79/18

EISCAT
TECHNICAL
NOTE

A PROGRAM LIBRARY FOR INCOHERENT SCATTER CALCULATION

by

G Lejeune

KIRUNA
Sweden

EISCAT Technical Note 79/18

A program library for incoherent scatter calculation

by

G. Lejeune

EISCAT Scientific Association
S-981 01 Kiruna, Sweden

Also submitted as CEPHAG Report No. 28/79, Grenoble 1979

EISCAT Technical Note 79/18
Printed in Sweden
EISCAT Scientific Association
September 1979

A PROGRAM LIBRARY FOR INCOHERENT SCATTER CALCULATION

G. LEJEUNE

-o-

SUMMARY :

This report describes in detail all the routines of the library for EISCAT data analysis, a part of which has already been implemented in real time at KIRUNA.

Efforts were made to allow a simple use of the analysis program : the only preparation needed is the edition of a symbolic file MODEL:DATA which will then control the overall analysis.

The coding of all the programs is given in appendix and examples of possible execution are shown.

CONTENTS

1. INTRODUCTION.

The program library in its present form follows the description given in the report "EISCAT DATA ANALYSIS" by LEJEUNE, 1979, hereafter referred to as EDA79. However, this report is not detailed enough for anybody not acquainted in the programs to be able to use them easily.

In this report, the different subroutines will be described in detail. The appendix contains the coding of the programs all in FORTRAN IV and sample outprints from CHATANIKA data analysis.

Figure 1 shows the hierarchy of the programs of the library. A control file MODEL:DATA is first prepared and treated by CRECONST (fig. 1A) ; the analysis (fig. 1B) can then proceed).

2. PROGRAM CRECONST.

The figure 2 is an outprint of a possible MODEL:DATA file, which must control the overall analysis. The numbering of parameters is given in EDA 79.

The file MODEL:DATA contains a header of 7 lines and several (limited to 4) groups of 16 + n lines corresponding to different altitude zones, where n is the maximum number of successive fits, on the 15th line (see later for explanation). Each line contains a comment in columns 1 to 63 and data from column 64.

2.1 MODEL:DATA header description.

line 1 : Transmission frequency in MHz

line 2 : Number of parameters fitted (here 4) in case of no subsequent requirement.

line 3 : Order of the parameters fitted (in case of no subsequent requirement). Here, parameters 1, 2, 3, 4 will be fitted, the others being fixed.

line 4 : Mass of the ion other than 0^+ (if nothing else later specified).

line 5 : Number of points for the FFT (if large oversampling, this number should be increased, but the speed is largely reduced).

line 6 : Index for outputs : if 1, minimum quantity of outprints
if 5, maximum quantity of outprints.

line 7 : Number of altitude zones (here 4).

2.2 MODEL:DATA altitude zone description.

line 1 : dummy

line 2 : limit of the altitude zone, in km

line 3 : mass of the ion other than O^+

lines 4 to 13 : default values of the parameters, if other "initial" values are not calculated later. The numbering of the parameters is the same as in EDA 79 from number 4 (ion wind in Hz). The numbers 1, 2 and 3 (NE, TE, TI) are not the parameters used in the fit

(number 1: $\left(\frac{1}{KD_e}\right)^2 \frac{TE}{TI}$; number 2 : $\frac{TE}{TI}$; number 3 : $\frac{2\pi}{kV_{TI}}$)

line 14 : says for which parameters an initial value other than the one fixed in the ten previous lines must be determined.

Format : $\underbrace{n_1 , n_2 , n_3 , \ldots, n_p}_{n_1 \text{ values}}$

n_1 is the number of subsequent values $n_2 \ldots, n_p$.
The possible values for $n_2, \ldots$ are 1, 2, 3, 4, 5 and 7.

line 15 gives the maximum number n of fits which will be performed on a single ACF. The description of each of those successive fits is given on lines 17, 18 ..., 16+n.

line 16 : If MODEL has been compiled with the conditional compilation P , the n values of ion composition on this line are successively used as initial values in the filts (can be useful in the F1 region).

lines 17, ..., 16+n : The format of each line is :

$$\underline{\text{Format}} \quad \underbrace{n_1, n_2, n_3, \ldots, n_p}_{n_1 \text{ values}}$$

They say that n_1 parameters ($n_2, n_3, \ldots, n_p$) will be adjusted in the fit. If there is convergence, and if more fits are required, the values of the fits are used as initial values for the next one.

Example : Assume that we want to analyse ACF data abtained at 90 km altitude.

From figure 2, we take specifications from zone 1 (MODEL not compiled with p).
Initial values : 4 will be determined. if enough SNR

- NE (from power)) in MODEL
- TI = TE } from a model) in MODEL
- X(7) } from a model)

- VI in VINIT.

The other initial values are the previous default values, which will remain unchanged during the analysis :

X(5) = 1
X(6) = 0
X(8) = 1
X(9) = 0
X(10) = 1

For the first fit, 3 parameters will be searched for : X(1), X(3), X(4). X(2) = TE/TI is thus fixed to 1, and X(7) fixed to its initial value.
For the second fit, the new initial values for X(1), X(3), X(4) are the ones just determined ; X(4) is fixed to this value (possible because this parameter is uncoupled to the others) and X(1), X(3), X(7) are the new floating parameters. In case of convergence, the additional parameter X(2) = TE/TI , together with X(1), X(3) and X(7), is included in the third fit.

2.3 Description of CRECONST.

The aim of CRECONST is to generate the binary file CONST:DATA which will monitor the analysis.

The data contained in the symbolic file MODEL:DATA are used to build the COMMON/MOD/(used by MODEL), and some constants of the COMMON/CONST/(AMMIN, default ion mass, $RI = \sqrt{AMMIN/16}$, $PK1 = 4\pi \frac{FREQ}{C}$, IESSAI).Some other constants are given in DATA or in the last 30 lines in CRECONST.

- IPAR describes the "parity" of the 10 parameters (used for the choice of the good FFT program for the different derivatives ; see FOURIER)

- IBEG is an initialisation parameter

- LN, LP, LQ, LR are constants used in the 64 points FFT routine (LM in COMMON/SPE/ is read in line 5 of MODEL:DATA).

- QN, CONV, QDIV, QINC are constants used in REGR. The value of CONV determines when the iterative process converges : the smaller it is, the more precise determination of the parameters.

- RPI : $\sqrt{\pi}$

- BE, BO, DEBE, DEBO, BFEO, DEBOP : constants assuming that O^+ ion is the reference ion.

- CO(16), SI(8,8) tables of cosines used in the 64 points FFT routine.

3. DESCRIPTION OF THE SUBROUTINES CALLED BY MAIN.

The program MAIN given in A2 can be reduced to the two simple forms :

```
CALL AINIT
CALL ANAL
CALL DIGER
```

or

```
CALL AINIT
CALL ANAL
CALL GENER
```

according to the answers to questions in MAIN. The second form is used only for testing purposes. The program GENER will be described in section 4.

If MAIN is used, all the subroutines must be compiled with COND-COMPIL - (background).

If instead, they are compiled with COND. COMPIL + (RT), there is no need of a MAIN program : the subroutines are called in real time by the RT program ANALYSIS written by Joe ARMSTRONG.

3.1 SUBROUTINE AINIT.

Reads the file CONST:DATA (see section 2).

3.2 SUBROUTINE DIGER.

This routine is a decoding of the CHATANIKA tapes CRIM (if compiled with M) and CROS (if compiled with S).

The results are put in 3 tables : IDAT(24), XCOR(64), ISIG2(32)

- IDAT(1)	counter
- IDAT(2→6)	free
- IDAT(7)	integration time (sec)
- IDAT(12)	number of pulses per sec.
- IDAT(8) =	100 x SNR
- IDAT(9) =	NSG, number of samples (used for single pulse)
- IDAT(13) =	LFI, number of points of the complex ACF used in the regression (limited to 32)
- IDAT(14) =	0.1 * SAMFRE (sampling frequency)
- IDAT(11)	free
- IDAT(15) =	altitude (km)
- IDAT(10) =	10^{-9} x power if the power is normalized such that power $\sim N_e(m^{-3})/(1 + \frac{T_e}{T_i})$
- IDAT(16,17,18,19) :	year, month, day hour
- IDAT(20,21) :	Minute, second at the beginning of data taking
- IDAT(22,23) :	Minute, second at the end of data taking
- IDAT(24) :	Cosine of the half of the angle between transmission and reception.

XCOR(1) = 10^6	zero lag real ACF point
XCOR(2) = 0	zero lag imag ACF point
.	
.	
.	
XCOR(2I-1)	Ith lag real CF point
XCOR(2I)	Ith lag imag ACF point
.	
.	
.	
XCOR(63)	32d lag real ACF point
XCOR(64)	32d lag imag ACF point.

ISIG2(1) = 10^{-3} x (σ^2 of the deviation on XCOR(1))

.
.
.

ISIG2(I) = 10^{-3} x (σ^2 of the deviation on XCOR(2I-1))
= " " " "

(one assumes that the σ's on the real and imaginary part of the ACF are equal).

3.3 SUBROUTINE ANAL

3.31 Input parameters :

NUMBER		counter
IDAT	)	
XCOR	))	see 3.2
ISIG2	)	

3.32 Output parameters :

RESULT(128)

This table contains the results of the analysis and is used in real time by the visualisation program HKEEPC to draw ACF and spectrum.

RESULT 1 = altitude (km)
2 = day
3 = hour
4 = minute
5 = second
6 = IDAT(7) x IDAT(12)
7 = IDAT(9)
8 = VAR (see later)
9 = σ of the 1st lag of the ACF
10 = IDAT(10)
11 = NE
12 = TE

13 = TI
14 = V
15 = P
16 = VE-VI
17 = error on NE
18 = " TE
19 = " TI
20 = " V
21 = " P
22 = " VE-VI
23)
24)
.) 20 lags of the experimental ACF (real, imag, real,...)
.)
.)
62)
63)
.)
.) 20 lags of the fitted ACF (only real parts)
.)
82)
83)
.)
.) 32 points of the spectrum
.)
114)
115 = IBEG created in REGR (if convergency, IBEG = 2)
116 = SNR
117 = LFI
118 = SAMFRE
119 = Year
120 = Month

3.33 Organigram of ANAL

3.331 ANAL used without the program TALK.

If in MAIN the use of TALK is not specified, the variable ITALK is zero and the corresponding organigram of ANAL is given in figure 3.1 . After the construction of the experimental

ACF normalized to NE (Y) and of the weights (POID) used in REGR, one enters a loop which can be performed at most the maximum number of fits given in MODEL:DATA. The first time the loop is described, INITCA = .TRUE. and initial values will be determined in MODEL according to the specifications of MODEL:DATA. The next times, the fit specifications only are read in MODEL (RET = .TRUE. when there is no more fit specified). Then the iterative process REGR is entered, ending with IBEG = 2 if convergence.

3.332 ANAL used with the program TALK.

Figure 3.2 shows that the route is determined in TALK. This mode is useful when testing the program or analysing in detail a small amount of data.

3.4 SUBROUTINE TALK.

The organigram of TALK is given in figure 4. This is a conversational program enough documented in itself.

3.5 SUBROUTINE MODEL.

Only one box of the organigram in figure 5 needs development, the one concerning the calculation of the initial values.

Referring to 2.2., line 14, MODEL performs an initial calculation for values 1, 2, 3, 4, 5 or 7.

Value 1 : NE = 2 x Y(1) x FD, where Y(1) x FD is the total spectrum power

Value 2 : Call TINIT which gives the TE and TI by searching the first zero crossing and the first minimum of the ACF

Value 3 : (incompatible with 2) : calculates TE=TI according to $TI = 195. + .2(ALT-90)^2$. This model should be used in the E region.

Value 4 : Calculates the initial value of the wind in VINIT.

Value 5 : Calculates an ion composition parameter according to

$$P = X(5) = 1/(1 + (\frac{ALT-120}{52.5})^4)$$

This model should be used in the F1 region.

Value 7 : Calculates the reduced frequency initial value by :

$$X(7) = EXP(- \frac{ALT-96}{7})$$

This model should be used in the E region.

Once all the prescribed initial values have been calculated, additional corrections are made :

If value 2 and $P \neq 0$: CALL MODCOMP, which modifies NE , TE, TI according to the composition P.

If value 1 : multiply NE by $\frac{1+TE/TI}{2}$ and correct the NE obtained from the DEBYE length effect.

3.6 SUBROUTINE TINIT

This SUBROUTINE has no branching : it first calculates the first minimum YMIN and the first zero crossing TZER of the ACF by parabolic interpolation and then applies formula built from figures 6 and 7 of WICKW AR's report.

$$A = MAX(0,1 - 2.17 \times YMIN)$$

$$TR = \frac{1-\sqrt{A}}{0.23}$$

$$TZI = (0.4043 + \frac{2.1825}{TR+0.9632}) \frac{39}{PK_1}$$

$$TI = (TZI/TZER)^2 \times 1.E-6$$

$$TE = TI \times TR$$

3.7 SUBROUTINE VINIT.

According to EDA 79, the initial value of the wind is calculated by :

$$V = X(4) = \frac{\sum_{1}^{LFI} (I-1) \times POID(I) \times Y(2I-1) \times Y(2I)}{\sum_{1}^{LFI} (I-1)^2 \times POID(I) \times Y(2I-1)^2}$$

3.8 SUBROUTINE MODCOMP.

The correction formulas are given in "Initial values determination" of EDA 79.

3.9 SUBROUTINE REGR.

The figure 6 gives the organigram of REGR. It is a standard routine of least square fit using the gradient method (EDA 79). The basic input for this routine is the matrix S calculated in FOURIER.

ITER is the number of successful iterations.

KTER is the number of successive tries after an unsuccessful iteration (VAR > VARO). In that case, the incrementation parameter Q is reduced by the constant QDIV. The iterative process stops for ITER = 7 and KTER = 3 and IBEG is equal to 2 if the convergence is achieved, 3 if ITER > 6 and 4 if KTER ≥ 3.

3.10 SUBROUTINE SPECT.

This SUBROUTINE follows almost exactly the descriptions given in EDA 79. Only small differences in notations have been introduced to allow an easy calculation of the derivative with respect to composition. The multiplicative factors AL_j entering the definition of the $AR_{1,2}$, $AI_{1,2}$ and $BI_{1,2}$ in EDA 79, annex I, have been dropped, which makes small differences on subsequent intermediate values (i.e. DAITHE, ...).

The electron-neutral collisions have not been considered.

The routine SPECT is built for 2 ions but could be easily changed to 3 ions by increasing the arrays of 2 and 3 elements into arrays of 3 and 4 elements.

One of the 2 ions can be a negative ion. For instance, if X(5) is larger than 1, ion 1 is a negative ion ; and if X(5) is negative, ion 2 is a negative ion. The derivative with respect to X(5) is not continuous in 0 or 1. The nature of the ion (positive or negative) should be fixed in order to make it continuous ; but in that case, some values of X(5) will have no physical meaning (for instance X(5) < 0 if ion 2 is a positive ion). There is still some need of testing this point.

A simple organigram explaining the signification of the logical arrays ITRA(3), ICOL(3) is given in figure 7.

The value of the spectrum, called GS, is put in DER(4) since the derivative with respect to the wind is not needed (included simply in FOURIER). The other derivatives with respect to the parameter J , J = 1, 2, 3, 5, 6, 7, 8,9 and 10 are put in DER(J). The calculation of the spectrum is stopped when the value is less than 0.2 % of the value at the central frequency, which defines IXM, the number of points in the spectrum.

The spectrum is stored in GARSPE at the end and the array DG(32,7) is built before entering FOURIER. The first dimension in DG corresponds to the maximum number of points calculated in the spectrum, and the second dimension to the maximum value of IDV. For each frequency L , the array DER is put in DG(L, 1→7) in the order specified by IORD.

3.11 SUBROUTINE FOURIER.

The organigram of FOURIER is shown in figure 8. The algorithms for FFT of even and odd functions will not be explained : let us only say that they are the quickest way of FOURIER transforming functions with such parity ; but they have been built only for LM = 64 points and cannot be adapted for another LM. In case of large oversampling (for instance 100 kHz in the E region), the use of these routines would be a possible cause of error, since then the definition of the spectrum would be so bad that only few values (< 10) would be calculated, the others being zero. This is the reason why a slow DFT routine has also been included (A in figure 8). The formulas are the following :

1) for even functions DG, the DFT (real) is calculated by :

$$\text{GFIN}(L,\ L = 1,32) = \text{DG}(1) + 2 \sum_{K=2}^{\text{IXM}} \cos 2\pi \frac{(L-1)(K-1)}{\text{LM}}\ \text{DG}(K)$$

2) For odd functions DG, the imaginary part of the DFT is :

$$GFIN(L,\ L=1,32) = -2 \sum_{K=2}^{IXM} \sin 2\pi\frac{(L-1)(K-1)}{LM}\ DG(K)$$

The definition of VAR is the following :

$$VAR=\sqrt{\frac{\sum_{1}^{LFI} (Y(2L-1)-GFIN(L))^2 POID(L)+(Y(2L)-ACFIM)^2 POID(L)}{2\ LFI\ -\ ID}}$$

In most cases, this value is close to 1 after the convergence. In case of a value larger than for instance 2 or 3, one should doubt the quality of the fit. In part B of figure 8, one also builds the array DEVI which will be used in D to calculate the vector $\vec{V}$ (referring to notations of EDA 79 following the EISCAT technical report on data analysis by T . HAGFORS in 1978).

In C, one builds the imaginary part of the derivative of ACF with respect to X(4), by simply multiplying the ACF (real part) by the time lag.

Finally, in D, the EDA 79 matrix S (IDxID) is calculated and the vector V is put in an additional column of the matrix S, i.e. S(., ID+1). What is referred to as inversion of matrix S in the organigram of REGR (figure 6) performs in fact the resolution of the system

$$S\delta X = V$$

The input is the array S = SV and the output, in the same array S, contains $S^{-1}\delta X$.

3.12 SUBROUTINE SORTIE.

This subroutine, called by REGR, first transforms the reduced parameters X(1), X(2), X(3) into the parameters NE, TE, TI and calculates the errors on ANE, TE and TI, put in EX(1), EX(2), EX(3). The value of the ion neutral collision frequency COL is calculated and the error DCOL. In order to calculate the errors on NE, TE, TI, COL, one needs to know not only the diagonal terms of the matrix S^{-1} but also coupling factors between X(1) and X(3), X(2) and X(3), X(3) and X(7), called COUF 13, COUF 23 and COUF 73, and equal to S^{-1}_{13} , S^{-1}_{23} , S^{-1}_{73} , which are saved in the common /COU/ in REGR before entering SORTIE.

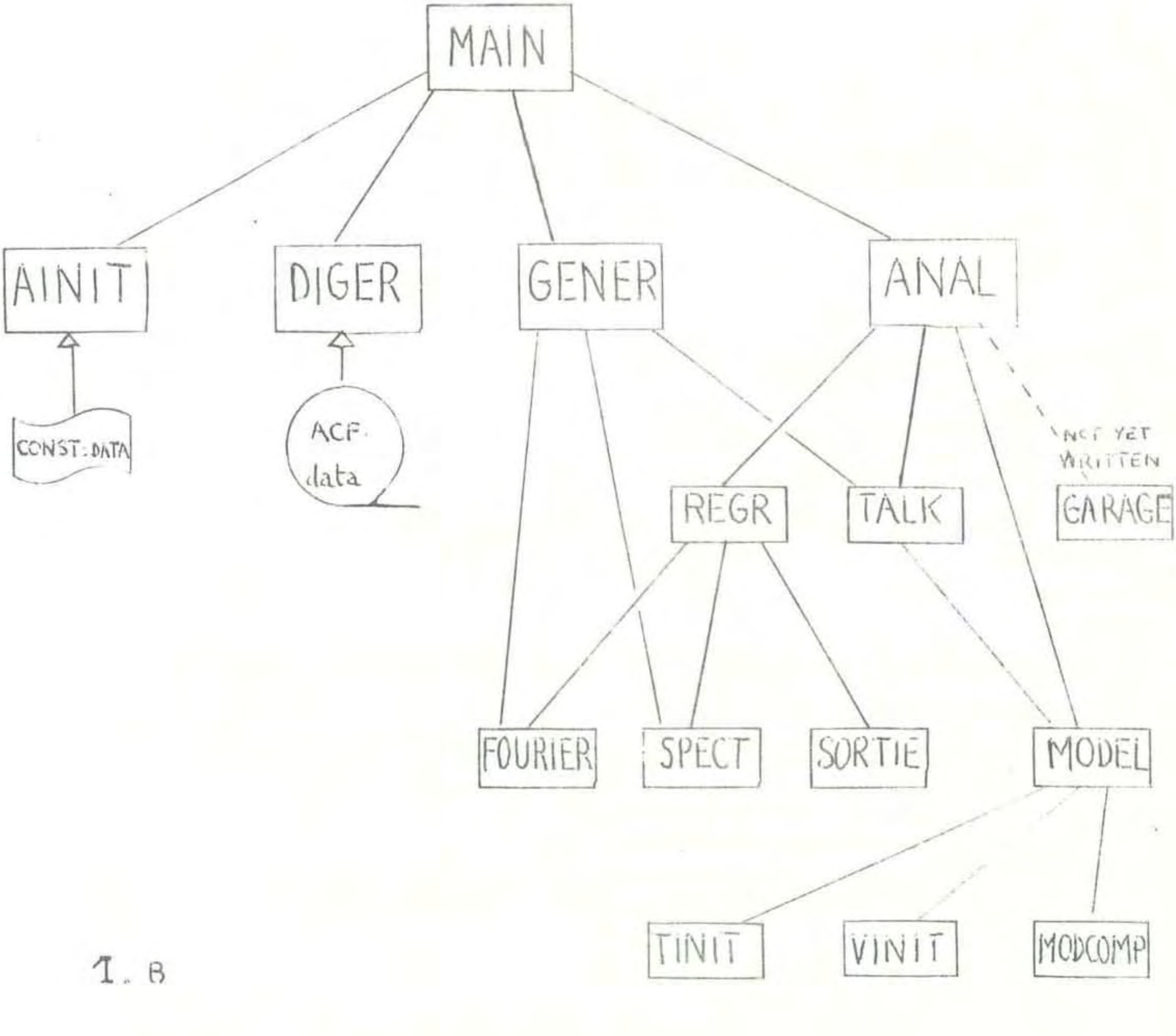
4. PROGRAM GENER.

This program has been made for testing purposes. A few conditional compilations are included. If not compiled with - , this routine generates random values of NE, TE, TI, V, TINT (integration time) and SNR, calculates the corresponding ACF as well as their σ^2 (in case of a single pulse transmission) and thus builds the arrays XCOR and ISIG2 which can be then analysed like real data, or put in a file which will be considered later as an output of the autorcorrelator for Real Time operation testing purposes.

If compiled with - , GENER is a conversational program, which first asks basic parameters : ALT, NE, TE, TI, V, P, TINT, ANFREQ (number of frequency channels), PULENG (length of the pulse in μs), PUREP (pulse repetition period in ms) SAMFRE (sampling frequency in kHz), SNR (signal to noiseratio), OVSAMP (oversampling factor, the width of the filter being SAMFRE/OVSAMP) and NLAG (the number of lags which will be used in a subsequent analysis, limited to 32).

Then it calls TALK in which one can specify other parameter values, ion mass and LM. Finally the calculation is the same as was said previously : XCOR and ISIG2 are built and outprints are given, according to the value of the parameter IESSAI.

I. A



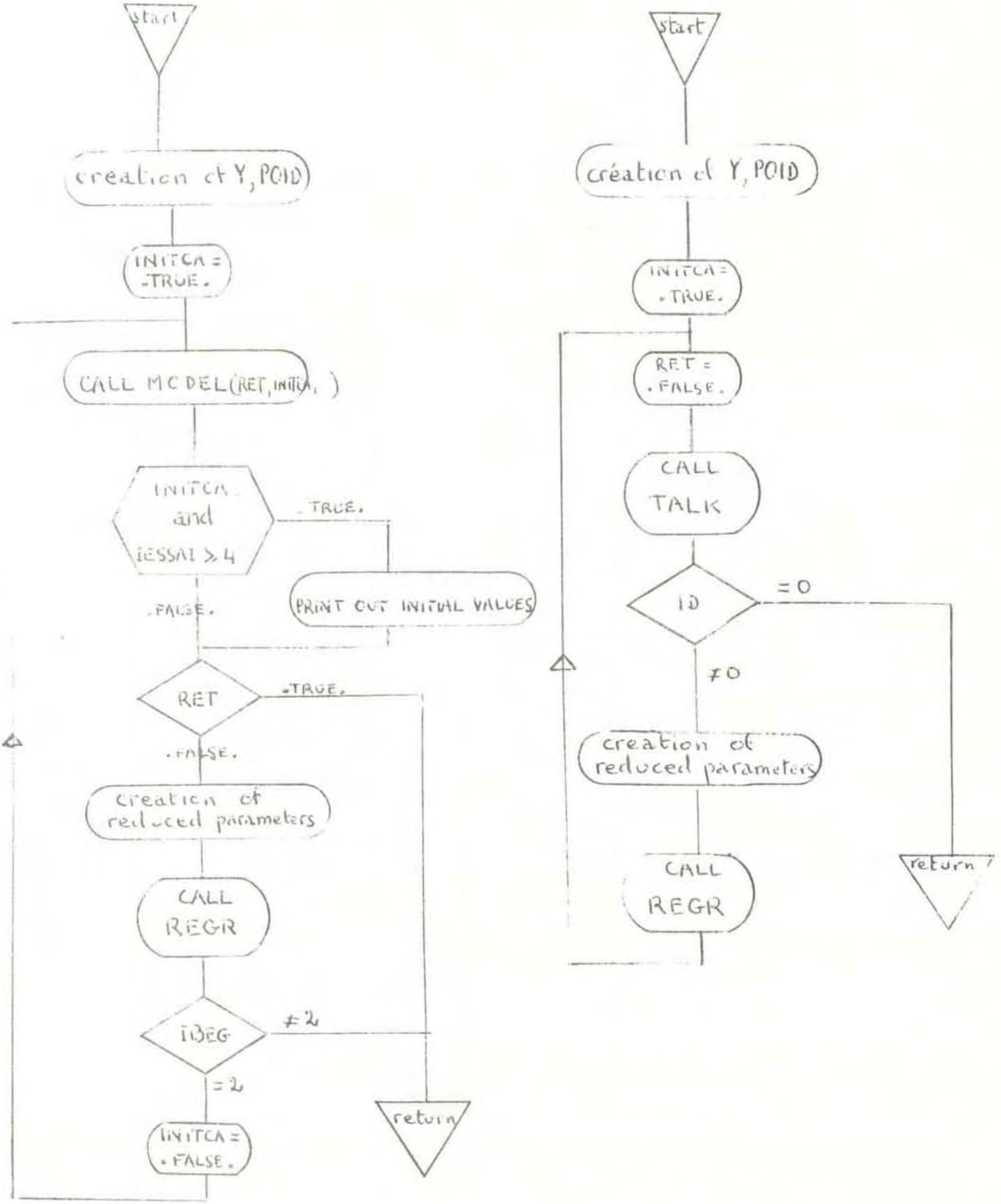
I. B

Fig. 1. Hierarchy of the programs

List of MODEL: DATA


```
TRANSMISSION FREQUENCY                        1290,
DEFAULT NB OF PARAM IN THE FIT                4,
DEFAULT ORDER OF PARAM                        4,1,3,2,5,6,7,8,9,10,
DEFAULT ION MASS                              31.,
NB OF POINTS IN SPECTRUM (FOR DFT)            64,
INDEX FOR OUTPUTS IN ANAL (1 TO 5)            1,
NB OF ALTITUDE ZONES                          3,
               =*=*=*=*=*=*=*=*
               ZONE NUMBER 1 : ALTITUDE<  116,KM
ION MASS                                      31.,
DEFAULT PARAM            1       NE=          1.,          (10**5 CM-3)
                         2       TE=          300.,        (K)
                         3       TI=          300.,        (K)
                         4       VI=          0.,          (HZ)
                         5       1-[O+]/NE=1.,
                         6       VE-VI=       0.,          (HZ)
                         7       COLL. FR.=0.,
                         8       TI2/TI1=     1.,
                         9       VI2-VI1=     0.,          (HZ)
                        10       CF2/CF1=     1.,
INIT. VAL.: NB AND DESIGNATION OF PAR.  3,1,3,7,4,
MAXIMUM NB OF FITS                            2,
ION COMP. PARAM. FOR THOSE FITS               1.,1.,1., (USED IF MODEL COMPILED WITH P
FIT 1 : NB AND DESIGNATION OF PARAM.          2,1,3,4,
FIT 2 : NB AND DESIGNATION OF PARAM.          3,1,3,7,
               =*=*=*=*=*=*=*=*
               ZONE NUMBER 2 : ALTITUDE<  220,KM
ION MASS                                      31.,
DEFAULT PARAM            1       NE=          5.,          (10**5 CM-3)
                         2       TE=          1000.,
                         3       TI=          500.,        (K)
                         4       VI=          0.,          (HZ)
                         5       1-[O+]/NE=.5,
                         6       VE-VI=       0.,          (HZ)
                         7       COLL. FR.=0.,
                         8       TI2/TI1=     1.,
                         9       VI2-VI1=     0.,          (HZ)
                        10       CF2/CF1=     1.,
INIT. VAL.: NB AND DESIGNATION OF PAR.  3,1,2,5,
MAXIMUM NB OF FITS                            1,
ION COMP. PARAM. FOR THOSE FITS               1.,1., (USED IF MODEL COMPILED WITH P)
FIT 1 : NB AND DESIGNATION OF PARAM.          3,1,2,3,4,
               =*=*=*=*=*=*=*=*
               ZONE NUMBER 3 : ALTITUDE<  1000,KM
ION MASS                                      4.,
DEFAULT PARAM            1       NE=          3.,          (10**5 CM-3)
                         2       TE=          2000.,       (K)
                         3       TI=          1000.,       (K)
                         4       VI=          0.,          (HZ)
                         5       1-[O+]/NE=0.,
                         6       VE-VI=       0.,          (HZ)
                         7       COLL. FR.=0.,
                         8       TI2/TI1=     1.,
                         9       VI2-VI1=     0.,          (HZ)
                        10       CF2/CF1=     1.,
INIT. VAL.: NB AND DESIGNATION OF PAR.  2,1,2,4,
MAXIMUM NB OF FITS                            1,
ION COMP. PARAM. FOR THOSE FITS               0.,0.,0., (USED IF MODEL COMPILED WITH P
FIT 1 : NB AND DESIGNATION OF PARAM.          3,1,2,3,4,
```

Figure 2



3.1 3.2

Fig 3 ANAL

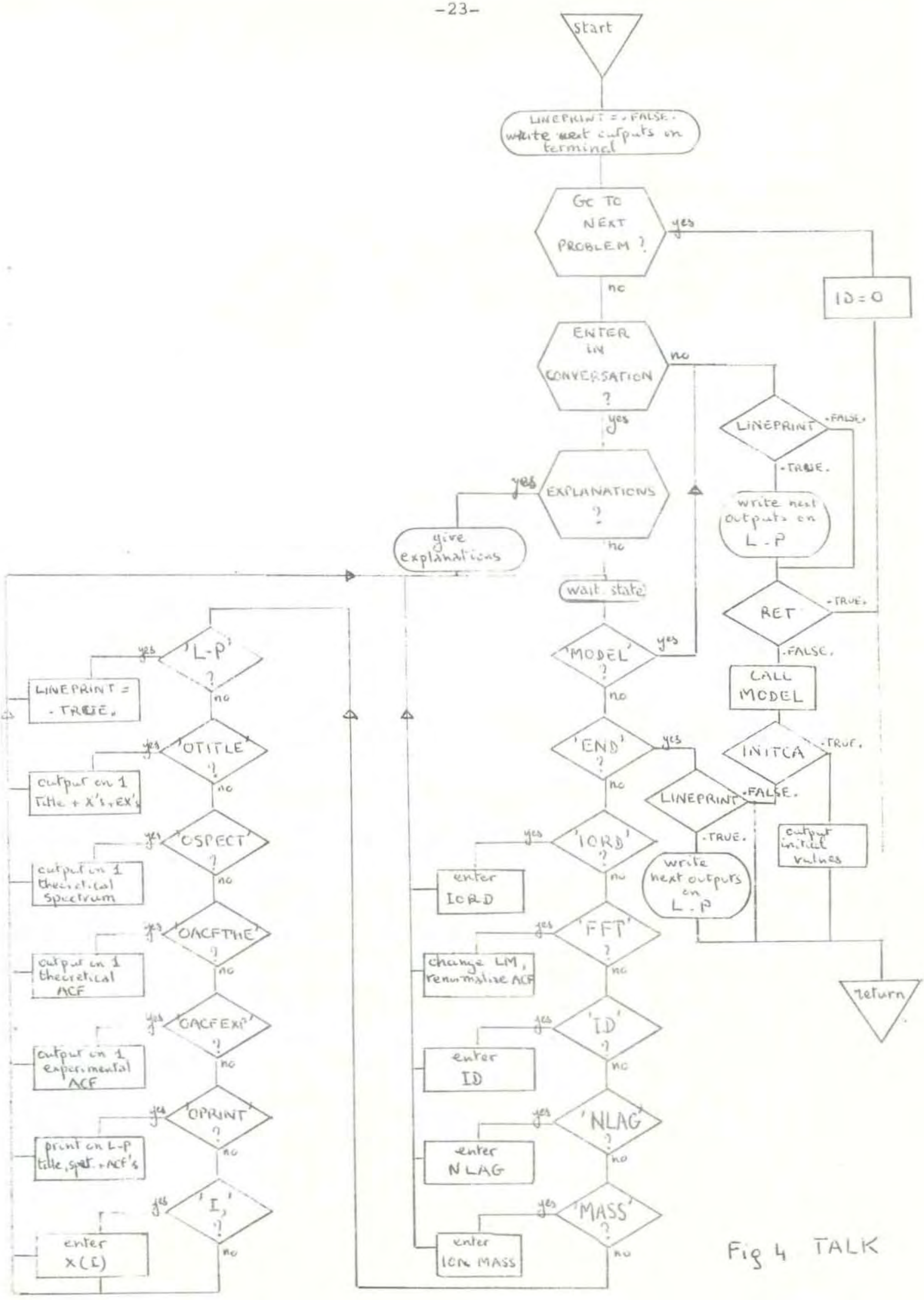

Start
LINEPRINT = .FALSE.
write next outputs on terminal
GO TO NEXT PROBLEM ?
yes
no
ID = 0
ENTER IN CONVERSATION ?
no
yes
LINEPRINT
.FALSE.
.TRUE.
write next outputs on L-P
RET
.TRUE.
.FALSE.
CALL MODEL
INITCA
.TRUE.
.FALSE.
output initial values
EXPLANATIONS ?
yes
no
give explanations
wait state
'MODEL' ?
'END' ?
LINEPRINT
.TRUE.
write next outputs on L-P
'IORD' ?
enter IORD
'FFT' ?
change LM, renormalize ACF
'ID' ?
enter ID
'NLAG' ?
enter NLAG
'MASS' ?
enter ion MASS
return
'L-P' ?
LINEPRINT = .TRUE.
'OTITLE' ?
output on 1 title + X's + EX's
'OSPECT' ?
output on 1 theoretical spectrum
'OACFTHE' ?
output on 1 theoretical ACF
'OACFEXP' ?
output on 1 experimental ACF
'OPRINT' ?
print on L-P title, spectr. + ACF's
'I,' ?
enter X(I)

Fig 4 TALK

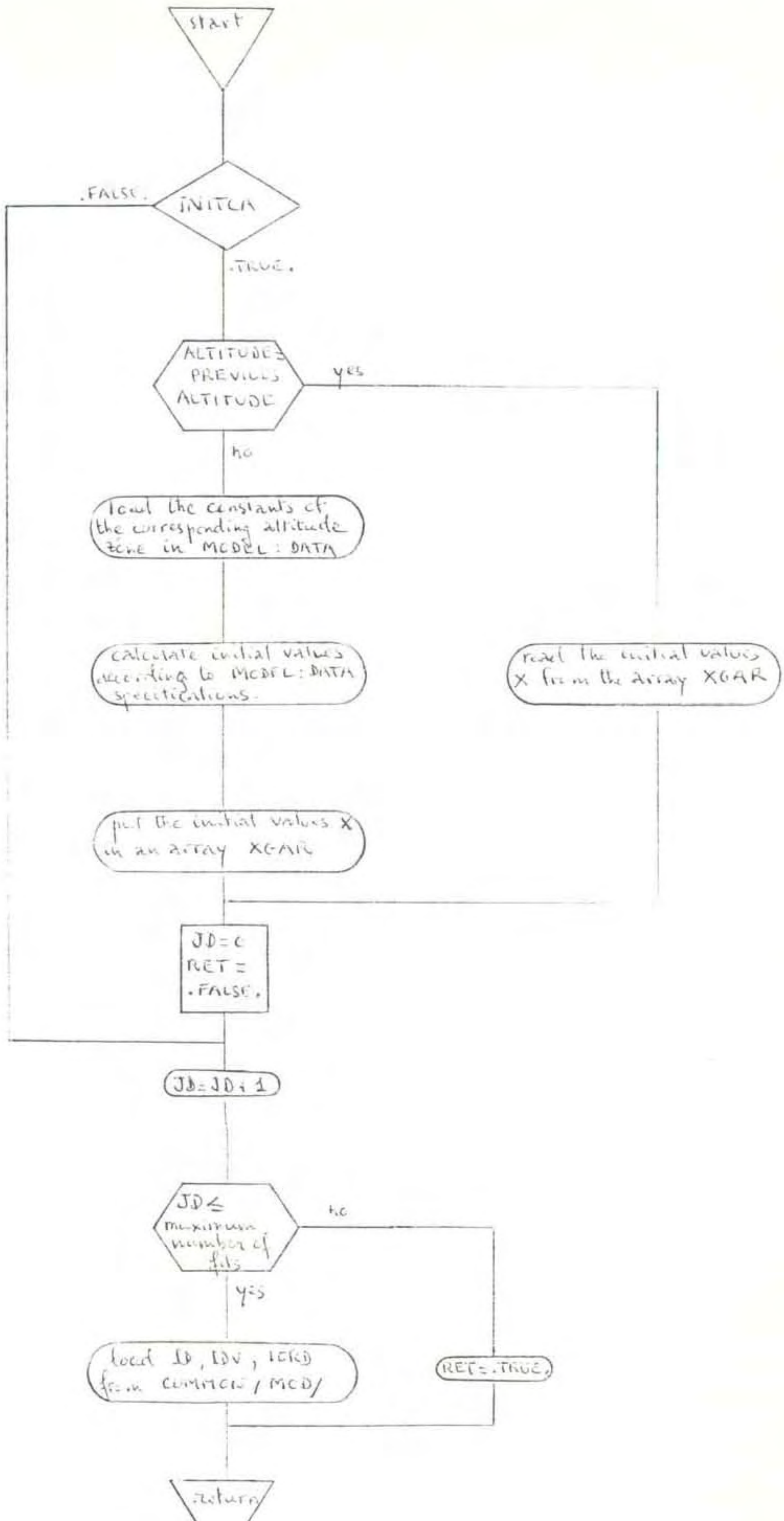

start
INITCH
.FALSE.
.TRUE.
ALTITUDE= PREVIOUS ALTITUDE
yes
no
read the constants of the corresponding altitude zone in MODEL:DATA
calculate initial values according to MODEL:DATA specifications.
read the initial values X from the array XGAR
put the initial values X in an array XGAR
JD=0
RET= .FALSE.
JD=JD+1
JD≤ maximum number of fits
no
yes
load ID, IDV, ICRD from COMMON/MOD/
RET=.TRUE.
return

Fig 5. MODEL

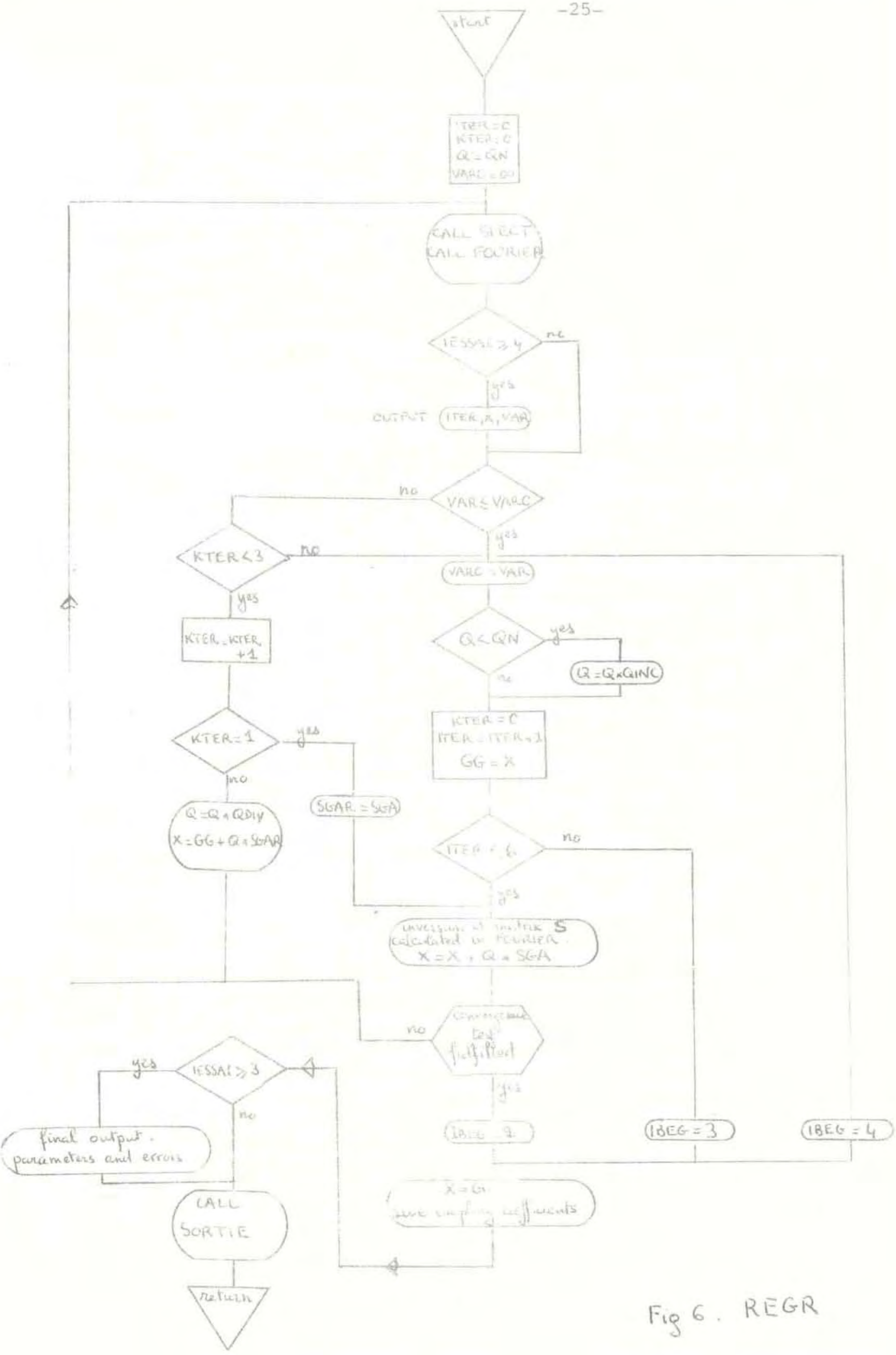

start
ITER = 0
KTER = 0
Q = QN
VARC = 00
CALL SELECT
CALL FOURIER
IESSAI ≥ 4
no
yes
OUTPUT
ITER, X, VAR
VAR ≤ VARC
no
yes
KTER < 3
no
yes
VARC = VAR
KTER = KTER + 1
Q < QN
yes
Q = Q x QINC
no
KTER = 0
ITER = ITER + 1
GG = X
KTER = 1
yes
no
SGAR = SGA
Q = Q x QDIV
X = GG + Q x SGAR
no
yes
calculated in FOURIER
X = X + Q x SGA
no
test
fulfilled
yes
IESSAI ≥ 3
yes
no
final output
parameters and errors
IBEG = 3
IBEG = 4
save coupling coefficients
CALL
SORTIE
return

Fig 6. REGR

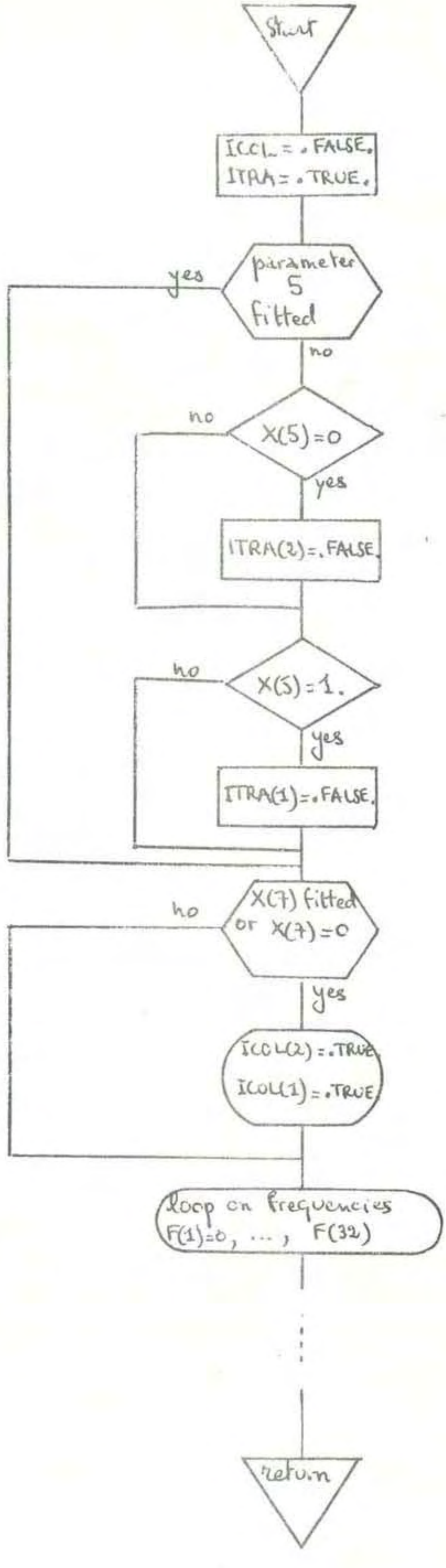

Start
ICOL = .FALSE.
ITRA = .TRUE.
parameter 5 fitted
yes
no
X(5)=0
no
yes
ITRA(2)=.FALSE.
X(5)=1.
no
yes
ITRA(1)=.FALSE.
X(7) fitted or X(7)=0
no
yes
ICOL(2)=.TRUE.
ICOL(1)=.TRUE.
loop on frequencies
F(1)=0, ..., F(32)
return

Fig 7. SPECT

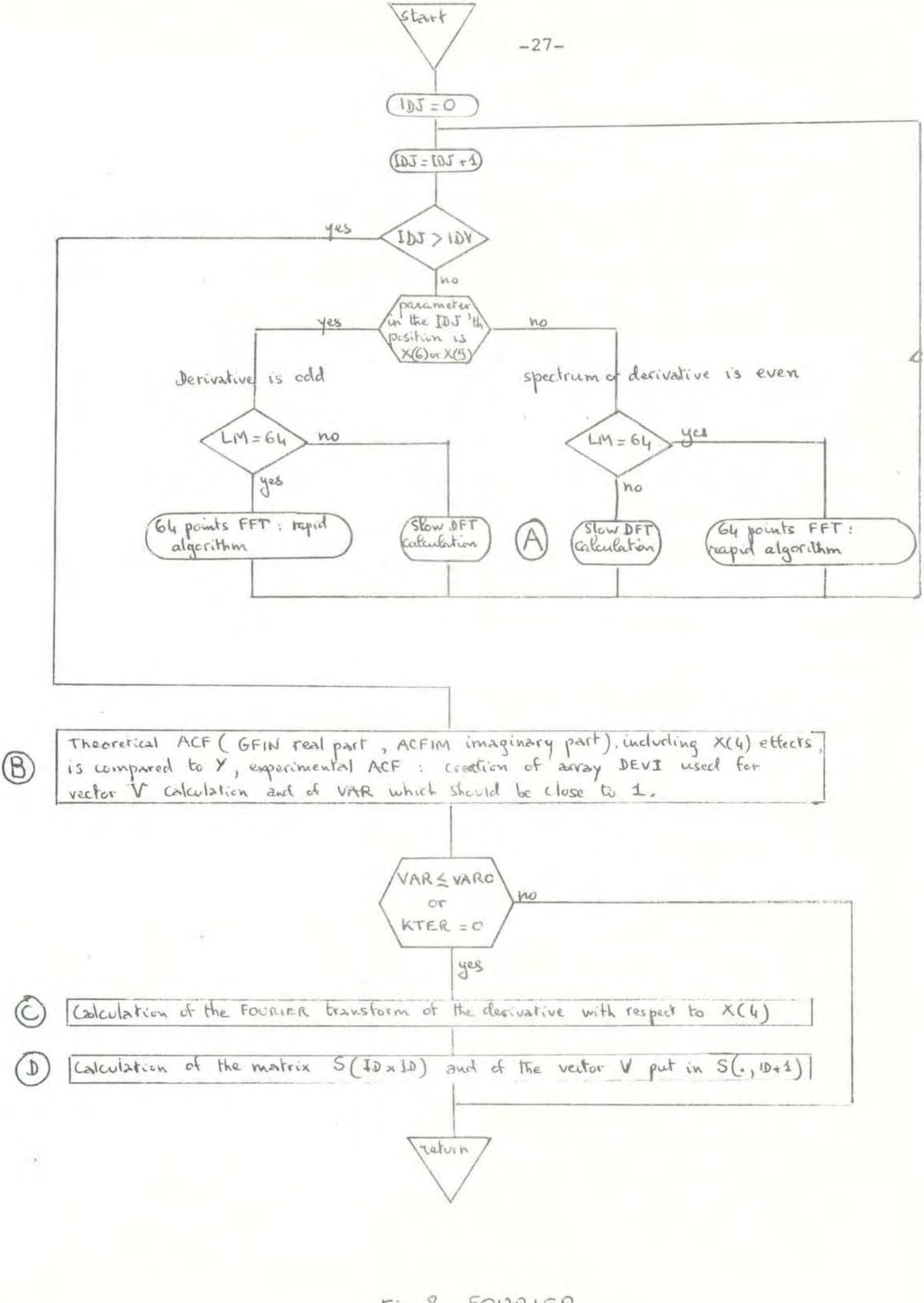


Fig 8. FOURIER

```
C *** THIS PROGRAM GENERATES CONSTANTS USED IN THE ANALYSIS
C       PROGRAM AND WRITE THEM ON FILE CONST:DATA
C
C             LN,LP,LQ,LR                    FOR FFT ALGORITHM (FIXED)
C             CO,SI                          FOR FFT ALGORITHM (FIXED)
C             IPAR                           PARITY OF PARAMETERS
C             QN,CONV,QDIV,QINC              FOR CONTROL OF THE ITERATIVE PROCESS
C
      COMMON/SPE/F(32),FD,DG(32,7),POID(32),X(10),EX(10),GFIN(32),
     *ITALK,LFIN,LM,IBEG,ID,IDV,LNOISE,IX(32),IXM,ANE,TE,TI,V,P,IA,Y(64)
     *,VAR,VARO,KTER,S(7,7),ATTEN(32),GARSPE(32),IPAR(10),DER(10)
      LOGICAL LNOISE
      COMMON/CONST/AMMIN,RI,BFE,BFO,DEBE,DEBO,BFEO,BE,BO,IESSAI,DEBOP,
     *RPI,ITERM,DEPI,LN,LP,LQ,LR,LFI,NSG,QN,CONV,QDIV,QINC,PK1
      COMMON/MOD/NBALT,KAL(4),AMMINO(4),NDEF(4),XPAR(10,4),KORD(10,5,4)
     * ,PIN(5,4),KD(5,4),KDV(5,4),KGAR(5,4),INITP(4,4),NINIT(4)
      DIMENSION IORD(10,2),CO(16),SI(8,8),JORD(10)
      LOGICAL L06,L09,N04,N06,N09
      CHARACTER CH*30
      DATA IPAR/1,1,1,2,1,2,1,1,2,1/,IBEG/0/,
     * LN/32/,LP/16/,LQ/17/,LR/34/,QN/1./,CONV/.25/,QDIV/.5/,QINC/2./
      OPEN(4,FILE='CONST:DATA',ACCESS='W',ERR=100)
      OPEN(3,FILE='MODEL:DATA',ACCESS='R',ERR=100)
200   FORMAT(39X,A)
      CH=''
      READ(3,200)CH
      READ(CH,*)FREQ
      CH=''
      READ(3,200)CH
      READ(CH,*)ID
      CH=''
      READ(3,200)CH
      READ(CH,*)(IORD(I,1),I=1,10)
      DO 1 I=1,10
1     IORD(IORD(I,1),2)=I
      CH=''
      READ(3,200)CH
      READ(CH,*)AMMIN
      RI=SQRT(AMMIN/16.)
      CH=''
      READ(3,200)CH
      READ(CH,*)LM
      CH=''
      READ(3,200)CH
      READ(CH,*)IESSAI
      CH=''
      READ(3,200)CH
      READ(CH,*)NBALT
      DO 10 INDAL=1,NBALT
      READ(3,200)CH
      CH=''
      READ(3,200)CH
      READ(CH,*)KAL(INDAL)
      CH=''
      READ(3,200)CH
      READ(CH,*)AMMINO(INDAL)
      DO 20 J=1,10
      CH=''
      READ(3,200)CH
20    READ(CH,*) XPAR(J,INDAL)
```

```
      CH=''
      READ(3,200)CH
      READ(CH,*)NIN,(INITP(I,INDAL),I=1,NIN)
      NINIT(INDAL)=NIN
      CH=''
      READ(3,200)CH
      READ(CH,*)ND
      NDEP(INDAL)=ND
      CH=''
      READ(3,200)CH
      READ(CH,*)(PIN(N,INDAL),N=1,ND)
      L06=.FALSE.
      L09=.FALSE.
      IF(XPAR(6,INDAL).NE.0.)L06=.TRUE.
      IF(XPAR(9,INDAL).NE.0.)L09=.TRUE.
      DO 30 N=1,ND
      CH=''
      READ(3,200)CH
      READ(CH,*)KDEP,(JORD(K),K=1,KDEP)
      NO4=.TRUE.
      NO6=.TRUE.
      NO9=.TRUE.
      DO 31 K=1,KDEP
      IF(JORD(K).EQ.4)NO4=.FALSE.
      IF(JORD(K).EQ.6)NO6=.FALSE.
      IF(JORD(K).EQ.9)NO9=.FALSE.
31    CONTINUE
      KDEPV=KDEP
      IF(.NOT.NO4) GO TO 32
      KDEPV=KDEPV+1
      JORD(KDEPV)=4
32    IF(.NOT.(L06.AND.NO6)) GO TO 33
      KDEPV=KDEPV+1
      JORD(KDEPV)=6
33    IF(.NOT.(L09.AND.NO9)) GO TO 34
      KDEPV=KDEPV+1
      JORD(KDEPV)=9
34    L=KDEPV
      DO 35 I=1,10
      DO 36 K=1,KDEPV
      IF(JORD(K).EQ.I) GO TO 35
36    CONTINUE
      L=L+1
      JORD(L)=I
35    CONTINUE
      KD(N,INDAL)=KDEP
      KDV(N,INDAL)=KDEPV
      DO 37 I=1,10
37    KORD(I,N,INDAL)=JORD(I)
30    CONTINUE
10    CONTINUE
      PI=3.14159265
      A=PI/LN
      B=2.
      DO 2 I=2,LP
      K=I-1
2     CO(K)=COS(A*K)*B
      DO 3 L=1,8
      DO 3 M=1,8
3     SI(L,M)=B*COS(A*(2*L-1)*(2*M-1))
```

```
      RPI=SQRT(PI)
      AK=1.380662E-23
      EPSZER=1./(36.E9*PI)
      EL=1.6021892E-19
      AME=.9109534E-30
      BE=2.*PI*SQRT(AME/(2.*AK))
      BO=BE*SQRT(16.*1836.15152)
      PK1=4.*PI*FREQ/299.792458
      DEBE=(EL/BE)**2/(AK*EPSZER)
      DEBO=(EL/BO)**2/(AK*EPSZER)
      BFEO=BE/BO
      DEBOP=RPI*DEBO
      WRITE(4)LM,LN,LP,LQ,LR,IBEG,IESSAI,IPAR,IORD,QN,CONV,QDIV,QINC
     * ,RPI,BFEO,DEBE,DEBO,BE,BO,DEBOP,AMMIN,RI,CO,SI,FREQ,PK1,
     * NBALT,KAL,AMMINO,NDEP,XPAR,KORD,PIN,KD,KDV,KGAR,INITP,NINIT
      OUTPUT(1) ' CONSTANTS READ'
100   CLOSE (4)
      STOP
      END
```

MAIN

```
C
C           *****  MAIN PROGRAM IN BACKGROUND  *****
C
C      CONDIT. COMPILATION    H        TO MAKE AN HISTOGRAM OF THE PARAM
C                             ‡ DUMP ARRAY RESULT ON RESAL DATA
C
C
      DOUBLE INTEGER ITIME,JTIME,TUSED,XCOR(64),NCALL,ISIG2(32),
     * XCORO(64),IDAT(24)
      COMMON/SPE/F(32),FD,DG(32,7),POID(32),X(10),EX(10),GFIN(32),
     *ITALK,LFIN,LM,IBEG,ID,IDV,LNOISE,IX(32),IXM,ANE,TE,TI,V,P,IA,Y(64)
     *,VAR,VARO,KTER,S(7,7),ATTEN(32),GARSPE(32),IPAR(10),DER(10)
      CHARACTER ACHAR*12
      LOGICAL GEN,LNOISE,WRIGE
      DIMENSION RESULT(128),IRES(16,2)
      WRITE(1,4)
    4 FORMAT(' SEED, NB PB ANALYZED',/)
      INPUT(1) SEED,NCAMA
CH    CALL START(SEED)
      WRITE(1,'(12H USE GENER ?,/)')
      ACHAR=''
      INPUT(1) ACHAR
      GEN=.FALSE.
      IF(ACHAR(1:1).EQ.'N') GO TO 9
      GEN=.TRUE.
      CALL START(SEED)
      WRITE(1,'(12H ADD NOISE ?,/)')
      INPUT(1) ACHAR
      LNOISE=.TRUE.
      IF(ACHAR(1:1).EQ.'N') LNOISE=.FALSE.
      WRITE(1,'(29H SAVE GENERATED ACF ON FILE ?,/)')
      INPUT(1) ACHAR
      WRIGE=.FALSE.
      IF(ACHAR(1:1).EQ.'N') GO TO 9
      WRIGE=.TRUE.
      WRITE(1,6)
    6 FORMAT(' NAME OF THE FILE READ',/)
      ACHAR=''
      READ(1,7)ACHAR
    7 FORMAT(A)
      OPEN(3,FILE=ACHAR,ACCESS='WA',ERR=100)
      OUTPUT(1) ' FILE OPENED'
9     WRITE(1,'(11H USE TALK ?,/)')
      INPUT(1) ACHAR
      ITALK=0
      IF(ACHAR(1:1).EQ.'Y') ITALK=1
      NCALL=0
      CALL AINIT
C‡    OPEN(4,FILE='RESAL:DATA',ACCESS='W',ERR=100)
    1 CONTINUE
      IF(.NOT.GEN) GO TO 3
      CALL GENER(NCALL,XCOR,ISIG2,IDAT)
      IF(WRIGE)WRITE(3)IDAT,XCOR,ISIG2
      GO TO 2
C     READ(3)IDAT,XCOR,ISIG2
3     CALL DIGER(IDAT,XCOR,ISIG2,IRES)
    2 JTIME=TUSED(I)
CH    DO 10 I=1,64
CH10  XCORO(I)=XCOR(I)
CH8   DO 11 I=1,64
```

```
CH      CALL GAUSS(RNDM)
CH11    XCOR(I)=XCORO(I)+RNDM*1.E+3*SQRT(FLOAT(ISIG2((I+1)/2)))
        IF(NCALL.GT.NCAMA) GO TO 100
        CALL ANAL(NCALL,XCOR,ISIG2,IDAT,RESULT,IRES)
        NCALL=NCALL+1
CH      GO TO 8
C÷      WRITE(4)RESULT
        IF(.NOT.GEN) GO TO 1
        ITIME=TUSED(I)-JTIME
        OUTPUT(1)ITIME
        GO TO 1
  100   STOP
        END
```

DIGER

```
C
C
C       DIGER READS THE CRIM OR CROS TAPES FROM CHATANIKA, AND REWRITE THEM
C       IN A FORM SUITABLE FOR -ANAL- IN ARRAYS IDAT,XCOR,ISIG2.
C
C       CONDITIONAL COMPILATIONS :       C          USED AS SUBROUTINE OF MAIN
C                                        (OTHERWISE CREATES THE FILE 4)
C                                        S          SINGLE PULSE CROS
C                                        M          MULTIPULSE CRIM
C                                        #          WRITE RESULTS ON RESAL:DATA
CC      SUBROUTINE DIGER(IDAT,XCOR,ISIG2,IRES)
        DIMENSION ATTEN(32),JHEU(8),KHEU(8),IRES(16,2),JRES(15,18,2),
       *IDATIN(4),IDATFI(4),IWRONG(16)
        CHARACTER ACHAR*80
        DOUBLE INTEGER IRECORD(750),IHEAD(148),XCOR(64),NCALL,ISIG2(32),
       * IDAT(24),IDATI,IDATF,JDAT
        COMMON/HEADER/NWORDS,NHEAD,NOTAPE,NYMD1,NYMD2,IDISH,JFILCO,NOISUB
       1 ,MAXDOF,JADIV,IREC,IRAS,IEX,IAY,ISOFF,ITAPE,IPART,ITRU,ISPA1,
       2 ISPA2,ITSYS,IPC,ICCP,ICCO,ICCM,NPTSNS,NPTSDT,JTIME(4),JAZ,JEL,
       3 KTIME(4),KAZ,KEL,MSCAN,MSCADE(2),KWPOW,NSTEMP,INTFRM,JBW,JDELAY
       4 ,JPLWD(4),JPASS(4),KONSYS,IFILTER(3),NORML,NGP,IDP,LDP,INP,LNP,
       5 ICP,LCP,NSUBP,MFRP,MFRRP,MDRP,MSSP,KBW,KDELAY,IGR,MAXMUL,IOLTEMP,
       6 IBS,NGS,LDS,INS,LAGSS,MFGO,MFRGO,MNGO,IGO,MSSO,LHT,IDENTIME,
       7 MUTTIME,IBGM,NGATM,LDGM,INGM,LAGU,LAGM,MFGM,MFRGM,MNGM,NNSM,
       8 NCSM,MASKRG(2),IRM,MPULD(2),MPLUG,LAGR,LAGUMR,MHT,NRDAT,IDATSAM
       9 ,LDATSAM,INOISAM,LNOISAM,ICALSAM,LCALSAM,IPLA(7),IDCODE,IDTAPE,
       A IACDAT,IACNOR,IFX(3),NPE,NEOF,NHVF,NSAT,ISPA3,ISPA4,ISPA5,INX(6)
        DOUBLE INTEGER NWORDS,NHEAD,NOTAPE,NYMD1,NYMD2,IDISH,JFILCO,NOISUB
       1 ,MAXDOF,JADIV,IREC,IRAS,IEX,IAY,ISOFF,ITAPE,IPART,ITRU,ISPA1,
       2 ISPA2,ITSYS,IPC,ICCP,ICCO,ICCM,NPTSNS,NPTSDT,JTIME,JAZ,JEL,
       3 KTIME,KAZ,KEL,MSCAN,MSCADE,KWPOW,NSTEMP,INTFRM,JBW,JDELAY
       4 ,JPLWD,JPASS,KONSYS,IFILTER,NORML,NGP,IDP,LDP,INP,LNP,
       5 ICP,LCP,NSUBP,MFRP,MFRRP,MDRP,MSSP,KBW,KDELAY,IGR,MAXMUL,IOLTEMP,
       6 IBS,NGS,LDS,INS,LAGSS,MFGO,MFRGO,MNGO,IGO,MSSO,LHT,IDENTIME,
       7 MUTTIME,IBGM,NGATM,LDGM,INGM,LAGU,LAGM,MFGM,MFRGM,MNGM,NNSM,
       8 NCSM,MASKRG,IRM,MPULD,MPLUG,LAGR,LAGUMR,MHT,NRDAT,IDATSAM
       9 ,LDATSAM,INOISAM,LNOISAM,ICALSAM,LCALSAM,IPLA,IDCODE,IDTAPE,
       A IACDAT,IACNOR,IFX,NPE,NEOF,NHVF,NSAT,ISPA3,ISPA4,ISPA5,INX
        EQUIVALENCE (IRECORD,NWORDS)
        EQUIVALENCE (IRECORD,IHEAD),(JHEU,JTIME),(KHEU,KTIME)
        INTEGER FLD
        ASSEMBLY FLD                                  was just tried once ; ignore
C       DATA ATTEN/1.0, .9062, .7141, .5283, .3528, .283, .4704, .5136,
        DATA ATTEN/1.0, .9062, .7141, .5283, .3528,.1703, .4704, .5136,
       1  .7425, .9693,1.0506, .9474, .7485, .5544, .3717, .1833, .0315,
       2  .0000, .0294, .1692, .3390, .5049, .6852, .8709, .9600, .8766,
       3  .6834, .4944, .3718, .3330, .3795, .5076/
        DATA NCALL/0/,NBEOF/0/,NCAMAX/200/,IDS/0/,NBALT/2/,IALTMI/0/,
       * IALTMA/0/
        ACHAR=''
        IF(IALTMA.NE.0) GO TO 2
C#      OPEN(4,FILE='(GER)RESAL:DATA',ACCESS='WA')
        WRITE(1,'(23H NB OF WRONG ACF LAGS ?,/)')
        READ(1,*)NWRONG
        WRITE(1,'(32H DESIGNATION OF WRONG ACF LAGS ?,/)')
        READ(1,*) (IWRONG(I),I=1,NWRONG)
        WRITE(1,200)
200     FORMAT(' ALTITUDE MINIMUM AND ALTITUDE MAXIMUM FOR ANALYSIS ?',/)
        READ(1,*) IALTMI,IALTMA
        WRITE(1,'(46H DIFFERENCE BETWEEN 2 SUCCESSIVE RANGES (KM) ?,/)')
```

```
      READ(1,*) IDALT
      WRITE(1,'(47H INITIAL AND FINAL DATES ?(YEAR,MONTH,DAY,HOUR),/)')
      READ(1,*) IDATIN,IDATFI
      IDATI=((IDATIN(1)*100+IDATIN(2))*100.+IDATIN(3))*100.+IDATIN(4)
      IDATF=((IDATFI(1)*100+IDATFI(2))*100.+IDATFI(3))*100.+IDATFI(4)
      WRITE(1,300)
300   FORMAT(' TAPE (0 OR 1),NB OF FILES SKIPPED, NB OF RECORDS SKIPPED
     *, NB OF FILES ANALIZED',/)
      READ(1,*) NUMBAN,NBEOF,NREC,NEOFMA
      IF(NBEOF.GE.0) WRITE(ACHAR,130) NUMBAN,NBEOF
      IF(NBEOF.LT.0) WRITE(ACHAR,132) NUMBAN,(-NBEOF)
      CALL COMND(ACHAR)
      IF(NREC.GE.0) WRITE(ACHAR,131) NUMBAN,NREC
      IF(NREC.LT.0) WRITE(ACHAR,133) NUMBAN,(-NREC)
131   FORMAT('DEV-F M-T-',I1,' ADV-REC ',I4,1H')
133   FORMAT('DEV-F M-T-',I1,' BAC-REC ',I4,1H')
130   FORMAT('DEV-F M-T-',I1,' ADV-T-E ',I4,1H')
132   FORMAT('DEV-F M-T-',I1,' REV-T-E ',I4,1H')
      CALL COMND(ACHAR)
      ACHAR=''
      WRITE(ACHAR,'(4HM-T-,I1)') NUMBAN
      OPEN(9,FILE=ACHAR,ACCESS='R')
      NBEOF=NBEOF+1
2     IF(IDS.NE.0) GO TO 30
1     ISTAT=MAGTP(0,IRECORD,9,2100,NMOLU)
      IF(ISTAT.NE.3) GO TO 3
      NBEOF=NBEOF+1
      NREC=0
      IF(NBEOF.GT.NEOFMA) STOP
3     IF(ISTAT.NE.0.OR.NMOLU.LT.500.OR.NMOLU.GT.1500) GO TO 1
      IF(LDS.GT.20.OR.LDGM.GT.20.OR.IABS(INX(6)).GT.2000) GO TO 1
      IF(LDS.LT.0.OR.LDGM.LT.0) GO TO 1
      IDAT(16)=10*FLD( 8,6,JHEU(1))+FLD( 0,4,JHEU(2))
      IDAT(17)=10*FLD( 4,6,JHEU(2))+FLD(10,6,JHEU(2))
      IDAT(18)=10*FLD( 8,6,JHEU(3))+FLD( 0,4,JHEU(4))
      IDAT(19)=FLD( 8,6,JHEU(5))*10+FLD(0,4,JHEU(6))
      JDAT=((IDAT(16)*100+IDAT(17))*100+IDAT(18))*100.+IDAT(19)
      IF(JDAT.LT.IDATI) GO TO 1
      IF(JDAT.GT.IDATF) STOP
      IDAT(20)=FLD( 4,6,JHEU(6))*10+FLD(10,6,JHEU(6))
      IDAT(22)=FLD( 4,6,KHEU(6))*10+FLD(10,6,KHEU(6))
      IDAT(23)=FLD( 8,6,KHEU(7))*10+FLD(0,4,KHEU(8))
      IDAT(21)=FLD( 8,6,JHEU(7))*10+FLD(0,4,JHEU(8))
      IF(IABS(IDAT(22)-IDAT(20)).LT.4) GO TO 1
      NREC=NREC+1
      IDAT(12)=10000./INTFRM
      IDAT(7)=NPTSDT/IDAT(12)
      IDAT(9)=JPLWD(3)/MSSO
CM    IDAT(9)=32
      IDAT(13)=50
      IDAT(11)=0
      IDAT(14)=1.E5/MSSO
      WRITE(1,400)NBEOF,NREC,(IDAT(K),K=16,21),IDAT(19),IDAT(22),IDAT(23
     *),IDAT(7),IDAT(9),IDAT(11),IDAT(12),IDAT(14)
400   FORMAT(' ******** FILE NB',I3,'   REC. NB',I4,'   DATE:',3I3,
     *' FROM ',3I2,' TO ',3I2,/,' IDAT(7,9,11,12,14)=',5I5)
      ANIP=1.E+6/NPTSDT
CM    LDS=LDGM
30    IDS=IDS+1
C#    IF(IDS.EQ.1) GO TO 31
```

```
C#        IF(ISAVE.LT.1.OR.ISAVE.GT.15) GO TO 33
C#        DO 32 I=1,2
C#        DO 32 J=1,16
C#32      JRES(ISAVE,J,I)=IRES(J,I)
C#        JRES(ISAVE,17,1)=IDAT(8)      (= 100 x SNR)
C#        JRES(ISAVE,18,1)=IDAT(10)     (~ raw densities)
C#        JRES(ISAVE,17,2)=IDAT(15)     (= range in km)
C#        JRES(ISAVE,18,2)=IDAT(24)     (= number)
33        IF(IDS.LE.LDS) GO TO 31
C#        WRITE(4)NBEOF,NREC,(IDAT(K),K=16,23),IDAT(7),IDAT(9),IDAT(11)
C#       *,IDAT(12),IDAT(14),JRES
          IDS=0
          GO TO 1
31        IDAT(8)=IRECORD(INX(3)+IDS-1)/100.
          UNSNR=100./IDAT(8)
          IDAT(10)=IRECORD(INX(2)+IDS-1)*5.E-3
          IDAT(15)=IRECORD(INX(1)+IDS-1)/100.
          WRITE(1,100)IDAT(8),IDAT(10),IDAT(15),IDAT(24)
100       FORMAT('  SNR=',I3,'/100, POWER=',I5,'  ALT=',I4,' KM, NUM=',I4)
          IRES(9,1)=0
          IRES(9,2)=0
          ISAVE=(IDAT(15)-IALTMI)/IDALT+1
          IF(IDAT(10).LT.10) GO TO 30
          IF(IDAT(15).LT.IALTMI.OR.IDAT(15).GT.IALTMA) GO TO 30
          NCALL=NCALL+1
          IDAT(24)=NCALL
          JWRONG=1
          DO 20 ISG=1,IDAT(9)
CS        XCOR(2*ISG-1)=IRECORD(INX(6)+32*(IDS-1)+ISG-1)*32/(33-ISG)
CM        ISIG2(ISG)=0.
CM        IF(ATTEN(ISG).LE.0.)GO TO 25
          IF(ISG.NE.IWRONG(JWRONG)) GO TO 26
          JWRONG=JWRONG+1
          GO TO 25
26        CONTINUE
CM        XCO=IRECORD(INX(6)+32*(IDS-1)+ISG-1)/ATTEN(ISG)
CM        XCOR(2*ISG-1)=XCO
CM        IF(ISG-1)21,21,22
CM21      A=1.+2.*UNSNR*(1.+UNSNR)
CM        ISIG2(ISG)=ANIP*A
CM        GO TO 25
CM22      IF(ISG-7)23,24,23
CM24      A=9.+2.*UNSNR*(3.+UNSNR)
CM23      ISIG2(ISG)=ANIP*.5*(1.E-12*XCO*XCO+A/ATTEN(ISG)**2)
CM25      CONTINUE
20        XCOR(2*ISG)=0.
CC        GO TO 10
          WRITE(4)IDAT,XCOR,ISIG2
10        CONTINUE
CC        RETURN
          IF(NCALL.EQ.NCAMAX)STOP
          GO TO 30
          END
```

ANAL

```
C
C       CONDIT. COMPIL.              -           BACKGROUND
C                                    G           WITH GENERATION (FOR SIMUL.)
C                                    $           SMEARING BY A TRIANGLE
C                                    S           SIMPLE FORMULA FOR ISIG2 CALC.
C                                    R           USE REDUCED PARAMETERS (AL,TE/TI,..)
C                                    +(RT)       MESSAGES ON LOGMES
C                                    ?(RT)       OUTPUTS ON TERMINAL 9
C
C
C
C ******* ANALYS FIRST READ IN CONSTANTS FROM FILE CONST,103B.
C ******* THEN , ACF ARE READ, NORMALIZED, APPROX. VALUES OF THE
C ******* VARIANCE ARE CALCULATED. FINALLY THE FIT IS COMPLETED.
C
C
       SUBROUTINE AINIT
       COMMON/MOD/NBALT,KAL(4),AMMINO(4),NDEP(4),XPAR(10,4),KORD(10,5,4)
     * ,PIN(5,4),KD(5,4),KDV(5,4),KGAR(5,4),INITP(4,4),NINIT(4)
       COMMON/FOUR/CO(16),SI(8,8)
      COMMON/ORDER/IORD(10,2)
      COMMON/SPE/F(32),FD,DG(32,7),POID(32),X(10),EX(10),GFIN(32),
     *ITALK,LFIN,LM,IBEG,ID,IDV,LNOISE,IX(32),IXM,ANE,TE,TI,V,P,IA,Y(64)
     *,VAR,VARO,KTER,S(7,7),ATTEN(32),GARSPE(32),IPAR(10),DER(10)
      COMMON/CONST/AMMIN,RI,BFE,BFO,DEBE,DEBO,BFEO,BE,BO,IESSAI,DEBOP,
     *RPI,ITERM,DEPI,LN,LP,LQ,LR,LFI,NSG,QN,CONV,QDIV,QINC,PK1
 1000  IFILE=103B
       OPEN(IFILE,FILE='CONST:DATA',ACCESS='R',ERR=1010)
      CALL RESRV(IFILE,0,0)
       READ(IFILE)LM,LN,LP,LQ,LR,IBEG,IESSAI,IPAR,IORD,QN,CONV,QDIV,QINC
     * ,RPI,BFEO,DEBE,DEBO,BE,BO,DEBOP,AMMIN,RI,CO,SI,FREQ,PK1,
     * NBALT,KAL,AMMINO,NDEP,XPAR,KORD,PIN,KD,KDV,KGAR,INITP,NINIT
C-     OUTPUT(1) ' CONSTANTS READ'
      CALL RELES(IFILE,0)
      CLOSE(IFILE)
C+     CALL LOGMES('$CONSTANTS READ$')
      RETURN
 1010 CONTINUE
      CLOSE(IFILE)
C+     CALL LOGERR('$ERROR IN OPENING CONST:DATA FOR ANAL$')
      RETURN
       END
       SUBROUTINE ANAL(NUMBER,XCOR,ISIG2,IDAT,RESULT,IRES)
       COMMON/MOD/NBALT,KAL(4),AMMINO(4),NDEP(4),XPAR(10,4),KORD(10,5,4)
     * ,PIN(5,4),KD(5,4),KDV(5,4),KGAR(5,4),INITP(4,4),NINIT(4)
       DOUBLE INTEGER XCOR(64),ISIG2(32),IDAT(24)
       DIMENSION RESULT(128),IRES(16,2)
       LOGICAL RET,INITCA,IGAR,LNOISE
       COMMON/FOUR/CO(16),SI(8,8)
      COMMON/ORDER/IORD(10,2)
      COMMON/SPE/F(32),FD,DG(32,7),POID(32),X(10),EX(10),GFIN(32),
     *ITALK,LFIN,LM,IBEG,ID,IDV,LNOISE,IX(32),IXM,ANE,TE,TI,V,P,IA,Y(64)
     *,VAR,VARO,KTER,S(7,7),ATTEN(32),GARSPE(32),IPAR(10),DER(10)
      COMMON/CONST/AMMIN,RI,BFE,BFO,DEBE,DEBO,BFEO,BE,BO,IESSAI,DEBOP,
     *RPI,ITERM,DEPI,LN,LP,LQ,LR,LFI,NSG,QN,CONV,QDIV,QINC,PK1
C?    CALL RESRV(9,1,0)
C?    WRITE(9,345)XCOR,IDAT
C?345 FORMAT(1X,5I12)
C?    CALL RELES(9,1)
       SNR=.01*FLOAT(IDAT(8))
```

```
      ANIP=FLOAT(IDAT(7))*FLOAT(IDAT(12))
      NSG=IDAT(9)
      LFI=IDAT(13)
      IF(LFI.GT.NSG.AND.NSG.NE.0) LFI=NSG
41    IF(LFI.GT.32) LFI=32
      LFIN=2*LFI
      IXM=LN
      FD=FLOAT(IDAT(14))*10./LM
      BINT=1000.*IDAT(10)/FD
      DEPI=6.283185308/(FD*LM)
      DO 40 I=1,LN
40    F(I)=(I-1.)*FD
      RAV=1.
C     RAV=IDAT(24)
     PK=PK1*RAV
      BFE=BE/PK
      BFO=BO/PK
C **** RAV IS THE COSINE BETWEEN TRANSMISSION AND RECEPTION. IT SHOULD
C **** BE TRANSMITTED IN A REAL EXPERIMENT
      B=(1.+1./SNR)*XCOR(1)
      B=B*B/(2.*ANIP)
      DO 55 I=1,64
55    Y(I)=0.
      ATT=1.
      DO 1 L=1,LFI
C ******* SIMPLE MODELISATION OF THE ATTENUATION OF ACF DUE TO FINITE
C ******* PULSE LENGTH ( SINGLEPULSE)
C$    ATT=1.-(L-1.)/NSG
CG    IF(NSG.EQ.0)ATT=ATTEN(L)
      A=BINT/ATT
      Y(2*L)=A*XCOR(2*L)
      Y(2*L-1)=A*XCOR(2*L-1)
      POID(L)=0.
      IF(ISIG2(L).NE.0)POID(L)=1.E-6/(A*A*ISIG2(L))
CS    IF(NSG.NE.0) POID(L)=(NSG-L+1)/(B*A*A)
1     CONTINUE
CS    IF(NSG.NE.0) POID(1)=POID(1)*.5
      IALT=IDAT(15)
      INITCA=.TRUE.
      JGAR=1
2000  IF(ITALK.NE.0) GO TO 32
      CALL MODEL(RET,INITCA,IGAR,IDAT,ISIG2(1))
      IF(IESSAI.GE.4.AND.INITCA) WRITE(1,104) X
104   FORMAT(' P. INI.',F7.3,2F8.1,F7.1,F7.3,F8.0,F9.2,F5.2,F8.0,F5.2)
      GO TO 33
  32  RET=.FALSE.
      CALL TALK(RET,INITCA,IGAR,IDAT,ISIG2(1))
      IF(ID.EQ.0)GO TO 8
      GO TO 12
33    IF(RET)RETURN
12    CONTINUE
CR    X(2)=X(2)/X(3)
CR    X(3)=BFO/SQRT(X(3))
CR    X(1)=DEBO*X(1)*1.E11*X(3)*X(3)
C **** ID IS THE NUMBER OF FLOATING PARAM. IN THE REGRESSION.
C **** IDV (GE.ID) HAS BEEN INTRODUCED FOR THE CASE OF FIXED DIFFERENTIAL
C **** WINDS (X(6) OR X(9)) ; IN THIS CASE, THE DERIVATIVES WITH RESPECT TO
C **** THOSE PARAMETERS MUST BE COMPUTED ALTHOUGH THEY ARE FIXED.
      CALL REGR
      INITCA=.FALSE.
```

```
C       IF(IGAR)CALL GARAGE
        IGAR=.FALSE.
C THE RESULT ARRAY IS NOW CREATED WITH IALT,AJOUR,HEURE,AMIN,SEC,
C NIP(=TINT*ANFREQ/PUREP),NSG(=PULENG*SAMFRE),VAR,DEL(2),BINT,X(1,6),
C EX(1,6),Y(1,40),DG(1,20),GARSPE(1,32)
        RESULT(1)=IALT
        RESULT(2)=AJOUR
        RESULT(3)=HEURE
        RESULT(4)=AMIN
        RESULT(5)=SEC
        RESULT(6)=ANIP
        RESULT(7)=NSG
        RESULT(8)=VAR
        RESULT(9)=1./SQRT(POID(2))
        RESULT(10)=BINT
        RESULT(11)=ANE
        RESULT(12)=TE
        RESULT(13)=TI
        RESULT(14)=V
        RESULT(15)=P
        RESULT(16)=DELV
        J=16
        DO 50 I=1,6
        J=J+1
  50    RESULT(J)=EX(I)
        DO 51 I=1,40
        J=J+1
  51    RESULT(J)=Y(I)
        DO 52 I=1,20
        J=J+1
  52    RESULT(J)=GFIN(I)
        DO 53 I=1,32
        J=J+1
  53    RESULT(J)=GARSPE(I)
        RESULT(115)=IBEG
        RESULT(116)=SNR
        RESULT(117)=LFI
        IF(ITALK.EQ.0.AND.IBEG.NE.2) GO TO 8
        IRES(1,JGAR)=200*ANE
        IRES(2,JGAR)=TE
        IRES(3,JGAR)=TI
        IRES(4,JGAR)=X(4)
        IRES(5,JGAR)=X(5)*500.
        IRES(6,JGAR)=X(6)*.2
        IRES(7,JGAR)=X(7)*100.
        IF(X(7).NE.0.) X(6)=0.
        IRES(8,JGAR)=500*VAR
        IRES(9,JGAR)=IBEG          (=2 if convergence)
        IRES(10,JGAR)=200*EX(1)
        IRES(11,JGAR)=EX(2)
        IRES(12,JGAR)=EX(3)
        IRES(13,JGAR)=EX(4)
        IRES(14,JGAR)=EX(5)*500.
        IRES(15,JGAR)=EX(6)*.2
        IRES(16,JGAR)=EX(7)*100
        JGAR=2
        GO TO 2000
8       RETURN
        END
```

```
      SUBROUTINE TALK(RET,INITCA,IGAR,IDAT,IDEL2)
      DOUBLE INTEGER IDAT(24)
      COMMON/GEN/ACFIM(32)
     COMMON/ORDER/IORD(10,2)
     COMMON/SPE/F(32),FD,DG(32,7),POID(32),X(10),EX(10),GFIN(32),
    *ITALK,LFIN,LM,IBEG,ID,IDV,LNOISE,IX(32),IXM,ANE,TE,TI,V,P,IA,Y(64)
    *,VAR,VARO,KTER,S(7,7),ATTEN(32),GARSPE(32),IPAR(10),DER(10)
      LOGICAL LNOISE,L06,L09,N04,N06,N09
     COMMON/CONST/AMMIN,RI,BFE,BFO,DEBE,DEBO,BFEO,BE,BO,IESSAI,DEBOP,
    *RPI,ITERM,DEPI,LN,LP,LQ,LR,LFI,NSG,QN,CONV,QDIV,QINC,PK1
      LOGICAL INITCA,RET,IGAR,LINEPRINT,LNOISE
      CHARACTER ACHAR*80
      DATA LINEPRINT/.FALSE./
      IF(.NOT.LINEPRINT) GO TO 1
      CLOSE(1)
      OPEN(1,FILE='TERM',ACCESS='RW')
      LINEPRINT=.FALSE.
1     WRITE(1,'(21H GO TO NEXT PROBLEM ?,/)')
      INPUT(1)ACHAR
      IF(ACHAR(1:1).EQ.'N') GO TO 10
      ID=0
      RETURN
10    WRITE(1,'(24H ENTER IN CONVERSATION ?,/)')
      INPUT(1)ACHAR
      IF(ACHAR(1:1).EQ.'Y') GO TO 11
12    IF(.NOT.LINEPRINT) GO TO 13
      CLOSE (1)
      OPEN(1,FILE='L-P',ACCESS='W')
13    IF(.NOT.RET)CALL MODEL(RET,INITCA,IGAR,IDAT,IDEL2)
      IF(INITCA) WRITE(1,104) X
104   FORMAT(' P. INI.',F7.3,2F8.1,F7.1,F7.3,F8.0,F9.2,F5.2,F8.0,F5.2)
      RETURN
11    WRITE(1,'(15H EXPLANATIONS ?,/)')
      INPUT(1)ACHAR
      IF(ACHAR(1:1).EQ.'N') GO TO 20
      WRITE (1,100)
100   FORMAT(' TYPE -FFT- TO CHANGE THE NB OF PTS OF FFT (DEFAULT 64)',
    * /,6X,'-PARAM- TO CHANGE THE PARAM. ANALYSED',/,6X,
    * '-MASS- TO CHANGE THE ION MASS',/,6X,
    * '-I,- (I=1,2,...,OR 10) TO CHANGE THE I TH PARAM.',/,6X,
    * '-OSPEC-,-OACFTHE-,-OACFEXP- TO GET CORRESPONDING OUTPUTS',/,6X,
    * '-OTIT- TO GET THE TITLE;-OPRINT- TO PRINT ACF-S + RES.',/,6X,
    * '-L-P- TO PUT THE NEXT OUTPUTS ON LINE-PRINTER',/,6X,
    * '-MOD- TO LEAVE THE CONVERSATION PROGRAM AND USE THE MODEL',/,6X,
    * 'VALUES FOR THE NEXT ANALYSIS',/,6X,
    * '-END- TO LEAVE THE CONVERSATION PROGRAM AND GO IN ANALYSIS')
20    WRITE(1,'(12H WAITING ...,/)')
      ACHAR=''
      READ(1,200)ACHAR
200   FORMAT(A)
      IF(ACHAR(1:3).EQ.'MOD') GO TO 12
      IF(ACHAR(1:1).EQ.'E') GO TO 90
      IF(ACHAR(1:1).EQ.'N') GO TO 25
      IF(ACHAR(1:2).EQ.'PA') GO TO 30
      IF(ACHAR(1:2).EQ.'IE') GO TO 50
      IF(ACHAR(1:1).EQ.'F') GO TO 40
      IF(ACHAR(1:1).EQ.'M') GO TO 60
      IF(ACHAR(1:1).NE.'L') GO TO 61
      LINEPRINT=.TRUE.
      GO TO 20
```

```
61      IF(ACHAR(1:2).EQ.'OT') GO TO 70
        IF(ACHAR(1:2).EQ.'OP') GO TO 74
        IF(ACHAR(1:5).EQ.'OSPEC') GO TO 71
        IF(ACHAR(1:5).EQ.'OACFT') GO TO 72
        IF(ACHAR(1:5).EQ.'OACFE') GO TO 73
        READ(ACHAR,*)I
        IF(I.GT.10.OR.I.LT.1) GO TO 20
        WRITE(1,'(3H X(,A1,3H) =,E12.4,/,3H X(,A1,5H) = ?,/)') ACHAR(1:1)
     *,X(I),ACHAR(1:1)
        INPUT(1) X(I)
        GO TO 20
25      WRITE(1,'(23H NUMBER OF LAGS USED  =,I3,5X,15H;NEW NUMBER = ?,/)'
     * )LFI
        INPUT(1) LFI
        LFIN=2*LFI
        GO TO 20
50      WRITE(1,'(20H INDEX FOR OUTPUTS =,I3,5X,15H;NEW NUMBER = ?,/)'
     * )IESSAI
        INPUT(1) IESSAI
        GO TO 20
30      WRITE(1,'(26H NUMBER OF PARAM. FITTED =,I2)') ID
        WRITE(1,107) (IORD(I,1),I=1,10)
  107   FORMAT(' ORDER OF PARAMETERS =',10I3)
        WRITE(1,'(47H NEW NUMBER AND DESIGNATION OF PARAM. FITTED= ?,/)')
        INPUT(1) ID,(IORD(I,1),I=1,ID)
        GO TO 20
40      WRITE(1,'(23H NUMBER OF FFT POINTS =,I3,5X,15H;NEW NUMBER = ?,/)'
     * )LM
        LMAV=LM
        INPUT(1) LM
        IF(LM.EQ.LMAV) GO TO 20
        RAP=FLOAT(LM)/FLOAT(LMAV)
        UNSRA2=1./(RAP*RAP)
        FD=FD/RAP
        DO 41 I=1,LN
        F(I)=(I-1.)*FD
        Y(2*I)=Y(2*I)*RAP
        Y(2*I-1)=Y(2*I-1)*RAP
41      POID(I)=POID(I)*UNSRA2
        GO TO 20
60      WRITE(1,'(11H ION MASS =,F5.2,5X,18H; NEW ION MASS = ?,/)')AMMIN
        INPUT(1) AMMIN
        RI=SQRT(AMMIN/16)
        GO TO 20
70      WRITE(1,106)(IDAT(K),K=19,23),IDAT(8),IDAT(15),IDAT(24)
106     FORMAT(' ****',' H=',I2,' DE ',I2,' MIN ',I2,' SEC A ',I2,
     * ' MIN ',I2,' SEC.  SNR=',I3,'/100, ALT=',I4,' KM, NUM=',I4)
        WRITE(1,108) LFI,IBEG,X,VAR,EX
  108 FORMAT(2I4,F7.3,2F8.1,F7.1,F7.3,F8.0,F9.2,F5.2,F8.0,F5.2,/
     * ,F8.4,F7.3,2F8.1,F7.1,F7.3,F8.0,F9.2,F5.2,F8.0,F5.2)
        GO TO 20
71      WRITE(1,171)IXM,FD,(GARSPE(L)/GARSPE(1),L=1,IXM)
171     FORMAT(' SPECTRUM ...     NB OF POINTS =',I3,10X,'FREQUENCY SPACIN
     *G =',F10.3,/,(10F8.4))
        GO TO 20
72      WRITE(1,172)LFI,(GFIN(L)/GFIN(1),ACFIM(L)/GFIN(1),L=1,LFI)
172     FORMAT(' ACF THEORETICAL ...   NB OF LAGS =',I3,/,(10F8.4))
        GO TO 20
73      WRITE(1,173) (Y(L)/Y(1),L=1,LFIN)
173     FORMAT(' ACF EXPERIMENTAL ...',/,(10F8.4))
```

```
      GO TO 20
74    OPEN (7,FILE='L-P',ACCESS='W')
      WRITE(7,106)(IDAT(K),K=19,23),IDAT(8),IDAT(15),IDAT(24)
      WRITE(7,108) LFI,IBEG,X,VAR,EX
      WRITE(7,171)IXM,FD,(GARSPE(L)/GARSPE(1),L=1,IXM)
      WRITE(7,172)LFI,(GFIN(L)/GFIN(1),ACFIM(L)/GFIN(1),L=1,LFI)
      WRITE(7,173)  (Y(L)/Y(1),L=1,LFIN)
      CLOSE(7)
      GO TO 20
90    L06=.FALSE.
      L09=.FALSE.
      IF(X(6).NE.0.)L06=.TRUE.
      IF(X(9).NE.0.)L09=.TRUE.
      N04=.TRUE.
      N06=.TRUE.
      N09=.TRUE.
      DO 31 K=1,ID
      IF(IORD(K,1).EQ.4)N04=.FALSE.
      IF(IORD(K,1).EQ.6)N06=.FALSE.
      IF(IORD(K,1).EQ.9)N09=.FALSE.
31    CONTINUE
      IDV=ID
      IF(.NOT.N04) GO TO 32
      IDV=IDV+1
      IORD(IDV,1)=4
32    IF(.NOT.(L06.AND.N06)) GO TO 33
      IDV=IDV+1
      IORD(IDV,1)=6
33    IF(.NOT.(L09.AND.N09)) GO TO 34
      IDV=IDV+1
      IORD(IDV,1)=9
34    L=IDV
      DO 35 I=1,10
      DO 36 K=1,IDV
      IF(IORD(K,1).EQ.I) GO TO 35
36    CONTINUE
      L=L+1
      IORD(L,1)=I
35    CONTINUE
      DO 9 I=1,10
9     IORD(IORD(I,1),2)=I
95    IF(.NOT.LINEPRINT) RETURN
      CLOSE (1)
      OPEN(1,FILE='L-P',ACCESS='W')
      RETURN
      END
```

MODEL

```
C
C      CONDITIONAL COMPILATION           P          USE THE P-INIT OF MOD:D
C
       SUBROUTINE MODEL(RET,INITCA,IGAR,IDAT,IDEL2)
      COMMON/SPE/F(32),FD,DG(32,7),POID(32),X(10),EX(10),GFIN(32),
     *ITALK,LFIN,LM,IBEG,ID,IDV,LNOISE,IX(32),IXM,ANE,TE,TI,V,P,IA,Y(64)
     *,VAR,VARO,KTER,S(7,7),ATTEN(32),GARSPE(32),IPAR(10),DER(10)
      COMMON/CONST/AMMIN,RI,BFE,BFO,DEBE,DEBO,BFEO,BE,BO,IESSAI,DEBOP,
     *RPI,ITERM,DEPI,LN,LP,LQ,LR,LFI,NSG,QN,CONV,QDIV,QINC,PK1
       COMMON/MOD/NBALT,KAL(4),AMMINO(4),NDEP(4),XPAR(10,4),KORD(10,5,4)
     * ,PIN(5,4),KD(5,4),KDV(5,4),KGAR(5,4),INITP(4,4),NINIT(4)
       COMMON/ORDER/IORD(10,2)
       DIMENSION XGAR(10)
       DOUBLE INTEGER IDAT(24)
       LOGICAL RET,INITCA,IGAR,INI1,INI2,LNOISE
       DATA IALT/0/
       IF(.NOT.INITCA) GO TO 10
C ****        IT IS NOT THE FIRST FIT, ONLY IORD AND ID,IDV ARE LOADED
       IF(IDAT(15).EQ.IALT) GO TO 20
C ****        SAME ALTITUDE AS THE PREVIOUS ANALYSIS ) SAME INIT. VALUES
       IALT=IDAT(15)
       DO 11 INDAL=1,NBALT
       IF(IALT.LT.KAL(INDAL)) GO TO 12
11     CONTINUE
12     AMMIN=AMMINO(INDAL)
       RI=SQRT(AMMIN/16.)
       ND=NDEP(INDAL)
       DO 13 I=5,10
13     X(I)=XPAR(I,INDAL)
       ANE=XPAR(1,INDAL)
       TE=XPAR(2,INDAL)
       TI=XPAR(3,INDAL)
       V=XPAR(4,INDAL)
       NIN=NINIT(INDAL)
       INI2=.FALSE.
       INI1=.FALSE.
       DO 6 N=1,NIN
       GO TO (1,2,3,4,5,6,7),INITP(N,INDAL)
1      ANE=2.*Y(1)*FD
       INI1=.TRUE.
       GO TO 6
2      IF(IDEL2.GT.1000) GO TO 6
       CALL TINIT
       INI2=.TRUE.
       GO TO 6
3      TI=195.+.2*(FLOAT(IALT)-90.)**2
       TE=TI
       GO TO 6
4      IF(IDEL2.LT.1000) CALL VINIT
       GO TO 6
5      X(5)=1./(1.+((FLOAT(IALT)-120.)/52.5)**4)
       GO TO 6
7      X(7)=EXP((96.-FLOAT(IALT))/7.)
6      CONTINUE
       P=X(5)
       IF(INI2.AND.P.NE.0.)CALL MODCOMP
       IF(.NOT.INI1) GO TO 30
       ANE=ANE*.5*(1.+TE/TI)
C ******* AL IS THE ALPHA PARAMETER IN THE SPECTRUM.
      AL=DEBO*ANE*BFO*BFO/TI
```

```
      ANE=ANE*1.E-11*(1.+1./AL)**2
30     X(1)=ANE
       X(2)=TE
       X(3)=TI
       X(4)=V
       DO 31 I=1,10
31     XGAR(I)=X(I)
       GO TO 32
20     DO 21 I=1,10
21     X(I)=XGAR(I)
32     JD=0
       RET=.FALSE.
10     JD=JD+1
       IF(JD.LE.ND) GO TO 40
       RET=.TRUE.
       RETURN
40     DO 41 I=1,10
41     IORD(I,1)=KORD(I,JD,INDAL)
       DO 42 I=1,10
42     IORD(IORD(I,1),2)=I
CP     X(5)=PIN(JD,INDAL)
       ID=KD(JD,INDAL)
       IDV=KDV(JD,INDAL)
       IF(KGAR(JD,INDAL).NE.0) IGAR=.TRUE.
       RETURN
       END
```

```
       SUBROUTINE MODCOMP
      COMMON/SPE/F(32),FD,DG(32,7),POID(32),X(10),EX(10),GFIN(32),
     *ITALK,LFIN,LM,IBEG,ID,IDV,LNOISE,IX(32),IXM,ANE,TE,TI,V,P,IA,Y(64)
     *,VAR,VARO,KTER,S(7,7),ATTEN(32),GARSPE(32),IPAR(10),DER(10)
       LOGICAL LNOISE
C ******* FORMULAS FOR PASSING FROM INITIAL VALUES FOR P=0 (O+ ONLY)
C ******* TO INIT. VAL FOR OTHER P (SIMILAR TO WICKWAR'S FIGURE 11).
      TE=TE*(.94+P-.36*(P-.55)**3)
      TI=TI*(-4.05-1.546*P+15.65/(3.1-P))
      GAR=0.
      IF(P.GT..5) GAR=40.8*(P-.625)**2-.64
      ANE=ANE*(1.+.01*(45.+62.2*P-76.5/(1.696-P)+GAR))
      RETURN
      END
C ******* INIT  DETERMINES  THE INITIAL VALUES OF PARAMETERS.
C
C
      SUBROUTINE TINIT
      COMMON/SPE/F(32),FD,DG(32,7),POID(32),X(10),EX(10),GFIN(32),
     *ITALK,LFIN,LM,IBEG,ID,IDV,LNOISE,IX(32),IXM,ANE,TE,TI,V,P,IA,Y(64)
     *,VAR,VARO,KTER,S(7,7),ATTEN(32),GARSPE(32),IPAR(10),DER(10)
       LOGICAL LNOISE
      COMMON/CONST/AMMIN,RI,BFE,BFO,DEBE,DEBO,BFEO,BE,BO,IESSAI,DEBOF,
     *RPI,ITERM,DEPI,LN,LP,LQ,LR,LFI,NSG,QN,CONV,QDIV,QINC,PK1
C ******* DET. OF THE FIRST MINIMUM OF ACF ( PARAB. INTERP.)
       DO 3 J=1,LFIN,2
       K=J+2
       IF(Y(J).GT.0.) GO TO 3
       IF(Y(K).GE.Y(J))GO TO 4
3      CONTINUE
4      I=J-2
       C=2.*Y(J)
       A=Y(I)+Y(K)-C
       B=Y(K)-Y(I)
       YMIN=(B*B/(4.*A)-C)/(2.*Y(1))
C *******  TZER, TIME OF 1 ST ZERO CROSSING, IS DETERMINED BY PARAB.
C ******* INTERPOLATION
       DO 1 JJ=1,LFIN,2
       IF(Y(JJ))2,2,1
1      CONTINUE
2      IF(JJ.EQ.J)GO TO 10
       C=2.*Y(JJ)
       A=Y(JJ+2)+Y(JJ-2)-C
       B=Y(JJ+2)-Y(JJ-2)
   10 CONTINUE
C?    IF(B*B-4.*A*C.LT.0.)CALL LOGERR('$*INIT* SQRT ERROR TZER$')
       TZER=(JJ-1.)/2.-2.*C/(B-SQRT(B*B-4.*A*C))
       TZER=TZER/(LM*FD)
C ******* THE NEXT TWO FORMULAS INTEND TO FIT FIGURES 6 AND 7 OF
C ******* WICKWAR'S REPORT.
    7 CONTINUE
       A=1.-2.17*YMIN
       IF(A.LT.0.)A=0.
       TR=(1.-SQRT(A))/.23
       TZI=(.4043+2.1825/(TR+.9632))*39./PK1
       TI=(TZI/TZER)**2*1.E-6
       TE=TI*TR
      RETURN
      END
      SUBROUTINE VINIT
```

```
      COMMON/SPE/F(32),FD,DG(32,7),POID(32),X(10),EX(10),GFIN(32),
     *ITALK,LFIN,LM,IBEG,ID,IDV,LNOISE,IX(32),IXM,ANE,TE,TI,V,P,IA,Y(64)
     *,VAR,VARO,KTER,S(7,7),ATTEN(32),GARSPE(32),IPAR(10),DER(10)
       LOGICAL LNOISE
      COMMON/CONST/AMMIN,RI,BFE,BFO,DEBE,DEBO,BFEO,BE,BO,IESSAI,DEBOF,
     *RPI,ITERM,DEPI,LN,LP,LQ,LR,LFI,NSG,QN,CONV,QDIV,QINC,PIN
C ******* DET. OF THE INITIAL VALUE OF THE WIND,V, IN HERTZ.
       A=0.
       SOM=0.
       T=0.
       DO 5 I=3,LFIN,2
       T=T+1.
       B=Y(I)*T
       C=B*POID((I+1)/2)
       A=A+B*C
5      SOM=SOM+C*Y(I+1)
       V=SOM/(A*DEPI)
      RETURN
      END
C
C
C      CONDITIONAL COMPILATIONS                -           BACKGROUND
C                                              R           REDUCED PARAM
C                                              H           HISTOGRAM
C                                              +           RT
C                                              #           HISTOGRAM ON RESAL:DATA
C                                              ? (RT)      MORE ON LOGMES
C
      SUBROUTINE SORTIE(ITER)
CH     DIMENSION IHIST(41,5)
      COMMON/COU/COUF13,COUF23,COUF73
      COMMON/SPE/F(32),FD,DG(32,7),POID(32),X(10),EX(10),GFIN(32),
     *ITALK,LFIN,LM,IBEG,ID,IDV,LNOISE,IX(32),IXM,ANE,TE,TI,V,P,IA,Y(64)
     *,VAR,VARO,KTER,S(7,7),ATTEN(32),GARSPE(32),IPAR(10),DER(10)
       LOGICAL LNOISE
      COMMON/CONST/AMMIN,RI,BFE,BFO,DEBE,DEBO,BFEO,BE,BO,IESSAI,DEBOF,
     *RPI,ITERM,DEPI,LN,LP,LQ,LR,LFI,NSG,QN,CONV,QDIV,QINC,PK1
CH     DATA NUMB/0/,IHIST/205*0/
C+     CHARACTER ACHAR*60
C?     CALL LOGMES('$*SORTIE* ENTERED$')
C+    ACHAR=''
       TI=X(3)
       TE=X(2)
       ANE=X(1)
CR    TI=(BFO/X(3))**2
CR    ANE=X(1)/(DEBO*X(3)*X(3)*1.E11)
      P=X(5)
      V=X(4)
      COUF33=EX(3)*EX(3)
CR    COUF11=EX(1)*EX(1)
CR    COUF22=EX(2)*EX(2)
CR    RF13=2.*X(1)/X(3)
C ****      THE ERRORS ON PHYSICAL PARAMETERS ARE DETERMINED FROM THE ERRORS
C ****   ON REDUCED PARAMETERS. SINCE ANE DEPENDS ON X(1) AND X(3), THE
C ****   ERROR ON ANE DEPENDS ON EX(1) AND EX(3) AND THE COUPLING COEF 1-3
C ****   IS DETERMINED IN "REGR" AND CALLED COUF13.
CR    EX(3)=2.*EX(3)/X(3)*TI
CR    EX(1)=(RF13*RF13*COUF33-2.*RF13*COUF13+COUF11)
CR     IF(EX(1).GT.0.)EX(1)=ANE/X(1)*SQRT(EX(1))
CR    TE=TI*X(2)
```

```
CR     RF23=2.*X(2)/X(3)
C **** COUPLING COEF. COUF23 DETERMINED IN 'REGR'
CR     EX(2)=(RF23*RF23*COUF33-2.*RF23*COUF23+COUF22)
CR      IF(EX(2).GT.0.)EX(2)=TI*SQRT(EX(2))
C **** IN BACKGROUND PROGRAM, HISTOGRAM OF THE VALUES OF THE DEDUCED
C **** PARAMETERS NORMALIZED TO THE ERRORS.
CH      NUMB=NUMB+1
CH      IF(NUMB.NE.200)GO TO 20
CH      NUMB=0
CH      WRITE(1,100)IHIST
C#      WRITE(4,100)IHIST
CH100   FORMAT(20I4,/,21I4)
CH20    I=(V)/EX(4)*10.+21.
CH      IF(I.LE.0)I=1
CH      IF(I.GT.41)I=41
CH      IHIST(I,4)=IHIST(I,4)+1
CH      I=(ANE-2.64)/EX(1)*10.+21.
CH      IF(I.LT.1)I=1
CH      IF(I.GT.41)I=41
CH      IHIST(I,1)=IHIST(I,1)+1
CH      I=(TE-135.)/EX(2)*10.+21.
CH      IF(I.LT.1)I=1
CH      IF(I.GT.41)I=41
CH      IHIST(I,2)=IHIST(I,2)+1
CH      I=(TI-113.5)/EX(3)*10.+21.
CH      IF(I.LT.1)I=1
CH      IF(I.GT.41)I=41
CH      IHIST(I,3)=IHIST(I,3)+1
C **** FOR LOW ALT. COLL. ANAL.
C **** X(7) IS THE REDUCED COLLISION FREQUENCY. ACCORDING TO WALDTEUFEL,
C **** IF MULTIPLIED BY C4,IT GIVES THE N2 DENSITY.
C **** THE ERROR ON X(7)*C4 DEPENDS ON EX(7) AND EX(3) AND THE COUPLING
C **** COEF COUF73 IS DETERMINED IN 'REGR'
        C4=1.0816*SQRT(TI)/(BFO*RI)
C C4=.083*2*PI*(1.+30.5/28.4)/X(3)
       COL=X(7)*C4
       RF73=-X(7)/(2.*X(3))
CR      RF73=-RF73*2.
        X(1)=ANE
        X(2)=TE
        X(3)=TI
        COUF77=EX(7)*EX(7)
       CCTMP=RF73*RF73*COUF33-2.*RF73*COUF73+COUF77
C?     IF(CCTMP.LT.0.)CALL LOGERR('$*SORTIE* CCTMP .LT.0$')
        DCOL=C4*SQRT(CCTMP)
        IF(COL.NE.0.) X(6)=COL
        IF(COL.NE.0.)EX(6)=DCOL
C-      WRITE(1,104) ITER,IBEG,X,VAR,EX
C+      WRITE(ACHAR,133)ITER,IBEG,(X(I),I=1,7)
C+      CALL LOGMES(ACHAR)
C+      WRITE(ACHAR,134) VAR,(EX(I),I=1,7)
C+      CALL LOGMES(ACHAR)
104     FORMAT(2I4,F7.3,2F8.1,F7.1,F7.3,F8.0,F9.2,F5.2,F8.0,F5.2,/
     * ,F8.4,F7.3,2F8.1,F7.1,F7.3,F8.0,F9.2,F5.2,F8.0,F5.2)
133     FORMAT(1H$,I3,I4,F7.3,2F8.1,F7.1,F7.3,F8.0,F9.2,1HC)
134     FORMAT(1HC,F7.4,F7.3,2F8.1,F7.1,F7.3,F8.0,F9.2,1H$)
       RETURN
       END
```

REGR

```
 C
C       COND. COMPIL.              -          BACKGROUND
C                                  *(BG)      CPU TIMES WRITTEN
C                                  +(RT)      MESSAGES ON LOGMES
C                                  ?          MORE ON LOGMES
      SUBROUTINE REGR
       COMMON/ORDER/IORD(10,2)
      COMMON/SPE/F(32),FD,DG(32,7),POID(32),X(10),EX(10),GFIN(32),
     *ITALK,LFIN,LM,IBEG,ID,IDV,LNOISE,IX(32),IXM,ANE,TE,TI,V,P,IA,Y(64)
     *,VAR,VARO,KTER,S(7,7),ATTEN(32),GARSPE(32),IPAR(10),DER(10)
      COMMON/CONST/AMMIN,RI,BFE,BFO,DEBE,DEBO,BFEO,BE,BO,IESSAI,DEBOP,
     *RPI,ITERM,DEPI,LN,LP,LQ,LR,LFI,NSG,QN,CONV,QDIV,QINC,PIN
      COMMON/COU/COUF13,COUF23,COUF73
      DIMENSION SGAR(8),SGA(7)
      DIMENSION GG(10),JE(8),JF(8),JL(8),JG(8),JK(8)
      LOGICAL JM,LNOISE
C*     DOUBLE INTEGER ITIME,JTIME,TUSED
C+     CHARACTER ACHAR*60
      IDJ=ID
C?     CALL LOGMES('$*REGR* ENTERED$')
      IIDJ=ID+1
      VARO=1.E20
      ITER=0
       KTER=0
      Q=QN
C *******  OFTEN THE CONVERGENCE IS MORE RAPID ,FOR 5 PARAMETERS FIT,
C ******* IF THE INCREMENTATION VECTOR IS SUBSTANTIALLY REDUCED
C ******* AT THE BEGINNING. QINC AND QDIV ARE FIXED TO 2 AND .5 IN CRECONST
       IF(IORD(5,2).LE.IDJ)Q=QN*QDIV*QDIV
      DO 312 I=1,10
  312 EX(I)=0.
      GO TO 400
309   IF(VAR.GT.VARO) GO TO 72
C *******  THE ITERATION HAS BEEN SUCCESSFUL ( VAR DECREASES)
       VARO=VAR
       IF(Q.LT.QN) Q=Q*QINC
      KTER=0
      DO 13 I=1,IDJ
      J=IORD(I,1)
13    GG(J)=X(J)
      ITER=ITER+1
C*     ITIME=TUSED(I)
C **** THE NUMBER OF ITERATIONS IS LIMITED TO 6.
      IF(ITER.LE.6) GO TO 88
      IBEG=3
      GO TO 313
88     CONTINUE
C ******* INVERSION OF MATRIX S.
      DO 20 I=1,IDJ
      DO 20 J=I,IDJ
20    S(J,I)=S(I,J)
      DO 10 K=1,IDJ
      JK(K)=0
      JE(K)=0
      JF(K)=0
      JG(K)=K
   10 JL(K)=K
      JQ=0
      K=1
  200 JQ=JQ+1
```

```
      U=0.
      DO 30 I=1,IDJ
      IF(JE(I).NE.0) GO TO 30
      IF(ABS(S(I,JQ)).LE.ABS(U)) GO TO 30
      U=S(I,JQ)
      JHI=I
   30 CONTINUE
      JF(K)=JHI
      JE(JHI)=1
      JK(K)=JQ
      S(JHI,JQ)=1./U
      DO 55 I=1,IDJ
   55 IF(I.NE.JHI)S(I,JQ)=-S(I,JQ)/U
      IF(JQ-1) 66,70,66
   66 J1=1
      JMAX=JQ-1
      JM=.TRUE.
      GO TO 90
   70 J1=JQ+1
      JMAX=IIDJ
      JM=.FALSE.
   90 DO 80 J=J1,JMAX
      ALPHA=S(JHI,J)
      S(JHI,J)=ALPHA/U
      DO 80 I=1,IDJ
   80 IF(I.NE.JHI) S(I,J)=S(I,J)+ALPHA*S(I,JQ)
      IF(JM.AND.JQ+1.LE.IIDJ) GO TO 70
      IF(K.GE.IDJ) GO TO 86
      K=K+1
      GO TO 200
   86 DO 130 K=1,IDJ
      JGK=JG(K)
      JFK=JF(K)
      IF(JGK.EQ.IFK) GO TO 130
      LAMBDA=JL(JFK)
      DO 140 LC=1,IIDJ
      VV=S(K,LC)
      S(K,LC)=S(LAMBDA,LC)
      S(LAMBDA,LC)=VV
  140 CONTINUE
      DO 160 LI=1,IDJ
      VV=S(LI,JFK)
      S(LI,JFK)=S(LI,JGK)
      S(LI,JGK)=VV
  160 CONTINUE
      JL(JFK)=K
      JL(JGK)=LAMBDA
      JG(LAMBDA)=JGK
      JG(K)=JFK
  130 CONTINUE
C*      JTIME=TUSED(I)-ITIME
C*      OUTPUT(1) ' MATRIX INVERSION TIME =',JTIME
C ******* CONVERGENCE TEST : CONV=.25 (IN CRECONST); CONVERGENCE IF,
C **** FOR ALL X DETERMINED, DELTA(X) IN THIS ITERATION IS SMALLER THAN
C **** .25*ERROR(X).
      KF=1
       CON=CONV*Q/QN
      DO 325 I=1,IDJ
       J=IORD(I,1)
       EX(J)=S(I,I)
```

```
      IF(EX(J).LE.0.) RETURN
       SGA(I)=S(I,IIDJ)
      EX(J)=SQRT(EX(J))
      X(J)=X(J)+Q*SGA(I)
  325 IF(ABS(X(J)-GG(J)).GT.CON*EX(J)) KF=2
      IF(KF.NE.1) GO TO 401
C **** CONVERGENCE TEST FULFILLED ****
C **** THE COUPLING COEFFICIENTS ARE SAVED.
      IBEG=2
313    DO 326 I=1,IDJ
       J=IORD(I,1)
326    X(J)=GG(J)
      COUF13=S(IORD(1,2),IORD(3,2))
      COUF23=S(IORD(2,2),IORD(3,2))
       COUF73=S(IORD(7,2),IORD(3,2))
       IF(IORD(7,2).GT.IDJ.OR.IORD(3,2).GT.IDJ)COUF73=0.
      IF(IORD(1,2).GT.IDJ.OR.IORD(3,2).GT.IDJ) COUF13=0.
      IF(IORD(2,2).GT.IDJ.OR.IORD(3,2).GT.IDJ) COUF23=0.
      GO TO 331
72    IF(KTER.LT.3) GO TO 19
C **** WE ARRIVE IN 72 WHEN THE PREVIOUS ITERATION IS NOT SUCCESSFUL.
C **** IF IT IS THE THIRD TIME, WE STOP TRYING TO SOLVE THIS PROBLEM.
C **** IF IT IS THE FIRST TIME, WE MAKE A NEW ITERATION AS IF THE PREVIOUS
C **** ONE HAD BEEN SUCCESSFUL. IF VAR IS STILL LARGER, WE CHANGE THE STRATEGY
C **** AND DECREASE THE INCREMENTATION OF THE PARAMETERS, STARTING FROM
C **** THE ITERATION WHICH HAD GIVEN THE SMALLEST VAR.
      IBEG=4
      GO TO 313
19    KTER=KTER+1
       IF(KTER.NE.1) GO TO 199
       DO 17 I=1,IDJ
17     SGAR(I)=SGA(I)
       GO TO 88
C ******* IN CASE OF A SECOND UNSUCCESSFUL ITERATION, ONE TRIES
C ******* AGAIN CLOSER TO THE VALUES GG OF PARAM. WHICH HAD GIVEN
C ******* THE SMALLEST VAR VALUE.
C
401    IF(X(1).GT.0..AND.X(2).GT.0..AND.X(3).GT.0..AND.
     * SIGN(1,X(5)).EQ.SIGN(1,GG(5)).AND.SIGN(1,1-X(5)).EQ.SIGN(1,1-
     * GG(5))) GO TO 400
199    Q=Q*QDIV
      DO 100 I=1,IDJ
      K=IORD(I,1)
 100  X(K)=GG(K)+Q*SGAR(I)
       GO TO 401
  400 CONTINUE
C?      CALL LOGMES('$*REGR* NEAR LABLE 400$')
C **** ONE CALLS NOW THE CALCULATION OF THE SPECTRUM AND DERIVATIVES.
      CALL SPECT
C **** AND OF THE FOURIER TRANSFORMS.
       CALL FOURIER
       IVAR=10.*VAR
       IF(IESSAI.LT.4) GO TO 309
C-     WRITE(1,701)ITER,X,VAR
C+    ACHAR=''
C+     WRITE(ACHAR,702)IVAR,(X(I),I=1,5)
C+702  FORMAT(1H$,I3,5E10.3,1H$)
C+     CALL LOGMES(ACHAR)
C-701 FORMAT(I4,8X,5E12.4,/,6E12.4)
2000  FORMAT(10E12.4)
```

```
       GO TO 309
331    IF(IESSAI.LT.3) GO TO 314
C-     WRITE(1,701)ITER,X,VAR
C-     WRITE(1,701)KTER,EX,VARO
C+    ACHAR=''
C+     WRITE(ACHAR,702)IVAR,(X(I),I=1,5)
C+     CALL LOGMES(ACHAR)
       IVAR=10.*VAR
C+     WRITE(ACHAR,702)IVAR,(EX(I),I=1,5)
C+     CALL LOGMES(ACHAR)
314   CALL SORTIE(ITER)
      RETURN
      END
```

```
 C
C       CONDIT. COMPIL.            N           USE OF NATURAL PARAM NE,TE,TI
C                                  O(BG)       OUTPUTS FOR SCHLEGEL
C                                  *(BG)       CPU TIMES WRITTEN
C                                  E           SAME AS J. MURDIN PROG.
C                                  R           USE REDUCED PARAMETERS
C                                  N           USE NATURAL PARAMETERS(NE,TE,TI)
C
C
C ******* SPECT CALCULATES THE EXACT SPECTRUM + ALL DERIVATIVES
C ******* FOR FREQ. F(IX(L),L=1,IXM)
C ******* THE NOTATIONS FOLLOW THE EISCAT REPORT ON DATA ANAL.
C ******* (FEW EXCEPTIONS : G41 AND G42 ARE CALLED HERE G3(1),G3(2) ;
C ******* AND THE QUANTITIES AR,AI,BI,DDI,DDR,DAITHE,DARTHE,DBITHE,
C ******* DAIPSI,DARPSI,DBIPSI FOR THE TWO IONS 1,2 MUST BE MULTIPLIED
C ******* BY THE QUANTITY AL TO MATCH THE DEFINITIONS OF THE REPORT ;
C ******* DI IS THEN IDENTICAL TO BI AND HAS THUS BEEN REPLACED BY BI)
C
C
      SUBROUTINE SPECT
C?     CHARACTER ACHAR*60
       COMMON/ORDER/IORD(10,2)
      COMMON/SPE/F(32),FD,DG(32,7),POID(32),X(10),EX(10),GFIN(32),
     *ITALK,LFIN,LM,IBEG,ID,IDV,LNOISE,IX(32),IXM,ANE,TE,TI,V,P,IA,Y(64)
     *,VAR,VARO,KTER,S(7,7),ATTEN(32),GARSPE(32),IPAR(10),DER(10)
      COMMON/CONST/AMMIN,RI,BFE,BFO,DEBE,DEBO,BFEO,BE,BO,IESSAI,DEBOP,
     *RPI,ITERM,DEPI,LN,LP,LQ,LR,LFI,NSG,QN,CONV,QDIV,QINC,PIN
      DIMENSION AL(3),XX(3),AR(3),AI(3),BI(3),DDDI(2)
       DIMENSION G5(2),G3(2),G7(2),COL(2),XXACV(2)
       LOGICAL ICOL(3),ITRA(3),LNOISE
      DIMENSION TEST(8),DF(3),DD(3),DDR(2),DDI(2),DR(2),DI(2),DDDR(2)
      COMPLEX FC(2),DERFC(2),CUGAR(2),CZ,CU,A1,A2,T
C*     DOUBLE INTEGER ITIME,JTIME,TUSED
C*     ITIME=TUSED(I)
CN     Y1=X(1)
CN     Y2=X(2)
CN     Y3=X(3)
CN     X(2)=X(2)/X(3)
CN     X(3)=BFO/SQRT(X(3))
CN     X(1)=X(1)*1.E11*DEBO*X(3)*X(3)
       DO 115 I=1,3
       ICOL(I)=.FALSE.
115    ITRA(I)=.TRUE.
       DO 113 I=1,2
       AI(I)=0.
       BI(I)=0.
       AR(I)=0.
       G3(I)=0.
113    G7(I)=0.
       IXM=32
       IF(X(5).EQ.0..AND.IORD(5,2).GT.IDV)ITRA(2)=.FALSE.
       IF(X(5).EQ.1..AND.IORD(5,2).GT.IDV)ITRA(1)=.FALSE.
       IF(X(7).NE.0..OR.IORD(7,2).LE.IDV)ICOL(2)=.TRUE.
       IF(ICOL(2))ICOL(1)=.TRUE.
       IF(ICOL(2))COL(2)=X(7)
       IF(ICOL(1))COL(1)=X(7)*X(10)
CO     IF(IDV.EQ.1.AND.IESSAI.GE.5)OUTPUT(1)' AE2=',(X(2)/X(1))
210   PE=BFEO/SQRT(X(2))
C             FOR COMPARISON WITH SIMPLE FORMULA USED BY JOHN MURDIN,
C             COMPILE WITH CO-CO E
```

```
CE      PE=0.
        P2=RI/SQRT(X(8))
       X3PE=X(3)*PE
       AL(1)=ABS(1.-X(5))*X(1)
       AL(2)=X(1)*ABS(X(5))
       AL(3)=X(1)/X(2)
        GAV=0.
       DO 100 L=1,IXM
       XX(1)=X(3)*F(L)
       XX(2)=XX(1)*P2
       XX(3)=PE*XX(1)
       DO 102 J=1,2
C **** IF ITRA=TRUE, THE CORRESPONDING ION IS CONSIDERED.
        IF(.NOT.ITRA(J)) GO TO 102
        IF(ICOL(J)) GO TO 114
C **** CALCULATION WITHOUT COLLISION ION NEUTRAL
        BI(J)=EXP(-XX(J)*XX(J))*RPI
       AI(J)=BI(J)*XX(J)
C ******* DAWSON'S INTEGRAL DF AND ITS DERIVATIVE DD
C              ALL APPROX. BUT THE SECOND ONE ARE TAKEN FROM
C       CODY,PACIOREK AND THACHER,AMS,1969,P. 171. PRECISION :4E-05 (RELATIVE)
        U=XX(J)
        X2=U*U
        IF(U.GT.2.5) GO TO 107
        DF(J)=U*(47.68008+X2*(-4.302978+X2*(1.365845-X2*.03566424)))
      */(47.67912+X2*(27.50551+X2*(6.900760+X2)))
        GO TO 111
107     IF(U.GE.3.5)GO TO 108
        X2=1./X2
C ****** APPROXIMATION OBTAINED BY JOE'S POLY
        DF(J)=(.5059167+X2*(.0831403+X2*(1.795869-X2*1.824025)))/U
        GO TO 111
108     IF(U.GE.5.) GO TO 109
        DF(J)=(.50022190+.24046413/(X2-1.9953175))/U
        GO TO 111
109     DF(J)=(.5+(.25001906+.372469387/(X2-2.748627766))/X2)/U
111    DD(J)=1.-2.*XX(J)*DF(J)
        AR(J)=DD(J)
        DDI(J)=AI(J)
        DDR(J)=AR(J)
        DR(J)=-2.*DF(J)
CO       FC(1)=CMPLX(-2.*DF(1),BI(1))
        GO TO 102
C
C **** CALCULATION WITH COLLISIONS
C
  114 IF(L.NE.1) GO TO 20
C ******* INITIALISATION OF THE FRIED-COMTE FUNCTION AT ZERO FREQU.
C              THE ABSOLUTE PRECISION IS BETTER THAN 2E-05
C
        G=COL(J)
        IF(G.GT.3.) GO TO 11
C ****** APPROXIMATION OBTAINED BY JOE'S POLY
        B2=RPI/(1.+G*(1.128235+G*(.274703+G*(-.07275088+G*(.0070044+
      * G*(.002294387+G*(-.0008457352+G*.00008479709)))))))
        GO TO 12
C ****** APPROXIMATION OF ABRAMOVITCH-SEGUN (END OF COMPL. ERROR F. TABLE)
  11    G2=G*G
        B2=RPI*G*(.4613135/(G2+.1901635)+.09999216/(G2+1.7844927)
      * +.002883894/(G2+5.5253437))
```

```
  12  FC(J)=CMPLX(0.,B2)
      DERFC(J)=CMPLX(1.-B2*G,0.)
      XXACV(J)=0.
      CUGAR(J)=3.+2.*G*G
      GO TO 10
C *******  THE FRIED-COMTE FUNCTION IS CALCULATED BY DIFFERENCE
C ******* THUS FREQUENCIES MUST NOT BE TOO FAR APART.
C             (THE RESULTS ARE EXCELLENT IF DX IS OF THE ORDER OF 0.1)
C
  20  DX=XX(J)-XXACV(J)
      XXACV(J)=XX(J)
      CU=CUGAR(J)
      DX2=DX*DX/6.
      CZ=CMPLX(XXACV(J),COL(J))
      CUGAR(J)=3.-2.*CZ*CZ
      FC(J)=(FC(J)-DX*(1.+DERFC(J)+DX2*(CUGAR(J)-2.+CU*DERFC(J))))
     * /(1.+DX*CZ*(1.+CUGAR(J)*DX2))
      DERFC(J)=1.+CZ*FC(J)
  10   T=DERFC(J)-XX(J)*FC(J)
       A1=FC(J)/T
       DR(J)=REAL(A1)
       BI(J)=AIMAG(A1)
       A2=(1.+XX(J)*A1)
       AR(J)=REAL(A2)
       AI(J)=AIMAG(A2)
       DDDR(J)=DR(J)*DR(J)-BI(J)*BI(J)
       DDDI(J)=2.*DR(J)*BI(J)
       A2=A2/T
       DDR(J)=REAL(A2)
       DDI(J)=AIMAG(A2)
102    CONTINUE
C ******* VERY SMALL XX FOR ELECTRONS, THUS SIMPLE APPROX. FOR INTEGRAL
      DEUX2=2.*XX(3)*XX(3)
      BI(3)=AL(3)*(1.-DEUX2*.5)*RPI
      AI(3)=BI(3)*XX(3)
      DF(3)=XX(3)*(1.-DEUX2/3.)
      DD(3)=1.-DEUX2
      AR(3)=AL(3)*DD(3)
  106 SRI=1.+AR(1)*AL(1)+AR(2)*AL(2)
CE     SRI=SRI-1.
      SII=AI(1)*AL(1)+AI(2)*AL(2)
      SR=AR(3)+SRI
      SI=AI(3)+SII
      CHI2=SRI*SRI+SII*SII
      CHE2=AR(3)*AR(3)+AI(3)*AI(3)
      VI=BI(1)*AL(1)+P2*BI(2)*AL(2)
      VE=BI(3)*X(2)*PE
      S1=VE*CHI2
      S2=CHE2*VI
       CH2=SR*SR+SI*SI
       U=(S1+S2)/CH2
       GS=U/(DEBOP*RPI*X(3))
       S4=2*GS/(S1+S2)
CO     ISOR=1
CO     IF(IA.EQ.0)ISOR=2
CO     IF(IDV.EQ.1.AND.IESSAI.GE.5)WRITE(1,567)F(L),XX(3),(2.*XX(3)),
CO    1 (RPI*(1.-DEUX2*.5))
CO    2,(AI(3)/AL(3)),DD(3),(S1+S2),CH2,F(L),XX(ISOR),FC(ISOR),
CO    3(AI(ISOR)/X(1)),(AR(ISOR)/X(1))
CO567 FORMAT(2X,F6.0,5E12.4,/,10X,2E12.4,/,2X,F6.0,5E12.4)
```

```
      VRI=VE*SRI-U*SR
      VII=VE*SII-U*SI
       VRE=VI*AR(3)-U*SR
      VIE=VI*AI(3)-U*SI
C **** THE SPECTRUM GS IS STORED IN PLACE OF DER(4), DERIVATIVE WITH
C **** RESPECT TO THE WIND (WHICH IS NOT NEEDED)
       DER(4)=GS
      Z=-S1*XX(3)-2.*(AL(3)*DF(3)+XX(3)*AR(3))*VRE
     *    +VIE*(BI(3)-2.*XX(3)*AI(3))
      DER(6)=S4*Z*X3PE
      DER(1)=(GS-S4*VRI)/X(1)
       CHES2=.5*CHE2
C ******* PREVIOUS FORMULA FOR DERIVATIVE W.R.TO ION COMPOSITION IN CASE
C ******* OF ZERO ION-NEUTRAL COLLISION FREQUENCY.
C     DER(5)=S4*X(1)*((P2*BI(2)-BI(1))*(CHES2+XX(1)*VII)
C    *+VRI*(DD(2)-DD(1)))
       DO 112 I=1,2
       IF(.NOT.ITRA(I)) GO TO 116
       DAITHE=BI(I)-2.*XX(I)*DDI(I)
       DARTHE=DR(I)-2.*XX(I)*DDR(I)
       DBITHE=-2.*DDI(I)
       IF(.NOT.ICOL(I)) GO TO 117
       DBIPSI=-2.*DDR(I)-DDDR(I)
       DAIPSI=XX(I)*DBIPSI
       DARPSI=(2.*DDI(I)+DDDI(I))*XX(I)
       G7(I)=(VRI*DARPSI+VII*DAIPSI+CHES2*DBIPSI)*AL(I)
117    G3(I)=(VRI*DARTHE+VII*DAITHE+CHES2*DBITHE)*AL(I)
       G5(I)=VRI*AR(I)+VII*AI(I)+CHES2*BI(I)
116    CHES2=CHES2*P2
112    CONTINUE
      DER(3)=(S4*(XX(1)*G3(1)+XX(2)*G3(2)+Z*XX(3))-GS)/X(3)
      DER(2)=-S4/X(2)*(.25*S1+Z*XX(3)*.5+AR(3)*VRE+AI(3)*VIE)
       DER(5)=S4*X(1)*(-G5(1)*SIGN(1.,1.-X(5))+G5(2)*SIGN(1.,X(5)))
       DER(7)=S4*(X(10)*G7(1)+G7(2))
       DER(9)=S4*X(3)*P2*G3(2)
       DER(8)=-1./(2.*X(8))*(F(L)*DER(9)+GS*CHE2/(S1+S2)*P2*BI(2)*AL(2))
       DER(10)=S4*X(7)*G7(1)
CN     DER(3)=-(X(1)*DER(1)+X(2)*DER(2)+.5*X(3)*DER(3))/Y3
CN     DER(1)=DER(1)*X(1)/Y1
CN     DER(2)=DER(2)/Y3
       DO 104 I=1,IDV
C
C **** THE DERIVATIVES ARE PUT IN THE RIGHT PLACE IN TABLE DG, ACCORDING
C **** TO THE TABLE GIVING THE ORDER OF PARAMETERS IORD, INITIALIZED IN
C **** CRECONST AND MODIFIED IN ANAL.
C
       J=IORD(I,1)
104    DG(L,I)=DER(J)
C **** THE SPECTRUM (EXCLUDING THE EFFECT OF AN ENSEMBLE SPEED X(4))
C **** IS STORED IN GARSPE.
       GARSPE(L)=GS+X(6)*DER(6)+X(9)*DER(9)
       IF(GS.GE.GAV)GO TO 103
       GO TO 110
103    IF(GAV.LE.0.) GAV=GS/500.
100   CONTINUE
       L=L-1
110    IXM=L
       IF(L.EQ.32) GO TO 200
       L=L+1
       DO 501 I=L,32
```

```
        DO 501 J=1,IDV
501     DG(I,J)=0.
C*      JTIME=TUSED(I)-ITIME
C*      OUTPUT(1) ' SPECTRUM CALC. TIME =',JTIME
200     CONTINUE
CN      X(1)=Y1
CN      X(2)=Y2
CN      X(3)=Y3
        RETURN
       END
        EOF
```

```
C ******* FOURIER CALCULATES THE REAL FOURIER TRANSFORMS OF THE
C ******* CENTERED SPECTRUM AND ITS DERIVATIVES WITH RESPECT TO
C ******* X(1),X(2),X(3),X(5),X(6) WHICH ARE ALL SYMMETRIC OR
C ******* ANTISYMMETRIC. IT ALSO CALCULATES THE SQUARE DEVIATION
C ******* FROM OBSERVED VALUES, AND THE MATRIX S WHICH INCLUDES
C ******* THE MATRIX T AND THE VECTOR V OF T. HAGFORS REPORT.
C
C       COND. COMPIL.           S           SINGLE PULSE
C                               M           MULTIPULSE
C                               -           BACKGROUND
C                               *(BG)       CPU TIMES WRITTEN
C                               +(RT)       MESSAGES ON LOGMES
C                               ?           MORE ON LOGMES
C                               V           DETERMIN. OF DIFFERENTIAL WINDS
C
        SUBROUTINE FOURIER
C?      CHARACTER*60 JOE
        COMMON/GEN/ACFIM(32)
       COMMON/SPE/F(32),FD,DG(32,7),POID(32),X(10),EX(10),GFIN(32),
      *ITALK,LFIN,LM,IBEG,ID,IDV,LNOISE,IX(32),IXM,ANE,TE,TI,V,P,IA,Y(64)
      *,VAR,VARO,KTER,S(7,7),ATTEN(32),GARSPE(32),IPAR(10),DER(10)
        LOGICAL LNOISE
        COMMON/FOUR/CO(16),SI(8,8)
        DIMENSION T1(16)
       COMMON/ORDER/IORD(10,2)
       COMMON/CONST/AMMIN,RI,BFE,BFO,DEBE,DEBO,BFEO,BE,BO,IESSAI,DEBOP,
      *RPI,ITERM,DEPI,LN,LP,LQ,LR,LFI,NSG,QN,CONV,QDIV,QINC,PIN
        DIMENSION T(32),GAR(32),DEVI(32,2)
C*      DOUBLE INTEGER JTIME,TUSED,ITIME
C?       CALL LOGMES('$*FOURIER* ENTERED$')
C*      ITIME=TUSED(I)
        DO 300 IDJ=1,IDV
C **** THE TABLE IPAR AFFECTS A PARITY INDEX TO EACH PARAMETER X(J) (SEE
C **** CRECONST).ALL IPAR = 1 EXCEPT FOR X(4),X(6),X(9) (WINDS) WHERE
C **** IPAR=2. BUT IN 'SPECT', THE SPECTRUM IS KEPT IN PLACE OF THE
C **** DERIVATIVE WITH RESPECT TO X(4); THUS IF IORD=4, THE FUNCTION TO
C **** FOURIER TRANSFORM IS SYMMETRIC.
CV      IF(IPAR(IORD(IDJ,1)).EQ.1) GO TO 1
CV      IF(IORD(IDJ,1).EQ.4) GO TO 1
C ******* DG(L,6) IS THE DERIVATIVE OF SPECTRUM WITH RESPECT TO X(6),
C ******* ANTISYMMETRIC . AFTER THAT, IT IS THE IMAGINARY PART OF FT.
C
        IF(LM.NE.64) GO TO 215
C****** IN THAT CASE, ONE DOES NOT USE THE FFT.
C
C                             FFT PROGRAM FOR 64 REAL POINTS
C                (IN FACT 32 BECAUSE OF THE ANTISYMMETRY)
C
CV      DO 201 I=1,LN
CV201   T(I)=DG(I,IDJ)
CV      TLP=-T(LQ)*2.
CV      GFIN(1)=0.
CV      DO 200 I=3,LP,2
CV      T1(I)=T(I)-T(LR-I)
CV200    T1(I-1)=T(I)+T(LR-I)
CV      A4=T1(9)+T1(9)
CV      A3=T1(3)+T1(15)
CV      A2=(T1(5)+T1(13))*CO(8)
CV      A1=T1(7)+T1(11)
CV      D=A1*CO(4)+A3*CO(12)
```

```
CV      E=A3*CO(4)-A1*CO(12)
CV      A5=A2+A4
CV      A6=A2-A4
CV      GFIN(3)=D+A5
CV      GFIN(7)=E+A6
CV      GFIN(11)=E-A6
CV      GFIN(15)=D-A5
CV      A1=T1(3)-T1(7)+T1(11)-T1(15)
CV      GFIN(9)=A1+A1
CV      A1=T1(5)-T1(13)
CV      A1=A1+A1
CV      GFIN(5)=(T1(3)+T1(7)-T1(11)-T1(15))*CO(8)+A1
CV      GFIN(13)=GFIN(5)-A1-A1
CV      E=T1(2)*CO(14)+T1(6)*CO(10)+T1(10)*CO( 6)+T1(14)*CO( 2)
CV      A1=T1(8)*CO(8)
CV      D=T1(4)*CO(12)+T1(12)*CO( 4)+A1
CV      GFIN(2)=E+D
CV      GFIN(16)=E-D
CV      E=T1(2)*CO( 2)-T1(6)*CO( 6)+T1(10)*CO(10)-T1(14)*CO(14)
CV      D=D-A1-A1
CV      GFIN(8)=E+D
CV      GFIN(10)=E-D
CV      E=T1(2)*CO(10)+T1(6)*CO( 2)+T1(10)*CO(14)-T1(14)*CO( 6)
CV      D=T1(4)*CO( 4)-T1(12)*CO(12)+A1
CV      GFIN(4)=E+D
CV      GFIN(14)=E-D
CV      E=T1(2)*CO( 6)+T1(6)*CO(14)-T1(10)*CO( 2)+T1(14)*CO(10)
CV      D=D+A1+A1
CV      GFIN(6)=E+D
CV      GFIN(12)=E-D
CV      DO 203 L=2,LP
CV203   GAR(L)=GFIN(L)
CV      DO 202 I=2,LP,2
CV      T1(I)=T(I)-T(LR-I)
CV202   T1(I-1)=T(I)+T(LR-I)
CV      A4=T1(2)+T1(16)
CV      A3=T1(4)+T1(14)
CV      A2=T1(6)+T1(12)
CV      A1=T1(8)+T1(10)
CV      GFIN(3)=A1*CO(2)+A2*CO(6)+A3*CO(10)+A4*CO(14)
CV      A6=T1(4)+T1(6)-T1(12)-T1(14)
CV      A5=T1(2)+T1(8)-T1(10)-T1(16)
CV      GFIN(5)=A6*CO(4)+A5*CO(12)
CV      GFIN(7)=-A1*CO(6)+A2*CO(14)+A3*CO(2)+A4*CO(10)
CV      GFIN(9)=(T1(2)+T1(4)-T1(6)-T1(8)+T1(10)+T1(12)-T1(14)-T1(16))*
CV    * CO(8)
CV      GFIN(11)=A1*CO(10)-A2*CO(2)+A3*CO(14)+A4*CO(6)
CV      GFIN(13)=-A6*CO(12)+A5*CO(4)
CV      GFIN(15)=-A1*CO(14)+A2*CO(10)-A3*CO(6)+A4*CO(2)
CV      ISI=-1
CV      DO 207 L=1,8
CV      ISI=-ISI
CV      D=0.
CV      DO 208 M=1,8
CV208   D=D+T1(17-2*M)*SI(L,M)
CV207   GFIN(2*L)=D*ISI
CV205   DO 206 L=2,LP
CV      GFIN(LR-L)=GFIN(L)-GAR(L)
CV206   GFIN(L)=GAR(L)+GFIN(L)
CV      B=0.
```

```
CV      DO 211 L=2,LN,4
CV      B=B+DG(L,IDJ)
CV211   GFIN(L)=GFIN(L)+TLP
CV      DO 212 L=4,LN,4
CV      B=B-DG(L,IDJ)
CV212   GFIN(L)=GFIN(L)-TLP
CV      GFIN(LQ)=-2.*B
CV      DO 213 L=1,LN
CV      DG(L,IDJ)=GFIN(L)
CV213   CONTINUE
CV      GO TO 300
C****** SIMPLE DFT ROUTINE IF MORE THAN 64 POINTS.
215     E=DEPI*FD
        DO 216 L=1,32
        B=0.
        D=E*(L-1.)
        DO 217 K=2,IXM
217     B=B+SIN(D*(K-1.))*DG(K,IDJ)
216     GFIN(L)=-2.*B
        DO 218 L=1,32
218     DG(L,IDJ)=GFIN(L)
C ******* THIS PART TRANSFORMS DG(L,IDJ), DERIV. OF SPECTRUM WITH
C ******* RESPECT TO IDJ, SYMMETRIC, INTO ITS FT, REAL.
C
C                             FFT PROGRAM FOR 64 REAL POINTS
C               (IN FACT 32 BECAUSE OF THE SYMMETRY)
C
1       B=0.
        IF(LM.NE.64) GO TO 115
C******IF MORE THAN 64 POINTS, DFT INSTEAD OF FFT
        DO 2 I=1,LN
        T(I)=DG(I,IDJ)
2       B=B+T(I)
        TEND=2*T(LN)-T(LN-1)
        TO=T(1)
        TLP=T(LQ)*2.
        GFIN(1)=2.*B-TO+TEND
        DO 100 I=3,LP,2
        T1(I)=T(I)+T(LR-I)
100     T1(I-1)=T(I)-T(LR-I)
        A1=T1(3)-T1(15)
        A2=(T1(5)-T1(13))*CO(8)
        A3=T1(7)-T1(11)
        D=A1*CO(4)+A3*CO(12)
        E=A3*CO(4)-A1*CO(12)
        GFIN(3)=D+A2
        GFIN(7)=-E-A2
        GFIN(11)=E-A2
        GFIN(15)=A2-D
        A1=T1(9)-T1(5)-T1(13)
        GFIN(9)=A1+A1
        A1=T1(9)+T1(9)
        GFIN(5)=(T1(3)-T1(7)-T1(11)+T1(15))*CO(8)-A1
        GFIN(13)=-GFIN(5)-A1-A1
        E=T1(2)*CO( 2)+T1(6)*CO( 6)+T1(10)*CO(10)+T1(14)*CO(14)
        A1=T1(8)*CO(8)
        D=T1(4)*CO( 4)+T1(12)*CO(12)+A1
        GFIN(2)=E+D
        GFIN(16)=D-E
        E=T1(2)*CO(14)-T1(6)*CO(10)+T1(10)*CO( 6)-T1(14)*CO( 2)
```

```
      D=-D+A1+A1
      GFIN(8)=E+D
      GFIN(10)=D-E
      E=T1(2)*CO( 6)-T1(6)*CO(14)-T1(10)*CO( 2)-T1(14)*CO(10)
      D=T1(4)*CO(12)-T1(12)*CO( 4)-A1
      GFIN(4)=E+D
      GFIN(14)=D-E
      E=T1(2)*CO(10)-T1(6)*CO(2)+T1(10)*CO(14)+T1(14)*CO( 6)
      D=-(D+A1+A1)
      GFIN(6)=E+D
      GFIN(12)=D-E
      E=1.
      DO 103 L=2,LP
      E=-E
103   GAR(L)=GFIN(L)+T0+E*TEND
      DO 101 I=2,LP,2
      T1(I)=T(I)+T(LR-I)
101   T1(I-1)=T(I)-T(LR-I)
      A1=T1(2)-T1(16)
      A2=T1(4)-T1(14)
      A3=T1(6)-T1(12)
      A4=T1(8)-T1(10)
      GFIN(3)=A1*CO(2)+A2*CO(6)+A3*CO(10)+A4*CO(14)
      A5=T1(4)-T1(6)-T1(12)+T1(14)
      A6=T1(2)-T1(8)-T1(10)+T1(16)
      GFIN(5)=A6*CO(4)+A5*CO(12)
      GFIN(7)=A1*CO(6)-A2*CO(14)-A3*CO(2)-A4*CO(10)
      GFIN(9)=(T1(2)-T1(4)-T1(6)+T1(8)+T1(10)-T1(12)-T1(14)+T1(16))*
     * CO(8)
      GFIN(11)=A1*CO(10)-A2*CO(2)+A3*CO(14)+A4*CO(6)
      GFIN(13)=A6*CO(12)-A5*CO(4)
      GFIN(15)=A1*CO(14)-A2*CO(10)+A3*CO(6)-A4*CO(2)
      DO 107 L=1,8
      D=0.
      DO 108 M=1,8
108   D=D+T1(2*M-1)*SI(L,M)
107   GFIN(2*L)=D
105   DO 106 L=2,LP
      GFIN(LR-L)=GAR(L)-GFIN(L)
106   GFIN(L)=GAR(L)+GFIN(L)
      B=0.
      DO 111 L=5,LN,4
      B=B+T(L)
111   GFIN(L)=GFIN(L)+TLP
      DO 112 L=3,LN,4
      B=B-T(L)
112   GFIN(L)=GFIN(L)-TLP
      GFIN(LQ)=2.*B+T0+TEND
      DO 113 L=1,LN
      DG(L,IDJ)=GFIN(L)
113   CONTINUE
      GO TO 300
C****** SIMPLE DFT ROUTINE FOR MORE THAN 64 POINTS
115   E=DEPI*FD
      DO 116 L=1,32
      B=0.
      D=E*(L-1.)
      DO 117 K=2,IXM
117   B=B+COS(D*(K-1.))*DG(K,IDJ)
116   GFIN(L)=DG(1,IDJ)+2.*B
```

```
      DO 118 L=1,32
118   DG(L,IDJ)=GFIN(L)
300   CONTINUE
C ******* VAR IS THE SUM OF THE SQUARES OF DEVIATIONS. DEVI IS AN
C ******* ARRAY USED FOR THE VECTOR V CALCULATION (S(I,ID+1))
      B=DEPI*X(4)
      VAR=0.
      A2=0.
      I=IORD(4,2)
CV    J=IORD(6,2)
CV    K=IORD(9,2)
CS    DO 114 L=1,LFI
CM    DO 114 L=2,LFI
      E=B*(L-1.)
      COV=COS(E)
      SIV=SIN(E)
      A1=DG(L,I)
CV    A2=X(6)*DG(L,J)+X(9)*DG(L,K)
C **** GFIN AND ACFIM ARE THE REAL AND IMAGINARY PARTS OF THE THEORETICAL
C **** ACF TO COMPARE TO THE DATA
      GFIN(L)=COV*A1-SIV*A2
      ACFIM(L)=SIV*A1+COV*A2
      E=Y(2*L-1)-GFIN(L)
      D=Y(2*L)-ACFIM(L)
      DEVI(L,1)=COV*E+SIV*D
      DEVI(L,2)=COV*D-SIV*E
114   VAR=VAR+POID(L)*(E*E+D*D)
      VAR=SQRT(VAR/(LFIN-ID))
C **** FOR A PERFECT FIT, VAR SHOULD BE CLOSE TO 1.
      IF(VAR.GT.VARO.AND.KTER.GE.1)RETURN
C ******* DER. W. R. TO X(4)=V
5     DO 10 L=2,LFI
10    DG(L,I)=(L-1)*DG(L,I)*DEPI
CS    DG(1,1)=0.
C ******* CALC. OF MATRIX S (MATRIX T(ID,ID) + VECT. V=S(.,ID+1) OF HAGFORS)
      IID=ID+1
      DO 16 I=1,IID
      DO 16 J=1,IID
16    S(I,J)=0
      DO 17 I=1,ID
      ICO=IPAR(IORD(I,1))
CS    DO 18 L=1,LFI
CM    DO 18 L=2,LFI
      VV=POID(L)*DG(L,I)
      S(I,IID)=S(I,IID)+VV*DEVI(L,ICO)
18    S(I,I)=S(I,I)+VV*DG(L,I)
      IF(I.EQ.ID) GO TO 17
      DO 15 J=I+1,ID
      JCO=IPAR(IORD(J,1))
      IF(JCO.NE.ICO) GO TO 15
CS    DO 19 L=1,LFI
CM    DO 19 L=2,LFI
19    S(I,J)=S(I,J)+POID(L)*DG(L,I)*DG(L,J)
15    CONTINUE
17    CONTINUE
C*    JTIME=TUSED(I)
C*    JTIME=JTIME-ITIME
C*    OUTPUT(1) ' FOURIER TIME =',JTIME
      RETURN
      END
```

```
 C               GENER
C
C            *****  MUST BE USED WITH **FOUREG** COMPILED WITH -G      *****
C
C                              H          SAME WEIGHTS IN ANALYSIS
C                              -          THE GENERATION PARAMETERS ARE ASKED
C                              (OTHERWISE, RANDOMLY GENERATED)
C                              $          SMEARING BY A TRIANGLE
C                              R          USE REDUCED PARAMETERS
C
C
C ******* GENER GENERATES A COMPLEX ACF WITH RANDOM INDEPENDANT NOISE
C
C
      SUBROUTINE GENER(NUMBER,XCOR,ISIG2,IDAT)
      DIMENSION ROPRI(128)
      COMMON/GEN/ACFIM(32)
     COMMON/SPE/F(32),FD,DG(32,7),POID(32),X(10),EX(10),GFIN(32),
    *LDEB,LFIN,LM,IBEG,ID,IDV,LNOISE,IX(32),IXM,ANE,TE,TI,V,P,IA,Y(64)
    *,VAR,VARO,KTER,S(7,7),ATTEN(32),GARSPE(32),IPAR(10),DER(10)
     COMMON/CONST/AMMIN,RI,BFE,BFO,DEBE,DEBO,BFEO,BE,BO,IESSA1,DEBOF,
    *RPI,ITERM,DEPI,LN,LP,LQ,LR,LFI,NSG,QN,CONV,QDIV,QINC,PK1
      COMMON/ORDER/IORD(10,2)
      DOUBLE INTEGER XCOR(64),NUMBER,ISIG2(32),IDAT(24)
      LOGICAL RET,INITCA,IGAR,LNOISE
      CALL GAUSS(RNDM)
      ANE=5.+2.*RNDM
      IF(ANE.LT.1.)ANE=1.
      P=0.
      PIN=0.
      X(8)=1.
      X(9)=0.
      X(10)=1.
      X(6)=0.
      IALT=200
      ANFREQ=4.
      PULENG=10000.
      PUREP=17.
      IXM=LN
      CALL GAUSS(RNDM)
      TI=1200.+300.*RNDM
      IF(TI.LT.700.)TI=700.
      CALL GAUSS(RNDM)
      TE=TI+1500.*ABS(RNDM)
      CALL GAUSS(RNDM)
      V=300.*RNDM
      X(1)=ANE
      X(2)=TE
      X(3)=TI
      X(4)=V
      X(5)=P
      CALL GAUSS(RNDM)
      TINT=90.+30.*RNDM
      IF(TINT.LT.30.)TINT=30.
      CALL GAUSS(RNDM)
      SNR=1.+.3*RNDM
      IF(SNR.LT..2)SNR=.2
C
C     THE SAMPLING FREQUENCY IS ASSUMED 50000 HZ, WHICH CORRESPONDS
C     TO THE MAXIMUM SPRECTRUM WIDTH. OVSAMP IS HERE THE OVERSAMPLING
```

```
C       VALUE (IF 1, NYQUIST SAMPLING, IF 2 ,OVERSAMPLING BY TWO,...)
C
        SAMFRE=50.
C              THE FILTER WIDTH IS ALSO ASSUMED 50000 HZ
        OVSAMP=1
        RI=SQRT(AMMIN/16.)
C-      WRITE(1,104)
  104   FORMAT(' ALTITUDE =',/)
C-      INPUT(1) IALT
        IDAT(15)=IALT
        BFO=BO/PK1
        IF(IALT.GE.120) GO TO 154
        COL=EXP((96.-IALT)/7.)
        TI=195.+.2*(IALT-90.)**2
        WRITE(1,105)COL,TI
  105   FORMAT(' COL=',E12.4,'    TI=',F8.3)
  154   CONTINUE
C-      WRITE(1,102)
102     FORMAT(' ANE,TE,TI,V,P',/)
C
C              TINT :INTEGR. TIME (SEC.) ; ANFREQ:NUMBER OF FREQU.
C              PULENG : LENGTH OF THE PULSE (MUSEC.);PUREP:TIME (MSEC.)
C              BETWEEN TWO PULSES; SNR: S/N RATIO IN THE BAND OF THE FILTER
C              SAMFRE,TOTWI:SAMPLING FREQUENCY AND FILTER WIDTH
C-      INPUT(1)X(1),X(2),X(3),X(4),X(5)
C
C       SPECTRUM WIDTH  CALCULATED THROUGH EMPIRICAL FORMULA
C
        SPEWI=931.*SQRT(X(3)+X(2)/3.)
C-      OUTPUT(1) ' SPECTRUM WIDTH =', SPEWI
C-      WRITE(1,103)
C-103   FORMAT(48H TINT(SEC),ANFREQ,PULENG(MUSEC),PUREP(MSEC),SAMF  ,
C-     * 23HRE(KHZ),SNR,OVSAMP,NLAG,/)
C-      INPUT(1)TINT,ANFREQ,PULENG,PUREP,SAMFRE,SNR,OVSAMP,NLAG
        SAMFRE=SAMFRE*1000.
        PUREP=PUREP*1.E-3
        TOTWI=SAMFRE/OVSAMP
        IDAT(13)=NLAG
        LFI=32
        LFIN=2*LFI
        IXM=32
        DEFI=6.283185308/SAMFRE
        IBEG=0
        FI=SAMFRE/LM
        IDAT(14)=SAMFRE*.1
        DO 1 I=1,LN
 1      F(I)=(I-1)*FI
        NSG=PULENG*SAMFRE*1.E-6
        IDAT(1)=100.*X(1)
        IDAT(2)=X(2)
        IDAT(3)=X(3)
        IDAT(4)=X(4)
        IDAT(5)=100.*X(5)
 C      CALL GAUSS(RNDM)
        A=P+.3*RNDM
        IF(A.LT.0.) A=0.
        IF(A.GT.1.) A=1.
 C-     A=PIN
        IDAT(11)=100.*A
        IDAT(6)=X(6)
```

```
      IDAT(7)=TINT
      IDAT(12)=ANFREQ/PUREP
      IDAT(8)=100.*SNR
      IDAT(9)=NSG
CH    IDAT(9)=0
      IF(NSG.GE.LFI) GO TO 4
      LFI=NSG
      LFIN=2*LFI
4     VARO=0
      ID=1
      IDV=1
      IORD(1,1)=4
      IORD(4,2)=1
      X(7)=0.
      X(6)=0.
7     KTER=1
      RET=.TRUE.
C-    CALL TALK(RET,INITCA,IGAR,IDAT,A)
      ID=1
      DO 40 L=1,32
C           TRIANGULAR SMEARING OF THE ACF IF COMPILED WITH $
      ATTEN(L)=1.
C$    ATTEN(L)=1.-(L-1.)/NSG
40    CONTINUE
CR    X(2)=X(2)/X(3)
CR    X(3)=BFO/SQRT(X(3))
CR    X(1)=X(1)*1.E11*DEBO*X(3)*X(3)
      CALL SPECT
CR    X(1)=X(1)/(DEBO*X(3)*X(3)*1.E11)
CR    X(3)=(BFO/X(3))**2
CR    X(2)=X(2)*X(3)
      INDS=IORD(4,2)
      IF(IESSAI.GE.3)WRITE(1,1000)((GARSPE(L)/GARSPE(1)),L=1,IXM)
1000  FORMAT(' SPECTRUM',/,(6E12.5))
      CALL FOURIER
      IF(IESSAI.GE.4) WRITE(1,1002)NSG,PULENG
1002  FORMAT(' NSG=',I5,'  PULENG=',E12.3)
      A=DEPI*X(4)
C           HERE,GFIN AND ACFIM WILL BE THE REAL AND IMAG. PARTS OF ACF
      DO 30 L=1,LFI
      ACFIM(L)=ACFIM(L)*ATTEN(L)
30    GFIN(L)=GFIN(L)*ATTEN(L)
C
C
C           *** CALCULATION OF THE VARIANCES OF ACF ***
C     THE FORMULAS ARE TAKEN FROM ALKER'S REPORT
C     THE FILTER IS ASSUMED TO BE FLAT
      BINT=1.E+06/GFIN(1)
      B=(1.+1./SNR)*GFIN(1)*SQRT(PUREP/(2.*ANFREQ*TINT))
      PNOISE=GFIN(1)/SNR
      DO 6 L=2,LFI
      A=TOTWI*DEPI/2.*(L-1.)
6     ROPRI(L)=GFIN(L)+PNOISE*SIN(A)/A
      ROPRI(1)=GFIN(1)+PNOISE
      CC=1./ROPRI(1)**2
      LFI=NLAG
      IF(LFI.GT.32) LFI=32
      IF(LFI.GT.NSG) LFI=NSG
      LFIN=2*LFI
      DO 2 L=1,LFI
```

```
      A1=0.
      A2=0.
      B1=0.
      B2=0.
      NSGA=NSG-L+1
      DO 5 I=2,NSGA
      A=1.-(I-1.)/NSGA
      J=I+L-1
      K=IABS(I-L)+1
      ICO=I
      IF(ICO.GT.LN)ICO=LN
      IF(J.GT.LN)J=LN
      IF(K.GT.LN)K=LN
      A1=A1+A*ROPRI(ICO)**2
      A2=A2+A*ROPRI(J)*ROPRI(K)
      B1=B1+A*ACFIM(ICO)**2
5     B2=B2+A*ACFIM(J)*ACFIM(K)
      A=CC*((A1+A2+B1+B2)*2.+ROPRI(L)**2-ACFIM(L)**2)
      C=CC*((A1-A2+B1-B2)*2.-ROPRI(L)**2+ACFIM(L)**2)
      DEL=B/SQRT(NSG-L+1.)*BINT
      POID(L)=DEL
      J=2*L-1
      K=J+1
      Y(J)=DEL*SQRT(1.+A)
      Y(K)=DEL*SQRT(ABS(1.+C))
      XCOR(J)=BINT*GFIN(L)
2     XCOR(K)=BINT*ACFIM(L)
      IDAT(10)=1.E-3/BINT*FD
      IF(IESSAI.LT.5) GO TO 21
      WRITE(1,1004) (Y(I),I=1,LFIN)
1003  FORMAT(' SIMPLE ERROR LAW',/,(6E12.5))
1004  FORMAT(' MEAN DEVIATIONS',/,(6E12.5))
1001  FORMAT(' COMPLEX ACF',/,(6I12))
      POID(1)=POID(1)*SQRT(2.)
C ******* POID ARE HERE THE MEAN SQUARE DEVIATION ON ACF GIVEN BY SIMPLE
C *****ERROR LAW.
      WRITE(1,1003)(POID(L),L=1,LFI)
21    IF(IESSAI.LT.2) GO TO 24
      WRITE(1,'(14H TOTAL POWER =,E12.4)')GFIN(1)*FD
      WRITE(1,1001) (XCOR(L),L=1,LFIN)
      WRITE(1,1006)X
1006  FORMAT(' X=',5E12.5)
      IF(IESSAI.GE.4)WRITE(1,1007)((DG(L,I)*BINT,L=1,32),I=1,IDV)
1007  FORMAT(' DERIVATIVES',/,(5(6E12.4,/),2E12.4))
24    IF(.NOT.LNOISE) GO TO 23
      DO 22 L=1,LFIN
      CALL GAUSS(RNDM)
22    XCOR(L)=XCOR(L)+RNDM*Y(L)
23    DO 3 L=1,LFI
      A1=Y(2*L-1)
      IF(Y(2*L).GT.A1)A1=Y(2*L)
C ******* ISIG2 IS EQUAL TO 1.E-6 TIMES THE SIGMA SQUARE ON THE POINTS
C *****OF THE ACF NORMALIZED TO 1.E6.
3     ISIG2(L)=A1*A1*1.E-6
      RETURN
      END
```

TRACE ON TEKTRONIX FROM FILE RESAL:DATA
(written in DIGER if compiled with #)

```
      DIMENSION LDAT(100),IPAR(65,14,18,2),IRAN(15),IPARAM(6),LEVEL(6)
      CHARACTER ACHAR*80,CHAR*20
      ASSEMBLY WIPE,DREL,POINT,MOVEC
      DOUBLE INTEGER IDAT(13)
      DIMENSION IH(24),JPAR(15,18,2),IDATIN(4),IDATFI(4),IYEAR(3)
      EQUIVALENCE (NEOF,IRECORD)
      LOGICAL PROFIL,INTRA,FITRA
      DATA JDAT1/0/,JREC/0/,NBA/0/,IH/1,2,3,4,5,6,7,8,9,10,11,12,13,14,
     * 15,16,17,18,19,20,21,22,23,24/,NREC/0/,NADEOF/0/
      WRITE(1,'(47H INITIAL AND FINAL DATES ?(YEAR,MONTH,DAY,HOUR),/)')
      READ(1,*) IDATIN,IDATFI
      IDATI=((IDATIN(1)*100+IDATIN(2))*100.+IDATIN(3))*100.+IDATIN(4)
      IDATF=((IDATFI(1)*100+IDATFI(2))*100.+IDATFI(3))*100.+IDATFI(4)
      WRITE(1,300)
300   FORMAT(' TAPE (0 OR 1),NB OF FILES SKIPPED, NB OF RECORDS SKIPPED
     *, NB OF FILES ANALIZED',/)
      READ(1,*) NUMBAN,NBEOF,NREC,NEOFMA
      IF(NBEOF.GE.0) WRITE(ACHAR,130) NUMBAN,NBEOF
      IF(NBEOF.LT.0) WRITE(ACHAR,132) NUMBAN,(-NBEOF)
      CALL COMND(ACHAR)
      IF(NREC.GE.0) WRITE(ACHAR,131) NUMBAN,NREC
      IF(NREC.LT.0) WRITE(ACHAR,133) NUMBAN,(-NREC)
131   FORMAT('DEV-F M-T-',I1,' ADV-REC ',I4,1H')
133   FORMAT('DEV-F M-T-',I1,' BAC-REC ',I4,1H')
130   FORMAT('DEV-F M-T-',I1,' ADV-T-E ',I4,1H')
132   FORMAT('DEV-F M-T-',I1,' REV-T-E ',I4,1H')
      CALL COMND(ACHAR)
      ACHAR=''
      WRITE(ACHAR,'(4HM-T-,I1)') NUMBAN
      OPEN(9,FILE=ACHAR,ACCESS='R')
      NBEOF=NBEOF+1
      WRITE(1,'(27H NUMBER OF FILES TOGETHER ?,/)')
      READ(1,*)NADMA
1     READ(9,ERR=200)NEOF,JRECO,IDAT,JPAR
      GO TO 3
200   IF(ERRCODE.NE.3) GO TO 1
      NBEOF=NBEOF+1
      IF(NBEOF.GT.NEOFMA) STOP
      GO TO 4
3     IF(JRECO.GT.JREC) GO TO 10
      NADEOF=NADEOF+1
      IF(NADEOF.LT.NADMA) GO TO 10
4     WRITE(1,1004)(IYEAR(I),I=1,3),LDAT(1)/60.,LDAT(NREC)/60.
1004  FORMAT(' DATE ',3I2,'  HMIN=',F6.2,'  HMAX=',F6.2)
      GO TO 5
10    IF(ERRCODE.EQ.3) GO TO 1
      JREC=JRECO
      NREC=NREC+1
      DO 11 I=1,2
      DO 11 J=1,18
      DO 11 K=1,14
11    IPAR(NREC,K,J,I)=JPAR(K,J,I)
      DO 12 I=1,3
12    IYEAR(I)=IDAT(I)
      JDAT=IDAT(5)*60+IDAT(6)
      KDAT=IDAT(7)*60+IDAT(8)
      IF(KDAT.LE.JDAT) KDAT=KDAT+3600
      JDAT=(JDAT+KDAT)/120+60*IDAT(4)
      LDAT(NREC)=JDAT
      GO TO 1
```

```
5       WRITE(1,'( 9H WAIT ...,/)')
        CHAR=''
        INPUT(1)CHAR
        IF(CHAR(1:1).EQ.'R') GO TO 20
        IF(CHAR(1:2).EQ.'PA') GO TO 30
        IF(CHAR(1:1).EQ.'T') GO TO 50
        IF(CHAR(1:1).EQ.'F') GO TO 100
        IF(CHAR(1:2).NE.'PR') GO TO 5
        PROFIL=.TRUE.
        WRITE(1,'(23H HOUR MIN ?, HOUR MAX ?,/)')
        INPUT(1) IHMIN,IHMAX
        IHMIN=IHMIN*60
        IHMAX=IHMAX*60
        GO TO 5
20      WRITE(1,'( 9H RANGES ?,/)')
        IR=0
21      INPUT(1)R
        IF(R.EQ.0) GO TO 5
        IR=IR+1
        IRAN(IR)=(R-76)/6
        GO TO 21
30      WRITE(1,'(21H PARAMETERS, LEVELS ?,/)')
        IP=0
31      INPUT(1)I,L
        IF(I.EQ.0) GO TO 5
        IP=IP+1
        IPARAM(IP)=I
        LEVEL(IP)=L
        GO TO 31
100     JDAT1=0
        JREC=0
        NREC=0
        WRITE(1,'(27H NUMBER OF FILES TOGETHER ?,/)')
        READ(1,*)NADMA
        NADEOF=0
        GO TO 10
50      IF(PROFIL) GO TO 80
        JDAT=LDAT(1)/60
        NDAT=JDAT*60
        KDAT=LDAT(NREC)/60+1
        IMOVE=700
        DO 52 IPA=1,IP
        NPAR=IPARAM(IPA)
        L=LEVEL(IPA)
        DO 52 NRAN=1,IR
        KRAN=IRAN(NRAN)
        IRANGE=77+6*KRAN
        CALL MOVEC(0,30)
        PAUSE
        CALL MOVEC(400,IMOVE)
        IMOVE=IMOVE-20
        WRITE(1,1002) NPAR,IRANGE,IPAR(1,KRAN,NPAR,L)
1002    FORMAT(' PARAM',I3,'  RANGE ',I4,'   VAL. =',E13.4)
        CALL POINT (1,50)
        CALL DREL(1,60)
        J=1
        DO 51 I=JDAT,KDAT
        CALL MOVEC(J,30)
        WRITE(1,1003)(I+1)
1003    FORMAT(I7)
```

```
      CALL POINT (J,60)
      J=J+60
      CALL DREL(J,60)
51    CALL DREL(J,50)
      FITRA=.TRUE.
GF    IF(L.EQ.2) GO TO 55
      IF(IPAR(1,KRAN,9,L).NE.2.OR.(IPAR(1,KRAN,8,L)/(20.+SQRT(FLOAT
     * (IPAR(1,KRAN,17,1))))).GT.100.) FITRA=.FALSE.
55    IY=60+IPAR(1,KRAN,NPAR,L)    50.
      IF(IY.GT.990)IY=990
      IF(IY.LT.1)IY=1
      IF(FITRA)CALL POINT(1,IY)
      DO 52 I=2,NREC
      INTRA=FITRA
      FITRA=.TRUE.
      IX=LDAT(I)-NDAT+1
      IY=IPAR(I,KRAN,NPAR,L)+60
      IF(IY.GT.990)IY=990
      IF(IY.LT.1)IY=1
GF    IF(L.EQ.2) GO TO 56
      IF(IPAR(I,KRAN,9,L).NE.2.OR.(IPAR(I,KRAN,8,L)/(20.+SQRT(FLOAT
     * (IPAR(I,KRAN,17,1))))).GT.100.)50 FITRA=.FALSE.
56    IF(INTRA.AND.FITRA)CALL DREL(IX,IY)
      IF(FITRA)CALL POINT(IX,IY)
52    CONTINUE
      CALL MOVEC(0,700)
      GO TO 5
80    IMOVE=700
      DO 81 I=1,NREC
      IF(LDAT(I).LT.IHMIN) GO TO 81
      IF(LDAT(I).GT.IHMAX) GO TO 84
      CALL MOVEC(1,IMOVE)
      IMOVE=IMOVE-30
      WRITE(1,'( 7H TIME =,F6.2)') LDAT(I)/60.
      CALL POINT (50,50)
      CALL DREL(50,60)
      IY=50
      DO 87 KRAN=2,15
      IRANGE=77+6*KRAN
      IY=IY+50
      CALL DREL(60,IY)
      CALL DREL(50,IY)
      CALL MOVEC(0,IY)
      WRITE(1,'(I4)')IRANGE
      CALL POINT (60,IY)
87    CONTINUE
      DO 82 IPA=1,IP
      NPAR=IPARAM(IPA)
      L=LEVEL(IPA)
      IY=50
      FITRA=.TRUE.
      IX=IPAR(I,1,NPAR,L)+60
      IF(IX.GT.990)IX=990
      IF(IX.LT.1)IX=1
GF    IF(L.EQ.2) GO TO 85
      IF(IPAR(I,1,9,L).NE.2.OR.(IPAR(I,1,8,L)/(20.+SQRT(FLOAT
     * (IPAR(I,1,17,1))))).GT.100.) FITRA=.FALSE.
85    IF(FITRA)CALL POINT(IX,IY)50.
      DO 82 KRAN=2,15
      INTRA=FITRA
```

this test says that if either ISEG ≠ 2 (no convergence) or VAR > 2 + √SNR, then the point is not drawn.

```
      FITRA=.TRUE.
      IY=50+IY
       IX=IPAR(I,KRAN,NPAR,L)+60
      IF(IX.GT.990)IX=990
      IF(IX.LT.1)IX=1
CF    IF(L.EQ.2) GO TO 86
      IF(IPAR(I,KRAN,9,L).NE.2.OR.(IPAR(I,KRAN,8,L)/(20.+SQRT(FLOAT
     * (IPAR(I,KRAN,17,1))))).GT.100.) FITRA=.FALSE.
86    IF(INTRA.AND.FITRA)CALL DREL(IX,IY)
      IF(FITRA)CALL POINT (IX,IY)
82    CONTINUE
      CALL MOVEC(0,30)
      PAUSE
81    CONTINUE
84    CALL MOVEC(0,1000)
      PROFIL=.FALSE.
      GO TO 5
      END
```

FLD (used in DIGER for decoding the time)

```
)9BEG
)9ENT FLD
FLD,STF P1 %RECUPERATION DES TROIS PARAMETRES D APPEL
LDA P1;ORA DECG;COPY SA DT
% T CONTIENT DECALAGE GAUCHE DE N POSITIONS
LDA  P3 %RECUPERATION DE LA VALEUR A BRICOLER
EXR ST;STA MAN %RESULTAT INTERMEDIAIRE STOCKE DS MAN
LDA P2;ORA DECAD;AAA -20;COPY SA DT
% T CONTIENT SHA SHR (16-M)
LDA MAN;EXR ST;STA MAN
SAA 20;SUB P2;ORA DECG;COPY SA DT
% T CONTIENT SHA (16-M)
LDA MAN;EXR ST;STA MAN
LDA P2;ORA DECC;COPY SA DT
% T CONTIENT SHA ROT M
LDA MAN;EXR ST
EXIT AD1 %RETOUR A L'APPELLANT
DECG,SHA
DECAD,SHA ZIN SHR
DECC,SHA ROT
P1,0;P2,0;P3,0
MAN,0
)9END
)LINE
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% HOPEFULLY VERY SMALL AND FAST GRAPHIC ROUTINES
% J.L.ARMSTRONG.
%
% USES ABSOLUTE SCREEN COORDIATES (0,0) - (1023,780)
% FORTRAN CALLS
% CALL WIPE                   TO CLEAR SCREEN
% CALL BELL                   FOR A BELL
% CALL DRAW(IX1,IY1,IX2,IY2)  TO DRAW A LINE
% CALL DREL(IX,IY)            DRAW A LINE RELATIVE TO CURRENT CURSOR POSN.
% CALL POINT(IX,IY)           TO DRAW A POINT
% CALL PLOT(IX,IY,IC)         TO PLOT A CHARACTER AT (IX,IY)
%                             IC MUST CONTAIN THE OCTAL CODE FOR THE
%                             REQUIRED CHARACTER
% CALL MOVEC(IX,IY)           TO MOVE CURSOR AND RESET TERMINAL IN
%                             ALPHANUMERIC MODE
% CALL GRDEV(IUNIT)           TO CHANGE GRAPHICS CHANNEL (FOR RT)
%
% NOTES       1) ALL ROUTINES MUST BE DECLAIRED 'ASSEMBLY' IN FORTRAN
%             2) ALWAYS USE MOVEC AFTER ANY OF THE CALLS DRAW,POINT
%                TO RESET TO ALPHANUMBERIC MODE BEFORE OUTPUTTING TEXT
%             3) ALWAYS USE ONE OF DRAW OR POINT BEFORE DREL (OBVIOUS)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
)9BEG
)9ENT   WIPE  DRAW  DREL  POINT  BELL MOVEC GRDEV PLOT
WIPE,
        SAA 33;LDT DEV;MON 2;MON 65
        SAA 14;MON 2;MON 65;LDX (-153;SAA 0
LOOP,   MON 2;MON 65;JNC LOOP
        EXIT AD1
)FILL
BELL,
        SAA 7;LDT DEV;MON 2;MON 65;EXIT AD1
GRDEV,
        STT DEV;EXIT AD1
POINT,
        STT XLOC;STA YLOC;COPY SL DA;STA SAVEL
        SAA 35;LDT DEV;MON 2;MON 65
        JPL OUT;JPL OUT;JMP FIN
DRAW,
        STT XLOC;STA YLOC;COPY SD DA;STA X2;STX Y2
        SAA 35;LDT DEV;MON 2;MON 65;COPY SL DA;STA SAVEL
        JPL OUT;LDA X2;STA XLOC;LDA Y2;STA YLOC;JPL OUT;JMP FIN
MOVEC,
        STT XLOC;STA YLOC;COPY SL DA;STA SAVEL
        SAA 35;LDT DEV;MON 2;MON 65;JPL OUT
        SAA 37;LDT DEV;MON 2;MON 65;JMP FIN
PLOT,
        COPY ST DX;AAX -5;STX XLOC
        AAA -10;STA YLOC
        COPY SD DA;STA SYMB
        COPY SL DA;STA SAVEL
        SAA 35;LDT DEV;MON 2;MON 65;JPL OUT
        SAA 37;LDT DEV;MON 2;MON 65
        LDA SYMB;MON 2;MON 65;JMP FIN
DREL,
        STT XLOC;STA YLOC;COPY SL DX;STX SAVEL;JPL OUT;JMP FIN
OUT,
        LDA    YLOC              % Y COORDINATE
        AND    MMASK             % SET TOP 6 BITS TO ZERO
```

```
        SAD   ZIN SHR 5         % SHIFT BOTTOM 5 BITS INTO D-REGISTER
        ORA   HIGH              % BUILD HIGH Y BYTE
        LDT   DEV               % ADDRESS OF TERMINAL
        MON   2                 % WRITE HIGH Y BYTE
        MON   65                % ERROR RETURN
        SAA   0                 % CLEAR A-REGISTER
        SAD   5                 % INSERT LOW ORDER 5 BITS OF Y
        ORA   LOWY              % BUILD LOW Y BYTE
        MON   2                 % WRITE IT
        MON   65                % ERROR RETURN
        LDA   XLOC              % GET X COORDINATE
        AND   MMASK             % CLEAR TOP 6 BITS OF A-REGISTER
        SAD   ZIN SHR 5         % SHIFT 5 LOW ORDER BITS INTO D-REGISTER
        ORA   HIGH              % BUILD HIGH X BYTE
        MON   2                 % WRITE IT
        MON   65                % ERROR RETURN
        SAA   0                 % CLEAR A-REGISTER
        SAD   5                 % INSERT LOW ORDER 5 BITS OF X
        ORA   LOWX              % BUILD LOW X BYTE
        MON   2                 % WRITE IT
        MON   65                % ERROR RETURN
        EXIT
FIN,
        LDX SAVEL;COPY SX DL;EXIT AD1
DEV,    1                       % DEVICE 1 (BACKGROUND)
SYMB,   0                       % SYMBOL TO BE PLOTTED
SAVEL,  0
MMASK,  1777
XLOC,   0
YLOC,   0
X2,     0
Y2,     0
HIGH,   40
LOWY,   140
LOWX,   100
)9END
)9EOF
)LINE
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Example of analysis of Chatanika tapes CRIM, using the program TALK

```
@CHATR   reduced parameters used

SEED, NB PB ANALYZED
123456,10000
USE GENER ?
N            No, since real data
USE TALK ?
Y
 CONSTANTS READ
NB OF WRONG ACF LAGS ?
1                                   } the ACF point at 50 µs is wrong.
DESIGNATION OF WRONG ACF LAGS ?     }
6
ALTITUDE MINIMUM AND ALTITUDE MAXIMUM FOR ANALYSIS ?
89,145
DIFFERENCE BETWEEN 2 SUCCESSIVE RANGES (KM) ?
6
INITIAL AND FINAL DATES ?(YEAR,MONTH,DAY,HOUR)
78,3,16,9,78,3,16,10
TAPE (0 OR 1),NB OF FILES SKIPPED, NB OF RECORDS SKIPPED, NB OF FILES ANALIZED
0,1,0,15
******** FILE NB  3   REC. NB   1   DATE: 78  3 16 FROM  9 6 8 10  912 0
IDAT(7,9,11,12,14)=  403   32    0   3310000
 SNR=  3/100, POWER=    2  ALT=  89 KM, NUM=   0
 SNR=  5/100, POWER=    4  ALT=  95 KM, NUM=   0
 SNR= 16/100, POWER=   12  ALT= 101 KM, NUM=   0
GO TO NEXT PROBLEM ?
N
ENTER IN CONVERSATION ?        1st Fit from Model · Data
N           Ne       Te       Ti       V      P        CoL        X(?)
P. INI.   0.305    219.2    219.2    0.0   1.000       0.        0.49 1.00       0. 1.00
  3   2   0.246    152.4    152.4    0.0   1.000    1298.        0.49 1.00       0. 1.00
 =0.8378  0.012     16.8     16.8    0.0   0.000      71.        0.00 0.00       0. 0.00
GO TO NEXT PROBLEM ?
N
ENTER IN CONVERSATION ?          2nd fit from Model : Data
N(2 iterations) converges
 (2)  (2) 0.248    159.0    159.0    0.0   1.000    1555.        0.57 1.00       0. 1.00
 0.8434  [0.014     33.0     33.0    0.0   0.000     796.        0.24] 0.00      0. 0.00
GO TO NEXT PROBLEM ?                                                  errors
N          <- we want to see the spectrum and ACF's for this problem.
ENTER IN CONVERSATION ?
Y
EXPLANATIONS ?
Y
TYPE -FFT- TO CHANGE THE NB OF PTS OF FFT (DEFAULT 64)
     -PARAM- TO CHANGE THE PARAM. ANALYSED
     -MASS- TO CHANGE THE ION MASS
     -I,- (I=1,2,...,OR 10) TO CHANGE THE I TH PARAM.
     -OSPEC-,-OACFTHE-,-OACFEXP- TO GET CORRESPONDING OUTPUTS
     -OTIT- TO GET THE TITLE;-OPRINT- TO PRINT ACF-S + RES.
     -L-P- TO PUT THE NEXT OUTPUTS ON LINE-PRINTER
     -MOD- TO LEAVE THE CONVERSATION PROGRAM AND USE THE MODEL
     VALUES FOR THE NEXT ANALYSIS
     -END- TO LEAVE THE CONVERSATION PROGRAM AND GO IN ANALYSIS
WAITING ...
OSPEC
SPECTRUM ...    NB OF POINTS = 15          FREQUENCY SPACING =   781.250
 1.0000  0.9664  0.8535  0.6535  0.4168  0.2260  0.1114  0.0536  0.0265  0.0139
 0.0079  0.0048  0.0032  0.0022  0.0016
WAITING ...
OACFEXP
```

```
ACF EXPERIMENTAL ...
 1.0000  0.0000  0.9545  0.0000  0.8069  0.0000  0.6604  0.0000  0.7619  0.0000
 0.0000  0.0000  0.5107  0.0000  0.4663  0.0000  0.4047  0.0000  0.3876  0.0000
 0.3530  0.0000  0.2862  0.0000  0.2071  0.0000  0.1518  0.0000  0.1097  0.0000
-0.0078  0.0000 -0.2792  0.0000  0.0000  0.0000  4.8253  0.0000  0.8351  0.0000
 0.0911  0.0000 -0.1593  0.0000 -0.1161  0.0000  0.0155  0.0000  0.0756  0.0000
 0.0203  0.0000 -0.1523  0.0000 -0.3868  0.0000 -0.5333  0.0000 -0.4608  0.0000
-0.2519  0.0000 -0.0698  0.0000
WAITING ...
OACFTHE
ACF THEORETICAL ...   NB OF LAGS = 32
 1.0000  0.0000  0.9892  0.0000  0.9578  0.0000  0.9079  0.0000  0.8433  0.0000
 0.7678  0.0000  0.6859  0.0000  0.6012  0.0000  0.5172  0.0000  0.4363  0.0000
 0.3603  0.0000  0.2905  0.0000  0.2279  0.0000  0.1729  0.0000  0.1258  0.0000
 0.0866  0.0000  0.0549  0.0000  0.0299  0.0000  0.0110  0.0000 -0.0029  0.0000
-0.0127  0.0000 -0.0192  0.0000 -0.0230  0.0000 -0.0247  0.0000 -0.0248  0.0000
-0.0236  0.0000 -0.0216  0.0000 -0.0190  0.0000 -0.0163  0.0000 -0.0137  0.0000
-0.0113  0.0000 -0.0092  0.0000
WAITING ...
PA
NUMBER OF PARAM. FITTED = 3
ORDER OF PARAMETERS =  1  3  7  4  2  5  6  8  9 10
NEW NUMBER AND DESIGNATION OF PARAM. FITTED= ?
0,1   <- to say that we go to next problem
WAITING ...
E
 SNR= 42/100, POWER=    37  ALT= 107 KM, NUM=    1
GO TO NEXT PROBLEM ?
N
ENTER IN CONVERSATION ?
N
P. INI.  0.812   252.8   252.8    0.0  1.000     0.      0.21 1.00      0. 1.00
  2   2  0.827   250.6   250.6    0.0  1.000   706.      0.21 1.00      0. 1.00
 0.9303  0.014    10.7    10.7    0.0  0.000    15.      0.00 0.00      0. 0.00
GO TO NEXT PROBLEM ?
N
ENTER IN CONVERSATION ?
N
  5   2  0.815   219.7   219.7    0.0  1.000   145.      0.05 1.00      0. 1.00
 0.8946  0.016    18.7    18.7    0.0  0.000   280.      0.09 0.00      0. 0.00
GO TO NEXT PROBLEM ?
N
ENTER IN CONVERSATION ?
Y
EXPLANATIONS ?
N
WAITING ...
PA
NUMBER OF PARAM. FITTED = 3
ORDER OF PARAMETERS =  1  3  7  4  2  5  6  8  9 10
NEW NUMBER AND DESIGNATION OF PARAM. FITTED= ?
3,2,3,7    we try to adjust Te, Ti, X(7)
WAITING ...
IE
INDEX FOR OUTPUTS =  1       ;NEW NUMBER = ?
4         to have many outputs
WAITING ...
E
  0           2.6678E+01  1.0000E+00  2.4404E-04  0.0000E+00  1.0000E+00
 0.0000E+00  4.5393E-02  1.0000E+00  0.0000E+00  1.0000E+00  8.9464E-01
  1           2.6678E+01  9.5826E-01  2.4706E-04  0.0000E+00  1.0000E+00
 0.0000E+00 -8.7209E-03  1.0000E+00  0.0000E+00  1.0000E+00  8.9358E-01
  2           2.6678E+01  9.5826E-01  2.4706E-04  0.0000E+00  1.0000E+00
 0.0000E+00 -8.7209E-03  1.0000E+00  0.0000E+00  1.0000E+00  8.9358E-01
  0           0.0000E+00  8.4900E-02  6.5310E-06  0.0000E+00  0.0000E+00
 0.0000E+00  1.1523E-01  0.0000E+00  0.0000E+00  0.0000E+00  8.9358E-01
```

```
  2   2  0.795   205.4   214.4    0.0  1.000     -27.     -0.01 1.00        0. 1.00
 0.8936  0.042    28.7    11.3    0.0  0.000     362.      0.12 0.00        0. 0.00
GO TO NEXT PROBLEM ?
Y
 SNR= 79/100, POWER=   78  ALT= 113 KM, NUM=    2
GO TO NEXT PROBLEM ?
Y
 SNR= 95/100, POWER=  104  ALT= 119 KM, NUM=    3
GO TO NEXT PROBLEM ?
Y
 SNR= 91/100, POWER=  109  ALT= 125 KM, NUM=    4
GO TO NEXT PROBLEM ?
Y
 SNR= 78/100, POWER=  103  ALT= 131 KM, NUM=    5
GO TO NEXT PROBLEM ?
N
ENTER IN CONVERSATION ?
N
P. INI.  1.892   339.0  2143.9    0.0  0.998      0.       0.00 1.00        0. 1.00
  0            6.3457E+00  1.5813E-01  7.8127E-05  0.0000E+00  9.9808E-01
 0.0000E+00  0.0000E+00  1.0000E+00  0.0000E+00  1.0000E+00  4.5585E+00
  1            9.3510E+00  4.3049E-01  9.8887E-05  0.0000E+00  9.9808E-01
 0.0000E+00  0.0000E+00  1.0000E+00  0.0000E+00  1.0000E+00  1.2636E+00
  2            1.4383E+01  7.2472E-01  1.1989E-04  0.0000E+00  9.9808E-01
 0.0000E+00  0.0000E+00  1.0000E+00  0.0000E+00  1.0000E+00  8.9292E-01
  3            1.7987E+01  8.5792E-01  1.2831E-04  0.0000E+00  9.9808E-01
 0.0000E+00  0.0000E+00  1.0000E+00  0.0000E+00  1.0000E+00  7.4388E-01
  4            1.8778E+01  8.7471E-01  1.2948E-04  0.0000E+00  9.9808E-01
 0.0000E+00  0.0000E+00  1.0000E+00  0.0000E+00  1.0000E+00  7.1695E-01
  5            1.8778E+01  8.7471E-01  1.2948E-04  0.0000E+00  9.9808E-01
 0.0000E+00  0.0000E+00  1.0000E+00  0.0000E+00  1.0000E+00  7.1695E-01
  0            2.0985E+00  1.0554E-01  4.3442E-06  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  7.1695E-01
  5   2  2.038   682.7   780.5    0.0  0.998      0.       0.00 1.00        0. 1.00
 0.7170  0.117    57.7    52.4    0.0  0.000      0.       0.00 0.00        0. 0.00
GO TO NEXT PROBLEM ?
N
ENTER IN CONVERSATION ?
Y
EXPLANATIONS ?
N
WAITING ...
OSPEC
SPECTRUM ....     NB OF POINTS = 11            FREQUENCY SPACING =  1562.500
 1.0000  1.0126  1.0454  1.0697  0.9911  0.6872  0.3111  0.1022  0.0282  0.0069
 0.0015
WAITING ...
OACFEXP
ACF EXPERIMENTAL ....
 1.0000  0.0000  0.9466  0.0000  0.7579  0.0000  0.4962  0.0000  0.3612  0.0000
 0.0000  0.0000  0.0405  0.0000 -0.1254  0.0000 -0.1618  0.0000 -0.1290  0.0000
-0.1012  0.0000 -0.0910  0.0000 -0.0693  0.0000  0.0076  0.0000  0.1371  0.0000
 0.2777  0.0000  0.3680  0.0000  0.0000  0.0000 -0.9229  0.0000 -0.0956  0.0000
-0.0365  0.0000 -0.0262  0.0000 -0.0071  0.0000  0.0087  0.0000  0.0041  0.0000
-0.0237  0.0000 -0.0689  0.0000 -0.1078  0.0000 -0.1045  0.0000 -0.0584  0.0000
-0.0194  0.0000 -0.0075  0.0000
WAITING ...
OACFTHE
ACF THEORETICAL ....    NB OF LAGS = 32
 1.0000  0.0000  0.9453  0.0000  0.7936  0.0000  0.5780  0.0000  0.3425  0.0000
 0.1292  0.0000 -0.0314  0.0000 -0.1261  0.0000 -0.1585  0.0000 -0.1438  0.0000
-0.1022  0.0000 -0.0527  0.0000 -0.0094  0.0000  0.0202  0.0000  0.0343  0.0000
 0.0354  0.0000  0.0280  0.0000  0.0170  0.0000  0.0062  0.0000 -0.0020  0.0000
-0.0067  0.0000 -0.0082  0.0000 -0.0072  0.0000 -0.0050  0.0000 -0.0024  0.0000
-0.0002  0.0000  0.0012  0.0000  0.0017  0.0000  0.0015  0.0000  0.0010  0.0000
 0.0004  0.0000 -0.0001  0.0000
```

```
WAITING ...
PA
NUMBER OF PARAM. FITTED = 3
ORDER OF PARAMETERS =  1  2  3  4  5  6  7  8  9 10
NEW NUMBER AND DESIGNATION OF PARAM. FITTED= ?
4,1,2,3,5
WAITING ...
E
  0          1.8778E+01  8.7471E-01  1.2948E-04  0.0000E+00  9.9808E-01
 0.0000E+00  0.0000E+00  1.0000E+00  0.0000E+00  1.0000E+00  7.2290E-01
  1          2.6987E+01  1.0591E+00  1.5194E-04  0.0000E+00  6.2205E-01
 0.0000E+00  0.0000E+00  1.0000E+00  0.0000E+00  1.0000E+00  8.4017E-01
  1          2.2882E+01  9.6692E-01  1.4071E-04  0.0000E+00  8.1006E-01
 0.0000E+00  0.0000E+00  1.0000E+00  0.0000E+00  1.0000E+00  7.1114E-01
  2          3.9636E+01  1.2382E+00  1.8394E-04  0.0000E+00  1.6555E-01
 0.0000E+00  0.0000E+00  1.0000E+00  0.0000E+00  1.0000E+00  1.3077E+00
  2          3.1259E+01  1.1026E+00  1.6232E-04  0.0000E+00  4.8781E-01
 0.0000E+00  0.0000E+00  1.0000E+00  0.0000E+00  1.0000E+00  8.2812E-01
  2          2.7071E+01  1.0347E+00  1.5152E-04  0.0000E+00  6.4894E-01
 0.0000E+00  0.0000E+00  1.0000E+00  0.0000E+00  1.0000E+00  7.2783E-01
  2          2.4977E+01  1.0008E+00  1.4611E-04  0.0000E+00  7.2950E-01
 0.0000E+00  0.0000E+00  1.0000E+00  0.0000E+00  1.0000E+00  7.1127E-01
  2          2.2882E+01  9.6692E-01  1.4071E-04  0.0000E+00  8.1006E-01
 0.0000E+00  0.0000E+00  1.0000E+00  0.0000E+00  1.0000E+00  7.1127E-01
  3          5.1890E+00  6.3156E-01  3.3095E-05  0.0000E+00  1.0742E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  7.1114E-01
  2  4  2.103    639.1    660.9     0.0  0.810       0.      0.00 1.00       0. 1.00
 0.7113  0.814   705.0    310.9     0.0  1.074       0.      0.00 0.00       0. 0.00
GO TO NEXT PROBLEM ?
Y         no convergence                  very large error on composition
 SNR= 62/100, POWER=    89  ALT= 137 KM, NUM=   6
GO TO NEXT PROBLEM ?
Y
 SNR= 53/100, POWER=    84  ALT= 143 KM, NUM=   7
GO TO NEXT PROBLEM ?
N
ENTER IN CONVERSATION ?
110
P. INI.  2.211   940.0   733.7     0.0  0.964       0.      0.00 1.00       0. 1.00
  0          2.1674E+01  1.2811E+00  1.3355E-04  0.0000E+00  9.6447E-01
 0.0000E+00  0.0000E+00  1.0000E+00  0.0000E+00  1.0000E+00  9.7368E-01
  1          1.4238E+01  8.7222E-01  1.2571E-04  0.0000E+00  9.6447E-01
 0.0000E+00  0.0000E+00  1.0000E+00  0.0000E+00  1.0000E+00  9.0336E-01
  2          1.5635E+01  9.3078E-01  1.2653E-04  0.0000E+00  9.6447E-01
 0.0000E+00  0.0000E+00  1.0000E+00  0.0000E+00  1.0000E+00  8.0418E-01
  3          1.5635E+01  9.3078E-01  1.2653E-04  0.0000E+00  9.6447E-01
 0.0000E+00  0.0000E+00  1.0000E+00  0.0000E+00  1.0000E+00  8.0418E-01
  0          2.1758E+00  1.3788E-01  5.1700E-06  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  8.0418E-01
  3  2  1.777    760.8    817.4     0.0  0.964       0.      0.00 1.00       0. 1.[illegible]
 0.8042  0.128    77.3     66.8     0.0  0.000       0.      0.00 0.00       0. 0.00
GO TO NEXT PROBLEM ?
Yes
 SNR= 42/100, POWER=    73  ALT= 149 KM, NUM=   8
 SNR= 39/100, POWER=    73  ALT= 155 KM, NUM=   8
 SNR= 35/100, POWER=    70  ALT= 161 KM, NUM=   8
 SNR= 29/100, POWER=    63  ALT= 167 KM, NUM=   8
******** FILE NB  3   REC. NB   2   DATE: 78  3 16 FROM  91258 TO  919 0
IDAT(7,9,11,12,14)=  403   32    0   3310000
 SNR= 22/100, POWER=    13  ALT=  89 KM, NUM=   8
GO TO NEXT PROBLEM ?
Yes
 SNR= 26/100, POWER=    18  ALT=  95 KM, NUM=   9
GO TO NEXT PROBLEM ?
Y
 SNR= 36/100, POWER=    28  ALT= 101 KM, NUM=  10
```

```
GO TO NEXT PROBLEM ?
N
ENTER IN CONVERSATION ?
N
P. INI.  0.623   219.2   219.2    0.0  1.000      0.      0.49 1.00       0. 1.00
 0            2.0428E+01  1.0000E+00  2.4433E-04  0.0000E+00  1.0000E+00
 0.0000E+00   4.8954E-01  1.0000E+00  0.0000E+00  1.0000E+00  9.7192E-01
  1           2.4467E+01  1.0000E+00  2.7688E-04  0.0000E+00  1.0000E+00
 0.0000E+00   4.8954E-01  1.0000E+00  0.0000E+00  1.0000E+00  7.2449E-01
  2           2.5036E+01  1.0000E+00  2.7968E-04  0.0000E+00  1.0000E+00
 0.0000E+00   4.8954E-01  1.0000E+00  0.0000E+00  1.0000E+00  7.2127E-01
  3           2.5036E+01  1.0000E+00  2.7968E-04  0.0000E+00  1.0000E+00
 0.0000E+00   4.8954E-01  1.0000E+00  0.0000E+00  1.0000E+00  7.2127E-01
  0           1.1809E+00  0.0000E+00  7.6352E-06  0.0000E+00  0.0000E+00
 0.0000E+00   0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  7.2127E-01
  3   2  0.582   167.3    167.3    0.0  1.000    1360.     0.49 1.00       0. 1.00
 0.7213  0.012     9.1      9.1    0.0  0.000      37.     0.00 0.00       0. 0.00
GO TO NEXT PROBLEM ?
Y
 SNR= 78/100, POWER=   68  ALT= 107 KM, NUM=  11
GO TO NEXT PROBLEM ?
N
ENTER IN CONVERSATION ?
N
P. INI.  1.431   252.8   252.8    0.0  1.000      0.      0.21 1.00       0. 1.00
 0            4.0716E+01  1.0000E+00  2.2752E-04  0.0000E+00  1.0000E+00
 0.0000E+00   2.0775E-01  1.0000E+00  0.0000E+00  1.0000E+00  6.9504E-01
  1           3.9260E+01  1.0000E+00  2.2500E-04  0.0000E+00  1.0000E+00
 0.0000E+00   2.0775E-01  1.0000E+00  0.0000E+00  1.0000E+00  6.6347E-01
  2           3.9260E+01  1.0000E+00  2.2500E-04  0.0000E+00  1.0000E+00
 0.0000E+00   2.0775E-01  1.0000E+00  0.0000E+00  1.0000E+00  6.6347E-01
  0           1.2592E+00  0.0000E+00  3.8266E-06  0.0000E+00  0.0000E+00
 0.0000E+00   0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  6.6347E-01
  2   2  1.411   258.5    258.5    0.0  1.000     717.     0.21 1.00       0. 1.00
 0.6635  0.017     8.8      8.8    0.0  0.000      12.     0.00 0.00       0. 0.00
GO TO NEXT PROBLEM ?

 BATCH USER LOGGED OFF AT 19.39.52      27 JULY   1979

 TIME USED IS   33 SECS OUT OF   58 SECS
```

EISCAT publications

F. du Castel, O. Holt, B. Hultqvist, H. Kohl and M. Tiuri:
A European Incoherent Scatter Facility in the Auroral Zone (EISCAT).
A Feasibility Study ("The Green Report") June 1971. (Out of print).

O. Bratteng and A. Haug:
Model Ionosphere at High Latitude, EISCAT Feasibility Study, Report No. 9.
The Auroral Observatory, Tromsö July 1971. (Out of print).

A European Incoherent Scatter Facility in the Auroral Zone, UHF System and Organization ("The Yellow Report"), June 1974.

EISCAT Annual Report 1976.(Out of print).

P.S. Kildal and T. Hagfors:
Balance between investment in reflector and feed in the VHF cylindrical antenna.
EISCAT Technical Notes No. 77/1, 1977.

T. Hagfors:
Least mean square fitting of data to physical models.
EISCAT Technical Notes No. 78/2, 1978.

T. Hagfors:
The effect of ice on an antenna reflector.
EISCAT Technical Notes No. 78/3, 1978.

T. Hagfors:
The bandwidth of a linear phased array with stepped delay corrections.
EISCAT Technical Notes No. 78/4, 1978.

Data Group meeting in Kiruna, Sweden, 18-20 Jan. 1978
EISCAT Meetings No. 78/1, 1978

EISCAT Annual Report 1977

H-J. Alker:
Measurement principles in the EISCAT system
EISCAT Technical Notes No. 78/5, 1978

EISCAT Data Group meeting in Tromsö, Norway 30-31 May, 1978
EISCAT Meetings No. 78/2, 1978.

P-S. Kildal:
Discrete phase steering by permuting precut phase cables.
EISCAT Technical Notes No. 78/6, 1978

EISCAT UHF antenna acceptance test.
EISCAT Technical Notes No. 78/7, 1978.

P-S. Kildal:
Feeder elements for the EISCAT VHF parabolic cylinder antenna.
EISCAT Technical Notes No. 78/8, 1978.

H-J. Alker:
Program CORRSIM: System for program development and software simulation of EISCAT digital correlator, User's Manual.
EISCAT Technical Notes No. 79/9, 1979.

H-J. Alker:
Instruction manual for EISCAT digital correlator.
EISCAT Technical Notes No. 79/10, 1979

H-J. Alker:
A programmable correlator module for the EISCAT radar system.
EISCAT Technical Notes No. 79/11, 1979.

T. Ho and H-J. Alker:
Scientific programming of the EISCAT digital correlator.
EISCAT Technical Notes No. 79/12, 1979.

S. Westerlund (editor):
Proceedings EISCAT Annual Review Meeting 1969. Part I and II,
Abisko, Sweden, 12-16 March 1979.
EISCAT Meetings No. 79/3, 1979.

J. Murdin:
EISCAT UHF Geometry.
EISCAT Technical Notes No. 79/13, 1979.

T. Hagfors:
Transmitter Polarization Control in the EISCAT UHF System.
EISCAT Technical Notes No. 79/14, 1979.

B. Törustad:
A description of the assembly language for the EISCAT digital
correlator.
EISCAT Technical Notes No. 79/15, 1979.

J. Murdin:
Errors in incoherent scatter radar measurements.
EISCAT Technical Notes No. 79/16, 1979.

EISCAT Digital Correlator. TEST MANUAL.
EISCAT Technical Notes No. 79/17, 1979.