KfK 2797
Februar 1980

On-line Gamma Spectroscopy Measuring Station for Cover Gas Monitoring at KNK II

G. Hoffmann, K.-D. Letz
Institut für Reaktorentwicklung
Projekt Schneller Brüter

Kernforschungszentrum Karlsruhe

KERNFORSCHUNGSZENTRUM KARLSRUHE
Institut für Reaktorentwicklung
Projekt Schneller Brüter

KfK 2797

On-line Gamma Spectroscopy Measuring Station for Cover Gas Monitoring at KNK II

G. Hoffmann
K.-D. Letz

Kernforschungszentrum Karlsruhe GmbH, Karlsruhe

Als Manuskript vervielfältigt
Für diesen Bericht behalten wir uns alle Rechte vor

Kernforschungszentrum Karlsruhe GmbH
ISSN 0303-4003

Meßplatz On-line Gammaspektroskopie zur Schutzgasüberwachung an der KNK II

Zusammenfassung

Zur Schutzgasüberwachung in der KNK II[1] wurde ein automatisiertes Gammaspektrometer entwickelt, das anhand der gemessenen Gammaspektren folgende Aussagen über Hüllrohrschäden ermöglichen soll: Art, Größe, zeitliche Änderung und weiterer Verlauf des Schadens. Im vorliegenden Bericht wird die Hardware und Software erläutert. Außerdem wurde eine Betriebsanleitung für den Meßplatz erstellt, die es erlaubt, den Meßplatz ohne nähere Kenntnis der Manuals für die einzelnen Hardware-Komponenten zu betreiben.

Abstract

An automated gamma spectrometer was developed for cover gas monitoring at KNK II[1], which, by the gamma spectra measured, is to allow the following statements to be made on fuel cladding failure: type, size, variation with time and subsequent development of the failure. In this report the hardware and software will be explained. Besides, an instruction manual was written for the measuring station, which allows to operate it without detailed knowledge of the manuals for the individual hardware components.

---

1) Kompakte Natriumgekühlte Kernreaktoranlage mit schnellem Kern.

Contents

1. Introduction

An automated, computer controlled measuring system for recording and evaluation of gamma spectra was developed which allows experiments to be made on the detection of gamma emitters in the KNK II cover gas. At a bypass of the cover gas pipe the distribution of the gamma energies of gaseous and volatile nuclides in argon is measured by a Ge-detector system and the integral gamma flow is determined by a precipitator. In the most adverse case, i.e. in the event of major damage occurring, changes in activity concentrations by 7 orders of magnitude are expected. To obtain still satisfactory gamma spectra after such considerable changes of activity concentration, a variable detector geometry must be used. These considerations led to the construction of a collimator /1/ positioned by the computer as a function of the precipitator counting rate. Moreover, the computer defines the measuring and sequence of measurement times for the multichannel analyzer in accordance with the precipitator counting rate and the reactor power. The gamma spectra so recorded are evaluated on-line, written out on a "Teletype" and stored on an IBM-compatible 9-track tape.

2. Measurement Task

By detection of about 8 significant gaseous fission nuclides in the cover gas the following primary information can be obtained:

a) types of nuclide present,

b) concentration of nuclides present,

c) variation with the time of the spectrum. The existing concentrations may undergo the following variations
   - by the same factors,
   - by dissimilar factors, or
   - additional nuclides might appear.

After insertion of free fuel test surfaces in KNK II and after relevant experience has been obtained, it is intended to extend the program so that the following secondary information might be obtained on the basis of criteria stored in the computer:

d) type and size of the failure,

e) variation with time of the failure,

f) prediction about the subsequent development of the failure (prediction of damage development).

3. Layout and Function of the Measuring Station

A Ge-detector and a precipitator are installed at a bypass of the cover gas pipe. The signal furnished by the Ge-detector is fed into the multichannel analyzer (MCA) via a preamplifier and main amplifier. The gamma spectrum recorded in the MCA is written both on a 9-track tape and into the computer, where it is evaluated, and printed out on the teletype. Then the computer causes the spectrum to be displayed on the x-y plotter. The pulse rate of the precipitator is monitored by the computer. Depending on whether limit values are attained, the computer controls the collimator interface via the device actuator and the collimator via the collimator control to make them move into a position corresponding to the pulse rate. The respective collimator position and operating and/or alarm conditions, respectively, of the collimator are indicated on the indicator panel of the collimator interface. The total process of measurement is controlled via the ONGAM program written in CLASS[1]. The measurement print out and the dialog example in annex C is written in German because these must be legible for German reactor operators.

---

1) Canberra's Laboratory Automation Software System

3.1 Hardware

A general view of the hardware structure is given in Fig. 1. The individual components of the measuring station are treated here only to the extent they are developments of our own or are equipment from outside modified by us, whilst in all other cases reference is made to the respective equipment manuals [2 to 10].

3.1.1 Collimator

The collimator is used to reduce the γ-intensity at the detector. Via two motors the following operations can be performed:

1st The cryostat with the detector can be moved away from the cover gas pipe about 20 cm.

2nd A lead attenuator can be elevated between the covergas pipe and detector for additional screening. The lead atternator contains two bore holes with different diameters.

The three possible positions of the collimator are shown in Fig. 2. The interlocking of the two collimator motors is checked twice as well as in the collimator interface as in the collimator control to prevent damaging of the detector.

3.1.2 Collimator Interface

a) Layout

The collimator interface has been designed as an NIM-plug-in unit [11]. The front side is equipped with a rotary switch with four switching positions: for starting by hand the three collimator positions, and one computer position. A green and a red signal lamp for each collimator position has been provided, which indicate the following conditions:

- Collimator in position m (m = 1,2,3);
  green m on.

- Collimator moves from m to n (n = 1,2,3);
  red m on, green n flashes until position n is reached.

- As soon as the position n is reached, red goes off and green goes on permanently.

b) Function

The computer transmits via the two relay outputs of the device actuator the following signals to the collimator interface:

00 Interface reset.
10 Collimator to position 1.
01 Collimator to position 2.
11 Collimator to position 3.

These signals are converted by the interface into moving instructions for the two collimator motors while the respective interlockings get effective. When the collimator has reached the requested position, a READY message is given from the interface to the computer.

3.1.3 Collimator Control

By the collimator control the moving instructions from the interface are checked once more for correctness of their "cryostat-attenuator" locking and the respective motors are connected or disconnected, respectively. If a motor protection gets actuated or a limit switch is exceeded, a disturbance message is given to the collimator interface. The two driving motors can also be operated manually by push buttons. Circuit diagrams are shown in Figs. 3 to 6.

3.1.4 Device Actuator

The device actuator operates by the following operation codes on address 81 [5].

Opcode 2: output 1 and output 2 off.

Opcode 8: output 1 on.

Opcode 9: output 2 on.

The "Jumper T" was set to BUSSY 2.

In the following paragraph the positions are listed for the programming switches which are on whilst all other switch positions are off.

Switch S1: pos. 1 and pos. 8.

Switch S2: pos. 1, pos. 4 and pos. 8.

Switch S3: pos. 1, pos. 2 and pos. 3.

Switch S4: pos. 1, pos. 4 and pos. 7.

Switch S5: pos. 1, pos. 3, pos. 5 and pos. 8.

Switch S6: pos. 1, pos. 5 and pos. 7.

3.1.5 Plotter Interface

a) Layout

The plotter interface has been designed as an NIM plug-in unit. The following switches are on the front side:

- rotary switch for 2 calibration positions (zero point and 5V full scale deflection) and 1 measuring position;
- reversing switch for manual or computer operation;
- button for chard drive of plotter;
- program stop switch with indication.

Diagrams see Figs. 7 to 9.

b) Function

The plotter interface is controlled via a voltage (5V DC) of the MCA (plug J 112, pin 9).

If the MCA is set in the plot mode by the computer, the voltage decreases to zero during the output. In the plotter interface this change in voltage is interpreted as follows:

- Voltage decreases to zero for the first time: chart drive on.

- Voltage on again: chart drive stops.
  Switching-on is done via the computer by interruption of the plot mode of the MCA in case the paper has sufficiently advanced, i.e. the previous spectrum plot has been fully wound up.

- Voltage returns to zero for the second time:
  lower the pen, write the spectrum. As soon as the last channel has been output, the MCA turns on again 5V.

- Voltage on again: lift up pen, disconnect the plotter input from the MCA and short-circuit its input terminal. The plotter interface has again reached its initial position. Independent of the chart feeding or plotting the position of the plotter interface is monitored by the computer via "External Interrupt" in the Datanim interface. The process of control is represented in Fig. 10.

3.2 Software

The ONGAM program runs as a user program under the CIMSES[1)] [12] and CLASS [13,14] programs. To run ONGAM the realtime clock in CLASS was modified such that access to the clock is possible in the 0.1 second mesh and not only in the minute mesh as in the original version. Due to this measure the time statements undergo variations [13, see 7.4].

---

1) Canberra Industry Mass Storage Executive System

a) Time setting $STIME (HH, MM, ZZZ)

b) Wait until time has been reached $WTIME (HH, MM, ZZZ).

c) Printing of time TYPE $TIME yields a printout like HHMMZZZ.O

HH = hour
MM = minute
ZZZ = tenth of a second

Contrary to the original version of CLASS the date is no longer contained in the $TIME function; it is taken over by the DATKOR subroutine in ONGAM.

3.2.1 Program Structure and Flow of ONGAM

Since CLASS does not allow to program interrupts and, on the other hand, the program flow depends on outside events (reactor power RL and precipitator counting rate PCPS) - RL and PCPS are measured by the RLCPS subroutine (A30, B18)[1] at predetermined dates - the 4K-memory of the multichannel analyzer is operated in a 2K mode so that no measuring values get lost. This means that spectra are collected for evaluation in the first memory half (2K) of the MCA. During evaluation and printing, respectively, measurement is started in the second half of MCA. Then RL and PCPS each are tested. In case significant changes are found, the spectrum of the second MCA half is read on tape for subsequent evaluation by means of DERAN [15]. If no change is found, the MCA is erased and possibly restarted.

A survey of the structure and flow of ONGAM (A1, B1) has been represented in Fig. 11. ONGAM calls the starting dialog programs for setting program control variables and then starts the STEUER control program (A8, B3) which, essentially, calls 6 different program flows as a function of the flags set in the subroutines. After termination or interruption of these program flows the control is returned to STEUER.

---

1) A = Annex A, B = Annex B

a) Cyclic Measurement

The NORMES subroutine (A9, B4) controls the sequence of the cyclic measurement by calling further subroutines. After NORMES has started, the starting date for the next spectrum is calculated then the collimator is moved in a position as a function of the precipitator counting rate and the multichannel analyzer is started. While the spectrum is collected RL and PCPS are continuously measured and compared to reference values. Depending on the results of this comparison it is decided whether the measurement is to be continued or interrupted. After termination of the measurement the spectrum is read out both on tape (data organization see Fig. 12) and into the computer and the second MCA half is started for measurement. By means of the AUSWER subroutine (A18, B13) the spectrum is evaluated and then the subroutines PEAVER (spectrum comparison; A22), MESPRO (measurement printout; A24), PLOT (display on x-y-plotter; A35, B23), and NOMORG (filing of spectrum data on the NORM1 to NORM10 files; A36) are started.

b) Special Measurement

If comparison of RL and PCPS with the respective reference values yields an intolerably great deviation, a special measurement is started. The additional measurement is controlled by the SONDER subroutine (A10, B5) and corresponds to the normal measurement except for the following additional measures required:

- Determination of new reference values of RL and PCPS prior to starting the measurements.
- Filing the results of evaluation in the LANG1 to LANG25 data files.

c) Decay Measurements

If the reactor power falls below a predetermined limit value, the ABKLI (A12, B7) subroutine is started. Until 24 hours after a scram a series of 12 spectra are started with different durations of measurement and pauses. All spectra collected are first stored by the MCLES (A33, B21) subroutine on the SP1 to SP12 cassette files without processing and written on tape at the same time. After the 9th decay measurement has been terminated, a sufficiently long measurement pause is available so that the spectra stored on the files SP1 to SP9 can be processed. The evaluation is made by the AUSWAB (A20, B14) subroutine, followed by the PEAVER, MESPRO and PLOT subroutines. After the 12th decay measurement the ABSCH (A13, B8) wait loop is started via STEUER. If during the decay measurement the reactor power exceeds the limit value Q(61), ABKLI is interrupted and a special measurement is started.

d) Calibration Measurement

At the end of a predeterminable number of days or if an intolerably great "channel drift" of the MCA is found in spectra evaluation, a calibration measurement is started. The EICHEN (A11, B6) subroutine controls the sequence of the quadratic energy calibration. First the collimator is brought into the calibration position (maximum shielding of the measurement section + calibration nuclides), then the MCA is started and after termination of the measurement by the FENAUS (A25, B15) subroutine the windows are read out, which have been set in the MCA for the calibration peaks, and, moreover, the complete spectrum is written on tape. Subsequently, the EIKOF (calculation of calibration coefficient; A26, B16), EIPRO (printout; A28) and PLOT subroutines are started and, moreover, the result of calibration is stored on the EICHØ file. In the course of calibration RL and PCPS are monitored as in every routine and EICHEN is possibly interrupted while SONDER or ABKLI is started.

e) Measurement Pause

The MEPAUS (A15, B10) subroutine is a wait loop until the next starting time for the following cyclic measurement has been reached or an interruption occurs, respectively, due to outside events (RL or PCPS). During the measurement pauses the measured results of RL and PCPS are printed on the teletype at predeterminable intervals of time.

f) Evaluations Performed at a Later Stage

If the NORMES or SONDER routines are precociously discontinued, the measured results are temporarily filed on the ALAR1 to ALAR5 alarm files.

Before activation of the MEPAUS routine it is verified whether measured results still to be logged are left in the alarm file, and if so, the ALBEA (alarm file processing; A16, B11) subroutine is started. In case non-evaluated spectra (due to the interruption of ABKLI) are still contained in the Sp1 to Sp12 files, ABNACH (decay measurement evaluation performed at a later stage; A17, B12) is started and only after that MEPAUS is started.

3.2.2 Description of Subroutines

For lack of space in the computer all the routines have been written without comments so that an additional description has to be given for some routines. Annex A contains the listings of all routines.

3.2.2.1 AUSWER Subroutine

AUSWER (A18, B13) serves to evaluate gamma spectra recorded and calculates by means of the algorithm used in DERAN the following characteristic data:

a) Precise Channel Position

Contrary to DERAN, no peak search run is used, but the channel with the maximum content is searched for predeterminable channel addresses and window widths in this window. If the channel address drives from the precise position of the channel by a predeterminable number of channels, a calibration measurement is initiated after the spectrum has been printout.

The precise channel position R is calculated by the equations (1)

$$R = K + \frac{I_{K+1} - I_{K-1}}{2(I_K - I_{K-1})} \qquad \text{for } I_{K+1} \geq I_{K-1}$$

(1)

$$R = K - \frac{I_{K-1} - I_{K+1}}{2(I_K - I_{K+1})} \qquad \text{for } I_{K+1} < I_{K-1}$$

K = channel address with maximum pulse number

I_K = counts in the channel K

b) Calculation of the Peak Limits

In the loops at label "rechts" and label "links" (see listing) the high and low limits of the peak are determined.

NR = number of channels on the high side of channel K

NL = number of channels on the low side of the channel K

c) Calculation of the Half-Value HV

HV = half amplitude of photo peak without background [counts]

$$HV = \frac{2\ I_K + I_{NL} + I_{NR}}{4} \qquad \text{for } I_{NL} > I_{NR}$$

(2)

$$HV = \frac{2\ I_K + I_{NL} + 0.9 \cdot I_{NL}}{4} \qquad \text{for } I_{NL} \leq I_{NR}$$

I_K = counts in the maximum

I_{NL} = counts at the low side limit

I_{NR} = counts at the high side limit

d) Full Width at Half Maximum HWB

HWB [keV] = NUEB · b

NUEB = number of channels above the half value of the photo peak

b = calibration factor (see Eq. 8)

e) Background UNT

$$N_p = N_{NR} + N_{NL} - 1$$

$$UNT = \frac{N_p}{2} (I_{NL} + I_{NR}) \quad \text{for} \quad I_{NL} > I_{NR} \qquad (3)$$

$$UNT = 0.95 \cdot N_p \cdot I_{NL} \quad \text{for} \quad I_{NL} \leq I_{NR}$$

N_p = number of channels contributing to the peak

f) Peak Content

$$I_p = \left(\sum_{Z=NL+1}^{NR-1} I_Z \right) - UNT$$

g) Relative Error

$$RF = \frac{DPK \cdot 100}{I_p} \qquad (4)$$

with

$$DPK = 2 \cdot \sqrt{Ip+UNT} + N_p \cdot \sqrt{I_{NL}} + N_p \cdot \sqrt{I_{NR}}$$

RF [%] = relative error

h) Determination of the Activity Concentration

$$AK = \frac{I_p \cdot 3{,}7 \cdot 10^{10}}{\varepsilon(E) \cdot \eta \cdot \alpha(E) \cdot V \cdot t} \qquad (5)$$

AK $\left[\frac{1}{s \cdot cm^3} \right]$ = activity concentration ε(E) [1] = relative detector response probability at the gamma energy E

η [1/s · Ci] = geometry factor

α(E) [1] = γ/d-ratio

V [cm^3] = pipe volume seen by the detector

t [s] = counting time

The values described above are then written into the array ROI1 for subsequent recording. Readout and evaluation of the spectrum from the MCA are made in two steps of 1K each for reasons of limited space in the computer memory:

- 1. Step

  Read 1.K into the locations SPEC(1) to SPEC(1o24) and evaluate until SPEC(1010); then SPEC(995) to SPEC(1024) are moved towards SPEC(1) to SPEC(30).

- 2. Step

  Read 2.K into the locations SPEC(31) to SPEC(1054) and evaluate it from SPEC(1) to SPEC(1054)

3.2.2.2 AUSWAB Subroutine

AUSWAB (A20, B14) is equally used for spectrum evaluation although only for decay measurements. The mode of operation is the same as for AUSWER except for the fact that the spectra are not read into the computer from the MCA but from the cassette.

3.2.2.3 EIKOF Subroutine

By means of EIKOF (A26, B16) the coefficients are calculated of the quadratic energy calibration and the channel addresses are determined for the peaks to be evaluated.

In case of quadratic dependence of the energy E of a peak on the channel position R, the following relation holds

$$E = a \cdot R^2 + b \cdot R + c \qquad (6)$$

To determine a, b and c, the 3 calibration peaks will be read out from the MCA by means of the FENAUS subroutine and evaluated with regard to the precise channel position, Eq.(1) The following relations hold for the 3 calibration factors a, b, and c

$$a = \frac{(E_3-E_1)(R_2-R_1) + (E_2-E_1)\ (R_1-R_3)}{(R_3{}^2-R_1{}^2)\ (R_2-R_1) + (R_2{}^2-R_1{}^2)(R_1-R_3)} \qquad (7)$$

$$b = \frac{E_2-E_1-a\ (R_2{}^2-R_1{}^2)}{R_2 - R_1} \qquad (8)$$

$$c = E_1 - a\ R_1{}^2 - b\ R_1 \qquad (9)$$

R_1, R_2, R_3 = precise channel position of the 3 calibration peaks;
E_1, E_2, E_3 [keV] = energy of the 3 calibration peaks

The channel address is calculated by resolution of Eq. 6 for R; the integer part of R gives the channel address.

3.2.2.4 Auxiliary Programs

The following auxiliary programs run directly under the CLASS Keyboard Processor:

a) INIT (A44)
   Call: ØU(3); RUN INIT
   INIT is an initializing program for the subroutines used in ONGAM such that they have to be called only

by their names without giving the unit number where they have been filed. This effects an automatic "RUN and DELETE" of these routines.

b) REMESS (A44)
Call: ØU(3); RUN REMESS
REMESS loads the arrays Q, ROI1 and ROI2 with the datas on the DATFIL, SAFIL1 and SAFIL2 data files.

c) STORE (A44)
Call: ØU(3); RUN STORE
Writes the contents of the arrays Q, ROI1 and ROI2 on the DATFIL, SAFIL1 and SAFIL2 files.

d) AUSGAB (A5)
Call: ØU(3); RUN AUSGAB
By means of AUSGAB all data arrays managed in the computer can be written fully or partly on teletype with any step width. AUSGAB provides a respective starting dialog.

e) SORT (A48)
Call: ØU(1); RUN SORT
Sorts the subroutines on unit 3 by an optimum proccessing sequence.

f) FILTES (A49)
Call: ØU(1); RUN FILTES
Test program for the data files on unit 2; tests new cassettes for correct writing and reading functions for all data files.

3.2.3 Data Organization

The arrays subsequently marked x are not present permanently in the computer; they are generated upon request and loaded from cassette and erased, after use.

a) The following arrays are managed in the computer:

Q (100)	contains controlling variables for the program sequence as well as measuring values (Fig. 12).
ROI1(300)	contains results of spectrum evaluation (Fig. 13).
ROI2(300)[x]	similar to ROI1 for comparison of spectra; contains results of the N-1 evaluation (Fig. 13).
EIF(200)[x]	contains results of the calibration measurement (Fig. 14).
SPEC(1054)[x]	array for recording a 1K-spectrum; it is used in normal and special measurements (Fig. 15).
SPEK(1090)[x]	arrays for recording a 1K-spectrum and of measured values; if it is used in the decay measurement (Fig. 16).

b) The following data files exist on the cassettes:

ba) Unit 3

DATFIL(100)	for recording the array Q
SAFIL1(300)	for recording the array ROI1
SAFIL2 (300)	for recording the array ROI2
EICHØ(200)	for recording the array EICH
SP1 -SP8(1090)	for recording the array SPEK
ALAR1 to ALAR5(300)	for recording the array ROI1

bb) Unit 2

NORM1 to NORM10(300) for recording ROI1.
The results of standard measurements are stored on these 10 data files, beginning with NORM1 to NORM10 and then again with NORM1, so that the results of the last ten measurements are available.

LANG1 to LANG25(300) likewise serve for recording ROI1, although here only special measurements are filed. In addition, the normal measurements can be filed at predeterminable intervals (days). This file was set up for a long-term retrospection still to be programmed.

SP9 - SP12(1090) for recording the array SPEK.

c) 9-Track Tape

Before a spectrum is written from the MCA on the 9-track tape, additional values are written into the channels 1-18 of the MCA by the computer (Fig. 17).

4. Operating Instruction

To operate the hardware of the measuring station reference is made to respective manuals; in this context, only the most significant items will be summarized.

4.1 Preparations for Startup

a) Starting CIMSES [12] and CLASS [13]
b) Multichannel Analyzer [2]

- Set 3 windows for the calibration peaks.
- Set cursor in one of the windows.
- Switch off all functions in the pushbottom field.
- Preset COUNT off.
- Preset TIME to LIVE and 00.
- FUNCTION to PHA.
- I/O DEVICE to REMOTE.

c) DATANIM Computer Interface [14]

- Switch to CLEAR, then to ONLINE.
- Check whether the "Manual" lamps are extinguished on all Datanim modules; if not, give a manual(O)↲ [1] instruction via keyboard.

---

[1] ↲ = RETURN-key

d) Dual Counter/Timer [4]

- TIME Switch off for both channels.
- Preset both channels to 00.
- Recycle off.

e) Plotter Interface

- Switch to Computer.
- Program stop off.

f) Collimator Interface

- Switch to Computer.

g) Tape Controller [8]

- Set tagword to number of the following spectrum -1.
- Switch to tagword increment.
- Automatic EOF off.
- Possibly clear "Error Lamp".

h) Pertec Tape [9]

- To Online.

4.2 Starting ONGAM

Before execution of the instructions a to c and after each RETURN the symbol * must be awaited which means that the CLASS Keyboard Processor is ready for accepting instructions.

a) Initializing the Subroutines
   ØU(3); RUN INIT ↵
   Duration about 15 minutes

b) Installation of Arrays
   DIMENS Q(100) ↵
   DIMENS ROI1(300) ↵
   DIMENS ROI2(300) ↵

c) Loading the arrays Q, ROI1 and ROI2 from the Casette
   $U(3); RUN REMESS ↵

d) ONGAM ↵
   The ONGAM routine is loaded from the cassette and starts
   with the starting dialog; see Annexex C1 to C5.

4.3 Loading of DUMMY Programs

In case of failure of hardware components part of the measuring station can continue operation; however, it is necessary in this case to load the respective DUMMY programs. This is done before execution of 4.2 d) in the following way:

a) 9-Track Tape Defective
   DELETE TAPE, MCLES, ABKLI ↵
   $U(2); DECLARE TAPE, MCLES, ABKLI ↵

b) Collimator Defective
   DELETE FAHR ↵
   $U(2); DECLARE FAHR ↵

c) x-y-Plotter Defective
   DELETE PLOT ↵
   $U(2); DECLARE PLOT ↵

4.4 Stop and Restart

By means of the CTRL P keys the running program can be stopped at any time by teletype. This measure brings about an immediate interruption and the control is transferred to the CLASS Keyboard Processor which gives a * signal.

Since under this method no program data can be saved and the date of stop is a random date, the program cannot be continued at the point of interruption, which means that ONGAM must be restarted; see 4.2 d).

By means of the "program stop" switch in the plotter interface the running program can be stopped at some predetermined points while saving some important data so that a defined restart is possible. If the program stop switch is set, the following printout is displayed on the teletype after the given point of interruption has been reached:

```
*** HHMMZZZ,Ø STOP-NAME

*
```

NAME = name of interrupted subroutine
HHMMZZZ.Ø = date of interruption according to 3.2 c)

The asterisk in the 2nd line indicates that the CLASS Keyboard Processor is ready to accept new instructions. Restarting is done by RESTART ↲

The RESTART routine (A7, B2) is loaded and starts with a short dialog. Now it is possible to call the start dialog programs or to restart immediately the interrupted program.The CHECK 1 contained in the dialog refers to items 4.2.1 to 4.2.7.

4.5 Starting an Extraordinary Calibration Measurement

To start an extraordinary calibration measurement, the following procedure is recommended:

- Stop the program by means of the program stop switch (most suitably during measurement pause) and wait for the READY message of the CLASS Keyboard Processor.
  Then do:
- Q(41) = Q(46) ↲

  or Q(81) = 1 ↲

  With Q(81) = 1 the calibration measurement is always terminated independent of outside events (the log is supressed if applicable). With Q(41) = Q(46) the calibration can be interrupted as a function of the measured values of RL and PCPS.
- RESTART ↲

5. Status of Work

The measuring station was installed in early March 1978 in the experimenter's room of KNK II. Since that date background measurements have been made. Both the spectrum of the environmental radiation at the measuring point have been recorded and the nuclides present in the cover gas produced by the uranium surface contamination of the fuel elements. Since starting the measuring station works without any severe malfunction and and fullfills all anticipations.

On April 1, 1979 the first cladding failure in the KNK II occured. Fig. 18 shows the printout of the last background spectrum No. 688 taken by a cyclic measurement. Initiated by the increasing of the integral detector counting rate[1)] the computer starts at 16.56.45 h a special measurement. The printout of this spectrum No. 689 is shown in Fig. 19. As mentioned before, all printout are in German, a translation to English is given in Fig. 20. Fig. 21 shows the last bakcground spectrum and the spectrum measured at the beginning of the cladding failure. The nuclids in parantheses are present in the spectrum but not yet set for evaluation in the computer.

---

1) Because the precipitator was not yet installed, the integral detector counting rate is taken to control the start of special measurements.

6. Literature

/¯1_7 G. Schmitz:
Personal communication

/¯2_7 Canberra:
Multichannel Analyzer Model 8100, Instruction Manual.

/¯3_7 Canberra:
Computer Interface Model 6911, Instruction Manual.

/¯4_7 Canberra:
Dual Counter/Timer Model 6325, Instruction Manual.

/¯5_7 Canberra:
Device Actuator Model 6624.

/¯6_7 Canberra:
Digitizer Model 6271, Instruction Manual.

/¯7_7 Canberra:
Cassette Loaded Magnetic Tape, Transport System Model 2020.

/¯8_7 Canberra:
Magnetic Tape Controller Model 8531A, Instruction Manual.

/¯9_7 Pertec:
Model 7X40 Synchronous Read after Write Tape Transport, Operating and Service Manual No. 101008.

/¯10_7 Digital:
PDP11, Processor Handbook.

/11/ Kruse, H.:
Personal communication

/12/ Canberra:
CIMSES/2020 Instruction Manual.

/13/ Canberra:
CLASS Instruction Manual.

/14/ Canberra:
Model 6911/CLASS Instruction Manual.

/15/ S. Jacobi, K. Letz, G. Schmitz:
DERAN: Ein Rechenprogramm zur Identifizierung radioaktiver Nuklide in gemessenen Gammaspektren.
KfK 2447, April 1977

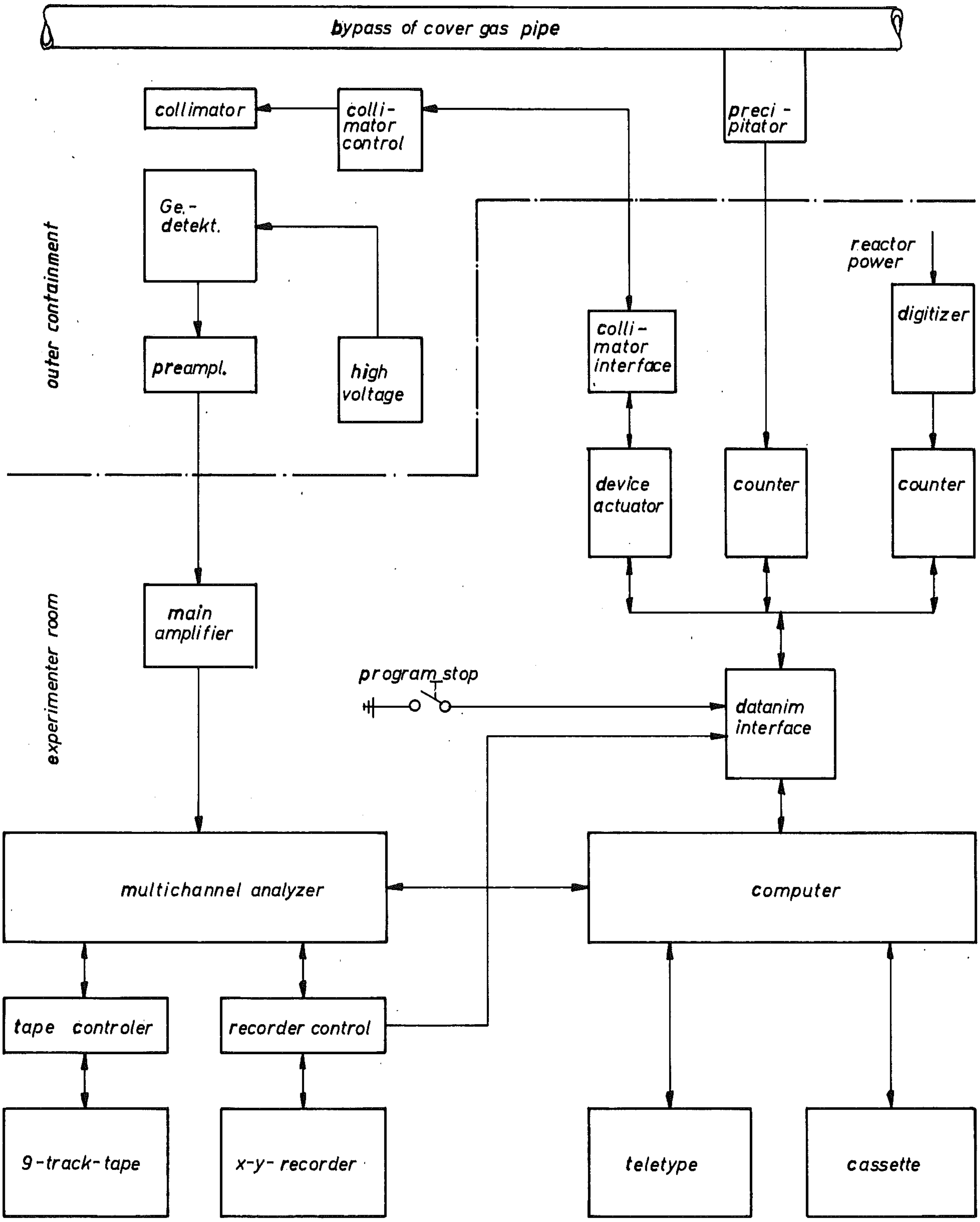


Fig. 1 Measuring Station for Gamma Spectroscopy

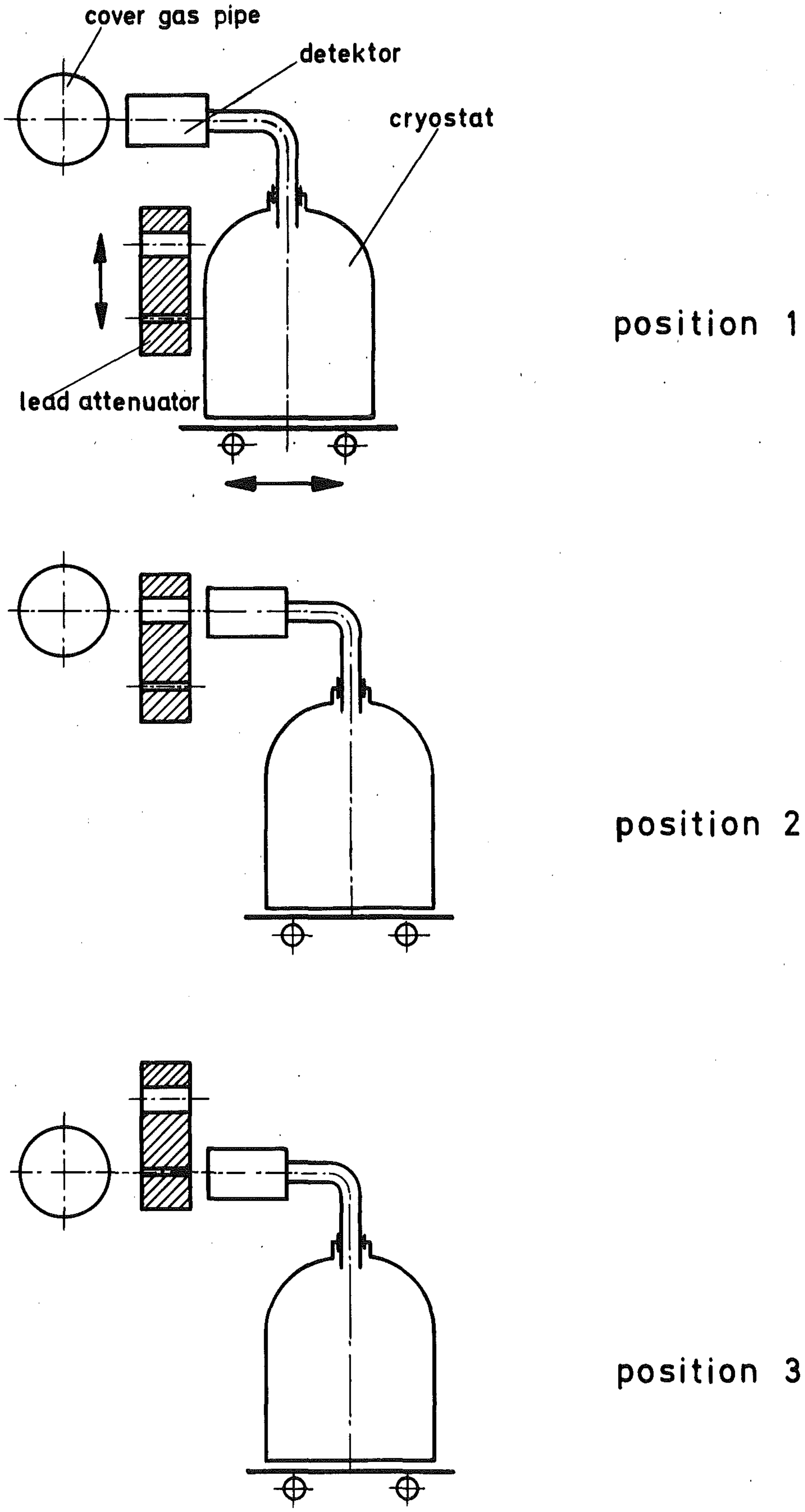
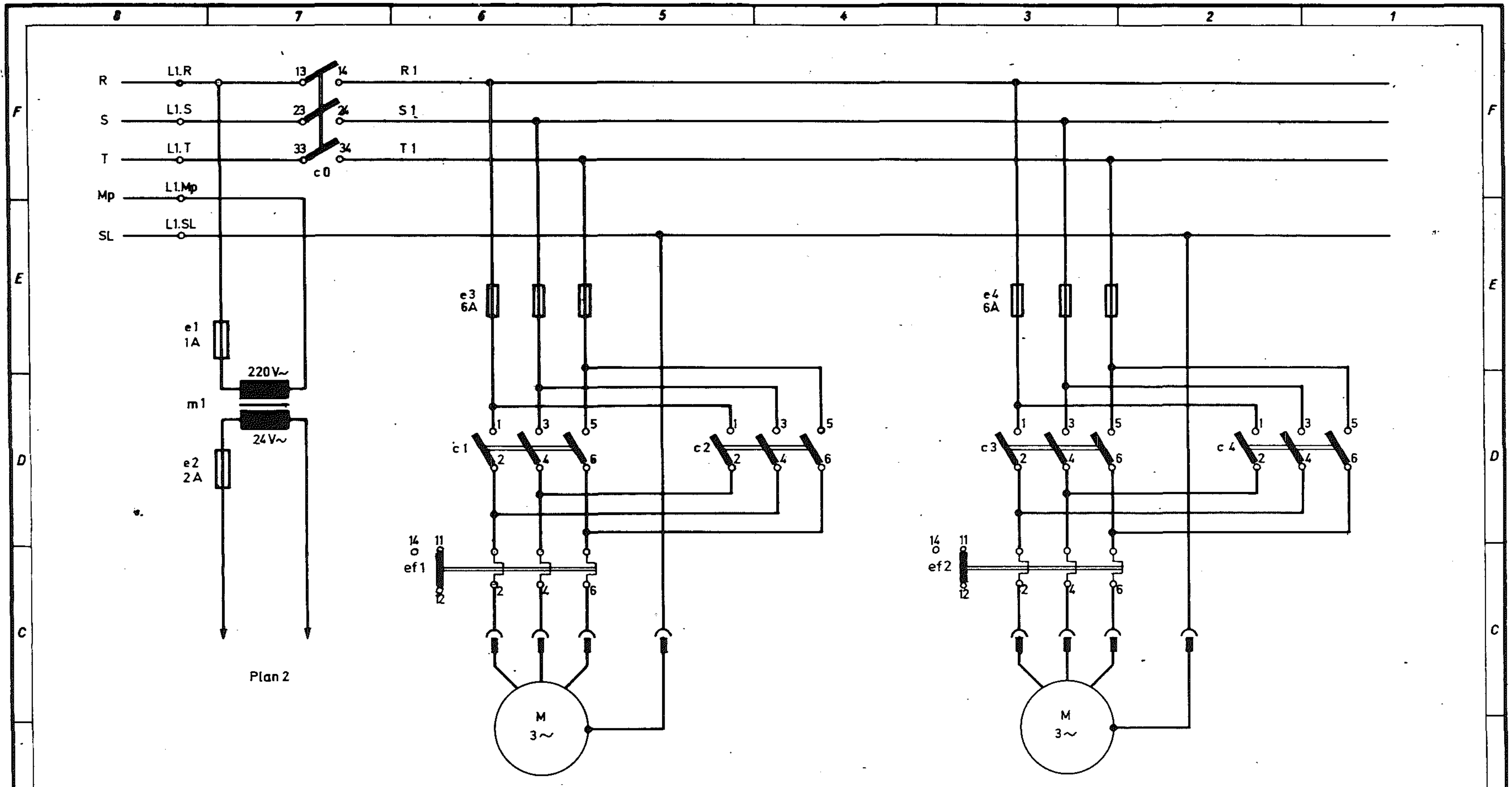


Fig.2 Collimator Positions



elevating motor
for attenuator

380 V ⅄ / 220 V Δ   0,43/0,74 A   50 Hz
120 W   1500 rpm

driving motor
for cryostat

380 V ⅄ / 220 V Δ   0,15/0,26 A   50 Hz
19 W   1200 rpm

Fig. 3

1978	Tag	Name
gez	9.2.	[signature]
gepr	14.2.	[signature]
ges		

Werkstoff

Gesellschaft für Kernforschung m b H
7500 Karlsruhe
Postfach 3640

Zugeh Zchng
Ersatz für
Ersetzt durch

Maßstab

Benennung: Collimator Drive Unit

Zeichnungs-Nr: Plan 1
IRE4-107-3-5416

Buchstabe	kommt vor	Änderung	Tag	Name

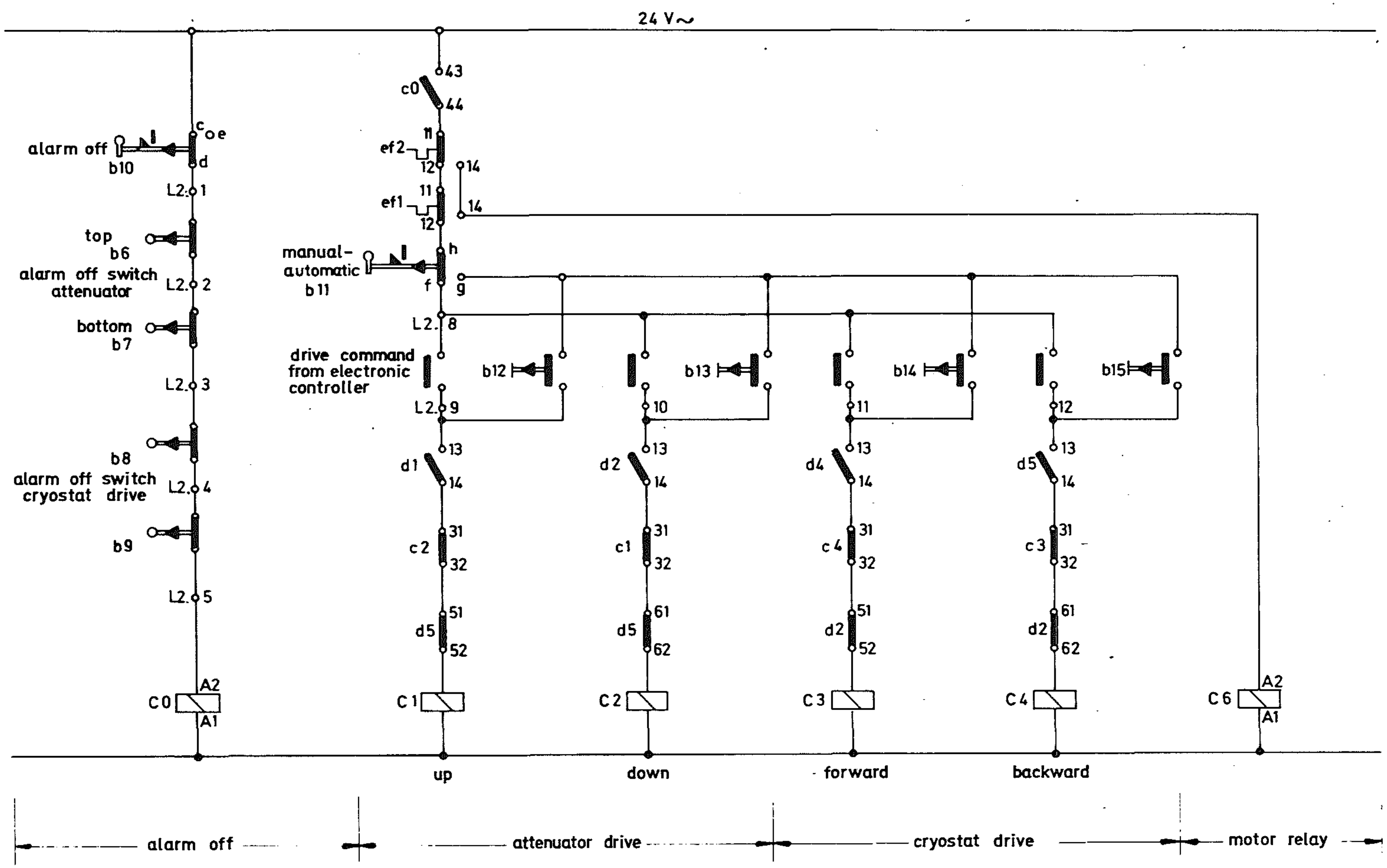


Fig. 4

allowed drive conditions

1. attenuator at bottom-drive cryostat
2. cryostat back-drive attenuator

1978	Tag	Name	Werkstoff	Gesellschaft für Kernforschung m b H 7500 Karlsruhe Postfach 3640	Zugeh. Zchng.
gez	13.2.				Ersatz für
gepr	14.2.				Ersetzt durch
Maßstab	Benennung	Collimator Control			Zeichnungs-Nr Plan 2 IRE 4-107-3-5417

Buchstabe	Anzahl	Änderung	Tag	Name

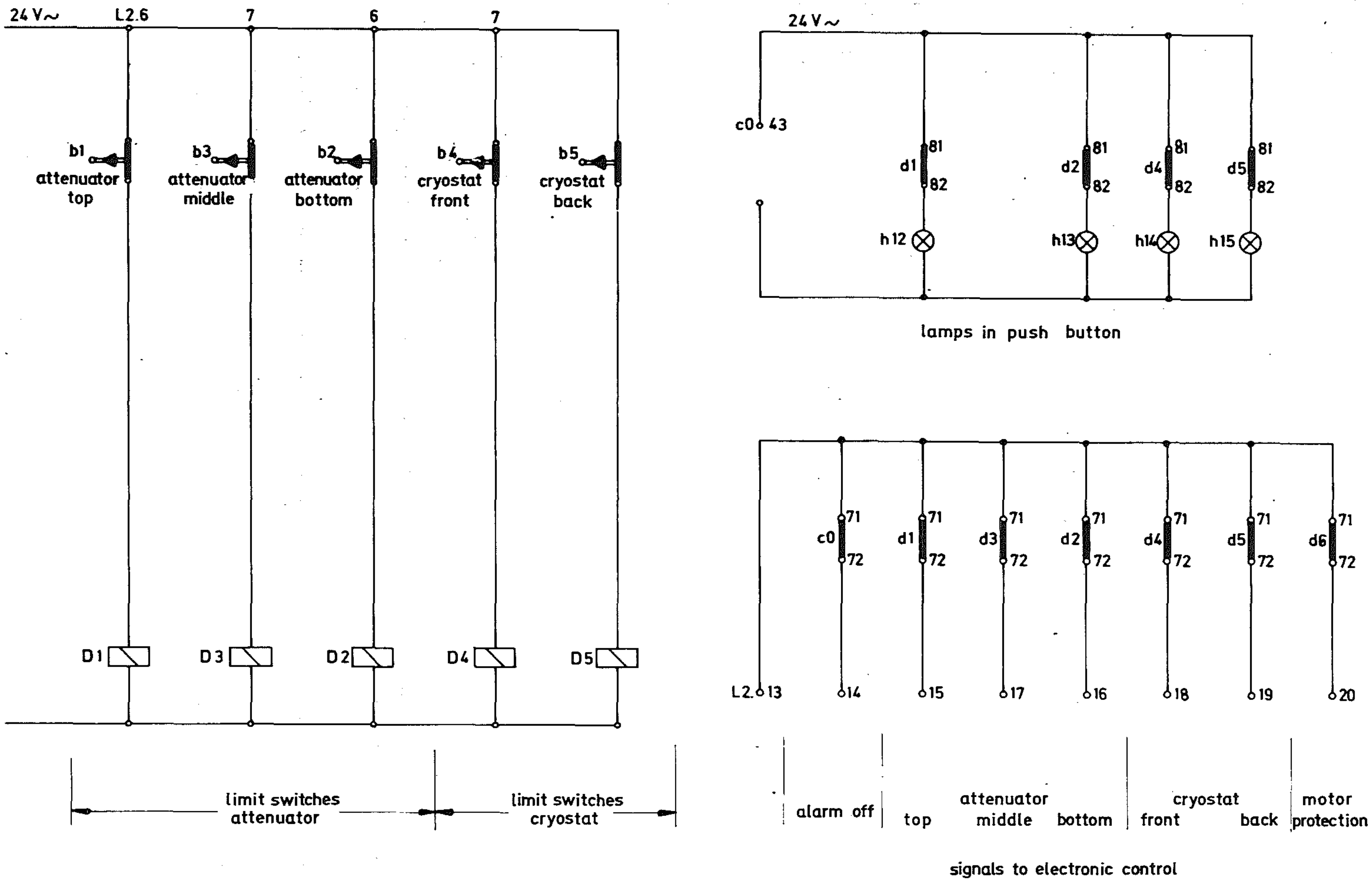


Fig. 5

1978	Tag	Name	Werkstoff	Gesellschaft für Kernforschung m b H 7500 Karlsruhe Postfach 3640	Zugeh. Zchn.
gez	9. 2.	[signature]			Ersatz für
gepr	14. 2.	[signature]			Ersetzt durch
Maßstab	Benennung: Collimator Control				Zeichnungs Nr: Plan 3 IRE 4 - 107 - 3 - 5418

Buchstabe	Stückzahl	Änderung	Tag	Name

30-

L1

	L1	
	R	c0.13
	R	e1
	S	c0.23
	T	c0.33
	Np	m1
	Pc	
c0.14	R1	e3
		e4
c0.24	S1	e3
		e4
c0.34	T1	e3
		e4

main power (R, R, S, T, Np, Pc)

L2

	L2	
b10d	1	b6
b7	2	b6
b7	3	b8
b9	4	b8
b9	5	c0.A2
b1-b2	6	
b3-b4	7	
2;4;6;8	8	b11.f
5	9	d1.13
7	10	d2.13
3	11	d4.13
1	12	d5.13
9	13	c0.53
6;7	14	c0.72
3	15	d1.72
5	16	d2.72
4	17	d3.72
2	18	d4.72
1	19	d5.72
8	20	d6.72

} 24 V (rows 6–7)

Lemo-connector 8 pin (rows 8–12)

Lemo-connector 10 pin (rows 13–20)

d1-d6.71

switch	pin	twisted cable color 25x1 mm²	terminal L1
b1	a1	red/bl	d1.A2
b1	b1	bl	6
b2	a2	red/bl	d2.A2
b2	b2	br	6
b3	a3	br/red	d3.A2
b3	b3	blue	7
b4	a4	blue/red	d4.A2
b4	b4	yel	7
b5	a5	yel/red	d5.A2
b5	b5	gn	7
b6	a6	wt/red	2
b6	b6	vio	1
b7	a7	gn/red	3
b7	b7	or	2
b8	a8	yel/wt	4
b8	b8	pink	3
b9	a9	blue/wt	5
b9	b9	be	4
	a0		
	b0		
	c1		
	c2		
	c3		
	c4		
	c5	yel/gn	
	c6		
	c7		
	c8		
	c9		
	c0		

(c1–c0: yel/gn)

Fig. 6

1978	Tag	Name	Werkstoff	Gesellschaft fur Kernforschung m b H 7500 Karlsruhe Postfach 3640	Zugeh. Zchng
gez	9.2.	Lang			Ersatz fur
gepr	14.2.	Brüggemann			Ersetzt durch
ges					

Maßstab

Benennung: Collimator Terminal Diagram

Zeichnungs-Nr: Plan 4 IRE4-107-3-5419

Buchstabe | kommt vor | Änderung | Tag | Name

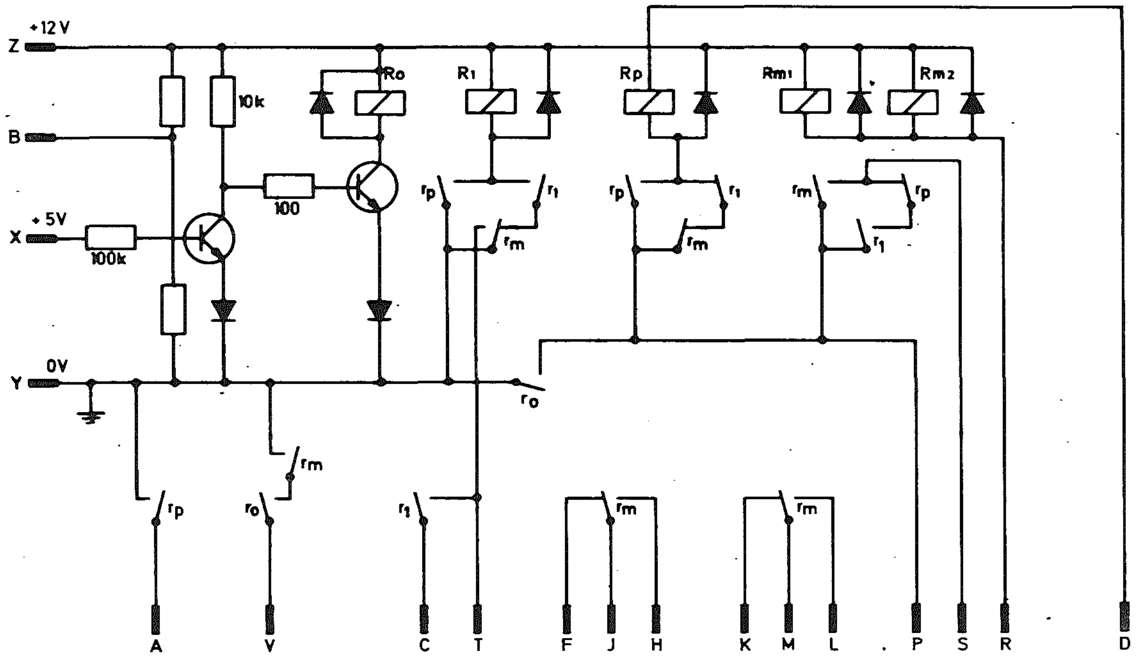


Fig. 7

1978	Tag	Name	Werkstatt	Gesellschaft für Kernforschung m. b. H. 7500 Karlsruhe Postfach 3640	Zugeh. Zchng.
gez	24. 5.	[signature]			Ersatz für
gepr					Ersetzt durch
ges.					

Maßstab

Benennung: X-Y Recorder Control (Plug in Board)

Zeichnungs-Nr.: IRE 5 - V107 - 3 - 1068

Buchstabe	komment vor	Änderung	Tag	Name

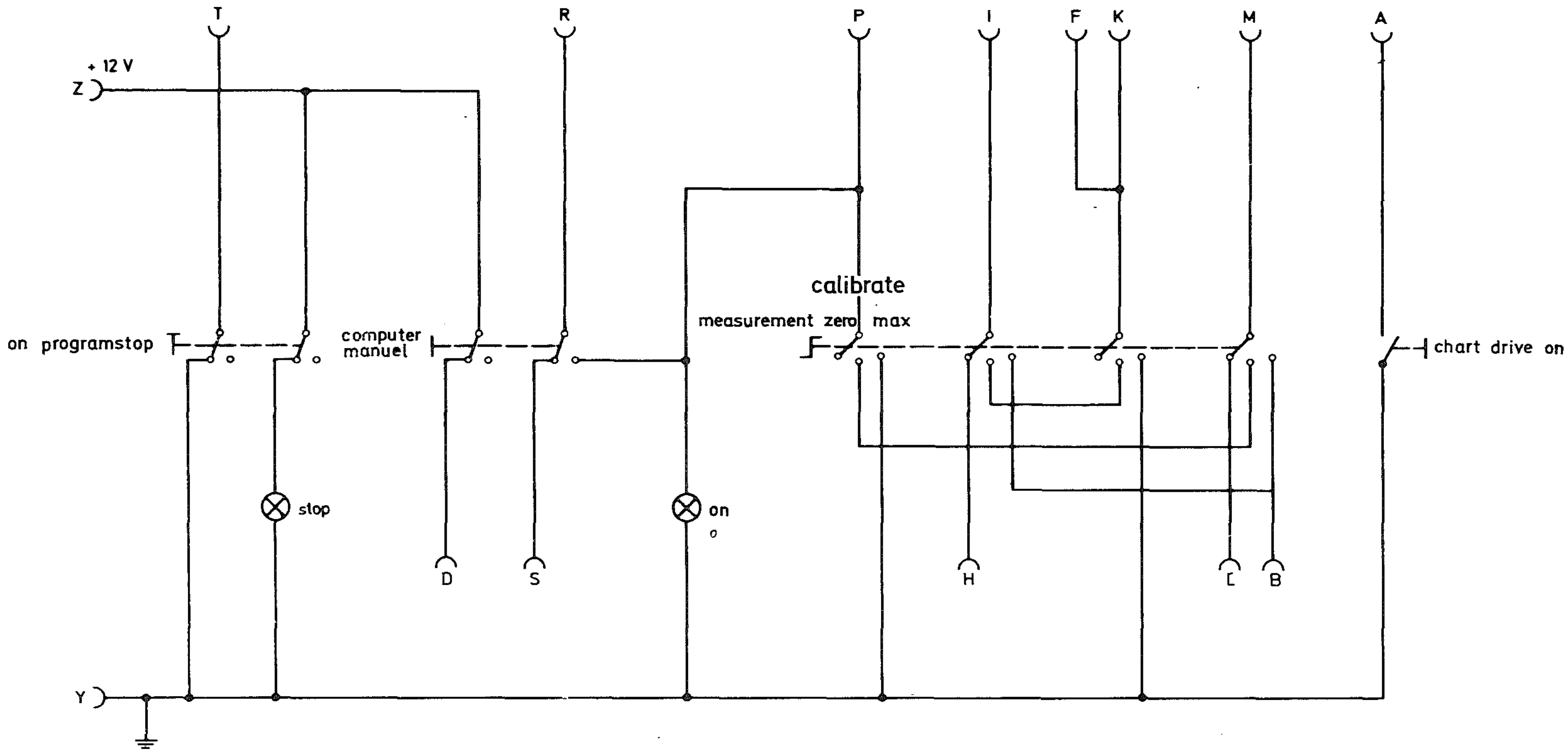


Fig. 8

1978	Tag	Name	Werkstoff	Gesellschaft für Kernforschung m. b. H. 7500 Karlsruhe Postfach 3640	Zugeh. Zchng.
gez	24.5	[signature]			Ersatz für
gepr					Ersetzt durch
gas					
Maßstab	Benennung: Recorder Control (Box)				Zeichnungs-Nr: IRE 5 - V107 - 3 - 1070

Buchstabe	Anzahl der	Änderung	Tag	Name

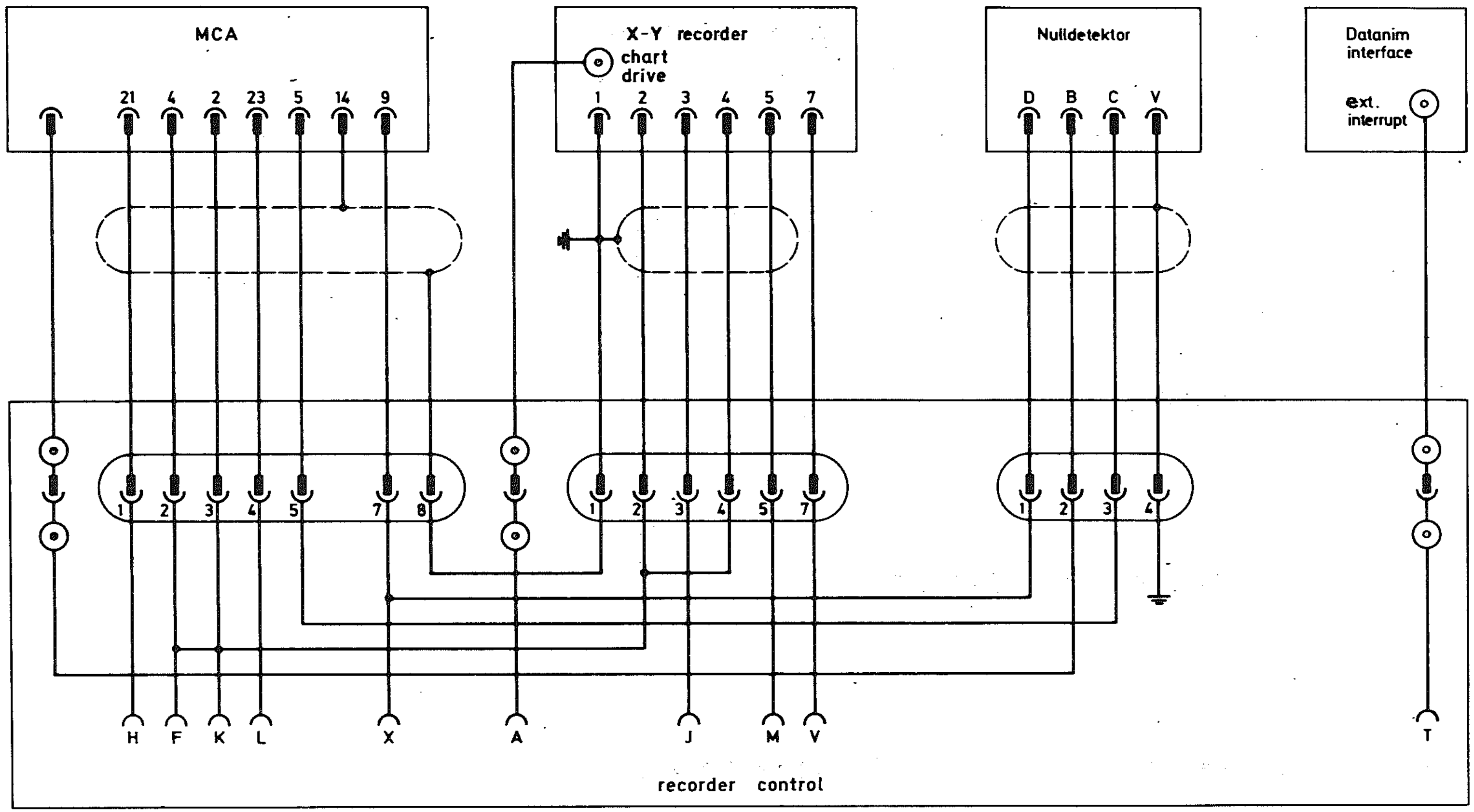


Fig. 9

1978	Tag	Name	Werkstoff	Gesellschaft für Kernforschung m. b. H. 7500 Karlsruhe Postfach 3640	Zugeh. Zchng.
gez	24. 5.	[signature]			Ersatz für
gepr					Ersetzt durch
ges					
Maßstab	Benennung	Cable Arrangement for Recorder Control			Zeichnungs-Nr. IRE 5 - V 107 - 3 - 1069

Buch-stabe	Anzahl vor	Änderung	Tag	Name

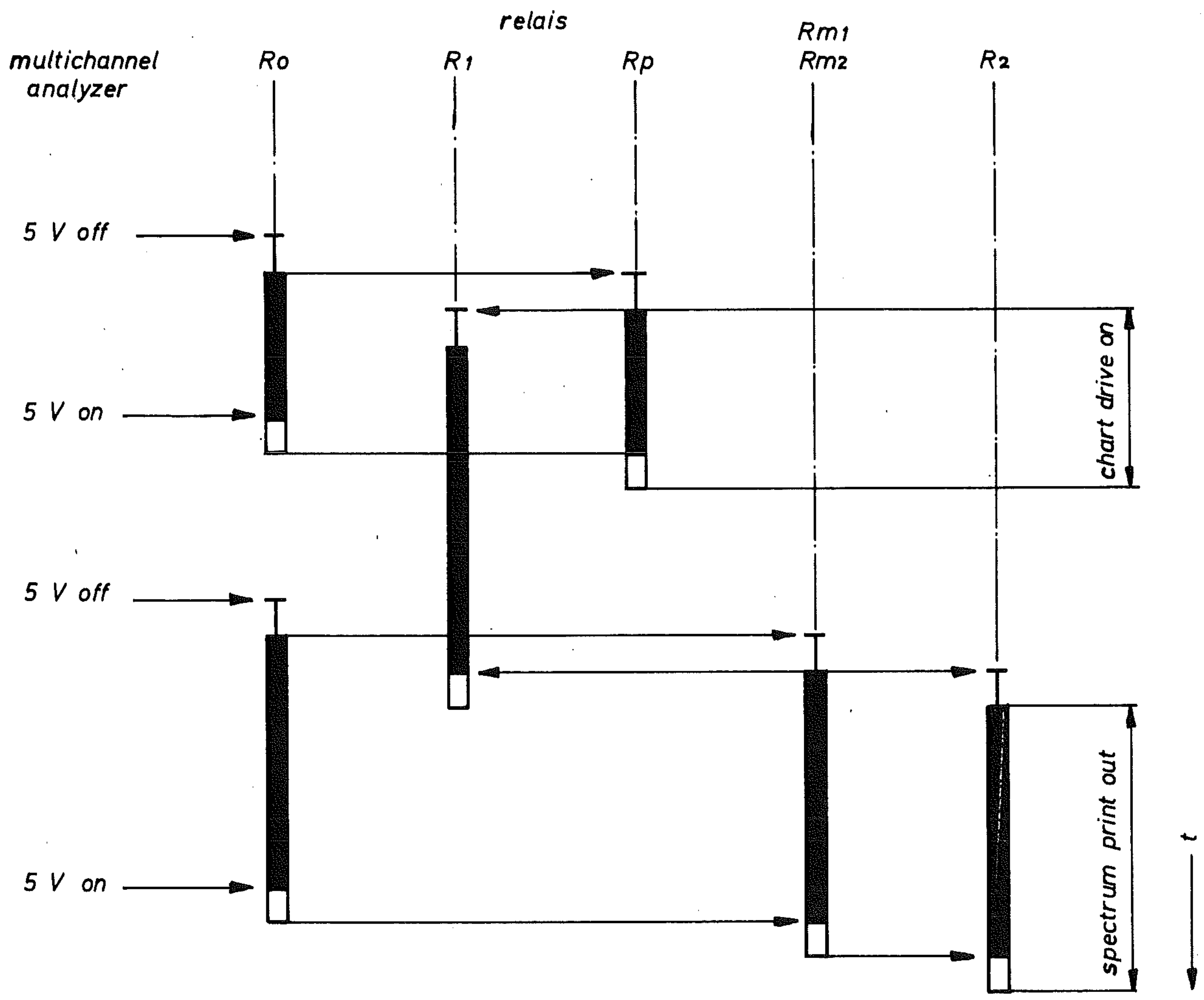


Fig. 10 Recorder Control (timing)

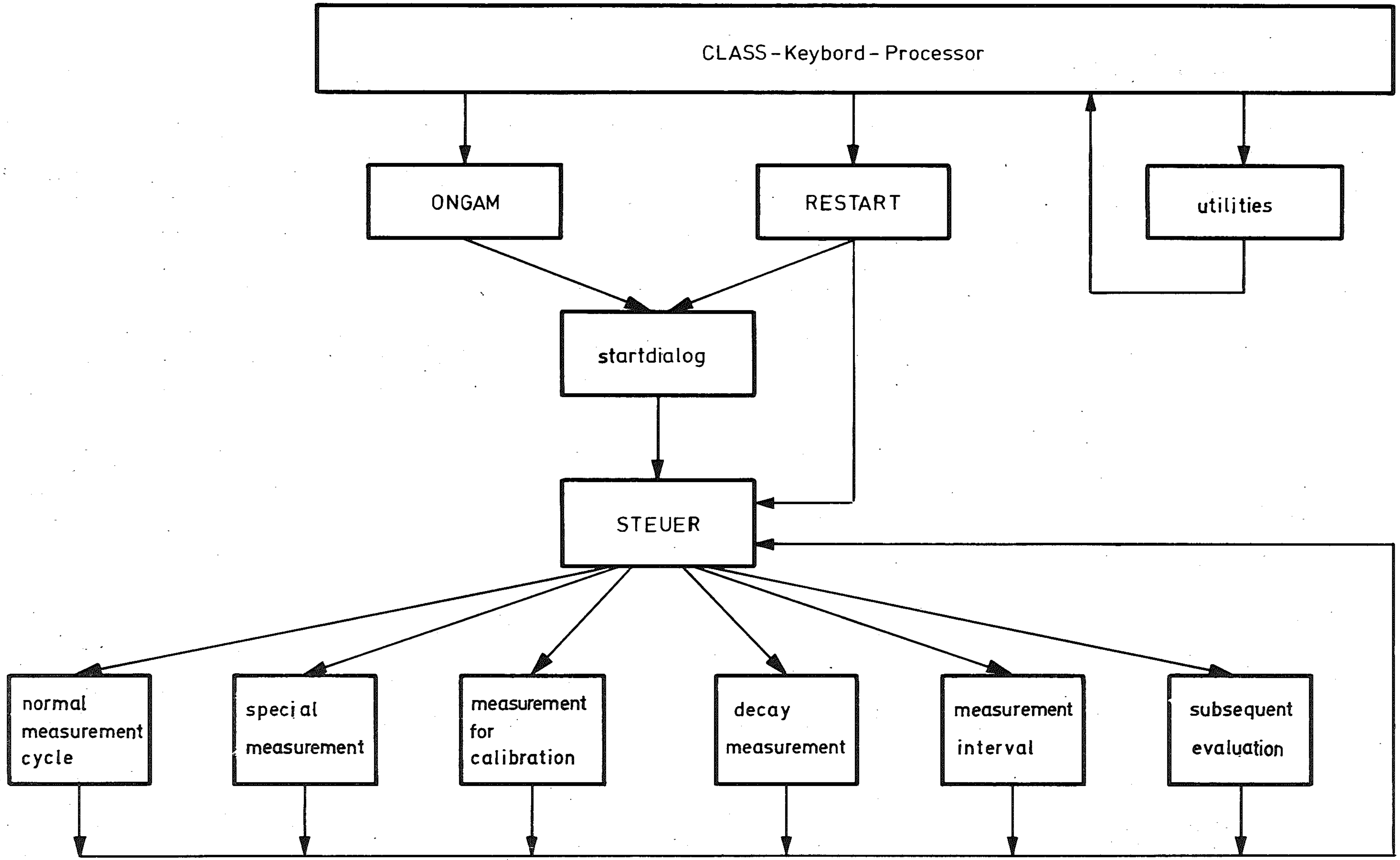


Fig.11 ONGAM Computer Operation

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
livetime setting for MCA	truetime MCA	deadtime MCA	spectrum nr.	calibr. nr. or decay meas. nr.	day	month	year	start moment MCA	calibration factor a	calibration factor b	calibration factor c	quantity of peaks to be evaluated	mark for type of measu-ring	colli. position	nr. of spectrum from which new calibr. fact. work	RL (reac-tor power) average	RL start	RL end	RL max

21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
RL min	prezipi-tator countrate average	prezip. countrate start	prezip. countrate end	prezip. countrate max	prezip. countrate min											mark for reactor power off=1	measuring time 1 MCA	measuring time 2 MCA	measuring time 3 MCA

41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60
new calibration if Q(41)= Q(46)	starting time MCA SAVE $TIME	start MCA half-nr.	alarm-mark controls SONDER ABKLI	mark for colli.	calibration sequence (day)	end of measuring pause $TIME	counter for decay measure-ment	I/0-se-quence while meas. pause	prezip. actuel	RL actuel	measurem. sequence (sec)	calling counter for RLCPS	acquisition time for calibration	counter for longtim. compari-sion file nr.	1. limit value for präzip.	2. limit value for präzip.	recover calibration protocol	run index	alarm-mark for calibration factor a

61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
limit of RL for decay measure	run index SONDER	end of acquisition $TIME	evaluation factor for RL	normal file counter	1 geome-tric factor for colli.	2 geome-tric factor for colli.	3 geome-tric factor for colli.	recover decay-measure. protocol		MCA half to be read out	event-time $TIME	save ROI1(40) in ALBEA	run-index ALBEA	decay measure. evaluation counter	acquisition time for decay-measure.	time for pause ABKLI	run index NORMES	sequence for storing on long-time file (days)	acquisition time drop because reading MCA

81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
no calibr. Interrupt	max. drift of channels	allowed difference RL %	allowed difference prezip. %	standart value RL	standart value prezip.	day coun-ter for longtime file	scram moment $TIME	alarmfile counter	starting time MCA 2. half						aktuel geometric-factor	detector volum	alarmfile evaluation counter	tape name	< ∅ don't inc. date (DATKOR)

Fig. 12 Data Organization Array Q (100)

Header

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
livetime MCA	truetime MCA	deadtime MCA	spectrum r.	calibration or decay-ms. r.	day	month	year	start moment of MCA $TIME	calibration factor a	calibration factor b	calibration factor c	quantity of peaks to be evaluated	mark for type of measurement	colli-position

16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
un index	RL average	RL start	RL end	RL max	RL min	prezipitator countrate average	prezip. countrate start	prezip. countrate end	prezip. countrate max	prezip. countrate min				

31	32	33	34	35	36	37	38	39	40	41	42	43	44	45
						contents of peaks Σ	activity concentration Σ		Q (65) while evaluation of alarm file					

windows

	X+1	X+2	X+3	X+4	X+5	X+6	X+7	X+8	X+9	X+10	X+11	X+12	X+13	X+14	X+15	X+16	X+17
1.	46 channel addresse	47 window width (channels)	48 exact channel position	49 energy computed (KeV)	50 $\varepsilon(E)$	51 back ground	52 contents of peak	53 quantity of channels belonging to the peak	54 FWHM	55 rel Error %	56 activity concentration	57 geometric factor $* \varepsilon(E)$	58 difference of activity concentration	59 nuclid name	60 energy (KeV)	61 half life periode	62 γ/d %

values from table-book (X+14 – X+17)

	X+1	X+2	…	X+17
2.	63	64	…	300

X+2 (window width) <0 ≙ peak not used for drift control

X+17 (abundance) <0 ≙ peak not added to peak sum

Fig. 13 Data Organization Arrays ROI 1 (300) and ROI 2 (300)

1 livetime MCA	2 truetime MCA	3 deadtime MCA	4 spectrum nr.	5 calibration nr.	6 day	7 month	8 year	9 start moment of MCA $TIME	10 calibration factor a (new)	11 calibration factor b (new)	12 calibration factor c (new)	13 quantity of peaks to be evaluated	14 mark for type of measurement	15 colli.-position
16 run index	17 RL average	18 RL start	19 RL end	20 RL max	21 RL min	22 prezipitator countrate average	23 prezip. countrate start	24 prezip. countrate end	25 prezip. countrate max	26 prezip. countrate min	27	28	29	30
31	32	33	34	35	36	37	38	39	40 Q (65) while evaluation of alarmfile	41	42	43 calibration factor a (old)	44 calibration factor b (old)	45 calibration factor c (old)

calibration peaks

1.	46 start channel	47 stop channel	48 quantity of channels	49 contents of channel	contents of channel	…	94 contents of channel	95 exact channel position	96 energy	
2.	97 start channel	98 stop channel	99 quantity of channels	100 contents of channel	contents of channel	…	145 contents of channel	146 exact channel position	147 energy	
3.	148 start channel	149 stop channel	150 quantity of channels	151 contents of channel	contents of channel	…	196 contents of channel	197 exact channel position	198 energy	199 / 200

Fig. 14 Data Organization Array EIF(200)

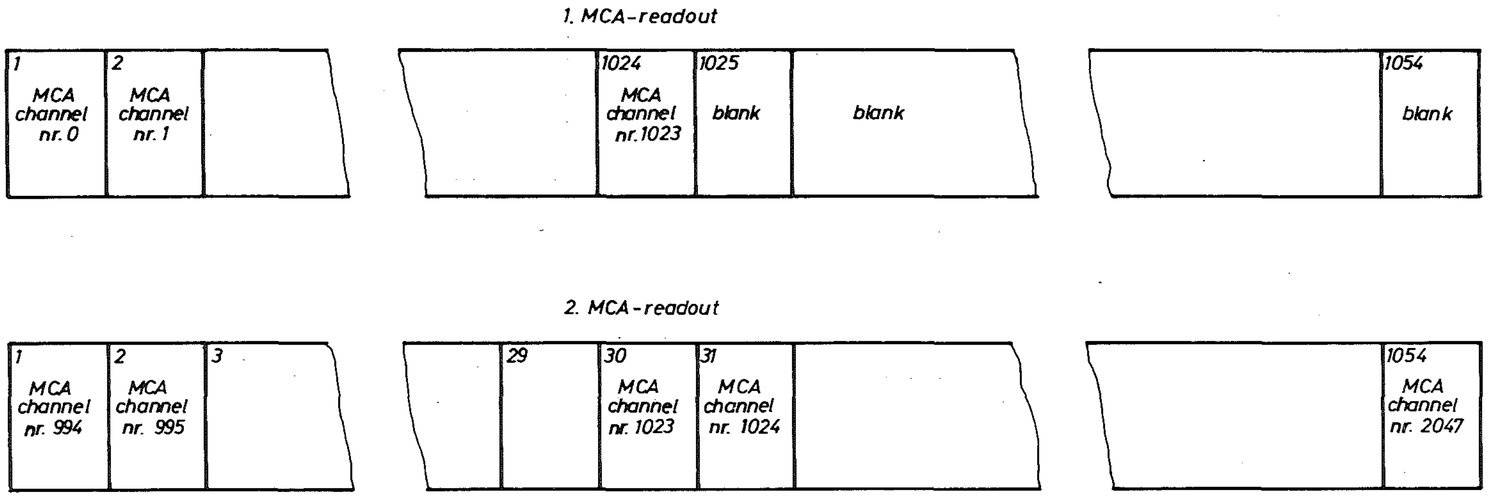


Fig. 15 Data Organization Array SPEC (1054)

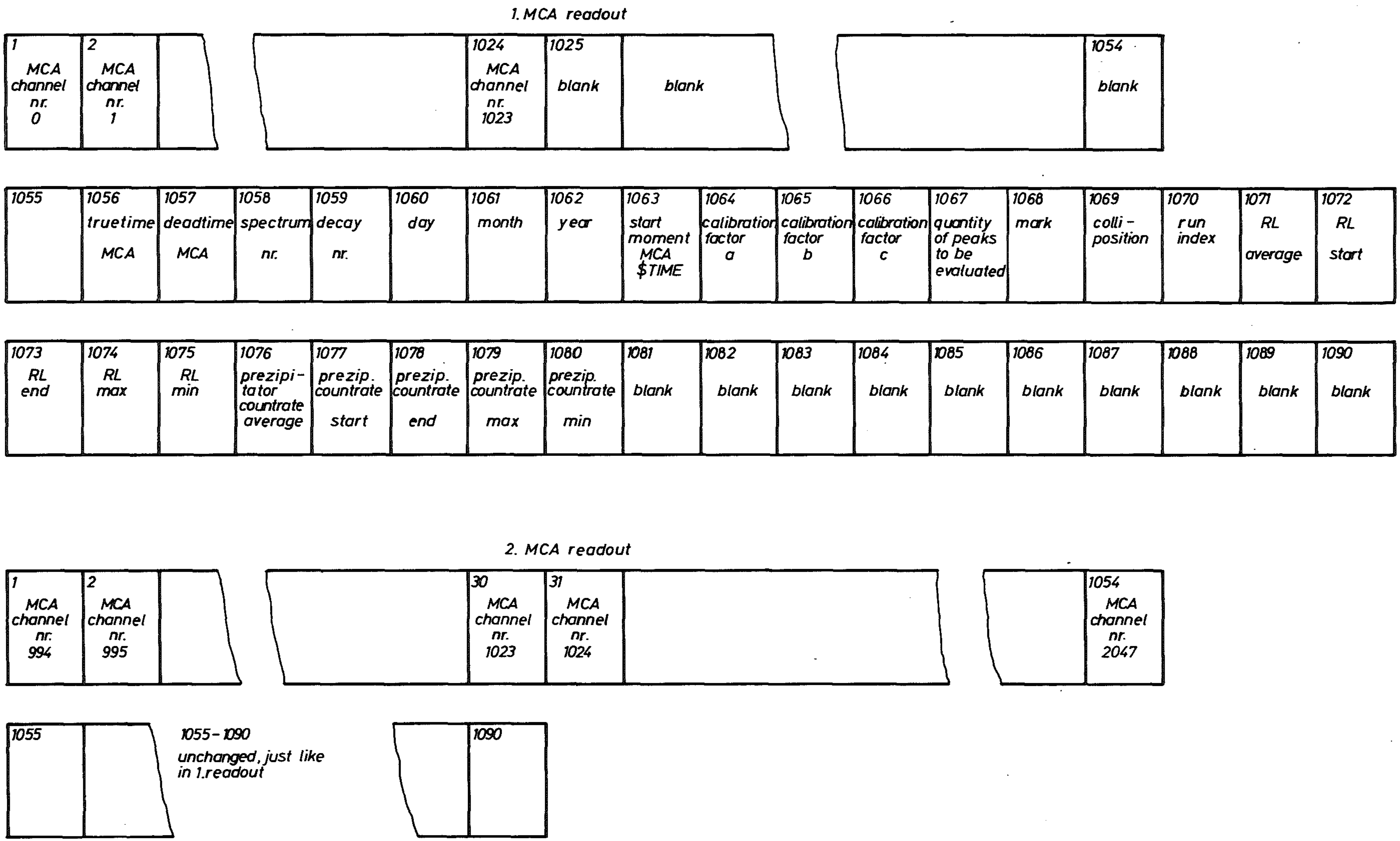


Fig. 16 Data Organization Array SPEK (1090)

tagword (format 16)	1 * livetime MCA	2 truetime MCA	3 deadtime MCA	4 spectrum nr.	5 calibration or decay nr.	6 day	7 month	8 year	9 start moment MCA	10 calibration factor $\lvert a\rvert \times 10^8$	11 calibration factor $\lvert b\rvert \times 10^4$	12 calibration factor $\lvert c\rvert \times 10^4$	13 quantity of peaks to be evaluated	14 mark for type of measurement	15 colli-position

16 RL average (kW)	17 prezipitator count average (cps)	18 =1 if a<0	19 =1 if c<0	20 contents of MCA channel	21 contents of MCA channel	1023	1024	tagword (format 16)	1025	1026 2047	2048 contents of MCA channel	

Nr. 14 (Mark)

= 1 normal measurement
= 2 calibration measurement
= 3 special measurement
= 4 ——
= 5 decay measurement
= 6 reactor off
= 7 sequence of special measurements

Nr. 5

number of decay measurements if mark (14) = 5
if not calibration measurement Nr.

Nr. 9 (start moment)

HHMMSS
HH = hour
MM = minute
SS = second

Nr. 15 (collimator position)

= 1,2 or 3 collimator exactly positioned
= 10,20 or 30 collimator defect in this case the exact colli-position must be 1,2 or 3 where 10≙1; 20≙2; 30≙3

* tape channel number n ≙ MCA channel number n-1

Fig. 17 Data Organization for 9-Track-Tape

SPEKTRUMS NR. = 688

EICHFAKTOREN:
A = .00000310 B = .99440770 C = 1.0819826

STARTZEIT = 16.00 UHR 7.9 S

TRUETIME = 486.79999 S
LIVETIME = 400.00000 S
DEADTIME = 86.799988 S

WAEHREND DER AUFNAHME DES SPEKTRUMS WURDEN FOLGENDE WERTE GEMESSEN

	MITTEL	START	ENDE	MAXIMAL	MINIMAL	
REAKTORLEISTUNG	58.30	58.39	58.40	58.85	57.94	MW
FRAEZIP-ZAEHLR.	9607	9520	9664	9766	9424	IMP/S

SUMME PEAK NR. 4;

4522 IMP 11.305000 IMP/S .88122018E 2 1/S*CCM

AUSGEWERTETE FENSTER

NR.	GENAUE KANAL-LAGE	ENERGIE (KEV)	UNTER-GRUND (IMP.)	SIGNAL (IMP.)	KZ.	HWB (KEV)	REL. FEHLER (%)
1							
2							
3							
4	251.03	250.90	25555	4522	6	3.97	23.20
5							
6							
7							
8							
9							
10							
11	1295.55	1294.60	24150	422084	37	7.95	0.75
12							
13							
14							
15							

FORTS.AUSW.FENSTER — TABELLENWERTE

NR.	AKTIV.KONZ. (1/S*CCM).	NUCLID	ENERGIE (KEV)	HWZ	ABUND. (%)
1		KR 85M	150.990	4.40 H	74.00
2		KR 88	196.100	2.80 H	37.80
3		XE133M	232.900	2.26 D	14.00
4	.88122018E 2	XE 135	249.800	9.14 H	91.00
5		XE 138	258.300	14.1 M	31.00
6		KR 87	402.800	76.0 M	53.00
7		NE 23	440.000	38.2 S	33.00
8		CS 138	462.800	32.2 M	27.80
9		XE135M	526.800	15.6 M	80.00
10		RB 89	1030.700	15.1 M	60.00
11	.28830076E 5	AR 41	1293.600	1.84 H	99.00
12		CS 138	1435.900	32.2 M	75.00
13		KR 89	1472.100	191 S	9.50
14		KR 88	1529.800	2.80 H	11.30
15		RB 88	1836.130	17.8 M	30.20

Fig. 18: Spectrum of the Background

SPEKTRUMS NR. = 689

EICHFAKTOREN:
A = .00000310   B = .99440770   C = 1.0819826

STARTZEIT = 16.56 UHR 45.9 S

TRUETIME = 543.09998 S
LIVETIME = 400.00000 S
DEADTIME = 143.09998 S

WAEHREND DER AUFNAHME DES SPEKTRUMS WURDEN FOLGENDE WERTE GEMESSEN

	MITTEL	START	ENDE	MAXIMAL	MINIMAL	
REAKTORLEISTUNG	58.23	58.06	58.17	58.57	57.77	MW
PRAEZIP-ZAEHLR.	14026	10468	32674	32674	10468	IMP/S

SUMME PEAK NR. 1; 2; 3; 4; 6; 7; 8; 9;12;

673071 IMP   1682.6799 IMP/S   .16307604E 5 1/S*CCM

AUSGEWERTETE FENSTER

NR.	GENAUE KANAL-LAGE	ENERGIE (KEV)	UNTER-GRUND (IMP.)	SIGNAL (IMP.)	KZ.	HWB (KEV)	REL. FEHLER (%)
1	152.43	152.73	223006	49581	16	7.95	9.23
2	197.75	197.85	165642	20395	14	6.96	18.05
3	233.90	233.85	38745	2721	5	4.97	40.78
4	250.91	250.79	362436	511253	33	8.95	1.67
5							
6	404.55	403.88	101538	32065	20	7.95	10.71
7	435.72	434.95	55357	8982	13	5.96	23.02
8	458.80	457.97	18129	1892	6	4.97	44.26
9	528.76	527.76	73831	43933	19	7.95	6.65
10							
11	1295.05	1294.11	42840	407332	37	8.95	0.92
12	1436.13	1435.60	5978	2248	13	5.96	30.87
13							
14							
15							

FORTS.AUSW.FENSTER		TABELLENWERTE			
NR.	AKTIV.KONZ. (1/S*CCM)	NUCLID	ENERGIE (KEV)	HWZ	ABUND. (%)
1	.89016409E 3	KR 85M	150.990	4.40 H	74.00
2	.82607413E 3	KR 88	196.100	2.80 H	37.80
3	.32984346E 3	XE133M	232.900	2.26 D	14.00
4	.99630009E 4	XE 135	249.800	9.14 H	91.00
5		XE 138	258.300	14.1 M	31.00
6	.14952176E 4	KR 87	402.800	76.0 M	53.00
7	.71938542E 3	NE 23	440.000	38.2 S	33.00
8	.18706245E 3	CS 138	462.800	32.2 M	27.80
9	.16725064E 4	XE135M	526.800	15.6 M	80.00
10		RB 89	1030.700	15.1 M	60.00
11	.27822454E 5	AR 41	1293.600	1.84 H	99.00
12	.22434921E 3	CS 138	1435.900	32.2 M	75.00
13		KR 89	1472.100	191 S	9.50
14		KR 88	1529.800	2.80 H	11.30
15		RB 88	1836.130	17.8 M	30.20

Fig. 19: Spectrum after the Occurrence of the Cladding Failure

spectrum
calibration factors:

startingtime = o'clock

truetime
lifetime
deadtime

Whilst sampling of the spectrum the following Values are measured

	mean	start	limit	maximal	minimal	
reactor power						MW
integral count rate[1]						cps

peak No. ...
... counts ... cps

results of the measurements

No.	precise channel	energy (keV)	back-ground (counts)	signal (counts)	KZ[2]	FWHM (keV)	rel. error (%)

cont. results Values from the literature

No.	activity concentration (1/s* cm³)	nuclid	energy (keV)	half-live	abundance (%)

[1] see footnote on page 21

[2] number of Channels belonging to the peak

Fig. 20: Translation for Fig. 18 and Fig. 19

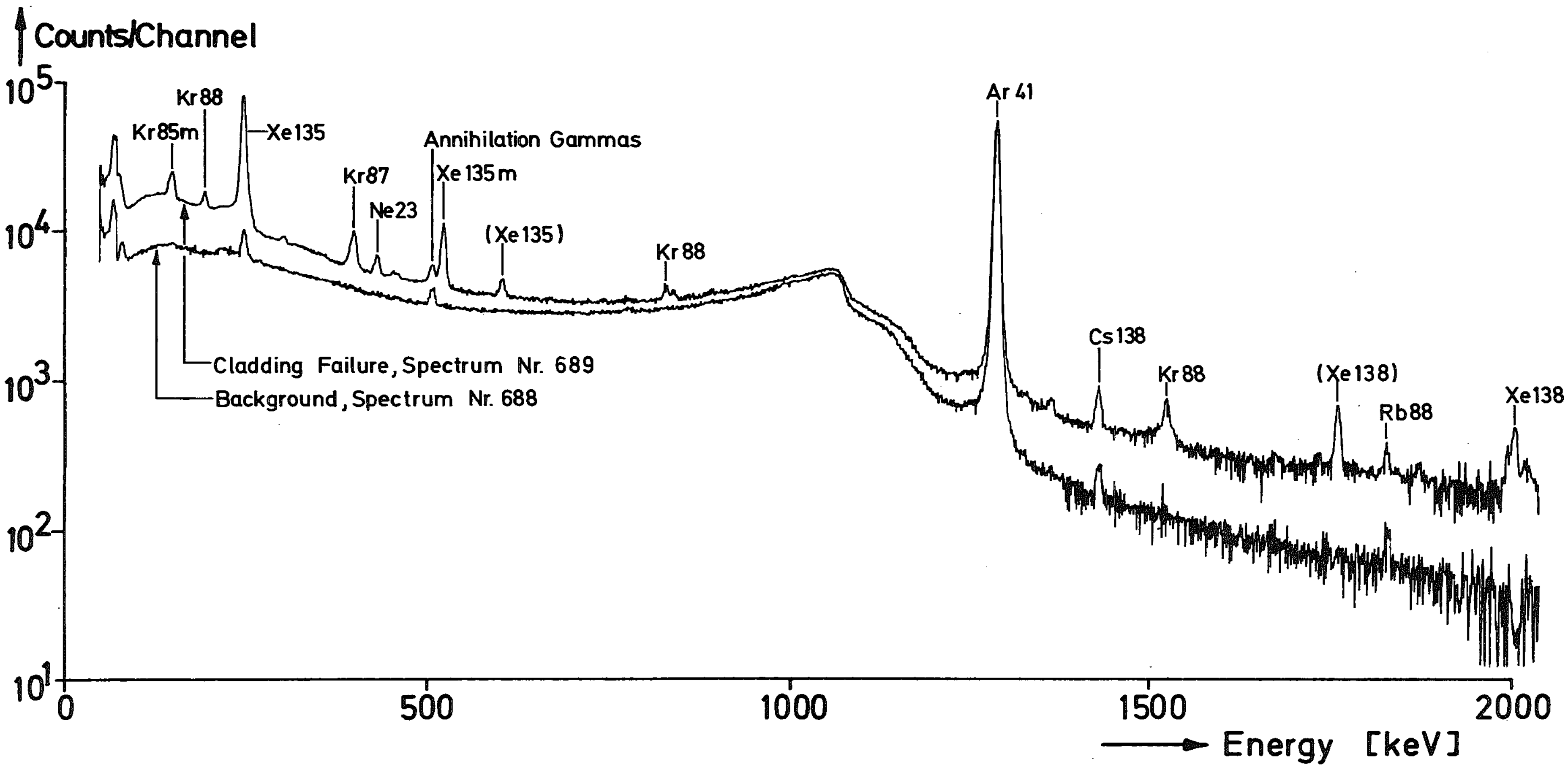


Fig. 21 Gammaspectra of the Background and at the Beginning of the Cladding Failure

A n n e x

ONGAM

CALLS START DIALOG PROGRAMS AND IS USED IF CLASS HAS BEEN STARTED VIA CIMSES KEYBOARD MONITOR.

CALL: ONGAM

```
DEFINE ONGAM<;          REMARK  7.9.77  &
       LOCAL FRAG,K,M,L,ONG,AEN
       AEN=Ø;IF(NE(DEF(R),1)) R=Ø
       IF(EQ(R,Ø)) TYPE!,"ONGAM VERS. Ø1.77 / 16K",!,!
       $U(3);RUN ONGA1;DELETE ONGA1;$U(3);RUN ONGA2;DELETE ONGA2
       IF(EQ(R,1))GOTO ONG1
       ERASE R;STEUER(Ø)
ONG1:  ONG=Ø>
```

DATEIN

DIALOG PROGRAM FOR DATE AND TIME INPUT.

CALL: $U(3);RUN DATEIN

```
DEFINE DATEIN<; REMARK  21.1Ø.76        &
       LOCAL STUNDE,MINUTE
       TYPE!,"DATUM EINGEBEN"
       TYPE!,"TAG ?"
       READ(1) Q(6)
       TYPE"MONAT ? (IN ZIFFERN EINGEBEN)  "
       READ(1) Q(7)
       TYPE"JAHR ?"
       READ(1) Q(8)
       TYPE"UHRZEIT EINGEBEN. STUNDE ?"
       READ(1) STUNDE
       TYPE"NACH EINGABE DER MINUTEN, AUF VOLLE MINUTE"
       TYPE!,"MIT RETURN-TASTE STARTEN!"
       TYPE!,"MINUTE ?"
       READ(1) MINUTE
       $STIME(STUNDE,MINUTE,ØØ)
       DELETE DATEIN>
```

ONGA1

DIALOG PROGRAM FOR OUTPUT AND MODIFICATION OF PROGRAM CONTROLLING VARIABLES.

CALL: $U(3);RUN ONGA1

```
DEFINE ONGA1<;  REMARK  29.9.77  &
        LOCAL ONG1,DET
        TYPE!,"UMRECHNUNGSFAKTOR FUER REAKTORLEISTUNG = ",Q(64)
        TYPE" RICHTIG ? ";READS(1)FRAG;IF(EQ(FRAG,"JA"))GOTO ON01
        AEN=1;TYPE"FAKTOR ? ";READ(1)Q(64)
ON01:   Q(37)=0;TYPE!,"NULLEISTUNGSBETRIEB ?  ";READS(1)FRAG
        IF(EQ(FRAG,"JA")) Q(37)=1
        DET=0;IF(EQ(R,1))GOTO ON05
        Q(100)=0;DATKOR;DATAUS;TYPE!,"DATUM UND UHRZEIT RICHTIG ?  "
        Q(81)=0;READS(1)FRAG;IF(NE(FRAG,"JA")) $U(3);RUN DATEIN;AEN=1
ON05:   TYPE!,"IN WELCHER STELLUNG BEFINDET SICH DER KOLLIMATOR ?  "
        READ(1)Q(15);IF(EQ(R,0))GOTO ON04
        TYPE!,"SOLLEN AENDERUNGEN BEZUEGLICH DER EICHLINIEN BZW."
        TYPE" DER ZU",!,"UEBERWACHENDEN PEAKS VORGENOMMEN WERDEN ? "
        READS(1)FRAG;IF(NE(FRAG,"JA"))GOTO ON03
ON04:   DIMENS EIF(200);EIORG(2)
ON0:    IF(EQ(R,1)) $U(3);RUN SETZ;DELETE SETZ;GOTO ON08
        TYPE!,"FOLGENDE ENERGIEN WERDEN ZUR QUADRATISCHEN"
        TYPE" EICHUNG VERWENDET";K=1;M=96;ONG1=0
        DO(3)<$I(2);TYPE!,"PEAK",K,"  ";$F(10);TYPE EIF(M),"  KEV"
        IF(EQ(EIF(M),0)) ONG1=1
        K=K+1;M=M+51>
        IF(GT(ABS(Q(10)),0.0001)+EQ(Q(11),0)) Q(81)=1
        TYPE!,!,"FOLGENDE PEAKS WERDEN UEBERWACHT.",!;K=0
        IF(GT(Q(13),0))GOTO ON06
        TYPE!,"A C H T U N G",!,"KEINE FENSTER GESETZT",!
ON06:   DO(Q(13))<TYPES ROI1(59+K);TYPE"  ",ROI1(60+K),"  KEV",!;K=K+17>
        IF(EQ(ONG1,1)) $U(3);RUN SETZ;DELETE SETZ;GOTO ON08
ON02:   TYPE!,"SOLLEN AENDERUNGEN BEZUEGLICH DER EICHLINIEN BZW."
        TYPE" DER ZU",!,"UEBERWACHENDEN PEAKS VORGENOMMEN WERDEN ? "
        READS(1) FRAG;IF(EQ(FRAG,"JA")) $U(3);RUN SETZ;DELETE SETZ
ON08:   IF(EQ(DET,1)) $U(3);RUN DETKOF;DELETE DETKOF
ON03:   ONG=0>
```

ONGA2

DIALOG PROGRAM FOR OUTPUT AND MODIFICATION OF PROGRAM CONTROLLING VARIABLES.

CALL: $U(3);RUN ONGA2

```
DEFINE ONGA2<;  REMARK  29.9.77  &
DEFINE DIA<$I(3);TYPE!,"NR.",K,". ";$F(10)>
        IF(EQ(DEF(EIF),1)) ERASE EIF
        TYPE!,"SOLLEN DIE PROGRAMMSTEUERDATEN GELISTET WERDEN ? "
        READS(1)FRAG;IF(NE(FRAG,"JA"))GOTO ON5
        TYPE!,"PROGRAMM STARTET MIT FOLGENDEN WERTEN:",!
        K=10;DIA;TYPE"EICHFAKTOR A=",Q(K);K=11
        DIA;TYPE"EICHFAKTOR B=",Q(K);K=12
        DIA;TYPE"EICHFAKTOR C=",Q(K);K=82
        DIA;$I(3);TYPE"BEI EINER DRIFT DES SPEKTRUMS UM MEHR ALS"
        TYPE Q(K)," KANAELE",!,"         WIRD EINE EICHMESSUNG"
        TYPE" DURCHGEFUEHRT";K=46
        DIA;TYPE"SPAETESTENS ALLE",$I(3)+Q(K),"  TAGE WIRD EINE EICH"
        TYPE"MESSUNG DURCHGEFUEHRT";K=79
        DIA;$I(3);TYPE"SPAETESTENS ALLE",Q(K)," TAGE WIRD EIN SPEKTRUM"
        TYPE!,"          AUF LANGZEITFILE GELEGT";K=54
        DIA;TYPE"EICHZEIT=",Q(K)," S";K=38
        DIA;TYPE"MESSZEIT1=",Q(K)," S";K=39
        DIA;TYPE"MESSZEIT2=",Q(K)," S";K=40
        DIA;TYPE"MESSZEIT3=",Q(K)," S";K=52
        DIA;TYPE"MESSFOLGE=",Q(K)," S";K=49
        DIA;TYPE"AUSGABEFOLGE WAEHREND MESSPAUSE=",Q(K)," S";K=56
        DIA;TYPE"GW1 FUER KOLLIMATOR-STEUERUNG=",Q(K)," COUNTS/S";K=57
        DIA;TYPE"GW2 FUER KOLLIMATOR-STEUERUNG=",Q(K)," COUNTS/S";K=61
        DIA;TYPE"GW FUER ABKLINGMESSUNG=",Q(K)*Q(64)," MW";K=66
        TYPE!,!,"GEOMETRIEFAKTOREN FUER DIE KOLLIMATOR-STELLUNGEN",!
        DIA;TYPE"STELLUNG 1   ",Q(K);K=67
        DIA;TYPE"STELLUNG 2   ",Q(K);K=68
        DIA;TYPE"STELLUNG 3   ",Q(K);K=97
        DIA;TYPE"VOLUMEN=",Q(K)," CCM";K=83
        TYPE!,!,"BEI FOLGENDEN ABWEICHUNGEN WIRD "
        TYPE"SONDERMESSUNG GESTARTET.",!
        DIA;TYPE"REAKTORLEISTUNG:  ",Q(K)," %";K=84
        DIA;TYPE"PRAEZIPITATOR-ZAEHLRATE:  ",Q(K)," %",!;K=99
        DIA;TYPES"BAND NR. = ",Q(K),!,!
        TYPE"SIND DIE GELISTETEN DATEN RICHTIG ?  ";READS(1)FRAG
        IF(EQ(FRAG,"JA"))GOTO ON5
ON3:    TYPE!,"WELCHE DATEN SOLLEN GEAENDERT WERDEN ?",!
        TYPE"AUF DIE FRAGE NR., OBIGE NR. EINTIPPEN, DANN RETURN."
        TYPE!,"AUF DIE FRAGE WERT, NEUEN WERT EINGEBEN, DANN RETURN.",!
        TYPE!,"WIRD AUF DIE FRAGE NR. EINE Ø EINGEGEBEN, DANN WIRD "
        TYPE"DIE EINGABE BEENDET.",!
ON4:    TYPE"NR.?  ";READ(1)K
        IF(EQ(K,Ø))GOTO ON5
        TYPE"WERT ?  ";AEN=1;IF(EQ(K,99)) READS(1) Q(K);GOTO ON4
        READ(1)Q(K);IF(EQ(K,61))Q(K)=Q(K)/Q(64)
        GOTO ON4
```

```
ON5:     TYPE!,"DIE SPEKTRENABLAGE BEGINNT BEI KASSETTENFILE NR."
         TYPE!,"NORMAL ",Q(65)+1,!,"LANGZEIT ",Q(55)+1,!,"ALARM ",Q(89)+1
         TYPE!,!,"RICHTIG? ";READS(1)FRAG;IF(EQ(FRAG,"JA"))GOTO ON6
         AEN=1;TYPE!,"NORMAL NR.? ";READ(1) Q(65);Q(65)=Q(65)-1
         TYPE"LANGZEIT NR.? ";READ(1) Q(55);Q(55)=Q(55)-1
         TYPE"ALARM NR.? ";READ(1) Q(89);Q(89)=Q(89)-1
ON6:     $I(4);TYPE!,"DAS NAECHSTE SPEKTRUM ERHAELT DIE NR.",Q(4)+1
         TYPE!,"DIE NAECHSTE EICHMESSUNG ERHAELT DIE NR.",Q(5)+1;$F(10)
         TYPE!,"RICHTIG ?  ";READS(1) FRAG
         IF(EQ(FRAG,"JA"))GOTO ON7
         TYPE!,"MIT WELCHER SPEKTRUMS NR. SOLL BEGONNEN WERDEN ?  "
         READ(1) Q(4);Q(4)=Q(4)-1;AEN=1
         TYPE!,"WELCHE NR. SOLL DIE NAECHSTE EICHMESSUNG ERHALTEN ?  "
         READ(1) Q(5);Q(5)=Q(5)-1;IF(EQ(Q(5),0)) Q(81)=1
ON7:     IF(EQ(AEN,0))GOTO ON8
         TYPE!,"SOLLEN DIE STEUERDATEN AUF KASSETTE GELEGT WERDEN ? "
         READS(1)FRAG;IF(EQ(FRAG,"JA")) $U(3);RUN STORE;DELETE STORE
ON8:     IF(EQ(R,1))GOTO ON10
         TYPE!,"ONGAM STARTET",!;Q(59)=0;Q(75)=1;Q(43)=2;Q(62)=0
         Q(78)=0;Q(44)=0;Q(45)=0;Q(48)=0
ON10:    DELETE DIA>
```

SETZ

DIALOG PROGRAM FOR SETTING OF WINDOWS AND CALIBRATION ENERGIES.

CALL: $U(3);RUN SETZ

```
DEFINE SETZ<;    REMARK  29.9.77          &
         LOCAL X,Y,Z,K,WAS,LAB,FEH,N,SE,EI,GZ
         SE=0;EI=0
         IF(EQ(DEF(EIF),0)) SE=1;AEN=0;R=0;DIMENS EIF(200);EIORG(2)
         IF(NE(EIF(10),0)*NE(EIF(11),0))GOTO START
         EI=1;TYPE!,"ES LIEGEN KEINE EICHFAKTOREN VOR, NAECHSTE"
         TYPE" MESSUNG IST EINE EICHMESSUNG",!;Q(81)=1
START:   DEFINE RECH<
         IF(EQ(EI,1))GOTO REC3
         Z=EIF(11)/(2*EIF(10));Y=EIF(12)-ROI1(X+15);Y=Y/EIF(10)
         N=Z*Z-Y;IF(LE(N,0))GOTO REC2
         N=SQRT(N);Y=-Z-N
         IF(LT(Y,0)+GE(Y,2048)) Y=-Z+N
         Y=IP(Y);IF(LT(Y,0)+GE(Y,2048))GOTO REC1
         ROI1(X+1)=Y;GOTO REC3
REC1:    TYPE!,"ENERGIE RICHTIG EINGEGEBEN ?  ";READS(1) WAS
         IF(EQ(WAS,"JA"))GOTO REC2
         GOTO REC3
REC2:    TYPE!,"EICHFAKTOREN ? KANALLAGE > 2048; KANALLAGE WIRD AUF"
         TYPE" 2047 GESETZT",!," ES FOLGT EICHMESSUNG",!;Q(81)=1
         ROI1(X+1)=2047
REC3:    Z=0>
```

```
        TYPE!,"ZULAESSIGE BEFEHLE AUF DIE FRAGE WAS",!
        TYPE!,"I  ,EINGABE"
        TYPE!,"V  ,VERSCHIEBEN VON FENSTERN"
        TYPE!,"A  ,AUSGABE"
        TYPE!,"L  ,LOESCHEN VON FENSTERN"
        TYPE!,"E  ,EICHLINIEN SETZEN"
        IF(EQ(R,Ø)) TYPE!,"U  ,UEBERTRAG VON ROI1 NACH ROI2"
        TYPE!,"$  ,DIALOG BEENDEN",!
FRAG:   FEH=Ø;TYPE!,"WAS ? ";READS(1) WAS
        IF(EQ(WAS,"I")) DET=1;GOTO EIN
        IF(EQ(WAS,"V"))GOTO SHIFT
        IF(EQ(WAS,"E"))GOTO EICH
        IF(EQ(WAS,"U")*EQ(R,Ø))GOTO UEBER
        IF(EQ(WAS,"A")+EQ(WAS,"L"))GOTO AUS
        IF(EQ(WAS,"$"))GOTO ENDE
        TYPE!,"FALSCHER BEFEHL!";GOTO FRAG
AUS:    TYPE"VON FENSTER NR. ";READ(1) X
        TYPE"BIS FENSTER NR. ";READ(1) Y
        LAB="AUS";GOTO ERR
AUS1:   Z=Y-X+1
        IF(EQ(WAS,"L"))GOTO CLEAR
        TYPE!,"ABUNDANZ NEGATIV: PEAK WIRD NICHT ZUR SUMMENBILDUNG "
        TYPE"VERWENDET",!
        TYPE"FENSTERBREITE NEGATIV: PEAK WIRD NICHT ZUR DRIFT"
        TYPE"KONTROLLE VERWENDET",!
        TYPE!,"I";DO(70)<TYPE"-">;TYPE"I",!
        TYPE"I FENSTER NR. I NUCLID I  ENERGIE   I   HWZ  I"
        TYPE"  ABUNDANZ  I FENS.BREIT.I"
        TYPE!,"I";DO(70)<TYPE"-">;TYPE"I"
L1:     Y=17*X+42
        TYPE!,"I ",X," I ";TYPES ROI1(Y)," I "
        TYPE ROI1(Y+1)," I ";TYPES ROI1(Y+2)," I "
        TYPE ROI1(Y+3)," I"
        IF(EQ(ROI1(Y+1),Ø)) TYPE"   000000   I";GOTO L2
        TYPE" ",ROI1(Y-12)," I"
L2:     X=X+1;Z=Z-1;IF(GT(Z,Ø))GOTO L1
        TYPE!,"I";DO(70)<TYPE"-">;TYPE"I",!
        GOTO FRAG
CLEAR:  Z=Z*17;X=X*17+28;K=1;DO(Z)<ROI1(X+K)=Ø;K=K+1>;AEN=1;GOTO FRAG
EIN:    TYPE"FENSTER NR. ";READ(1)X;GZ=Ø
        Y=15;LAB="EIN";GOTO ERR
EIN1:   X=17*X+28;AEN=1
EINØ:   TYPE"NUCLIDNAME ";READS(1) ROI1(X+14)
        TYPE"SOLL ENERGIE EINGEGEN WERDEN ? ";READS(1) WAS
        IF(EQ(WAS,"JA"))GOTO ENER
        TYPE"KANALLAGE ";READ(1) ROI1(X+1);ROI1(X+1)=IP(ROI1(X+1))
        IF(EQ(EI,1)) ROI1(X+15)=ROI1(X+1);GOTO HW
        ROI1(X+15)=EIF(1Ø)*ROI1(X+1)*ROI1(X+1)+EIF(11)*ROI1(X+1)+EIF(12)
        GOTO HW
ENER:   TYPE"ENERGIE IN KEV ";READ(1) ROI1(X+15);WAS="JA"
        RECH;IF(NE(WAS,"JA"))GOTO ENER
HW:     TYPE"HALBWERTSZEIT ";READS(1) ROI1(X+16)
        TYPE"ABUNDANZ ";READ(1) ROI1(X+17)
        TYPE"FENSTERBREITE KANAELE ? ";READ(1) ROI1(X+2)
        IF(GE(ROI1(X+2),29)) ROI1(X+2)=28;GZ=3
        IF(GT(ROI1(X+1)+(ROI1(X+2)/2),2048)) GZ=1
        IF(LT(ROI1(X+1)-(ROI1(X+2)/2),1)) GZ=2
        IF(EQ(GZ,1)) ROI1(X+2)=2*(2048-ROI1(X+1))
```

```
        IF(EQ(GZ,2)) ROI1(X+2)=2*(ROI1(X+1)-1)
        IF(GE(GZ,1)) TYPE!,"FENSTERBREITE NUR ",ROI1(X+2)," KANAELE",!,!
        TYPE"SOLL DIESER PEAK ZUR DRIFTUEBERWACHUNG VERWENDET"
        TYPE" WERDEN ?  ";READS(1) WAS
        IF(NE(WAS,"JA")) ROI1(X+2)=ROI1(X+2)*-1
        TYPE"SOLL DIESER PEAK ZUR SUMMENBILDUNG VERWENDET WERDEN ? "
        READS(1)WAS;IF(NE(WAS,"JA")) ROI1(X+17)=ROI1(X+17)*-1
        GOTO FRAG
SHIFT:  TYPE"VON FENSTER NR. ";READ(1)X
        TYPE"NACH FENSTER NR ";READ(1)Y
        LAB="SHIFT";GOTO ERR
SHI1:   X=17*X+42;Y=17*Y+42;K=Ø;AEN=1
        DO(4)<ROI1(Y+K)=ROI1(X+K);K=K+1>;ROI1(Y-12)=ROI1(X-12)
        ROI1(Y-13)=ROI1(X-13);ROI1(Y-9)=ROI1(X-9);GOTO FRAG
UEBER:  X=1
        DO(15)<K=Ø;Y=17*X+42
        DO(4)<ROI2(Y+K)=ROI1(Y+K);K=K+1>;ROI2(Y-12)=ROI1(Y-12)
        X=X+1>;GOTO FRAG
EICH:   TYPE!,"NICHT VERGESSEN FENSTER VKA SETZEN",!
        TYPE!,"1. EICHENERGIE IN KEV ?  ";READ(1) EIF(96)
        TYPE!,"2. EICHENERGIE IN KEV ?  ";READ(1) EIF(147)
        TYPE!,"3. EICHENERGIE IN KEV ?  ";READ(1) EIF(198)
        EIORG(1);Q(81)=1;GOTO FRAG
ERR:    IF(GE(X,16)+GE(Y,16)) FEH=1;GOTO ERR1
        IF(EQ(X,Ø)+EQ(Y,Ø)) FEH=1;GOTO ERR3
        IF(EQ(LAB,"SHIFT"))GOTO ERRA
        IF(GT(X,Y)) FEH=1;GOTO ERR2
        GOTO ERRA
ERR1:   TYPE!,"FENSTER NR. GROESSER ALS 15.",!;GOTO ERRA
ERR2:   TYPE!,"STARTFENSTER NR. GROESSER ALS ENDEFENSTER NR.",!
        GOTO ERRA
ERR3:   TYPE!,"FENSTER NR. Ø NICHT ZULAESSIG",!
ERRA:   IF(EQ(LAB,"AUS")*EQ(FEH,1)) FEH=Ø;GOTO AUS
        IF(EQ(LAB,"AUS"))GOTO AUS1
        IF(EQ(LAB,"EIN")*EQ(FEH,1)) FEH=Ø;GOTO EIN
        IF(EQ(LAB,"EIN"))GOTO EIN1
        IF(EQ(LAB,"SHIFT")*EQ(FEH,1)) FEH=Ø;GOTO SHIFT
        IF(EQ(LAB,"SHIFT"))GOTO SHI1
        IF(EQ(FEH,1)) FEH=Ø;GOTO EI1
        GOTO EI2
ENDE:   X=60;Q(13)=Ø
        DO(15)<IF(GT(ROI1(X),Ø)) Q(13)=Q(13)+1
        X=X+17>;IF(EQ(SE,1)) ERASE EIF,R,AEN
        DELETE RECH>
```

RESTART

ALLOWS THE DEFINED RESTART OF THE PROGRAM AFTER INTERRUPTION BY MEANS OF PROGRAM STOP SWITCH. BESIDES, IT OPENS A WAY INTO THE DIALOG PROGRAMS FOR MODIFICATION OF CONTROLLING VARIABLES.

CALL: RESTART

```
DEFINE RESTAR<;          REMARK  23.9.77          &
        LOCAL A,R
        IF(EQ(DEF(SPEC),1)) ERASE SPEC
        IF(EQ(DEF(SPEK),1)) ERASE SPEK
        IF(EQ(DEF(EIF),1)) ERASE EIF
        IF(EQ(DEF(WACH),1)) DELETE WACH
        TYPES!,"BAND NR. ",Q(99),"   RICHTIG ? ";READS(1) A
        IF(NE(A,"JA")) TYPE"BAND NR.? ";READS(1) Q(99)
        TYPE!,"SOLLEN PROGRAMMSTEUERDATEN GEAENDERT WERDEN ? "
        READS(1) A;IF(NE(A,"JA"))GOTO RST1
        R=1;ONGAM
RST1:   CONTROL"B00FB";TYPE!,"A C H T U N G !",!,!
        TYPE"VOR START CHECK #1 DURCHFUEHREN, DANN RETURNTASTE"
        KBWAIT;TYPE!,!,"RESTART LAEUFT  ",$TIME,!,!;ACT(81)0
        $F(10);IF(EQ(DEF(ROI2),1))GOTO RST5
        DIMENS ROI2(300);$U(3);OPEN SAFIL2;COPY SAFIL2(1):ROI2
        ERASE SAFIL2
RST5:   IF(EQ(Q(44),1)*EQ(Q(48),0)) FAHR(0);Q(43)=3;GOTO RST2
        IF(EQ(Q(44),1))GOTO RST4
        GOTO RST3
RST2:   MCACOL(400,1);Q(100)=-1;Q(53)=0;RLCPS(Q(61),1);Q(44)=1;GOTO RST4
RST3:   IF(EQ(Q(44),0))GOTO RST4
        IF(GE(Q(44),2)) FAHR(0);MCACOL(Q(1),1);Q(100)=-1
        Q(53)=0;RLCPS(Q(61),1);Q(44)=2
RST4:   STEUER(1)>
```

STEUER

CONTROLS THE SEQUENCE OF THE PROGRAM AS A FUNCTION OF THE FLAGS:
Q(44)=Ø, NORMAL MEASUREMENT OR MEASUREMENT PAUSE IF Q(47)<$TIME
Q(44)=1, DECAY MEASUREMENTS
Q(44)>1, SPECIAL MEASUREMENT
Q(41)=Q(46), CALIBRATION MEASUREMENT IF Q(44)=Ø

CALL: STEUER(A)

A=Ø, CALL BY ONGAM, PROGRAM START WITH CALIBRATION MEASUREMENT
A=1, CALL BY RESTART, CONTINUANCE OF PROGRAM

```
DEFINE STEUER<; REMARK  22.8.77 &
        IF(EQ(ARG(1),1))GOTO STE1
        COUNT(4Ø,1Ø);Q(86)=Q(5Ø);Q(85)=Q(51);EICHEN
        IF(EQ(Q(44),Ø)) SO=1;SONDER
STE1:   IF(EQ(DEF(SO),1)) ERASE SO
        IF(EQ(SENSE(6),1)) TYPE!,"*** ",$TIME," STOP -STEUER-",!;QUIT
        IF(GE($TIME,24ØØØØØ)*GE(Q(1ØØ),Ø)) DATKOR
        IF(GE(Q(48),9)*LE(Q(75),Q(48))*NE(Q(1ØØ),-2))GOTO STE2
        GOTO STE3
STE2:   IF(EQ(Q(44),1)) ABAUS
STE3:   IF(EQ(Q(44),1)*GE(Q(75),13)) ABSCH;GOTO STE1
        IF(EQ(Q(44),1)) ABKLI;GOTO STE1
        IF(EQ(Q(44),2)) SONDER;GOTO STE1
        IF(GE($TIME,Q(47))) NORMES;GOTO STE1
        IF(GE(Q(41),Q(46))+EQ(Q(81),1)) EICHEN;GOTO STE1
        IF(NE(Q(89),Ø)+NE(Q(58),Ø)) ALBEA;GOTO STE1
        IF(EQ(Q(44),Ø)*EQ(Q(69),1)) ABNACH
        IF(EQ(Q(44),Ø)) MEPAUS
        GOTO STE1>
```

NORMES

ORGANIZES THE SEQUENCE OF THE NORMAL MEASUREMENT BY CALLING THE RESPECTIVE SUBROUTINES. THE STARTING DATE FOR THE NEXT MEASUREMENT IS WRITTEN TO Q(47).

CALL: NORMES

```
DEFINE NORMES<; REMARK  28.7.77 &
DEFINE WACH<Q(80)=0
        CONTROL"C00FB";CONTROL"C00KB";WAITD;CONTROL"C13A@";Q(90)=$TIME>
        LOCAL MES1,I,Z,K
        Q(78)=1;ROI1(45)=0;Q(14)=1
        Q(47)=SEC($TIME)+Q(52);Q(47)=TIME(Q(47))
        FAHR(0);MCACOL(Q(1),1);Q(42)=Q(9);Q(53)=0;RLCPS(Q(61),1)
        MES1=SEC(Q(9))+Q(1)-15;MES1=TIME(MES1)
NO01:   RLCPS(Q(61),0)
        IF(EQ(Q(44),1)) Q(4)=Q(4)+1;GOTO NOEX
        IF(GT(MES1,$TIME))GOTO NO01
        WAITD;Q(63)=$TIME;Q(78)=2;WACH
        RLCPS(Q(61),2);TAPE(0);IF(EQ(Q(44),1))GOTO NOEX
        WACH;DUPL(ROI1,ROI2);AUSWER;Q(78)=3;Q(100)=0
        Q(53)=0;RLCPS(Q(61),0)
        IF(GE(Q(44),1))GOTO NOEX
        WACH;PEAVER;Q(78)=4
        Q(53)=0;RLCPS(Q(61),0)
        IF(GE(Q(44),1))GOTO NOEX
        WACH;ROI1(16)=Q(78);Q(65)=Q(65)+1;NOMORG(Q(65),1)
        IF(GE(Q(87),Q(79))) Q(55)=Q(55)+1;LANORG(Q(55),1)
        MESPRO;Q(78)=5
        Q(53)=0;RLCPS(Q(61),0)
        IF(GE(Q(44),1))GOTO NOEX
        WACH;PLOT;ROI1(16)=Q(78);GOTO NO2
NOEX:   TYPE!,"+++ PROG. NORMES ABGEBROCHEN  / ";$I(4);TYPE"SPEKT.NR. "
        TYPE Q(4)," / LAUFINDEX =",$I(2)+Q(78),!;$F(10);Q(88)=Q(72)
        ROI1(16)=Q(78);K=78;TAPE(1);GOTO NOEND
NO2:    CONTROL"C00FB";CONTROL"C00KB";WAITD;CONTROL"B00FB"
NOEND:  DELETE WACH>
```

SONDER

ORGANIZES THE RUN OF A SPECIAL MEASUREMENT BY CALLING THE RESPECTIVE SUBROUTINES. STORES IN ADDITION ROI1 IN A PERMANENT FILE. SETS NEW REFERENCE VALUES FOR RL AND PCPS. STARTING DATE FOR NEXT NORMAL MEASUREMENT IS WRITTEN TO Q(47).

CALL: SONDER

```
DEFINE SONDER<; REMARK  28.9.77         &
DEFINE WACH<Q(80)=0
        CONTROL"C00FB";CONTROL"C00KB";WAITD;CONTROL"C13A@";Q(90)=$TIME>
        LOCAL MES1,Z,I,K
        Q(47)=SEC($TIME)+Q(52);Q(47)=TIME(Q(47))
        IF(EQ(Q(45),-2)+EQ(Q(45),-4))GOTO SO1
        IF(EQ(Q(44),0)) FAHR(0)
        MCACOL(Q(1),1);Q(53)=0;RLCPS(Q(61),1)
SO1:    Q(14)=3;Q(85)=Q(51);Q(42)=Q(9);Q(86)=Q(50)
        IF(EQ(ROI1(14),7)) Q(14)=7;GOTO SO01
        IF(EQ(ROI1(14),3)*GE(ROI1(16),5)) Q(14)=7
SO01:   MES1=SEC(Q(9))+Q(1)-15;MES1=TIME(MES1);Q(62)=1
SO2:    RLCPS(Q(61),0)
        IF(EQ(Q(44),1)) Q(4)=Q(4)+1;GOTO SOEX
        IF(GT(MES1,$TIME))GOTO SO2
        WAITD;Q(63)=$TIME;WACH;Q(62)=2
        RLCPS(Q(61),2);TAPE(0);IF(EQ(Q(44),1))GOTO SOEX
SO4:    WACH;DUPL(ROI1,ROI2);AUSWER;Q(62)=3;Q(100)=0
        Q(53)=0;RLCPS(Q(61),0)
        IF(GE(Q(44),1))GOTO SOEX
        WACH;PEAVER;Q(62)=4
        Q(53)=0;RLCPS(Q(61),0)
        IF(GE(Q(44),1))GOTO SOEX
        WACH;ROI1(16)=Q(62);Q(65)=Q(65)+1
        IF(NE(ROI1(14),7)) Q(55)=Q(55)+1;GOTO SO5
        IF(EQ(Q(55),0)) Q(55)=!
SO5:    LANORG(Q(55),1);NOMORG(Q(65),1)
        MESPRO;Q(62)=5
        Q(53)=0;RLCPS(Q(61),0)
        IF(GE(Q(44),1))GOTO SOEX
        WACH;PLOT;ROI1(16)=Q(62);GOTO SO6
SOEX:   TYPE!,"+++ PROG. SONDER ABGEBROCHEN  / ";$I(4);TYPE"SPEKT.NR. "
        TYPE Q(4)," / LAUFINDEX =",$I(2)+Q(62);$F(10);Q(88)=Q(72)
        ROI1(16)=Q(62);K=62;TAPE(1);GOTO SOEND
SO6:    CONTROL"C00FB";CONTROL"C00KB";WAITD;CONTROL"B00FB"
SOEND:  DELETE WACH>
```

EICHEN

ORGANIZES THE RUN OF A QUADRATIC CALIBRATION BY CALLING THE RESPECTIVE SUBROUTINES. BRINGS THE COLLIMATOR TO POSITION 3 FOR THIS PURPOSE.

CALL: EICHEN

```
DEFINE EICHEN<;    REMARK  3.10.77          &
DEFINE WACH<Q(80)=0
        CONTROL"C00FB";CONTROL"C00KB";WAITD;CONTROL"C13A@";Q(90)=$TIME>
        LOCAL MES1,I,Z,K
        Q(59)=0;Q(60)=0;Q(14)=2
        DIMENS EIF(200);EIORG(2)
LEI0:   FAHR(3);MCACOL(Q(54),1);Q(53)=0;RLCPS(Q(61),1)
        Q(42)=Q(9);Q(59)=1;EIF(200)=Q(15)
        MES1=SEC(Q(9))+Q(54)-15
        MES1=TIME(MES1)
LEI1:   RLCPS(Q(61),0)
        IF(GE(Q(44),1)*EQ(Q(81),0))GOTO EIEX
        IF(GT(MES1,$TIME))GOTO LEI1
        WAITD;Q(63)=$TIME;RLCPS(Q(61),2);Q(59)=2;FAHR(0);MCACOL(10,0)
        WACH;TAPE(0);IF(GE(Q(44),1)*EQ(Q(81),0))GOTO EIEX
        WACH;FENAUS;Q(100)=0
        EIKOF;EIORG(1);Q(59)=3;Q(41)=0;Q(81)=0
        Q(53)=0;RLCPS(Q(61),0)
        IF(GE(Q(44),1)) Q(58)=Q(5);GOTO EIEX
        WACH;EIPRO;Q(59)=4;Q(58)=0
        Q(53)=0;RLCPS(Q(61),0)
        IF(GE(Q(44),1))GOTO EIEX
        WACH;PLOT;ERASE EIF;GOTO LEI2
EIEX:   TYPE!,"+++ PROG. EICHEN ABGEBROCHEN  / ";$I(4);TYPE"SPEKT.NR. "
        TYPE Q(4)," / LAUFINDEX =",$I(2)+Q(59),!;$F(10);Q(88)=Q(72)
        ERASE EIF;K=59;TAPE(1);GOTO EIEND
LEI2:   CONTROL"C00FB";CONTROL"C00KB";WAITD;CONTROL"B00FB"
EIEND:  DELETE WACH;>
```

ABKLI

ORGANIZES A SEQUENCE OF MEASUREMENTS IF THE REACTOR POWER IS LOWER THAN THE LIMIT VALUE Q(61)

CALL: ABKLI

MEASUREMENT	MEASUREMENT TIME	STARTING DATE
1.	400 S	IMMEDIATLY AFTER SCRAM
2.	400 S	IMMEDIATLY AFTER 1ST MEAS.
3.	400 S	IMMEDIATLT AFTER 2ND MEAS.
4.	400 S	30 MINUTES AFTER SCRAM
5.	400 S	60 MINUTES AFTER SCRAM
6.	400 S	90 MINUTES AFTER SCRAM
7.	1000 S	IMMEDIATLY AFTER 6TH MEAS.
8.	1000 S	2 HOURS AFTER SCRAM
9.	1000 S	4 HOURS AFTER SCRAM
10.	1000 S	8 HOURS AFTER SCRAM
11.	10000 S	12 HOURS AFTER SCRAM
12.	10000 S	24 HOURS AFTER SCRAM

THE MEASUREMENTS ARE STARTED ALTERNATELY IN THE 1ST OR 2ND MCA-HALF, BEGINNING WITH THE 2ND HALF. VIA MCLES THE MCA IS READ ON CASSETTE AND ON TAPE EITHER DURING THE MEASUREMENT OR DURING THE PAUSE. THE EVALUATION OF SPECTRA STARTS AFTER THE 9TH MEASUREMENT.
Q(48)= DECAY MEASUREMENT COUNTER CONTROLS THIS SEQUENCE
Q(43) SELECTS MCA-HALF FOR START, Q(71) SELECTS MCA-HALF FOR READOUT
Q(75)= COUNTER FOR MEASUREMENTS ALREADY EVALUATED
Q(88) CONTAINS STARTING DATE FOR THE NEXT DECAY MEASUREMENT

```
DEFINE ABKLI<;  REMARK  12.9.77  &
        LOCAL LES,A,B,Z,ALT,MES1,ABK
        $U(3);OPEN SAFIL2;COPY RO12:SAFIL2(1);ABK=1;Z=0;Q(86)=Q(50)
        ERASE RO12;DIMENS SPEK(1090);Q(14)=5;Q(69)=0;Q(85)=Q(51)
        IF(NE(Q(48),0))GOTO PA1
        TYPE!,"+++ ABKLINGMESSUNGEN, ABSCHALTUNG UM ";UHRAUS(Q(88),1)
        IF(EQ(Q(45),-2)) Q(45)=-1
        TYPE!,!;Q(71)=3;Q(80)=0;Q(75)=1;Q(76)=400;Q(100)=-1;GOTO AB3
AB0:    IF(EQ(Q(43),3)) Q(43)=2;GOTO AB1
        Q(43)=3
AB1:    ABK=1;Q(80)=Z;MCACOL(Q(76),1);Q(53)=0;RLCPS(Q(61),1)
        IF(EQ(Q(45),-2)) Q(45)=-1
AB3:    IF(GT(Q(48),0)*LE(Q(48),3)) MCLES;GOTO AB01
        IF(EQ(Q(48),6)+EQ(Q(48),7)) MCLES;GOTO AB01
        GOTO AB02
AB01:   IF(EQ(Q(71),3)) Q(71)=2;GOTO AB02
        Q(71)=3
AB02:   MES1=SEC(Q(9))+Q(76)+Z-20;MES1=TIME(MES1)
AB2:    RLCPS(Q(61),0);IF(NE(Q(44),1))GOTO ABEX
        IF(GT(MES1,$TIME))GOTO AB2
        WAITD;Q(63)=$TIME;RLCPS(Q(61),2);Q(48)=Q(48)+1;Q(4)=Q(4)+1
        A=17;DO(5)<Q(A)=Q(A)*Q(64);A=A+1>
        A=2;DO(35)<SPEK(1054+A)=Q(A);A=A+1>
```

```
        SPEK(1059)=Q(48);SPEK(1063)=Q(9)
        IF(LE(Q(48),2))GOTO ABØ
PAUS:   IF(EQ(Q(48),6)) Q(76)=1000;GOTO ABØ
        IF(EQ(Q(48),10)) Q(76)=10000
        IF(EQ(Q(48),3)) Q(77)=1800
        IF(EQ(Q(48),8)) Q(77)=7200
        IF(EQ(Q(48),9)) Q(77)=14400
        IF(EQ(Q(48),11)) Q(77)=43200
        IF(EQ(Q(48),12)) MCLES;Z=Ø;Q(100)=Ø;Q(43)=2;ABK=Ø;GOTO PA3
        Q(88)=SEC(Q(88))+Q(77);Q(88)=TIME(Q(88))
        TYPE!,"+++ NAECHSTE ABKLINGMESSUNG UM ";UHRAUS(Q(88),Ø);TYPE!
        IF(EQ(Q(48),3)+EQ(Q(48),6)+EQ(Q(48),7))GOTO PA1
        MCLES;Z=Ø;IF(EQ(Q(71),3)) Q(71)=2;GOTO PA1
        Q(71)=3
PA1:    ABK=Ø;Q(53)=Ø;RLCPS(Q(61),Ø);IF(NE(Q(44),1))GOTO ABEX
        IF(GE(Q(48),9)*GE(Q(48),Q(75))*NE(Q(100),-2))GOTO ABEND
        IF(EQ(Q(48),9)*EQ(Q(75),1))GOTO ABEND
        IF(EQ(Q(48),3)+EQ(Q(48),6)+EQ(Q(48),7))GOTO PA2
        IF(EQ(SENSE(6),1)) TYPE!,"*** ",$TIME," STOP -ABKLI-",!;QUIT
PA2:    IF(GT(Q(88),$TIME))GOTO PA1
        IF(GE($TIME,2400000)) Q(100)=Ø;DATKOR
        Q(100)=-1;FAHR(Ø);GOTO ABØ
PA3:    Q(53)=Ø;RLCPS(Q(61),Ø);IF(NE(Q(44),1))GOTO ABEX
        GOTO ABEND
ABEX:   TYPE!,"+++ ABKLINGMESSUNGEN ABGEBROCHEN.",!;ERASE SPEK;Q(44)=2
        IF(EQ(ABK,Ø))GOTO ABEX1
        CONTROL Q(43),"ØØFB";CONTROL Q(43),"ØØHB";WAITD
        Q(4)=Q(4)+1;TYPE"+++ ANGEFANGENE MESSUNG UNTER NR. ",$I(4)+Q(4)
        TYPES" AUF TAPE / BAND NR. ",Q(99);$F(1Ø)
        TYPE!,"+++ STARTZEIT ";UHRAUS(Q(9),1);TYPE!
ABEX1:  Q(43)=2;Q(100)=Ø;FAHR(Ø);IF(EQ(Q(45),-1)) MCACOL(Q(1),1)
        Q(44)=2;IF(GE(Q(48),1)) Q(69)=1
ABEND:  IF(EQ(DEF(SPEK),1)) ERASE SPEK
        $U(3);DIMENS ROI2(300)
        COPY SAFIL2(1):ROI2;ERASE SAFIL2>
```

ABSCH

WAIT LOOP FOR PROGRAM AFTER DECAY MEASUREMENTS HAVE BEEN TERMINATED. PROGRAM WAITS UNTIL RL>Q(61) AND STARTS THEN BY A SPECIAL MEASUREMENT.

CALL: ABSCH

```
DEFINE ABSCH<;  REMARK  16.8.77 &
        Q(100)=Ø;Q(14)=6;IF(NE(Q(44),1))GOTO AB2
        IF(GE($TIME,2400000)) DATKOR
        DATAUS
        $F(6);TYPE!,"+++ PROGRAMM WARTET BIS REAKTORLEISTUNG"
        TYPE" GROESSER ALS",Q(61)*Q(64)," MW WIRD.",!;$F(1Ø)
AB1:    Q(53)=Ø;RLCPS(Q(61),Ø)
        IF(GE($TIME,2400000)) DATKOR
        IF(EQ(SENSE(6),1)) TYPE!,"*** ",$TIME," STOP -ABSCH-",!;QUIT
        IF(EQ(Q(44),1))GOTO AB1
AB2:    TYPE!,"+++ REAKTOR WIEDER IN BETRIEB.",!;DATAUS;Q(44)=2
        FAHR(Ø);IF(EQ(Q(45),-1)) MCACOL(Q(1),1);Q(53)=Ø;RLCPS(Q(61),1)
        Q(37)=Ø;Q(75)=1;Q(44)=2;Q(48)=Ø>
```

ABAUS

ORGANIZES THE EVALUATION OF DECAY MEASUREMENTS BY CALLING SUBROUTINES, BEGINNING AFTER THE 9TH DECAY MEASUREMENT. THE ALARM FILE MIGHT BE EVALUTED BEFORE. THE EVALUTION PROCEEDS DURING THE BREAKS BETWEEN THE DECAY MEASUREMENTS.

CALL: ABAUS

```
DEFINE ABAUS<;  REMARK  16.8.77  &
        LOCAL MES1,W1,W2,W3,W4,ZEIT,K,CP
        ZEIT=SEC(Q(88))-1800;ZEIT=TIME(ZEIT)
        Q(100)=-1;IF(NE(Q(75),1))GOTO ABA0
        IF(NE(Q(89),0)+NE(Q(58),0)) ALBEA
ABA0:   DEFINE WACH<CONTROL"C00FB";CONTROL"C00KB";WAITD
        CONTROL"C13A@";Q(90)=$TIME;Q(80)=0>
        IF(NE(Q(44),1))GOTO ABAEX1
ABA1:   CP=0;IF(EQ(SENSE(6),1)) TYPE!,"*** ",$TIME;CP=1;GOTO ABA5
        WACH;DUPL(ROI1,ROI2)
        AUSWAB;Q(80)=SEC(W2)-SEC(W1)+SEC(W4)-SEC(W3)
        Q(53)=0;RLCPS(Q(61),0);IF(NE(Q(44),1))GOTO ABAEX
ABA2:   WACH;PEAVER;Q(75)=Q(75)+1;Q(53)=0;RLCPS(Q(61),0)
        IF(NE(Q(44),1))GOTO ABAEX
ABA3:   WACH;MESPRO
        IF(EQ(Q(75),13)) Q(65)=Q(65)+1;NOMORG(Q(65),1)
        Q(53)=0;RLCPS(Q(61),0);IF(NE(Q(44),1))GOTO ABAEX
        WACH;PLOT;IF(NE(Q(44),1))GOTO ABAEX
        IF(GE(Q(48),12))GOTO ABA4
        IF(GE($TIME,ZEIT)) Q(100)=-2;GOTO ABA5
ABA4:   IF(LE(Q(75),Q(48)))GOTO ABA1
        IF(GE(Q(75),13)) Q(100)=0;Q(43)=2;Q(69)=0
        GOTO ABA5
ABAEX:  TYPE!,"+++ AUSWERTUNG ABKLINGMESSUNGEN ABGEBROCHEN",!
ABAEX1: Q(43)=2;Q(69)=1;Q(100)=0;MES1=Q(44);TAPE(2)
ABA5:   DELETE WACH;IF(EQ(CP,1)) TYPE" STOP -ABAUS-",!;QUIT
        CP=0>
```

MEPAUS

WAIT LOOP FOR THE PROGRAM UNTIL THE DEFINED END OF THE MEASUREMENT PAUSE Q(47) HAS BEEN REACHED. DURING THE MEASUREMENT PAUSE RL AND PCPS ARE PERMANENTLY MEASURED AND CHECKED FOR EXEEDING OF THE LIMIT VALUES. (THE MEASUREMENT PAUSE MIGHT BE PRECOCIOULY TERMINATED) AT PREDETERMINED TIME INTERVALS Q(49) AVERAGE, INITIAL, FINAL, MAXIMUM AND MINIMUM VALUES OF RL AND PCPS ARE PRINTED.

CALL: MEPAUS

```
DEFINE MEPAUS<; REMARK  30.8.77          &
        LOCAL AUS,ENDE,ANF,K
        TYPE!,!,"+++ MESSPAUSE",!;DATAUS
        TYPE!,"+++ NAECHSTE MESSUNG UM ";UHRAUS(Q(47),0);TYPE!,!
        $F(10);TYPE!,!,"-----",!,!
ME1:    IF(GE($TIME,2400000)) DATKOR
        AUS=SEC($TIME)+Q(49);AUS=TIME(AUS)
        ANF=$TIME;Q(53)=0;RLCPS(Q(61),1)
        IF(NE(Q(44),0))GOTO MEPEX
ME2:    RLCPS(Q(61),0)
        IF(NE(Q(44),0))GOTO MEPEX
        IF(GE($TIME,Q(47)))GOTO ME3
        IF(EQ(SENSE(6),1)) TYPE!,"*** ",$TIME," STOP -MEPAUS-",!;QUIT
        IF(GT(AUS,$TIME))GOTO ME2
ME3:    RLCPS(Q(61),2);ENDE=$TIME
        K=17;DO(5)<Q(K)=Q(K)*Q(64);K=K+1>
        TYPE"IN DER ZEIT VON ";UHRAUS(ANF,0);TYPE" BIS ";UHRAUS(ENDE,0)
        TYPE!,"WURDEN FOLGENDE WERTE GEMESSEN.",!,!
        DO(19)<TYPE" ">;TYPE"MITTEL     START     ENDE      "
        TYPE"MAXIMAL   MINIMAL",!
        TYPE"REAKTORLEISTUNG   ";$F(6);K=17
        DO(5)<TYPE Q(K),"     ";K=K+1>;TYPE"MW",!,"PRAEZIP-ZAEHLR. "
        $I(8);DO(5)<TYPE Q(K),"  ";K=K+1>;TYPE" IMP/S",!,!;$F(10)
        DO(5)<TYPE"-">;TYPE!,!
        IF(GE($TIME,Q(47))) TYPE"+++ MESSPAUSE BEENDET",!;GOTO ME5
        GOTO ME1
MEPEX:  TYPE!,"+++ MESSPAUSE VORZEITIG BEENDET",!;Q(88)=Q(72);K=Q(44)
        IF(EQ(Q(44),1)) Q(48)=0;Q(43)=3;FAHR(0);MCACOL(400,1);GOTO ME4
        FAHR(0);IF(EQ(Q(45),-1)) MCACOL(Q(1),1);GOTO ME4
        GOTO ME5
ME4:    Q(53)=0;RLCPS(Q(61),1);Q(44)=K
ME5:    K=0>
```

ALBEA

PROCESSES DURING THE MEASUREMENT PAUSE THE SPECTRA ON THE ALAR1 TO ALAR5 ALARM FILES. SEARCHES FOR PEAK COMPARISION THE LAST SPECTRUM BEFORE ALAR1 ON THE NORMAL FILE VIA ROI1(40). IN CASE OF INTERRUPTION OF ALBEA Q(73) CONTAINS THE CONTINUING ADDRESS FOR LATER PROCESSING.

CALL: ALBEA

```
DEFINE ALBEA<;  REMARK  26.9.77  &
DEFINE WACH<Q(80)=0
        CONTROL"C00FB";CONTROL"C00KB";WAITD;CONTROL"C13A@";Q(90)=$TIME>
        LOCAL MES1,AUS,SUC,ALAR,K,CP
        TYPE!,"+++ ALARMFILE-BEARBEITUNG LAEUFT.",!;ALAR=1;CP=0
        IF(EQ(Q(89),0)) ALAR=2;GOTO EINA
        $U(3);OPEN SAFIL1;COPY ROI1:SAFIL1(1)
        IF(EQ(Q(98),0)+GT(Q(98),5)) Q(98)=1
STA:    IF(EQ(SENSE(6),1)) TYPE!,"*** ",$TIME;CP=1;GOTO ALEND
        WACH;ALORG(Q(98),2);SUC=1
        IF(EQ(Q(98),1)) NOMORG(ROI1(40),3);GOTO TEST
        IF(NE(ROI1(40),Q(73))) NOMORG(ROI1(40),3);GOTO TEST
        ALORG(Q(98)-1,3)
TEST:   Q(53)=0;RLCPS(Q(61),0)
        IF(EQ(Q(14),5)*NE(Q(44),1))GOTO ALEX
        IF(NE(Q(14),5)*GT(Q(44),0))GOTO ALEX
        IF(EQ(SUC,1)) SUC=0;GOTO AUSW
        IF(EQ(AUS,1))GOTO AUS1
        IF(EQ(AUS,0))GOTO AUS2
        GOTO AUSW3
AUSW:   WACH;Q(73)=ROI1(40)
        IF(LE(Q(74),3)) Q(74)=ROI1(16)
        IF(EQ(Q(74),4))GOTO AUS1
        IF(GE(Q(74),5))GOTO AUS2
        PEAVER;Q(74)=4;AUS=1;GOTO TEST
AUS1:   MESPRO;Q(74)=5;AUS=0;GOTO TEST
AUS2:   Q(98)=Q(98)+1;Q(74)=0;IF(LE(Q(98),Q(89)))GOTO STA
EINA:   IF(EQ(Q(58),0))GOTO AUS3
        WACH;DIMENS EIF(200);EIORG(2);EIPRO;Q(58)=0;ERASE EIF
        GOTO TEST
AUS3:   CONTROL"B00FB";Q(89)=0;Q(98)=1;GOTO ALEND
ALEX:   TYPE!,"+++ ALARMFILE-BEARBEITUNG ABGEBROCHEN.",!
        MES1=Q(44);TAPE(2)
ALEND:  IF(EQ(ALAR,2))GOTO AUS4
        $U(3);COPY SAFIL1(1):ROI1;ERASE SAFIL1
AUS4:   DELETE WACH;IF(EQ(CP,1)) TYPE" STOP -ALBEA-",!;QUIT
        ALAR=0>
```

ABNACH

IN CASE THE DECAY MEASUREMENTS ARE TERMINATED BEFORE THE 12TH MEASUREMENT IS DONE BECAUSE RL>Q(61), THE EVALUATION OF THE DECAY MEASUREMENT BY ABNACH IS FOLLWED UP DURING THE NEXT MEASUREMENT PAUSE. ABNACH IS CALLED IN STEUER VIA THE FLAG Q(69)=1

CALL: ABNACH

```
DEFINE ABNACH<;           REMARK  16.3.77  &
DEFINE WACH<CONTROL"C00FB";CONTROL"C00KB";WAITD;CONTROL"C13A@"
        Q(90)=$TIME;Q(80)=0>
        LOCAL ALAR,ABN,ZEIT,TEXT,W1,W2,W3,W4,CP
        CP=0;IF(GT(Q(75),Q(48))+GE(Q(75),13)) Q(69)=0;Q(48)=0;GOTO ABN3
        ZEIT=SEC(Q(47))-1800;ZEIT=TIME(ZEIT)
        IF(GE($TIME,ZEIT))GOTO ABN3
        TYPE!,"+++ AUSWERTUNG ABKLINGMESSUNG WIRD NACHGEHOLT",!
        ALAR=1;TEXT="A"
        $U(3);OPEN SAFIL1;COPY ROI1:SAFIL1(1)
        ABN=0;IF(EQ(Q(75),1)) ABN=1
        IF(EQ(ABN,0)) Q(75)=Q(75)-1
        WACH;AUSWAB;Q(80)=SEC(W2)-SEC(W1)+SEC(W4)-SEC(W3)
        IF(EQ(ABN,0)) Q(75)=Q(75)+1
        Q(53)=0;RLCPS(Q(61),0);IF(GE(Q(44),1))GOTO ABNEX
        IF(EQ(ABN,1))GOTO ABN01
ABN1:   WACH;DUPL(ROI1,ROI2);AUSWAB
        Q(80)=SEC(W2)-SEC(W1)+SEC(W4)-SEC(W3);Q(53)=0
        RLCPS(Q(61),0);IF(GE(Q(44),1))GOTO ABNEX
ABN01:  Q(75)=Q(75)+1;WACH;PEAVER;ABN=0;Q(53)=0;RLCPS(Q(61),0)
        IF(GE(Q(44),1))GOTO ABNEX
        WACH;MESPRO;Q(53)=0;RLCPS(Q(61),0);IF(GE(Q(44),1))GOTO ABNEX
        IF(GT(Q(75),Q(48))) Q(69)=0;GOTO ABN2
        IF(GE($TIME,ZEIT))GOTO ABN2
        IF(EQ(SENSE(6),1)) TYPE!,"*** ",$TIME;CP=1;GOTO ABN2
        GOTO ABN1
ABNEX:  IF(EQ(Q(44),1)) Q(48)=0;Q(69)=0
        MES1=Q(44);TAPE(2)
ABN2:   $U(3);COPY SAFIL1(1):ROI1;ERASE SAFIL1
ABN3:   DELETE WACH;IF(EQ(CP,1)) TYPE" STOP -ABNACH-",!;QUIT
        CP=0>
```

AUSWER

PROGRAM FOR SPECTRUM EVALUATION; WRITES THE RESULT IN ARRAY ROI1. THE EVALUTION IS MADE IN TWO STEPS: 1. QUARTER, THEN 2. QUARTER OF THE MCA IS WRITTEN TO ARRAY SPEC WITH AN OVERLAPPING OF 30 CHANNELS. IN CASE OF AN EXCESSIVE CHANNEL DRIFT, A FLAG IS SET FOR NEW CALIBRATION, READING TIME OF MCA IS STORED IN Q(80)

CALL: AUSWER

K= WINDOW COUNTER
X= ARRAY ADDRESS FOR ROI1
ISUPK= CONTENT PEAK + BACKGROUND
DPK= DELTA PEAK CONTENT
MAXI= ADDRESS WITH MAXIMUM CHANNEL CONTENT
NL= NUMBER OF CHANNELS LEFT OF THE PEAK MAXIMUM
NR= NUMBER OF CHANNELS RIGHT OF THE PEAK MAXIMUM
NUEB= NUMBER OF CHANNELS ABOVE HALF VALUE
LKN= LAST CHANNEL (1024 OR 1054)
R= PRECISE CHANNEL POSITION
HW= HALF VALUE

```
DEFINE AUSWER<; REMARK 7.12.77    &
        LOCAL K,KK,X,I,ISUPK,DPK,NL,NR,NUEB,M,MAXI,T,R,N
        LOCAL LOOP,LKN,STAT,KONZ,HW,A,B,C,J
        $U(3);OPEN SAFIL2;COPY ROI2:SAFIL2(1);ERASE ROI2
        DIMENS SPEC(1054);Q(96)=Q(66);$I(3)
        CONTROL"D00FB";A=$TIME;CONTROL"D00IB";TRANSFER(SPEC,1,1024,-1)
        CONTROL"C00FB";B=$TIME;CONTROL"C13A@";ROI1(1)=SPEC(1)
        IF(LE(Q(2),ROI1(1))) Q(2)=ROI1(1)
        Q(3)=Q(2)-ROI1(1);T=17;DO(5)<Q(T)=Q(T)*Q(64);T=T+1>
        T=1;DO(35)<T=T+1;ROI1(T)=Q(T)>;ROI1(9)=Q(42)
        IF(EQ(ROI1(15),2)+EQ(ROI1(15),20)) Q(96)=Q(67)
        IF(EQ(ROI1(15),3)+EQ(ROI1(15),30)) Q(96)=Q(68)
        A=SEC(B)-SEC(A);I=17;J=0;K=1;LOOP=0;LKN=1024;KK=0
AUSW1:  X=I*K+28;IF(GE(K,16))GOTO AUSEX
        IF(LE(ROI1(X+15),0))GOTO LOPEX
        IF(GE(ROI1(X+1),1010)*EQ(LOOP,0)) LOOP=1;KK=1;GOTO LOPEX
        MAXI=ROI1(X+1)-J;N=ABS(ROI1(X+2))
        M=MAXI-IP(N/2);DO(N)<IF(LE(SPEC(MAXI),SPEC(M))) MAXI=M
        M=M+1>
        M=SPEC(MAXI+1);IF(GE(M,SPEC(MAXI-1)))M=SPEC(MAXI-1)
        R=SPEC(MAXI+1)-SPEC(MAXI-1);N=2*(SPEC(MAXI)-M)
        IF(EQ(N,0)) N=1
        R=R/N;R=R+J+MAXI
        IF(GE(ABS(R-ROI1(X+1)),Q(82))*GT(ROI1(X+2),0)) Q(41)=Q(46)
        ROI1(X+3)=R;ROI1(X+4)=ROI1(10)*R*R+ROI1(11)*R+ROI1(12)
        NR=1;NL=1
RECHTS: IF(GT(MAXI+NR+1,LKN)) TYPE!,"+++ PEAK",K," RECHTS";GOTO LINKS
        STAT=SPEC(MAXI+NR+1)+2*SQRT(SPEC(MAXI+NR+1))
        IF(GT(SPEC(MAXI+NR),STAT)) NR=NR+1;GOTO RECHTS
        IF(GT(MAXI+NR+2,LKN)) TYPE!,"+++ PEAK",K," RECHTS";GOTO LINKS
        STAT=SPEC(MAXI+NR+2)+2*SQRT(SPEC(MAXI+NR+2))
```

```
        IF(GT(SPEC(MAXI+NR),STAT)) NR=NR+1;GOTO RECHTS
LINKS:  IF(LT(MAXI-NL-1,1)) TYPE!,"+++ PEAK",K," LINKS";GOTO WEIT
        STAT=SPEC(MAXI-NL-1)+2*SQRT(SPEC(MAXI-NL-1))
        IF(GT(SPEC(MAXI-NL),STAT)) NL=NL+1;GOTO LINKS
        IF(LT(MAXI-NL-2,1)) TYPE!,"+++ PEAK",K," LINKS";GOTO WEIT
        STAT=SPEC(MAXI-NL-2)+2*SQRT(SPEC(MAXI-NL-2))
        IF(GT(SPEC(MAXI-NL),STAT)) NL=NL+1;GOTO LINKS
WEIT:   ROI1(X+8)=NR+NL+1
        M=1;N=NR;IF(LE(SPEC(MAXI-NL),SPEC(MAXI+NR))) M=Ø.9;N=-1*NL
        HW=(2*SPEC(MAXI)+SPEC(MAXI-NL)+M*(SPEC(MAXI+N)))/4
        NUEB=Ø;N=Ø;M=NR+NL+1
        DO(M)<STAT=SPEC(MAXI-NL+N)-2*SQRT(SPEC(MAXI-NL+N))
        IF(GT(STAT,HW)) NUEB=NUEB+1
        N=N+1>
        ROI1(X+9)=(NUEB+1)*ROI1(11)
        IF(LE(SPEC(MAXI-NL),SPEC(MAXI+NR))) M=1.9*SPEC(MAXI-NL);GOTO U1
        M=SPEC(MAXI-NL)+SPEC(MAXI+NR)
U1:     ROI1(X+6)=(NL+NR-1)*M/2;N=NR+NL-1;M=MAXI-NL+1;ISUPK=Ø
        DO(N)<ISUPK=ISUPK+SPEC(M);M=M+1>
        DPK=2*SQRT(ISUPK)+N*SQRT(SPEC(MAXI-NL))+N*SQRT(SPEC(MAXI+NR))
        ROI1(X+7)=ISUPK-ROI1(X+6)
        IF(EQ(ROI1(X+7),Ø)) ROI1(X+7)=1
        ROI1(X+1Ø)=DPK*1ØØ/ROI1(X+7)
        ROI1(X+12)=ROI1(X+5)*Q(96)
        KONZ=ROI1(X+12)*Q(97)*ABS(ROI1(X+17))*ROI1(1)
        IF(EQ(KONZ,Ø)) KONZ=1Ø↑-9Ø
        ROI1(X+11)=(ROI1(X+7)*3.7*1Ø↑1Ø)/KONZ
        ROI1(X+13)=(DPK*3.7*1Ø↑1Ø)/KONZ
LOPEX:  IF(GT(K,Q(13)))GOTO AUSEX
        IF(EQ(KK,1)) KK=Ø;GOTO AUSW2
        K=K+1
AUSW2:  IF(NE(LOOP,1))GOTO AUSW1
        M=1;LKN=1Ø54;DO(3Ø)<SPEC(M)=SPEC(M+994);M=M+1>
        CONTROL"EØØFB";B=$TIME;CONTROL"EØØIB";TRANSFER(SPEC,31,1Ø24,-1)
        CONTROL"CØØFB";CONTROL"C13A@";C=$TIME
        Q(8Ø)=A+SEC(C)-SEC(B);J=994;LOOP=2;GOTO AUSW1
AUSEX:  ERASE SPEC
        DIMENS ROI2(3ØØ);$U(3);COPY SAFIL2(1):ROI2;ERASE SAFIL2
        $F(1Ø);TYPE!>
```

AUSWAB

PROGRAM FOR EVALUATION OF DECAY MEASUREMENTS; WRITES THE RESULT IN ARRAY ROI1. THE EVALUTION IS MADE IN TWO 1K-STEPS WITH AN OVERLAPPING TAKING PLACE OF 30 CHANNELS. THE SPECTRA ARE READ FROM CASSETTE BY MEANS OF SPORG.

CALL: AUSWAB

K= WINDOW COUNTER
X= ARRAY ADDRESS FOR ROI1
ISUPK= CONTENT PEAK + BACKGROUND
DPK= DELTA PEAK CONTENT
MAXI= ADDRESS WITH MAXIMUM CHANNEL CONTENT
NL= NUMBER OF CHANNELS LEFT OF THE PEAK MAXIMUM
NR= NUMBER OF CHANNELS RIGHT OF THE PEAK MAXIMUM
NUEB= NUMBER OF CHANNELS ABOVE HALF VALUE
LKN= LAST CHANNEL (1024 OR 1054)
R= PRECISE CHANNEL POSITION
HW= HALF VALUE

```
DEFINE AUSWAB<; REMARK 7.12.77   &
        LOCAL X,I,K,KK,ISUPK,DPK,NL,NR,NUEB,M,MAXI,T,R,N
        LOCAL LOOP,LKN,STAT,KONZ,HW,A,B,C,J,FL
        Q(96)=Q(66);I=17;K=1;J=0;LOOP=0;LKN=1024
        KK=0;$U(3);OPEN SAFIL2;COPY ROI2:SAFIL2(1);ERASE ROI2
        DIMENS SPEK(1090);SPORG(Q(75),1,0);ROI1(1)=SPEK(1);$I(3)
        M=2;DO(35)<ROI1(M)=SPEK(1054+M);M=M+1>
        CONTROL"D00FB";CONTROL"D00JB";W1=$TIME
        TRANSFER(SPEK,1,1024,+1);CONTROL"C00FB";W2=$TIME;CONTROL"C13A@"
        IF(EQ(ROI1(15),2)+EQ(ROI1(15),20)) Q(96)=Q(67)
        IF(EQ(ROI1(15),3)+EQ(ROI1(15),30)) Q(96)=Q(68)
AUSW1:  X=I*K+28;IF(GE(K,16))GOTO AUSEX
        IF(LE(ROI1(X+15),0))GOTO LOPEX
        IF(GE(ROI1(X+1),1010)*EQ(LOOP,0)) LOOP=1;KK=1;GOTO LOPEX
        MAXI=ROI1(X+1)-J;N=ABS(ROI1(X+2))
        M=MAXI-IP(N/2);DO(N)<IF(LE(SPEK(MAXI),SPEK(M))) MAXI=M
        M=M+1>;M=SPEK(MAXI+1);IF(GE(M,SPEK(MAXI-1)))M=SPEK(MAXI-1)
        R=SPEK(MAXI+1)-SPEK(MAXI-1);N=2*(SPEK(MAXI)-M);IF(EQ(N,0)) N=1
        R=R/N;R=R+J+MAXI
        ROI1(X+3)=R;ROI1(X+4)=ROI1(10)*R*R+ROI1(11)*R+ROI1(12)
        NR=1;NL=1
RECHTS: IF(GT(MAXI+NR+1,LKN)) TYPE!,"+++ PEAK",K," RECHTS";GOTO LINKS
        STAT=SPEK(MAXI+NR+1)+2*SQRT(SPEK(MAXI+NR+1))
        IF(GT(SPEK(MAXI+NR),STAT)) NR=NR+1;GOTO RECHTS
        IF(GT(MAXI+NR+2,LKN)) TYPE!,"+++ PEAK",K," RECHTS";GOTO LINKS
        STAT=SPEK(MAXI+NR+2)+2*SQRT(SPEK(MAXI+NR+2))
```

```
        IF(GT(SPEK(MAXI+NR),STAT)) NR=NR+1;GOTO RECHTS
LINKS:  IF(LT(MAXI-NL-1,1)) TYPE!,"+++ PEAK",K," LINKS";GOTO WEIT
        STAT=SPEK(MAXI-NL-1)+2*SQRT(SPEK(MAXI-NL-1))
        IF(GT(SPEK(MAXI-NL),STAT)) NL=NL+1;GOTO LINKS
        IF(LT(MAXI-NL-2,1)) TYPE!,"+++ PEAK",K," LINKS";GOTO WEIT
        STAT=SPEK(MAXI-NL-2)+2*SQRT(SPEK(MAXI-NL-2))
        IF(GT(SPEK(MAXI-NL),STAT)) NL=NL+1;GOTO LINKS
WEIT:   ROI1(X+8)=NR+NL+1
        M=1;N=NR;IF(LE(SPEK(MAXI-NL),SPEK(MAXI+NR))) M=Ø.9;N=-1*NL
        HW=(2*SPEK(MAXI)+SPEK(MAXI-NL)+M*(SPEK(MAXI+N)))/4
        NUEB=Ø;N=Ø;M=NR+NL+1
        DO(M)<STAT=SPEK(MAXI-NL+N)-2*SQRT(SPEK(MAXI-NL+N))
        IF(GT(STAT,HW)) NUEB=NUEB+1
        N=N+1>
        ROI1(X+9)=(NUEB+1)*ROI1(11)
        IF(LE(SPEK(MAXI-NL),SPEK(MAXI+NR))) M=1.9*SPEK(MAXI-NL);GOTO U1
        M=SPEK(MAXI-NL)+SPEK(MAXI+NR)
U1:     ROI1(X+6)=(NL+NR-1)*M/2;N=NR+NL-1;M=MAXI-NL+1;ISUPK=Ø
        DO(N)<ISUPK=ISUPK+SPEK(M);M=M+1>
        DPK=2*SQRT(ISUPK)+N*SQRT(SPEK(MAXI-NL))+N*SQRT(SPEK(MAXI+NR))
        ROI1(X+7)=ISUPK-ROI1(X+6)
        IF(EQ(ROI1(X+7),Ø)) ROI1(X+7)=1
        ROI1(X+10)=DPK*100/ROI1(X+7)
        ROI1(X+12)=ROI1(X+5)*Q(96)
        KONZ=ROI1(X+12)*Q(97)*ABS(ROI1(X+17))*ROI1(1)
        IF(EQ(KONZ,Ø)) KONZ=10↑-90
        ROI1(X+11)=(ROI1(X+7)*3.7*10↑10)/KONZ
        ROI1(X+13)=(DPK*3.7*10↑10)/KONZ
LOPEX:  IF(GT(K,Q(13)))GOTO AUSEX
        IF(EQ(KK,1)) KK=Ø;GOTO AUSW2
        K=K+1
AUSW2:  IF(NE(LOOP,1))GOTO AUSW1
        SPORG(Q(75),2,Ø);ERASE @FL;J=994;LOOP=2;LKN=1054
        CONTROL"EØØFB";CONTROL"EØØJB";W3=$TIME
        TRANSFER(SPEK,31,1024,+1);CONTROL"CØØFB";W4=$TIME
        CONTROL"C13A@";GOTO AUSW1
AUSEX:  ERASE SPEK;DIMENS ROI2(300);$U(3);COPY SAFIL2(1):ROI2
        ERASE SAFIL2;$F(10);TYPE!>
```

PEAVER

PERFORMS PEAK COMPARISION OF THE ARRAYS ROI1 AND ROI2 AND DISPLAYS THE LOG. IF THE DIFFERENCE OF THE ACTIVITY CONCENTRATION EQUALS THE ACTIVITY CONCENTRATION IN ROI1, AN ADDITIONAL MESSAGE IS GIVEN BECAUSE OF CONSIDERABLE CHANGES ALTHOUGH IT IS SURPRESSED IF THE RELATIVE ERROR IS GREATER THAN 100%.

CALL: PEAVER

```
DEFINE PEAVER<;  REMARK  28.9.77         &
        LOCAL I,K,DIF,X,FEH,Z,Y
        DIMENS H(16);K=1;I=17;ROI1(37)=0;ROI1(38)=0
PE0:    X=I*K+28;IF(LT(ROI1(X+17),0))GOTO PE1
        ROI1(37)=ROI1(37)+ROI1(X+7);ROI1(38)=ROI1(38)+ROI1(X+11)
PE1:    IF(LT(K,ROI1(13))) K=K+1;GOTO PE0
        IF(EQ(ROI1(1),0)) ROI1(1)=1
        IF(EQ(ROI2(1),0)) ROI2(1)=1
        I=40;TYPE!;DO(25)<TYPE"-">;TYPE!,!,!
        DO(20)<TYPE" ">;TYPE"MESSPROTOKOLL"
        IF(EQ(ROI1(14),5)) TYPE" (",$I(2)+ROI1(5),". ABKLINGMESSUNG )"
        IF(EQ(ROI1(14),3)+EQ(ROI1(14),7)) TYPE" ( SONDERMESSUNG )"
        IF(NE(DEF(ALAR),1)*NE(DEF(TEXT),1))GOTO PEA0
        TYPE!;DO(21)<TYPE" ">;TYPE"(NACHTRAG)"
PEA0:   TYPE!,!;DATAUS;TYPE!,!
        IF(EQ(Q(37),1)) TYPE"N U L L E I S T U N G",!,!
        $I(4);TYPE"SPEKTRUMS NR. =",ROI1(4),!,!;$I(3)
PEA01:  TYPE"DIE MESSUNG WURDE IN KOLLIMATORSTELLUNG",ROI1(15)
        TYPE" DURCHGEFUEHRT",!,!,"EICHFAKTOREN:",!;$F(10)
        TYPE"A = ",ROI1(10),"   B = ",ROI1(11),"   C = ",ROI1(12),!,!
        TYPE"STARTZEIT = ";UHRAUS(ROI1(9),1)
        IF(NE(DEF(ALAR),1)*NE(DEF(TEXT),1))GOTO PEA02
        $I(2);TYPE"  (VOM ",ROI1(6),".",ROI1(7),".",$I(4)+ROI1(8),")"
PEA02:  TYPE!,!;$F(10)
        TYPE"TRUETIME = ",ROI1(2)," S",!
        TYPE"LIVETIME = ",ROI1(1)," S",!
        TYPE"DEADTIME = ",ROI1(3)," S",!,!
        TYPE"WAEHREND DER AUFNAHME DES SPEKTRUMS WURDEN FOLGENDE"
        TYPE" WERTE GEMESSEN",!,!
        DO(19)<TYPE" ">;TYPE"MITTEL    START     ENDE      "
        TYPE"MAXIMAL   MINIMAL",!
        TYPE"REAKTORLEISTUNG   ";$F(6);K=17
        DO(5)<TYPE ROI1(K),"     ";K=K+1>;TYPE"MW",!,"PRAEZIP-ZAEHLR. "
        $I(8);DO(5)<TYPE ROI1(K),"  ";K=K+1>;TYPE" IMP/S"
        TYPE!,!,!;IF(EQ(Q(65),0)*EQ(Q(75),1))GOTO PEA04
        IF(EQ(Q(65),0)*NE(Q(44),1))GOTO PEA04
        IF(EQ(DEF(ABN),1))GOTO PEA05
        GOTO PEA03
PEA05:  IF(EQ(ABN,0))GOTO PEA03
PEA04:  TYPE"KEIN PEAKVERGLEICH, DA ERSTES SPEKTRUM.",!,!;GOTO PEA4
PEA03:  DO(20)<TYPE" ">;TYPE"PEAKVERGLEICH",!,!
        $I(4);TYPE"DES SPEKTR.NR.",ROI1(4);$I(2);TYPE" VOM "
        TYPE ROI1(6),".",ROI1(7),".",$I(4)+ROI1(8),"  STARTZEIT "
```

```
        UHRAUS(ROI1(9),1);$I(4);TYPE!,"MIT SPEKTR.NR.",ROI2(4);$I(2)
        TYPE" VOM ",ROI2(6),".",ROI2(7),".",$I(4)+ROI2(8)
        TYPE"  STARTZEIT ";UHRAUS(ROI2(9),1);$I(4)
        TYPE!,!,"I";DO(70)<TYPE"-">;TYPE"I",!,"I";DO(13)<TYPE" ">
        TYPE"I  MESSWERTE ZU NR.",ROI1(4)," I DIFFERENZEN NR."
        TYPE ROI1(4)," - NR.",ROI2(4)," I",!,"I";DO(70)<TYPE"-">
        TYPE"I",!,"I NR.I NUCLID I  AKTIV.KONZ.  I REL.FEHI"
        TYPE"  AKTIV.KONZ.  I FEHLER (ABS.) I",!,"I    I";DO(8)<TYPE" ">
        TYPE"I   (1/S*CCM)   I  (%)   I   (1/S*CCM)   I   (1/S*CCM)   I"
        TYPE!,"I";DO(70)<TYPE"-">;TYPE"I"
        K=1;I=17
PEA2:   X=I*K+28;IF(LE(ROI1(X+1),0))GOTO PEA5
        $I(2);TYPE!,"I ",K," I ";TYPES ROI1(X+14)
        $E;TYPE" I",ROI1(X+11)," I ";FORM(ROI1(X+10),3,2)
        TYPE" I";$E;TYPE ROI1(X+11)-ROI2(X+11)," I"
        FEH=SQRT(ROI1(X+13)*ROI1(X+13)+ROI2(X+13)*ROI2(X+13))
        TYPE FEH," I"
        DIF=ROI1(X+11)-ROI2(X+11)
        IF(GE(DIF,0)) Z=ROI2(X+11)
        IF(LT(DIF,0)) Z=ROI1(X+11)
        IF(LT(ABS(DIF),Z))GOTO PEA5
        IF(GE(ROI1(X+10),100)*GE(ROI2(X+10),100))GOTO PEA5
        Y=ROI2(X+11);IF(EQ(Y,0)) Y=1
        H(K)=ROI1(X+11)/Y
PEA5:   IF(LT(K,ROI1(13))) K=K+1;GOTO PEA2
        $E;TYPE!,"I    SUMME    I",ROI1(38)," I          I"
        TYPE ROI1(38)-ROI2(38)," I                  I",!,"I"
PEA3:   DO(70)<TYPE"-">;TYPE"I"
        K=1;I=17;TYPE!,!,"UEBER FOLGENDE FENSTER WURDE SUMMIERT:",!
        DO(ROI1(13))<X=I*K+28;IF(GT(ROI1(X+17),0)) TYPE $I(2)+K,"; "
        K=K+1>;TYPE!,!,!
        K=1;DO(ROI1(13))<IF(NE(H(K),0)) H(16)=1
        K=K+1>
        IF(NE(H(16),1))GOTO PEA4
        DO(25)<TYPE"=">;TYPE" ACHTUNG ";DO(25)<TYPE"=">
        TYPE!,"BEI FOLGENDEN PEAKS TRATEN GROSSE AENDERUNGEN AUF",!
        TYPE!,"NR.    AENDERUNGSFAKTOR"
        K=1;DO(ROI1(13))<IF(EQ(H(K),0))GOTO PEA6
        TYPE!,$I(2)+K,"       ";$E;TYPE H(K)
PEA6:   K=K+1>;TYPE!,!,!
PEA4:   DO(25)<TYPE"-">
        $F(10);TYPE!,!;ERASE H>
```

MESPRO

PRINTS OUT THE MEASURING AND CALCULATED VALUES OF THE ARRAY ROI1.

CALL: MESPRO

```
DEFINE MESPRO<; REMARK  26.8.77 &
        LOCAL I,K,X,Y
LØ:     TYPE!,!;DO(25)<TYPE"-">;TYPE!,!
        $I(4);TYPE"      MESSDATEN ZU SPEKTR.NR.",ROI1(4),"   VOM "
        $I(2);TYPE ROI1(6),".",ROI1(7),".",$I(4)+ROI1(8),!,!
        TYPE"STARTZEIT = ";UHRAUS(ROI1(9),1)
        TYPE!,!,"TRUETIME = ",ROI1(2)," S",!
        TYPE"LIVETIME = ",ROI1(1)," S",!
        TYPE"DEADTIME = ",ROI1(3)," S",!,!
        K=1;I=17;TYPE"UEBER FOLGENDE FENSTER WURDE SUMMIERT:",!
        DO(ROI1(13))<X=I*K+28;IF(GT(ROI1(X+17),Ø))TYPE $I(2)+K,"; "
        K=K+1>;TYPE!,!;$I(1Ø)
        TYPE"SUMME PEAKINHALT =",ROI1(37)," IMPULSE",!
        $E;TYPE"SUMME AKTI.KONZ. =",ROI1(38)," 1/S*CCM",!,!
        TYPE"I";DO(68)<TYPE"-">;TYPE"I";TYPE!,"I";DO(24)<TYPE" ">
        I=17;K=1
        TYPE"AUSGEWERTETE FENSTER";DO(24)<TYPE" ">;TYPE"I",!
        TYPE"I";DO(68)<TYPE"-">;TYPE"I",!
        TYPE"I NR.I GENAUE  I ENERGIE I  UNTER-  I  SIGNAL  I KZ."
        TYPE"I  HWB  I REL.   I",!
        TYPE"I     I KANAL-  I  (KEV)  I  GRUND  I  (IMP.)  I    "
        TYPE"I (KEV) I FEHLER I",!
        TYPE"I     I LAGE    I         I  (IMP.)  I          I    "
        TYPE"I         I  (%)   I",!
        TYPE"I";DO(68)<TYPE"-">;TYPE"I",!
L1:     X=K*I+28;IF(LE(ROI1(X+15),Ø))GOTO L3
        $I(2);TYPE"I ",K," I "
        FORM(ROI1(X+3),4,2);TYPE" I "
        FORM(ROI1(X+4),4,2);TYPE" I "
        $I(8);TYPE ROI1(X+6)," I ",ROI1(X+7)," I "
        $I(2);TYPE ROI1(X+8)," I "
        FORM(ROI1(X+9),2,2);TYPE" I "
        FORM(ROI1(X+1Ø),3,2);TYPE" I",!
L3:     IF(LT(K,ROI1(13))) K=K+1;GOTO L1
        TYPE"I";DO(68)<TYPE"-">;TYPE"I",!,!,"I";DO(57)<TYPE"-">
        TYPE"I";I=17;K=1
        TYPE!,"I FORTS.AUSW.FENSTER I";DO(12)<TYPE" ">
        TYPE"TABELLENWERTE";DO(11)<TYPE" ">;TYPE"I"
        TYPE!,"I";DO(2Ø)<TYPE"-">;TYPE"I";DO(36)<TYPE"-">;TYPE"I",!
        TYPE"I NR.I  AKTIV.KONZ.  I NUCLID I  ENERGIE I   HWZ  I"
        TYPE" ABUND.I",!
        TYPE"I     I    (1/S*CCM)    I         I  (KEV)    I        I"
        TYPE"  (%)  I",!,"I"
        DO(2Ø)<TYPE"-">;TYPE"I";DO(36)<TYPE"-">;TYPE"I",!
L4:     X=K*I+28;IF(LE(ROI1(X+15),Ø))GOTO L6
        $I(2);TYPE"I ",K," I";$E;TYPE ROI1(X+11)," I "
        TYPES ROI1(X+14)," I ";FORM(ROI1(X+15),4,3);Y=ABS(ROI1(X+17))
        TYPES" I ",ROI1(X+16)," I ";FORM(Y,2,2);TYPE" I",!
L6:     IF(LT(K,Q(13))) K=K+1;GOTO L4
        $F(1Ø);TYPE"I";DO(57)<TYPE"-">;TYPE"I",!,!
        DO(25)<TYPE"-">;TYPE!,!
L5:     X=Ø>
```

FENAUS

READS OUT 3 WINDOWS SET IN THE MCA (CALIBRATION PEAKS) INTO THE ARRAY EIF.

CALL: FENAUS

```
DEFINE FENAUS<; REMARK  29.6.77          &
        LOCAL RLOW,ANZ,RHIGH,I,KANAL,MREG,KNZ,KK,A,B
        I=46
        ANZ=Ø
        KANAL=-1
        MREG=1Ø24
        CONTROL"BØØFA";A=$TIME;CONTROL"BØØIC"
FROI:   RLOW=LOW
        IF(LT(RLOW,Ø))GOTO FEND
        KANAL=KANAL+RLOW+1
        EIF(I)=KANAL+1
        RHIGH=HIGH
        KANAL=KANAL+RHIGH+1
        EIF(I+2)=RHIGH+1
        EIF(I+1)=KANAL
        I=I+51
        ANZ=ANZ+1
        IF(LT(ANZ,3))GOTO FROI
FEND:   CONTROL"BØØFB"
        IF(EQ(ANZ,Ø))GOTO LROI1
        CONTROL"BØØIA"
        I=49
        KNZ=EIF(48)+1
        TRANSFER(EIF,I,KNZ,-1)
        EIF(1)=EIF(49)
        KNZ=KNZ-1
        DO(KNZ)<I=I+1;EIF(I-1)=EIF(I);EIF(I)=Ø>
        IF(EQ(ANZ,1))GOTO LROI2
        I=1ØØ;KK=1
        DO(ANZ-1)<KNZ=EIF(I-1);TRANSFER(EIF,I,KNZ,-1)
        KK=KK+1;I=I+51>
        GOTO LROI2
LROI1:  TYPE!,"KEINE FENSTER GESETZT !!!"
LROI2:  CONTROL"BØØFB";CONTROL"C13A@";B=$TIME;EIF(15)=Q(15)
        IF(GE(EIF(1),Q(2))) Q(2)=EIF(1)
        Q(3)=Q(2)-EIF(1);Q(8Ø)=SEC(B)-SEC(A)
        I=17;DO(5)<Q(I)=Q(I)*Q(64);I=I+1>
        I=1;DO(35)<I=I+1;EIF(I)=Q(I)>;EIF(9)=Q(42)>
```

EIKOF

CALCULATES THE CALIBRATION COEFFICIENTS FOR THE QUADRATIC CALIBRATION AND SETS THE ALARM FLAG Q(60) IN CASE THE QUADRATIC CALIBRATION FACTOR GETS TOO GREAT.
Q(60)=0, CALIBRATION FACTOR GOOD
Q(60)=1, CALIBRATION FACTOR>=0.0001
Q(60)=2, CALIBRATION FACTOR>=0.00001
BESIDES, NEW WINDOW ADDRESSES ARE CALCULATED FOR THE PEAKS TO BE EVALUATED.

CALL: EIKOF

```
DEFINE EIKOF<;  REMARK  29.9.77          &
        LOCAL X,R,Y,Z,R1,R2,R3,E1,E2,E3,I,K,N,M,MAXI,FEAN,KMAX,GZ,VZ
        K=0;DO(3)<EIF(43+K)=EIF(10+K);K=K+1>;K=0
EIC1:   X=45+51*K
        MAXI=X+4
        N=EIF(X+3)
        FEAN=X+1
        M=MAXI
        DO(N)<IF(LE(EIF(MAXI),EIF(M))) MAXI=M
        M=M+1>
        N=MAXI-FEAN-3
        KMAX=EIF(FEAN)+N
        M=EIF(MAXI+1)
        IF(GE(M,EIF(MAXI-1)))M=EIF(MAXI-1)
        R=EIF(MAXI+1)-EIF(MAXI-1)
        N=2*(EIF(MAXI)-M)
        IF(EQ(N,0)) N=1
        R=R/N;R=R+KMAX
        EIF(X+50)=R;K=K+1
        IF(LT(K,3))GOTO EIC1
        E1=EIF(96);E2=EIF(147);E3=EIF(198)
        R1=EIF(95);R2=EIF(146);R3=EIF(197)
        X=E3-E1;Y=R2-R1;X=X*Y
        Z=E2-E1;Y=R1-R3;Z=Z*Y;Z=Z+X
        X=R3*R3-R1*R1;Y=R2-R1;X=X*Y
        N=R2*R2-R1*R1;Y=R1-R3;N=N*Y;N=N+X
        IF(EQ(N,0)) N=1*10↑-99
        EIF(10)=Z/N
        X=R2*R2-R1*R1;X=EIF(10)*X
        Z=E2-E1-X;N=R2-R1
        IF(EQ(N,0)) N=1*10↑-99
        EIF(11)=Z/N
        EIF(12)=E1-EIF(10)*R1*R1-EIF(11)*R1
        I=0;DO(3)<Q(10+I)=EIF(10+I);I=I+1>;Q(60)=0
        IF(GE(ABS(Q(10)),0.001)) Q(60)=2
        IF(GE(ABS(Q(10)),0.0001)) Q(60)=1
        R=17;K=1
EIC2:   X=R*K+28;GZ=0
        IF(LE(ROI1(X+15),0))GOTO EIC3
        IF(EQ(EIF(10),0)) EIF(10)=1*10↑-99
        Z=EIF(11)/(2*EIF(10));Y=EIF(12)-ROI1(X+15);Y=Y/EIF(10)
        N=Z*Z-Y;IF(GE(N,0))GOTO EIC5
        $I(3);TYPE!,"+++ WURZEL NEGATIV,PEAK ",K,!;N=N*-1
```

```
EIC5:    N=SQRT(N);Y=-Z-N
         IF(LE(Y,Ø)+GE(Y,2048)) Y=-Z+N
         Y=IP(Y);IF(GT(Y,Ø)*LE(Y,2048)) ROI1(X+1)=Y;GOTO EIC6
         TYPE!,"+++ FENSTER NR. ",$I(2)+K," AUSSERHALB, KEINE "
         TYPE"AENDERUNG",!;GOTO EIC3
EIC6:    VZ=1;IF(LT(ROI1(X+2),Ø)) VZ=-1
         IF(GT(ROI1(X+1)+ABS(ROI1(X+2)/2),2048)) GZ=1
         IF(LT(ROI1(X+1)-ABS(ROI1(X+2)/2),1)) GZ=2
         IF(EQ(GZ,1)) ROI1(X+2)=2*VZ*(2048-ROI1(X+1))
         IF(EQ(GZ,2)) ROI1(X+2)=2*VZ*(ROI1(X+1)-1)
         IF(EQ(GZ,Ø))GOTO EIC3
         TYPE!,"+++ FENSTER NR. ",$I(2)+K," NUR NOCH ",ROI1(X+2)
         TYPE" KANAELE BREIT",!
EIC3:    IF(GE(K,Q(13)))GOTO EIC4
         K=K+1;GOTO EIC2
EIC4:    Q(16)=Q(4)+1;$F(1Ø)>
```

EIPRO

CAUSES LOG PRINTOUT FOR CALIBRATION MEASUREMENT, POSSIBLY ALARM MESSAGE AS A FUNCTION OF Q(61).

CALL: EIPRO

```
DEFINE EIPRO<;  REMARK  28.9.77          &
        LOCAL I,K
        TYPE!;DO(25)<TYPE"-">;TYPE!,!,!
        DO(20)<TYPE" ">;TYPE"EICHPROTOKOLL"
        IF(NE(DEF(ALAR),1)*NE(DEF(TEXT),1))GOTO EIP02
        TYPE!;DO(21)<TYPE" ">;TYPE"(NACHTRAG)"
EIP02:  TYPE!,!;DATAUS;TYPE!,!
        IF(EQ(Q(37),1)) TYPE"N U L L E I S T U N G",!,!
        $I(4);TYPE"SPEKTRUMS NR. =",EIF(4),"        "
        TYPE"EICHUNG NR. =",EIF(5),!,!;$I(3)
        TYPE"DIE EICHUNG WURDE IN KOLLIMATORSTELLUNG",EIF(200);$F(10)
        TYPE" DURCHGEFUEHRT",!,!,"EICHPEAKS (QUADRATISCHE EICHUNG)"
        I=1;K=0;DO(3)<$I(2);TYPE!,"PEAK",I,"       "
        FORM(EIF(96+K),4,2);TYPE"  KEV  ;  "
        FORM(EIF(95+K),4,2);TYPE"  KANAL";K=K+51;I=I+1>;TYPE!,!;$F(10)
        TYPE"STARTZEIT = ";UHRAUS(EIF(9),1)
        IF(NE(DEF(ALAR),1)*NE(DEF(TEXT),1))GOTO EIP03
        $I(2);TYPE"  (VOM ",EIF(6),".",EIF(7),".",$I(2)+EIF(8),")"
EIP03:  TYPE!,!;$F(10)
        TYPE"TRUETIME = ",EIF(2)," S",!
        TYPE"LIVETIME = ",EIF(1)," S",!
        TYPE"DEADTIME = ",EIF(3)," S",!,!
        TYPE"WAEHREND DER AUFNAHME DES SPEKTRUMS WURDEN FOLGENDE"
        TYPE" WERTE GEMESSEN",!,!
        DO(19)<TYPE" ">;TYPE"MITTEL    START     ENDE      "
        TYPE"MAXIMAL   MINIMAL",!
        TYPE"REAKTORLEISTUNG   ";$F(6);K=17
        DO(5)<TYPE EIF(K),"     ";K=K+1>;TYPE"MW",!,"PRAEZIP-ZAEHLR. "
        $I(8);DO(5)<TYPE EIF(K),"  ";K=K+1>;TYPE" IMP/S",!,!,!;$I(4)
        TYPE"DIE NEUEN EICHFAKTOREN GELTEN AB SPEKTRUM NR. ",Q(16)
        TYPE!,!;$F(10)
        TYPE"EICHFAKTOREN:",!,!
        IF(EQ(EIF(5),1))GOTO EIP01
        TYPE"ALT:    A = ",EIF(43),"   B = ",EIF(44),"    C = "
        TYPE EIF(45),!
EIP01:  TYPE"NEU:    A = ",EIF(10),"   B = ",EIF(11),"    C = "
        TYPE EIF(12)
        IF(EQ(Q(60),1))GOTO EIP1
        IF(EQ(Q(60),2))GOTO EIP2
        GOTO EIP3
EIP1:   TYPE!,!,!,"        A C H T U N G"
        TYPE!,!,"EICHFAKTOR A BEDENKLICH GROSS !!!"
        GOTO EIP3
EIP2:   TYPE!,!;DO(2)<TYPE!,"A L A R M ";DO(15)<TYPE "-">
        TYPE " A L A R M ";DO(15)<TYPE "-">;TYPE " A L A R M">
        TYPE!,!,"EICHFAKTOR A ZU GROSS !!!"
EIP3:   TYPE!,!,!;DO(25)<TYPE"-">;TYPE!,!;$F(10)>
```

COUNT

STARTS COUNTER/TIMER

CALL: COUNT(A,B)

A= COUNTER ADDRESS
B= COUNTING TIME IN SECONDS (MIN. 10 S)

WRITES RESULT OF ADDRESS 41 TO Q(50) (PRECIPITATOR COUNTING RATE) AND RESULT OF ADDRESS 42 TO Q(51) (REACTOR POWER)

```
DEFINE COUNT<;  REMARK  21.10.76         &
        LOCAL UHR,UHRH,UHRM,UHRS,CHAN,COUZEI
        CHAN=ARG(1)
        COUZEI=ARG(2)*10
        ACT(CHAN)1
        ACT(CHAN)2
        UHR=$TIME+COUZEI
        UHRS=IP(UHR/1000)*1000
        UHRS=UHR-UHRS
        UHRH=IP(UHR/100000)
        UHRM=UHRH*100000
        UHRM=UHR-UHRM-UHRS
        UHRM=UHRM/1000
        IF(LT(UHRS,600))GOTO CL1
        UHRS=UHRS-600
        UHRM=UHRM+1
CL1:    IF(LT(UHRM,60))GOTO CL2
        UHRM=UHRM-60
        UHRH=UHRH+1
CL2:    $WTIME(UHRH,UHRM,UHRS)
        ACT(CHAN)0
        IF(EQ(CHAN,40))GOTO CL4
        IF(EQ(CHAN,42))GOTO CL3
        READOUT(CHAN)Q(50);Q(51)=0
        Q(50)=Q(50)*10/COUZEI
        GOTO CL5
CL3:    READOUT(CHAN)Q(51);Q(50)=0
        Q(51)=Q(51)*10/COUZEI
        GOTO CL5
CL4:    READOUT(41)Q(50);READOUT(42)Q(51)
        Q(50)=Q(50)*10/COUZEI
        Q(51)=Q(51)*10/COUZEI
CL5:    UHR=UHR>
```

RLCPS

FOR MEASUREMENT OF RL AND PCPS. FORMS AN AVERAGE VALUE, RETAINS THE INITAL, FINAL, MAXIMUM AND MINIMUM VALUES DURING THE MEASUREMENT AND SETS THE EVENT FLAG Q(44) AS A FUNCTION OF THE MEASURED RESULTS OF RL AND PCPS.

CALL: Q(53)=A;RLCPS(Q(61),B)

A=Ø, DELETE ALL VALUES
B=Ø, RETAIN MAXIMUM AND MINIMUM VALUES, COTINUE AVERAGING
B=1, SET INITIAL VALUE
B=2, SET FINAL VALUE

FLAG Q(44)
=Ø, NO EVENT
=1, RL<Q(61), START DECAY MEASUREMENT
=2, RL>Q(83), START SPECIAL MEASUREMENT
=3, PCPS>Q(84), START SPECIAL MEASUREMENT

```
DEFINE RLCPS<;  REMARK  4.8.77  &
         LOCAL RLH,CPSH,PAU,J,L
         IF(EQ(Q(53),Ø)) PAU=17;DO(1Ø)<Q(PAU)=Ø;PAU=PAU+1>
         COUNT(4Ø,1Ø)
         IF(EQ(ARG(2),Ø))GOTO LPAU1
         IF(NE(ARG(2),1)) Q(19)=Q(51);Q(24)=Q(5Ø);GOTO LPAU1
         J=18;L=23;DO(4)<Q(J)=Q(51);Q(L)=Q(5Ø);J=J+1;L=L+1>
LPAU1:   IF(GT(Q(51),Q(2Ø))) Q(2Ø)=Q(51)
         IF(LT(Q(51),Q(21))) Q(21)=Q(51)
         IF(GT(Q(5Ø),Q(25))) Q(25)=Q(5Ø)
         IF(LT(Q(5Ø),Q(26))) Q(26)=Q(5Ø)
         CPSH=Q(5Ø)
         RLH=Q(51)
         Q(22)=(Q(22)*Q(53)+CPSH)/(Q(53)+1)
         Q(17)=(Q(17)*Q(53)+RLH)/(Q(53)+1)
         IF(LT(RLH,ARG(1))*EQ(Q(37),Ø)) Q(72)=$TIME;Q(44)=1;GOTO LPAU4
         PAU=Q(85)-RLH;IF(EQ(Q(85),Ø)) Q(85)=1
         PAU=PAU*1ØØ/Q(85)
         IF(GT(ABS(PAU),Q(83))) Q(72)=$TIME;Q(44)=2;GOTO LPAU4
         PAU=Q(86)-CPSH;IF(EQ(Q(86),Ø)) Q(86)=1
         PAU=PAU*1ØØ/Q(86)
         IF(GT(ABS(PAU),Q(84))) Q(72)=$TIME;Q(44)=3;GOTO LPAU4
         Q(44)=Ø
LPAU4:   Q(53)=Q(53)+1>
```

FAHR

POSITIONS THE COLLIMATOR, Q(15) POINTS TO THE TARGET.

CALL: FAHR(A)

A=Ø, MOVES THE COLLIMATOR AS A FUNCTION OF THE INTEGRAL COUNTING RATE TO POSITION 1,2 OR 3 AND DEFINES THE MEASURING TIME FOR THE MCA.
A=1,2 OR 3; MOVES COLLIMATOR IN THE RESPECTIVE POSITION.

TIME OF ARRIVAL IS ENTERED IN Q(45); Q(45)=-1 IF COLLIMATOR IS ALREADY IN TARGET POSITION; Q(45) IS CHECKED IN MCACOL.

```
DEFINE FAHR<;   REMARK  18.8.77 &
        LOCAL ZIEL
        Q(1)=Q(39);ZIEL=ARG(1);ACT(81)2
        IF(NE(ZIEL,Ø))GOTO FA1
        ZIEL=2;COUNT(4Ø,1Ø);Q(5Ø)=Q(5Ø)+2*SQRT(Q(5Ø))
        IF(LT(Q(5Ø),Q(56))) ZIEL=1
        IF(LT(Q(5Ø),Ø.2*Q(56))) Q(1)=Q(4Ø)
        IF(GT(Q(5Ø),Q(57))) ZIEL=3
        IF(GT(Q(5Ø),5*Q(57))) Q(1)=Q(38)
FA1:    IF(NE(Q(15),ZIEL)+LE(Q(45),-3))GOTO FA2
        Q(45)=-1;GOTO FA3
FA2:    IF(EQ(ZIEL,1)) ACT(81)9
        IF(EQ(ZIEL,2)) ACT(81)8
        IF(EQ(ZIEL,3)) ACT(81)8;ACT(81)9
        TYPE!,"+++ KOLLIMATOR FAEHRT"
        Q(45)=SEC($TIME)+12Ø;Q(45)=TIME(Q(45))
FA3:    $F(1Ø);Q(15)=ZIEL>
```

MCACOL

STARTS THE MCA AND CHECKS VIA Q(45) WHETHER COLLIMATOR HAS BEEN CORRECTLY POSITIONED. IF AN ERROR OCCURS IN THE COLLIMATOR, Q(15)=Q(15)*10.

CALL: MCACOL(A,B)

A= MEASURING TIME IN SECONDS
B=0, NO START, CHECK COLLIMATOR ONLY
B=1, CHECK COLLIMATOR AND START

FLAG Q(45)
>=0, COLLIMATOR MOVES, CHECK MOVING TIME
=-1, COLLIMATOR AT TARGET
=-2, COLLIMATOR AT TARGET, MEASUREMENT STARTED
=-3, COLLIMATOR DEFECTED, MESSAGE
=-4, COLLIMATOR DEFECTED, MEASUREMENT STARTED, MESSAGE

THE TYPE OF MEASUREMENT STARTED IS PRINTED OUT AS A FUNCTION OF Q(44); THE MCA HALF IS SELECTED VIA Q(43).

```
DEFINE MCACOL<; REMARK  12.9.77 &
        LOCAL X,Y
        IF(EQ(Q(44),1)*LE(Q(45),-3))GOTO MC2
        IF(EQ(Q(45),-1)*EQ(ARG(2),0))GOTO MC04
        IF(EQ(Q(45),-1)) Q(45)=-2;GOTO MC2
MC1:    IF(GE(Q(45),$TIME)*EQ(SENSE(3),0))GOTO MC1
        IF(EQ(SENSE(3),1)*EQ(ARG(2),0)) Q(45)=-1;GOTO MC04
        IF(EQ(SENSE(3),0)*EQ(ARG(2),0)) Q(45)=-3;GOTO MC4
        IF(EQ(SENSE(3),0)) Q(45)=-4;GOTO MC2
        Q(45)=-2
MC2:    Y=ARG(1);X=0;IF(NE(Q(44),1)) Q(43)=2
MC3:    IF(GE(Y,10)) Y=Y/10;X=X+1;GOTO MC3
        Y=IP(Y);CONTROL Q(43),"00FB";CONTROL Q(43),"00KB";WAITD
        CONTROL Q(43),Y,X,"A@";Q(9)=$TIME;Q(100)=-1
        IF(LE(Q(45),-3))GOTO MC4
MC04:   IF(GE(Q(15),10)) Q(15)=Q(15)/10
        TYPE!,"+++ KOLLIMATOR IN STELLUNG ",$I(1)+Q(15);GOTO MC5
MC4:    IF(GE(Q(15),10)) Q(15)=Q(15)/10
        TYPE!,!;DO(20)<TYPE"=">;TYPE" KOLLIMATOR DEFEKT "
        DO(20)<TYPE"=">;TYPE!,"+++ KOLLIMATOR NICHT IN STELLUNG "
        TYPE $I(1)+Q(15)," ANGEKOMMEN !",!;Q(15)=Q(15)*10
MC5:    IF(EQ(ARG(2),0))GOTO MC7
        IF(EQ(DEF(SO),1)) ERASE SO;TYPE!,"+++ SONDERMESSUNG";GOTO MC6
        IF(EQ(Q(14),2)*EQ(Q(59),0)) TYPE!,"+++ EICHMESSUNG";GOTO MC6
        IF(EQ(Q(44),0)) TYPE!,"+++ NORMALMESSUNG"
        IF(EQ(Q(44),1)) TYPE!,"+++ ",$I(2)+Q(48)+1,". ABKLINGMESSUNG"
        IF(GT(Q(44),1)) TYPE!,"+++ SONDERMESSUNG"
MC6:    TYPE", MESSZEITVORGABE =",$I(5)+ARG(1)," S; "
        TYPE $I(1)+Q(43)-1,". HAELFTE VKA",!
MC7:    $F(10);TYPE!>
```

MCLES

WRITES THE DATE, SPECTRUM NUMBER, STARTING TIME, INITIAL, FINAL AVERAGE, MAXIMUM AND MINIMUM VALUES OF RL AND PCPS, AS WELL AS THE CALIBRATION FACTORS IN THE FIRST 20 CHANNELS OF THE MCA. THEN THE SPECTRUM IS WRITTEN ON TAPE. SUBSEQUENTLY THE SPECTRUM IS PUT ON CASSETTE FOR LATER EVALUATION BY STEPS OF 1K EACH. READING TIME OF THE MCA IS PUT INTO THE VARIABLE Q(80).

CALL: MCLES

```
DEFINE MCLES<;  REMARK  16.8.77 &
        LOCAL X,Y,T,FL
        SPEK(1056)=SEC(Q(63))-SEC(SPEK(1063))-Q(80)
        DIMENS W(19);DIMENS H(19)
        Y=Q(76);X=0
MCL0:   IF(GE(Y,10)) Y=Y/10;X=X+1;GOTO MCL0
        Y=IP(Y);CONTROL Q(71),"00FB"
        A=$TIME;CONTROL Q(71),"00IB";TRANSFER(H,1,19,-1)
        IF(GE(H(1),SPEK(1056))) SPEK(1056)=H(1)
        SPEK(1057)=SPEK(1056)-H(1)
        T=2;DO(14)<W(T)=SPEK(1054+T);T=T+1>;W(1)=H(1);W(11)=W(11)*10↑4
        W(10)=W(10)*10↑8;IF(LT(W(10),0)) W(10)=W(10)*-1;W(18)=1
        W(12)=W(12)*10↑4;IF(LT(W(12),0)) W(12)=W(12)*-1;W(19)=1
        W(16)=SPEK(1071)*1000;W(17)=SPEK(1076);W(9)=W(9)/10
        CONTROL Q(71),"00FB";CONTROL Q(71),"00JB";TRANSFER(W,1,19,+1)
        LES=1;Z=0;CONTROL Q(71),"00FB";CONTROL Q(71),"00HB";WAITD
        TYPE"+++ ",$I(2)+Q(48),". ABKLINGMESSUNG AUF TAPE, "
        TYPE"SPEKTRUM NR. ",$I(4)+Q(4)," / BAND NR. ";TYPES Q(99),!
        CONTROL Q(71),"00FB";CONTROL Q(71),"00JB";TRANSFER(H,1,19,+1)
        $F(10);CONTROL Q(71),"00FB";ERASE H,W
        IF(GT(Q(48),3)*NE(Q(48),6)*NE(Q(48),7)) Z=0;GOTO MCL1
        CONTROL Q(43),Y,X,"A@";B=$TIME;Z=SEC(B)-SEC(A)
MCL1:   IF(EQ(Q(71),2)) ALT=4;GOTO MCL2
        ALT=6
MCL2:   CONTROL ALT,"00FB";CONTROL ALT,"00IB";A=$TIME
        TRANSFER(SPEK,LES,1024,-1);CONTROL Q(43),"00FB"
        IF(GT(Q(48),3)*NE(Q(48),6)*NE(Q(48),7))GOTO MCL3
        CONTROL Q(43),Y,X,"A@";B=$TIME;Z=Z+SEC(B)-SEC(A)
MCL3:   IF(NE(LES,1))GOTO MCL4
        SPORG(Q(48),1,1)
        A=1;DO(30)<SPEK(A)=SPEK(A+994);A=A+1>;ALT=ALT+1
        LES=31;GOTO MCL2
MCL4:   SPORG(Q(48),2,1);ERASE @FL>
```

TAPE

WRITES THE 2K-SPECTRUM ON TAPE

CALL: TAPE(A)

A=Ø, NORMAL ROUTINE, WRITES IN THE FIRST 2Ø CHANNELS OF THE MCA THE DATE, INITIAL, FINAL, AVERAGE, MAXIMUM AND MINIMUM VALUES OF RL AND PCPS AS WELL AS THE CALIBRATION FACTORS. THEN THE SPECTRUM IS WRITTEN ON TAPE. READING TIME OF THE MCA TO Q(8Ø).

A=1 OR 2, IS USED IN CASE OF PROGRAM INTERRUPTION; PUTS ADDITIONAL SPECTRUM ON TAPE, PREVENTS THE ADVANCEMENT OF DATE, IF NECESSARY, STARTS THE MCA FOR THE NEXT MEASUREMENT. A=1 IS USED IN THE ROUTINES NORMES, SONDER AND EICHEN AND, DEPENDING ON THE SEQUENCE INDEX, IT CAUSES WRITING OF THE INTERRUPTED MEASUREMENT ON THE ALAR1 TO ALAR5 ALARM FILES.

```
DEFINE TAPE<;    REMARK  5.1Ø.77  &
        IF(EQ(ARG(1),1)) Q(88)=Q(72);Q(1ØØ)=-1;GOTO TA3
        IF(NE(ARG(1),Ø)) Q(88)=Q(72);Q(1ØØ)=-1;GOTO TA5
        DIMENS H(19);DIMENS W(19);LOCAL I,A,B
        CONTROL"BØØFB";A=$TIME;CONTROL"BØØIB";TRANSFER(H,1,19,-1)
        Q(4)=Q(4)+1;IF(EQ(DEF(EIF),1)) Q(5)=Q(5)+1
        Q(2)=SEC(Q(63))-SEC(Q(42))
        I=1;DO(15)<W(I)=Q(I);I=I+1>;W(1)=H(1);W(11)=W(11)*1Ø↑4
        IF(LE(Q(2),H(1))) Q(2)=H(1)
        W(3)=Q(2)-H(1)
        W(1Ø)=W(1Ø)*1Ø↑3;IF(LT(W(1Ø),Ø)) W(1Ø)=W(1Ø)*-1;W(18)=1
        W(12)=W(12)*1Ø↑4;IF(LT(W(12),Ø)) W(12)=W(12)*-1;W(19)=1
        W(16)=Q(17)*Q(64)*1ØØØ;W(17)=Q(22);W(9)=W(9)/1Ø
        CONTROL"BØØFB";CONTROL"BØØJB";TRANSFER(W,1,19,+1)
        CONTROL"BØØFB";CONTROL"BØØHB";WAITD
        CONTROL"BØØFB";CONTROL"BØØJB";TRANSFER(H,1,19,+1);CONTROL"BØØFB"
        CONTROL"C13A@";B=$TIME;Q(8Ø)=SEC(B)-SEC(A)
        $I(4);IF(EQ(DEF(EIF),1))GOTO TA1
        TYPE!,!,"+++ SPEKTRUM NR.",W(4);GOTO TA2
TA1:    TYPE!,!,"+++ EICHSPEKTRUM NR.",W(4)," ;",W(5)
TA2:    TYPES" AUF TAPE / BAND NR. ",Q(99),!
        $F(1Ø);ERASE H,W;GOTO TAEND
TA3:    MES1=Q(44);IF(EQ(K,59)*LE(Q(K),1))GOTO TA6
        IF(EQ(K,59))GOTO TA5
        IF(NE(Q(K),1))GOTO TA5
TA4:    CONTROL"BØØFB";Q(63)=$TIME;CONTROL"BØØHB";WAITD
        Z=SEC(Q(63))-SEC(Q(9));TYPE!,"+++ SPEKTRUM AUF TAPE / "
        TYPES"BAND NR. ",Q(99),!,"+++ KEINE"
        TYPE" BEARBEITUNG,MESSZEIT= ",Z," S",!,"+++ STARTZEIT "
        UHRAUS(Q(9),1);TYPE!;GOTO TA6
TA5:    Q(4)=Q(4)+1;CONTROL"CØØFB";Z=$TIME;CONTROL"CØØHB";WAITD
TA6:    IF(EQ(Q(44),1)) Q(48)=Ø;Q(43)=3;FAHR(Ø);MCACOL(4ØØ,1);GOTO TAØ6
        FAHR(Ø);IF(EQ(Q(45),-1)) MCACOL(Q(1),1);GOTO TAØ6
        GOTO TA8
TAØ6:   Q(53)=Ø;RLCPS(Q(61),1)
TA8:    Q(44)=MES1;IF(EQ(ARG(1),2))GOTO TA9
        IF(LE(Q(K),1))GOTO TAEND
        IF(GE(Q(K),5)+EQ(K,59)+LE(Q(K),2))GOTO TA9
        IF(EQ(Q(65),Ø)) Q(65)=1;NOMORG(1,1);GOTO TA9
        ROI1(4Ø)=Q(65);Q(89)=Q(89)+1;ALORG(Q(89),1)
TA9:    $I(4);TYPE!,!,"+++ ZUSAETZLICHES SPEKTRUM UNTER NR=",Q(4);$F(1Ø)
        TYPES" AUF TAPE / BAND NR. ",Q(99)
        TYPE!,"+++ STARTZEIT ";UHRAUS(Q(9Ø),1)
        Z=SEC(Z)-SEC(Q(9Ø))-Q(8Ø);TYPE!,"+++ MESSZEIT = ",Z," S",!
TAEND:  K=Ø>
```

PLOT

WRITES THE FIRST HALF OF THE MCA ON THE X-Y-PLOTTER, CONTROLS VIA PLOTTER INTERFACE AND EXTERNAL INTERRUPT IN DATANIM BEFOR OUTPUT THE CHART DRIVE OF THE PLOTTER.

CALL: PLOT

```
DEFINE PLOT<;           REMARK  24.8.77          &
        LOCAL A,B,PLOT1,PLOT2
        PLOT2=Ø;CONTROL"BØØFB";A=$TIME;CONTROL"BØØGB"
TES1:   ACT(81)Ø;IF(EQ(SENSE(6),1)) PLOT2=PLOT2+1;GOTO TES2
        GOTO PAP
TES2:   CONTROL"BØØFB";IF(GE(PLOT2,2))GOTO PL1Ø
        CONTROL"GØØGB";GOTO TES1
PAP:    B=Ø;DO(4ØØØ)<B=B+1>;CONTROL"BØØFB";DO(5Ø)<B=B+1>
        CONTROL"BØØGB";PLOT1=ROI1(4)
        ACT(81)Ø;IF(EQ(SENSE(6),Ø))GOTO PL9
PL4:    $I(4);TYPE!,"+++ SPEKTRUM NR.",PLOT1;GOTO PL8
PL7:    $I(4);TYPE!,"+++ EICHSPEKTRUM NR.",EIF(4)," ;",EIF(5)
PL8:    TYPE" AUF X-Y-SCHREIBER",!;$F(1Ø)
        PLOT1=SEC($TIME)+6ØØ;PLOT1=TIME(PLOT1)
PL5:    Q(53)=Ø;RLCPS(Q(61),Ø)
        IF(NE(Q(14),5)*NE(Q(44),Ø))GOTO PLEX
        IF(EQ(Q(14),5)*NE(Q(44),1))GOTO PLEX
        ACT(81)Ø;IF(EQ(SENSE(6),Ø))GOTO PL6
        IF(GE(PLOT1,$TIME)*EQ(SENSE(6),1))GOTO PL5
        CONTROL"BØØFB";GOTO PL9
PL9:    TYPE!;DO(2Ø)<TYPE"=">;TYPE" X-Y-SCHREIBER DEFEKT "
        DO(2Ø)<TYPE"=">;TYPE!;GOTO PL6
PL1Ø:   TYPE!;DO(1Ø)<TYPE"=">;TYPE" X-Y-SCHREIBER DEFEKT,"
        TYPE" ODER PROGRAMMSTOP GESETZT ";DO(1Ø)<TYPE"=">
        TYPE!;GOTO PL6
PLEX:   TYPE"+++ SCHREIBERAUSGABE ABGEBROCHEN",!
PL6:    CONTROL"BØØFB";CONTROL"C13A@";B=$TIME
        Q(8Ø)=Q(8Ø)+SEC(B)-SEC(A)
PLEND:  ACT(81)Ø;B=Ø>
```

NOMORG

ORGANIZES FILING OF EVALUATED SPECTRA ON THE NORMAL FILES NORM1 TO NORM1Ø. (CASSETTE)

CALL: NOMORG(A,B)

A=1 TO 1Ø, SPECTRA ON NORM1 TO NORM1Ø
A=11 TO 2Ø, SPECTRA ON NORM1 TO NORM1Ø, ETC.
B=1, ROI1 ON CASSETTE FILE
B=2, CASSETTE FILE TO ROI1
B=3, CASSETTE FILE TO ROI2
B=4, ROI2 TO CASSETTE FILE

```
DEFINE NOMORG<; REMARK  15.2.77          &
         LOCAL X,I
         X=ARG(1)
LSP1:    IF(LE(X,1ØØØ))GOTO LSP2
         X=X-1ØØØ
         GOTO LSP1
LSP2:    IF(LE(X,1ØØ))GOTO LSP3
         X=X-1ØØ
         GOTO LSP2
LSP3:    IF(LE(X,1Ø))GOTO LSP4
         X=X-1Ø
         GOTO LSP3
LSP4:    IF(GT(X,5))GOTO L6
         IF(GT(X,2))GOTO L3
         IF(EQ(X,1)) FILE="NORM1";GOTO LUM
         IF(EQ(X,2)) FILE="NORM2";GOTO LUM
L3:      IF(EQ(X,3)) FILE="NORM3";GOTO LUM
         IF(EQ(X,4)) FILE="NORM4";GOTO LUM
         FILE="NORM5";GOTO LUM
L6:      IF(GT(X,7))GOTO L8
         IF(EQ(X,6)) FILE="NORM6";GOTO LUM
         IF(EQ(X,7)) FILE="NORM7";GOTO LUM
L8:      IF(EQ(X,8)) FILE="NORM8";GOTO LUM
         IF(EQ(X,9)) FILE="NORM9";GOTO LUM
         FILE="NORM1Ø"
LUM:     $U(2);OPEN @FILE;IF(EQ(ARG(2),Ø))GOTO LENDE
         IF(EQ(ARG(2),1))GOTO LUM1
         IF(EQ(ARG(2),2))GOTO LUM2
         IF(EQ(ARG(2),3))GOTO LUMØ
         COPY ROI2:@FILE(1);GOTO LENDE
LUMØ:    COPY @FILE(1):ROI2;GOTO LENDE
LUM1:    COPY ROI1:@FILE(1)
         $I(4);TYPE!,"+++ SPEKTR.NR.",ROI1(4)," AUF FILE "
         TYPES FILE," ABGELEGT.",!;$F(1Ø);GOTO LENDE
LUM2:    COPY @FILE(1):ROI1
LENDE:   ERASE @FILE,FILE>
```

LANORG

ORGANIZES FILING OF EVALUATED SPECTRA ON THE LONG-TERM FILES LANG1 TO LANG25. (CASSETTE)

CALL: LANORG(A,B)

A=1 TO 25, SPECTRA ON LANG1 TO LANG25
A>25, FILLING ON LANG1, AT THE SAME TIME THE VARIABLE Q(55) IS SET=1.
B=1, ROI1 ON CASSETTE FILE
B=2, CASSETTE FILE TO ROI1
B=3, CASSETTE FILE TO ROI2
B=4, ROI2 ON CASSETTE FILE

```
DEFINE LANORG<; REMARK  15.2.77 &
        LOCAL X,I
        X=ARG(1)
        IF(GT(X,25)) X=1;Q(55)=1
        IF(GT(X,20))GOTO L21
        IF(GT(X,10))GOTO L11
        IF(GT(X,5))GOTO L6
        IF(GT(X,2))GOTO L3
        IF(EQ(X,1)) FILE="LANG1";GOTO LUM
        IF(EQ(X,2)) FILE="LANG2";GOTO LUM
L3:     IF(EQ(X,3)) FILE="LANG3";GOTO LUM
        IF(EQ(X,4)) FILE="LANG4";GOTO LUM
        FILE="LANG5";GOTO LUM
L6:     IF(GT(X,7))GOTO L8
        IF(EQ(X,6))FILE="LANG6";GOTO LUM
        FILE="LANG7";GOTO LUM
L8:     IF(EQ(X,8)) FILE="LANG8";GOTO LUM
        IF(EQ(X,9)) FILE="LANG9";GOTO LUM
        FILE="LANG10";GOTO LUM
L11:    IF(GT(X,15))GOTO L16
        IF(GT(X,12))GOTO L13
        IF(EQ(X,11)) FILE="LANG11";GOTO LUM
        FILE="LANG12";GOTO LUM
L13:    IF(EQ(X,13)) FILE="LANG13";GOTO LUM
        IF(EQ(X,14)) FILE="LANG14";GOTO LUM
        FILE="LANG15";GOTO LUM
L16:    IF(GT(X,17))GOTO L18
        IF(EQ(X,16)) FILE="LANG16";GOTO LUM
        FILE="LANG17";GOTO LUM
L18:    IF(EQ(X,18)) FILE="LANG18";GOTO LUM
        IF(EQ(X,19)) FILE="LANG19";GOTO LUM
        FILE="LANG20";GOTO LUM
L21:    IF(GT(X,22))GOTO L23
        IF(EQ(X,21)) FILE="LANG21";GOTO LUM
        FILE="LANG22";GOTO LUM
L23:    IF(EQ(X,23)) FILE="LANG23";GOTO LUM
        IF(EQ(X,24)) FILE="LANG24";GOTO LUM
        FILE="LANG25"
```

```
LUM:     $U(2);OPEN @FILE;IF(EQ(ARG(2),Ø))GOTO LENDE
         IF(EQ(ARG(2),1))GOTO LUM1
         IF(EQ(ARG(2),2))GOTO LUM2
         IF(EQ(ARG(2),3))GOTO LUMØ
         COPY ROI2:@FILE(1);GOTO LENDE
LUMØ:    COPY @FILE(1):ROI2;GOTO LENDE
LUM1:    COPY ROI1:@FILE(1);Q(87)=Ø
         $I(4);TYPE!,"+++ SPEKTR.NR.",ROI1(4)," AUF FILE "
         TYPES FILE," ABGELEGT.",!;$F(1Ø);GOTO LENDE
LUM2:    COPY @FILE(1):ROI1
LENDE:   ERASE @FILE,FILE>
```

ALORG

ORGANIZES FILING OF EVALUATED SPECTRA ON THE ALARM FILES ALAR1 TO ALAR5. (CASSETTE)

CALL: ALORG(A,B)

A=1 TO 5, INITIATES ALAR1 TO ALAR5
B=1, ROI1 ON ALARM FILE
B=2, ALARM FILE TO ROI1
B=3, ALARM FILE TO ROI2
B=4, ROI2 ON ALARM FILE

```
DEFINE ALORG<;  REMARK  8.2.77 &
         LOCAL X
         X=ARG(1)
         IF(GT(X,2))GOTO L3
         IF(EQ(X,1)) FILE="ALAR1";GOTO LUM
         IF(EQ(X,2)) FILE="ALAR2";GOTO LUM
L3:      IF(EQ(X,3)) FILE="ALAR3";GOTO LUM
         IF(EQ(X,4)) FILE="ALAR4";GOTO LUM
         FILE="ALAR5"
LUM:     $U(3);OPEN @FILE;IF(EQ(ARG(2),Ø))GOTO LENDE
         IF(EQ(ARG(2),1))GOTO LUM1
         IF(EQ(ARG(2),2))GOTO LUM2
         IF(EQ(ARG(2),3))GOTO LUMØ
         COPY ROI2:@FILE(1);GOTO LENDE
LUMØ:    COPY @FILE(1):ROI2;GOTO LENDE
LUM1:    IF(GT(X,5)) TYPE!,"+++ ALARMFILE VOLL",!
         COPY ROI1:@FILE(1)
         $I(2);TYPE!,"+++ SPEKTR.NR.",ROI1(4)," AUF FILE "
         TYPES FILE," ABGELEGT.",!;$F(1Ø);GOTO LENDE
LUM2:    COPY @FILE(1):ROI1
LENDE:   ERASE @FILE,FILE>
```

EIORG

ORGANIZES FILING OF THE CALIBRATION SPECTRUM ON CASSETTE.

CALL: EIORG(A)

A=1, ARRAY EIF ON CASSETTE FILE EICHØ
A=2, CASETTE FILE TO ARRAY EIF

```
DEFINE EIORG<;          REMARK 28.9.77  &
        $U(3);OPEN EICHØ
        IF(EQ(ARG(1),2)) COPY EICHØ(1):EIF;GOTO LEX
        COPY EIF:EICHØ(1);$I(4);TYPE!,"+++ EICHUNG NR.",EIF(5)
        TYPE" AUF KASSETTE",!;$F(1Ø)
LEX:    ERASE EICHØ>
```

SPORG

ORGANIZES SPECTRUM FILING DURING THE DECAY MEASUREMENT ON THE CASSETTE FILES SP1 TO SP12, SP1 TO SP7 ON UNIT3, SP8 TO SP12 ON UNIT2. EACH FILE PROCESSES TWO RECORDS FOR ONE 1K-SPECTRUM EACH.

CALL: SPORG(A,B,C)

A=1 TO 12, INITIATES FILES SP1 TO SP12
B=1 OR 2, RECORD 1 OR 2
C=1, ARRAY SPEK ON CASSETTE FILE

```
DEFINE SPORG<;          REMARK  26.9.77          &
        LOCAL FX
        FX=ARG(1)
        IF(EQ(FX,1)) FL="SP1"
        IF(EQ(FX,2)) FL="SP2"
        IF(EQ(FX,3)) FL="SP3"
        IF(EQ(FX,4)) FL="SP4"
        IF(EQ(FX,5)) FL="SP5"
        IF(EQ(FX,6)) FL="SP6"
        IF(EQ(FX,7)) FL="SP7"
        IF(EQ(FX,8)) FL="SP8"
        IF(EQ(FX,9)) FL="SP9"
        IF(EQ(FX,1Ø)) FL="SP1Ø"
        IF(EQ(FX,11)) FL="SP11"
        IF(EQ(FX,12)) FL="SP12"
        $U(3);IF(GE(FX,8)) $U(2)
        OPEN @FL;IF(EQ(ARG(3),1))GOTO WR
        COPY @FL(ARG(2)):SPEK;GOTO SPEX
WR:     COPY SPEK:@FL(ARG(2))
SPEX:   FX=Ø>
```

SEC

CONVERTS EXPRESSIONS LIKE $TIME IN SECONDS

CALL: B=SEC(A)

```
DEFINE SEC<;     REMARK  21.10.76          &
        LOCAL SEC1,SEC2,SEC3,SEC4
        SEC1=(ARG(1)-IP(ARG(1)/1000)*1000)/10
        SEC3=IP(ARG(1)/100000)
        SEC2=IP((ARG(1)-SEC3*100000-SEC1*10)/1000
        SEC4=SEC3*3600+SEC2*60+SEC1>
```

TIME

CONVERTS SECONDS IN EXPRESSIONS LIKE $TIME

CALL: B=TIME(A)

```
DEFINE TIME<;          REMARK    22.10.76          &
        LOCAL X,TIM1,TIM2,TIM3
        TIM1=0;TIM2=0;TIM3=0
        X=ARG(1)
LTI1:   IF(LT(X,3600))GOTO LTI2
        X=X-3600
        TIM1=TIM1+1
        GOTO LTI1
LTI2:   IF(LT(X,60))GOTO LTI3
        X=X-60
        TIM2=TIM2+1
        GOTO LTI2
LTI3:   TIM3=TIM1*100000+TIM2*1000+X*10>
```

FORM

FORMATING ROUTINE FOR FLOATING OUTPUT

CALL: FORM(A,B,C)

A= VARIABLE TO BE PRINTED OUT
B= NUMBER OF SYMBOLS PRECEDING THE DECIMAL POINT
C= NUMBER OF SYMBOLS FOLLOWING THE DECIMAL POINT

```
DEFINE FORM<;    REMARK  29.11.76          &
         LOCAL Z,U,V,W
         IF(GT(IP(ARG(1)),10↑ARG(2)-1)+LT(ARG(1),0))GOTO FO3
         Z=ARG(1)+10↑-(ARG(3)+1)
         U=FP(Z)*10↑ARG(3);$I(ARG(2));TYPE IP(Z),".";W=1
FO1:     IF(EQ(W,ARG(3)))GOTO FO2
         V=10↑(ARG(3)-W)
         IF(LT(U,V)) W=W+1;GOTO FO1
FO2:     DO(W-1)<TYPE"0">;Z=ARG(3)-W+1;TYPE $I(Z)+U
         GOTO FO4
FO3:     W=ARG(2)+ARG(3)+1;$F(W);TYPE ARG(1)
FO4:     $F(10)>
```

UHRAUS

PRINTS OUT THE HOUR

CALL: UHRAUS(A,B)

A= AN EXPRESSION LIKE $TIME
B=0, HOUR AND MINUTE IS PRINTED OUT
B=1, ADDITIONAL SECONDS ARE PRINTED OUT

```
DEFINE UHRAUS<; REMARK  4.2.77  &
         LOCAL ZEIT
         ZEIT=ARG(1)/100000
         $F(5)
         IF(GE(ZEIT,10)) $F(6)
         IF(EQ(IP(ZEIT),0)) $F(4)
         TYPE ZEIT," UHR"
         IF(EQ(ARG(2),0))GOTO UHREX
         ZEIT=FP(ZEIT*100)
         ZEIT=ZEIT*100
         $F(4)
         IF(GE(ZEIT,10)) $F(5)
         TYPE ZEIT," S"
UHREX:   $F(10)>
```

DATAUS

PRINTS OUT DATE AND HOUR

CALL: DATAUS

```
DEFINE DATAUS<; REMARK  4.2.77          &
        LOCAL ZEIT
        ZEIT=$TIME/100000.
        $I(2)
        TYPE!,Q(6),". "
        IF(GE(Q(7),10))GOTO DATA1
        $I(1)
        GOTO DATA2
DATA1:  $I(2)
DATA2:  TYPE Q(7),". "
        $I(4)
        TYPE Q(8)
        IF(GE(ZEIT,10))GOTO DATA3
        IF(EQ(IP(ZEIT),0)) $F(4);GOTO DATA4
        $F(5)
        GOTO DATA4
DATA3:  $F(6)
DATA4:  TYPE "                  ZEIT: ",ZEIT,!
        $F(10)>
```

DETKOF

CALCULATES THE EPSILON (E) DETECTOR COEFFICIENTS RELATED TO THE ENERGIES ROI1(X+10) TO BE EVALUATED. THE VALUES CALCULATED ARE FILED IN ROI1(X+5) AND DISPLAYED ON TELETYPE.

CALL: $U(3);RUN DEKOF

```
DEFINE DETKOF<;          REMARK  25.8.77          &
        LOCAL E1,E2,E3,E4,E5,K,EPS,X,Y
        TYPE!;K=1;X=50;E1=-.4915;E2=3.036;E3=-1.942;E4=1.0437;E5=.3
        DO(Q(13))<Y=ROI1(X+10)/1000
        EPS=E1*Y↑3+E2*Y↑2+E3*Y+E4
        IF(EQ(EPS,0)) EPS=10↑-90
        ROI1(X)=E5+1/EPS;TYPE!,"EPS",$I(2)+K," = "
        $F(10);TYPE ROI1(X);X=X+17;K=K+1>;TYPE!>
```

DATKOR

ADVANCES THE DATE IF $TIME>2400000, INCREMENTS Q(41)= SEQUENCE OF CALIBRATION MEASUREMENTS, Q(87)= SEQUENCE FOR SPECTRUM FILING ON LONG-TERM FILE. REDUCES BY 2400000 THE VARIABLES Q(47)= SEQUENCE OF MEASUREMENT AND Q(88)= STARTING DATE FOR THE NEXT DECAY MEASUREMENT. Q(100)<0 PREVENTS ADVANCING OF THE DATE.

CALL: DATKOR

Q(6)=DAY, Q(7)=MONTH, Q(8)=YEAR

```
DEFINE DATKOR<; REMARK  5.10.77          &
         LOCAL DATX,UHR,UHRH,UHRHN,UHRM,UHRS,X29,DK
DAT0:    DATX=0; DK=0
         IF(GT(2400000,$TIME)+EQ(Q(100),-1))GOTO DAT4
         DATX=1; Q(41)=Q(41)+1; Q(87)=Q(87)+1
         Q(6)=Q(6)+1
DAT01:   UHR=$TIME+10+DK
         UHRS=IP(UHR/1000)*1000
         UHRS=UHR-UHRS
         UHRH=IP(UHR/100000)
         UHRM=UHRH*100000
         UHRM=UHR-UHRM-UHRS
         UHRM=UHRM/1000
         IF(LT(UHRS,600))GOTO DAT1
         UHRS=UHRS-600
         UHRM=UHRM+1
DAT1:    IF(LT(UHRM,60))GOTO DAT2
         UHRM=UHRM-60
         UHRH=UHRH+1
DAT2:    UHRHN=UHRH-24
         IF(GE($TIME,UHR-3)) DK=DK+10; GOTO DAT01
         $WTIME(UHRH,UHRM,UHRS)
         $STIME(UHRHN,UHRM,UHRS)
         Q(47)=Q(47)-2400000; Q(88)=Q(88)-2400000
         IF(LT(Q(88),0)) Q(88)=0
DAT5:    X29=29
         IF(EQ(FP(Q(8)/4),0))X29=X29+1
         IF(LT(Q(6),X29))GOTO DAT3
         IF(EQ(Q(6),X29))GOTO FEB
         IF(EQ(Q(6),31))GOTO M30
         IF(EQ(Q(6),32))GOTO NEU
         GOTO DAT3
FEB:     IF(NE(Q(7),2))GOTO DAT3
         GOTO NEU
M30:     IF(EQ(Q(7),4))GOTO NEU
         IF(EQ(Q(7),6))GOTO NEU
         IF(EQ(Q(7),9))GOTO NEU
         IF(EQ(Q(7),11))GOTO NEU
         GOTO DAT3
NEU:     Q(6)=1
         IF(EQ(Q(7),12))GOTO NEUJ
         Q(7)=Q(7)+1
         GOTO DAT3
NEUJ:    Q(7)=1
         Q(8)=Q(8)+1
DAT3:    IF(EQ(DATX,1))GOTO DAT0
DAT4:    X29=29>
```

INIT

INITIATES THE INDIVIDUAL PROGRAMS BY MEANS OF DECLARE SO THAT THE PROGRAMS CAN BE CALLED EXCLUSIVELY BY THEIR NAMES WITHOUT INDICATION OF THE UNIT NUMBER.

CALL: $U(3);RUN INIT

```
DEFINE INI<;            REMARK  2.8.77  &
$U(3);DECLARE DATAUS,DATKOR,UHRAUS,MCLES,FAHR,TIME,SEC,COUNT,TAPE
DECLARE ALORG,FENAUS,RLCPS,FORM,AUSWER,MESPRO,PEAVER,NORMES,ALBEA
DECLARE EIKOF,EICHEN,EIPRO,ABKLI,ABAUS,AUSWAB,MEPAUS,SONDER,ABSCH
DECLARE STEUER,ABNACH,NOMORG,LANORG,EIORG,MCACOL,PLOT,SPORG,RESTAR
DECLARE ONGAM;DELETE INIT>
```

REMESS

LOADS DATA FILES FROM CASSETTE INT THE ARRAYS ROI1, ROI2 AND Q.

CALL: $U(3);RUN REMESS

```
DEFINE REMESS<; REMARK  29.9.77        &
        IF(EQ(DEF(ROI1),Ø)) DIMENS ROI1(3ØØ)
        IF(EQ(DEF(ROI2),Ø)) DIMENS ROI2(3ØØ)
        IF(EQ(DEF(Q),Ø)) DIMENS Q(1ØØ)
        $U(3);OPEN SAFIL1,SAFIL2,DATFIL
        COPY SAFIL1(1):ROI1;COPY SAFIL2(1):ROI2;COPY DATFIL(1):Q
        DELETE REMESS;ERASE SAFIL1,SAFIL2,DATFIL>
```

STORE

STORES THE ARRAYS ROI1, ROI2 AND Q ON CASSETTE FILES.

CALL: $U(3);RUN STORE

```
DEFINE STORE<;  REMARK  2.5.77        &
        $U(3)
        IF(EQ(DEF(ROI1),1)) OPEN SAFIL1;COPY ROI1:SAFIL1(1);GOTO LS1
        TYPE!,"ROI1 NICHT IM SPEICHER, NICHT AUF KASSETTE GELEGT!"
LS1:    IF(EQ(DEF(ROI2),1)) OPEN SAFIL2;COPY ROI2:SAFIL2(1);GOTO LS2
        TYPE!,"ROI2 NICHT IM SPEICHER, NICHT AUF KASSETTE GELEGT!"
LS2:    IF(EQ(DEF(Q),1)) OPEN DATFIL;COPY Q:DATFIL(1);GOTO LS3
        TYPE!,"FELD Q NICHT IM SPEICHER, NICHT AUF KASSETTE GELEGT!"
LS3:    IF(EQ(DEF(ROI1),1)) ERASE SAFIL1
        IF(EQ(DEF(ROI2),1)) ERASE SAFIL2
        IF(EQ(DEF(Q),1)) ERASE DATFIL
        TYPE!;DELETE STORE>
```

AUSGAB

DIALOG PROGRAM FOR COMPLETE OR PARTIAL OUTPUT WITH ANY STEP WIDTH OF THE ARRAY ROI1, ROI2, EIF, SPEC, Q OR OF A ARRAY XX OF ANY LENGTH.

CALL: $U(3);RUN AUSGAB

```
DEFINE AUSGAB<; REMARK  26.9.77 &
        LOCAL A,B,C,D,K,L,X,Y,BMAX,NR,NAME
ANF:    TYPE!,"WELCHES FELD SOLL AUSGEGEBEN WERDEN ? "
        READS(1) NAME;BMAX=300
        IF(EQ(NAME,"XX"))GOTO FEL
        IF(EQ(NAME,"SPEC"))GOTO SPE
        IF(EQ(NAME,"ROI1")) RENAME ROI1:FELD;GOTO AUS
        IF(EQ(NAME,"ROI2"))GOTO ROI
        IF(EQ(NAME,"Q")) RENAME Q:FELD;BMAX=100;GOTO AUS
        IF(EQ(NAME,"EIF")) BMAX=200;GOTO EICH
        IF(EQ(NAME,"$")) $F(10);GOTO FERT
        TYPE!,"FALSCHER FELDNAME !!",!;GOTO ANF
SPE:    IF(EQ(DEF(SPEC),1)) RENAME SPEC:FELD;BMAX=1054;GOTO AUS
        TYPE!,"FELD SPEC NICHT IN MASCHINE",!;GOTO ANF
FEL:    IF(EQ(DEF(XX),0))GOTO FEL1
        TYPE!,"FELDLAENGE ? ";READ(1) BMAX;RENAME XX:FELD;GOTO AUS
FEL1:   TYPE!,"FELD XX NICHT DEFINIERT";GOTO ANF
EICH:   IF(EQ(DEF(EIF),0)) DIMENS EIF(200)
        EIORG(2);RENAME EIF:FELD;GOTO AUS
ROI:    TYPE!,"WELCHER DATENFILE SOLL AUSGEGBEN WERDEN, NR=-1 ",!
        TYPE"BEDEUTET KEINEN DATENFILE LADEN.",!,"NR ? "
        READ(1) NR
        IF(LE(NR,0)*EQ(DEF(ROI2),0))GOTO RO1
        IF(LE(NR,0)) RENAME ROI2:FELD;GOTO AUS
        IF(EQ(DEF(ROI2),0)) DIMENS ROI2(300)
        NOMORG(NR,3);RENAME ROI2:FELD;GOTO AUS
RO1:    TYPE!,"KEIN DATENFELD IN MASCHINE.";GOTO ROI
AUS:    TYPE"START BEI FELD NR. ? ";READ(1) A
        TYPE"ENDE BEI FELD NR. ? ";READ(1) B
        IF(GE(B,BMAX)) B=BMAX
        TYPE"SCHRITTWEITE ? ";READ(1) C
        TYPE!,!,!,"FELD ";TYPES NAME
        IF(EQ(NAME,"SPEC")+EQ(NAME,"XX")+EQ(NAME,"Q"))GOTO AU1
        TYPE"  SPEKTRUM NR. ",$I(4)+FELD(4)
AU1:    TYPE!
AU2:    I=0;D=A;$I(5);TYPE!,!;DO(5)<TYPE D,"           ";D=D+C>;$F(10)
        TYPE!
AU3:    IF(GT(A,B)) $F(10);GOTO RES
        IF(EQ(NAME,"ROI1")+EQ(NAME,"ROI2"))GOTO AU01
        GOTO AU0
AU01:   K=59;L=61
AU02:   IF(EQ(A,K)+EQ(A,L)) TYPES" ",FELD(A),"           ";GOTO AU9
        K=K+17;IF(GT(K,300))GOTO AU0
        L=L+17;GOTO AU02
```

```
AUØ:    IF(GT(ABS(FELD(A)),10↑7-1))GOTO AU4
        IF(LT(ABS(FELD(A)),10↑-7)*NE(FELD(A),Ø))GOTO AU6
        TYPE FELD(A),"    ";GOTO AU9
AU4:    X=FELD(A);Y=Ø
AU5:    IF(GT(ABS(X),1Ø)) X=X/1Ø;Y=Y+1;GOTO AU5
        GOTO AU8
AU6:    X=FELD(A);Y=Ø
AU7:    IF(LT(ABS(X),1)) X=X*1Ø;Y=Y-1;GOTO AU7
AU8:    $F(6);TYPE X," E";$I(4);TYPE Y,"  ";$F(1Ø)
AU9:    A=A+C;I=I+1;IF(LT(I,5))GOTO AU3
        GOTO AU2
RES:    TYPE!;IF(EQ(NAME,"ROI1")) RENAME FELD:ROI1
        IF(EQ(NAME,"SPEC")) RENAME FELD:SPEC
        IF(EQ(NAME,"ROI2")) RENAME FELD:ROI2
        IF(EQ(NAME,"Q")) RENAME FELD:Q
        IF(EQ(NAME,"EIF")) RENAME FELD:EIF
        IF(EQ(NAME,"XX")) RENAME FELD:XX
        GOTO ANF
FERT:   DELETE AUSGAB>
```

ENKA

PROGRAM FOR CONVERTING THE CHANNEL POSITION INTO THE ENERGY AND VICE VERSA. THE PROGRAM CALCULATES WITH THE HELP OF THE CALIBRATION FACTORS Q(10)=A, Q(11)=B, Q(12)=C

CALL: $U(3);RUN ENKA

```
DEFINE ENKA<;          REMARK  22.8.77           &
        LOCAL X,Y,Z,N,WAS
DEFINE KANAL<
        Z=Q(11)/(2*Q(10));Y=Q(12)-X;Y=Y/Q(10)
        N=Z*Z-Y;N=SQRT(N);Y=-Z-N
        IF(LT(Y,0)+GT(Y,2048)) Y=-Z+N
        Y=Y>
DEFINE ENER<
        Y=Q(10)*X*X+Q(11)*X+Q(12)>
        IF(EQ(Q(10),0)+EQ(Q(11),0))GOTO KEIN
        TYPE!,"ZULAESSIGE BEFEHLE AUF DIE FRAGE WAS",!
        TYPE!,"K  ,EINGABE DER KANALLAGE"
        TYPE!,"E  ,EINGABE DER ENERGIE"
        TYPE!,"$  ,PROGRAMM BEENDEN"
LOOP:   TYPE!,"WAS ? ";READS(1) WAS
        IF(EQ(WAS,"K")) TYPE"KANALLAGE ? ";READ(1) X;GOTO KANL
        IF(EQ(WAS,"E")) TYPE"ENERGIE IN KEV ? ";READ(1) X;GOTO ENE
        IF(EQ(WAS,"$"))GOTO EX
        TYPE!,"FALSCHER BEFEHL !";GOTO LOOP
KANL:   ENER;TYPE"ENERGIE = ",Y," KEV",!;GOTO LOOP
ENE:    KANAL;TYPE"KANALLAGE = ",Y,!;GOTO LOOP
KEIN:   TYPE!,"KEINE EICHFAKTOREN VORHANDEN",!
EX:     DELETE KANAL,ENER,ENKA>
```

SORT

AUXILIARY PROGRAM FOR GENERATING A NEW CASSETTE FOR UNIT3. THE PROGRAM OF UNIT3 ARE SORTED BY SORT IN AN OPTIMUM SEQUENCE SO THAT THE LEAST POSSIBLE LOADING TIME IS REQUIRED WHEN CALLING THE PROGRAM. THE SEQUENCE CAN BE VARIED IN THE STARTING DIALOG.

CALL: $U(1);RUN SORT

```
DEFINE SORT<;    REMARK  29.9.77 &
DEFINE LOOK<
        IF(EQ(DEF(X),1)) ERASE X
        IF(EQ(NAM,"ALAR1")+EQ(NAM,"ALAR2")) K=300;CP1;GOTO LO1
        IF(EQ(NAM,"ALAR3")+EQ(NAM,"ALAR4")) K=300;CP1;GOTO LO1
        IF(EQ(NAM,"ALAR5")) K=300;CP1;GOTO LO1
        IF(EQ(NAM,"SAFIL1")+EQ(NAM,"SAFIL2")) K=300;CP2;GOTO LO1
        IF(EQ(NAM,"DATFIL")) K=100;CP2;GOTO LO1
        IF(EQ(NAM,"EICHØ")) K=200;CP2;GOTO LO1
        IF(EQ(NAM,"SP1")+EQ(NAM,"SP2")) K=1090,CP3;GOTO LO1
        IF(EQ(NAM,"SP3")+EQ(NAM,"SP4")) K=1090;CP3;GOTO LO1
        IF(EQ(NAM,"SP5")+EQ(NAM,"SP6")) K=1090;CP3;GOTO LO1
        IF(EQ(NAM,"SP7")) K=1090;CP3;GOTO LO1
        UML
LO1:    NAM=Ø>

DEFINE CP1<
        DIMENS X(K);$U(3);CREATE @NAM(K);APPEND X:@NAM>

DEFINE CP2<
        DIMENS X(K);$U(3);CREATE @NAM(K);APPEND X:@NAM;ERASE @NAM
        $U(2);OPEN @NAM;COPY @NAM(1):X;ERASE @NAM;$U(3);OPEN @NAM
        COPY X:@NAM(1)>

DEFINE CP3<
        DIMENS X(K);$U(3);CREATE @NAM(K);DO(2)<APPEND X:@NAM>>

DEFINE UML<
        $U(2);LOAD @NAM;$U(3);SAVE @NAM;DELETE @NAM>

        TYPE!,"NEUE KASSETTE IN UNIT #3, ALTE KASSETTE IN UNIT #2, "
        TYPE"DANN RETURNTASTE",!;KBWAIT
        TYPE"STARTDIALOG ABWARTEN",!
        DIMENS A(100);$U(1);OPEN NAMFIL;COPY NAMFIL(1):A;I=Ø
L2:     I=I+1;TYPE!,$I(2)+I,"  ";$F(10);TYPES A(I)
        IF(NE(A(I),"$"))GOTO L2
        J=I;TYPE!,"REIHENFOLGE RICHTIG ? ";READS(1)FRAG
        IF(EQ(FRAG,"JA"))GOTO START
L4:     TYPE!,"AENDERN=A; EINFUEGEN=E; STREICHEN=S",!,"WAS ? "
        READS(1)FRAG
        TYPE"NR. ? ";READ(1)I
        IF(EQ(FRAG,"S")) DO(J-I)<A(I)=A(I+1);I=I+1>;J=J-1;GOTO L5
        TYPE"NAME ? ";READS(1)NAM
        IF(EQ(FRAG,"A")) A(I)=NAM;GOTO L5
```

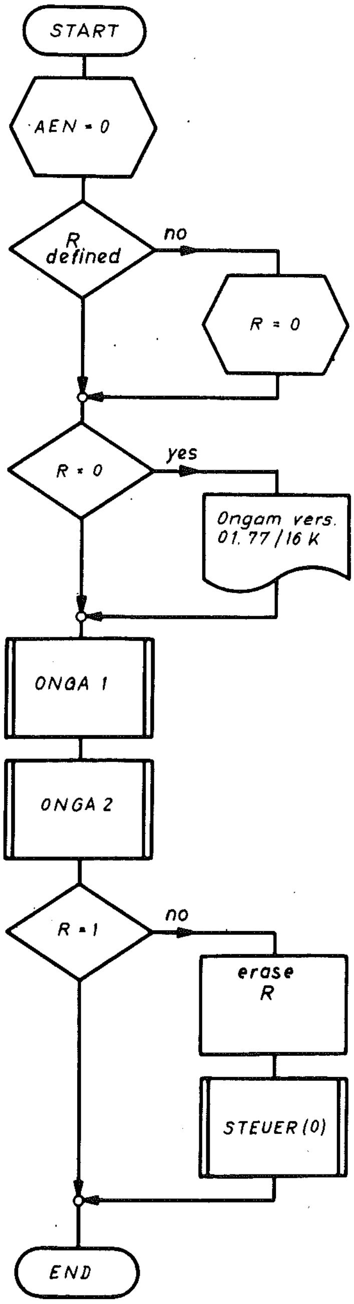
```
        K=J;DO(J-I+1)<A(K+1)=A(K);K=K-1>;A(I)=NAM;J=J+1
L5:     TYPE!,"ALLES O.K. ? ";READS(1)FRAG
        IF(NE(FRAG,"JA")) GOTO L4
        TYPE!,"NOCHMAL LISTEN ? ";READS(1)FRAG
        IF(EQ(FRAG,"JA")) I=Ø;GOTO L2
START:  I=Ø;TYPE!,"LAUFZEIT CA 2,5 STUNDEN",!
        COPY A:NAMFIL(1)
S1:     I=I+1;NAM=A(I)
        IF(NE(NAM,"$")) LOOK;GOTO S1
        TYPE!,"FILE COPY TEST",!;DIMENS ROI1(3ØØ);$U(3);LOAD ALORG
        I=1;DO(5)<ALORG(I,1);I=I+1>;DELETE ALORG;ERASE ROI1
        DIMENS SPEK(1Ø9Ø);I=1;$U(3);LOAD SPORG
        DO(7)<SPORG(I,1,1);TYPE!,I,"  ";SPORG(I,2,1);TYPE I;I=I+1>
        TYPE!;EXIT>
```

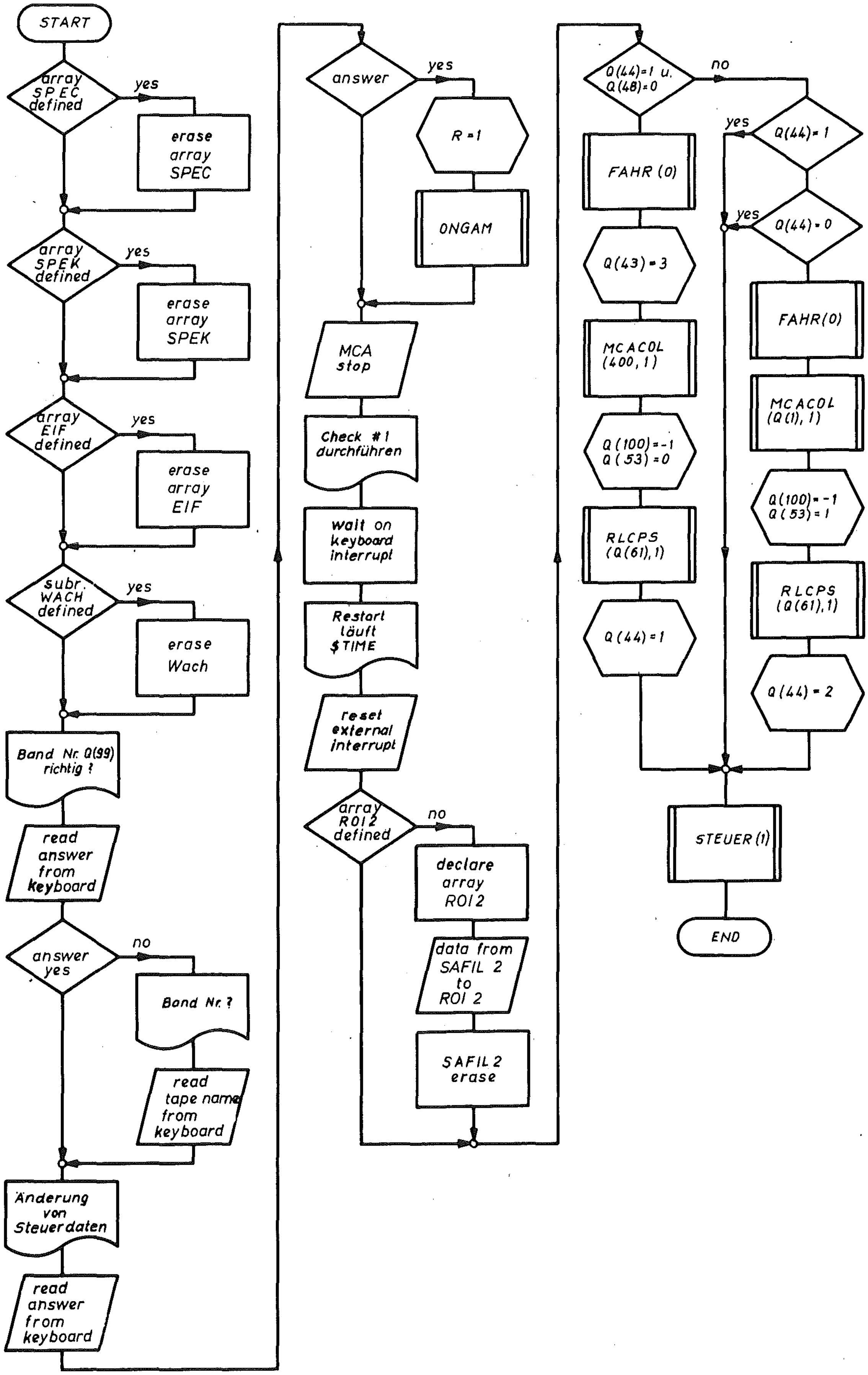
FILTES

PERFORMS A FILE COPY TEST FOR A NEWLY STARTED CASSETTE OF UNIT2.

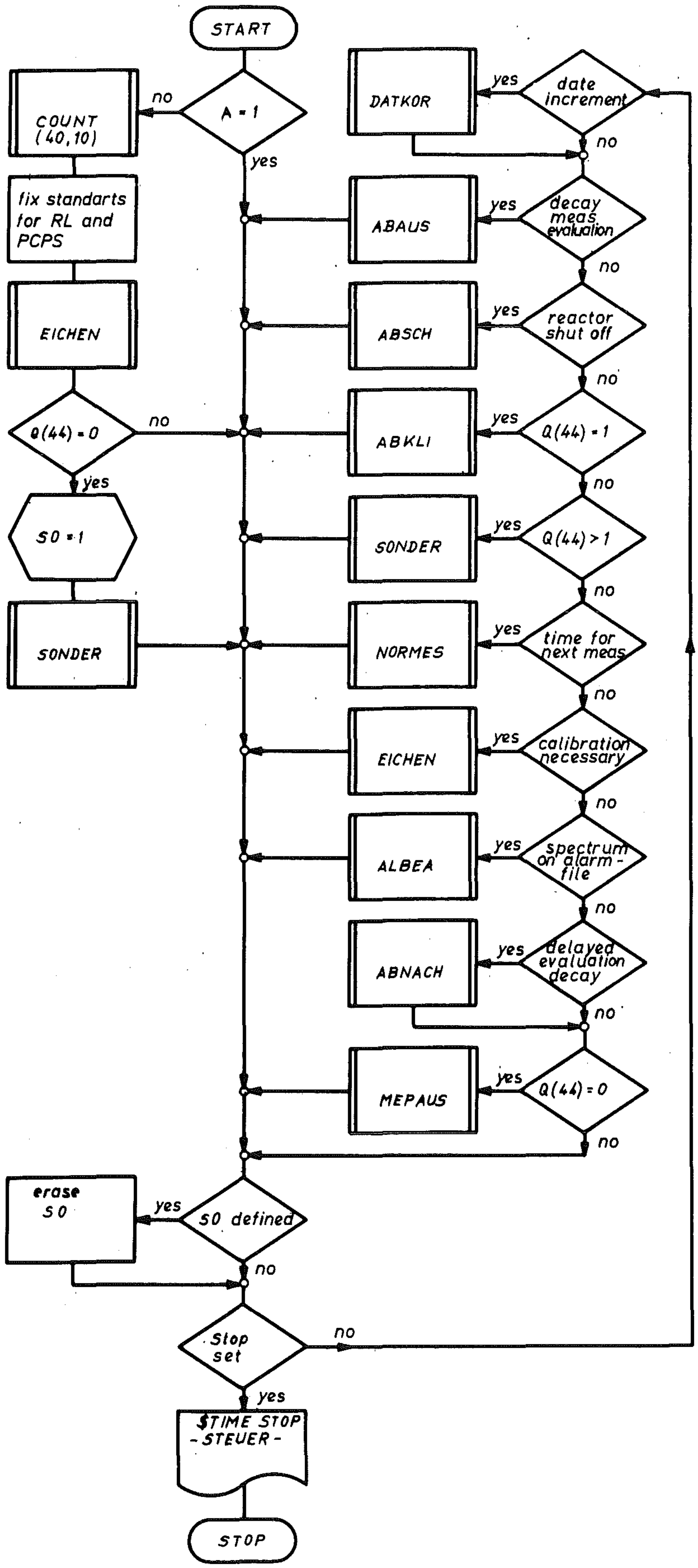
CALL: $U(1);RUN FILTES

```
DEFINE FILTES<;          REMARK  1Ø.1Ø.77          &
TYPE!,"FILE-COPY-TEST FUER KASSETTE ONDAT1, LAUFZEIT CA. 35 MIN",!
DIMENS SPEK(1Ø9Ø);$U(3);LOAD SPORG;I=8;TYPE!
$U(2);OPEN SP8,SP9,SP1Ø,SP11,SP12
DO(5)<SPORG(I,1,1);TYPE I,"  ";SPORG(I,2,1);TYPE I,!;I=I+1>
TYPE"FILE SP8 BIS SP12 O.K.",!,!
ERASE SPEK;DELETE SPORG;DIMENS ROI2(3ØØ);$U(3);LOAD NOMORG;I=1
$U(2);OPEN NORM1,NORM2,NORM3,NORM4,NORM5,NORM6,NORM7,NORM8,NORM9,NORM1Ø
DO(1Ø)<NOMORG(I,3);TYPE I,!;I=I+1>
TYPE"FILE NORM1 BIS NORM1Ø O.K.",!,!
DELETE NOMORG;$U(3);LOAD LANORG;I=1
$U(2);OPEN LANG1,LANG2,LANG3,LANG4,LANG5,LANG6,LANG7,LANG8,LANG9,LANG1Ø
OPEN LANG11,LANG12,LANG13,LANG14,LANG15,LANG16,LANG17,LANG18,LANG19
OPEN LANG2Ø,LANG21,LANG22,LANG23,LANG24,LANG25
DO(25)<LANORG(I,3);TYPE I,!;I=I+1>
TYPE"FILE LANG1 BIS LANG25 O.K.",!;EXIT>
```

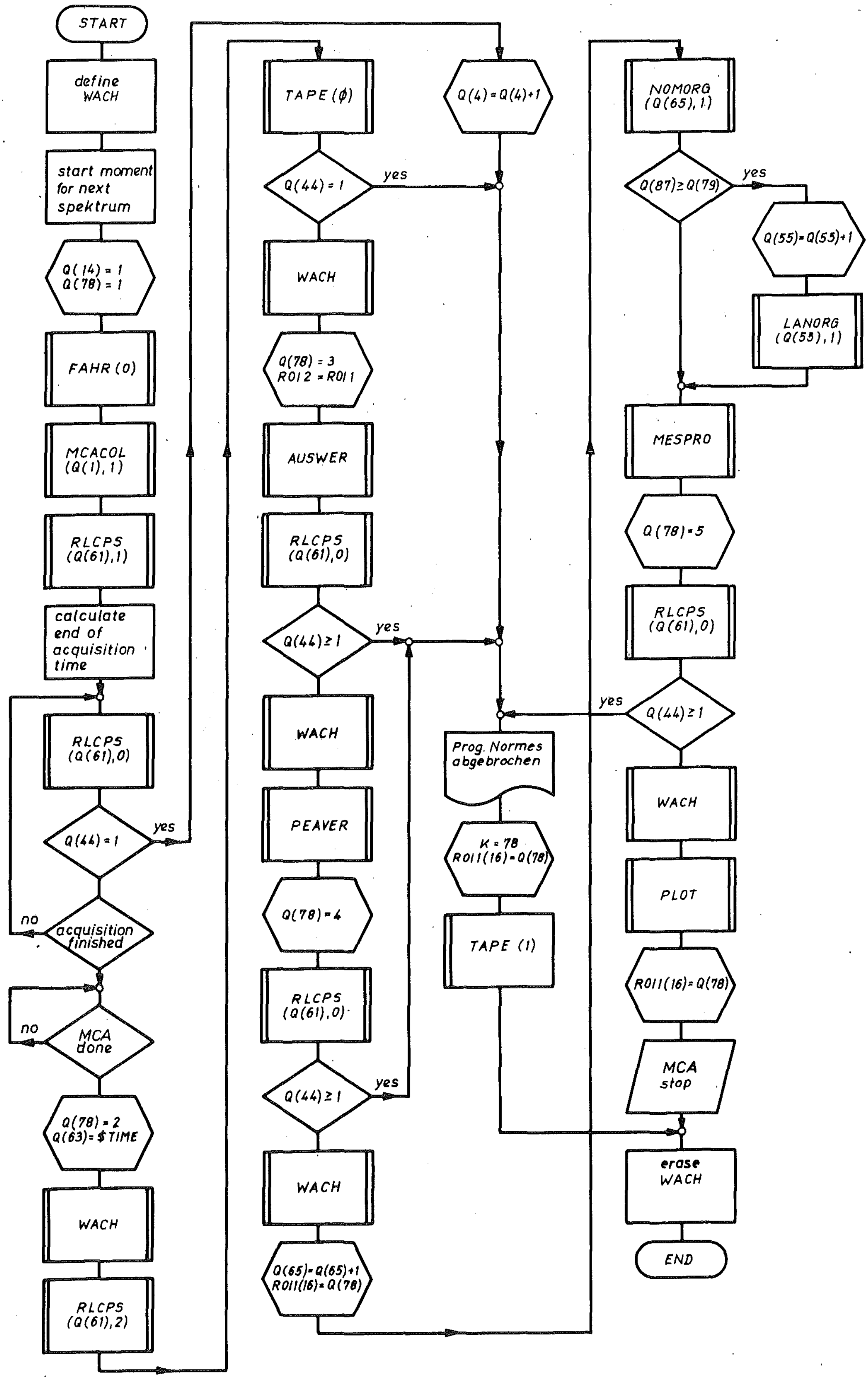

START
AEN = 0
R defined
no
R = 0
R = 0
yes
Ongam vers. 01.77/16 K
ONGA 1
ONGA 2
R = 1
no
erase R
STEUER (0)
END


START
array SPEC defined
yes
erase array SPEC
array SPEK defined
yes
erase array SPEK
array EIF defined
yes
erase array EIF
subr. WACH defined
yes
erase Wach
Band Nr. Q(99) richtig ?
read answer from keyboard
answer yes
no
Band Nr. ?
read tape name from keyboard
Änderung von Steuerdaten
read answer from keyboard
answer
yes
R = 1
ONGAM
MCA stop
Check #1 durchführen
wait on keyboard interrupt
Restart läuft $TIME
reset external interrupt
array ROI2 defined
no
declare array ROI2
data from SAFIL 2 to ROI 2
SAFIL 2 erase
Q(44)=1 u. Q(48)=0
no
FAHR (0)
Q(43) = 3
MCACOL (400,1)
Q(100) = -1
Q(53) = 0
RLCPS (Q(61),1)
Q(44) = 1
Q(44) = 1
yes
Q(44) = 0
yes
FAHR(0)
MCACOL (Q(1), 1)
Q(100) = -1
Q(53) = 1
RLCPS (Q(61),1)
Q(44) = 2
STEUER (1)
END

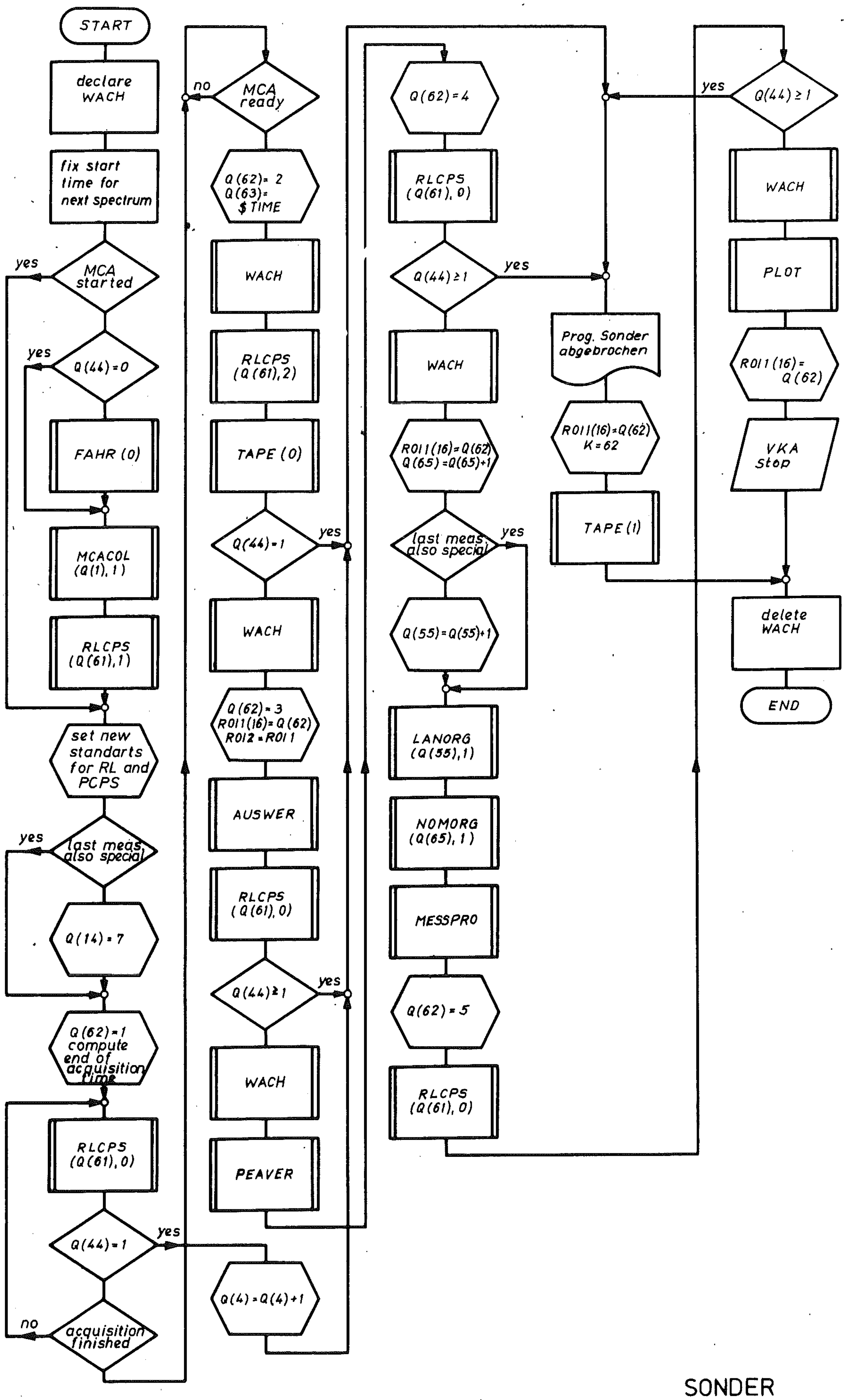
RESTAR


START
A = 1
no
yes
COUNT (40,10)
fix standarts for RL and PCPS
EICHEN
Q(44) = 0
no
yes
S0 = 1
SONDER
DATKOR
yes
date increment
no
ABAUS
yes
decay meas evaluation
no
ABSCH
yes
reactor shut off
no
ABKLI
yes
Q(44) = 1
no
SONDER
yes
Q(44) > 1
no
NORMES
yes
time for next meas
no
EICHEN
yes
calibration necessary
no
ALBEA
yes
spectrum on alarm-file
no
ABNACH
yes
delayed evaluation decay
no
MEPAUS
yes
Q(44) = 0
no
erase S0
yes
S0 defined
no
Stop set
no
yes
$TIME STOP - STEUER -
STOP

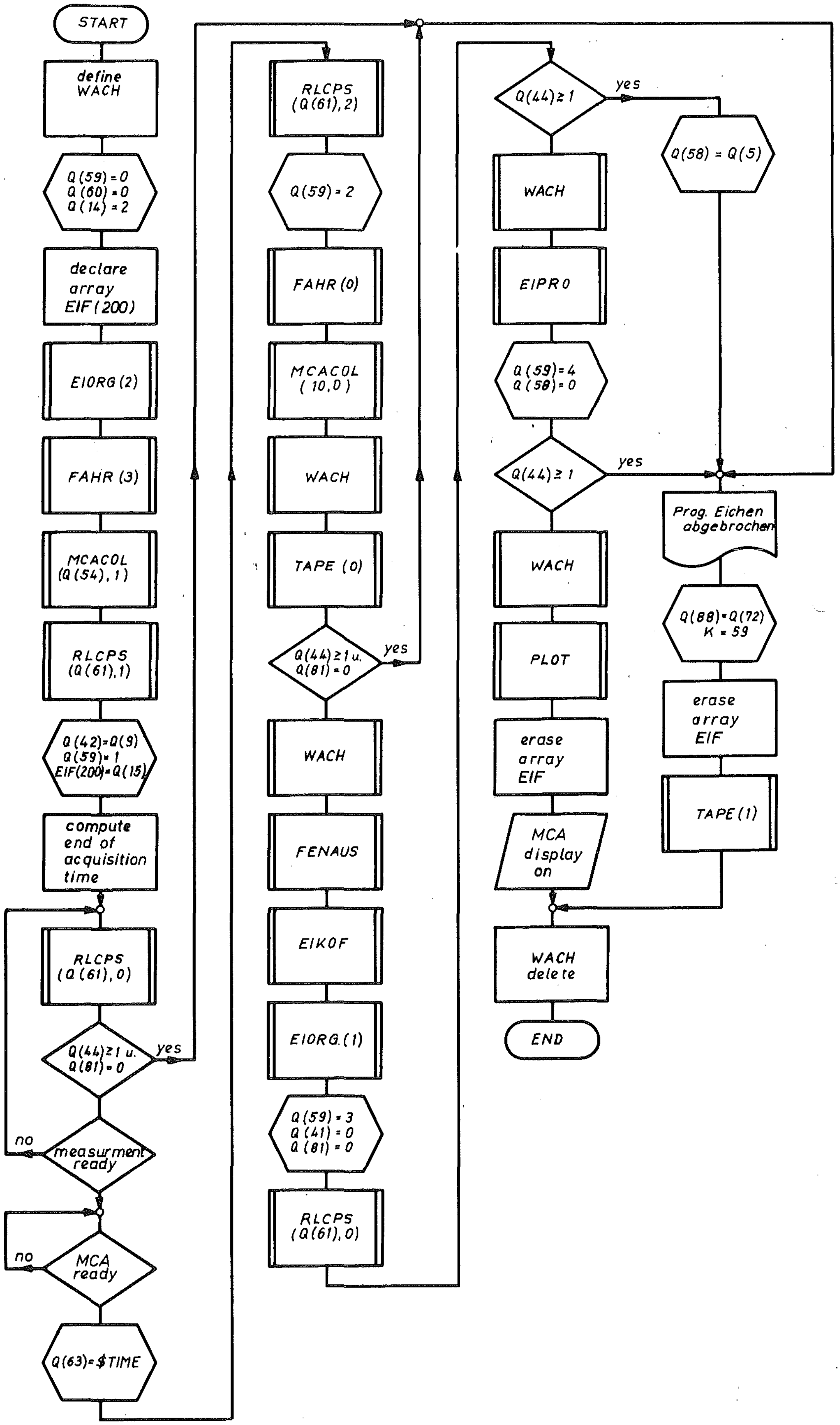
STEUER (A)


START
define WACH
start moment for next spektrum
Q(14) = 1
Q(78) = 1
FAHR (0)
MCACOL (Q(1), 1)
RLCPS (Q(61),1)
calculate end of acquisition time
RLCPS (Q(61),0)
Q(44) = 1
yes
acquisition finished
no
MCA done
no
Q(78) = 2
Q(63) = $TIME
WACH
RLCPS (Q(61),2)
TAPE (∅)
Q(44) = 1
yes
WACH
Q(78) = 3
RO12 = RO11
AUSWER
RLCPS (Q(61),0)
Q(44) ≥ 1
yes
WACH
PEAVER
Q(78) = 4
RLCPS (Q(61),0)
Q(44) ≥ 1
yes
WACH
Q(65) = Q(65)+1
RO11(16) = Q(78)
Q(4) = Q(4)+1
Prog. Normes abgebrochen
K = 78
RO11(16) = Q(78)
TAPE (1)
NOMORG (Q(65), 1)
Q(87) ≥ Q(79)
yes
Q(55) = Q(55)+1
LANORG (Q(55), 1)
MESPRO
Q(78) = 5
RLCPS (Q(61),0)
Q(44) ≥ 1
yes
WACH
PLOT
RO11(16) = Q(78)
MCA stop
erase WACH
END

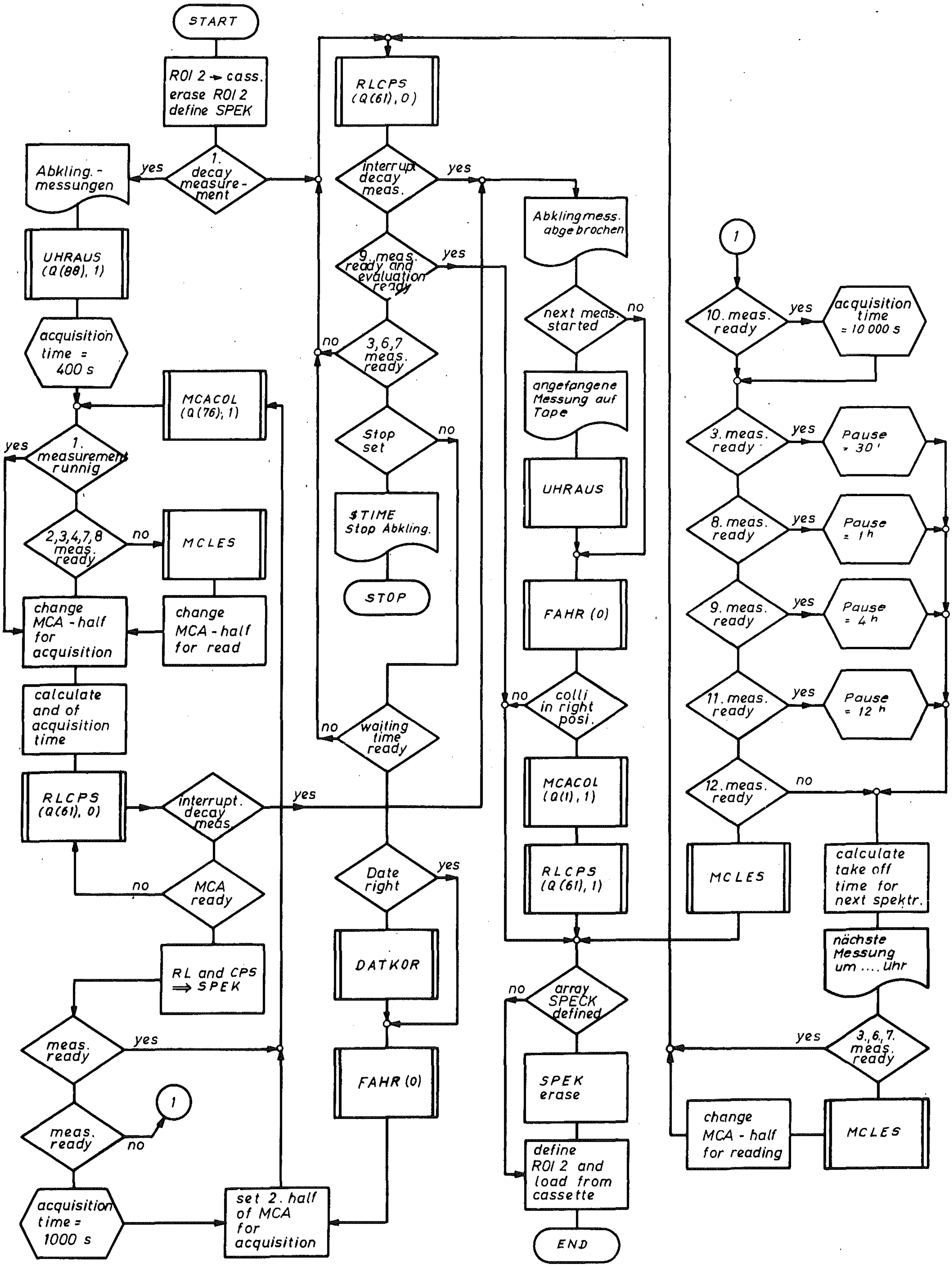
NORMES


START
declare WACH
fix start time for next spectrum
MCA started
yes
Q(44)=0
yes
FAHR (0)
MCACOL (Q(1), 1)
RLCPS (Q(61),1)
set new standarts for RL and PCPS
last meas. also special
yes
Q(14)=7
Q(62)=1 compute end of acquisition time
RLCPS (Q(61),0)
Q(44)=1
yes
acquisition finished
no
Q(4)=Q(4)+1
MCA ready
no
Q(62)=2 Q(63)= $TIME
WACH
RLCPS (Q(61),2)
TAPE (0)
Q(44)=1
yes
WACH
Q(62)=3 R011(16)=Q(62) R012=R011
AUSWER
RLCPS (Q(61),0)
Q(44)≥1
yes
WACH
PEAVER
Q(62)=4
RLCPS (Q(61),0)
Q(44)≥1
yes
WACH
R011(16)=Q(62) Q(65)=Q(65)+1
last meas. also special
yes
Q(55)=Q(55)+1
LANORG (Q(55),1)
NOMORG (Q(65),1)
MESSPRO
Q(62)=5
RLCPS (Q(61),0)
Prog. Sonder abgebrochen
R011(16)=Q(62) K=62
TAPE (1)
Q(44)≥1
yes
WACH
PLOT
R011(16)= Q(62)
VKA Stop
delete WACH
END

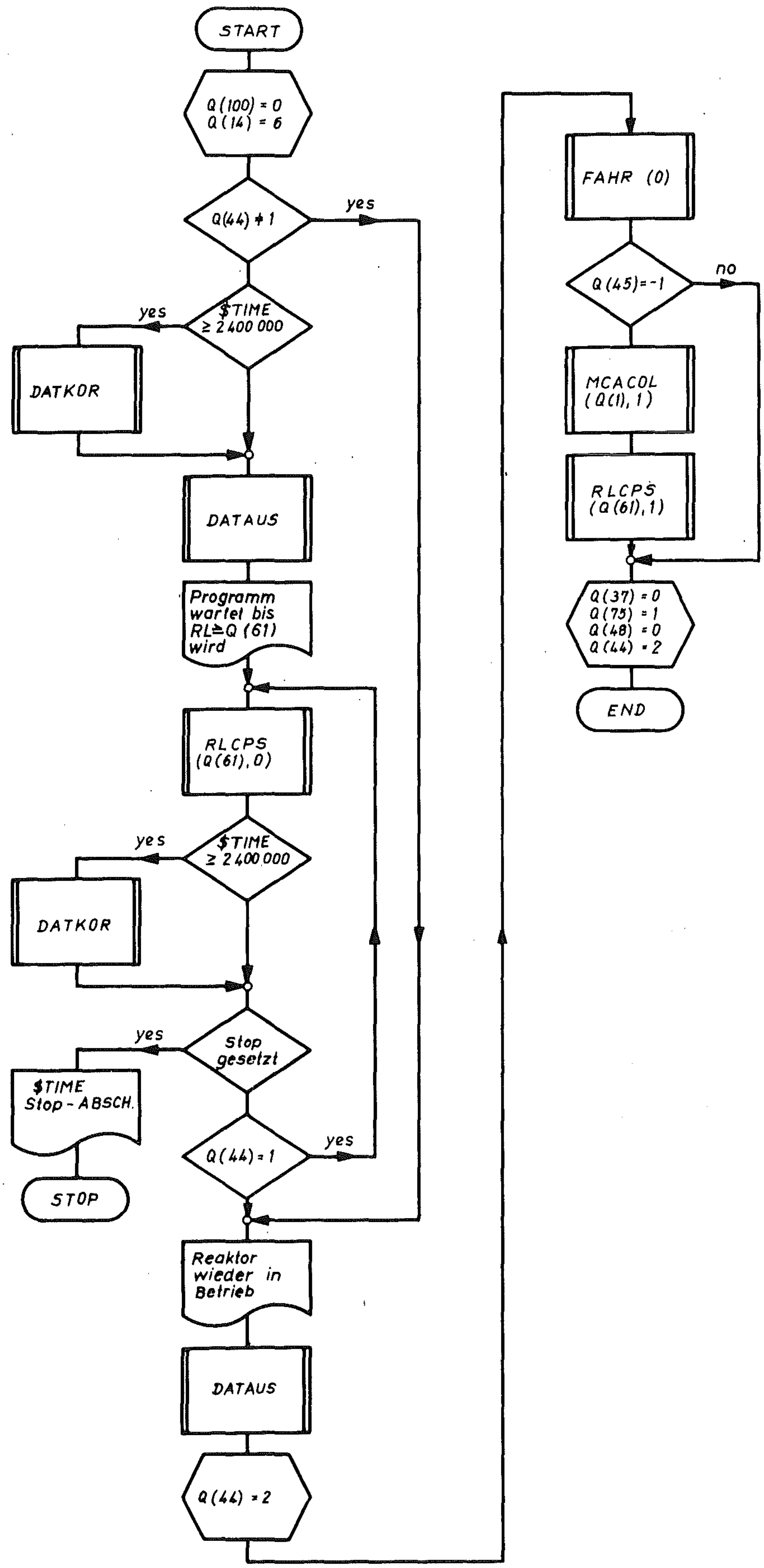
SONDER


START
define WACH
Q (59) = 0
Q (60) = 0
Q (14) = 2
declare array EIF (200)
EIORG (2)
FAHR (3)
MCACOL (Q (54), 1)
RLCPS (Q (61), 1)
Q (42) = Q (9)
Q (59) = 1
EIF(200) = Q (15)
compute end of acquisition time
RLCPS (Q (61), 0)
Q (44) ≥ 1 u. Q (81) = 0
yes
measurment ready
no
MCA ready
no
Q (63) = $TIME
RLCPS (Q (61), 2)
Q (59) = 2
FAHR (0)
MCACOL (10, 0)
WACH
TAPE (0)
Q (44) ≥ 1 u. Q (81) = 0
yes
WACH
FENAUS
EIKOF
EIORG (1)
Q (59) = 3
Q (41) = 0
Q (81) = 0
RLCPS (Q (61), 0)
Q(44) ≥ 1
yes
Q(58) = Q(5)
WACH
EIPRO
Q (59) = 4
Q (58) = 0
Q(44) ≥ 1
yes
Prog. Eichen abgebrochen
WACH
PLOT
erase array EIF
MCA display on
Q(88) = Q(72)
K = 59
erase array EIF
TAPE (1)
WACH delete
END

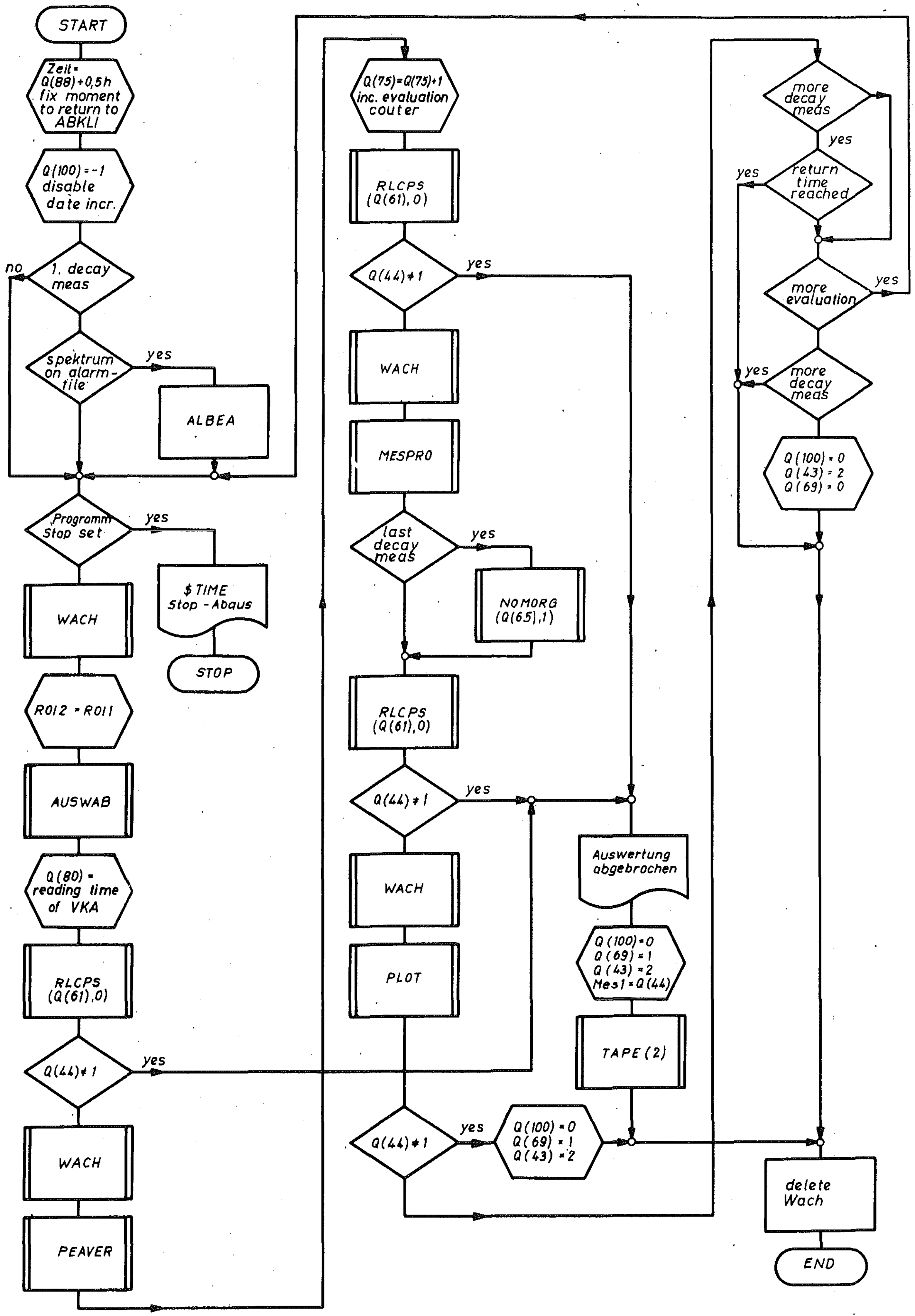
EICHEN


START
ROI 2 → cass.
erase ROI 2
define SPEK
1. decay measurement
yes
Abkling.-messungen
UHRAUS (Q(88), 1)
acquisition time = 400 s
MCACOL (Q(76); 1)
1. measurement runnig
yes
2,3,4,7,8 meas. ready
no
MCLES
change MCA-half for acquisition
change MCA-half for read
calculate and of acquisition time
RLCPS (Q(61), 0)
interrupt. decay meas.
yes
MCA ready
no
RL and CPS ⇒ SPEK
meas. ready
yes
meas. ready
no
1
acquisition time = 1000 s
set 2. half of MCA for acquisition
RLCPS (Q(61), 0)
interrupt decay meas.
yes
9. meas. ready and evaluation ready
yes
3,6,7 meas. ready
no
Stop set
no
$TIME Stop Abkling.
STOP
waiting time ready
no
Date right
yes
DATKOR
FAHR (0)
Abklingmess. abgebrochen
next meas. started
no
angefangene Messung auf Tape
UHRAUS
FAHR (0)
colli in right posi.
no
MCACOL (Q(1), 1)
RLCPS (Q(61), 1)
array SPECK defined
no
SPEK erase
define ROI 2 and load from cassette
END
1
10. meas. ready
yes
acquisition time = 10 000 s
3. meas. ready
yes
Pause = 30'
8. meas. ready
yes
Pause = 1 h
9. meas. ready
yes
Pause = 4 h
11. meas. ready
yes
Pause = 12 h
12. meas. ready
no
MCLES
calculate take off time for next spektr.
nächste Messung um .... Uhr
3.,6.,7. meas. ready
yes
change MCA-half for reading
MCLES

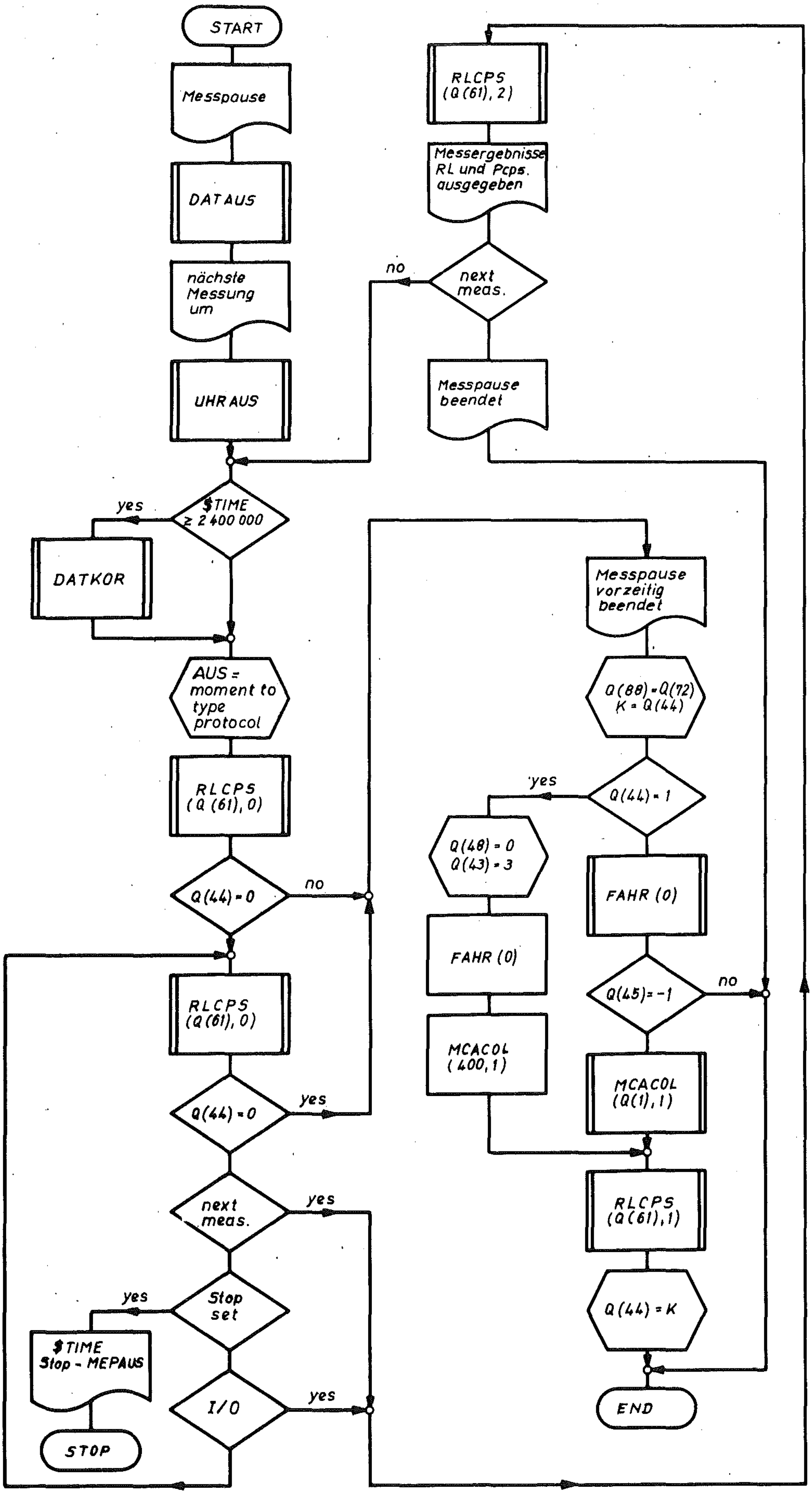
ABKLI


START
Q (100) = 0
Q (14) = 6
Q(44) ≠ 1
yes
$TIME ≥ 2 400 000
yes
DATKOR
DATAUS
Programm wartet bis RL≥Q (61) wird
RLCPS (Q(61), 0)
$TIME ≥ 2 400 000
yes
DATKOR
Stop gesetzt
yes
$TIME Stop - ABSCH.
STOP
Q(44) = 1
yes
Reaktor wieder in Betrieb
DATAUS
Q (44) = 2
FAHR (0)
Q (45) = -1
no
MCACOL (Q(1), 1)
RLCPS (Q (61), 1)
Q (37) = 0
Q (75) = 1
Q (48) = 0
Q (44) = 2
END

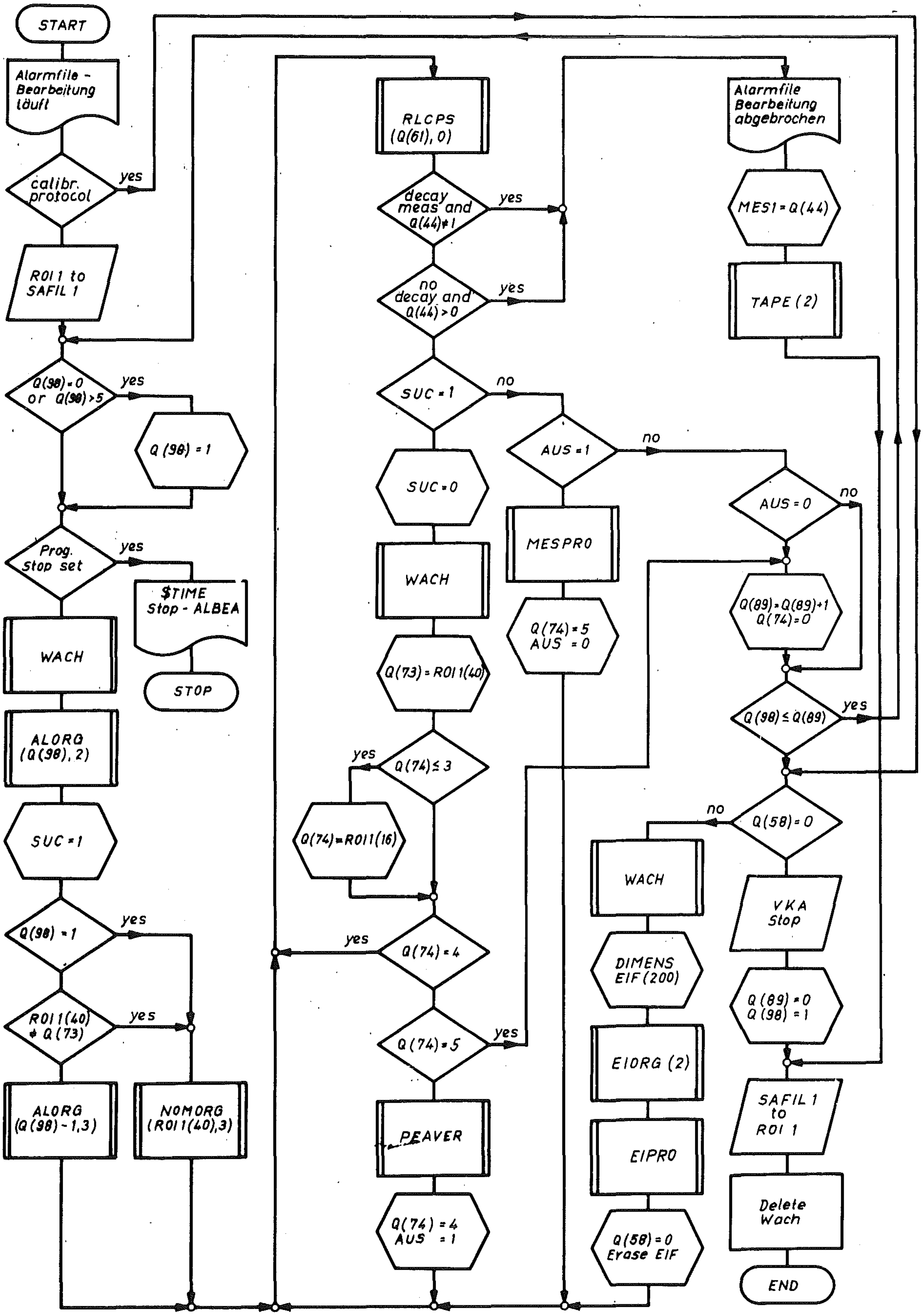
ABSCH


START
Zeit= Q(88)+0,5h fix moment to return to ABKLI
Q(100)=-1 disable date incr.
1. decay meas
no
spektrum on alarm-file
yes
ALBEA
Programm Stop set
yes
$TIME Stop - Abaus
STOP
WACH
R012 = R011
AUSWAB
Q(80) = reading time of VKA
RLCPS (Q(61),0)
Q(44)≠1
yes
WACH
PEAVER
Q(75)=Q(75)+1 inc. evaluation couter
RLCPS (Q(61),0)
Q(44)≠1
yes
WACH
MESPRO
last decay meas
yes
NOMORG (Q(65),1)
RLCPS (Q(61),0)
Q(44)≠1
yes
WACH
PLOT
Q(44)≠1
yes
Q(100)=0 Q(69)=1 Q(43)=2
Auswertung abgebrochen
Q(100)=0 Q(69)=1 Q(43)=2 Mes1=Q(44)
TAPE(2)
more decay meas
yes
return time reached
yes
more evaluation
yes
more decay meas
yes
Q(100)=0 Q(43)=2 Q(69)=0
delete Wach
END

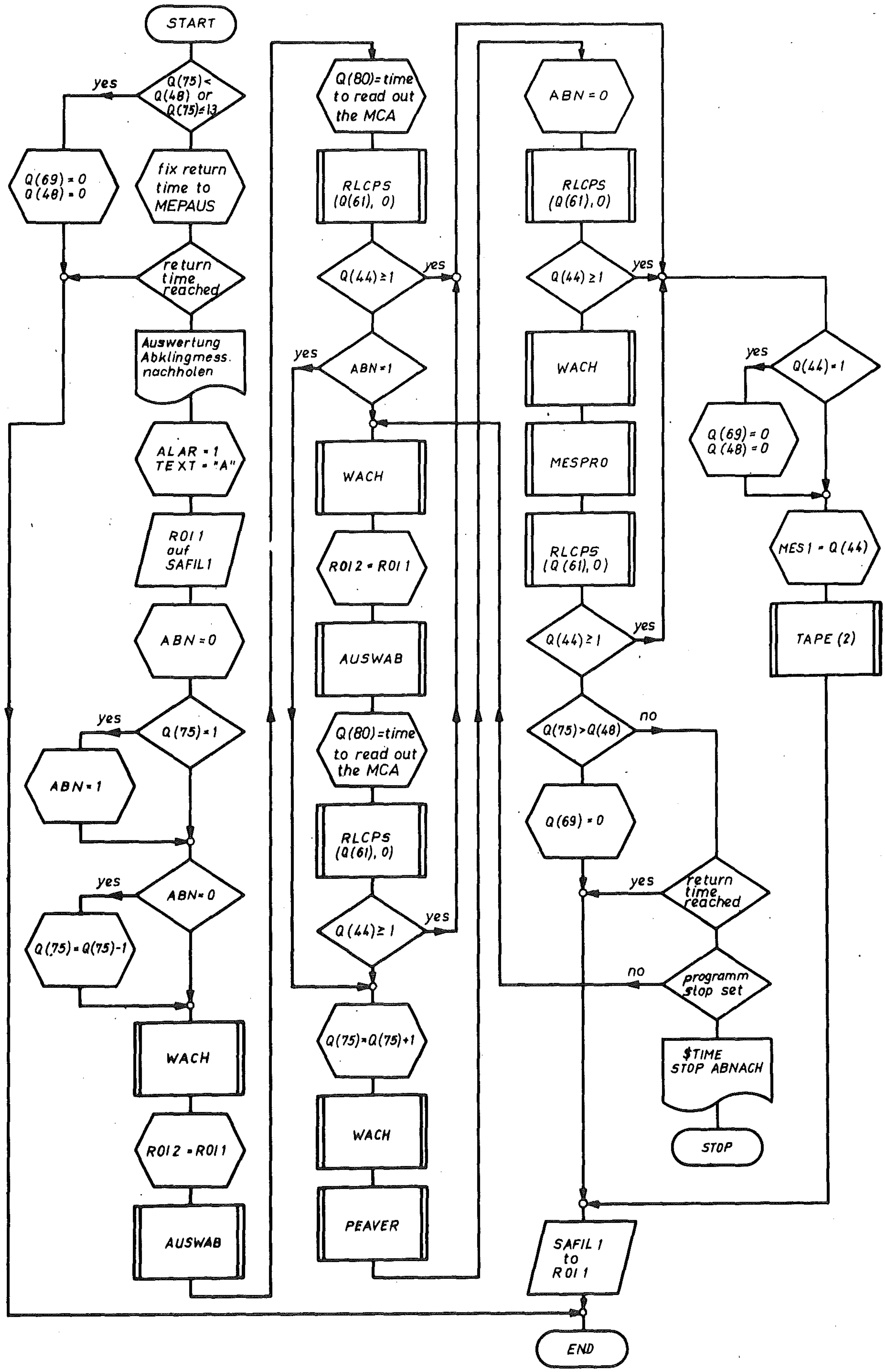
ABAUS


START
Messpause
DATAUS
nächste Messung um
UHRAUS
yes
$TIME ≥ 2 400 000
DATKOR
AUS = moment to type protocol
RLCPS (Q (61), 0)
Q(44) = 0
no
RLCPS (Q(61), 0)
Q(44) = 0
yes
next meas.
yes
Stop set
yes
$TIME Stop - MEPAUS
STOP
I/O
yes
RLCPS (Q(61), 2)
Messergebnisse RL und Pcps. ausgegeben
next meas.
no
Messpause beendet
Messpause vorzeitig beendet
Q(88) = Q(72) K = Q(44)
Q(44) = 1
yes
Q(48) = 0 Q(43) = 3
FAHR (0)
MCACOL (400,1)
FAHR (0)
Q(45) = -1
no
MCACOL (Q(1), 1)
RLCPS (Q(61),1)
Q(44) = K
END

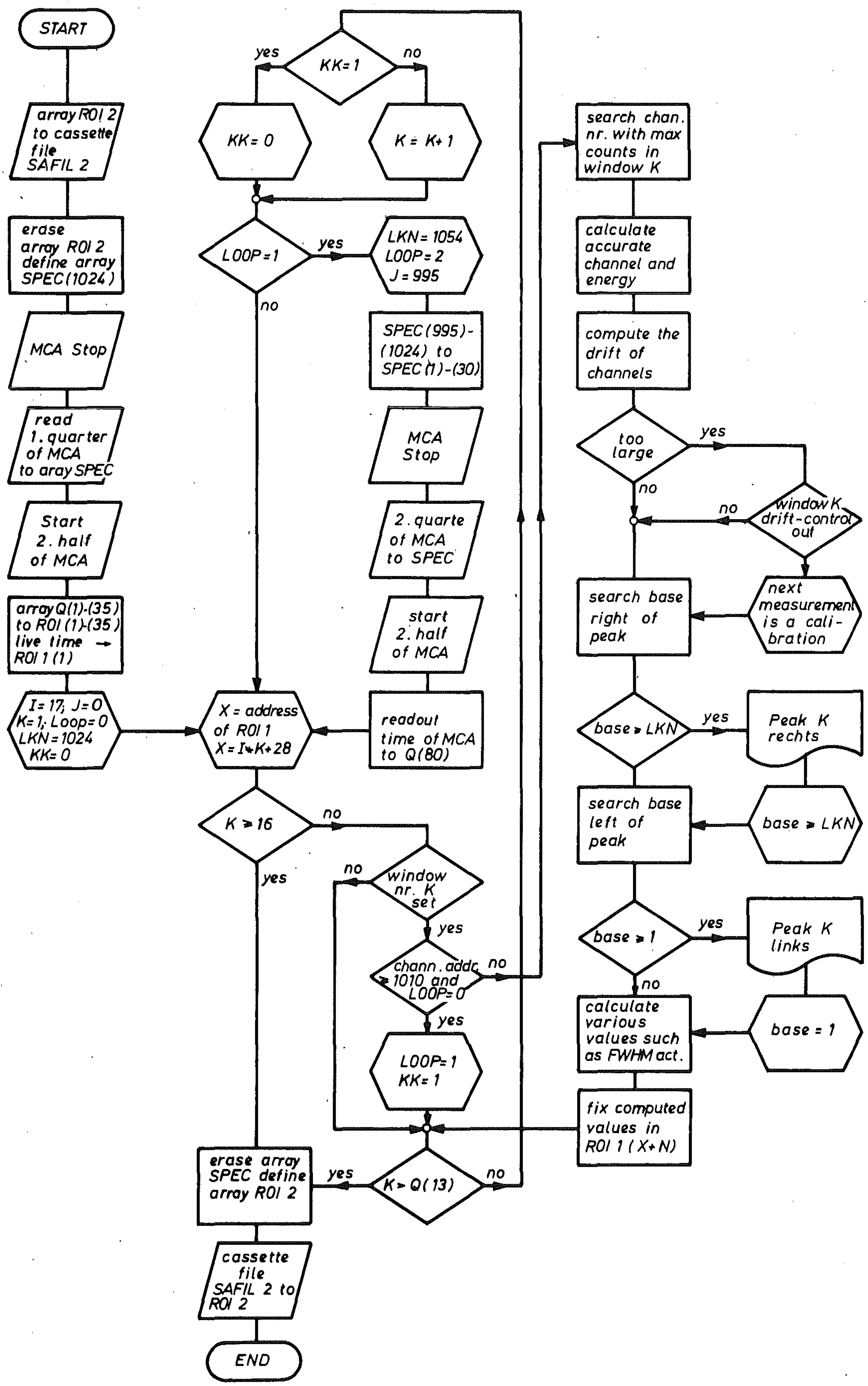
MEPAUS


START
Alarmfile - Bearbeitung läuft
calibr. protocol
yes
ROI 1 to SAFIL 1
Q(98)=0 or Q(98)>5
yes
Q (98) = 1
Prog. Stop set
yes
$TIME Stop - ALBEA
STOP
WACH
ALORG (Q(98), 2)
SUC = 1
Q(98) = 1
yes
ROI 1(40) ≠ Q (73)
yes
ALORG (Q(98) - 1,3)
NOMORG (ROI 1(40),3)
RLCPS (Q(61), 0)
decay meas and Q(44)≠1
yes
no decay and Q(44) > 0
yes
SUC = 1
no
SUC = 0
WACH
Q(73) = ROI 1(40)
yes
Q(74) ≤ 3
Q(74) = ROI 1(16)
yes
Q(74) = 4
Q(74) = 5
yes
PEAVER
Q(74) = 4 AUS = 1
AUS = 1
no
MESPRO
Q(74) = 5 AUS = 0
Alarmfile Bearbeitung abgebrochen
MES1 = Q(44)
TAPE (2)
AUS = 0
no
Q(89) = Q(89)+1 Q(74) = 0
Q(98) ≤ Q(89)
yes
Q(58) = 0
no
WACH
DIMENS EIF (200)
EIORG (2)
EIPRO
Q(58) = 0 Erase EIF
VKA Stop
Q(89) = 0 Q(98) = 1
SAFIL 1 to ROI 1
Delete Wach
END

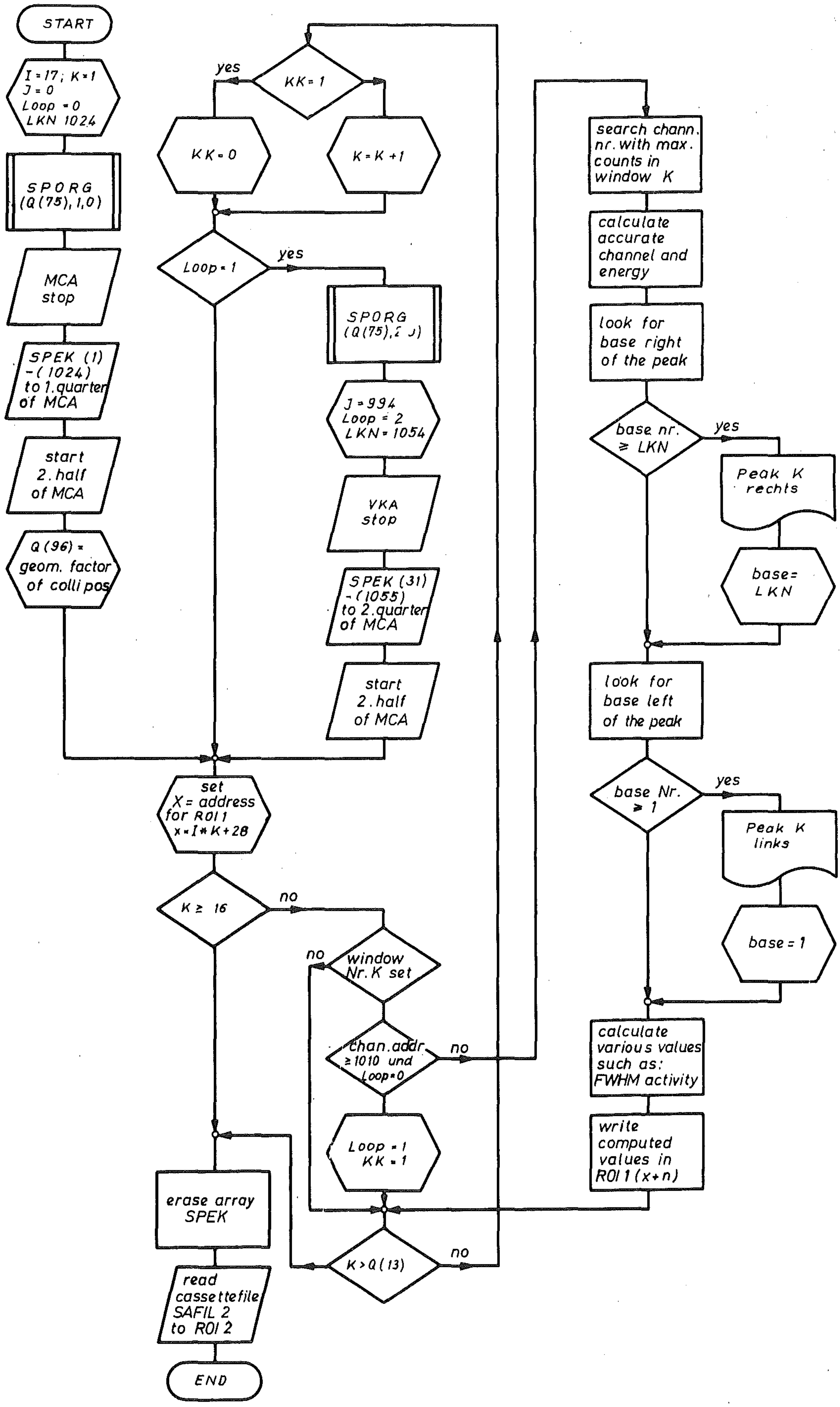
ALBEA


START
Q(75)< Q(48) or Q(75)≤13
yes
Q(69) = 0
Q(48) = 0
fix return time to MEPAUS
return time reached
Auswertung Abklingmess. nachholen
ALAR = 1
TEXT = "A"
ROI 1 auf SAFIL 1
ABN = 0
Q(75) = 1
yes
ABN = 1
ABN = 0
yes
Q(75) = Q(75)-1
WACH
ROI 2 = ROI 1
AUSWAB
Q(80) = time to read out the MCA
RLCPS (Q(61), 0)
Q(44) ≥ 1
yes
ABN = 1
yes
WACH
ROI 2 = ROI 1
AUSWAB
Q(80) = time to read out the MCA
RLCPS (Q(61), 0)
Q(44) ≥ 1
yes
Q(75) = Q(75)+1
WACH
PEAVER
ABN = 0
RLCPS (Q(61), 0)
Q(44) ≥ 1
yes
WACH
MESPRO
RLCPS (Q(61), 0)
Q(44) ≥ 1
yes
Q(75) > Q(48)
no
Q(69) = 0
return time reached
yes
programm stop set
no
$TIME STOP ABNACH
STOP
SAFIL 1 to ROI 1
END
Q(44) = 1
yes
Q(69) = 0
Q(48) = 0
MES 1 = Q(44)
TAPE (2)

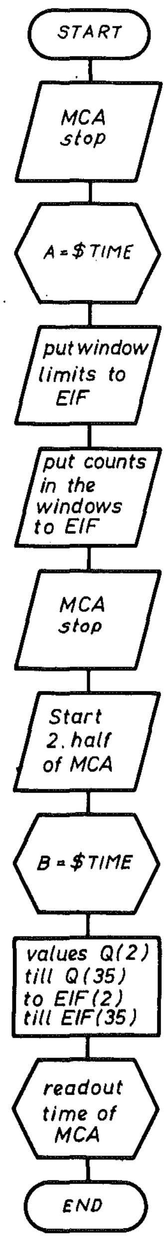
ABNACH


START
array ROI 2 to cassette file SAFIL 2
erase array ROI 2 define array SPEC(1024)
MCA Stop
read 1. quarter of MCA to aray SPEC
Start 2. half of MCA
array Q(1)-(35) to ROI (1)-(35) live time → ROI 1 (1)
I=17; J=0 K=1; Loop=0 LKN=1024 KK=0
KK=1
yes
no
KK=0
K=K+1
LOOP=1
yes
no
LKN=1054 LOOP=2 J=995
SPEC(995)-(1024) to SPEC(1)-(30)
MCA Stop
2. quarte of MCA to SPEC
start 2. half of MCA
readout time of MCA to Q(80)
X = address of ROI 1 X=I*K+28
K ≥ 16
no
yes
window nr. K set
no
yes
chann. addr. ≥ 1010 and LOOP=0
no
yes
LOOP=1 KK=1
K>Q(13)
yes
no
erase array SPEC define array ROI 2
cassette file SAFIL 2 to ROI 2
END
search chan. nr. with max counts in window K
calculate accurate channel and energy
compute the drift of channels
too large
yes
no
window K drift-control out
no
next measurement is a calibration
search base right of peak
base ≥ LKN
yes
Peak K rechts
base ≥ LKN
search base left of peak
base ≥ 1
yes
no
Peak K links
base = 1
calculate various values such as FWHM act.
fix computed values in ROI 1 (X+N)

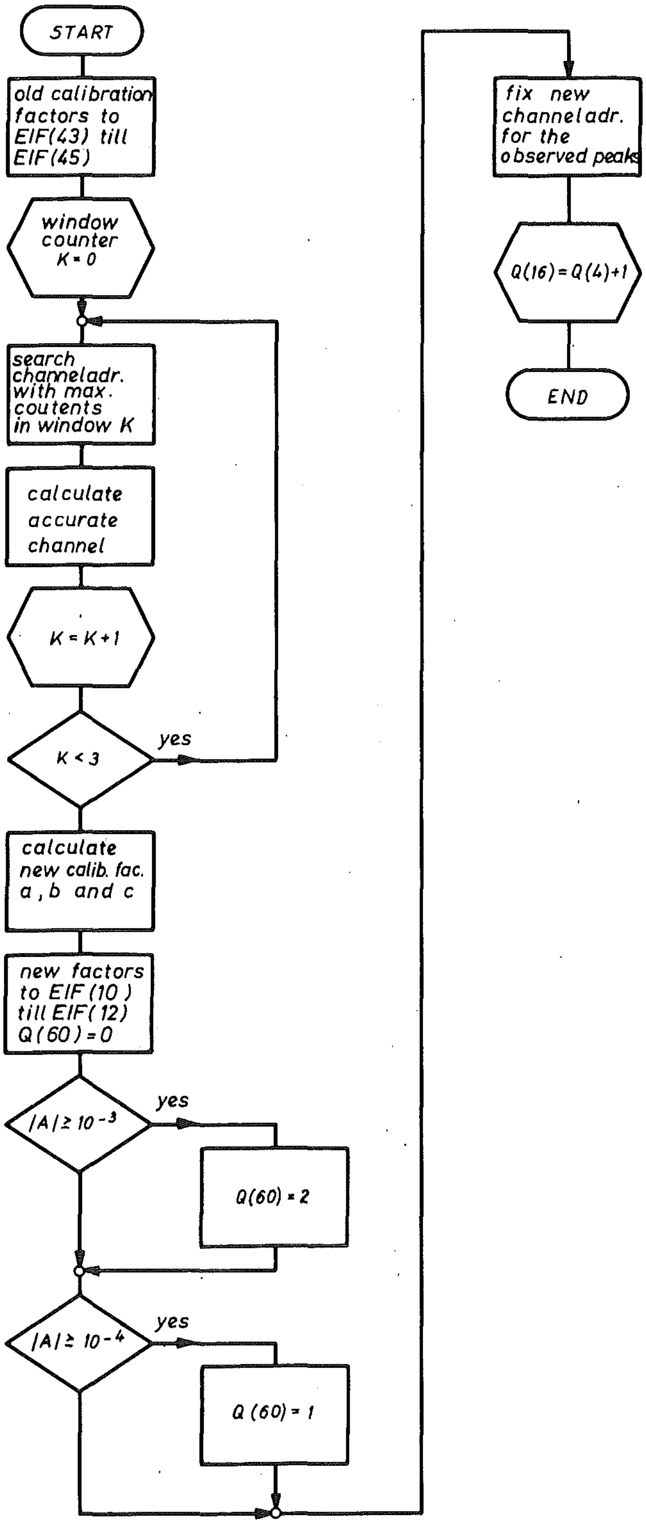
AUSWER

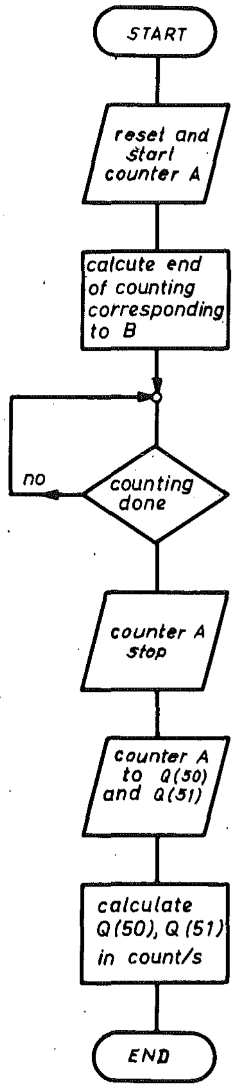

START
I = 17; K = 1
J = 0
Loop = 0
LKN 1024
SPORG
(Q(75), 1,0)
MCA
stop
SPEK (1)
-(1024)
to 1. quarter
of MCA
start
2. half
of MCA
Q (96) =
geom. factor
of colli pos
KK = 1
yes
KK = 0
K = K + 1
Loop = 1
yes
SPORG
(Q(75), 2 J)
J = 994
Loop = 2
LKN = 1054
VKA
stop
SPEK (31)
-(1055)
to 2. quarter
of MCA
start
2. half
of MCA
set
X = address
for ROI 1
x = I * K + 28
K ≥ 16
no
window
Nr. K set
no
chan. addr.
≥ 1010 und
Loop = 0
no
Loop = 1
KK = 1
K > Q(13)
no
erase array
SPEK
read
cassettefile
SAFIL 2
to ROI 2
END
search chann.
nr. with max.
counts in
window K
calculate
accurate
channel and
energy
look for
base right
of the peak
base nr.
≥ LKN
yes
Peak K
rechts
base =
LKN
look for
base left
of the peak
base Nr.
≥ 1
yes
Peak K
links
base = 1
calculate
various values
such as:
FWHM activity
write
computed
values in
ROI 1 (x+n)

AUSWAB

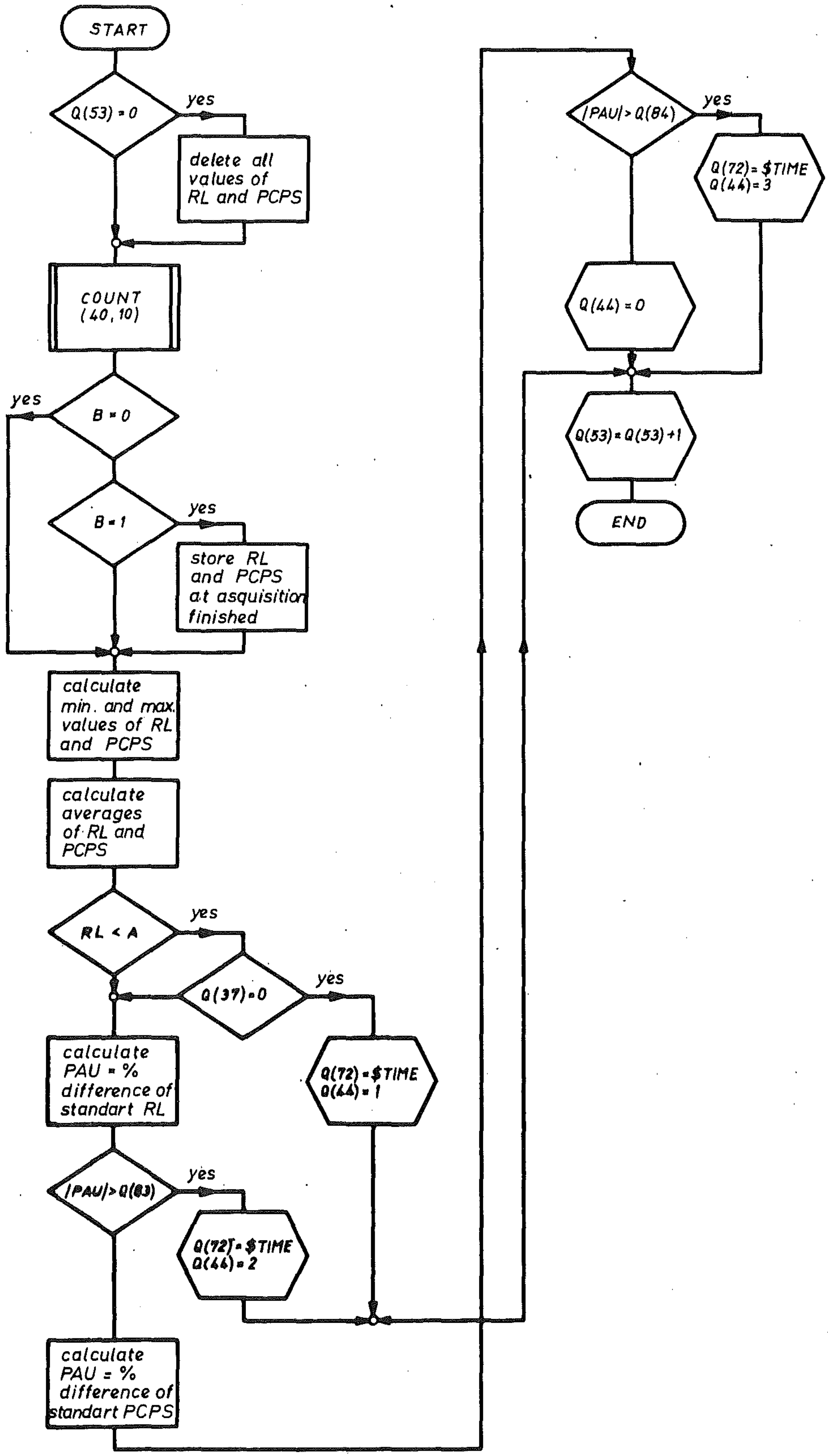

START
MCA stop
A = $TIME
put window limits to EIF
put counts in the windows to EIF
MCA stop
Start 2. half of MCA
B = $TIME
values Q(2) till Q(35) to EIF(2) till EIF(35)
readout time of MCA
END

FENAUS

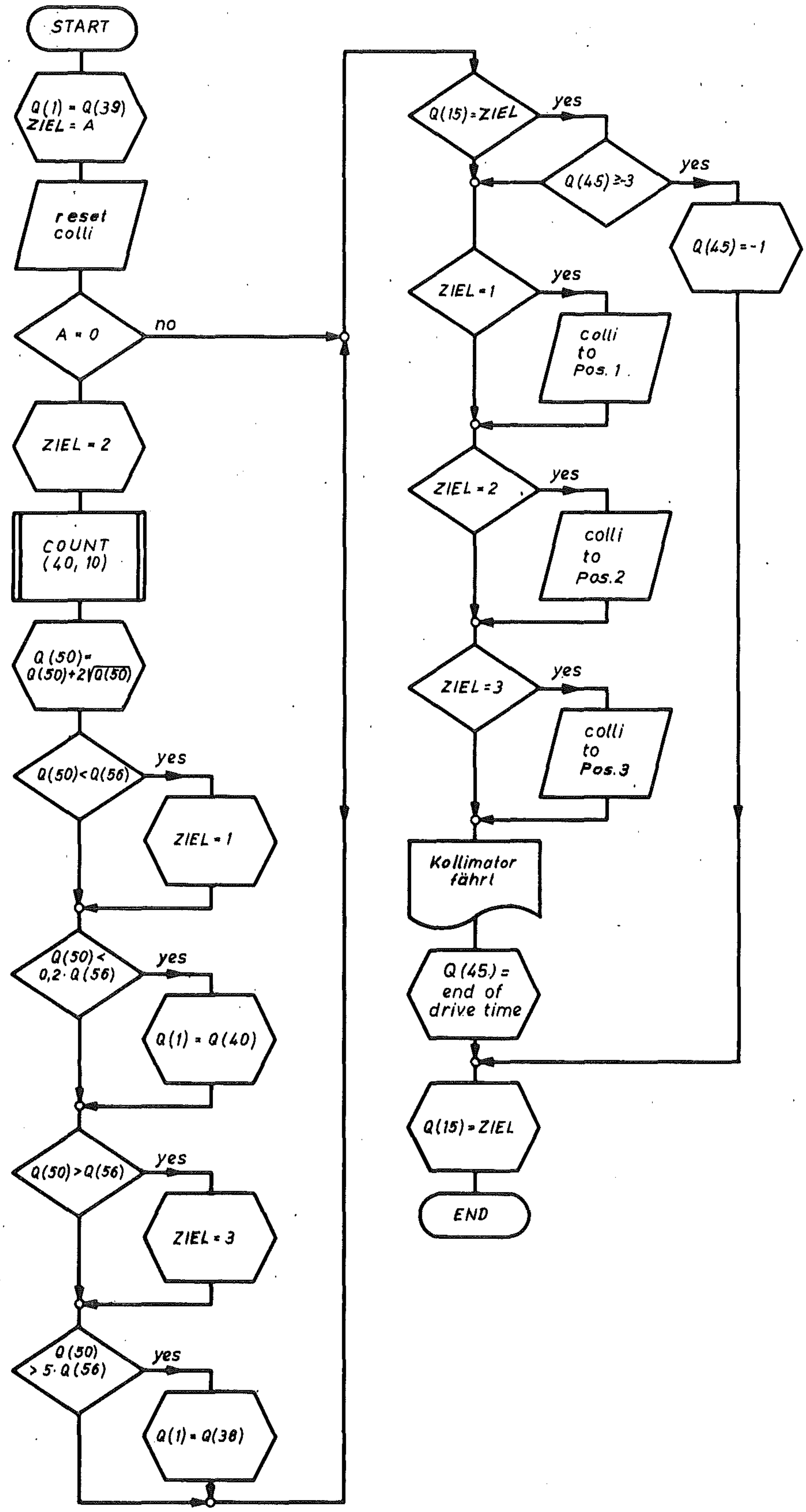

START
old calibration factors to EIF(43) till EIF(45)
window counter K = 0
search channeladr. with max. coutents in window K
calculate accurate channel
K = K+1
K < 3
yes
calculate new calib. fac. a, b and c
new factors to EIF(10) till EIF(12) Q(60) = 0
|A| ≥ 10⁻³
yes
Q(60) = 2
|A| ≥ 10⁻⁴
yes
Q (60) = 1
fix new channeladr. for the observed peaks
Q(16) = Q(4)+1
END


START
reset and start counter A
calcute end of counting corresponding to B
no
counting done
counter A stop
counter A to Q(50) and Q(51)
calculate Q(50), Q(51) in count/s
END

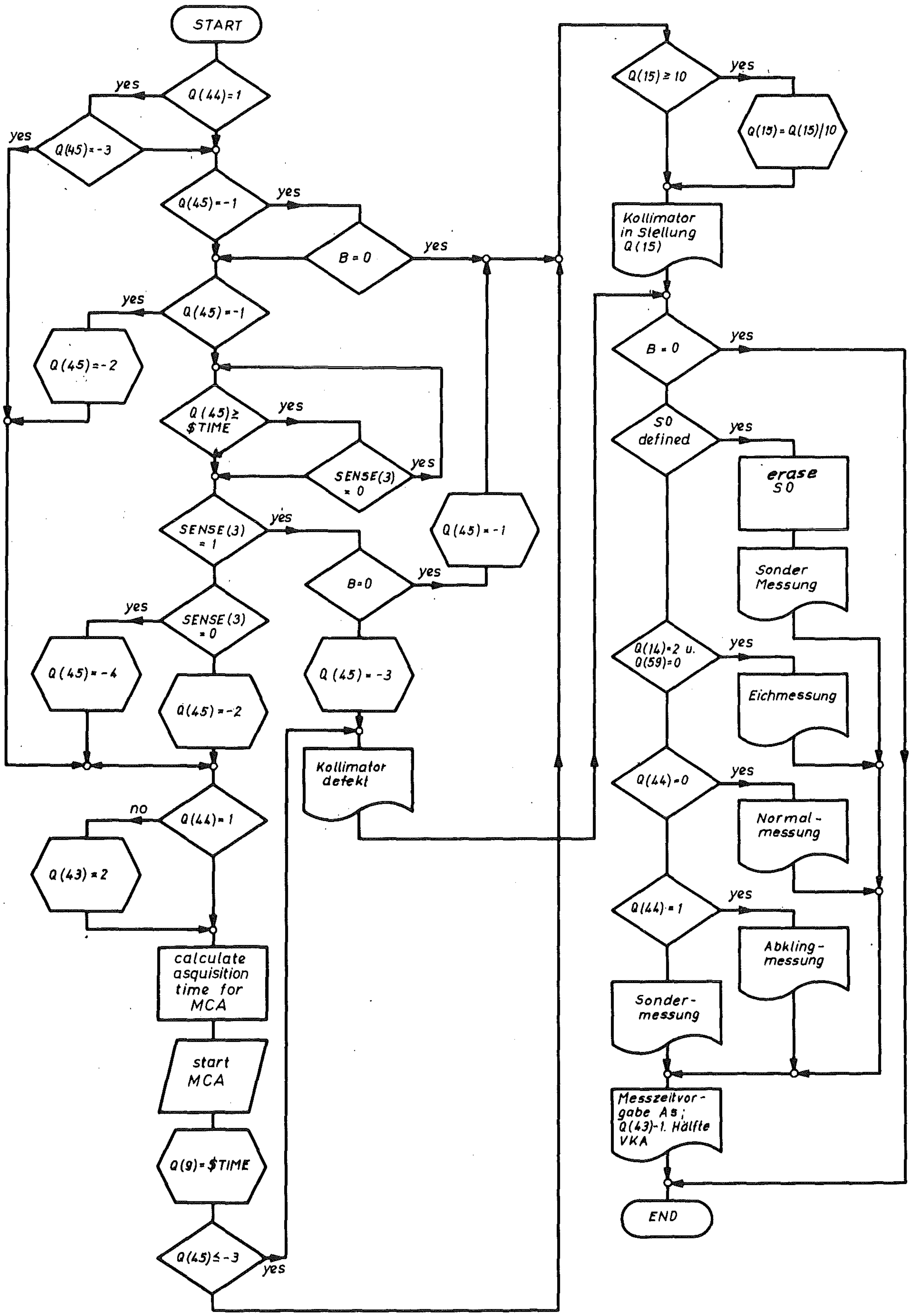
COUNT (A,B)


START
Q(53) = 0
yes
delete all values of RL and PCPS
COUNT (40,10)
B = 0
yes
B = 1
yes
store RL and PCPS at asquisition finished
calculate min. and max. values of RL and PCPS
calculate averages of RL and PCPS
RL < A
yes
Q(37)=0
yes
Q(72) = $TIME Q(44) = 1
calculate PAU = % difference of standart RL
|PAU| > Q(83)
yes
Q(72) = $TIME Q(44) = 2
calculate PAU = % difference of standart PCPS
|PAU| > Q(84)
yes
Q(72) = $TIME Q(44) = 3
Q(44) = 0
Q(53) = Q(53) + 1
END

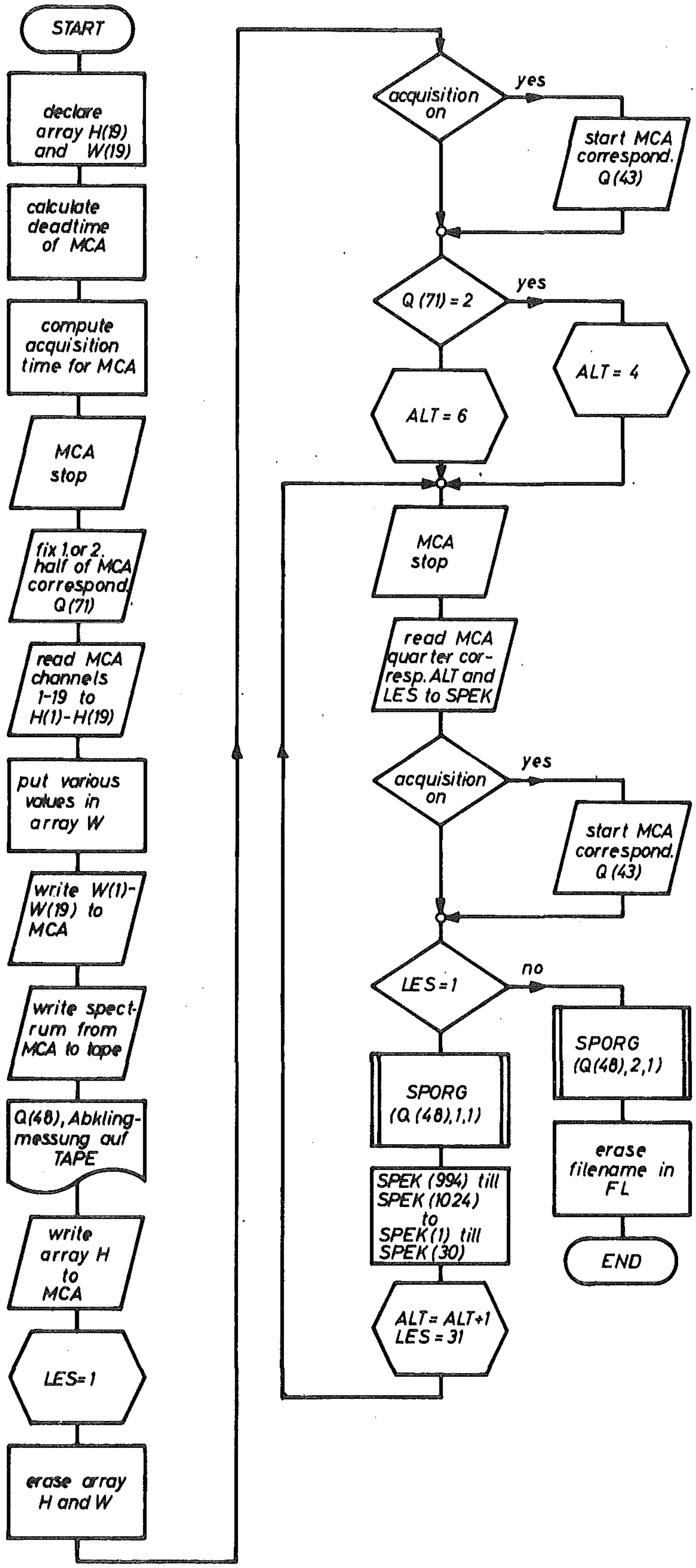
RLCPS (A,B)


START
Q(1) = Q(39)
ZIEL = A
reset colli
A = 0
no
ZIEL = 2
COUNT (40, 10)
Q(50)= Q(50)+2√Q(50)
Q(50)<Q(56)
yes
ZIEL = 1
Q(50) < 0,2·Q(56)
yes
Q(1) = Q(40)
Q(50) > Q(56)
yes
ZIEL = 3
Q(50) > 5·Q(56)
yes
Q(1) = Q(38)
Q(15) = ZIEL
yes
Q(45) ≥ -3
yes
Q(45) = -1
ZIEL = 1
yes
colli to Pos. 1
ZIEL = 2
yes
colli to Pos. 2
ZIEL = 3
yes
colli to Pos. 3
Kollimator fährt
Q(45) = end of drive time
Q(15) = ZIEL
END

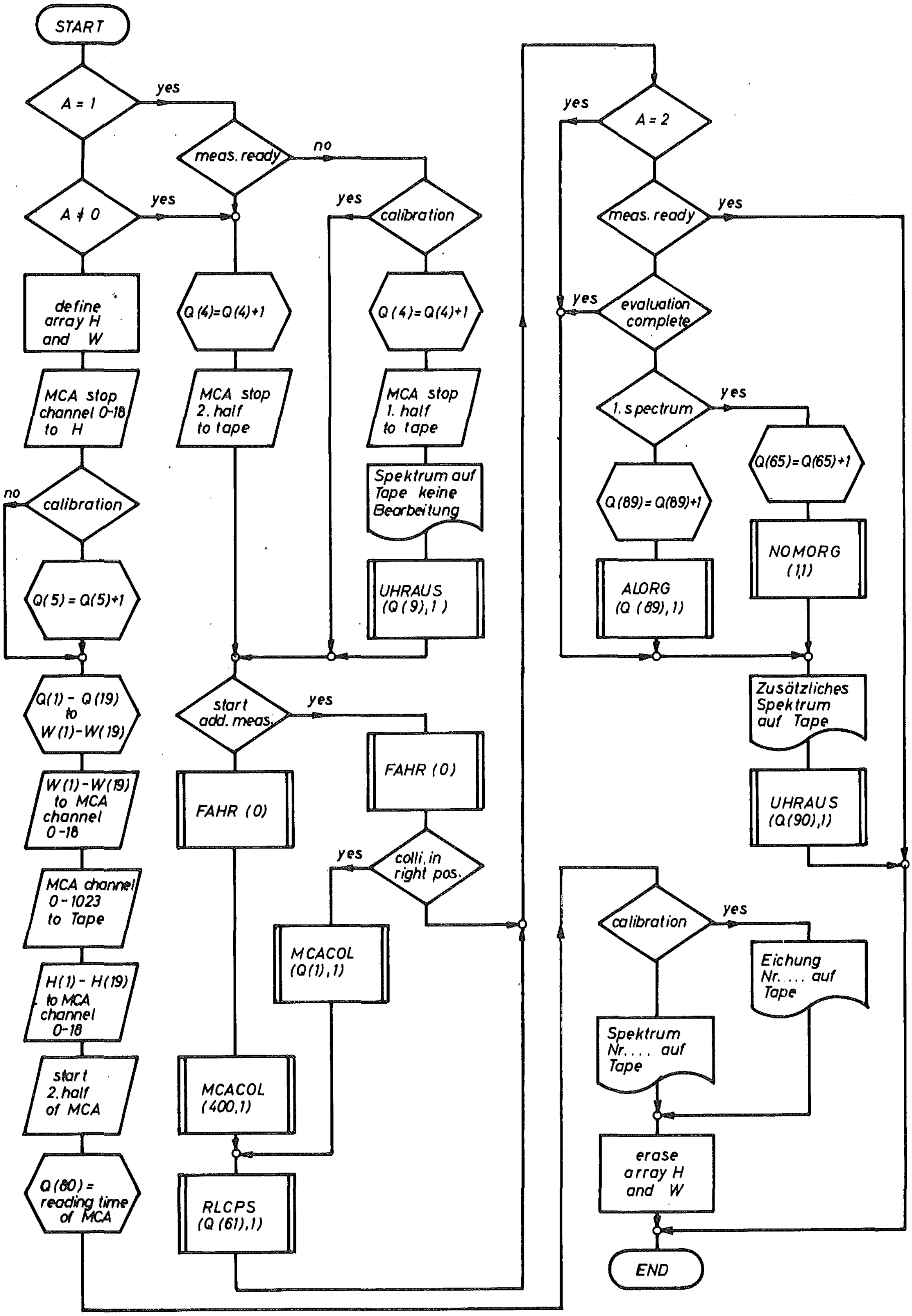
FAHR (A)


START
Q(44)=1
yes
Q(45)=-3
yes
Q(45)=-1
yes
B=0
yes
Q(45)=-1
yes
Q(45)=-2
Q(45)≥ $TIME
yes
SENSE(3) = 0
yes
SENSE(3) = 1
yes
B=0
yes
Q(45)=-1
SENSE(3) = 0
yes
Q(45)=-4
Q(45)=-2
Q(45)=-3
Kollimator defekt
Q(44)=1
no
Q(43)=2
calculate asquisition time for MCA
start MCA
Q(9)=$TIME
Q(45)≤-3
yes
Q(15)≥10
yes
Q(15)=Q(15)/10
Kollimator in Stellung Q(15)
B=0
yes
S0 defined
yes
erase S0
Sonder Messung
Q(14)=2 u. Q(59)=0
yes
Eichmessung
Q(44)=0
yes
Normal-messung
Q(44)=1
yes
Abkling-messung
Sonder-messung
Messzeitvor-gabe As; Q(43)-1. Hälfte VKA
END

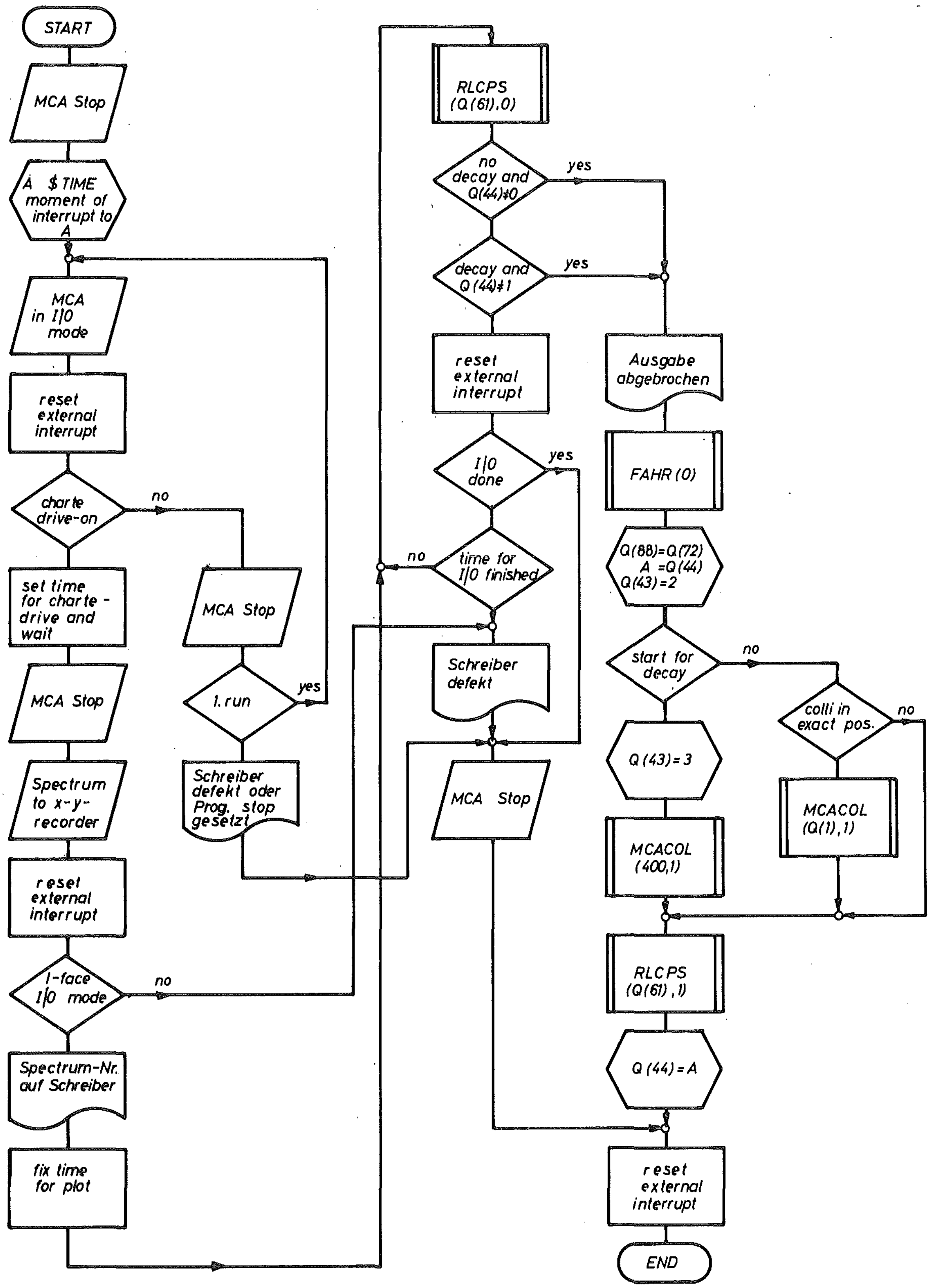
MCACOL (A,B)


START
declare array H(19) and W(19)
calculate deadtime of MCA
compute acquisition time for MCA
MCA stop
fix 1.or 2. half of MCA correspond. Q(71)
read MCA channels 1-19 to H(1)-H(19)
put various values in array W
write W(1)-W(19) to MCA
write spectrum from MCA to tape
Q(48), Abklingmessung auf TAPE
write array H to MCA
LES=1
erase array H and W
acquisition on
yes
start MCA correspond. Q(43)
Q (71) = 2
yes
ALT = 4
ALT = 6
MCA stop
read MCA quarter corresp. ALT and LES to SPEK
acquisition on
yes
start MCA correspond. Q (43)
LES = 1
no
SPORG (Q(48),2,1)
erase filename in FL
END
SPORG (Q (48),1,1)
SPEK (994) till SPEK (1024) to SPEK (1) till SPEK (30)
ALT = ALT+1 LES = 31

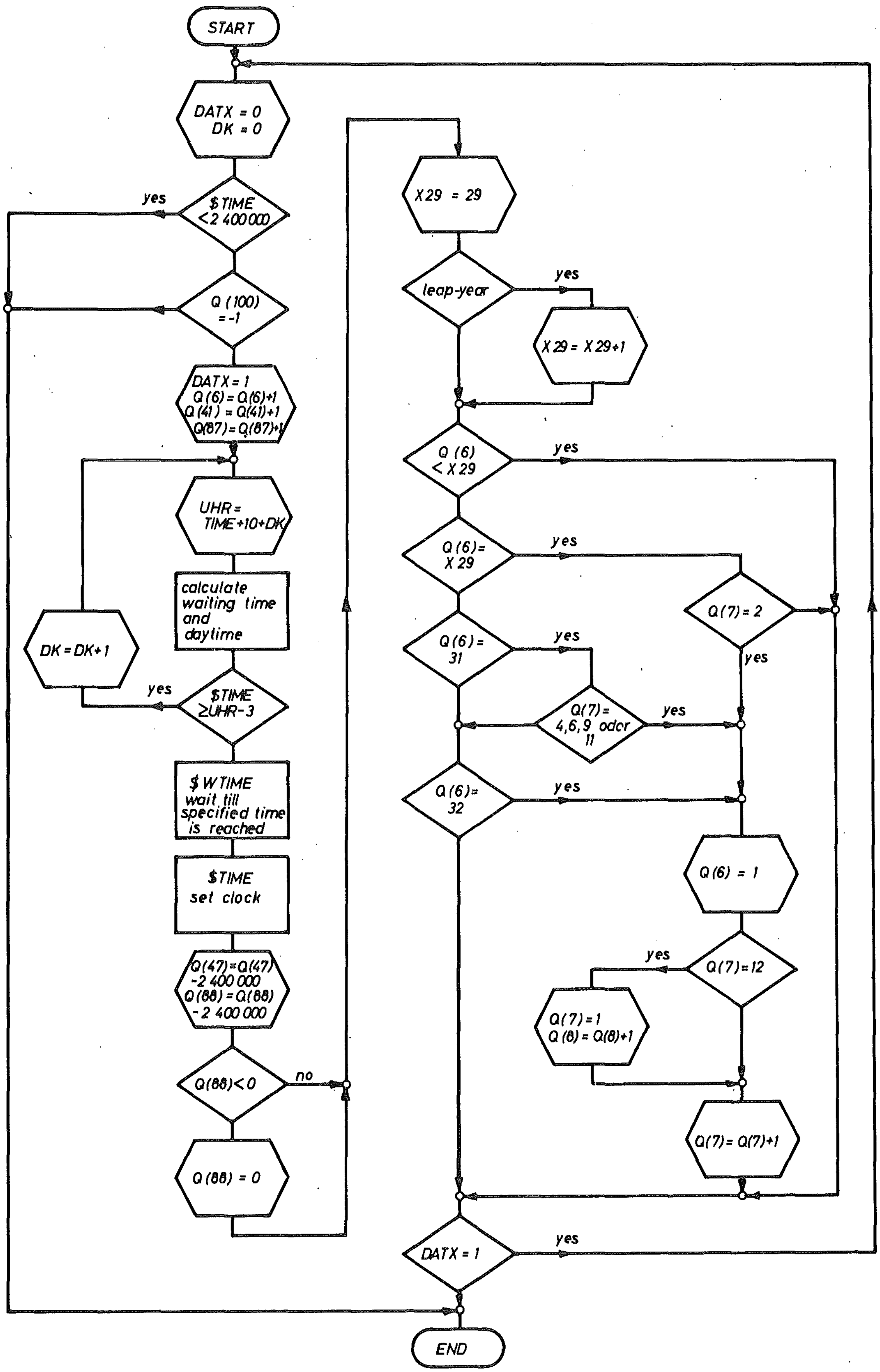
MCLES


START
A = 1
yes
A ≠ 0
yes
define array H and W
MCA stop channel 0-18 to H
calibration
no
Q(5) = Q(5)+1
Q(1) - Q(19) to W(1)-W(19)
W(1)-W(19) to MCA channel 0-18
MCA channel 0-1023 to Tape
H(1)- H(19) to MCA channel 0-18
start 2. half of MCA
Q(60) = reading time of MCA
meas. ready
no
Q(4)=Q(4)+1
MCA stop 2. half to tape
calibration
yes
Q(4)=Q(4)+1
MCA stop 1. half to tape
Spektrum auf Tape keine Bearbeitung
UHRAUS (Q(9),1)
start add. meas.
yes
FAHR (0)
FAHR (0)
colli. in right pos.
yes
MCACOL (Q(1),1)
MCACOL (400,1)
RLCPS (Q(61),1)
A = 2
yes
meas. ready
yes
evaluation complete
yes
1. spectrum
yes
Q(89)=Q(89)+1
ALORG (Q(89),1)
Q(65)=Q(65)+1
NOMORG (1,1)
Zusätzliches Spektrum auf Tape
UHRAUS (Q(90),1)
calibration
yes
Spektrum Nr.... auf Tape
Eichung Nr.... auf Tape
erase array H and W
END

TAPE (A)


START
MCA Stop
A $ TIME moment of interrupt to A
MCA in I/O mode
reset external interrupt
charte drive-on
no
set time for charte - drive and wait
MCA Stop
1. run
yes
MCA Stop
Schreiber defekt oder Prog. stop gesetzt
Spectrum to x-y-recorder
reset external interrupt
I-face I/O mode
no
Spectrum-Nr. auf Schreiber
fix time for plot
RLCPS (Q(61),0)
no decay and Q(44)≠0
yes
decay and Q(44)≠1
yes
reset external interrupt
I/O done
yes
time for I/O finished
no
Schreiber defekt
MCA Stop
Ausgabe abgebrochen
FAHR (0)
Q(88)=Q(72) A =Q(44) Q(43)=2
start for decay
no
colli in exact pos.
no
Q (43) = 3
MCACOL (400,1)
MCACOL (Q(1),1)
RLCPS (Q(61),1)
Q (44) = A
reset external interrupt
END

PLOT


START
DATX = 0
DK = 0
$TIME
<2 400 000
yes
Q (100)
= -1
DATX = 1
Q (6) = Q (6)+1
Q (41) = Q(41)+1
Q(87) = Q(87)+1
UHR =
TIME +10+DK
calculate
waiting time
and
daytime
DK = DK+1
$TIME
≥UHR-3
yes
$ WTIME
wait till
specified time
is reached
$TIME
set clock
Q (47)=Q(47)
-2 400 000
Q (88) = Q(88)
- 2 400 000
Q(88)<0
no
Q (88) = 0
X29 = 29
leap-year
yes
X29 = X29+1
Q (6)
< X29
yes
Q (6)=
X29
yes
Q (7) = 2
yes
Q (6)=
31
yes
Q(7)=
4,6,9 oder
11
yes
Q (6)=
32
yes
Q (6) = 1
Q (7) = 12
yes
Q (7) = 1
Q (8) = Q(8)+1
Q(7) = Q(7)+1
DATX = 1
yes
END

DATKOR

Example: Starting dialog (Operator input underlined)

```
*ONGAM 2

ONGAM VERS. Ø1.77 / 16K

UMRECHNUNGSFAKTOR FUER REAKTORLEISTUNG =  .Ø4ØØØØØØ RICHTIG ? JA

NULLEISTUNGSBETRIEB ?  NEIN

3Ø. 1. 1978                    ZEIT:  .ØØ

DATUM UND UHRZEIT RICHTIG ?  NEIN

DATUM EINGEBEN
TAG ?2
MONAT ? (IN ZIFFERN EINGEBEN)  2
JAHR ?1978
UHRZEIT EINGEBEN. STUNDE ?9
NACH EINGABE DER MINUTEN, AUF VOLLE MINUTE
MIT RETURN-TASTE STARTEN!
MINUTE ?34

IN WELCHER STELLUNG BEFINDET SICH DER KOLLIMATOR ?  1

FOLGENDE ENERGIEN WERDEN ZUR QUADRATISCHEN EICHUNG VERWENDET
PEAK 1    96.73ØØØØ  KEV
PEAK 2    897.98ØØØ  KEV
PEAK 3    1836.11ØØ  KEV

FOLGENDE PEAKS WERDEN UEBERWACHT.
AM 241    59.537ØØØ  KEV
SE 75     96.73ØØØØ  KEV
SE 75     121.15ØØØ  KEV
SE 75     136.ØØØØØ  KEV
SE 75     264.561ØØ  KEV
SE 75     279.527ØØ  KEV
SE 75     3Ø3.911ØØ  KEV
SE 75     4ØØ.644ØØ  KEV
VERN.     511.ØØØØØ  KEV
CS 137    661.63ØØØ  KEV
Y 88      897.98ØØØ  KEV
CO 6Ø     1173.22ØØ  KEV
CO 6Ø     1332.5ØØØ  KEV
Y 88      1836.11ØØ  KEV

SOLLEN AENDERUNGEN BEZUEGLICH DER EICHLINIEN BZW. DER ZU
UEBERWACHENDEN PEAKS VORGENOMMEN WERDEN ? JA
```

ZULAESSIGE BEFEHLE AUF DIE FRAGE WAS

I ,EINGABE
V ,VERSCHIEBEN VON FENSTERN
A ,AUSGABE
L ,LOESCHEN VON FENSTERN
E ,EICHLINIEN SETZEN
U ,UEBERTRAG VON ROI1 NACH ROI2
$,DIALOG BEENDEN

WAS ? A
VON FENSTER NR. 1
BIS FENSTER NR. 15

ABUNDANZ NEGATIV: PEAK WIRD NICHT ZUR SUMMENBILDUNG VERWENDET
FENSTERBREITE NEGATIV: PEAK WIRD NICHT ZUR DRIFTKONTROLLE VERWENDET

FENSTER NR.	NUCLID	ENERGIE	HWZ	ABUNDANZ	FENS.BREIT.
1.0000000	AM 241	59.537000	432.9Y	35.900000	-10.000000
2.0000000	SE 75	96.730000	120.4D	.16000000	-10.000000
3.0000000	SE 75	121.15000	120.4D	6.2000000	-12.000000
4.0000000	SE 75	136.00000	120.4D	55.600000	-12.000000
5.0000000	SE 75	264.56100	120.4D	59.100000	-14.000000
6.0000000	SE 75	279.52700	120.4D	25.000000	-14.000000
7.0000000	SE 75	303.91100	120.4D	1.3600000	-14.000000
8.0000000	SE 75	400.64400	120.4D	12.400000	-14.000000
9.0000000	VERN.	511.00000	XXXXXX	1.0000000	-14.000000
10.000000	CS 137	661.63000	30.5Y	85.100000	-14.000000
11.000000	Y 88	897.98000	107.4D	91.400000	20.000000
12.000000	CO 60	1173.2200	5.27Y	99.000000	-16.000000
13.000000	CO 60	1332.5000	5.27Y	99.100000	-16.000000
14.000000	Y 88	1836.1100	107.4D	99.400000	-14.000000
15.000000	@@@@@@	0.0000000	@@@@@@	0.0000000	@@@@@@

WAS ? V
VON FENSTER NR. 14
NACH FENSTER NR 15

WAS ? I
FENSTER NR. 14
NUCLIDNAME K 40
SOLL ENERGIE EINGEGEN WERDEN ? JA
ENERGIE IN KEV 1460.
HALBWERTSZEIT 1.26Y9
ABUNDANZ 11
FENSTERBREITE KANAELE ? 14
SOLL DIESER PEAK ZUR DRIFTUEBERWACHUNG VERWENDET WERDEN ? NEIN
SOLL DIESER PEAK ZUR SUMMENBILDUNG VERWENDET WERDEN ? NEIN

WAS ? A
VON FENSTER NR. 1
BIS FENSTER NR. 15

ABUNDANZ NEGATIV: PEAK WIRD NICHT ZUR SUMMENBILDUNG VERWENDET
FENSTERBREITE NEGATIV: PEAK WIRD NICHT ZUR DRIFTKONTROLLE VERWENDET

FENSTER NR.	NUCLID	ENERGIE	HWZ	ABUNDANZ	FENS.BREIT.
1.0000000	AM 241	59.537000	432.9Y	35.900000	-10.000000
2.0000000	SE 75	96.730000	120.4D	.16000000	-10.000000
3.0000000	SE 75	121.15000	120.4D	6.2000000	-12.000000
4.0000000	SE 75	136.00000	120.4D	55.600000	-12.000000
5.0000000	SE 75	264.56100	120.4D	59.100000	-14.000000
6.0000000	SE 75	279.52700	120.4D	25.000000	-14.000000
7.0000000	SE 75	303.91100	120.4D	1.3600000	-14.000000
8.0000000	SE 75	400.64400	120.4D	12.400000	-14.000000
9.0000000	VERN.	511.00000	XXXXXX	1.0000000	-14.000000
10.000000	CS 137	661.63000	30.5Y	85.100000	-14.000000
11.000000	Y 88	897.98000	107.4D	91.400000	20.000000
12.000000	CO 60	1173.2200	5.27Y	99.000000	-16.000000
13.000000	CO 60	1332.5000	5.27Y	99.100000	-16.000000
14.000000	K 40	1460.0000	1.26Y9	-11.000000	-14.000000
15.000000	Y 88	1836.1100	107.4D	99.400000	-14.000000

WAS ? $

EPS 1 = 1.3652611
EPS 2 = 1.4314618
EPS 3 = 1.4735532
EPS 4 = 1.4983144
EPS 5 = 1.6636638
EPS 6 = 1.6748678
EPS 7 = 1.6886589
EPS 8 = 1.6862577
EPS 9 = 1.5844900
EPS10 = 1.3576609
EPS11 = 1.0183598
EPS12 = .76501155
EPS13 = .67261714
EPS14 = .61742981
EPS15 = .51409527

SOLLEN DIE PROGRAMMSTEUERDATEN GELISTET WERDEN ? JA

PROGRAMM STARTET MIT FOLGENDEN WERTEN:

NR. 10. EICHFAKTOR A=-.00000040
NR. 11. EICHFAKTOR B= .99949098
NR. 12. EICHFAKTOR C=-.50941402
NR. 82. BEI EINER DRIFT DES SPEKTRUMS UM MEHR ALS 2 KANAELE
WIRD EINE EICHMESSUNG DURCHGEFUEHRT
NR. 46. SPAETESTENS ALLE 1 TAGE WIRD EINE EICHMESSUNG DURCHGEFUEHRT
NR. 79. SPAETESTENS ALLE 7 TAGE WIRD EIN SPEKTRUM
AUF LANGZEITFILE GELEGT
NR. 54. EICHZEIT= 1000.0000 S
NR. 38. MESSZEIT1= 1000.0000 S
NR. 39. MESSZEIT2= 1000.0000 S
NR. 40. MESSZEIT3= 1000.0000 S
NR. 52. MESSFOLGE= 28800.000 S
NR. 49. AUSGABEFOLGE WAEHREND MESSPAUSE= 7200.0000 S
NR. 56. GW1 FUER KOLLIMATOR-STEUERUNG= 300.00000 COUNTS/S
NR. 57. GW2 FUER KOLLIMATOR-STEUERUNG= 3000.0000 COUNTS/S
NR. 61. GW FUER ABKLINGMESSUNG= 2.0000000 MW

GEOMETRIEFAKTOREN FUER DIE KOLLIMATOR-STELLUNGEN

NR. 66. STELLUNG 1 1.0000000
NR. 67. STELLUNG 2 1.0000000
NR. 68. STELLUNG 3 1.0000000
NR. 97. VOLUMEN= 1.0000000 CCM

BEI FOLGENDEN ABWEICHUNGEN WIRD SONDERMESSUNG GESTARTET.

NR. 83. REAKTORLEISTUNG: 20.000000 %
NR. 84. PRAEZIPITATOR-ZAEHLRATE: 50.000000 %

NR. 99. BAND NR. = 648G01

SIND DIE GELISTETEN DATEN RICHTIG ? NEIN

WELCHE DATEN SOLLEN GEAENDERT WERDEN ?
AUF DIE FRAGE NR., OBIGE NR. EINTIPPEN, DANN RETURN.
AUF DIE FRAGE WERT, NEUEN WERT EINGEBEN, DANN RETURN.

WIRD AUF DIE FRAGE NR. EINE 0 EINGEGEBEN, DANN WIRD DIE EINGABE BEENDET.
NR.? 99
WERT ? 648G01
NR.? 0

```
DIE SPEKTRENABLAGE BEGINNT BEI KASSETTENFILE NR.
NORMAL   30.000000
LANGZEIT  8.0000000
ALARM   1.0000000

RICHTIG? NEIN

NORMAL NR.? 30
LANGZEIT NR.? 1
ALARM NR.? 1

DAS NAECHSTE SPEKTRUM ERHAELT DIE NR.  94
DIE NAECHSTE EICHMESSUNG ERHAELT DIE NR.   9
RICHTIG ?  NEIN

MIT WELCHER SPEKTRUMS NR. SOLL BEGONNEN WERDEN ?  94

WELCHE NR. SOLL DIE NAECHSTE EICHMESSUNG ERHALTEN ?  8

SOLLEN DIE STEUERDATEN AUF KASSETTE GELEGT WERDEN ? JA

ONGAM STARTET
```

Example : Restart

```
*RESTART

BAND NR. 648G01   RICHTIG ? JA

SOLLEN PROGRAMMSTEUERDATEN GEAENDERT WERDEN ? NEIN

A C H T U N G !

VOR START CHECK #1 DURCHFUEHREN, DANN RETURNTASTE

RESTART LAEUFT   1001017.0
```