REAL*8 (A-H,O-Z)
          C*
          C*    Logical unit numbers used:
          C*
          C*    LUNSPE: Lun to read spectrum
          C*    LUNPAR: Lun for PARFIL
          C*    LUNPRT: Lun for print output
          C*    LUNCAL: Lun for calibration input, and energy table
          C*    LUNTRM: Lun for terminal i/o (incl. DLPACK, MNPACK)
          C*
0004      *   PARAMETER LUNSPE=1, LUNCAL=2, LUNPRT=3, LUNPAR=4, LUNTRM = 5
          C*
0005      *   PARAMETER MAXCHN=4096, MAXPK=30, MAXCO=6, MAXREG=7, MAXPAR=40,
                     NPPARM=6, MXCALF=3, MPOINT=500, MXLINK=20
          C*
0006      *   BYTE SPESPC(30),PARSPC(30),CALSPC(30),FILPRT(30),ENESPC(30)
          C*
0007      *   CHARACTER*20 HDGLIN
          C*
0008      *   INTEGER*4 JCOUNT
          C*
0009      *   REAL*4 T0,SECNDS
          C*
0010      *   COMMON /SPECTR/ JCOUNT(MAXCHN)
0011      *   COMMON /SPDESC/ NXCHAN,NYCHAN,IDATIM(9),NRUN,ISPILL,ITARG,
                     ITEXT(18),LABEL
0012      *   COMMON /OPPARS/ MDEV,SPESPC,PARSPC,MXITER,LBFIX,INREG,LIXLO,LIXHI,
                     LIYLO,LIYHI,IYPWR,LIFLO,LIFHI,IPEAK,LBACK,LPEAK,
                     LPEBA,NCOEFF,COEFF(MAXCO),ITERA,NFREE,
                     PEAK(NPPARM,MAXPK),IFLAG(NPPARM,MAXPK),
                     INDEX(NPPARM,MAXPK),ISELEC,IGNOR(MAXPK),NREG,
                     KREGLO(MAXREG),KREGHI(MAXREG),IMODE,INTCOR,NCALF,
                     CALFAC(MXCALF),SIGMA(MAXPK),GAMMA(MAXPK),CALSPC,
                     FILPRT,ENESPC
0013      *   COMMON /OPPART/ IOPARS
0014      *   COMMON /RESULT/ CHINEW,CHIOLD,FVBACK,FVPEAK
0015      *   COMMON /ERRCOM/ IERROR
0016      *   COMMON /PARAM/  ILINK(NPPARM,MXLINK),NPAR,NDIMM,KUSE,KSKIP,
                     TOLD(6),TOHI(6)
0017      *   COMMON /ARRAYS/ VMAT(MAXPAR*(MAXPAR+1)/2),DERIV(MAXPAR),
                     RVEC(MAXPAR),HDERIV(MAXPAR)
0018      *   COMMON /CONST/  PICON,EPSILN,SQRTPI,SRLN2
          C*
0019      *   LOGICAL FILOPN


PDP-11 FORTRAN-77 V5.0-0        09:39:17    26-Sep-86          Page 2
FITST1.FTN;1              /F77/OP/TR:BLOCKS/WR

0020      *   COMMON /GLOBAL/ IOP,T0,IFIT,FILOPN
          C*
0021      *   DATA IOPARS /0/
          C*
0022          COMMON /BUFMEN/ MBUFF(600)
          C*
0023          CHARACTER*16 KTEXT(6)
          C*
0024          DATA KTEXT /'Position is     ','Gauss-FWHM is   ',
                     'Heigth is       ','Low tail is     ',
                     'High tail is    ','Lorentz-FWHM is '/
          C**
0025          INTCOR = 0
          C*
          C*    Display menu to ease changings
          C*
0026       1000 CALL MNCLR
0027          CALL MNBUFF (MBUFF,600)
0028          CALL MNHEAD ('Set parameters of peaks')
0029          CALL MNDEC  ( 1,'Peak Nr ',IPEAK,IPEAK,1,MAXPK)
0030          CALL MNOPT  ( 3,'Set position')
0031          CALL MNOPT  ( 4,'Set Gauss-FWHM')
0032          CALL MNOPT  ( 5,'Set height')
0033          CALL MNOPT  ( 6,'Set low tail')
0034          CALL MNOPT  ( 7,'Set high tail')
0035          CALL MNOPT  ( 8,'Set Lorentz-FWHM')
0036          CALL MNDEC (10,'Position:      -1=fix, 0=flt, >0=group-id      ',
                     IFLAG(1,IPEAK),IFLAG(1,IPEAK),-1,MXLINK)
0037          CALL MNDEC (11,'FWHM-Gauss:    -1=fix, 0=flt, >0=group-id      ',
                     IFLAG(2,IPEAK),IFLAG(2,IPEAK),-1,MXLINK)
0038          CALL MNDEC (12,'Height:        -1=fix, 0=flt, >0=group-id      ',
                     IFLAG(3,IPEAK),IFLAG(3,IPEAK),-1,MXLINK)
0039          CALL MNDEC (13,'Low tail:      -2=no tail, -1=exp tail, 0=no tail',
                     IFLAG(4,IPEAK),IFLAG(4,IPEAK),-2,MXLINK)
0040          CALL MNDEC (14,'High tail:     -2=no tail, -1=exp tail, 0=no tail',
                     IFLAG(5,IPEAK),IFLAG(5,IPEAK),-2,MXLINK)
0041          CALL MNDEC (15,'FWHM-Lorentz: -1=fix, 0=flt, >0=group-id      ',
                     IFLAG(6,IPEAK),IFLAG(6,IPEAK),-2,MXLINK)
0042          CALL MNYEND (18,'Ignore this peak (Do not fit)',
                     IGNOR(IPEAK),IGNOR(IPEAK),-2,MXLINK)
          C*
0043          CALL MNDISP
0044          CALL MNIN (IOP)
          C*
0045          IF (IOP .LT. 0)  GOTO 9000
          C*
          C*    The big option switch
          C*
0046          GOTO (1000, 1000, 1005, 1005, 1005, 1005, 1005, 1005,
                     1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000,
                     1000, 1000), IOP
0047          GOTO 1000
          C*
          C*    Option C(3) - H(8): Set peak parameters
          C*


PDP-11 FORTRAN-77 V5.0-0        09:39:17    26-Sep-86          Page 3
FITST1.FTN;1              /F77/OP/TR:BLOCKS/WR

0048       1005 IOP = IOP - 2
0049          CALL MNERAS
0050          WRITE (5,1010) KTEXT(IOP),PEAK(IOP,IPEAK)
0051       1010 FORMAT ('0',A16,F10.3,/,' ',' Enter new value')
0052          READ (5,*) PEAK(IOP,IPEAK)
0053          IF (IOP .EQ. 2)  SIGMA(IPEAK) = PEAK(2,IPEAK) / SRLN2
0054          IF (IOP .EQ. 6)  GAMMA(IPEAK) = PEAK(6,IPEAK) / 2 D0
0055          IF (PEAK(4,IPEAK) .GT. 0.D0)  PEAK(4,IPEAK) = -PEAK(4,IPEAK)
0056          IF (PEAK(5,IPEAK) .LT. 0.D0)  PEAK(5,IPEAK) = -PEAK(5,IPEAK)
0057          GOTO 1000
          C*
          C*    Return to calling program
          C*
0058       9000 RETURN
          C*   ------
0059          END
```

```
PDP-11 FORTRAN-77 V5.0-0        09:39:35    26-Sep-86          Page 1
FITST2.FTN;1              /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE FITST2
                C*  ****************
                C*
                C*
                C*  This subroutine changes the flags globally
                C*
                C*
                C*  Ver. 1.0  03-May-84   (DR)  Original version
                C*
                C*
0002            INCLUDE 'FIT.COM'
0003  *         IMPLICIT REAL*8 (A-H,O-Z)
      *   C*
      *   C*    Logical unit numbers used:
      *   C*
      *   C*    LUNSPC: Lun to read spectrum
      *   C*    LUNPAR: Lun for PARFIL
      *   C*    LUNPRT: Lun for print output
      *   C*    LUNCAL: Lun for calibration input, and energy table
      *   C*    LUNTRM: Lun for terminal I/o (incl. DLPACK, MNPACK)
      *   C*
0004  *         PARAMETER LUNSPE=1, LUNCAL=2, LUNPRT=3, LUNPAR=4, LUNTRM = 5
      *   C*
0005  *         PARAMETER MAXCHN=4096, MAXPK=30, MAXCO=6, MAXREG=7, MAXPAR=40,
      *        +          NPPARM=6, MXCALF=3, MPOINT=500, MXLINK=20
      *   C*
0006  *         BYTE SPESPC(30),PARSPC(30),CALSPC(30),FILPRT(30),ENESPC(30)
      *   C*
0007  *         CHARACTER*20 HDGLIN
      *   C*
0008  *         INTEGER*4 JCOUNT
      *   C*
0009  *         REAL*4 T0,SECNDS
      *   C*
0010  *         COMMON /SPECTR/ JCOUNT(MAXCHN)
0011  *         COMMON /SPDESC/ NXCHAN,NYCHAN,IDATIM(9),NRUN,ISPILL,ITARG,
      *        +                ITEXT(18),LABEL
0012  *         COMMON /OPPARS/ MDEV,SPESPC,PARSPC,MXITER,LBFIX,INREG,LIXLO,LIXHI,
      *        +                LIYLO,LIYHI,IYPWR,LIFLO,LIFHI,IPEAK,LBACK,LPEAK,
      *        +                LPEBA,NCOEFF,COEFF(MAXCO),ITERA,NFREE,
      *        +                PEAK(NPPARM,MAXPK),IFLAG(NPPARM,MAXPK),
      *        +                INDEX(NPPARM,MAXPK),ISELEC,IGNOR(MAXPK),NREG,
      *        +                KREGLO(MAXREG),KREGHI(MAXREG),IMODE,INTCOR,NCALF,
      *        +                CALFAC(MXCALF),SIGMA(MAXPK),GAMMA(MAXPK),CALSPC,
      *        +                FILPRT,ENESPC
0013  *         COMMON /OPPART/ IOPARS
0014  *         COMMON /RESULT/ CHINEW,CHIOLD,FVBACK,FVPEAK
0015  *         COMMON /ERRCOM/ IERROR
0016  *         COMMON /PARAM/  ILINK(NPPARM,MXLINK),NPAR,NDIMM,KUSE,KSKIP,
      *        +                TOLO(6),TOHI(6)
0017  *         COMMON /ARRAYS/ VMAT(MAXPAR*(MAXPAR+1)/2),DERIV(MAXPAR),
      *        +                RVEC(MAXPAR),HDERIV(MAXPAR)
0018  *         COMMON /CONST/  PICON,EPSILN,SQRTPI,SRLN2
      *   C*
0019  *         LOGICAL FILOPN

PDP-11 FORTRAN-77 V5.0-0        09:39:35    26-Sep-86          Page 2
FITST2.FTN;1              /F77/OP/TR:BLOCKS/WR

0020  *         COMMON /GLOBAL/ IOP,T0,IFIT,FILOPN
      *   C*
0021  *         DATA IOPARS /0/
      *   C*
0022            LOGICAL LLINE
      *   C*
0023            CHARACTER*12 KTEXT(8)
      *   C*
0024            DATA KTEXT /'Position    ','Gauss-FWHM  ',
                +           'Height      ','Low tail    ',
                +           'High tail   ','Lorentz-FWHM',
                +           '            ','Ignor - flag'/
      *   C*=
      *   C*    Display possible flags
      *   C*
0025     1000   CALL DIPAGE
0026            CALL DLOUT (' 1: Position:      -1=fix, 0=fit, >0=group-id')
0027            CALL DLOUT (' 2: FWHM-Gauss:    -1=fix, 0=fit, >0=group-id')
0028            CALL DLOUT (' 3: Height         -1=fix, 0=fit, >0=group-id')
0029            CALL DLOUT (' 4  Low tail:      -2.0=no tail, -1=exp tail')
0030            CALL DLOUT (' 5: High tail:     -2.0=no tail, -1=exp tail')
0031            CALL DLOUT (' 6: FWHM-Lorentz: -1=fix, 0=fit, >0=group-id')
0032            CALL DLOUT (' ')
0033            CALL DLOUT (' 8: Ignor - flag : YES / NO')
0034            CALL DLOUT (' ')
0035            CALL DLOUT ('Which flag do you want to set')
0036            CALL DLDEC (' (0 = No setting wanted)',KFLAG,KFLAG,0,8)
0037            IF (KFLAG .EQ. 0)  GOTO 9000
0038            IF (KFLAG .GT. 6)  GOTO 1025
      *   C*

0039            LLINE = .TRUE.
0040            DO 1020 I=1,MAXPK,1
0041               IF (LLINE)  THEN
0042     1010         ILINE = 0
0043                  CALL DIPAGE
0044                  WRITE (5,1015) KTEXT(KFLAG)
0045     1015         FORMAT (' Set ',A12,' flag :')
0046                  ILINE = 1
0047                  LLINE = .FALSE.
0048               ENDIF
0049               IF (IGNOR(I) .EQ. 1)  GOTO 1020
0050               ILINE = ILINE + 2
0051               CALL DLOUT (' Peak Nr.',I)
0052               CALL DLDEC (' to ',IFLAG(KFLAG,I),IFLAG(KFLAG,I),-2,MXLINK)
0053               IF (ILINE .GT. 28)  THEN
0054                  LLINE = .TRUE.
0055               ENDIF
0056     1020   CONTINUE
0057            GOTO 1000
      *   C*
      *   C*    Set ignor-flag
      *   C*
0058     1025 IF (KFLAG .GT. 8)  GOTO 1000
0059            LLINE = .TRUE.
0060            DO 1040 I=1,MAXPK,1

PDP-11 FORTRAN-77 V5.0-0        09:39:35    26-Sep-86          Page 3
FITST2.FTN;1              /F77/OP/TR:BLOCKS/WR

0061               IF (LLINE)  THEN
0062     1030         ILINE = 0
0063                  CALL DIPAGE
0064                  WRITE (5,1035) KTEXT(KFLAG)
0065     1035         FORMAT (' Set ',A12)
0066                  ILINE = 1
0067                  LLINE = .FALSE.
0068               ENDIF
0069               ILINE = ILINE + 2
0070               CALL DLOUT (' Peak Nr.',I)
0071               CALL DLYENO (' to ',IGNOR(I),IGNOR(I),-2,MXLINK)
0072               IF (ILINE .GT. 28)  THEN
0073                  LLINE = .TRUE.
0074               ENDIF
0075     1040   CONTINUE
0076            GOTO 1000
      *   C*
      *   C*    Return to calling program
      *   C*
0077     9000   RETURN
      *   C*    ******
0078            END
```

```
PDP-11 FORTRAN-77 V5.0-0        09:40:17    26-Sep-86          Page 1
PEKDEF.FTN;1              /F77/OP/TR:BLOCKS/WR

0001          SUBROUTINE PEKDEF (IPK)
          C*  **********************
          C*
          C*  This subroutine sets default values for a selected peak
          C*
          C*
          C*  Ver. 1.0/22-Nov-83(TK)  Original version
          C*  Ver. 1.1/17-Jan-84(TK)  Better handling of background under peak
          C*
0002          INCLUDE 'FIT.COM'
0003 *        IMPLICIT REAL*8 (A-H,O-Z)
     *    C*
     *    C*  Logical unit numbers used:
     *    C*
     *    C*  LUNSPE: Lun to read spectrum
     *    C*  LUNPAR: Lun for PARFIL
     *    C*  LUNPRT: Lun for print output
     *    C*  LUNCAL: Lun for calibration input, and energy table
     *    C*  LUNTRM: Lun for terminal I/O (incl. DLPACK, MNPACK)
     *    C*
0004 *        PARAMETER LUNSPE=1, LUNCAL=2, LUNPRT=3, LUNPAR=4, LUNTRM = 5
     *    C*
0005 *        PARAMETER MAXCHN=4096, MAXPK=30, MAXCO=6, MAXREG=7, MAXPAR=40,
     *   +            NPPARM=6, MXCALF=3, MPOINT=500, MXLINK=20
     *    C*
0006 *        BYTE SPESPC(30),PARSPC(30),CALSPC(30),FILPRT(30),ENESPC(30)
     *    C*
0007 *        CHARACTER*20 HDGLIN
     *    C*
0008 *        INTEGER*4 JCOUNT
     *    C*
0009 *        REAL*4 T0,SECNDS
     *    C*
0010 *        COMMON /SPECTR/ JCOUNT(MAXCHN)
0011 *        COMMON /SPDESC/ NXCHAN,NYCHAN,IDATIM(9),NRUN,ISPILL,ITARG,
     *   +                    ITEXT(10),LABEL
0012 *        COMMON /OPPARS/ MDEV,SPESPC,PARSPC,MXITER,LBFIX,INREG,LIXLO,LIXHI,
     *   +                    LIYLO,LIYHI,IYPWR,LIFLO,LIFHI,IPEAK,LBACK,LPEAK,
     *   +                    LPEDA,NCOEFF,COEFF(MAXCO),ITERA,NFREE,
     *   +                    PEAK(NPPARM,MAXPK),IFLAG(NPPARM,MAXPK),
     *   +                    INDEX(NPPARM,MAXPK),ISELEC,IGNOR(MAXPK),NREG,
     *   +                    KREGLO(MAXREG),KREGHI(MAXREG),IMODE,INTCOR,NCALF,
     *   +                    CALFAC(MXCALF),SIGMA(MAXPK),GAMMA(MAXPK),CALSPC,
     *   +                    FILPRT,ENESPC
0013 *        COMMON /OPPART/ IOPARS
0014 *        COMMON /RESULT/ CHINEW,CHIOLD,FVBACK,FVPEAK
0015 *        COMMON /ERRCOM/ IERROR
0016 *        COMMON /PARAM/  ILINK(NPPARM,MXLINK),NPAR,NDIMM,KUSE,KSKIP,
     *   +                    IOLO(6),IOHI(6)
0017 *        COMMON /ARRAYS/ VMAT(MAXPAR*(MAXPAR+1)/2),DERIV(MAXPAR),
     *   +                    RVEC(MAXPAR),HDERIV(MAXPAR)
0018 *        COMMON /CONST/  PICON,EPSILN,SQRTPI,SRLN2
     *    C*
0019 *        LOGICAL FILOPN
0020 *        COMMON /GLOBAL/ IOP,T0,IFIT,FILOPN

PDP-11 FORTRAN-77 V5.0-0        09:40:17    26-Sep-86          Page 2
PEKDEF.FTN;1              /F77/OP/TR:BLOCKS/WR

     *    C*
0021 *        DATA IOPARS /0/
          C*
0022          INTEGER*4 J4
          C**
          C*  First do some adjustement of the position (parameter 1)
          C*
0023          IF (IFLAG(1,IPK) .EQ. 0)  THEN
0024             I = INT(PEAK(1,IPK)+0.5D0)
0025             J4 = JCOUNT(I)
0026             IF (JCOUNT(I-1) .GT. J4)  THEN
0027                J4 = JCOUNT(I-1)
0028                PEAK(1,IPK) = DBLE(I-1)
0029             ENDIF
0030             IF (JCOUNT(I+1) .GT. J4)  PEAK(1,IPK) = I+1
0031          ENDIF
          C*
          C*  Find in which fit region the peak is situated
          C*
0032          I = INT(PEAK(1,IPK)+0.5D0)
0033          DO 1000 J=1,NREG,1
0034             IF (KREGLO(J) .GT. I  .OR.  KREGHI(J) .LT. I)  GOTO 1000
0035             ICL = KREGLO(J)
0036             ICH = KREGHI(J)
0037             GOTO 1005
0038  1000    CONTINUE
          C*
          C*  Then set height (parameter 3)
          C*
0039  1005    IF (IFLAG(3,IPK) .EQ. 0  .OR.  PEAK(3,IPK) .LE. 0.0D0)  THEN
0040             PEAK(3,IPK) = JCOUNT(INT(PEAK(1,IPK)+0.5D0))
0041             CALL FITBAC (PEAK(1,IPK),1)
0042             PEAK(3,IPK) = PEAK(3,IPK)-(0.9D0*FVBACK)
0043          ENDIF
          C*
          C*  Now try to estimate a Gauss FWHM (parameter 2)
          C*  1. Find nearest peak
          C*
0044          IF (IFLAG(2,IPK) .EQ. 0  .OR.  PEAK(2,IPK) .LE. 0.0D0)  THEN
0045             I = 0
0046             DIFF = 1.0D7
0047             DO 1010 J=1,MAXPK,1
0048                IF (IGNOR(J) .EQ. 1)  GOTO 1010
0049                IF (J .NE. IPK)  THEN
0050                   IF (ABS(PEAK(1,IPK)-PEAK(1,J)) .LT. DIFF)  THEN
0051                      DIFF = ABS(PEAK(1,IPK)-PEAK(1,J))
0052                   ENDIF
0053                ENDIF
0054  1010       CONTINUE
          C*
          C*  If distance to nearest peak is less then 1.5 channels
          C*  do some corrections
          C*
0055          IF (I .NE. 0  .AND.  DIFF .LT. 1.5D0)  THEN
0056             PEAK(1,IPK) = PEAK(1,IPK)+SIGN(0.75D0,DBLE(IPK-I))

PDP-11 FORTRAN-77 V5.0-0        09:40:17    26-Sep-86          Page 3
PEKDEF.FTN;1              /F77/OP/TR:BLOCKS/WR

0057             PEAK(3,IPK) = JCOUNT(INT(PEAK(1,IPK)+0.5D0))
0058             DIFF = 1.5D0
0059          ENDIF
          C*
          C*  2. Set Gauss-FWHM to a resonable value (straight forward)
          C*
0060          I = INT(PEAK(1,IPK)+0.5D0)
0061          CALL FITBAC (DFLOAT(I),1)
0062          R8 = 0.5D0 * (PEAK(3,IPK)-FVBACK)
0063          IL = I
0064          IH = I
0065  1015    IL = IL-1
0066          IF (IL .GT. ICL)  THEN
0067             CALL FITBAC (DFLOAT(IL),1)
0068             IF ((DBLE(JCOUNT(IL))-FVBACK) .GT. R8)  GOTO 1015
0069          ENDIF
0070  1020    IH = IH+1
0071          IF (IH .LT. ICH)  THEN
0072             CALL FITBAC (DFLOAT(IH),1)
0073             IF ((DBLE(JCOUNT(IH))-FVBACK) .GT. R8)  GOTO 1020
0074          ENDIF
0075          PEAK(2,IPK) = MIN(DBLE(IH-IL),15.0D0)
          C*
          C*  3. Correct first estimate in cases of near peaks
          C*
0076          IF (DIFF .LT. 1.0D5  .AND.  PEAK(2,IPK) .GT. DIFF)
     *       PEAK(2,IPK) = DIFF/2.
0077       ENDIF
0078       IF (PEAK(2,IPK) .LE. 0.0D0)  THEN
0079          CALL DLOUT ('Merde',IPK)
0080          CALL DLKEY
0081          PEAK(2,IPK) = 1.5D0
0082       ENDIF
          C*
          C*  Return to calling program
          C*
0083       RETURN
          C*  ******
0084       END
```

```
PDP-11 FORTRAN-77 V5.0-0        09:34:14    26-Sep-86          Page 1
SETUP.FTN;1                /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE SETUP
          C*    ****************
          C*
          C*    This subroutine sets up the fit program
          C*
          C*
          C*    Ver. 1.0/22-Nov-83(TK)  Original version
          C*
0002            INCLUDE 'FIT.COM'
0003    *       IMPLICIT REAL*8 (A-H,O-Z)
        *  C*
        *  C*    Logical unit numbers used:
        *  C*
        *  C*    LUNSPE: Lun to read spectrum
        *  C*    LUNPAR: Lun for PARFIL
        *  C*    LUNPRT: Lun for print output
        *  C*    LUNCAL: Lun for calibration input, and energy table
        *  C*    LUNTRM: Lun for terminal I/O (incl. DLPACK, MNPACK)
        *  C*
0004    *       PARAMETER LUNSPE=1, LUNCAL=2, LUNPRT=3, LUNPAR=4, LUNTRM = 5
        *  C*
0005    *       PARAMETER MAXCHN=4096, MAXPK=30, MAXCO=6, MAXREG=7, MAXPAR=40,
        *      +          NPPARM=6, MXCALF=3, MPOINT=500, MXLINK=20
        *  C*
0006    *       BYTE SPESPC(30),PARSPC(30),CALSPC(30),FILPRT(30),ENESPC(30)
        *  C*
0007    *       CHARACTER*20 HDGLIN
        *  C*
0008    *       INTEGER*4 JCOUNT
        *  C*
0009    *       REAL*4 TO,SECNDS
        *  C*
0010    *       COMMON /SPECTR/ JCOUNT(MAXCHN)
0011    *       COMMON /SPDESC/ NXCHAN,NYCHAN,IDATIM(9),NRUN,ISPILL,ITARG,
        *      +                ITEXT(18),LABEL
0012    *       COMMON /OPPARS/ MDEV,SPESPC,PARSPC,MXITER,LBFIX,INREG,LIXLO,LIXHI,
        *      +                LIYLO,LIYHI,IYPWR,LIFLO,LIFHI,IPEAK,LBACK,LPEAK,
        *      +                LPEBA,NCOEFF,COEFF(MAXCO),ITERA,NFREE,
        *      +                PEAK(NPPARM,MAXPK),IFLAG(NPPARM,MAXPK),
        *      +                INDEX(NPPARM,MAXPK),ISELEC,IGNOR(MAXPK),NREG,
        *      +                KREGLO(MAXREG),KREGHI(MAXREG),IMODE,INTCOR,NCALF,
        *      +                CALFAC(MXCALF),SIGMA(MAXPK),GAMMA(MAXPK),CALSPC,
        *      +                FILPRT,ENESPC
0013    *       COMMON /OPPART/ IOPARS
0014    *       COMMON /RESULT/ CHINEW,CHIOLD,FVBACK,FVPEAK
0015    *       COMMON /ERRCOM/ IERROR
0016    *       COMMON /PARAM/  ILINK(NPPARM,MXLINK),NPAR,NDIMM,KUSE,KSKIP,
        *      +                TOLO(6),TOHI(6)
0017    *       COMMON /ARRAYS/ VMAT(MAXPAR*(MAXPAR+1)/2),DERIV(MAXPAR),
        *      +                RVEC(MAXPAR),HDERIV(MAXPAR)
0018    *       COMMON /CONST/  PICON,EPSILN,SQRTPI,SRLN2
        *  C*
0019    *       LOGICAL FILOPN
0020    *       COMMON /GLOBAL/ IOP,TO,IFIT,FILOPN
        *  C*


PDP-11 FORTRAN-77 V5.0-0        09:34:14    26-Sep-86          Page 2
SETUP.FTN;1                /F77/OP/TR:BLOCKS/WR

0021    *       DATA IOPARS /0/
          C*
0022            NCALF = 0
0023            CALL PARRD
0024            CALL VJZERO (JCOUNT,MAXCHN)
0025            CALL DIWNDW (105,1023,30,700)
0026            CALL ERRSET (29, .TRUE.,.FALSE.,.TRUE.,.FALSE.)
0027            CALL MNLUN (LUNTRM)
          C*
          C*    Return to calling program
          C*
0028            RETURN
          C*    ******
0029            END
```

```
PDP-11 FORTRAN-77 V5.0-0        13:23:15    25-Feb-85          Page 1
CWERF.FTN;1                /F77/OP/TR:BLOCKS/WR

0001            COMPLEX FUNCTION CWERF(Z)
          C*    ************************
          C*
          C*    This function is used for calculating the complex error function
          C*              w(z)  = exp(-z**2) * erfc(-iz)
          C*    where     erfc(z) = 1. - erf(z)
          C*
          C*    Literature :  Abramowitx, Stegun, chp.7
          C*                  W.Gautschi,   comm.acm 12 (1969) 635
          C*                  K.S.Koelbig,  comm.acm 15 (1972) 465
          C*                  D.M.Wilkinson,    NIM   95 (1971) 259
          C*                  C.J.Batty et al,  NIM 137 (1976) 179
          C*
          C*
          C*    Ver. 1.0  06-Feb-84  (DR)  Original version
          C*    Ver. 1.1  10-Mar-84  (DR)  Bug removed
          C*
          C**
0002            IMPLICIT REAL*8 (A-H,O-Z)
          C*
0003            COMPLEX Z
          C*
0004            LOGICAL B
          C*
0005            REAL*8 LAMBDA
          C*
0006            DATA XLIMIT /5.33D0/, YLIMIT /4.29D0/,
               +     DCONST /1.12837916709551D0/
          C**   .
          C*
          C*
0007            XX = DBLE(REAL (Z))
0008            YY = DBLE(AIMAG (Z))
0009            X = ABS(XX)
0010            Y = ABS(YY)
0011            IF (Y .LT. YLIMIT  .AND.  X .LT. XLIMIT)  THEN
0012               S = (1.D0 - Y / YLIMIT) * SQRT (1.D0 - X / XLIMIT * X / XLIMIT;
0013               H = 1.6D0 * S
0014               H2 = 2.D0 * H
0015               NC = 6 + INT (23.D0 * S)
0016               NU = 9 + INT (21.D0 * S)
0017               LAMBDA = H2**NC
0018               B = LAMBDA .EQ. 0.D0
0019            ELSE
0020               H = 0.D0
0021               NC = 0
0022               NU = 8
0023               LAMBDA = 0.D0
0024               B = .TRUE.
0025            ENDIF
0026            R1 = 0.D0
0027            R2 = 0.D0
0028            S1 = 0.D0
0029            S2 = 0.D0
0030            N = NU + 1

PDP-11 FORTRAN-77 V5.0-0        13:23:15    25-Feb-85          Page 2
CWERF.FTN;1                /F77/OP/TR:BLOCKS/WR

0031         1000 N = N - 1
0032            FN = DFLOAT(N + 1)
0033            T1 = Y + H + FN * R1
0034            T2 = X - FN * R2
0035            C = 0.5D0 / (T1 * T1 + T2 * T2)
0036            R1 = C * T1
0037            R2 = C * T2
0038            IF (H .GT. 0.D0  .AND.  N .LE. NC)  THEN
0039               T1 = LAMBDA * S1
0040               S1 = R1 * T1 - R2 * S2
0041               S2 = R2 * T1 + R1 * S2
0042               LAMBDA = LAMBDA / H2
0043            ENDIF
0044            IF (N .GT. 0)  GOTO 1000
0045            IF (B)  THEN
0046               RS1 = R1
0047               RS2 = R2
0048            ELSE
0049               RS1 = S1
0050               RS2 = S2
0051            ENDIF
0052            RS1 = DCONST * RS1
0053            IF (Y .EQ. 0.D0)  RS1 = EXP (- X * X)
0054            CWERF = CMPLX (RS1,DCONST * RS2)
0055            IF (YY .LT. 0.D0)  THEN
0056               CWERF = 2.D0 * CEXP (- CMPLX (X,Y) * CMPLX (X,Y)) - CWERF
0057               IF (XX .GT. 0.D0)  CWERF = CONJG (CWERF)
0058            ELSE
0059               IF (XX .LT. 0.D0)  CWERF = CONJG (CWERF)
0060            ENDIF
          C*
          C*    Return to calling program
          C*
0061            RETURN
          C*    ******
0062            END
```

```
PDP-11 FORTRAN-77 V5.0-0        13:23:46    25-Feb-85          Page 1
DATE.FTN;1                 /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE DATE (KDATE)
          C*    ***********************
          C*
          C*    This program gets the actual date
          C*
          C*    Ver 1.0  10-Jan-85  DR       Original Version
          C*
          C*
0002            INTEGER*2 KDATE(5),DATIM(8),CAL1(12),CAL2(12)
          C*
0003            BYTE STRING,BLANK
          C*
0004            CHARACTER*10 FORM
          C*
0005            BYTE FORM1(10)
          C*
0006            DATA CAL1 /'Ja','Fe','Ma','Ap','Ma','Ju',
               +          'Ju','Au','Se','Oc','No','De'/
0007            DATA CAL2 /'n-','b-','r-','r-','y-','n-',
               +          'l-','g-','p-','t-','v-','c-'/
0008            DATA STRING /'-'/, BLANK /' '/
          C*
0009            EQUIVALENCE (FORM,FORM1(1))
          C*
          C*
0010            CALL GETTIM (DATIM)
          C*
0011            WRITE (FORM,1000) DATIM(3),STRING,CAL1(DATIM(2)),
               +                  CAL2(DATIM(2)),DATIM(1),BLANK
0012         1000 FORMAT (I2,A1,A2,A2,I2,A1)
0013            IF (FORM1(1) .EQ. BLANK)  FORM1(1) = '0'
0014            IF (FORM1(8) .EQ. BLANK)  FORM1(8) = '0'
0015            READ (FORM,1005) (KDATE(I),I=1,5)
0016         1005 FORMAT (5A2)
          C*
          C*    Return to calling program
          C*
0017            RETURN
          C*    ******
0018            END
```

```
PDP-11 FORTRAN-77 V5.0-0        13:25:21    25-Feb-85          Page 1
TIME.FTN;1                 /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE TIME (KTIME)
          C*    ***********************
          C*
          C*    This program gets the actual time
          C*
          C*    Ver 1.0  10-Jan-85  DR       Original Version
          C*
          C*
0002            INTEGER*2 KTIME(4),DATIM(8)
          C*
0003            BYTE STRING,BLANK
          C*
0004            CHARACTER*8 FORM
          C*
0005            BYTE FORM1(8)
          C*
0006            DATA STRING /':'/, BLANK /' '/
          C*
0007            EQUIVALENCE (FORM,FORM1(1))
          C*
          C*
0008            CALL GETTIM (DATIM)
          C*
0009            WRITE (FORM,1000) DATIM(4),STRING,DATIM(5),STRING,DATIM(6)
0010         1000 FORMAT (I2,A1,I2,A1,I2)
0011            IF (FORM1(1) .EQ. BLANK)  FORM1(1) = '0'
0012            IF (FORM1(4) .EQ. BLANK)  FORM1(4) = '0'
0013            IF (FORM1(7) .EQ. BLANK)  FORM1(7) = '0'
0014            READ (FORM,1005) (KTIME(I),I=1,4)
0015         1005 FORMAT (4A2)
          C*
          C*    Return to calling program
          C*
0016            RETURN
          C*    ******
0017            END
```

```
PDP-11 FORTRAN-77 V5.0-0        13:25:06    25-Feb-85          Page 1
SPXINV.FTN;1                /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE SPXINV (A,NDIM)
                C* ************************
                C*
                C*    Inversion of a packed symmetric matrix CERN Lib. F106
                C*
                C*
                C*          +1: Everything was ok.
                C*    IERR: -1: Bad matrix
                C*          -2: Invalid dimension parameter (NDIM)
                C*
                C*    Ver. 1.0/16-DEC-79(RG)  Implemeted under RSX-11-M
                C*    Ver. 1.1/11-SEP-83(TK)  Minor modifications
                C*    Ver. 1.2/01-OCT-83(DR)  Minor modifications
                C*    Ver. 1.3/08-MAR-84(TK)  Some corrections
                C*
0002            IMPLICIT REAL*8 (A-H,O-Z)
                C*
0003            PARAMETER MAXDIM = 40
                C*
0004            COMMON /ERRCOM/ IERR
                C*
0005            DIMENSION INDEX(MAXDIM),RK(MAXDIM),A(MAXDIM*(MAXDIM+1)/2)
                C*
0006            DATA  TOL/ 1.D-12/
                C**
                C*    Set up subroutine
                C*
0007            IERR = 1
0008            IF (NDIM .GT. MAXDIM) THEN
0009              IERR = -2
0010              GOTO 9000
0011            ENDIF
0012            NMAX  = NDIM * (NDIM+1) / 2
0013            NDIM1 = NDIM - 1
                C*
0014            DO 1000  I=1,NDIM
0015               INDEX(I) = 1
0016     1000 CONTINUE
                C*
0017            DO 1020  I=1,NDIM
                C*
                C*       Find pivot
                C*
0018               JJ    = 1
0019               MDIM  = NDIM
0020               PIVOT = 0.D0
0021               DO 1005  J=1,NDIM
0022                  IF (INDEX(J) .NE. 0)  THEN
0023                     AJJ = ABS(A(JJ))
0024                     IF (AJJ .GT. PIVOT)  THEN
0025                        PIVOT = AJJ
0026                        K     = J
0027                        KK    = JJ
0028                     ENDIF
0029                  ENDIF

PDP-11 FORTRAN-77 V5.0-0        13:25:06    25-Feb-85          Page 2
SPXINV.FTN;1                /F77/OP/TR:BLOCKS/WR

0030                  JJ    = JJ + MDIM
0031                  MDIM  = MDIM - 1
0032     1005      CONTINUE
0033               IF (PIVOT/ABS(A(1)) .LT. TOL)  THEN
0034                  IERR = -1
0035                  GOTO 9000
0036               ENDIF
0037               INDEX(K) = 0
0038               PIVOT    = - A(KK)
                C*
                C*       Elimination
                C*
0039               NP = NDIM1
0040               JK = K
0041               NM = 1
0042               DO 1015  J=1,NDIM
0043                  IF (J .EQ. K)  THEN
0044                     A(JK) = 1.D0 / PIVOT
0045                     RK(J) = 0.D0
0046                     NM = 0
0047                     NP = 1
0048                  ELSE
0049                     AJK = SIGN (MAX (TOL,ABS(A(JK))),A(JK))
0050                     RK(J) = AJK / PIVOT
0051                     RK(J) = SIGN (MAX (TOL,ABS(RK(J))),RK(J))
0052                     IF (AJK .NE. 0.D0)  THEN
0053                        NPC = NDIM1
0054                        JL  = J
0055                        DO 1010  L=1,J
0056                           A(JL) = A(JL) + AJK * RK(L)
0057                           JL    = JL + NPC
0058                           NPC   = NPC - 1
0059     1010                  CONTINUE
0060                     ENDIF
0061                     A(JK) = RK(J)
0062                  ENDIF
0063               JK = JK+NP
0064               NP = NP-NM
0065     1015    CONTINUE
0066     1020 CONTINUE
                C*
                C*    Change the sign
                C*
0067            DO 1025  I=1,NMAX
0068               A(I) = - A(I)
0069     1025 CONTINUE
                C*
                C*    Return to calling program
                C*
0070     9000 RETURN
                C*    ******
0071            END
```



```
PDP-11 FORTRAN-77 V5.0-0        13:25:02    25-Feb-85          Page 1
PRTHDR.FTN;1                /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE PRTHDR (JMAX,JSUM)
                C* ****************************
                C*
                C*    This subroutine prints the spectrum header on top of a picture
                C*
                C*    Ver  1.0/08-Dec-83(TK)  Original Version
                C*    Ver. 1.1/14-Dec-83(TK)  LABEL implemented
                C*
0002            COMMON /GPDESC/ NXCHAN,NYCHAN,IDATIM(9),
                +                IRUN,ISPILL,ITARG,ITEXT(18),LABEL
                C*
0003            INTEGER*4 JMAX,JSUM
                C**
                C*    Set cursor to homeposition and switch to alpha mode
                C*
0004            CALL DIHOME
0005            CALL DIANU
                C*
                C*    Print header
                C*
0006            WRITE (5,1000) IDATIM,IRUN,ISPILL,ITARG,LABEL,JSUM,JMAX,ITEXT
0007     1000 FORMAT ('+',9A2,',  Run',I5,',  Spill',I3,',  Target',I3,
                +          ',  Label',I4,/
                +          ' ',I11,' = Sum ',I11,' = Maximum',2X,18A2)
                C*
                C*    Return to calling program
                C*
0008            RETURN
                C*    ******
0009            END
```

```
BEEP     MACRO M1200   25-FEB-85 13:23   PAGE 1

                                  .TITLE BEEP
                          ;************************************************************
                          ;
                          ; THIS ROUTINE SENDS A CNTL-G (BEL) TO LUN 5
                          ; CALLING SEQUENCE:   CALL BEEP
                          ;
                          ; VER 1.0/03-OCT-83(TK)   ORIGINAL VERSION
                          ;
                          ;************************************************************
                          ;
                                  .MCALL QIOW$S
                          ;
     000000      007      BEEPY:  .BYTE   007                     ;CNTL-G (THE BEEP)
                                  .EVEN
     000002               IOSTAT: .BLKW   2                       ;THE STATUS OF I/O
                          ;
                          ; ISSUE QIO TO SEND BEEP
                          ;
     000006   010046      BEEP::  MOV     R0,-(SP)                ;SAVE REGISTER
     000010                       QIOW$S #IO.WAL,#5,#5,,#IOSTAT,,<#BEEPY,#1,#0>
     000062   012600              MOV     (SP)+,R0                ;RESTORE REGISTER
     000064   000207              RTS     PC                      ;AND RETURN TO CALLING PROGRAM
              000001              .END
```

```
PDP-11 FORTRAN-77 V5.0-0          13:25:30    25-Feb-85           Page 1
VIZERO.FTN;1              /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE VIZERO (IARRAY,N)
        C*      ****************************
        C*
        C*      This subroutine zeroes the integer*2 array IARRAY of length N
        C*
        C*
        C*      On entry:
        C*
        C*      IARRAY (I*2):  Array to be zeroed
        C*      N (I*2):       Index up to which IARRAY will be zeroed
        C*
        C*
        C*      On exit:
        C*
        C*      IARRAY (I*2):  Array filled with zeroes up to index N
        C*
        C*
        C*      Ver. 1.0/29-FEB-83(TK)   Original version
        C*
0002            INTEGER*2 IARRAY(*)
        C*
        C*      Zero out array
        C*
0003            DO 1000 J=1,N,1
0004                IARRAY(J) = 0
0005       1000 CONTINUE
        C*
        C*      Return to calling program
        C*
0006            RETURN
        C*      ******
0007            END
```

```
PDP-11 FORTRAN-77 V5.0-0          13:25:34    25-Feb-85           Page 1
VJZERO.FTN;1              /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE VJZERO (JARRAY,N)
        C*      ****************************
        C*
        C*      This subroutine zeroes the integer*4 array JARRAY of length N
        C*
        C*
        C*      On entry:
        C*
        C*      JARRAY (I*4):  Array to be zeroed
        C*      N (I*2):       Index up to which JARRAY will be zeroed
        C*
        C*
        C*      On exit:
        C*
        C*      JARRAY (I*4):  Array filled with zeroes up to index N
        C*
        C*
        C*      Ver. 1.0/08-Dec-83(TK)   Original version
        C*
0002            INTEGER*4 JARRAY(*)
        C*
        C*      Zero out array
        C*
0003            DO 1000 J=1,N,1
0004                JARRAY(J) = 0
0005       1000 CONTINUE
        C*
        C*      Return to calling program
        C*
0006            RETURN
        C*      ******
0007            END
```

```
PDP-11 FORTRAN-77 V5.0-0          13:25:27    25-Feb-85           Page 1
VDZERO.FTN;1              /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE VDZERO (DARRAY,LENGTH)
        C*
0002            REAL*8 DARRAY(LENGTH)
0003            DO 1000 J=1,LENGTH,1
0004                DARRAY(J) = 0.0D0
0005       1000 CONTINUE
        C*
0006            RETURN
0007            END
```

```
PDP-11 FORTRAN-77 V5.0-0        12:56:56    25-Feb-85          Page 1
DJCURI.FTN;1                /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE DJCURI (ICHAR,IX,JY)
          C*    ******************************
          C*
          C*    This routine displays a crosshair cursor on the screen,
          C*    gets the input from the keyboard, converts the (absolute)
          C*    screen coordinates to user coordinates and returns them to
          C*    the calling program
          C*
          C*    Calling sequence:   CALL DJCURI (ICHAR,IX,JY)
          C*
          C*    On exit:   ICHAR (I*2):  character typed (left justified)
          C*               IX (I*2):     X coordinate of cursor (user system)
          C*               JY (I*4):     Y coordinate of cursor (user system)
          C*
          C*
          C*    Ver. 1.0/22-Sep-83(TK)  Original version
          C*
0002            INTEGER*4 JYMIN,JYMAX,JYRAN,JUSERX
0003            COMMON /DICOM / IXMIN,IXMAX,IXRAN,JYMIN,JYMAX,JYRAN,
                +               IMODE,IUSERX,JUSERX,KXLO,KXHI,KXRAN,
                +               KYLO,KYHI,KYRAN
          C*
0004            INTEGER*4 JY,JXX
          C**
          C*    Call DICURI to get the absolute screen coordinates
          C*
0005            CALL DICURI (ICHAR,IXX,IYY)
          C*
          C*    Convert X coordinate
          C*
0006            JXX = IXX-KXLO
0007            JXX = JXX*IXRAN
0008            JXX = JXX/KXRAN
0009            IX = JXX+IXMIN
          C*
          C*    Convert Y coordinate
          C*
0010            JY = IYY-KYLO
0011            JY = JY*JYRAN
0012            JY = JY/KYRAN
0013            JY = JY+JYMIN
          C*
          C*    Return to calling program
          C*
0014            RETURN
          C*    ******
0015            END
```

```
PDP-11 FORTRAN-77 V5.0-0        12:57:11    25-Feb-85          Page 1
PREPDI.FTN;1                /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE PREPDI (JCOUNT,IXL,IXH,IYL,IYH,IYPWR,JMIN,JMAX,JSUM)
          C*    ***************************************************************
          C*
          C*    This subroutine prepares the data needed to produce a plot
          C*
          C*    Ver. 0.0/07-SEP-83(TK)  Original version
          C*
0002            INTEGER*4 JMIN,JMAX,JSUM,JCOUNT(*)
          C**
          C*    Check Y-limits and prepare them
          C*
0003            IF (IYL.GE.IYH) THEN
0004               JMAX = 0
0005               JMIN = 999999
0006               DO 1000 J=IXL,IXH,1
0007                  JMIN = MIN(JMIN,JCOUNT(J))
0008                  JMAX = MAX(JMAX,JCOUNT(J))
0009    1000       CONTINUE
0010            ELSE
0011               JMIN = IYL
0012               JMAX = IYH
0013               JMIN=JMIN*10**IYPWR
0014               JMAX=JMAX*10**IYPWR
0015            ENDIF
          C*
          C*    Calculate integrated contents
          C*
0016            JSUM = 0
0017            DO 1005 J=IXL,IXH,1
0018               JSUM = JSUM+JCOUNT(J)
0019    1005    CONTINUE
          C*
          C*    Return to calling program
          C*
0020            RETURN
          C*    ******
0021            END
```

```
PDP-11 FORTRAN-77 V5.0-0        12:57:22    25-Feb-85          Page 1
SDISPL.FTN;1                /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE SDISPL (JCOUNT,IXLO,IXHI,JMIN,JMAX,ITYPE)
          C*    ****************************************************
          C*
          C*    This subroutine displays a spectrum -JCOUNT- from
          C*    channel IXLO to channel IXHI and sets the Y limits
          C*    to JMIN and JMAX. The display type is taken from ITYPE.
          C*
          C*
          C*    Ver. 0.0/08-SEP-83(TK)  Original Version
          C*
0002            INTEGER*4 JCOUNT(*),JMIN,JMAX
          C**
          C*    Define display window, erase screen, draw frame and axis
          C*
0003            CALL DJWNDW (IXLO,IXHI,JMIN,JMAX)
0004            CALL DIPAGE
0005            CALL DIFRAM
0006            CALL DIAXIS
          C*
          C*    Display spectrum of type ITYPE and "leave the picture open"
          C*
0007            CALL DJSPEC (JCOUNT,IXLO,IXHI,ITYPE)
          C*
          C*    Return to calling program
          C*
0008            RETURN
          C*    ******
0009            END
```

```
PDP-11 FORTRAN-77 V5.0-0        12:57:04    25-Feb-85          Page 1
DJSPEC.FTN;1                /F77/OP/TR:BLOCKS/WR

0001            SUBROUTINE DJSPEC (NN,KA,KB,IPLOT)
                C
                C*****************************************************************
                C DISPLAY INTEGER*4 SPECTRUM BETWEEN CHANNELS KA AND KB ACCORDING TO IPLOT
                C
                C IPLOT  = 0 HISTOGRAM PLOT
                C        = 1 ERROR BARS
                C        = 2 SPECTRUM PLOT
                C        = 3 POINT PLOT
                C*****************************************************************
                C
0002            INTEGER*4 NN(1),JY,JJY
                C
                C
0003            JY=NN(KA)
0004            CALL DJMOVA (KA,JY)
0005            DO 200 I=KA,KB
0006               JY=NN(I)
0007               IF (IPLOT .NE. 3) GOTO 100
0008               CALL DJPNTA (I,JY)
0009               GOTO 200
0010    100        IF (IPLOT .NE. 1) GOTO 150
0011               IF (JY .LE. 0) GOTO 150
0012               E=JY
0013               E=SQRT (E)
0014               JJY=JY-E
0015               CALL DJMOVA (I,JJY)
0016               JY=JY+E
0017    150        CALL DJDRWA (I,JY)
0018               IF (IPLOT .NE. 0) GOTO 200
0019               I1=I+1
0020               CALL DJDRWA (I1,JY)
0021    200     CONTINUE
0022            RETURN
0023            END
```

```
DIPACK  MACRO M1200  25-FEB-85 13:11  PAGE 1


                                        .TITLE DIPACK
                                ;
                                ;*****************************************************************
                                ;
                                ; THIS PACKAGE PROVIDES THE BASIC SUPPORT OF A TEKTRONIX 4010 DISPLAY
                                ;
                                ; THE FOLLOWING ENTRIES ARE PROVIDED:
                                ;
                                ; DIINI  - INITIALIZE THE GRAPHICS SOFTWARE
                                ; DIPAGE - ERASE THE SCREEN
                                ; DIHOME - MOVE CURSOR TO HOME POSITION AND CALL DIANU
                                ; DIANU  - SET TERMINAL TO ALPHNUMERIC MODE
                                ; DIVEC  - SET TERMINAL TO GRAPHIC (VECTOR) MODE
                                ; DIPNT  - SET TERMINAL TO POINT MODE
                                ; DITXT  - CALL DIANU AND OUTPUT A TEXT STRING
                                ; DIOUT  - OUTPUT A STRING OF CHARACTERS TERMINATED BY SOME BYTES
                                ; DIGCUR - SET CROSSHAIR CURSOR AND WAIT FOR GRAPHIC INPUT
                                ;
                                ;
                                ; VER. 1.0/17-FEB-82(RG)  ORIGINAL VERSION
                                ; VER. 1.1/21-SEP-83(TK)  IMPLEMENTATION OF DICURI
                                ;
                                ;*****************************************************************
                                ;
                                ;
                                        .GLOBL DIMOVA
                                        .MCALL QIOWSS
                                ;
 000110                         MAXCNT = 72.                    ;MAX. NUMBER OF ENTRIES IN BUFFER
                                ;
                                ;   CODES DEFINING VARIOUS FUNCTIONS OF TEKTRONIX DISPLAY
                                ;
000000  033  014  000  PAGE:   .BYTE 33,14,0                    ;ERASE SCREEN
000003  037  000       ALPHA:  .BYTE 37,0                      ;ALPHANUMERIC MODE
000005  035  000       GRAPH:  .BYTE 35,0                      ;GRAPHIC MODE
000007  033  032  377  GINPUT: .BYTE 33,32,377                 ;GRAPHIC CURSOR (FORCE OUTPUT)
000012  026  000       DUMMY:  .BYTE 26,0                      ;NO FUNCTION
                                .EVEN
000014                  IOSTAT: .BLKW 2
000020  000000          BUFCNT: .WORD 0                         ;NUMBER OF ENTRIES
000022                  BUFFER: .BLKB MAXCNT                    ;BUFFER FOR OUTPUT TO TERMINAL
                                .EVEN
                                ;
                                ;
                                ;-----------------------------------------------------------------
                                ; INIT DISPLAY PACKAGE, ERASE SCREEN, MOVE CURSOR HOME AND SET ALPHA MODE
                                ;
                                ; CALLING SEQUENCE:          CALL DIINI (IDELAY)
                                ;
                                ; ON ENTRY:  IDELAY = DELAY AFTER SCREEN ERASE BEFORE NEW CHARACTERS
                                ;                     ARE ACCEPTED = TERMINAL SPEED IN CHARACTERS/SECOND
                                ;-----------------------------------------------------------------
                                ;

DIPACK  MACRO M1200  25-FEB-85 13:11  PAGE 1-1

000132  017567  000002  000054' DIINI:: MOV   @2(R5),KSPEED      ;GET TERMINAL SPEED (CHARS/SEC)
000140  005067  177654          CLR   BUFCNT                    ;START WITH EMPTY BUFFER
000144  005067  000046'         CLR   KKMODE                    ;ASSUME ALPHA MODE
                                ;
                                ;-----------------------------------------------------------------
                                ; ERASE SCREEN, MOVE CURSOR TO HOME POSITION AND SET ALPHA-MODE
                                ;
                                ; CALLING SEQUENCE:          CALL DIPAGE
                                ;-----------------------------------------------------------------
                                ;
000150  012700  000000' DIPAGE:: MOV   #PAGE,R0                 ;R0 POINTS TO PAGE FUNCTION
000154  004767  000162          JSR   PC,DIOUT                  ;OUTPUT CHARS
000160  016701  000054'         MOV   KSPEED,R1                 ;GET SPEED OF TERMINAL
000164  005201                  INC   R1                        ;AT LEAST ONE DUMMY CHAR
000166  012700  000012' LOOP:   MOV   #DUMMY,R0                 ;LOAD DUMMY FUNCTION
000172  004767  000144          JSR   PC,DIOUT                  ;OUTPUT A SYN CHARACTER AS FILLER
000176  077105                  SOB   R1,LOOP                   ;LOOP COUNT PROPORTIONAL TO SPEED
                                ;
                                ;-----------------------------------------------------------------
                                ; MOVE CURSOR TO UPPER LEFT CORNER AND SET TERMINAL TO ALPHA-MODE
                                ;
                                ; CALLING SEQUENCE:          CALL DIHOME
                                ;-----------------------------------------------------------------
                                ;
000200  012705  000066' DIHOME:: MOV   #ABEAM,R5                 ;POINTER TO BEAM COORDINATES
000204  005067  000050'         CLR   KBEAMX                    ;LEFT CORNER
000210  012767  001377  000052'  MOV   #767.,KBEAMY             ;POSITION OF FIRST CHARACTER
000216  004767  000000G         JSR   PC,DIMOVA                 ;GO HOME!
                                ;
                                ;-----------------------------------------------------------------
                                ; SET TERMINAL TO ALPHANUMERIC MODE
                                ;
                                ; CALLING SEQUENCE:          CALL DIANU
                                ;-----------------------------------------------------------------
                                ;
000222  005067  000046' DIANU:: CLR   KKMODE                    ;FLAG ALPHA MODE
000226  012700  000003'         MOV   #ALPHA,R0                 ;LOAD FUNCTION
000232  000427                  BR    DOIT
                                ;
                                ;-----------------------------------------------------------------
                                ; SET TERMINAL TO GRAPHIC (VECTOR) MODE
                                ;
                                ; CALLING SEQUENCE:          CALL DIVEC
                                ;-----------------------------------------------------------------
                                ;
000234  012767  000001  000046' DIVEC:: MOV   #1,KKMODE         ;FLAG GRAPHIC MODE
000242  012700  000005'         MOV   #GRAPH,R0                 ;LOAD FUNCTION
000246  000405                  BR    CLRCHA
                                ;
                                ;-----------------------------------------------------------------
                                ; SET TERMINAL TO POINT MODE
                                ;
                                ; CALLING SEQUENCE:          CALL DIPNT

DIPACK  MACRO M1200  25-FEB-85 13:11  PAGE 1-2

                                ;-----------------------------------------------------------------
                                ;
000250  012767  000002  000046' DIPNT:: MOV   #2,KKMODE         ;FLAG POINT MODE
000256  012700  000003'         MOV   #ALPHA,R0                 ;LOAD FUNCTION
000262  012767  177777  000056' CLRCHA: MOV   #-1,KPCHAR        ;FLAG UNDEFINED OLD COORDINATES
000270  012767  177777  000060'         MOV   #-1,KPCHAR+2
000276  012767  177777  000062'         MOV   #-1,KPCHAR+4
000304  012767  177777  000064'         MOV   #-1,KPCHAR+6
000312  004767  000024          DOIT:   JSR   PC,DIOUT          ;OUTPUT CHARS
000316  000207                  RTS   PC
                                ;
                                ;-----------------------------------------------------------------
                                ; SET ALPHA MODE AND OUTPUT STRING TERMINATED BY A ZERO BYTE
                                ;
                                ; CALLING SEQUENCE:          CALL DITXT (STRING)
                                ;-----------------------------------------------------------------
                                ;
000320  004767  177676  DITXT:: JSR   PC,DIANU                  ;SET TERMINAL TO ALPHA MODE
000324  016500  000002          MOV   2(R5),R0                  ;GET ADDRESS OF STRING
000330  004767  000006          JSR   PC,DIOUT
000334  004767  177662          JSR   PC,DIANU                  ;FORCE OUTPUT OF LINE
000340  000207                  RTS   PC
```

```
DIPACK  MACRO M1200  25-FEB-85 13:11  PAGE 2


                                        ;
                                        ;-----------------------------------------------------------------
                                        ; FILL STRING TERMINATED BY A NULL INTO OUTPUT BUFFER WHICH IS
                                        ; OUTPUT TO TERMINAL IN A SINGLE QIO IF :
                                        ;    - FULL - OR <CR> OR <US> OR 377 ENCOUNTERED
                                        ;
                                        ; CALLING SEQUENCE:             JSR PC,DIOUT
                                        ; ON ENTRY:                     R0 POINTS TO STRING TO BE OUTPUT
                                        ;-----------------------------------------------------------------
                                        ;
000342  010146                  DIOUT:: MOV  R1,-(SP)                   ;SAVE R1
000344  016701  177450                  MOV  BUFCNT,R1                  ;GET ACTUAL BUFFER COUNT
000350  111061  000022'          GETCHA: MOVB (R0),BUFFER(R1)            ;GET CHARACTER
000354  001447                          BEQ  EXOUT                      ;NO MORE CHARS IF ZERO
000356  005201                          INC  R1                         ;VALID CHAR => INCREMENT COUNT
000360  005200                          INC  R0                         ;... AND POINTER
                                        ;
                                        ; LOOK WETHER AN OUTPUT IS TO BE DONE OR NOT
                                        ;
000362  122761  000015  000021'         CMPB #15,BUFFER-1(R1)           ;IF CARRIAGE RETURN
000370  001413                          BEQ  OUT                        ;... THEN FORCE OUTPUT OF BUFFER
000372  122761  000037  000021'         CMPB #37,BUFFER-1(R1)           ;IF US
000400  001407                          BEQ  OUT                        ;... THEN FORCE OUTPUT OF BUFFER
000402  122761  000377  000021'         CMPB #377,BUFFER-1(R1)          ;IF 377
000410  001403                          BEQ  OUT                        ;... THEN FORCE OUTPUT OF BUFFER
000412  020127  000110                  CMP  R1,#MAXCNT                 ;IF STILL PLACE IN BUFFER
000416  002754                          BLT  GETCHA                     ;... THEN GET NEXT CHAR
000420                          OUT:    QIOW$S  #IO.WAL,#5,#5,,#IOSTAT,,<#BUFFER,R1,#0>
000470  005001                          CLR  R1                         ;RESET BUFFER COUNT
000472  000726                          BR   GETCHA
000474  010167  177320          EXOUT:  MOV  R1,BUFCNT                  ;SAVE BUFFER POINTER
000500  012601                          MOV  (SP)+,R1                   ;RESTORE R1
000502  000207                          RTS  PC

DIPACK  MACRO M1200  25-FEB-85 13:11  PAGE 3


                                        ;
                                        ; DIGCUR - SHOW GRAPHICS (CROSSHAIR) CURSOR AND WAIT FOR INPUT
                                        ;
                                        ; CALLING SEQUENCE:       JSR  PC,DIGCUR
                                        ; ON ENTRY:               R0 POINTS TO A STRING OF 5 BYTES WHERE
                                        ;                         THE GRAPHIC INPUT WILL BE STORED
                                        ;
000504  010046                  DIGCUR:: MOV    R0,-(SP)                ;SAVE R0 FOR A WHILE
000506  012700  000007'                  MOV    #GINPUT,R0              ;GINPUT PROD. CROSSHAIR CURSOR
000512  004767  177624                  JSR    PC,DIOUT                ;OUTPUT (FORCED BY 377 AT END)
000516  012600                          MOV    (SP)+,R0                ;RESTORE R0
000520                                  QIOW$S #IO.RAL,#5,#5,,#IOSTAT,,<R0,#6,#0> ; READ INPUT
000570  000207                          RTS    PC                       ;AND RETURN TO CALLING PROGRAM

DIPACK  MACRO M1200  25-FEB-85 13:11  PAGE 4


                                        ;
                                        ;-----------------------------------------------------------------
                                        ;
                                        ; FORTRAN COMMON /DICOM/ FOR DISPLAY ROUTINES
                                        ;
000000                                  .PSECT DICOM,RW,D,GBL,OVR,REL
000000                          IXMIN:  .BLKW 1         ;USER LEFT MARGIN
000002                          IXMAX:  .BLKW 1         ;USER RIGHT MARGIN
000004                          IXRAN:  .BLKW 1         ;USER X-RANGE
000006                          IYMIN:  .BLKW 2         ;USER BOTTOM MARGIN (INTEGER*4)
000012                          IYMAX:  .BLKW 2         ;USER TOP MARGIN    (INTEGER*4)
000016                          IYRAN:  .BLKW 2         ;USER Y-RANGE       (INTEGER*4)
000022                          IMODE:  .BLKW 1         ;CURRENT SOFTWARE DISPLAY MODE
000024                          IUSERX: .BLKW 1         ;CURRENT USER X-POSITION
000026                          IUSERY: .BLKW 2         ;CURRENT USER Y-POSITION (INTEGER*4)
                                        ;
000032                          KXLO:   .BLKW 1         ;SCREEN LEFT MARGIN
000034                          KXHI:   .BLKW 1         ;SCREEN RIGHT MARGIN
000036                          KXRAN:  .BLKW 1         ;SCREEN X-RANGE
000040                          KYLO:   .BLKW 1         ;SCREEN BOTTOM MARGIN
000042                          KYHI:   .BLKW 1         ;SCREEN TOP MARGIN
000044                          KYRAN:  .BLKW 1         ;SCREEN Y-RANGE
000046                          KKMODE: .BLKW 1         ;CURRENT HARDWARE DISPLAY MODE
000050                          KBEAMX: .BLKW 1         ;CURRENT BEAM X-POSITION
000052                          KBEAMY: .BLKW 1         ;CURRENT BEAM Y-POSITION
000054                          KSPEED: .BLKW 1         ;TERINAL SPEED IN CHARACTERS PER SECOND
000056                          KPCHAR: .BLKW 4         ;CURRENT BEAM POSITION IN CHARACTER FORMAT
000066      002     000         ABEAM:  .BYTE 2,0       ;ARGUMENT BLOCK FOR CURRENT SCREEN COORDINATES
000070  000050'                         .WORD KBEAMX
000072  000052'                         .WORD KBEAMY
000074      002     000         AUSER:  .BYTE 2,0       ;ARGUMENT BLOCK FOR CURRENT USER COORDINATES
000076  000024'                         .WORD IUSERX
000100  000026'                         .WORD IUSERY
000102      002     000         AKXKY:  .BYTE 2,0       ;ARGUMENT BLOCK TEMPORARY SCREEN COORDINATES
000104  000110'                         .WORD KX
000106  000112'                         .WORD KY
000110                          KX:     .BLKW 1         ;TEMPORARY SCREEN X-COORDINATE
000112                          KY:     .BLKW 1         ;TEMPORARY SCREEN Y-COORDINATE
000114      002     000         AIXIY:  .BYTE 2,0       ;ARGUMENT BLOCK TEMPORARY USER COORDINATES
000116  000122'                         .WORD IX
000120  000124'                         .WORD IY
000122                          IX:     .BLKW 1         ;TEMPORARY USER X-COORDINATE (INTEGER*2)
000124                          IY:     .BLKW 2         ;TEMPORARY USER Y-COORDINATE (INTEGER*4)
                                        .END
            000001
```

```
DIABS   MACRO M1200  25-FEB-85 13:10  PAGE 1


                                        .TITLE DIABS
                                ;
                                ;-----------------------------------------------------------------
                                ; DRAW ABSOLUTE VECTOR OR POINT TO SCREEN POSITION (IX,IY)
                                ;
                                ; CALLING SEQUENCE:              CALL DIABS  (IX,IY,IMODE)
                                ;                               CALL DIMOVA (IX,IY)     ;DARK VECTOR
                                ;                               CALL DIDRWA (IX,IY)     ;BRIGHT VECTOR
                                ;                               CALL DIPNTA (IX,IY)     ;POINT
                                ;
                                ; ON ENTRY:  IX      = ABSOLUTE SCREEN X-COORDINATE
                                ;            IY      = ABSOLUTE SCREEN Y-COORDINATE
                                ;            IMODE   = 1 => DARK VECTOR
                                ;                    = 2 => BRIGHT VECTOR
                                ;                    = 3 => POINT
                                ;-----------------------------------------------------------------
                                ;
                                        .GLOBL DIVEC,DIPNT,DICNVT
                                ;
                                ;
000000  017567  000006  000022' DIABS::  MOV   @6(R5),IMODE              ;GET DESIRED MODE
000006  026727  000022' 000002  DIA::    CMP   IMODE,#2                 ;TAKE ACTION ACCORDING TO MODE
000014  003021                           BGT   DIPNTA                   ;  MODE = 3
000016  001403                           BEQ   DIDRWA                   ;  MODE = 2
                                ;
000020  004767  0000000G        DIMOVA:: JSR   PC,DIVEC                 ;  MODE = 1
000024  000423                           BR    GOTOXY
                                ;
000026  022767  000001  000046' DIDRWA:: CMP   #1,KKMODE                ;IF ALREADY IN VECTOR MODE
000034  001417                           BEQ   GOTOXY                   ;... THEN DRAW VECTOR
000036  004767  0000000G                 JSR   PC,DIVEC                 ;... ELSE SWITCH TO VECTOR MODE
000042  010546                           MOV   R5,-(SP)                 ;SAVE ARGUMENT POINTER
000044  012705  000066'                  MOV   #ABEAM,R5                ;POINTER TO CURRENT BEAM COORD.
000050  004767  0000000G                 JSR   PC,DICNVT                ;DAW DARK VECTOR TO LAST POSITION
000054  012605                           MOV   (SP)+,R5                 ;RESTORE POINTER TO (X,Y)
000056  000406                           BR    GOTOXY
                                ;
000060  022767  000002  000046' DIPNTA:: CMP   #2,KKMODE                ;IF ALREADY IN POINT MODE
000066  001402                           BEQ   GOTOXY                   ;... THEN DRAW POINT
000070  004767  0000000G                 JSR   PC,DIPNT                 ;... ELSE SWITCH TO POINT MODE
000074  004767  0000000G        GOTOXY: JSR   PC,DICNVT                ;DRAW VECTOR TO (X,Y)
000100  000207                           RTS   PC
                                ;
                                ;-----------------------------------------------------------------
                                ;
                                ; FORTRAN COMMON /DICOM/ FOR DISPLAY ROUTINES
                                ;
                                        .PSECT DICOM,RW,D,GBL,OVR,REL
000000                          IXMIN:  .BLKW 1         ;USER LEFT MARGIN
000000                          IXMAX   .BLKW 1         ;USER RIGHT MARGIN
000002                          IXRAN:  .BLKW 1         ;USER X-RANGE
000004                          IYMIN:  .BLKW 2         ;USER BOTTOM MARGIN (INTEGER*4)
000006                          IYMAX:  .BLKW 2         ;USER TOP MARGIN    (INTEGER*4)
000012

DIABS   MACRO M1200  25-FEB-85 13:10  PAGE 1-1


000016                          IYRAN:  .BLKW 2         ;USER Y-RANGE       (INTEGER*4)
000022                          IMODE:  .BLKW 1         ;CURRENT SOFTWARE DISPLAY MODE
000024                          IUSERX: .BLKW 1         ;CURRENT USER X-POSITION
000026                          IUSERY: .BLKW 2         ;CURRENT USER Y-POSITION (INTEGER*4)
                                ;
000032                          KXLO:   .BLKW 1         ;SCREEN LEFT MARGIN
000034                          KXHI:   .BLKW 1         ;SCREEN RIGHT MARGIN
000036                          KXRAN:  .BLKW 1         ;SCREEN X-RANGE
000040                          KYLO:   .BLKW 1         ;SCREEN BOTTOM MARGIN
000042                          KYHI:   .BLKW 1         ;SCREEN TOP MARGIN
000044                          KYRAN:  .BLKW 1         ;SCREEN Y-RANGE
000046                          KKMODE: .BLKW 1         ;CURRENT HARDWARE DISPLAY MODE
000050                          KBEAMX: .BLKW 1         ;CURRENT BEAM X-POSITION
000052                          KBEAMY: .BLKW 1         ;CURRENT BEAM Y-POSITION
000054                          KSPEED: .BLKW 1         ;TERINAL SPEED IN CHARACTERS PER SECOND
000056                          KPCHAR: .BLKW 4         ;CURRENT BEAM POSITION IN CHARACTER FORMAT
000066      002     000         ABEAM:  .BYTE 2,0       ;ARGUMENT BLOCK FOR CURRENT SCREEN COORDINATES
000070  000050'                         .WORD KBEAMX
000072  000052'                         .WORD KBEAMY
000074      002     000         AUSER:  .BYTE 2,0       ;ARGUMENT BLOCK FOR CURRENT USER COORDINATES
000076  000024'                         .WORD IUSERX
000100  000026'                         .WORD IUSERY
000102      002     000         AKXKY:  .BYTE 2,0       ;ARGUMENT BLOCK TEMPORARY SCREEN COORDINATES
000104  000110'                         .WORD KX
000106  000112'                         .WORD KY
000110                          KX:     .BLKW 1         ;TEMPORARY SCREEN X-COORDINATE
000112                          KY:     .BLKW 1         ;TEMPORARY SCREEN Y-COORDINATE
000114      002     000         AIXIY:  .BYTE 2,0       ;ARGUMENT BLOCK TEMPORARY USER COORDINATES
000116  000122'                         .WORD IX
000120  000124'                         .WORD IY
000122                          IX:     .BLKW 1         ;TEMPORARY USER X-COORDINATE (INTEGER*2)
000124                          IY:     .BLKW 2         ;TEMPORARY USER Y-COORDINATE (INTEGER*4)
            000001                      .END
```

```
DIREL   MACRO M1200  25-FEB-85 13:12  PAGE 1


                                        .TITLE DIREL
                                ;
                                ;-----------------------------------------------------------------
                                ; DRAW VECTOR OR POINT TO RELATIVE SCREEN COORDINATES (INCX,INCY)
                                ;
                                ;               ROLF GUIGAS     FEB-1982
                                ;
                                ; CALLING SEQUENCE:    CALL DIREL  (INCX,INCY,IMODE)  ;POINT OR VECTOR
                                ;                      CALL DIMOVR (INCX,INCY)        ;DARK VECTOR
                                ;                      CALL DIDRWR (INCX,INCY)        ;BRIGHT VECTOR
                                ;                      CALL DIPNTR (INCX,INCY)        ;POINT
                                ;
                                ; ON ENTRY:            INCX    = SCREEN X-INCREMENT
                                ;                      INCY    = SCREEN Y-INCREMENT
                                ;                      IMODE   = 1 => DARK VECTOR
                                ;                                2 => BRIGHT VECTOR
                                ;                                3 => POINT
                                ;-----------------------------------------------------------------
                                ;
                                        .GLOBL DIA
                                ;
000000  017567  000006  000022' DIREL::: MOV   @6(R5),IMODE              ;GET DESIRED MODE
000006  067567  000002  000050' RELOC:   ADD   @2(R5),KBEAMX             ;ADD X-INCREMENT TO CURRENT POS
000014  067567  000004  000052'          ADD   @4(R5),KBEAMY             ;ADD Y-INCREMENT TO CURRENT POS
000022  010546                           MOV   R5,-(SP)                  ;SAVE R5
000024  012705  000066'                  MOV   #ABEAM,R5                 ;POINTER TO ARGUMENT BLOCK
000030  004767  000000G                  JSR   PC,DIA                    ;DRAW ACCORDING TO MODE
000034  012605                           MOV   (SP)+,R5                  ;RESTORE R5
000036  000207                           RTS   PC
                                ;
000040  012767  000001  000022' DIMOVR::: MOV   #1,IMODE                 ;MODE=1 => DARK VECTOR
000046  000757                           BR    RELOC
000050  012767  000002  000022' DIDRWR::: MOV   #2,IMODE                 ;MODE=2 => BRIGHT VECTOR
000056  000753                           BR    RELOC
000060  012767  000003  000022' DIPNTR::: MOV   #3,IMODE                 ;MODE=3 => POINT
000066  000747                           BR    RELOC
                                ;
                                ;-----------------------------------------------------------------
                                ;
                                ; FORTRAN COMMON /DICOM/ FOR DISPLAY ROUTINES
                                ;
000000                                  .PSECT DICOM,RW,D,GBL,OVR,REL
000000                          IXMIN:  .BLKW 1         ;USER LEFT MARGIN
000002                          IXMAX:  .BLKW 1         ;USER RIGHT MARGIN
000004                          IXRAN:  .BLKW 1         ;USER X-RANGE
000006                          IYMIN:  .BLKW 2         ;USER BOTTOM MARGIN (INTEGER*4)
000012                          IYMAX:  .BLKW 2         ;USER TOP MARGIN    (INTEGER*4)
000016                          IYRAN:  .BLKW 2         ;USER Y-RANGE       (INTEGER*4)
000022                          IMODE:  .BLKW 1         ;CURRENT SOFTWARE DISPLAY MODE
000024                          IUSERX: .BLKW 1         ;CURRENT USER X-POSITION
000026                          IUSERY: .BLKW 2         ;CURRENT USER Y-POSITION (INTEGER*4)
```

```
DIREL   MACRO M1200  25-FEB-85 13:12  PAGE 1-1


000032                          KXLO:   .BLKW 1         ;SCREEN LEFT MARGIN
000034                          KXHI:   .BLKW 1         ;SCREEN RIGHT MARGIN
000036                          KXRAN:  .BLKW 1         ;SCREEN X-RANGE
000040                          KYLO:   .BLKW 1         ;SCREEN BOTTOM MARGIN
000042                          KYHI:   .BLKW 1         ;SCREEN TOP MARGIN
000044                          KYRAN:  .BLKW 1         ;SCREEN Y-RANGE
000046                          KKMODE: .BLKW 1         ;CURRENT HARDWARE DISPLAY MODE
000050                          KBEAMX: .BLKW 1         ;CURRENT BEAM X-POSITION
000052                          KBEAMY: .BLKW 1         ;CURRENT BEAM Y-POSITION
000054                          KSPEED: .BLKW 1         ;TERINAL SPEED IN CHARACTERS PER SECOND
000056                          KPCHAR: .BLKW 4         ;CURRENT BEAM POSITION IN CHARACTER FORMAT
000066      002       000       ABEAM:  .BYTE 2,0       ;ARGUMENT BLOCK FOR CURRENT SCREEN COORDINATES
000070  000050'                          .WORD KBEAMX
000072  000052'                          .WORD KBEAMY
000074      002       000       AUSER:  .BYTE 2,0       ;ARGUMENT BLOCK FOR CURRENT USER COORDINATES
000076  000024'                          .WORD IUSERX
000100  000026'                          .WORD IUSERY
000102      002       000       AKXKY:  .BYTE 2,0       ;ARGUMENT BLOCK TEMPORARY SCREEN COORDINATES
000104  000110'                          .WORD KX
000106  000112'                          .WORD KY
000110                          KX:     .BLKW 1         ;TEMPORARY SCREEN X-COORDINATE
000112                          KY:     .BLKW 1         ;TEMPORARY SCREEN Y-COORDINATE
000114      002       000       AIXIY:  .BYTE 2,0       ;ARGUMENT BLOCK TEMPORARY USER COORDINATES
000116  000122'                          .WORD IX
000120  000124'                          .WORD IY
000122                          IX:     .BLKW 1         ;TEMPORARY USER X-COORDINATE (INTEGER*2)
000124                          IY:     .BLKW 2         ;TEMPORARY USER Y-COORDINATE (INTEGER*4)
            000001                      .END
```

```
DJABS   MACRO M1200  25-FEB-85 13:12  PAGE 1


                                        .TITLE DJABS
                                ;
                                ;-----------------------------------------------------------------
                                ; DRAW VECTOR OR POINT TO ABSOLUTE USER COORDINATES (IX,JY)
                                ;
                                ;               ROLF GUIGAS     FEB-1982
                                ;
                                ; CALLING SEQUENCE:    CALL DJABS  (IX,JY,IMODE)  ;POINT OR VECTOR
                                ;                      CALL DJMOVA (IX,JY)        ;DARK VECTOR
                                ;                      CALL DJDRWA (IX,JY)        ;BRIGHT VECTOR
                                ;                      CALL DJPNTA (IX,JY)        ;POINT
                                ;
                                ; ON ENTRY:            IX      = USER X-COORDINATE (INTEGER*2)
                                ;                      JY      = USER Y-COORDINATE (INTEGER*4)
                                ;                      IMODE   = 1 => DARK VECTOR
                                ;                                2 => BRIGHT VECTOR
                                ;                                3 => POINT
                                ;-----------------------------------------------------------------
                                ;
                                        .GLOBL DJCNVT
                                ;
000000  017567  000006  000022' DJABS::  MOV   @6(R5),IMODE              ;GET DESIRED MODE
000006  004767  000000G         CNVT:   JSR   PC,DJCNVT                 ;CONVERT TO SCREEN COORD.
000012  000207                          RTS   PC
                                ;
000014  012767  000001  000022' DJMOVA:: MOV   #1,IMODE                  ;MODE=1 => DARK VECTOR
000022  000771                          BR    CNVT
000024  012767  000002  000022' DJDRWA:: MOV   #2,IMODE                  ;MODE=2 => BRIGHT VECTOR
000032  000765                          BR    CNVT
000034  012767  000003  000022' DJPNTA:: MOV   #3,IMODE                  ;MODE=3 => POINT
000042  000761                          BR    CNVT
                                ;
                                ;-----------------------------------------------------------------
                                ;
                                ; FORTRAN COMMON /DICOM/ FOR DISPLAY ROUTINES
                                ;
                                        .PSECT DICOM,RW,D,GBL,OVR,REL
000000                          IXMIN:  .BLKW 1         ;USER LEFT MARGIN
000002                          IXMAX:  .BLKW 1         ;USER RIGHT MARGIN
000004                          IXRAN:  .BLKW 1         ;USER X-RANGE
000006                          IYMIN:  .BLKW 2         ;USER BOTTOM MARGIN (INTEGER*4)
000012                          IYMAX:  .BLKW 2         ;USER TOP MARGIN    (INTEGER*4)
000016                          IYRAN:  .BLKW 2         ;USER Y-RANGE       (INTEGER*4)
000022                          IMODE:  .BLKW 1         ;CURRENT SOFTWARE DISPLAY MODE
000024                          IUSERX: .BLKW 1         ;CURRENT USER X-POSITION
000026                          IUSERY: .BLKW 2         ;CURRENT USER Y-POSITION (INTEGER*4)
                                ;
000032                          KXLO:   .BLKW 1         ;SCREEN LEFT MARGIN
000034                          KXHI:   .BLKW 1         ;SCREEN RIGHT MARGIN
000036                          KXRAN:  .BLKW 1         ;SCREEN X-RANGE
000040                          KYLO:   .BLKW 1         ;SCREEN BOTTOM MARGIN
000042                          KYHI:   .BLKW 1         ;SCREEN TOP MARGIN

DJABS   MACRO M1200  25-FEB-85 13:12  PAGE 1-1

000044                          KYRAN:  .BLKW 1         ;SCREEN Y-RANGE
000046                          KKMODE: .BLKW 1         ;CURRENT HARDWARE DISPLAY MODE
000050                          KBEAMX: .BLKW 1         ;CURRENT BEAM X-POSITION
000052                          KBEAMY: .BLKW 1         ;CURRENT BEAM Y-POSITION
000054                          KSPEED: .BLKW 1         ;TERINAL SPEED IN CHARACTERS PER SECOND
000056                          KPCHAR: .BLKW 4         ;CURRENT BEAM POSITION IN CHARACTER FORMAT
000066      002         DC:     ABEAM:  .BYTE 2,0       ;ARGUMENT BLOCK FOR CURRENT SCREEN COORDINATES
000070  000050'                         .WORD KBEAMX
000072  000052'                         .WORD KBEAMY
000074      002         DC:     AUSER:  .BYTE 2,0       ;ARGUMENT BLOCK FOR CURRENT USER COORDINATES
000076  000024'                         .WORD IUSERX
000100  000026'                         .WORD IUSERY
000102      002         DC:     AKXKY:  .BYTE 2,0       ;ARGUMENT BLOCK TEMPORARY SCREEN COORDINATES
000104  000110'                         .WORD KX
000106  000112'                         .WORD KY
000110                          KX:     .BLKW 1         ;TEMPORARY SCREEN X-COORDINATE
000112                          KY:     .BLKW 1         ;TEMPORARY SCREEN Y-COORDINATE
000114      002         DC:     AIXIY:  .BYTE 2,0       ;ARGUMENT BLOCK TEMPORARY USER COORDINATES
000116  000122'                         .WORD IX
000120  000124'                         .WORD IY
000122                          IX:     .BLKW 1         ;TEMPORARY USER X-COORDINATE (INTEGER*2)
000124                          IY:     .BLKW 2         ;TEMPORARY USER Y-COORDINATE (INTEGER*4)
                                        .END

        000001
```

```
DJREL   MACRO M1200  25-FEB-85 13:12  PAGE 1


                                        .TITLE DJREL
                                ;
                                ;-----------------------------------------------------------------
                                ; DRAW VECTOR OR POINT TO RELATIVE USER COORDINATES (INCX,INCY)
                                ;
                                ;               ROLF GUIGAS     FEB-1982
                                ;
                                ; CALLING SEQUENCE:    CALL DJREL  (INCX,INCY,IMODE)  ;POINT OR VECTOR
                                ;                      CALL DJMOVR (INCX,INCY)        ;DARK VECTOR
                                ;                      CALL DJDRWR (INCX,INCY)        ;BRIGHT VECTOR
                                ;                      CALL DJPNTR (INCX,INCY)        ;POINT
                                ;
                                ; ON ENTRY:            INCX    = USER X-INCREMENT (INTEGER*2)
                                ;                      INCY    = USER Y-INCREMENT (INTEGER*4)
                                ;                      IMODE   = 1 => DARK VECTOR
                                ;                                2 => BRIGHT VECTOR
                                ;                                3 => POINT
                                ;-----------------------------------------------------------------
                                ;
                                        .GLOBL DJCNVT
                                ;
                                ;
000000  017567  000006  000022' DJREL::  MOV   @6(R5),IMODE              ;GET DESIRED MODE
000006  067567  000002  000024' RELOC:   ADD   @2(R5),IUSERX             ;ADD X-INCREMENT TO CURRENT POS.
000014  010546                          MOV   R5,-(SP)                  ;SAVE R5
000016  016505  000004                  MOV   4(R5),R5                  ;POINTER TO LOW Y
000022  062567  000026'                ADD   (R5)+,IUSERY              ;ADD Y-INCREMENT TO CURRENT POS.
000026  005567  000030'                ADC   IUSERY+2                  ;... IN
000032  061567  000030'                ADD   (R5),IUSERY+2             ;... DOUBLE PRECISION
000036  012705  000074'                MOV   #AUSER,R5                 ;POINTER TO ARGUMENT BLOCK
000042  004767  000000G                JSR   PC,DJCNVT                 ;DISPLAY ACCORDING TO MODE
000046  012605                          MOV   (SP)+,R5                  ;RESTORE R5
000050  000207                          RTS   PC
                                ;
000052  012767  000001  000022' DJMOVR:: MOV   #1,IMODE                  ;MODE=1 => DARK VECTOR
000060  000752                          BR    RELOC
000062  012767  000002  000022' DJDRWR:: MOV   #2,IMODE                  ;MODE=2 => BRIGHT VECTOR
000070  000746                          BR    RELOC
000072  012767  000003  000022' DJPNTR:: MOV   #3,IMODE                  ;MODE=3 => POINT
000100  000742                          BR    RELOC
                                ;
                                ;-----------------------------------------------------------------
                                ;
                                ; FORTRAN COMMON /DICOM/ FOR DISPLAY ROUTINES
                                ;
                                        .PSECT DICOM,RW,D,GBL,OVR,REL
000000                          IXMIN:  .BLKW 1         ;USER LEFT MARGIN
000002                          IXMAX:  .BLKW 1         ;USER RIGHT MARGIN
000004                          IXRAN:  .BLKW 1         ;USER X-RANGE
000006                          IYMIN:  .BLKW 2         ;USER BOTTOM MARGIN (INTEGER*4)
000012                          IYMAX:  .BLKW 2         ;USER TOP MARGIN    (INTEGER*4)
000016                          IYRAN:  .BLKW 2         ;USER Y-RANGE       (INTEGER*4)
000022                          IMODE:  .BLKW 1         ;CURRENT SOFTWARE DISPLAY MODE

DJREL   MACRO M1200  25-FEB-85 13:12  PAGE 1-1


000024                          IUSERX: .BLKW 1         ;CURRENT USER X-POSITION
000026                          IUSERY: .BLKW 2         ;CURRENT USER Y-POSITION (INTEGER*4)
                                ;
000032                          KXLO:   .BLKW 1         ;SCREEN LEFT MARGIN
000034                          KXHI:   .BLKW 1         ;SCREEN RIGHT MARGIN
000036                          KXRAN:  .BLKW 1         ;SCREEN X-RANGE
000040                          KYLO:   .BLKW 1         ;SCREEN BOTTOM MARGIN
000042                          KYHI:   .BLKW 1         ;SCREEN TOP MARGIN
000044                          KYRAN:  .BLKW 1         ;SCREEN Y-RANGE
000046                          KKMODE: .BLKW 1         ;CURRENT HARDWARE DISPLAY MODE
000050                          KBEAMX: .BLKW 1         ;CURRENT BEAM X-POSITION
000052                          KBEAMY: .BLKW 1         ;CURRENT BEAM Y-POSITION
000054                          KSPEED: .BLKW 1         ;TERINAL SPEED IN CHARACTERS PER SECOND
000056                          KPCHAR: .BLKW 4         ;CURRENT BEAM POSITION IN CHARACTER FORMAT
000066      002     000         ABEAM:  .BYTE 2,0       ;ARGUMENT BLOCK FOR CURRENT SCREEN COORDINATES
000070  000050'                         .WORD KBEAMX
000072  000052'                         .WORD KBEAMY
000074      002     000         AUSER:  .BYTE 2,0       ;ARGUMENT BLOCK FOR CURRENT USER COORDINATES
000076  000024'                         .WORD IUSERX
000100  000026'                         .WORD IUSERY
000102      002     000         AKXKY:  .BYTE 2,0       ;ARGUMENT BLOCK TEMPORARY SCREEN COORDINATES
000104  000110'                         .WORD KX
000106  000112'                         .WORD KY
000110                          KX:     .BLKW 1         ;TEMPORARY SCREEN X-COORDINATE
000112                          KY:     .BLKW 1         ;TEMPORARY SCREEN Y-COORDINATE
000114      002     000         AIXIY:  .BYTE 2,0       ;ARGUMENT BLOCK TEMPORARY USER COORDINATES
000116  000122'                         .WORD IX
000120  000124'                         .WORD IY
000122                          IX:     .BLKW 1         ;TEMPORARY USER X-COORDINATE (INTEGER*2)
000124                          IY:     .BLKW 2         ;TEMPORARY USER Y-COORDINATE (INTEGER*4)
                                        .END
            000001
```

```
DIAXIS  MACRO M1200  25-FEB-85 13:10  PAGE 1


                                                .TITLE DIAXIS
                                                .IDENT /010382/
                                        ;
                                        ;-----------------------------------------------------------------
                                        ; DRAW X- AND Y-AXIS WITH TIC MARKS AND PRINT NUMERIC LABELS
                                        ;
                                        ;                       ROLF GUIGAS     1-MAR-82
                                        ;
                                        ; CALLING SEQUENCE:             CALL DIAXIS
                                        ;-----------------------------------------------------------------
                                        ;
                                                .GLOBL DIDRWR,DIMOVR,DIDEC,DITXT,DJMOVA
                                        ;
                000006                          LTIC    = 6             ;LENGTH OF TIC MARK
                000113                          LABXX   = 75.           ;X-OFFSET OF X-LABEL
                000030                          LABXY   = 24.           ;Y-OFFSET OF X-LABEL
                000143                          LABYX   = 99.           ;X-OFFSET OF Y-LABEL
                000010                          LABYY   = 8.            ;Y-OFFSET OF Y-LABEL
                                        ;
000000      002     000                 AXY:    .BYTE 2,0               ;ARGUMENT BLOCK FOR TIC POSITION
000002  000006'                                 .WORD XX
000004  000010'                                 .WORD YY
000006                                  XX:     .BLKW 1
000010                                  YY:     .BLKW 2
000014      001     000                 ALABX:  .BYTE 1,0               ;ARGUMENT BLOCK FOR X-LABEL VALUE
000016  000006'                                 .WORD XX
000020      001     000                 ALABY:  .BYTE 1,0               ;ARGUMENT BLOCK FOR Y-LABEL VALUE
000022  000024'                                 .WORD ILAB
000024  000000                          ILAB:   .WORD 0
000026  000000                          INC:    .WORD 0
000030  000000  000000                  JINC:   .WORD 0,0
000034      001     000                 AUNIT:  .BYTE 1,0               ;ARGUMENT BLOCK FOR UNIT
000036  000040'                                 .WORD UNIT
000040  000000                          UNIT:   .WORD 0
                                        ;
                                        ; CALCULATE SUITABLE INCREMENT BETWEEN X-TICS
                                        ;
000042  016701  000004'                 DIAXIS::MOV   IXRAN,R1          ;RANGE TO SCALE
000046  004767  000536                        ,JSR   PC,DELTA           ;FIND SUITABLE INCREMENT BETWEEN TICS
000052  016767  000000'  177726                MOV   IXMIN,XX          ;START
000060  016767  000006'  177722                MOV   IYMIN,YY          ;...    AT
000066  016767  000010'  177716                MOV   IYMIN+2,YY+2      ;...       ORIGIN
                                        ;
                                        ; DRAW X-AXIS WITH TIC MARKS AND LABELS
                                        ;
000074  012705  000000'                 NEXTX:  MOV   #AXY,R5           ;POINTER TO ARGUMENTS
000100  004767  000000G                        JSR   PC,DJMOVA           ;GO TO TIC MARK POSITION
000104  005067  000110'                        CLR   KX                  ;LOAD OFFSETS
000110  012767  177772  000112'               MOV   #-LTIC,KY           ;... FOR TIC MARK
000116  012705  000102'                        MOV   #AKXKY,R5
000122  004767  000000G                        JSR   PC,DIDRWR           ;DRAW TIC MARK
000126  012767  177665  000110'               MOV   #-LABXX,KX          ;LOAD OFFSETS
000134  012767  177750  000114'               MOV   #-LABXY,KY          ;... FOR LABEL POSITION

DIAXIS  MACRO M1200  25-FEB-85 13:10  PAGE 1-1


000142  004767  000000G                        JSR   PC,DIMOVR           ;GO TO LABEL START
000146  012705  000014'                        MOV   #ALABX,R5           ;ARGUMENT POINTER FOR LABEL
000152  004767  000000G                        JSR   PC,DIDEC            ;PRINT LABEL
000156  060367  177624                         ADD   R3,XX               ;INCREMENT X
000162  026767  177620  000002'                CMP   XX,IXMAX            ;IF STILL IN RANGE
000170  003741                                 BLE   NEXTX               ;... THEN MAKE NEXT X-TIC
                                        ;
                                        ; FIND SCALE UNIT FOR Y-AXIS ( 1 OR 10**3 OR 10**6 )
                                        ;
000172  016701  000012'                         MOV   IYMAX,R1          ;SCALE MAXIMUM
000176  016700  000014'                         MOV   IYMAX+2,R0
000202  016703  000016'                         MOV   IYRAN,R3          ;LOAD RANGE
000206  016702  000020'                         MOV   IYRAN+2,R2
000212  005700                                  TST   R0                ;IF > 0
000214  001010                                  BNE   1$                ;... THEN MAX EXCEEDS 16 BIT
000216  020127  023420                         CMP   R1,#10000.        ;IF MAX < 10**4
000222  103005                                  BHIS  1$
000224  012767  000040  177606                 MOV   #40,UNIT          ;... THEN UNIT = 1 (BLANK)
000232  010301                                  MOV   R3,R1             ;LOAD NUMBER TO SCALE
000234  000425                                  BR    GETINC
000236  020027  000230                  1$:     CMP   R0,#152.          ;IF MAX > 10**7
000242  003012                                  BGT   E6                ;... THEN UNIT = M (10**6)
000244  002403                                  BLT   E3
000246  020127  113200                         CMP   R1,#38528.
000252  103006                                  BHIS  E6
000254  012767  000113  177556          E3:     MOV   #113,UNIT         ;UNIT = K (10**3)
000262  071227  001750                         DIV   #1000.,R2         ;DIVIDE RANGE BY 1000
000266  000407                                  BR    LDINC
000270  012767  000115  177542          E6:     MOV   #115,UNIT         ;UNIT = M (10**6)
000276  073227  177772                         ASHC  #-6,R2            ;DIVIDE RANGE BY 2**6
000302  071227  036411                         DIV   #15625.,R2        ;DIVIDE RANGE BY 5**6
000306  010201                          LDINC:  MOV   R2,R1             ;LOAD NUMBER TO SCALE
                                        ;
                                        ; CALCULATE SUITABLE INCREMENT BETWEEN Y-TICS
                                        ;
000310  004767  000274                  GETINC: JSR   PC,DELTA          ;FIND SUITABLE INCREMENT
000314  010367  177506                         MOV   R3,INC             ;SAVE INCREMENT
000320  010302                                 MOV   R3,R2
000322  016701  000006'                        MOV   IYMIN,R1
000326  016700  000010'                        MOV   IYMIN+2,R0
000332  016767  000000'  177446                MOV   IXMIN,XX          ;START
000340  010167  177444                         MOV   R1,YY             ;...    AT
000344  010067  177442                         MOV   R0,YY+2           ;...       ORIGIN
000350  026727  177464  000040                CMP   UNIT,#40          ;IF UNIT = 1
000356  001003                                 BNE   K
000360  010100                                 MOV   R1,R0             ;... THEN FIRST LABEL = IYMIN
000362  005002                                 CLR   R2                ;... AND HIGH ORDER INCR. = 0
000364  000421                                 BR    YAX
000366  026727  177446  000113          K:     CMP   UNIT,#113         ;IF UNIT = K
000374  001005                                 BNE   M
000376  070227  001750                         MUL   #1000.,R2         ;... THEN JINC = INC * K
000402  071027  001750                         DIV   #1000.,R0         ;...       AND ILAB = IYMIN / K
000406  000410                                 BR    YAX
```

```
DIAXIS  MACRO M1200  25-FEB-85 13:10  PAGE 1-2


 107 000410  070227  036411          M:      MUL  #15625.,R2       ; JINC = INC * 5**6
 108 000414  073227  000006                  ASHC #6,R2            ;...              * 2**6
 109 000420  073027  177772                  ASHC #-6,R0           ; ILAB = IYMIN / 2**6
 110 000424  071027  036411                  DIV  #15625.,R0       ;                 / 5**6
 111 000430  010067  177370          YAX:    MOV  R0,ILAB          ;FIRST LABEL VALUE
 112 000434  010367  177370                  MOV  R3,JINC          ;LONG WORD
 113 000440  010267  177366                  MOV  R2,JINC+2        ;...        INCREMENT
 114                                 ;
 115                                 ; DRAW Y-AXIS WITH TIC MARKS AND LABELS
 116                                 ;
 117 000444  012705  000000'         NEXTY:  MOV  #AXY,R5          ;POINTER TO ARGUMENTS
 118 000450  004767  000000G                 JSR  PC,DJMOVA        ;GO TO TIC POSITION
 119 000454  012767  177772  000110'        MOV  #-LTIC,KX        ;LOAD OFFSETS
 120 000462  005067  000112'                 CLR  KY               ;...            FOR TIC MARK
 121 000466  012705  000102'                 MOV  #AKXKY,R5
 122 000472  004767  000000G                 JSR  PC,DIDRWR        ;DRAW TIC MARK
 123 000476  012767  177635  000110'        MOV  #-LABYX,KX       ;LOAD OFFSETS
 124 000504  012767  177770  000112'        MOV  #-LABYY,KY       ;...            FOR LABEL POSITION
 125 000512  004767  000000G                 JSR  PC,DIMOVR        ;MOVE BEAM TO LABEL START
 126 000516  012705  000020'                 MOV  #ALABY,R5        ;ARGUMENT POINTER FOR LABEL
 127 000522  004767  000000G                 JSR  PC,DIDEC         ;PRINT LABEL
 128 000526  012705  000034'                 MOV  #AUNIT,R5
 129 000532  004767  000000G                 JSR  PC,DITXT         ;PRINT UNIT OF LABEL VALUE
 130 000536  066767  177264  177260        ADD  INC,ILAB         ;INCREMENT LABEL VALUE
 131 000544  066767  177260  177236        ADD  JINC,YY          ;ADVANCE
 132 000552  005567  177234                 ADC  YY+2             ;...       TO
 133 000556  066767  177250  177226        ADD  JINC+2,YY+2      ;...            NEXT TIC MARK
 134 000564  026767  177222  000014'        CMP  YY+2,IYMAX+2     ;IF STILL IN RANGE
 135 000572  002724                         BLT  NEXTY            ;... THEN DRAW NEXT Y-TIC
 136 000574  003004                         BGT  EXAX
 137 000576  026767  177206  000012'        CMP  YY,IYMAX
 138 000604  101717                         BLOS NEXTY
 139 000606  000207          EXAX:   RTS  PC
 140                                 ;
 141                                 ; CALCULATE SUITABLE INCREMENT BETWEEN TIC MARKS
 142                                 ;
 143                                 ; ON ENTRY:   R1 = NUMBER TO SCALE
 144                                 ; ON EXIT:    R3 = INCREMENT
 145                                 ;
 146                                 ; METHOD:   NUMBER = A * 10**I + B
 147                                 ;                 A < 2       => INC = 2 * 10**(I-1)
 148                                 ;           1< A < 5          => INC = 5 * 10**(I-1)
 149                                 ;              A > 4          => INC = 10**I
 150                                 ;
 151 000610  012703  000001          DELTA:  MOV  #1,R3            ;START WITH 10**0
 152 000614  020127  000012                  CMP  R1,#10.          ;IF <= 10
 153 000620  003425                          BLE  .EXDEL           ;... THEN INCREMENT = 1
 154 000622  070327  000012          POWER:  MUL  #10.,R3          ;NEXT POWER OF TEN
 155 000626  005000                          CLR  R0               ;ZERO HIGH ORDER PART FOR DIVISION
 156 000630  071027  000012                  DIV  #10.,R0          ;DIVIDE BY TEN
 157 000634  010001                          MOV  R0,R1            ;LOAD RESULT FOR NEXT DIVISION
 158 000636  020127  000012                  CMP  R1,#10.          ;IF STILL >= TEN
 159 000642  002367                          BGE  POWER            ;... THEN TRY NEXT POWER

DIAXIS  MACRO M1200  25-FEB-85 13:10  PAGE 1-3


 160 000644  020127  000004                  CMP  R1,#4            ;IF > 4
 161 000650  003011                          BGT  EXDEL            ;... THEN INC = 10**I
 162 000652  020127  000002                  CMP  R1,#2            ;IF > 1
 163 000656  002402                          BLT  1$
 164 000660  006203                          ASR  R3               ;... THEN INC = 5 * 10**(I-1)
 165 000662  000404                          BR   EXDEL
 166 000664  005002          1$:     CLR  R2
 167 000666  071227  000005                  DIV  #5,R2            ; INC = 2 * 10**(I-1)
 168 000672  010203                          MOV  R2,R3
 169 000674  000207          EXDEL:  RTS  PC
 170                                 ;
 171                                 ;-----------------------------------------------------------------
 172                                 ;
 173                                 ; FORTRAN COMMON /DICOM/ FOR DISPLAY ROUTINES
 174                                 ;
 175 000000                          .PSECT DICOM,RW,D,GBL,OVR,REL
 176 000000                  IXMIN:  .BLKW 1       ;USER LEFT MARGIN
 177 000002                  IXMAX:  .BLKW 1       ;USER RIGHT MARGIN
 178 000004                  IXRAN:  .BLKW 1       ;USER X-RANGE
 179 000006                  IYMIN:  .BLKW 2       ;USER BOTTOM MARGIN (INTEGER*4)
 180 000012                  IYMAX:  .BLKW 2       ;USER TOP MARGIN    (INTEGER*4)
 181 000016                  IYRAN:  .BLKW 2       ;USER Y-RANGE       (INTEGER*4)
 182 000022                  IMODE:  .BLKW 1       ;CURRENT SOFTWARE DISPLAY MODE
 183 000024                  IUSERX: .BLKW 1       ;CURRENT USER X-POSITION
 184 000026                  IUSERY: .BLKW 2       ;CURRENT USER Y-POSITION (INTEGER*4)
 185                                 ;
 186 000032                  KXLO:   .BLKW 1       ;SCREEN LEFT MARGIN
 187 000034                  KXHI:   .BLKW 1       ;SCREEN RIGHT MARGIN
 188 000036                  KXRAN:  .BLKW 1       ;SCREEN X-RANGE
 189 000040                  KYLO:   .BLKW 1       ;SCREEN BOTTOM MARGIN
 190 000042                  KYHI:   .BLKW 1       ;SCREEN TOP MARGIN
 191 000044                  KYRAN:  .BLKW 1       ;SCREEN Y-RANGE
 192 000046                  KKMODE: .BLKW 1       ;CURRENT HARDWARE DISPLAY MODE
 193 000050                  KBEAMX: .BLKW 1       ;CURRENT BEAM X-POSITION
 194 000052                  KBEAMY: .BLKW 1       ;CURRENT BEAM Y-POSITION
 195 000054                  KSPEED: .BLKW 1       ;TERINAL SPEED IN CHARACTERS PER SECOND
 196 000056                  KPCHAR: .BLKW 4       ;CURRENT BEAM POSITION IN CHARACTER FORMAT
 197 000066      002     000     ABEAM:  .BYTE 2,0     ;ARGUMENT BLOCK FOR CURRENT SCREEN COORDINATES
 198 000070  000050'                 .WORD KBEAMX
 199 000072  000052'                 .WORD KBEAMY
 200 000074      002     000     AUSER:  .BYTE 2,0     ;ARGUMENT BLOCK FOR CURRENT USER COORDINATES
 201 000076  000024'                 .WORD IUSERX
 202 000100  000026'                 .WORD IUSERY
 203 000102      002     000     AKXKY:  .BYTE 2,0     ;ARGUMENT BLOCK TEMPORARY SCREEN COORDINATES
 204 000104  000110'                 .WORD KX
 205 000106  000112'                 .WORD KY
 206 000110                  KX:     .BLKW 1       ;TEMPORARY SCREEN X-COORDINATE
 207 000112                  KY:     .BLKW 1       ;TEMPORARY SCREEN Y-COORDINATE
 208 000114      002     000     AIXIY:  .BYTE 2,0     ;ARGUMENT BLOCK TEMPORARY USER COORDINATES
 209 000116  000122'                 .WORD IX
 210 000120  000124'                 .WORD IY
 211 000122                  IX:     .BLKW 1       ;TEMPORARY USER X-COORDINATE (INTEGER*2)
 212 000124                  IY:     .BLKW 2       ;TEMPORARY USER Y-COORDINATE (INTEGER*4)

DIAXIS  MACRO M1200  25-FEB-85 13:10  PAGE 1-4


 213         000001                  .END
```

```
DICNVT  MACRO M1200  25-FEB-85 13:10  PAGE 1

                                        .TITLE DICNVT
                                ;
                                ;-----------------------------------------------------------------
                                ; CONVERT SCREEN COORDINATES (IX,IY) TO CHARACTERS AND OUTPUT THEM
                                ;
                                ; CALLING SEQUENCE:          CALL DICNVT (IX,IY)
                                ;-----------------------------------------------------------------
                                ;
                                        .GLOBL DIOUT
                                ;
                                ;       OLD LO X    = KPCHAR
                                ;       OLD HI X    = KPCHAR+2
                                ;       OLD LO Y    = KPCHAR+4
                                ;       OLD HI Y    = KPCHAR+6
                                ;
000000                          NEWX:
000000  000             NLX:    .BYTE 0
000001  000             NHX:    .BYTE 0
000002                  NEWY:
000002  000             NLY:    .BYTE 0
000003  000             NHY:    .BYTE 0
000004  000     000     000     BYTSTK: .BYTE 0,0,0,0,0
000007  000     000
000011  035     000             GSBUF:  .BYTE 29.,0
                                        .EVEN
                                ;
000014  010046                  DICNVT::MOV   R0,-(SP)                  ;SAVE REGISTERS
000016  010146                          MOV   R1,-(SP)
000020  010246                          MOV   R2,-(SP)
000022  017501  000122                  MOV   @2(R5),R1                 ;GET SCREEN X-COORDINATE
000026  020127  002000                  CMP   R1,#1024.
000032  103402                          BLO   1$
000034  012701  001777                  MOV   #1023.,R1
000040  072127  000003          1$:     ASH   #3,R1
000044  106001                          RORB  R1
000046  106001                          RORB  R1
000050  106001                          RORB  R1
000052  052701  020000                  BIS   #20000,R1                 ;FLAG X
000056  010167  177776                  MOV   R1,NEWX                   ;STORE NEW X
000062  017501  000114                  MOV   @4(R5),R1                 ;GET SCREEN Y-COORDINATE
000066  072127  000003                  ASH   #3,R1
000072  106001                          RORB  R1
000074  106001                          RORB  R1
000076  106001                          RORB  R1
000100  052701  020140                  BIS   #20140,R1                 ;FLAG Y
000104  010167  177672                  MOV   R1,NEWY                   ;STORE NEW Y
000110  012702  000004'                 MOV   #BYTSTK+4,R2              ;POINTER TO BYTE STACK
000114  116742  177672                  MOVB  NLX,-(R2)
000120  126767  177635  000060'         CMPB  NHX,KPCHAR+2
000126  001403                          BEQ   LYCK
000130  116742  177645                  MOVB  NHX,-(R2)
000134  000404                          BR    PUSHLY
000136  126767  177640  000062' LYCK:   CMPB  NLY,KPCHAR+4

DICNVT  MACRO M1200  25-FEB-85 13:10  PAGE 1-1

000144  001402                          BEQ   HYCK
000146  116742  177630          PUSHLY: MOVB  NLY,-(R2)
000152  126767  177625  000064' HYCK:   CMPB  NHY,KPCHAR+6
000160  001402                          BEQ   DMPBS
000162  116742  177615                  MOVB  NHY,-(R2)
000166  022767  000002  000046' DMPBS:  CMP   #2,KKMODE                 ;IF POINT MODE
000174  001004                          BNE   PASSA
000176  012700  000011'                 MOV   #GSBUF,R0                 ;... THEN FORCE DARK VECTOR
000202  004767  000000G                 JSR   PC,DIOUT
000206  010200                  PASSA:  MOV   R2,R0
000210  004767  000000G                 JSR   PC,DIOUT                  ;OUTPUT COORDINATES
000214  022767  000002  000046'         CMP   #2,KKMODE                 ;IF POINT MODE
000222  001004                          BNE   PASSO
000224  012700  000007'                 MOV   #BYTSTK+3,R0              ;... THEN DRAW END POINT
000230  004767  000000G                 JSR   PC,DIOUT
000234  116767  177542  000056' PASSO:  MOVB  NLX,KPCHAR                ;UPDATE CODED COORDINATES
000242  116767  177533  000060'         MOVB  NHX,KPCHAR+2
000250  116767  177526  000062'         MOVB  NLY,KPCHAR+4
000256  116767  177521  000064'         MOVB  NHY,KPCHAR+6
000264  017567  000002  000050'         MOV   @2(R5),KBEAMX             ;UPDATE SCREEN COORDINATES
000272  017567  000004  000052'         MOV   @4(R5),KBEAMY
000300  012602                          MOV   (SP)+,R2                  ;RESTORE REGISTERS
000302  012601                          MOV   (SP)+,R1
000304  012600                          MOV   (SP)+,R0
000306  000207                          RTS   PC
                                ;
                                ;-----------------------------------------------------------------
                                ;
                                ; FORTRAN COMMON /DICOM/ FOR DISPLAY ROUTINES
                                ;
                                        .PSECT DICOM,RW,D,GBL,OVR,REL
000000                          IXMIN:  .BLKW 1         ;USER LEFT MARGIN
000002                          IXMAX:  .BLKW 1         ;USER RIGHT MARGIN
000004                          IXRAN:  .BLKW 1         ;USER X-RANGE
000006                          IYMIN:  .BLKW 2         ;USER BOTTOM MARGIN (INTEGER*4)
000012                          IYMAX:  .BLKW 2         ;USER TOP MARGIN    (INTEGER*4)
000016                          IYRAN:  .BLKW 2         ;USER Y-RANGE       (INTEGER*4)
000022                          IMODE:  .BLKW 1         ;CURRENT SOFTWARE DISPLAY MODE
000024                          IUSERX: .BLKW 1         ;CURRENT USER X-POSITION
000026                          IUSERY: .BLKW 2         ;CURRENT USER Y-POSITION (INTEGER*4)
                                ;
000032                          KXLO:   .BLKW 1         ;SCREEN LEFT MARGIN
000034                          KXHI:   .BLKW 1         ;SCREEN RIGHT MARGIN
000036                          KXRAN:  .BLKW 1         ;SCREEN X-RANGE
000040                          KYLO:   .BLKW 1         ;SCREEN BOTTOM MARGIN
000042                          KYHI:   .BLKW 1         ;SCREEN TOP MARGIN
000044                          KYRAN:  .BLKW 1         ;SCREEN Y-RANGE
000046                          KKMODE: .BLKW 1         ;CURRENT HARDWARE DISPLAY MODE
000050                          KBEAMX: .BLKW 1         ;CURRENT BEAM X-POSITION
000052                          KBEAMY: .BLKW 1         ;CURRENT BEAM Y-POSITION
000054                          KSPEED: .BLKW 1         ;TERINAL SPEED IN CHARACTERS PER SECOND
000056                          KPCHAR: .BLKW 4         ;CURRENT BEAM POSITION IN CHARACTER FORMAT
000066  002     006             ABEAM:  .BYTE 2,0       ;ARGUMENT BLOCK FOR CURRENT SCREEN COORDINATES

DICNVT  MACRO M1200  25-FEB-85 13:10  PAGE 1-2

000070  000050'                        .WORD KBEAMX
000072  000052'                        .WORD KBEAMY
000074  002     006             AUSER:  .BYTE 2,0       ;ARGUMENT BLOCK FOR CURRENT USER COORDINATES
000076  000024'                        .WORD IUSERX
000100  000026'                        .WORD IUSERY
000102  002     006             AKXKY:  .BYTE 2,0       ;ARGUMENT BLOCK TEMPORARY SCREEN COORDINATES
000104  000110'                        .WORD KX
000106  000112'                        .WORD KY
000110                          KX:     .BLKW 1         ;TEMPORARY SCREEN X-COORDINATE
000112                          KY:     .BLKW 1         ;TEMPORARY SCREEN Y-COORDINATE
000114  002     006             AIXIY:  .BYTE 2,0       ;ARGUMENT BLOCK TEMPORARY USER COORDINATES
000116  000122'                        .WORD IX
000120  000124'                        .WORD IY
000122                          IX:     .BLKW 1         ;TEMPORARY USER X-COORDINATE (INTEGER*2)
000124                          IY:     .BLKW 2         ;TEMPORARY USER Y-COORDINATE (INTEGER*4)
                                        .END

        000001
```

```
DJCNVT  MACRO M1200  25-FEB-85 13:12  PAGE 1


                                        .TITLE DJCNVT
                                ;
                                ;-----------------------------------------------------------------
                                ; CONVERT USER COORDINATES TO SCREEN COORDINATES AND OUPUT THEM
                                ;
                                ;               ROLF GUIGAS             5-FEB-82
                                ;
                                ; CALLING SEQUENCE:   CALL DJCNVT (IX,JY)
                                ;
                                ; ON ENTRY: IX      = INTEGER*2 USER X-COORDINATE
                                ;           JY      = INTEGER*4 USER Y-COORDINATE
                                ;-----------------------------------------------------------------
                                ;
                                        .GLOBL DIA,RGSAVE,RGREST
                                ;
000000  004767  000000G        DJCNVT::JSR   PC,RGSAVE                 ;SAVE REGISTERS
000004  017500  000002                  MOV   @2(R5),R0                 ;GET X-COORDINATE
000010  010067  000024'                 MOV   R0,IUSERX                 ;UPDATE USER X
000014  166700  000000'                 SUB   IXMIN,R0                  ;SUBTRACT LEFT MARGIN
000020  070067  000036'                 MUL   KXRAN,R0                  ;SCALE
000024  071067  000004'                 DIV   IXRAN,R0                  ;... USER DISTANCE
000030  066700  000032'                 ADD   KXLO,R0                   ;... TO SCREEN DISTANCE
000034  010067  000110'                 MOV   R0,KX                     ;SAVE CONVERTED X
                                ;
                                ; CONVERT Y COORDINATE
                                ;
000040  016500  000004                  MOV   4(R5),R0                  ;POINTER TO Y COORDINATE
000044  012001                          MOV   (R0)+,R1                  ;GET LOW  PART
000046  011000                          MOV   (R0),R0                   ;GET HIGH PART
000050  010167  000026'                 MOV   R1,IUSERY                 ;UPDATE
000054  010067  000030'                 MOV   R0,IUSERY+2               ;...    USER Y
000060  166701  000006'                 SUB   IYMIN,R1                  ;DOUBLE PRECISION
000064  005600                          SBC   R0                        ;... SUBTRACTION
000066  166700  000010'                 SUB   IYMIN+2,R0                ;... OF BOTTOM MARGIN
000072  016703  000016'                 MOV   IYRAN,R3                  ;GET LOW DIVISOR
000076  016702  000020'                 MOV   IYRAN+2,R2                ;GET HIGH DIVISOR
                                ;
                                ; NORMALIZE DOUBLE INTEGERS SO THAT MSB = BIT 31
                                ;
000102  073227  000001        SHIFT:    ASHC  #1,R2                     ;SHIFT DIVISOR
000106  102403                          BVS   NORM                      ;... UNTIL BIT 31 SET
000110  073027  000001                  ASHC  #1,R0                     ;SHIFT DIVIDENT
000114  000772                          BR    SHIFT
000116  006002                  NORM:   ROR   R2                        ;RESTORE SIGN BIT
000120  070067  000044'                 MUL   KYRAN,R0                  ;SCALE
000124  071002                          DIV   R2,R0                     ;... USER Y DISTANCE
000126  066700  000040'                 ADD   KYLO,R0                   ;... TO SCREEN Y DISTANCE
000132  010067  000112'                 MOV   R0,KY                     ;SAVE CONVERTED Y
000136  012705  000102'                 MOV   #AKXKY,R5                 ;POINTER TO SCREEN ARGUMENTS
000142  004767  000000G                 JSR   PC,DIA                    ;DISPLAY ACCORDING TO MODE
000146  004767  000000G                 JSR   PC,RGREST                 ;RESTORE REGISTERS
000152  000207                          RTS   PC

DJCNVT  MACRO M1200  25-FEB-85 13:12  PAGE 1-1


                                ;
                                ;-----------------------------------------------------------------
                                ;
                                ; FORTRAN COMMON /DICOM/ FOR DISPLAY ROUTINES
                                ;
000000                                  .PSECT DICOM,RW,D,GBL,OVR,REL
000000                          IXMIN:  .BLKW 1         ;USER LEFT MARGIN
000002                          IXMAX:  .BLKW 1         ;USER RIGHT MARGIN
000004                          IXRAN:  .BLKW 1         ;USER X-RANGE
000006                          IYMIN:  .BLKW 2         ;USER BOTTOM MARGIN (INTEGER*4)
000012                          IYMAX:  .BLKW 2         ;USER TOP MARGIN    (INTEGER*4)
000016                          IYRAN:  .BLKW 2         ;USER Y-RANGE       (INTEGER*4)
000022                          IMODE:  .BLKW 1         ;CURRENT SOFTWARE DISPLAY MODE
000024                          IUSERX: .BLKW 1         ;CURRENT USER X-POSITION
000026                          IUSERY: .BLKW 2         ;CURRENT USER Y-POSITION (INTEGER*4)
                                ;
000032                          KXLO:   .BLKW 1         ;SCREEN LEFT MARGIN
000034                          KXHI:   .BLKW 1         ;SCREEN RIGHT MARGIN
000036                          KXRAN:  .BLKW 1         ;SCREEN X-RANGE
000040                          KYLO:   .BLKW 1         ;SCREEN BOTTOM MARGIN
000042                          KYHI:   .BLKW 1         ;SCREEN TOP MARGIN
000044                          KYRAN:  .BLKW 1         ;SCREEN Y-RANGE
000046                          KKMODE: .BLKW 1         ;CURRENT HARDWARE DISPLAY MODE
000050                          KBEAMX: .BLKW 1         ;CURRENT BEAM X-POSITION
000052                          KBEAMY: .BLKW 1         ;CURRENT BEAM Y-POSITION
000054                          KSPEED: .BLKW 1         ;TERINAL SPEED IN CHARACTERS PER SECOND
000056                          KPCHAR: .BLKW 4         ;CURRENT BEAM POSITION IN CHARACTER FORMAT
000066     002      000         ABEAM:  .BYTE 2,0       ;ARGUMENT BLOCK FOR CURRENT SCREEN COORDINATES
000070  000050'                         .WORD KBEAMX
000072  000052'                         .WORD KBEAMY
000074     002      000         AUSER:  .BYTE 2,0       ;ARGUMENT BLOCK FOR CURRENT USER COORDINATES
000076  000024'                         .WORD IUSERX
000100  000026'                         .WORD IUSERY
000102     002      000         AKXKY:  .BYTE 2,0       ;ARGUMENT BLOCK TEMPORARY SCREEN COORDINATES
000104  000110'                         .WORD KX
000106  000112'                         .WORD KY
000110                          KX:     .BLKW 1         ;TEMPORARY SCREEN X-COORDINATE
000112                          KY:     .BLKW 1         ;TEMPORARY SCREEN Y-COORDINATE
000114     002      000         AIXIY:  .BYTE 2,0       ;ARGUMENT BLOCK TEMPORARY USER COORDINATES
000116  000122'                         .WORD IX
000120  000124'                         .WORD IY
000122                          IX:     .BLKW 1         ;TEMPORARY USER X-COORDINATE (INTEGER*2)
000124                          IY:     .BLKW 2         ;TEMPORARY USER Y-COORDINATE (INTEGER*4)
                                        .END

        000001
```

```
DICURI  MACRO M1200  25-FEB-85 13:10  PAGE 1

                                        .TITLE DICURI
                                ;
                                ;*****************************************************************
                                ;
                                ;  THIS SUBROUTINE DISPLAYS A CROSSHAIR CURSOR ON THE SCREEN, WAITS
                                ;  UNTIL A CHARACTER IS TYPED AT THE KEYBOARD AND RETURNS THE CURSOR
                                ;  COORDINATES (IN ABSOLUTE SCREEN COORDINATES) TO THE CALLER.
                                ;
                                ;  CALLING SEQUENCE:   CALL DICURI (ICHAR,IX,IY)
                                ;
                                ;  ON EXIT:   ICHAR (I*2):  CHARACTER TYPED ('RIGHT JUSTIFIED')
                                ;             IX:           X SCREEN COORDINATE
                                ;             IY:           Y SCREEN COORDINATE
                                ;
                                ;
                                ;  VER. 1.0/21-SEP-83(TK)  ORIGINAL VERSION
                                ;
                                ;*****************************************************************
                                ;
                                        .GLOBL DIGCUR,RGSAVE,RGREST
                                ;
000000                          GIBUFF: .BLKB  6.                       ;TEMPORARY STORAGE FOR GINPUT
                                ;
000006  004767  0000006G        DICURI::JSR    PC,RGSAVE                ;SAVE REGISTERS
000012  012700  000000'                MOV    #GIBUFF,R0               ;WHERE TO STORE GRAPH. INPUT
000016  004767  0000006G                JSR    PC,DIGCUR                ;GET GRAPHICS INPUT
000022  142767  000200  177750          BICB   #200,GIBUFF              ;MASK OFF 8TH BIT
000030  116775  177744  000002          MOVB   GIBUFF,2(R5)             ;STORE SENT CHARACTER
                                ;
                                ;  PROCESS X COORDINATE
                                ;
000036  116700  177737                  MOVB   GIBUFF+1,R0              ;PICK UP FIRST CHARACTER
000042  116701  177734                  MOVB   GIBUFF+2,R1              ;PICK UP SECOND CHARACTER
000046  004767  000032                  JSR    PC,EXTRA                 ;PRODUCE 16 BIT COORDINATE (R0)
000052  010075  000004                  MOV    R0,4(R5)                 ;STORE X COORDINATE
                                ;
                                ;  PROCESS Y COORDINATE
                                ;
000056  116700  177721                  MOVB   GIBUFF+3,R0              ;PICK UP FIRST CHARACTER
000062  116701  177716                  MOVB   GIBUFF+4,R1              ;PICK UP SECOND CHARACTER
000066  004767  000012                  JSR    PC,EXTRA                 ;PRODUCE 16 BIT COORDINATE (R0)
000072  010075  000006                  MOV    R0,6(R5)                 ;STORE Y COORDINATE
                                ;
000076  004767  0000006G              JSR    PC,RGREST                ;RESTORE REGISTERS
000102  000207                        RTS    PC                       ;AND RETURN TO CALLING PROGRAM

DICURI  MACRO M1200  25-FEB-85 13:10  PAGE 2

                                ;
                                ;  INTERNAL SUBROUTINE TO PRODUCE 16 BIT COORDINATE FROM 2 CHARACTERS
                                ;
                                ;  ON ENTRY:  R0:  FIRST CHARACTER,  R1:  SECOND CHARACTER
                                ;  ON EXIT:   R0:  16 BIT COORDINATE,  R1:  DESTROYED
                                ;
000104  042700  177740          EXTRA:  BIC    #177740,R0               ;MASK OFF ALL UNUSED STUFF
000110  000300                          SWAB   R0                       ;MOVE IT TO UPPER BYTE
000112  006000                          ROR    R0                       ;AND MOVE
000114  006000                          ROR    R0                       ;     IT BACK TO
000116  006000                          ROR    R0                       ;          POSITION TO 10 BIT
000120  042701  177740                  BIC    #177740,R1               ;CLEAN LOWER PART
000124  050100                          BIS    R1,R0                    ;MOVE IT INTO OUTPUT VALUE
000126  000207                          RTS    PC                       ;AND RETURN TO MAINLINE CODE
        000001                          .END
```

```
DIFRAM  MACRO M1200  25-FEB-85 13:11  PAGE 1


                                        .TITLE DIFRAM
                                ;
                                ;-----------------------------------------------------------------
                                ; DRAW A RECTANGULAR BOX ACCORDING TO SCREEN WINDOW
                                ;
                                ;               ROLF GUIGAS 17-FEB-1982
                                ;
                                ; CALLING SEQUENCE:            CALL DIFRAM
                                ;-----------------------------------------------------------------
                                ;
                                        .GLOBL DIDRVA,DIMOVA
                                ;
                                ;
000000  012705  000102'        DIFRAM::MOV   #AKXKY,R5                 ;POINTER TO ARGUMENTS
000004  016767  000032'  000110'        MOV   KXLO,KX                   ;START WITH LOWER LEFT CORNER
000012  016767  000040'  000112'        MOV   KYLO,KY
000020  004767  000000G                 JSR   PC,DIMOVA                 ;DARK VECTOR TO LOWER LEFT CORNER
000024  016767  000034'  000110'        MOV   KXHI,KX
000032  004767  000000G                 JSR   PC,DIDRVA                 ;VECTOR TO LOWER RIGHT CORNER
000036  016767  000042'  000112'        MOV   KYHI,KY
000044  004767  000000G                 JSR   PC,DIDRVA                 ;VECTOR TO UPPER RIGHT CORNER
000050  016767  000032'  000110'        MOV   KXLO,KX
000056  004767  000000G                 JSR   PC,DIDRVA                 ;VECTOR TO UPPER LEFT CORNER
000062  016767  000040'  000112'        MOV   KYLO,KY
000070  004767  000000G                 JSR   PC,DIDRVA                 ;VECTOR TO LOWER LEFT CORNER
000074  000207                          RTS PC
                                ;
                                ;-----------------------------------------------------------------
                                ;
                                ; FORTRAN COMMON /DICOM/ FOR DISPLAY ROUTINES
                                ;
000000                                  .PSECT DICOM,RW,D,GBL,OVR,REL
000000                          IXMIN:  .BLKW 1         ;USER LEFT MARGIN
000002                          IXMAX:  .BLKW 1         ;USER RIGHT MARGIN
000004                          IXRAN:  .BLKW 1         ;USER X-RANGE
000006                          IYMIN:  .BLKW 2         ;USER BOTTOM MARGIN (INTEGER*4)
000012                          IYMAX:  .BLKW 2         ;USER TOP MARGIN    (INTEGER*4)
000016                          IYRAN:  .BLKW 2         ;USER Y-RANGE       (INTEGER*4)
000022                          IMODE:  .BLKW 1         ;CURRENT SOFTWARE DISPLAY MODE
000024                          IUSERX: .BLKW 1         ;CURRENT USER X-POSITION
000026                          IUSERY: .BLKW 2         ;CURRENT USER Y-POSITION (INTEGER*4)
                                ;
000032                          KXLO:   .BLKW 1         ;SCREEN LEFT MARGIN
000034                          KXHI:   .BLKW 1         ;SCREEN RIGHT MARGIN
000036                          KXRAN:  .BLKW 1         ;SCREEN X-RANGE
000040                          KYLO:   .BLKW 1         ;SCREEN BOTTOM MARGIN
000042                          KYHI:   .BLKW 1         ;SCREEN TOP MARGIN
000044                          KYRAN:  .BLKW 1         ;SCREEN Y-RANGE
000046                          KKMODE: .BLKW 1         ;CURRENT HARDWARE DISPLAY MODE
000050                          KBEAMX  .BLKW 1         ;CURRENT BEAM X-POSITION
000052                          KBEAMY: .BLKW 1         ;CURRENT BEAM Y-POSITION
000054                          KSPEED: .BLKW 1         ;TERINAL SPEED IN CHARACTERS PER SECOND
000056                          KPCHAR: .BLKW 4         ;CURRENT BEAM POSITION IN CHARACTER FORMAT

DIFRAM  MACRO M1200  25-FEB-85 13:11  PAGE 1-1

000066      002     000         ABEAM:  .BYTE 2,0       ;ARGUMENT BLOCK FOR CURRENT SCREEN COORDINATES
000070  000050'                        .WORD KBEAMX
000072  000052'                        .WORD KBEAMY
000074      002     000         AUSER:  .BYTE 2,0       ;ARGUMENT BLOCK FOR CURRENT USER COORDINATES
000076  000024'                        .WORD IUSERX
000100  000026'                        .WORD IUSERY
000102      002     000         AKXKY:  .BYTE 2,0       ;ARGUMENT BLOCK TEMPORARY SCREEN COORDINATES
000104  000110'                        .WORD KX
000106  000112'                        .WORD KY
000110                          KX:     .BLKW 1         ;TEMPORARY SCREEN X-COORDINATE
000112                          KY:     .BLKW 1         ;TEMPORARY SCREEN Y-COORDINATE
000114      002     000         AIXIY:  .BYTE 2,0       ;ARGUMENT BLOCK TEMPORARY USER COORDINATES
000116  000122'                        .WORD IX
000120  000124'                        .WORD IY
000122                          IX:     .BLKW 1         ;TEMPORARY USER X-COORDINATE (INTEGER*2)
000124                          IY:     .BLKW 2         ;TEMPORARY USER Y-COORDINATE (INTEGER*4)
                                        .END

        000001
```

```
DIWNDW  MACRO M1200  25-FEB-85 13:12  PAGE 1


                                        .TITLE DIWNDW
                                ;
                                ;-----------------------------------------------------------------
                                ; DEFINE SCREEN WINDOW (SCREEN UNITS!)
                                ;       ROLF GUIGAS     17-FEB-1982
                                ;
                                ; CALLING SEQUENCE:  CALL DIWNDW (KXLO,KXHI,KXLO,KYHI)
                                ;
                                ; ON ENTRY: ABSOLUTE COORDINATES OF SCREEN WINDOW
                                ;-----------------------------------------------------------------
                                ;
000000  005725                  DIWNDW::TST   (R5)+                   ;SKIP NUMBER OF ARGUMENTS
000002  012704  000032'                MOV   #KXLO,R4                ;POINTER TO ENTRY IN COMMON
000006  013524                        MOV   @(R5)+,(R4)+            ;GET KXLO
000010  017524  000000                MOV   @(R5),(R4)+             ;GET KXHI
000014  013514                        MOV   @(R5)+,(R4)             ;GET KXHI AGAIN
000016  166714  000032'                SUB   KXLO,(R4)               ;CALCULATE X-RANGE
000022  005224                        INC   (R4)+                   ;KXRAN := KXHI-KXLO+1
000024  013524                        MOV   @(R5)+,(R4)+            ;GET KYLO
000026  017524  000000                MOV   @(R5),(R4)+             ;GET KYHI
000032  013514                        MOV   @(R5)+,(R4)             ;GET KYHI AGAIN
000034  166714  000040'                SUB   KYLO,(R4)               ;CALCULATE Y-RANGE
000040  005214                        INC   (R4)                    ;KYRAN := KYHI-KYLO+1
000042  000207                        RTS   PC
                                ;
                                ;-----------------------------------------------------------------
                                ;
                                ; FORTRAN COMMON /DICOM/ FOR DISPLAY ROUTINES
                                ;
000000                                  .PSECT DICOM,RW,D,GBL,OVR,REL
000000                          IXMIN:  .BLKW 1         ;USER LEFT MARGIN
000002                          IXMAX:  .BLKW 1         ;USER RIGHT MARGIN
000004                          IXRAN:  .BLKW 1         ;USER X-RANGE
000006                          IYMIN:  .BLKW 2         ;USER BOTTOM MARGIN (INTEGER*4)
000012                          IYMAX:  .BLKW 2         ;USER TOP MARGIN    (INTEGER*4)
000016                          IYRAN:  .BLKW 2         ;USER Y-RANGE       (INTEGER*4)
000022                          IMODE:  .BLKW 1         ;CURRENT SOFTWARE DISPLAY MODE
000024                          IUSERX: .BLKW 1         ;CURRENT USER X-POSITION
000026                          IUSERY: .BLKW 2         ;CURRENT USER Y-POSITION (INTEGER*4)
                                ;
000032                          KXLO:   .BLKW 1         ;SCREEN LEFT MARGIN
000034                          KXHI:   .BLKW 1         ;SCREEN RIGHT MARGIN
000036                          KXRAN:  .BLKW 1         ;SCREEN X-RANGE
000040                          KYLO:   .BLKW 1         ;SCREEN BOTTOM MARGIN
000042                          KYHI:   .BLKW 1         ;SCREEN TOP MARGIN
000044                          KYRAN:  .BLKW 1         ;SCREEN Y-RANGE
000046                          KKMODE: .BLKW 1         ;CURRENT HARDWARE DISPLAY MODE
000050                          KBEAMX: .BLKW 1         ;CURRENT BEAM X-POSITION
000052                          KBEAMY: .BLKW 1         ;CURRENT BEAM Y-POSITION
000054                          KSPEED: .BLKW 1         ;TERINAL SPEED IN CHARACTERS PER SECOND
000056                          KPCHAR: .BLKW 4         ;CURRENT BEAM POSITION IN CHARACTER FORMAT
000066      002       000       ABEAM:  .BYTE 2,0       ;ARGUMENT BLOCK FOR CURRENT SCREEN COORDINATES
000070  000050'                        .WORD KBEAMX

DIWNDW  MACRO M1200  25-FEB-85 13:12  PAGE 1-1

000072  000052'                        .WORD KBEAMY
000074      002       000       AUSER:  .BYTE 2,0       ;ARGUMENT BLOCK FOR CURRENT USER COORDINATES
000076  000024'                        .WORD IUSERX
000100  000026'                        .WORD IUSERY
000102      002       000       AKXKY:  .BYTE 2,0       ;ARGUMENT BLOCK TEMPORARY SCREEN COORDINATES
000104  000110'                        .WORD KX
000106  000112'                        .WORD KY
000110                          KX:     .BLKW 1         ;TEMPORARY SCREEN X-COORDINATE
000112                          KY:     .BLKW 1         ;TEMPORARY SCREEN Y-COORDINATE
000114      002       000       AIXIY:  .BYTE 2,0       ;ARGUMENT BLOCK TEMPORARY USER COORDINATES
000116  000122'                        .WORD IX
000120  000124'                        .WORD IY
000122                          IX:     .BLKW 1         ;TEMPORARY USER X-COORDINATE (INTEGER*2)
000124                          IY:     .BLKW 2         ;TEMPORARY USER Y-COORDINATE (INTEGER*4)
        000001                          .END
```

```
DJWNDW  MACRO M1200  25-FEB-85 13:12  PAGE 1


                                        .TITLE DJWNDW
                                ;
                                ;-----------------------------------------------------------------
                                ; DEFINE USER WINDOW
                                ;               ROLF GUIGAS     FEB-1982
                                ;
                                ; BUG TAKEN OUT (IXRAN=IXMAX-IXMIN, NOT IXRAN=IXMAX-IXMIN+1)
                                ;               THOMAS KOEHLER  MAY-1983
                                ;
                                ; CALLING SEQUENCE:   CALL DJWNDW (IXMIN,IXMAX,JYMIN,JYMAX)
                                ;
                                ; ON ENTRY:           USER COORDINATES OF WINDOW (Y IN INTEGER*4 !)
                                ;-----------------------------------------------------------------
                                ;
000000  005725                  DJWNDW::TST   (R5)+                   ;SKIP NUMBER OF ARGUMENTS
000002  012704  000000'                MOV   #IXMIN,R4               ;POINTER TO ENTRY IN COMMON
000006  013524                        MOV   @(R5)+,(R4)+            ;GET IXMIN
000010  017524  000000                MOV   @(R5),(R4)+             ;GET IXMAX
000014  013514                        MOV   @(R5)+,(R4)             ;GET IXMAX AGAIN
000016  011467  000024'                MOV   (R4),IUSERX             ;IUSERX := IXMIN
000022  166714  000000'                SUB   IXMIN,(R4)              ;CALCULATE USER X-RANGE
                                ;;;;;   INC   (R4)+                   ;IXRAN := IXMAX-IXMIN+1
000026  005724                        TST   (R4)+                   ;POINT CORRECT
                                ;
000030  012503                        MOV   (R5)+,R3                ;POINTER TO IYMIN
000032  011324                        MOV   (R3),(R4)+              ;GET IYMIN LOW
000034  012367  000026'                MOV   (R3)+,IUSERY            ;USERY := IYMIN
000040  011367  000030'                MOV   (R3),IUSERY+2           ;... DOUBLE PRECISION
000044  011324                        MOV   (R3),(R4)+              ;GET IYMIN HIGH
000046  011503                        MOV   (R5),R3                 ;POINTER TO IYMAX
000050  011324                        MOV   (R3),(R4)+              ;GET IYMAX LOW
000052  012367  000016'                MOV   (R3)+,IYRAN             ;... AGAIN
000056  011324                        MOV   (R3),(R4)+              ;GET IYMAX HIGH
000060  011367  000020'                MOV   (R3),IYRAN+2            ;... AGAIN
000064  166724  000006'                SUB   IYMIN,(R4)+             ;DOUBLE PRECISION
000070  005614                        SBC   (R4)                    ;... CALCULATION
000072  166714  000010'                SUB   IYMIN+2,(R4)            ;... OF USER Y-RANGE
000076  005744                        TST   -(R4)
000100  062724  000001                ADD   #1,(R4)+
000104  005514                        ADC   (R4)                    ;IYRAN := IYMAX-IYMIN+1
000106  000207                        RTS   PC
                                ;
                                ;-----------------------------------------------------------------
                                ;
                                ; FORTRAN COMMON /DICOM/ FOR DISPLAY ROUTINES
                                ;
000000                                  .PSECT DICOM,RW,D,GBL,OVR,REL
000000                          IXMIN:  .BLKW 1       ;USER LEFT MARGIN
000002                          IXMAX:  .BLKW 1       ;USER RIGHT MARGIN
000004                          IXRAN:  .BLKW 1       ;USER X-RANGE
000006                          IYMIN:  .BLKW 2       ;USER BOTTOM MARGIN (INTEGER*4)
000012                          IYMAX:  .BLKW 2       ;USER TOP MARGIN    (INTEGER*4)
000016                          IYRAN:  .BLKW 2       ;USER Y-RANGE       (INTEGER*4)

DJWNDW  MACRO M1200  25-FEB-85 13:12  PAGE 1-1

000022                          IMODE:  .BLKW 1       ;CURRENT SOFTWARE DISPLAY MODE
000024                          IUSERX: .BLKW 1       ;CURRENT USER X-POSITION
000026                          IUSERY: .BLKW 2       ;CURRENT USER Y-POSITION (INTEGER*4)
                                ;
000032                          KXLO:   .BLKW 1       ;SCREEN LEFT MARGIN
000034                          KXHI:   .BLKW 1       ;SCREEN RIGHT MARGIN
000036                          KXRAN:  .BLKW 1       ;SCREEN X-RANGE
000040                          KYLO:   .BLKW 1       ;SCREEN BOTTOM MARGIN
000042                          KYHI:   .BLKW 1       ;SCREEN TOP MARGIN
000044                          KYRAN:  .BLKW 1       ;SCREEN Y-RANGE
000046                          KKMODE: .BLKW 1       ;CURRENT HARDWARE DISPLAY MODE
000050                          KBEAMX: .BLKW 1       ;CURRENT BEAM X-POSITION
000052                          KBEAMY: .BLKW 1       ;CURRENT BEAM Y-POSITION
000054                          KSPEED: .BLKW 1       ;TERINAL SPEED IN CHARACTERS PER SECOND
000056                          KPCHAR: .BLKW 4       ;CURRENT BEAM POSITION IN CHARACTER FORMAT
000066      002       000       ABEAM:  .BYTE 2,0     ;ARGUMENT BLOCK FOR CURRENT SCREEN COORDINATES
000070  000050'                        .WORD KBEAMX
000072  000052'                        .WORD KBEAMY
000074      002       000       AUSER:  .BYTE 2,0     ;ARGUMENT BLOCK FOR CURRENT USER COORDINATES
000076  000024'                        .WORD IUSERX
000100  000026'                        .WORD IUSERY
000102      002       000       AKXKY:  .BYTE 2,0     ;ARGUMENT BLOCK TEMPORARY SCREEN COORDINATES
000104  000110'                        .WORD KX
000106  000112'                        .WORD KY
000110                          KX:     .BLKW 1       ;TEMPORARY SCREEN X-COORDINATE
000112                          KY:     .BLKW 1       ;TEMPORARY SCREEN Y-COORDINATE
000114      002       000       AIXIY:  .BYTE 2,0     ;ARGUMENT BLOCK TEMPORARY USER COORDINATES
000116  000122'                        .WORD IX
000120  000124'                        .WORD IY
000122                          IX:     .BLKW 1       ;TEMPORARY USER X-COORDINATE (INTEGER*2)
000124                          IY:     .BLKW 2       ;TEMPORARY USER Y-COORDINATE (INTEGER*4)
            000001                      .END
```

```
PDP-11 FORTRAN-77 V5.0-0        13:05:59      3-May-84            Page 1
FITSPY.FTN;1                /F77/OP/TR:BLOCKS/WR

0001            PROGRAM FITSPY
          C=    ++++++++++++++
          C=
          C=    This routine prints the results on the line-printer
          C=
          C=
          C=    Ver. 1.0   29-Feb-84  (OR)  Original version
          C=
          C=
0002            IMPLICIT REAL*8 (A-H,O-Z)
          C=
0003            PARAMETER MAXCHN=4096, MAXPK=30, MAXCO=6, MAXREG=7, MAXPAR=40,
                +          LUNSPE=1, LUNPAR=2, LUNPRT=3, NPPARM=6, MXCALF=3,
                +          MPOINT=500, MXLINK=20, MAXLIN=10, LUN=6
          C=
0004            BYTE SPESPC(30), PARSPC(30)
0005            COMMON /OPPARS/ MDEV,SPESPC,PARSPC,MXITER,LBFIX,IREG,IXLO,IXHI,
                +               IYLO,IYHI,IYPWR,IFLO,IFHI,IPEAK,LBACK,LPEAK,LALL,
                +               NCOEFF,COEFF(MAXCO),ITER,NDF,PEAK(NPPARM,MAXPK),
                +               IFLAG(NPPARM,MAXPK),INDEX(NPPARM,MAXPK),ISEL,
                +               IGNOR(MAXPK),NREG,KREGLO(MAXREG),KREGHI(MAXREG),
                +               IMODE,INTCOR,NCALF,CALFAC(MXCALF),SIGMA(MAXPK),
                +               GAMMA(MAXPK)
0006            COMMON /OPPART/ IOPARS
0007            COMMON /ERRCOM/ IERR
          C=
0008            DATA IOPARS /0/, DT /1000.D0/, PI /3.141592654D0/,
                +     SRPI /1.772453851D0/
          C=+
          C=    Read FIT parameter-file
          C=
0009            CALL PARNAM ('FIT')
0010            CALL PARRD
          C=
          C=    Start printout with header lines
          C=
0011            WRITE (LUN,1000)
0012       1000 FORMAT ('1')
          C=
          C=    Printout of FIT regions
          C=
0013            WRITE (LUN,1005) ITER,((KREGLO(J),KREGHI(J)),J=1,NREG)
0014       1005 FORMAT ('0',I2,'. Iteration'/
                +       '0','FIT regions used: ',/,' ',6(I5,'-',I4))
          C=
          C=    Printout of background
          C=
0015            WRITE (LUN,1010) NCOEFF
0016       1010 FORMAT ('0','Background with',I3,' coefficients:')
0017            WRITE (LUN,1015) (COEFF(J),J=1,NCOEFF)
0018       1015 FORMAT (' ',' Coeffs: ',3(1PE16.8)/
                +       ' ',' ',3(1PE16.8))
          C=
          C=    Printout of calibration polynomial
          C=
```

```
PDP-11 FORTRAN-77 V5.0-0        13:05:59      3-May-84            Page 2
FITSPY.FTN;1                /F77/OP/TR:BLOCKS/WR

0019            IF (NCALF .GT. 0) THEN
0020               WRITE (LUN,1020)
0021       1020    FORMAT ('0','Calibration polynomial:')
0022               WRITE (LUN,1015) (CALFAC(J),J=1,NCALF)
0023            ENDIF
          C=
          C=    Printout of peaks
          C=
0024            NPLINE = 0
0025            DO 1050  IPEAK=1,MAXPK,1
0026               IF (IGNOR(IPEAK) .EQ. 1) GOTO 1050
          C=
          C=       Convert width to FWHM
          C=
0027               IWIDTH = INDEX(3,IPEAK)
          C=
          C=       Gaussian
          C=
0028               IPARM = 3
0029               XPI = SRPI
0030               WIDTH = SIGMA(IPEAK)
          C=
          C=       Lorentzian
          C=
0031               IF (IFLAG(6,IPEAK) .NE. -2)  THEN
0032                  IPARM = 6
0033                  XPI = PI
0034                  WIDTH = GAMMA(IPEAK)
0035               ENDIF
0036               IWIDTH = INDEX(IPARM,IPEAK)
          C=
          C=       Area
          C=
0037               AREA = XPI * PEAK(3,IPEAK) * WIDTH
```

```
          C=
          C=    Start output
          C=
0038            IF (NPLINE.EQ.0) THEN
0039               WRITE (LUN,1025)
0040       1025    FORMAT (' ')
0041               WRITE (LUN,1030)
0042       1030    FORMAT (' ','Position',2X,'Gauss-FWHM',3X,'Height',1X,
                +          'Low-tail',1X,'High-tail',1X,'Lorentz-FWHM',
                +          1X,'Intensity')
0043            ENDIF
0044            WRITE (LUN,1035) IPEAK,(IFLAG(I,IPEAK),I=1,NPPARM)
0045       1035 FORMAT ('0',' Peak',I3,10X,'Flags: ',5(I2,','),I2,')')
0046            WRITE (LUN,1040) (PEAK(I,IPEAK),I=1,NPPARM),AREA
0047       1040 FORMAT (' ',F8.3,1X,F7.3,1X,F13.2,2X,0PF7.3,2X,F7.3,2X,F7.3,
                +          1X,F13.2)
          C=
          C=    Print calibration (if any)
          C=
0048            IF (NCALF .GT. 0)  THEN
0049               XPI = PEAK(1,IPEAK)
```

```
PDP-11 FORTRAN-77 V5.0-0        13:05:59      3-May-84            Page 3
FITSPY.FTN;1                /F77/OP/TR:BLOCKS/WR

0050               WIDTH = 0.5D0*PEAK(2,IPEAK)
0051               DWIDTH = 0.5D0*PEAK(6,IPEAK)
0052               WRITE (LUN,1045) CALVAL(XPI),
                +                DT=(CALVAL(XPI+WIDTH)-CALVAL(XPI-WIDTH)),
                +                DT=(CALVAL(XPI+DWIDTH)-CALVAL(XPI-DWIDTH))
0053       1045    FORMAT (' ','E:',F9.4,',  ','G:',F8.1,',  ','L:',F8.1)
0054            ENDIF
          C=
          C=    Make a nice paging
          C=
0055            NPLINE = NPLINE + 1
0056            IF (NPLINE .GE. MAXLIN)  THEN
0057               WRITE (LUN,1047)
0058       1047    FORMAT ('1')
0059               NPLINE = 0
0060            ENDIF
0061       1050 CONTINUE
          C=
          C=    Terminate silently
          C=
0062        9000 END
```

```
PDP-11 FORTRAN-77 V5.0-0        13:06:16      3-May-84            Page 6
FITSPY.FTN;1                /F77/OP/TR:BLOCKS/WR

0001            DOUBLE PRECISION FUNCTION CALVAL (DCHAN)
          C=    ++++++++++++++++++++++++++++++++++++++++
          C=
          C=    This function returns the ...
          C=
          C=
0002            IMPLICIT REAL*8 (A-H,O-Z)
          C=
0003            PARAMETER MAXCHN=4096, MAXPK=30, MAXCO=6, MAXREG=7, MAXPAR=40,
                +          LUNSPE=1, LUNPAR=2, LUNPRT=3, NPPARM=6, MXCALF=3,
                +          MPOINT=500, MXLINK=20, MAXLIN=6, LUN=6
          C=
0004            BYTE SPESPC(30),PARSPC(30)
0005            COMMON /OPPARS/ MDEV,SPESPC,PARSPC,MXITER,LBFIX,IREG,IXLO,IXHI,
                +               IYLO,IYHI,IYPWR,IFLO,IFHI,IPEAK,LBACK,LPEAK,LALL,
                +               NCOEFF,COEFF(MAXCO),ITER,NDF,PEAK(NPPARM,MAXPK),
                +               IFLAG(NPPARM,MAXPK),INDEX(NPPARM,MAXPK),ISEL,
                +               IGNOR(MAXPK),NREG,KREGLO(MAXREG),KREGHI(MAXREG),
                +               IMODE,INTCOR,NCALF,CALFAC(MXCALF),SIGMA(MAXPK),
                +               GAMMA(MAXPK)
0006            COMMON /OPPART/ IOPARS
0007            COMMON /ERRCOM/ IERR
          C=
          C=
          C=    Apply calibration (if any)
          C=
0008            IF (NCALF.GT.0) THEN
0009               DTEMP = DCHAN
0010               CALVAL = CALFAC(1)
0011               DO 1000 J=2,NCALF,1
0012                  CALVAL = CALVAL+(DTEMP*CALFAC(J))
0013                  DTEMP = DTEMP*DTEMP
0014       1000    CONTINUE
0015            ELSE
0016               CALVAL = 0.0D0
0017            ENDIF
          C=
          C=    Return to calling program
          C=
0018            RETURN
          C=    ******
0019            END
```