KONINKLIJK NEDERLANDS METEOROLOGISCH INSTITUUT

WETENSCHAPPELIJK RAPPORT

SCIENTIFIC REPORT

W.R. 80 - 3

A.G.M. Driedonks, P.A.T. Nieuwendijk
and C.J. Goes

A set of computer programs to process turbulence data measured at the 200 m mast at Cabauw

De Bilt 1980

Publikatienummer: K.N.M.I. W.R. 80 - 3 (M.O.)

Koninklijk Nederlands Meteorologisch Instituut,
Meteorologisch Onderzoek,
Postbus 201,
3730 AE De Bilt,
Nederland.

U.D.C.: 551.507.7 ;
551.524.4 :
551.554 :
551.551.2

Abstract

A set of computer programs is described to process turbulence data as measured along the 200 m high meteorological mast at Cabauw, the Netherlands (Van Ulden et al., 1976; Driedonks et al., 1978).

The following parameters are measured simultaneously at six levels: three wind components, dry bulb temperature T_d, wet bulb temperature T_w.

Measuring heights are 20, 40, 80, 120, 160, 200 m. Standard sampling rate is 5 Hz.

For wind measurements, trivanes are used (Wieringa, 1967 and 1972), which measure wind speed, azimuth and elevation angle.

Temperature fluctuations are measured with two pair of unventilated thermocouples (dry and wet bulb). (Kohsiek and Monna, 1980).

Registration of the turbulence data is performed by a Hewlett and Packard 21 MX minicomputer, which formats the data on magnetic tape.

For the processing of these data on the Burroughs B6700 computer at the KNMI a series of computer programs has been written. This sequence of programs performs the following main actions:

First, the format is checked and calculations are carries out to find possible technical malfunctioning of the instruments. If suitable, corrections can be supplied.

In the second stage, calibration factors are introduced in order to transform the data into SI-units. Then, several calculations are performed in order to provide averages over any chosen time interval. Also, the time history of all variables can be plotted.

The third action is to calculate the variances and covariances over any chosen time interval between variables at each level. Also, the spectral distribution of the variances and covariances can be calculated.

-o-o-o-

CONTENTS

A. General information

A.1 Introduction

The Royal Netherlands Meteorological Institute (KNMI) carries out a boundary layer research program at the 200 m high meteorological mast at Cabauw. (Van Ulden et al., 1976; Driedonks et al. 1978).

At this mast a continuous measuring program provides data on average values of meteorological parameters with high vertical resolution. Sampling time for these parameters is 120 s and standard averaging time is 30 min. (Driedonks et al., 1978).

On selected day it is possible to measure simultaneously at six heights the turbulent fluctuations of three wind components, dry bulb and wet bulb temperature. From these data standard deviations and turbulent fluxes are derived. Standard sampling rate for these turbulence data is 5 Hz.

For wind measurements trivanes are used (Wieringa, 1967 and 1972), which measure the length, azimuth and elevation angle of the wind vector. The propeller for wind speed measurements has a first-order response length of 0.5 m, the directional part has a damped wave length of 4 m and a damping ratio 0.56 (Monna and Driedonks, 1979).

Temperature fluctuations are measured with two pair of unventilated thermocouples with lag time <0.1 s; the wet bulb temperature sensors have a lag time <0.4 s (Kohsiek and Monna, 1980). The temperature sensors are placed in an array on both sides of the trivane so that they are ≈ 1 m apart. The response of such an array is shown by Wieringa and Lindert (1971) to be equivalent to a first-order sensor with a distance of 0.5 m.

The total array of trivane + thermocouples can measure reliably turbulent fluctuations down to 5 m wave length with less than 15% attenuation (Sabinin, 1967).

A Hewlett & Packard 21 MX minicomputer has been in use since May 1977 for the registration of the turbulence data. In Fig. 1 the configuration of this minicomputer is shown. A more detailed description can be found in Nieuwendijk and Van Vuure (1978).
For the processing of the tapes with the raw data a series of computer programs has been written for the Burroughs B6700 computer at the KNMI. This series consists of several subsets with different functions.

The main functions are:

a. Technical inspection

The format of the registration tapes is checked.
The mean value, standard deviation, maximum and minimum for each variable are calculated and the samples which exceed a given excentricity. From these calculations and the log-book describing the features of the measuring period, periods of technical malfunctioning of some instruments can be identified.

b. Transformation to SI-units

Calibration factors are introduced to transform the data into SI-units. The calibration factors for wind speed are supplied by wind tunnel calibrations. The azimuth and elevation angles are measured with potentiometers, which are all individually calibrated. The thermocouples have a standard calibration, only depending on the materials used.
After transformation to SI-units, the following quantities can be calculated for all variables over any chosen time interval: mean, standard deviation, maximum, minimum, skewness, kurtosis, and linear trend. As an option, the time history of all variables can be plotted.

c. Calculation of covariance

After transformation from polar to Cartesian coördinates, the covariances between all variables as measured at each level are calculated for any chosen time interval. This is done in three coördinate frames: x-axis pointing eastwards, x-axis along local mean horizontal wind and x-axis along mean horizontal wind at lowest height. Also, several derived quantities (e.g. L, u_*) are calculated. The independent triple correlations are also given.

Furthermore, the spectral distribution of all variances and covariances can be calculated and plotted.

A.2 Layout of the Hewlett & Packard magnetic tapes

The HP minicomputer measures the momentaneous values of 64 channels and writes them on a 9-track, 1600 bpi magnetic tape. 63 out of these 64 channels can contain one parameter each, channel no. 64 always contains a sample counter. Later on in the series of programs only 40 out of these 63 measuring channels are retained, the other ones are skipped to keep computer processingtime within acceptable limits. Standard sampling rate is 5 times per second, and there is an option of 10 times per second.

The first block of a tape is used as an identification block. This identification block contains the start time and end time of the measurements on the tape and an identification of the contents of each channel which is in use. For a detailed description of this identification block see program TRIVDUMP.

The HP channels are normally divided into groups, each group containing the channels of trivane measurements at one height. The order of channels within one group is normally: wind speed, horizontal wind direction (azimuth), vertical wind direction elevation), dry bulb temperature, wet bulb temperature. This order is recommended, but not necessary.

Word length of the HP is 16 bits; the first one is a sign bit; the next 11 bits contain the measuring value and the last four bits contain a control number (which has to be 1101_2)

The signals supplied to the HP have to be DC voltages within the range -10.24 V - +10.235 V.
These voltages are converted to numbers between -2048 and +2047. This number is represented in the first 12 bits. For positive numbers the sign bit is 0 and the number itself is in the next 11 bits. If the number is negative, then the sign bit is 1 and (2048 + the number itself) is in the next 11 bits. So 011111111111 ≡ +2047 and 100000000000 = -2048. ("Two's complement").

The HP tapes are unlabeled but identified by the start time and end time of the registration period as given in the identification block.

With a registration frequency of 5 times per second, one magnetic tape can contain a period of 6 hours. A measuring period may consist of several tapes interrupted by about 10 minutes for tape exchange.

More details on the format of the HP tapes can be found in the description of program TRIVDUMP.

A.3 Outline of programs

In order to achieve the results on turbulence from the HP tapes, the data have to be processed by a series of B6700 programs. This set consists of many subprograms, some of which have a control function, others perform calculations, others are sideline options. As the interconnection of all these subprograms can easily be obscured by the description in full detail of each individual subprogram (as given in part B), we will give an outline of the main actions performed in each one.

Subprogram 1 : **Check on format and transformation of a HP** registration tape to a tape with B6700 words.
program name: TRIVDUMP
input : unlabeled HP tape
output: magnetic tape with data transformed to B6700 words. This tape is titled TRIVAAN <dtg> where <dtg> consists of the start time:
(year-1900)*10^7+(day number)*10^4+hour 10^2+min.
e.g. TRIVAAN 772131230.

Subprogram 2 : (optional). Makes corrections in the TRIVAAN-tape.
program name: TRIVCOR
input : TRIVAAN-tape + correction data
output: corrected TRIVAAN-tape

Subprogram 3 : Performs checks on the data so that technical malfunctioning of the instruments can be found.
program name: TRIVTOETS
input : TRIVAAN-tape
output: printed results per channel per 10-minute interval. For 40 channels the following quantities are printed: mean, standard deviation, maximum value and its sample number, minimum value and its sample number, values that exceed a given excentricity.

Subprogram 4 : Transforms the numbers on the TRIVAAN-tape (voltages) to SI-units by using given calibration factors. Also a correction is applied for the tilt of the vertical axis of the trivane.
program name: TRIVOM
input : TRIVAAN tape + calibration factors (for 40 channels)
output: new magnetic tape with title TRIVOM <dtg> where <dtg> is the same as for the original TRIVAAN tape. This new tape contains transformed data for 40 channels (all values multiplied by 100). The size of a sample is reduced from 64 to 52 from which channel 41-49 are free and 50-52 contain the sample number and exact measuring time. The format of the TRIVOM tapes differs from that of the TRIVAAN tapes. (See description of program TRIVOM).

Subprogram 5 : Calculates (for 40 channels) the mean, standard deviation, maximum, minimum, skewness, kurtosis, and linear trend. Calculations are done over successive time intervals of given duration between two given times.
program name: TRIVBER
input : TRIVOM tape + cards
output: printed results

Subprogram 6 : (optional). Plots the time history for given channels from TRIVOM tape, between two given times and with a given reduction factor (block averaging before plotting). Several channels can be plotted at the same time.
program name: TRIVPLOT
input : TRIVOM tape + cards
output: plotfile

Subprogram 7 : (optional). Calculates the correlation between given pairs of channels from the TRIVOM tape. The calculation is done over successive time intervals of given duration between two given times.
program name: TRIVFLUX
input : TRIVOM tape + cards
output: correlations as calculated and mean values for given channel numbers.

Subprogram 8 : Transforms polar coordinates on TRIVOM tape to Cartesian coordinates (x-axis pointing East). Also the specific humidity is calculated (in grams of water vapor per kg of dry air) when both dry and wet bulb temperature are present.
program name: CARTESE
input : TRIVOM tape + cards
output: new tape with converted values and title TRICAR <dtg> with <dtg> the same as in TRIVOM <dtg>. Specific humidities are stored in any chosen channels 41-49.

Subprogram 9 : Calculates all correlations between two and three channels of a trivane, in various coordinate systems. Also several derived quantities are calculated (e.g. L,u_*). Calculations are done over specified time intervals (see Subprogram 5).
program name: KORREL
input : TRICAR tape + cards
output: line printer output for each trivane of calculated correlations and derived quantities.

Subprogram 10: Calculates the covariance functions, auto- and cospectra for given groups of channels from TRICAR tape for a given period.
program name: PREPCOV (including procedure COVSPEK by E. Vermaas)
input : TRICAR tape + cards
output: disk files with calculated covariance functions and spectra. These files are also printed.

Subprogram 11: (optional). Plots the results of spectral calculations on a double logarithmic scale.
program name: SPECTRAPLOT (author: R. Hoogendoorn)
input : file with calculated spectra as produced by PREPCOV + cards
output: plotfile

B. Detailed description of the programs

B.1 Program TRIVDUMP

This program transforms an unlabeled HP registration tape with 16-bit word length into a labeled B6700 tape (48 bits).

The first 12 bits of each HP word contain the measured value, the last 4 bits form a control number, which has to be 1101_2
If this is not the case, the measured value is wrong and replaced by 9999 on the B6700 tape.

Measured values themselves lie between -2048 and +2047 and are represented in 12 bits, from which the first one is a sign bit. If the first bit is 0, then the measured value is positive and placed in the next 11 bits. If the first bit is 1, then the measured value is negative and the complement of its absolute value minus 1 is placed in the next 11 bits.
So 011111111111 ≡ +2047 and 100000000000 ≡ -2048.
The HP numbers arise from applied DC voltages between -10.24 V and +10.235 V. So the original voltage of the signal can be retained by considering the decimal value of the first 12 HP-bits and applying the following rule:
Voltage=0.005 * (if dec. value ⩽ 2047, then dec. value HP else (dec. value HP -4096)).
In this program however the original HP-values are maintained. Three HP-values are packed in one B6700 word by concatenation.

The first block of 320 HP words contains identification data and its format deviates from the next blocks with measurements. In this first block no control numbers or sign bits occur, but the full 16 bits are in use. The identification is described in the input TRIVDUMP.

Furthermore this program TRIVDUMP has an option to combine three temperature type channels and reduce them to two. In case of dry and wet bulb temperature measurements three signals are measured: S_r (signal from a reference block),

S_d (signal dry bulb thermocouple) and S_n signal wet bulb thermocouple). These three can be reduced to two temperatures: T_d (dry bulb temperature) and T_n (wet bulb temperature). If this option is chosen for three signals S_r, S_d, S_n, then the following sequence of conversions is made:

1. S_r, S_d, S_n are converted to voltages, as given above V_r, V_d, V_n.
2. The thermocouple signal of the reference block is calculated in Volts: $DUR=0.5*10^6*V_r/6250$, where 6250 is an amplification factor, and the reference temperature itself is calculated: $T_r=15+0.02515*DUR - 808*10^{-9}*DUR^2$ (in ^oC).
3. The temperature differences between dry and wet bulb thermocouple and the reference are calculated: $DTD=25000*0.5*(V_d-V_r)/6382$; $DTN=25000*0.5*(V_n-V_r)/6382$.
4. The dry and wet bulb temperatures themselves are calculated, multiplied by 100 and made of integer type (so units are 0.01 ^oC): $T_d=(T_r+DTD)*100$; $T_n=(T_r+DTN)*100$.
5. T_d and T_n are written in prescribed channel numbers, and the corresponding identification of these channels is changed to 25, denoting that they already contain SI-units.

Input TRIVDUMP:

- cards with on each card in 5I5 format the channels to be recombined:
  - channel number dry bulb temperature (S_d) } before conversion
  - " wet bulb temperature (S_n) } before conversion
  - " reference signal (S_r) } before conversion
  - " dry bulb temperature (T_d) } after conversion
  - " wet bulb temperature (T_n) } after conversion
- card with 999 in I5 format
- unlabeled 1600 bpi tape with turbulence data as registered on the HP minicomputer at Cabauw.

Description of HP tape:

BLOCKSIZE = 320 HP words = 640 B6700 bytes (1 byte = 8 bits).
The first block contains the identification of the measurements.
First block = 320 HP words.

word no.	content
1:6	Start of measurements on this tape: year, day number, hour, minutes, sec., msec.
7:12	End of measurements on this tape: year, day number, hour, minutes, sec., msec.
13:320	Identification of the channels for max. 64 channels; each channel identification occupies 4 words; not occupied HP words have all 16 bits set at 1 ($=65535_{10}$); Each channel identification consists of:

1. channel number (1-64)
2. type of mast boom (00=N, 12=SE, 24=SW, 99=no boom).
3. type of instrument:
   - 0 = no instrument
   - 1 = wind speed cup anemometer
   - 2 = wind speed trivane
   - 3 = wind direction vane or trivane
   - 4 = elevation
   - 5 = dry-bulb thermocouple
   - 6 = absolute temperature from mast
   - 7 = temperature difference between two heights
   - 8 = radiation
   - 9 = wind direction trivane
   - 10 = wet-bulb thermocouple
   - 11 = reference temperature (NTC)
   - 12 = microbarograph
   - 25 = converted temperatures
4. height*10 + mast number
   mast number can be:
   - 1 = main 200 m mast
   - 2 = 20 m SE mast
   - 3 = 20 m NW mast
   - 4 = Energy Balance plot
   - 5 = 10 m S mast

So a maximum of 12 + 64 * 4 = 268 words is filled in this first block. All identification words use the full 16 bits.
The next blocks on the HP tape contain 5 samples of 64 data words each. In each sample channel no. 64 contains a sample number, supplied by the internal HP-clock. This sample number increases every 0.2 s, and is reset every time when the number of minutes of the clock is a multiple of 10. So normally it runs from 1-3000. This sample number is later on in the programs used for correct timing of each sample and for checking for discontinuities.

Output TRIVDUMP:

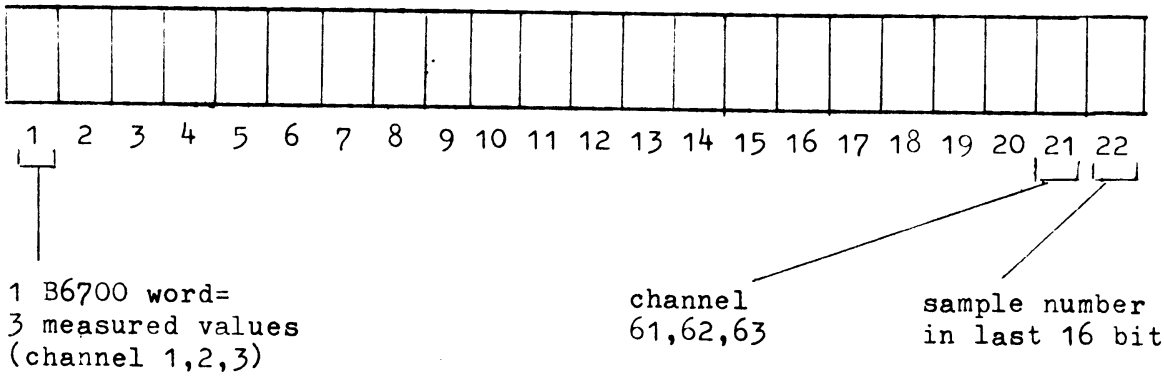
- On the lineprinter the identification plus the first data block is printed. Furthermore every 1000 blocks one block is printed. With each printed block the temperatures as calculated with the given procedure are given separately.
- a 1600 bpi tape with B6700 words.
  TITLE = TRIVAAN <dtg> where dtg is the start time of the tape in the form: (year-1900)*10^7+day*10^5+hr*10^2+min e.g. TRIVAAN772131230. The <dtg> is formed from the first 4 HP-words.

Description of TRIVAAN-tape:

BLOCKSIZE = 1100;MAXRECSIZE = 22
So 1 block = 50 records, 1 record = 22 B6700 words

In each record the first 21 B6700 words are filled with 3 HP words each, the 22nd B6700 word contains only 1 HP word in the last 16 bits. So 1 record = 64 HP words.
First 5 records contain the identification of the tape (first 320 HP-words). Next records are filled with 21*3 measurement values + sample number in word no. 22.
So a data record looks as follows:

data record TRIVAAN-tape



B.2 Program TRIVCOR

With this program the TRIVAAN tapes as produced by TRIVDUMP can be corrected in several ways.
The following corrections are possible:

1. The TRIVAAN tape contains two identification blocks, of which the second one is correct. With TRIVCOR the first identification is skipped and the second identification plus the data are copied. The <dtg> in the title TRIVAAN <dtg> is adapted to the correct identification.
2. A TRIVAAN tape is copied for a given number of samples.
3. The identification can be changed.

The selection which correction must be applied is supplied by cards together with the correction data.

Input TRIVCOR:

- card with in <I9,I2> format:
  - <dtg> of TRIVAAN tape to be corrected.
  - type of correction to be applied (1,2,3 as above)
- if type of correction = 2 then a card with <I6> format:
  - number of samples to be copied.
- if type of correction = 3 then
  a. one card with format <2(I4,I3,4I2)> containing: year, day, hr, min., sec., msec. of start and end time as they must be on the new tape.
  b. for each channel for which the identification has to be corrected a card with format <I3, 2I2,I4>: channel number, boom, instrument, type, height*10+mast (see TRIVDUMP).

Output TRIVCOR:

- all information needed for control is printed, e.g. serial numbers and titles of magnetic tapes, new identification block etc.
- a new corrected tape with title=TRIVAAN <dtg> with dtg as derived from the input cards (in case 3) or from the old tape (cases 1 and 2).

B.3 Program TRIVTOETS

This program does calculations to check the 5 Hz data on a TRIVAAN tape as produced by TRIVDUMP.

It is tested whether the sample number increases correctly from 1-3000. Missing sample numbers are printed. Each time when the sample number passes 3000, i.e. normally after ±10 minutes measuring time, calculations are carried out over the foregoing "block" of ~ 3000 samples. Sample numbers larger than 3000, which sometimes occur, are handled as separate "blocks". When the start time is not a multiple of 10 minutes, the first "block" consists of less than 3000 samples. Each time when calculations are carried out, the time which is printed with the results is increased with 10 minutes, unless the number of samples involved was less than 2000.

For a "block" of ~ 10 minutes the following calculations are done: mean, standard deviation, maximum value + sample no., minimum value + sample no., excentricity of each sample. The excentricity E for a channel containing N values x_i (i=1,...,N) is calculated as:

$$E_i = (x_i - \bar{x})/\sigma_x$$

where $\bar{x} = (1/N) \sum_{i=1}^{N} x_i$

$$\sigma_x = \mathrm{SQRT}\left(\left(1/(N-1)\right) * \left(\sum_{i=1}^{N} x_i^2 - N\,\bar{x}^2\right)\right)$$

The excentricity E_i is compared to a value
$C = 1.42 + 0.3013 * \ln(N-4)$. If $E_i > C$, then x_i,i, $_i$ are printed.

Input TRIVTOETS

- Tape with title TRIVAAN <dtg> as produced by TRIVDUMP.
  The exact title of the tape has to be given by a label equation card for the file TRIV:
  FILE TRIV (TITLE=TRIVAAN772131230).

Output TRIVTOETS

Each time the sample number passes 3000, i.e. each 10 min. period, a table for 40 channels is printed with for each channel:
channel number, number of samples, mean, standard deviation, maximum, sample no. of maximum, minimum, sample no. of minimum, extremely excentric value is printed and for a maximum of 100 the value itself, the excentricity and the sample number.
In the heading of the table the missing sample numbers are given and the starting time of the 10 min. period.

B.4 Program TRIVOM

In this program the values as measured by the HP minicomputer are transformed into SI-units (m/s, degrees, ^{o}C) by means of calibration factors. This transformation is done for 40 channels only. For the channels with elevation signals a correction is applied for the deviation of the vertical axis of the trivane from the real vertical axis. This deviation is measured by means of two electrolytic levels. The results of these calculations are collected on a new tape with only 52 of the original 64 channels left and with title TRIVOM <dtg> where dtg consists of year, day, hour, min. of the starting time, e.g. TRIVOM772131230.

The transformation into SI-units is done with the formula: $X = A*D^2 + B* D + C$, where D is the originally measured value in Volts and A, B and C are calibration factors. (D is calculated as printed out in program TRIVDUMP).
This information is done for 40 channels. The factors A, B, C have to be read from cards with the channel numbers in numerical order. The calibration factors can be changed for a channel during the measuring period (e.g. when an instrument is exchanged). The new calibration factors must then be read from a card after the first 40 cards. This has to be done in order of time of change (if more than 1 channel changes at the same time, then the order is the channel number).

E.g. measuring period from 6.40-9.30;
instrument in channel 10 changed at 8.30;
calibration factors channel 10:

A_1, B_1, C_1 from 6.40-8.30 (card 10)
A_2, B_2, C_2 from 8.30-9.30 (card 41);

remark: periods with different calibration factors for the same channel have to be successive.

The elevation correction is applied at channels with channel identification 4. If E' is the value of the elevation and α the value of the corresponding azimuth angle (supposed to be in the channel previous to E'), then E' is replaced by E:

$E = \arcsin(\cos\theta * \sin E' - \sin\theta * \cos E' * \sin(\alpha + \Phi - \varphi))$.

Here θ, Φ, φ are correction-angles supplied by the electrolytic levels. θ is the angle between the trivane-vertical and the real vertical (positive), φ is the direction of the normal of the plane through these two axes with respect to the direction of a water level (positive for turning to the right) and Φ is the angle of this level with respect to North (positive left from North). θ, φ and Φ are given on the same card as the calibration factors.

The TRIVOM tape contains 52 channels per sample. The first 40 channels contain measurements, channel 41-49 are not occupied, channel 50 and 51 contain the exact measuring time of the sample (channel 50 contains (year-1900) + day e.g. 77213, channel 51 contains hour * 10^5 + min. * 10^3 + sec. * 10^2 + msec. e.g. 1230000), channel 52 contains the sample number. The measuring time in channel 50, 51 is obtained by using the sample sequence number, which runs as an internal clock.
Each B6700 word contains two values, so record length = 26.
The identification is contained in the first four records (only 40 channels are identified).

Input TRIVOM

- tape supplie by TRIVDUMP with title TRIVAAN <dtg>.
- card with 3I8 format containing starting time and frequency
  10^3*(year-1900) + day, hour * 100 + min., registration frequency (Hz)
  e.g. 77213 1230 5
- 40 cards (one for each channel) with calibration factors and elevation corrections.
  format is 9I8 : channel number (1-40)
  A * 10^5
  B * 10^4
  C * 10^2
  both as day*10^4+hr*10^2+min. { start of period for which A, B, C hold
  end " " " " " " " "
  θ * 10^{+1}
  Φ
  φ

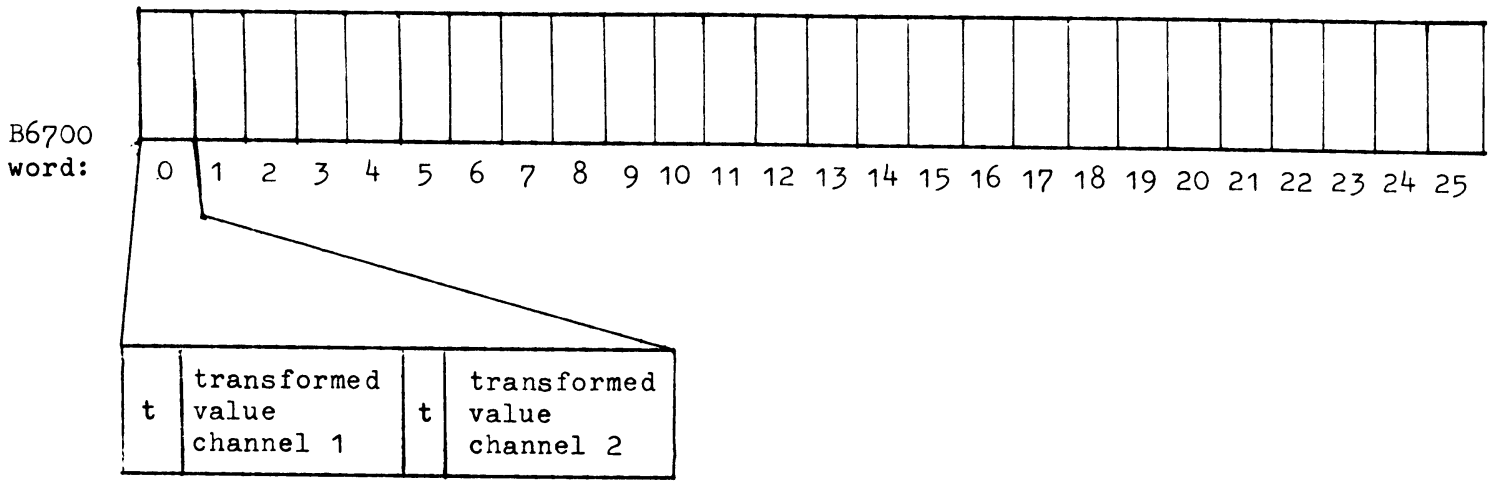
- in case of changing calibration factors, more cards as above in order of time of change (or channel number when time of change is equal).

Output TRIVOM:

line printer : - identification
- the calibration factors
- the first and last transformed sample of each 10 min. period of registration. For these samples the original and transformed values are given together with the elevation data before and after correction. (Each value is multiplied by 100)
- at the end of the program the first 20 transformed samples are printed.

tape : title = TRIVOM <dtg> with dtg is starting time (yydddhhmm e.g. 772131230).
BLOCKSIZE = 2600, MAXRECSIZE = 26.
Each B6700 word contains two data values plus sign bits. The transformed data are multiplied by 100 and of type integer.
Record description (length = 26 words):
Record 0 : 3 contain the identification of the measurements. This identification is the same as given in TRIVDUMP; only 40 channels are identified. Each B6700 word contains 2 identification data. So 6 + 6 + 40 * 4 = 172 identification data occupy 86 B6700 words. No sign bits accompany the identification, so each identification number occupies 24 bits.

From record 4 on, the records contain measuring data, one sample in each record:



t = sign bit (1 = negative),
so transformed values occupy
23 bits each.

B6700 word 0:19 contain 40 measurements.
B6700 word 20:24½ are free (filled with 99999).
In the last 24 bits of B6700 word no. 24 1000*(year-1900)+day is supplied; the first 24 bits of word no. 25 contain $hr*10^5+min*10^3+sec*10+msec$ of this sample, the last 24 bits of word 25 contain the original sample sequence number.

N.B. On the TRIVAAN tapes sometimes sample sequence numbers greater than 600*registration frequency occurred. These samples are skipped from the TRIVOM tape.

B.5 Program TRIVBER

This program calculates several quantities for 40 channels on TRIVBER tapes.
The following calculations are made for each channel: mean, standard deviation, maximum value, minimum value, skewness, kurtosis, linear trend (slope and intersection). If the values in a channel are given by $x_i (i=1,\ldots,N)$ where N denotes the number of samples, then the following formulas have been used.

mean : $\bar{x} = (1/N) \sum_{i=1}^{N} x_i$

standard deviation : $SD = \mathrm{sqrt}\left(\frac{1}{N-1} \left(\sum_{i=1}^{N} x_i^2 - N\bar{x}^2 \right) \right)$.

maximum : $MAX = \max_i(x_i)$

minimum : $MIN = \min_i(x_i)$

skewness : $S = \frac{1}{N*SD^3} \left(\sum_{i=1}^{N} x_i^3 + 2N\bar{x}^3 - 3\bar{x} \sum_{i=1}^{N} x_i^2 \right)$

kurtosis : $K = \frac{1}{N*SD^4} \left(\sum_{i=1}^{N} x_i^4 - 4\bar{x} \sum_{i=1}^{N} x_i^3 - 3N\bar{x}^4 + 6\bar{x}^2 \sum_{i=1}^{N} x_i^2 \right)$

linear trend slope: ($x_i = a*i+b$)

$$a = \frac{1}{N} \left(\sum_{i=1}^{N} i*x_i - \bar{x} \frac{N(N+1)}{2} \right) \Big/ \left(\frac{1}{12} \cdot (N^2-1) \right) .$$

linear trend intersection: $b = \bar{x} - a * \frac{(N+1)}{2}$.

All calculations are done in DOUBLE PRECISION.

In order to remove a possible discontinuity at 360^{o} in the wind direction these channels are pre-processed in the following way:

IF $90 < \text{ABS}\ (x_{i+1}-x_i) < 270$ THEN $x_{i+1} := x_i$;

IF $(x_{i+1}-x_i) < -180$ THEN $x_{i+1} := x_{i+1}+360$;

IF $(x_{i+1}-x_i) > 180$ THEN $x_{i+1} := x_{i+1}-360$;

In this way the sequence 5, 345, 350, 10, 350, 355 is transformed into the "continuous" sequence: 5, -15, -10, 10, -10, -5.

When the calculations have been made the output is again transformed into the range 0-360.

All calculations can be done between two given times and over a given interval.

Input TRIVBER:

- tape with title TRIVOM <dtg >, e.g. TRIVOM772131230.
- card with 2I9 format containing <dtg> identification of TRIVOM tape and time interval (in minutes) over which calculations have to bemade, e.g. 772131230 10
- card with 2I9 format containing the first and last time of the period in which the calculations have to be done, e.g. 1250 1350
  ↓
  hr*100+min

In the example the calculations are done over intervals of 10 min., starting at 1250 and ending at 1350.

Output TRIVBER:

For each calculated interval the following printed output is given:

- begin and end of interval (both in day, hr, min.).
- for each channel (1-40):

channel number
number of samples (N)
mean
standard deviation
maximum + sample number
minimum + sample number
skewness
kurtosis
intersection linear trend (b)
difference between begin and end of
linear trend (N*a).

B.6 Program TRIVPLOT

With this program given channels from the TRIVOM-files (with a maximum of 10 at the same time) can be plotted on the XYNETICS plotter, between two given times.
Before plotting, the sequences can be reduced by averaging a given number of samples (reduction factor).
If a channel contains wind direction data, then before plotting the discontinuity at 360^{o} is removed in the same way as done in program TRIVBER.
The length of the x-axis is determined by the difference between begin and end of the period (length x-axis = (T_e-T_o)/LXAXIS). Minimum length of x-axis is 15 cm.
The length of the y-axis is determined by dividing the maximum difference between the maximal value of a selected channel and its given lower bound by the scale factor LYAXIS.
When a value is met which is lower than the given lower bound, this value is not plotted.
Since each channel has its own lower bound, it is possible to plot several channels in one graph, with shifted ordinates.

Input TRIVPLOT:

- tape with title TRIVOM <dtg>. The exact title of the tape has to be supplied by a LABEL EQUATION CARD,
  e.g. FILE TAPE (TITLE=TRIVOM 772131230).
- card with format <2I9,2F4.2,I3> containing:
  T_o start time of sequence to be potted
  T_e end time of sequence to be plotted
  LYAXIS scaling factor of x-axis in minutes/cm.
  LYAXIS scaling factor of y-axis in units/cm.
  RED reduction factor (averaging before plotting).
- for each channel to be plotted a card must be given with format <I2,F5.2> containing:
  channel number
  lower bound of y-axis

Output TRIVPLOT:

- all input parameters are printed.
- plotfile with title:
  TRPL/$<T_o>$/channel numbers/reduction factor
  e.g. TRPL/772131250/010509/10
  This plotfile can be plotted on the XYNETICS plotter.

B.7 Program TRIVFLUX

This program calculates correlations between pairs of channels on a TRIVOM tape. The calculations are carried out over given time intervals in between a given start and end time. The pairs of channels can be chosen by means of card input. Besides, for any given channel the mean and standard deviation are calculated and printed.
Correlations are calculated as follows:

if $a[i,k]$ $(k=1,\ldots,N)$ contains N values of channel no. i,
and $a[j,k]$ $(k=1,\ldots,N)$ contains N values of channel no. j,
then the mean values are calculated by

$$\overline{a[i]} = \frac{1}{N}\sum_{k=1}^{N} a[i,k] \quad ; \quad \overline{a[j]} = \frac{1}{N}\sum_{k=1}^{N} a[j,k]$$

and the correlations by:

$$R[i,j] = \frac{1}{N}\sum_{k=1}^{N} a[i,k] * a[j,k] - \overline{a[i]} * \overline{a[j]} \ .$$

If channel no. i or j is a wind direction, then before starting the calculations first the discontinuity at 360° is removed by the same procedure as described in program TRIVBER.

Input TRIVFLUX:

- tape with title TRIVOM <dtg>
- card with format <2I9> containing:
  <dtg> of title of TRIVOM tape
  calculation interval in minutes
  e.g. 772131230 10
- card with format <2I9> containing 10^2* hr+min of start and end time of calculations
  e.g. 1230 1400
- cards with format <I4>, each card containing a pair of channels for which correlations have to be calculated.
- card with 999 in <I4>-format.

- cards with format <I4>, each card containing a channel no. for which mean and standard deviation have to be calculated and printed.
- card with 999 in <I4>-format.

Output TRIVFLUX:

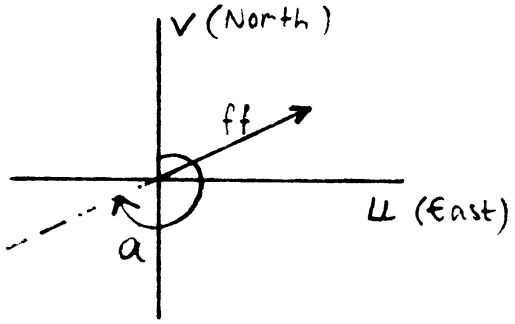
On the lineprinter the following is printed:

- all input data
- the identification of the channels on the input-tape
- for each calculated interval the starting time of that interval is printed, the channel numbers for which mean and standard deviation are calculated, with the number of samples (N), each pair of channels with the calculated correlation.

B.8 Program CARTESE

This program transforms polar coordinates from the TRIVOM tape to a Cartesian coordinate system with x-axis pointing eastward. Also, the specific humidity is calculated (in grams of water vapour per kg of dry air) when both dry and wet bulb temperature is measured.

The trivane data on a TRIVOM tape contain the wind speed (ff), azimuth (a) and elevation angle (e). In each sample these are replaced by Cartesian components u, v, w as follows:



$u = ff \cos e \sin (a-180)$

$v = ff \cos e \cos (a-180)$

$w = ff \sin e$

When also dry and wet bulb temperature (T_d,T_n) are measured in degrees Celsius for a trivane, then the specific humidity q is calculated by using:

$$q = 0.616*10**(7.5\ T_n/(237.3+T_n) + 0.78571) - 0.41*(T_d-T_n)$$

A new file is produced with title TRICAR <dtg> in which ff, a, e are replaced by u, v, w, and q is placed in one of the given channels 41-49.
Tape format is equal to that of TRIVOM (so all data are multiplied by 100).

Input CARTESE

- a tape with title=TRIVOM <dtg>.
- the <dtg> of the TRIVOM tape on a card in <I9>-format, e.g. 772131230
- for each trivane a card with <6I2>-format containing the channel number of:

1. wind speed (ff)
2. azimuth (a)
3. elevation (e)
4. dry bulb temperature (T_d) (0 = missing)
5. wet bulb temperature (T_n) (0 = missing)
6. channel no. in which calculated q has to be placed.

Output CARTESE:

- a tape with transformed data and title = TRICAR <dtg>
  BLOCKSIZE = 2600, MAXRECSIZE = 26.
  Each B6700 word contains two information words, packed by concatenation.
  First 4 records contain identification.
  Next records contain measuring data.
  Format: see TRIVOM.
- the identification of the channels and the input data is printed.
- every 3000 samples, (10 min), the values of all channels before and after transformation are printed.

B.9 Program KORREL

This program calculates from a TRICAR tape all independent correlations between two and three variables within groups of channels which belong to a trivane.
The calculations are performed in three coordinate systems:

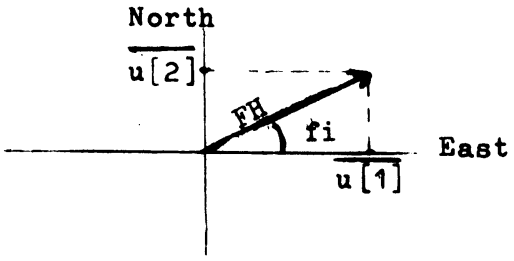
a. x-axis pointing East, y-axis pointing North.
b. x-axis along mean local horizontal wind vector.
c. x-axis along mean horizontal wind vector at lowest measuring level.

The correlations are calculated over given subsequent time intervals between given start and end time.

An outline of the calculation procedure is as follows:
Let n channels form a group which belongs to one trivane.
Several groups are present in the data.
For each trivane the n channel numbers have to be given.
(n is minimally 3 for u, v, w and maximally 6 for u, v, w, T_d, T_n, q).
Let a period of N samples be selected for calculations.
Then for each trivane group the following procedure is followed where u[i,1]=u, u[i,2]=v, u[i,3]= w, u[i,4]=T_d, u[i,5]=T_n, u[i,6]=q for sample i (i=1,...,N).

1. Calculate the mean values $\overline{u[j]} = \frac{1}{N} \sum_{i=1}^{N} u[i,j]$. (j=1,...,n).

2. Calculate the mean horizontal wind vector (FH,fi) in polar coordinates with respect to coordinate system a.



3. Calculate the slope and intersection of a linear trend in the data for j=1,...,n:

$$tr[1,j] = \frac{1}{N}\left(\sum_{i=1}^{N} i*u[i,j] - \overline{u[j]}*\frac{(N+1)}{2}\right) \Big/ \left(\frac{1}{12}*(N^2-1)\right).$$

$$tr[2,j] = \overline{u[j]} - tr[1,j]*\frac{(N+1)}{2}.$$

Then form the differences between end point and begin of the linear trend for the various variables as follows:

d[j] = tr[1,j] * N.
dFH = FHN-FHO where
FHO = SQRT(tr[2,1]^2 + tr[2,2]^2)
FHN = SQRT(UN^2 + VN^2) where
UN = tr[1,1]*N + tr[2,1]
VN = tr[2,1]*N + tr[2,2].
dR = FIO-FIN where
FIO is the direction of the vector tr[2,1], tr[2,2] and
FIN is the direction of the vector UN,VN.
Both directions determined as in 2.

4. Calculate the correlation matrix:

$$FLUX[k,l] = \frac{1}{N}*\sum_{i=1}^{N}\left(u[i,k]-(tr[1,k]*i+tr[2,k])\right)\left(u[i,l]-(tr[1,l]*i+tr[2,l])\right)$$

These are the correlations in coordinate system a, corrected for linear trend.

5. Calculate the correlation matrix in two other coordinate systems:

b. x-axis along local mean horizontal wind vector.

c. x-axis along mean horizontal wind at lowest measuring level.

This calculation is carried out by means of a rotation R[i,j] (i=1,...,n), (j=1,...,n).

for b. fi locally calculated as in 2.

R[1,1]= R[2,2]= cos (fi);

R[1,2]= sin (fi); R[2,1]= -sin (fi);

R[i,i]= 1 (i=3,...,n)

All R[i,j] not specified, are 0.

Then: $FLUXLOK[k,l] = \sum_{m=1}^{n} \sum_{p=1}^{n} R[k,m]*R[l,p]*FLUX[m,p]$.

for c. Do not take the locally calculated fi, but the fi as calculated at the lowest measuring level. Fill R[i,j] as above, using this fi.

Then: $FLUXT[k,l] = \sum_{m=1}^{n} \sum_{p=1}^{n} R[k,m]*R[l,p]*FLUX[m,p]$.

6. Calculate secondary quantities:

$u_* = SQRT(-FLUXLOK[1,3])$;

$T_* = -FLUXLOK[3,4]/u_*$;

$\sigma_i = SQRT(FLUXLOK[i,i])$;

$L = u_*^2 . (273+\overline{u[4]})/(0.35*T_**9.81)$

The absolute value of L is cut off at 5000.

7. Calculate triple-correlation (only in coordinate system a):

$$TRIPEL[k,l,m] = \frac{1}{N} \sum_{i=1}^{N} \big(u[i,k]-(tr[1,k]*i+tr[2,k])\big)* \big(u[i,l]-(tr[1,l]*i+tr[2,l])\big)* \big(u[i,m]-(tr[1,m]*i+tr[2,m])\big).$$

As an option a selection of the results can be archived on disk. When this option is set by means of a WFL card: ? VALUE=1, then a disk file is produced with the following specifications:

TITLE = "KOR/start time/end time/interval

MAXRECSIZE=BLOCKSIZE=209

UNITS=WORDS

A record contains the results of one time interval.
Record description. (Index runs from o...(n-1) where n is the number of trivanes ($n \leqslant 6$)).

element	contents
0	start time of interval
1	end time of interval
2	number of samples
3+index*34+0	height in m
+1	boom direction in degr.
+2	mean wind speed (m/s)
+3	mean wind direction (degr.)
+4	mean dry bulb temperature T_d (^oC)
+5	mean wet bulb temperature T_n (^oC)
+6	mean specific humidity q (gr/kg)
+7	standard deviation u (along local mean wind)
+8	" " v (perpendicular to local mean wind)
+9	" " w (vertical velocity)
+10	" " T_d
+11	" " T_n
+12	" " q
+13	fluxes FLUXT[i,j] (i=0,...5; j=i,...,5)
.	
.	
.	
+33	

Missing quantities are replaced by 99999.

Input KORREL:

- tape with title TRICAR <dtg>.
- card with <dtg> in <I9>-format, e.g. 772131230.
  card with <2I9,I2>-format with:
  start time of calculation period, e.g. 771131240
  end time of calculation period, e.g. 772131400
  calculation intervalin minutes, e.g. 10
  %. The calculations are carried out in the given example for all subsequent time intervals of 10 minutes, between 1240 and 1400 hr. of day no. 213 of year 1977.

- for each trivane a card with <6I2>-format, containing the channel numbers of: u, v, w, T_d, T_n, q, in this order (when a parameter is missing, the corresponding channel number has to be 0).
  The trivane at the lowest measuring level has to be on the first card.

Output KORREL:

The output consists of printed data.

- the title of the input tape is printed, together with the identification of the channels, and the period and interval of calculations.
- the groups of channel numbers for each trivane are given as read from the input cards.
- for each interval the results are printed, each trivane in a column of 44 spaces. Only independent results are printed.

B.10 Program PREPCOV/COVSPEK

This program selects a given period from a TRICAR tape and pre-processes them for the calculation of covariances and spectra. These are calculated and stored in files COVFILE and SPEKFILE by a procedure COVSPEK. These files are also printed. A group of trivane channels is pre-processed in the following way:

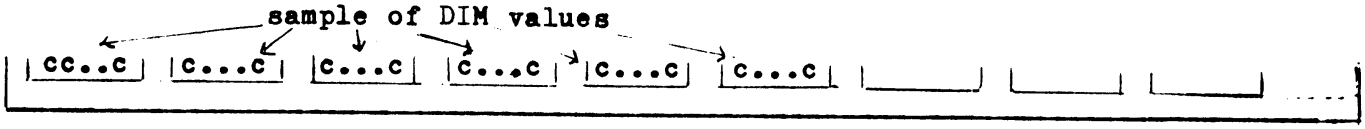
1. The mean horizontal wind vector is calculated.
2. The fluctuating wind components are calculated in a coordinate frame with x-axis along the mean horizontal wind vector.
3. By subtracting the mean value of T_d, T_n, q, the fluctuating parts remain.
4. All fluctuating data components are stored in a file.
5. This file is reduced by block-averaging over a given amount of samples.
6. From this reduced file a given number of channels is stored in a file with title:

TIMESERIE /<dtg of start>/<dtg of end>/<selected channel numbers>/ <reduction factor>

This file TIMESERIE is used as an input for the calculation of covariances and spectra by means of COVSPEK.

Description of COVSPEK:

COVSPEK is a procedure which calculates the covariances (COVFILE) and spectra (SPEKFILE) from a given file TIMESERIE, where TIMESERIE consists of N samples, each sample of DIM data. Record size of TIMESERIE has to be DIM.
Blocksize of TIMESERIE has to be multiple of DIM, and must be given in the label equation.
Format of a block of TIMESERIE :



The covariance functions and spectra are calculated by means of Fast Fourier Transform techniques (E. Vermaas, 1976). For this calculation a set of parameters has to be specified:

DIM	(number of components of each sample in TIMESERIE).
$M = 2^{MM}$	(maximum time lag. In this program a standard value $M = \frac{1}{2} N$ is used).
N	(total number of samples).
WINDOW	(spectral filter. In this program always the Hanning filter is used).
NSTART	(first sample no. of sequence, always set at 0).

In the files COVFILE and SPEKFILE the results are packed in the following way:
the blocksize is M+1 words, there are DIM^2 blocks

COVFILE block i contains:

$$COV^{[i \text{ div } DIM][i \text{ mod } DIM]}(lag) \qquad (lag = 0,1,\ldots,M)$$

SPEKFILE block i contains:

$$\text{Re}\,(S^{pq}(k/2M)) \quad \text{if } p \leqslant q \quad (k=0,\ldots,M)$$

$$\text{Im}\,(S^{qp}(k/2M)) \quad \text{if } p > q \quad (k=0,\ldots,M)$$

where p = i div DIM, q = i mod DIM

So, if e.g. DIM = 3 (0,1,2), then COVFILE contains 9 blocks in the order COV^{00}, COV^{01}, COV^{02}, COV^{10}, COV^{11}, COV^{12}, COV^{20}, COV^{21}, COV^{22}.

and SPEKFILE contains also 9 blocks in the order:

RE S^{00}, Re S^{01}, Re S^{02}, IM S^{10}, Re S^{11}, Re S^{12}, IM S^{20}, IM S^{21}, RE S^{22}.

Input PREPCOV:

- tape with title TRICAR <dtg>
- card with <dtg> of title in <I9>-format
- card with format <2I9,I2>:
  <dtg> of start and end time of selected period
  registration frequency.
- for each trivane which has to be pre-processed a card with in <6I2> format the channel numbers of u, v, w, T_d, T_n, q. (0 = missing channel).
- card with 99 in I2 format.
- card with <I1,*I2,I2> format containing:
  1. DIM (the number of channels to be selected for actual spectral calculations).
  2. The channel numbers to be selected for actual calculations.
  3. The reduction factor over which the data are block-averaged before calculations.

Output PREPCOV:

- title of input tape and identification of channels.
- channel numbers of pre-processed data.
- selected period and selected channel numbers for spectral calculations.
- DIM, reduction factor, frequency, length of reduced series, maximum lag M, mean velocities for each selected trivane.
- block of time lags for which covsriances are calculated (in sec.).
- blocks of COVFILE (M+1 values in each block).
- block with frequencies (Hz) for which spectral estimates are made.
- blocks of SPEKFILE (M+1 values in each block).

B.11 Program SPECTRAPLOT

This program selects one block from the SPEKFILE as calculated with PREPCOV and plots it on a log-log scale, together with identification data.

Input SPECTRAPLOT:

4 cards.

card 1. Format <I6,x1,I9,x1,I3,x1,I2,x1,I2,x1,I2,x1,I5,x1,I6.1> with:
- chosen identification of plotfile (I6)
- <dtg> of start of calculated period (I9)
- <dtg> of end of calculated period (I9)
- height of measurement (I3)
- length of x-axis (cm)
- length of y-axis (cm)
- block number to be selected from SPEKFILE
- block length - 1 of selected block (= M)
- time interval between two samples (sec.).

card 2. Format <I2,A78> with:
- number of characters to be plotted along x-axis (I2)
- text to be plotted along y-axis.

card 3. As card 2, for text along y-axis.

card 4. Format <I2,A78> with:
- number of characters of title of SPEKFILE (I2)
- complete title of SPEKFILE.

Output SPECTRAPLOT:

- plotfile

References

A.G.M. Driedonks, H. van Dop, and W.H. Kohsiek (1978): Meteorological observations on the 213 m mast at Cabauw, in the Netherlands. Fourth Symp. Met. Obs. and Instr., Denver, Am. Met. Soc., 41-46.

W.H. Kohsiek and W.A. Monna (1980): A fast response psychrometer. KNMI Scientific report (in press).

W.A. Monna and A.G.M. Driedonks (1979): Experimental data on the dynamic properties of several propeller vanes. Journ. Appl. Meteor. 18 (5), 699-702.

P.A.T. Nieuwendijk en J.C. van Vuure (1978): Handleiding computersysteem meetmast Cabauw. KNMI, Verslagen V-303 (in Dutch).

K.D. Sabinin (1967): Selection of the relation between periodicity of measurement and instrument inertia in sampling. Atm. Oc. Phys. 3, 973-980.

A.P. van Ulden, J.G. van der Vliet, and J. Wieringa (1976): Temperature and wind observations at heights from 2 to 200 m at Cabauw in 1973. KNMI Scientific Report WR 76-7.

E.H.J. Vermaas (1974): Fast Fourier Transformatie. KNMI Scientific Report WR 74-1 (in Dutch).

J. Wieringa (1967): Evaluation and design of wind vanes. Journ. Appl. Meteor. 6, 1114-1122.

J. Wieringa and F.C.M. van Lindert (1971): Applications of double fin and coupled wind vanes. Journ. Appl. Meteor. 10, 137-145.

J. Wieringa (1972): Tilt errors and precipitation effects in trivane measurements of turbulent fluxes over water. Boundary-Layer Meteor. 2, 406-426.

```
 1000  BEGIN                                                                  1
 2000     COMMENT TRIVDUMP LEEST VAN TAPE AANGEMAAKT DOOR HP
 3000     TE CABAUW EN SCHRIJFT DE OMGEZETTE WAARDEN NAAR TAPE.
 4000     DE KANALEN WAARIN DE OMGEZETTE TEMPERATUREN KOMEN TE STAAN
 5000     KRIJGEN ALS IDENTIFICATIE 25.DIT VOOR DE VOLGENDE PROGRAMMAS.
 6000     1 B6700WOORD ZIJN 3 HP WOORDEN,1 HP WOORD BESTAAT UIT :
 8000     1 TEKENBIT,11 INFOBITS,4 CONTROLEBITS WAAR 1101 IN MOET STAAN.
 9000     KNMI-GOEK-23 MEI 1978,BONNO MOB77208;
10000
11000     REAL JAAR,DAGNR,INDEX,RECNR,I,J,K,TELLER,L,Z,UD,UN,UR,TNTC,
12000     OUD,OUN,URO,OTD,OTN,TD,TN,A0,A1,A2,B0,B1,B2,C1,C2,V,
13000     TEKEN1,TEKEN2,TEKEN3,KANNR,SPAN1,SPAN2,SPAN3,VR,DUR;
14000     ARRAY B6700[0:109],WHP[0:106],WERK[1:64],OM[0:20,0:4],
15000     HULP,BLOK[0:319];
16000     HEX ARRAY HP[0]=WHP;
17000     EBCDIC ARRAY CH[0:255];
18000     BOOLEAN FIRST;
19000
20000     FILE LP(KIND=PRINTER,FILETYPE=3,MAXRECSIZE=132,UNITS=CHARACTERS);
21000     FILE HPTAPE(KIND=PETAPE,BLOCKSIZE=640,MAXRECSIZE=640,
22000     UNITS=CHARACTERS);
23000     FILE TRIVTAPE(KIND=PETAPE,BLOCKSIZE=1100,MAXRECSIZE=22,
24000     TITLE="TRIVAAN000000000.");
25000     FILE KAART(KIND=READER);
26000
27000     LABEL EOF,H,NW,EOK,KLAAR,VUL,HERH,EX;
28000
29000     PROCEDURE PRINT(X); VALUE X; REAL X;
30000     BEGIN                                                              2
31000        INDEX:=-4;
32000        WRITE(LP,</,"GELEZEN 5 SAMPLES">);
33000        FOR RECNR:=1 STEP 1 UNTIL 16 DO
34000        BEGIN                                                          3
35000           WRITE(LP[SPACE 1]);
36000           THRU 20 DO
37000           BEGIN                                                       4
38000              INDEX:=*+4;
39000              IF (INDEX+4) MOD 256=0 THEN
40000              WRITE(LP,<I5>,REAL(HP[INDEX],4)) ELSE
41000              WRITE(LP,<I5>,REAL(HP[INDEX],X));
42000           END;                                                       4 37000
43000        END;                                                          3 34000
44000        IF X=3 THEN
45000        BEGIN                                                          3
46000           WRITE(LP,</,"LAATST GESCHREVEN SAMPLE">);
47000           INDEX:=0;
48000           WRITE(LP,<22I6>,THRU 22 DO WERK[INDEX:=*+1]);
49000           WRITE(LP,<22I6>,THRU 22 DO WERK[INDEX:=*+1]);
50000           WRITE(LP,<20I6>,THRU 20 DO WERK[INDEX:=*+1]);
51000        END;                                                          3 45000
52000     END;                                                             2 30000
53000
54000     A0:=15;
55000     A1:=0.02515;
56000     A2:=-8082-9;
57000     B0:=0;
58000     B1:=402-6;
59000     B2:=0;
60000     C1:=25000;
61000     C2:=0;
62000     V:=6382;
63000     SPAN1:=0.005;
64000     SPAN2:=0.005;
65000     SPAN3:=0.005;
66000     VR:=6250;
67000
68000     WRITE(LP[SKIP 1]);
69000     WRITE(LP,</,"GELEZEN KANAALNRS VOOR TEMPERATUURBEREKENING">);
70000     I:=0;
71000  NW:
72000     J:=-1;
73000     READ(KAART,<5I5>,THRU 5 DO OM[I,J:=*+1])[EOK];
74000     J:=-1;
75000     WRITE(LP,</,5I5>,THRU 5 DO OM[I,J:=*+1]);
76000     I:=I+1;
77000     IF OM[I-1,0] NEQ 999 THEN GO NW;
78000  EOK:
79000
80000     READ(HPTAPE,640,WHP[*])[EOF];
81000     REPLACE CH BY "TRIVAAN",REAL(HP[0],4)-1900 FOR 2 DIGITS,
```

```
 82000     REAL(HP[4],4) FOR 3 DIGITS,REAL(HP[8],4) FOR 2 DIGITS,
 83000     REAL(HP[12],4) FOR 2 DIGITS,".";
 84000     REPLACE TRIVTAPE.TITLE BY CH;
 85000     WRITE(LP,</,40A1>,FOR INDEX:=0 STEP 1 UNTIL 39 DO CH[INDEX]);
 86000     PRINT(4); %PRINT IDENTIFICATIEBLOK
 87000     FOR I:=0 STEP 1 UNTIL 319 DO
 88000     HULP[I]:=REAL(HP[I*4],4);
 89000     J:=-1;
 90000     FOR I:=12 STEP 4 UNTIL 316 DO
 91000     BEGIN                                                          2
 92000        KANNR:=HULP[I];
 93000        IF KANNR<1 OR KANNR>64 THEN ELSE
 94000        BEGIN                                                       3
 95000           FOR K:=0,1,2,3 DO BLOK[KANNR*4+8+K]:=HULP[I+K]
 96000        END;                                                       3 94000
 97000     END;                                                          2 91000
 98000
 99000     FOR I:=0 STEP 1 UNTIL 11 DO BLOK[I]:=HULP[I];
100000     J:=-1;
101000     WHILE TRUE DO
102000     BEGIN                                                          2
103000        IF OM[J:=J+1,0]=999 THEN GO EX;
104000        BLOK[OM[J,3]*4+10]:=25; %OMGEREKENDE TEMPERATUUR
105000        BLOK[OM[J,4]*4+10]:=25;
106000     END;                                                          2 102000
107000 EX:
108000     IF J=0 THEN
109000     WRITE(LP,</,"NIET GEWIJZIGD,MAAR EVT IN GOEDE VOLGORDE GEZET">)
110000     ELSE WRITE(LP,</,"GEWIJZIGD EN EVT IN GOEDE VOLGORDE GEZET">);
111000     WRITE(LP,<" IDENTIFICATIEBLOK: ">);
112000     WRITE(LP,</,12I6>,FOR I:=0 STEP 1 UNTIL 11 DO BLOK[I]);
113000     FOR KANNR:=0 STEP 1 UNTIL 63 DO
114000     WRITE(LP,</,4I7>,FOR K:=0,1,2,3 DO BLOK[KANNR*4+K+12]);
115000
116000     J:=-1;
117000     FOR I:=0 STEP 1 UNTIL 105 DO
118000     B6700[I]:=0 & BLOK[J:=J+1][47:16] &
119000     BLOK[J:=J+1][31:16] & BLOK[J:=J+1][15:16];
120000     B6700[106]:=0 & BLOK[J:=J+1][47:16];
121000     J:=-22;
122000     THRU 5 DO
123000     WRITE(TRIVTAPE,22,B6700[J:=J+22]);
124000
125000     FIRST:=TRUE;
126000 H:
127000     READ(HPTAPE,640,WHP[*])[EOF];
128000     L:=-1;
129000     THRU 5 DO
130000     BEGIN                                                          2
131000        FOR J:=1 STEP 1 UNTIL 63 DO
132000        WERK[J]:=IF REAL(HP[L:=L+4],1)=13 THEN
133000        REAL(HP[L-3],3) ELSE 9999;
134000        WERK[64]:=REAL(HP[L+1],4); %SAMPLENUMMER
135000        L:=L+4;
136000        Z:=0;
137000 HERH:
138000        TD:=TN:=9999;
139000        IF OM[Z,0]=999 THEN GO KLAAR;
140000        IF UD:=WERK[OM[Z,0]]=9999 THEN GO VUL;
141000        IF UN:=WERK[OM[Z,1]]=9999 THEN GO VUL;
142000        IF UR:=WERK[OM[Z,2]]=9999 THEN GO VUL;
143000        UD:=(IF UD>2047 THEN UD-4096 ELSE UD)*SPAN1;
144000        UN:=(IF UN>2047 THEN UN-4096 ELSE UN)*SPAN2;
145000        UR:=(IF UR>2047 THEN UR-4096 ELSE UR)*SPAN3;
146000        DUR:=(UR/VR)*1000000; DUR:=DUR/2;
147000        TNTC:=A0+A1*DUR+A2*DUR**2;
148000        DUD:=(UD-UR)/V; DUD:=DUD/2;
149000        DUN:=(UN-UR)/V; DUN:=DUN/2;
150000        URO:=B0+B1*TNTC+B2*TNTC**2;
151000        DTD:=C1*DUD+2*C2*URO*DUD+C2*DUD**2;
152000        DTN:=C1*DUN+2*C2*URO*DUN+C2*DUN**2;
153000        TD:=(TNTC+DTD)*100;
154000        TN:=(TNTC+DTN)*100;
155000 VUL:
156000        WERK[OM[Z,0]]:=INTEGER(TD);
157000        WERK[OM[Z,1]]:=INTEGER(TN);
158000        IF (TELLER+1) MOD 1000=0 THEN
159000        WRITE(LP,</,"TD,TN:",2I8>,TD,TN);
160000        Z:=Z+1;
161000        GO HERH;
```

```
162000 KLAAR:
163000       J:=0;
164000       FOR I:=0 STEP 1 UNTIL 21 DO
165000       BEGIN                                                          3
166000          TEKEN1:=TEKEN2:=TEKEN3:=0;
167000          IF I=21 THEN B6700[21]:=0&WERK[64][15:16] ELSE
168000          BEGIN                                                       4
169000             IF WERK[J:=J+1]<0 THEN
170000             BEGIN                                                    5
171000                WERK[J]:=-WERK[J]; TEKEN1:=1
172000             END;                                                     5 170000
173000             IF WERK[J:=J+1]<0 THEN
174000             BEGIN                                                    5
175000                WERK[J]:=-WERK[J]; TEKEN2:=1
176000             END;                                                     5 174000
177000             IF WERK[J:=J+1]<0 THEN
178000             BEGIN                                                    5
179000                WERK[J]:=-WERK[J]; TEKEN3:=1
180000             END;                                                     5 178000
181000             B6700[I]:=0&
182000             TEKEN1[47:1] & WERK[J-2][46:15] &
183000             TEKEN2[31:1] & WERK[J-1][30:15] &
184000             TEKEN3[15:1] & WERK[J][14:15];
185000          END;                                                        4 168000
186000       END;                                                          3 165000
187000       WRITE(TRIVTAPE,22,B6700[*]);
188000    END;                                                             2 130000
189000
190000    IF TELLER:=J+1=1000 THEN
191000    BEGIN                                                            2
192000       TELLER:=0; PRINT(3)
193000    END;                                                             2 191000
194000    IF FIRST THEN   %PRINT EERSTE BLOK INFORMATIE
195000    BEGIN                                                            2
196000       FIRST:=FALSE; PRINT(3)
197000    END;                                                             2 195000
198000    GO H;
199000 EOF:
200000    LOCK(TRIVTAPE); PRINT(3); CLOSE(HPTAPE);
201000 END.                                                               1 1000
202000
```

```
BEGIN                                                                    1
 $SET OMIT
   ************************************************************************
   *                                                                      *
   *              O B J E C T / T R I V C O R                             *
   *                                                                      *
   * MET BEHULP VAN DIT PROGRAMMA KAN DE DOOR HET PROGRAMMA              *
   * OBJECT/DMPTRIVEXTRA AANGEMAAKTE MAGNEETBAND WORDEN GECORRIGEERD.    *
   * DE VOLGENDE KORREKTIES KUNNEN D.M.V. DIT PROGRAMMA WORDEN           *
   * IN/UITGEVOERD :                                                     *
   * 1.  OP DE IN TE VOEREN MAGNEETBAND KOMT TWEEMAAL EEN               *
   *     IDENTIFIKATIE BLOK VOOR.HET EERSTE VAN DEZE BLOKKEN WORDT       *
   *     GESKIPT,WAARNA DE OVERIGE BLOKKEN WORDEN GEDUPLICEERD.          *
   *     DE TITLE VAN DE AANGEMAAKTE MAGNEETBAND WORDT AANGEMAAKT        *
   *     AAN DE HAND VAN DE INFORMATIE IN HET TWEEDE IDENTIFIKATIE       *
   *     BLOK.                                                           *
   * 2.  EEN MAGNEETBAND KAN VOOR HET OP TE GEVEN AANTAL SAMPLES        *
   *     WORDEN GEDUPLICEERD.                                            *
   * 3.  EEN IDENTIFIKATIE BLOK KAN WORDEN GEWIJZIGD.                    *
   *                                                                     *
   * DE GEWENSTE SOORT KAN WORDEN GESELEKTEERD DOOR HET BOVENGE-         *
   * NOEMDE NUMMER OP TE GEVEN IN DE EERSTE DATA-KAART.                  *
   *                                                                     *
   * FORMAT INVOERKAARTEN :                                              *
   * 1.  <I9,I2>.          PARAMETERS :DATUMTIJDGROEP,SOORT.             *
   * 2.  INDIEN KORREKTIE 2 :                                            *
   *     <I6>.             PARAMETERS :NRSAMPLES.                        *
   * 3.  INDIEN KORREKTIE 3 :                                            *
   *     <I4,I3,4I2>/,I4,I3,4I2>.                                        *
   *                               PARAMETERS :JAAR,DAGNUMMER,UUR,       *
   *                                           MINUUT,SECONDEN,          *
   *                                           MILLISEC*10.              *
   *          DE PARAMETERS MOETEN ZOWEL VOOR DE BEGIN- ALS EINDTIJD     *
   *          WORDEN OPGEGEVEN.                                          *
   *     B.   VOOR IEDER KANAAL WAARVAN DE IDENTIFIKATIE MOET WORDEN     *
   *          GEWIJZIGD EEN KAART :                                      *
   *          <I3,2I2,I4>.         PARAMETERS :KANAALNUMMER,UITHOUDER,   *
   *                                           INSTRUMENT,PLAATS.        *
   *                                                                     *
   * PROGRAMMEUR :P.NIEUWENDIJK                                          *
   * DATUM       :2 NOVEMBER 1978                                        *
   *                                                                     *
   ************************************************************************
 $POP OMIT
 $PAGE
   % D E C L A R A T I E S

   FILE IN    (KIND=PETAPE,BLOCKSIZE=1100,MAXRECSIZE=22),
   OUT   (KIND=PETAPE,BLOCKSIZE=1100,MAXRECSIZE=22,
   MYUSE=OUT,SAVEFACTOR=999),
   LINE  (KIND=PRINTER,UNITS=CHARACTERS),
   CARD  (KIND=READER,UNITS=CHARACTERS);

   INTEGER    DATUMTIJDGROEP,SOORT,I,J,K,NRSAMPLES;
   EBCDIC ARRAY RULE[0:131];
   ARRAY      TR[0:21],M[0:330],DATUM[0:11],COR[0:3];
   LABEL      EOF;

   % M A I N
   % INVOER DATUMTIJDGROEP EN SOORT

   REPLACE RULE BY " " FOR 132;
   READ(CARD,<I9,I2>,DATUMTIJDGROEP,SOORT);
   REPLACE RULE BY "TRIVAAN",DATUMTIJDGROEP FOR 9 DIGITS,".";
   REPLACE IN.TITLE BY RULE;
   IN.OPEN:=TRUE;
   WRITE(LINE,<"INVOER MAGNEETBAND ",A16," SERIALNO ",A6>,
   RULE,IN.SERIALNO);

   % SKIP EERSTE IDENTIFIKATIE BLOK

   IF SOORT=1 THEN
   THRU 5 DO READ(IN,22,TR);

   % LEES JUISTE IDENTIFIKATIE BLOK

   REPLACE M[*] BY 99 FOR 331 WORDS;
   FOR I:=0 STEP 1 UNTIL 4 DO
   BEGIN                                                                 2
      READ(IN,22,TR);
```

```
81000        FOR J:=0 STEP 1 UNTIL 21 DO
82000        FOR K:=0,1,2 DO M[I+66+J+3+K]:=TR[J].[47-K*16:16];
83000     END;                                                          2 79000
84000
85000     % INDIEN IDENTIFIKATIE BLOK MOET WORDEN GEWIJZIGD
86000
87000     IF SOORT=3 THEN
88000     BEGIN                                                         2
89000        READ(CARD,<I4,I3,4I2,/,I4,I3,4I2>,DATUM);
90000        REPLACE M BY DATUM FOR 12 WORDS;
91000
92000        % PER KANAAL EEN KORREKTIE
93000
94000        WHILE NOT READ(CARD,<I3,2I2,I4>,COR) DO
95000        BEGIN                                                      3
96000           FOR I:=12 STEP 4 UNTIL 316 DO
97000           IF M[I]=COR[0] THEN
98000           BEGIN                                                   4
99000              M[I+1]:=COR[1];
100000             M[I+2]:=COR[2];
101000             M[I+3]:=COR[3];
102000             I:=317;
103000          END;                                                    4 98000
104000       END;                                                       3 95000
105000    END;                                                          2 88000
106000    IF SOORT=2 THEN
107000    READ(CARD,<I6>,NRSAMPLES);
108000
109000    % PRINT IDENTIFIKATIE BLOK
110000
111000    FOR I:=12 STEP 4 UNTIL 316 DO
112000    IF M[I]<1 OR M[I]>64 THEN
113000    M[I]:=M[I+1]:=M[I+2]:=M[I+3]:=99;
114000    WRITE(LINE,<" I D E N T I F I K A T I E ">);
115000    IF SOORT=3 THEN
116000    WRITE(LINE,<" G E C O R R I G E E R D.  ">);
117000    WRITE(LINE,<//,"BEGINDATUM     ",6I6,/,
118000    "EINDDATUM      ",6I6,/,
119000    "KAN#    UITH    INST    PL",/,
120000    64(I4,2I8,I6,/)>,
121000    FOR I:=0 STEP 1 UNTIL 171 DO M[I]);
122000
123000    % OPBOUWEN TITLE SCHRIJFTAPE
124000
125000    REPLACE RULE BY "TRIVAAN",
126000    (M[0]-1900) FOR 2 DIGITS,
127000    M[1] FOR 3 DIGITS,
128000    M[2] FOR 2 DIGITS,
129000    M[3] FOR 2 DIGITS,".";
130000    REPLACE OUT.TITLE BY RULE;
131000    OUT.OPEN:=TRUE;
132000    WRITE(LINE,<"SCHRIJFTAPE ",A16," SERIALNO ",A6>,
133000    RULE,OUT.SERIALNO);
134000
135000    %  WEGSCHRIJVEN IDENTIFIKATIE
136000
137000    FOR I:=0 STEP 1 UNTIL 4 DO
138000    BEGIN                                                         2
139000       FOR J:=0 STEP 1 UNTIL 21 DO
140000       FOR K:=0,1,2 DO
141000       TR[J].[47-K*16:16]:=M[I*66+J*3+K];
142000       WRITE(OUT,22,TR);
143000    END;                                                          2 138000
144000
145000    % COPIEREN VOOR NRSAMPLES RECORDS
146000
147000    IF SOORT NEQ 2 THEN NRSAMPLES:=12*6;
148000    THRU NRSAMPLES DO
149000    BEGIN                                                         2
150000       READ(IN,22,TR)[EOF];
151000       WRITE(OUT,22,TR);
152000    END;                                                          2 149000
153000 EOF:
154000    WRITE(LINE,<"GESCHREVEN RECORDS",I8>,OUT.RECORD+1);
155000    CLOSE(IN);
156000    LOCK(OUT);
157000 END.                                                            1 1000
158000
159000    % COPIEREN VOOR NRSAMPLES RECORDS
160000
```

```
161000      IF SOORT NEQ 2 THEN NRSAMPLES:=10+6;
162000      THRU NRSAMPLES DO
163000      BEGIN                                                    2
164000         READ(IN,22,TR)[EOF];
165000         WRITE(OUT,22,TR);
166000      END;                                                     2 163000
167000   EOF:
168000      WRITE(LINE,<"GESCHREVEN RECORDS",I8>,OUT.RECORD+1);
169000      CLOSE(IN);
170000      LOCK(OUT);
171000   END.                                                        1 1000
172000
```

```
 1000  BEGIN                                                                    1
 2000     COMMENT TRIVTOETS-KNMI-GOEK-4 JULI 1977,BONNO MOB77208.
 3000     WIJZIGINGEN TBV TEMPERATUURBEREKENING UIT 3 THERMOKOPPELSPANNINGEN
 4000     DOORGEVOERD MEI 1978.
 5000     5-HERZ-VERWERKING VAN TRIVAAN-TAPES.
 6000     DE TITLE VAN DE TE LEZEN TAPE WORDT OP EEN KAART MEEGEGEVEN.
 7000     PROGRAMMA LEEST TAPE TRIVAAN000000000 EN CONTROLEERT OP
 8000     UITSCHIETERS EN OP RECORDVOLGORDE.
 9000     BEREKEND WORDEN GEMIDDELDE EN STANDAARDDEVIATIE;
10000
11000     PROCEDURE KOP; FORWARD;
12000     PROCEDURE NULSTAR; FORWARD;
13000     BOOLEAN FIRST;
14000     REAL KANNR,I,J,AANTALK,KANAAL,SJAAR,SDAGNR,SUUR,SMIN,SSEC,
15000     SMSEC,EJAAR,EDAGNR,EUUR,EMIN,ESEC,EMSEC,INSTR,OTEL,TELLER,
16000     GETAL,INDEX,K,EXC,VAR,N,SOM,SOMKW,STDEV,GEM,MAXM,MINM,
17000     LASTNO,BLOKNO;
18000     ARRAY TRC[0:109],TRC[0:319],KANID[0:64,1:3],INTEL[1:64,1:9],
19000     DSKAR[0:63],C[1:64],UIT[0:2,1:40,0:100];
20000     LABEL EOF,EX,EXT,HERH,VERDER;
21000
22000     FILE LP(KIND=PRINTER,MAXRECSIZE=132,FILETYPE=3,UNITS=CHARACTERS);
23000     FILE DSK(KIND=DISK,MAXRECSIZE=64,BLOCKSIZE=6400,
24000     AREASIZE=500,AREAS=12);
25000     FILE TRIV(KIND=PETAPE,BLOCKSIZE=1100,MAXRECSIZE=22);
26000
27000     PROCEDURE VULCONSTANTEN;
28000     BEGIN                                                                2
29000        FOR KANAAL:=1 STEP 1 UNTIL 63 DO
30000        BEGIN                                                            3
31000           IF INTEL[KANAAL,2]>4 AND KANID[KANAAL,2] NEQ 0 THEN
32000           C[KANAAL]:=1.42+0.3013*LN(INTEL[KANAAL,2]-4) ELSE
33000           C[KANAAL]:=0;
34000        END;                                                             3 30000
35000     END;                                                                2 28000
36000
37000     PROCEDURE NULSTUIT;
38000     BEGIN                                                                2
39000        FOR KANAAL:=1 STEP 1 UNTIL 40 DO
40000        FOR J:=1 STEP 1 UNTIL 100 DO
41000        BEGIN                                                            3
42000           UIT[0,KANAAL,0]:=0; %AANTAL UITSCHIETERS
43000           UIT[0,KANAAL,J]:=UIT[1,KANAAL,J]:=UIT[2,KANAAL,J]:=-99999;
44000        END;                                                             3 41000
45000     END;                                                                2 38000
46000
47000     PROCEDURE BEREKEN;
48000     BEGIN                                                                2
49000        LABEL L,LL,LLL;
50000        FILL DSKAR[*] WITH -99999;
51000        WRITE(DSK,64,DSKAR[*]); %SLUITARRAY WEGSCHRIJVEN
52000        WRITE(LP,</," DAG UUR MIN SEC">);
53000        WRITE(LP,</,4I4,>,SDAGNR,SUUR,SMIN,SSEC);
54000        IF INTEL[1,2]>2000 AND INTEL[2,2]>2000 THEN
55000        BEGIN                                                            3
56000           BLOKNO:=BLOKNO+1; SMIN:=SMIN+10; IF SMIN>60 THEN
57000           BEGIN                                                         4
58000              SMIN:=SMIN-60; SUUR:=SUUR+1; IF SUUR>24 THEN
59000              BEGIN                                                      5
60000                 SUUR:=SUUR-24; SDAGNR:=SDAGNR+1
61000              END;                                                       5 59000
62000           END;                                                          4 57000
63000        END;                                                             3 55000
64000        WRITE(LP,</,"BLOKNO : ",I5>,BLOKNO);
65000        KOP;
66000        NULSTUIT;
67000        VULCONSTANTEN;
68000        SEEK(DSK[0]);
69000        FOR KANAAL:=1 STEP 1 UNTIL 63 DO
70000        BEGIN                                                            3
71000           IF KANID[KANAAL,2] NEQ 0 THEN
72000           BEGIN                                                         4
73000              SOM:=INTEL[KANAAL,1];
74000              N:=INTEL[KANAAL,2];
75000              SOMKW:=INTEL[KANAAL,3];
76000              GEM:=IF N NEQ 0 THEN SOM/N ELSE 0;
77000              STDEV:=IF N<2 THEN 0 ELSE
78000              SQRT((SOMKW-SOM*SOM/N)/(N-1));
79000              INTEL[KANAAL,8]:=GEM;
80000              INTEL[KANAAL,9]:=STDEV;
```

```
81000          END;                                                    4 72000
82000       END;                                                       3 70000
83000       WHILE TRUE DO
84000       BEGIN                                                      3
85000          READ(OSK,64,OSKAR[*])[LL];
86000          IF OSKAR[0]=-99999 THEN GO LL;
87000          LASTNO:=OSKAR[63];
88000          FOR KANAAL:=1 STEP 1 UNTIL 40 DO
89000          BEGIN                                                   4
90000             IF KANID[KANAAL,2]=0 THEN GO L;
91000             IF INTEL[KANAAL,2]<5 THEN GO L; %ALS N<5 ,NIETS BEREKENEN
92000             GEM:=INTEL[KANAAL,8];
93000             STDEV:=INTEL[KANAAL,9];
94000             IF STDEV=0 THEN GO L;
95000             IF OSKAR[KANAAL-1]=9999 THEN GO L;
96000             EXC:=(OSKAR[KANAAL-1]-GEM)/STDEV;
97000             IF EXC>C[KANAAL] THEN
98000             BEGIN                                                5
99000                TELLER:=UIT[0,KANAAL,0]:=UIT[0,KANAAL,0]+1;
100000               IF UIT[0,KANAAL,0]<100 THEN
101000               BEGIN                                             6
102000                  UIT[0,KANAAL,TELLER]:=OSKAR[KANAAL-1];
103000                  UIT[1,KANAAL,TELLER]:=OSKAR[63];
104000                  UIT[2,KANAAL,TELLER]:=EXC;
105000               END;                                              6 101000
106000            END;                                                 5 98000
107000  L:
108000         END;                                                    4 89000
109000      END;                                                       3 84000
110000  LL:
111000      FOR KANAAL:=1 STEP 1 UNTIL 40 DO
112000      BEGIN                                                      3
113000         WRITE(LP,</,I3,I7,2R9.2>,KANAAL,INTEL[KANAAL,2],
114000         INTEL[KANAAL,8],INTEL[KANAAL,9]);
115000         FOR J:=4,5,6,7 DO WRITE(LP,<I7>,INTEL[KANAAL,J]);
116000         J:=1;
117000         WHILE UIT[0,KANAAL,J] NEQ -99999 DO
118000         BEGIN                                                   4
119000            WRITE(LP,</,X61,2I7,R9.2>,UIT[0,KANAAL,J],
120000            UIT[1,KANAAL,J],UIT[2,KANAAL,J]);
121000            IF J=1 THEN WRITE(LP,<X3,I9>,UIT[0,KANAAL,0]);
122000            J:=J+1; IF J>100 THEN GO LLL;
123000         END;                                                    4 118000
124000  LLL:
125000      END;                                                       3 112000
126000      WRITE(LP,</,"LAATSTE SAMPLENR IS: ",I7>,LASTNO);
127000      SEEK(OSK[0]);
128000      NULSTAR;
129000      WRITE(LP[SKIP 1]);
130000   END;                                                          2 48000
131000
132000   PROCEDURE PRTRC;
133000   BEGIN                                                         2
134000      J:=-1;
135000      FOR I:=0 STEP 1 UNTIL 19 DO
136000      WRITE(LP,</,16I8>,"THRU 16 DO TRC[J:=J+1]);
137000   END;                                                          2 133000
138000
139000   PROCEDURE NULSTAR;
140000   BEGIN                                                         2
141000      FOR KANAAL:=1 STEP 1 UNTIL 64 DO
142000      BEGIN                                                      3
143000         FOR J:=1 STEP 1 UNTIL 9 DO
144000         INTEL[KANAAL,J]:=0;
145000         INTEL[KANAAL,4]:=-2047;
146000         INTEL[KANAAL,6]:=2047;
147000      END;                                                       3 142000
148000   END;                                                          2 140000
149000
150000   PROCEDURE VULTRC;
151000   BEGIN                                                         2
152000      FOR I:=0 STEP 1 UNTIL 21 DO
153000      BEGIN                                                      3
154000         IF I=21 THEN TRC[63]:=TR[I] ELSE
155000         BEGIN                                                   4
156000            FOR J:=0,1,2 DO
157000            BEGIN                                               5
158000               GETAL:=TR[I].[47-J*16:16];
159000               IF KANID[I*3+J,2]=25 THEN %25 IS TEMP KAN.
160000               BEGIN                                            6
```

```
161000                  IF GETAL NEQ 9999 THEN GETAL:=
162000                  TR[I].[46-J*16:15]*(IF TR[I].[47-J*16:1]=1
163000                  THEN -1 ELSE 1);
164000               END                                                    6 160000
165000               ELSE
166000               BEGIN                                                  6
167000                  IF GETAL NEQ 9999 THEN
168000                  BEGIN                                               7
169000                     IF GETAL>2047 THEN GETAL:=GETAL-4096
170000                  END;                                                7 168000
171000               END;                                                   6 166000
172000               TRC[J+I*3]:=GETAL;
173000            END;                                                      5 157000
174000         END;                                                         4 155000
175000      END;                                                            3 153000
176000   END;                                                               2 151000
177000
178000   PROCEDURE KOP;
179000   BEGIN                                                              2
180000      WRITE(LP,</,"KAN AANTAL      GEM     STDEV  ",
181000      "MAX  SAMPLE   MIN  SAMPLE ",
182000      "UITSCHIETERS SAMPLE EXCENTRICITEIT AANTAL">);
183000   END;                                                               2 179000
184000
185000   J:=-22; THRU 5 DO
186000   READ(TRIV,22,TR[J:=J+22])[EOF]; %LEES EN PRINT IDENTIFICATIEBLOK
187000   FOR I:=0 STEP 1 UNTIL 105 DO
188000   FOR J:=0,1,2 DO TRC[I*3+J]:=TR[I].[47-J*16:16];
189000   TRC[318]:=TR[106].[47:16];
190000   PRTRC;
191000   SJAAR:=TRC[0]; SDAGNR:=TRC[1]; SUUR:=TRC[2];
192000   SMIN:=TRC[3]; SSEC:=TRC[4]; SMSEC:=TRC[5];
193000   EJAAR:=TRC[6]; EDAGNR:=TRC[7]; EUUR:=TRC[8];
194000   EMIN:=TRC[9]; ESEC:=TRC[10]; EMSEC:=TRC[11];
195000   WRITE(LP,</,"GEBRUIKTE KANAALNUMMERS ZIJN: ",/>);
196000   J:=0;
197000   FOR I:=12 STEP 4 UNTIL 316 DO
198000   BEGIN                                                              2
199000      KANNR:=TRC[I];
200000      IF KANNR NEQ J:=J+1 THEN
201000      BEGIN                                                           3
202000         FOR INDEX:=315 STEP -1 UNTIL I DO
203000         TRC[INDEX+4]:=TRC[INDEX];
204000         TRC[I]:=J; %KANAALNUMMER
205000         IF J=22 THEN%FOUTJE IN HP TAPES
206000         BEGIN
207000            TRC[I+2]:=1; TRC[I+3]:=23                                 4
208000         END
209000         ELSE                                                        4 206000
210000         TRC[I+2]:=TRC[I+3]:=99;
211000         TRC[I+1]:=99;
212000      END;
213000      IF J>64 THEN GO TO EX;                                         3 201000
214000      KANNR:=TRC[I];
215000      IF KANNR<0 OR KANNR>64 THEN
216000      BEGIN
217000         WRITE(LP,</,"FOUT KANAALNUMMER: ",I9>,KANNR);               3
218000         WRITE(LP,</,"RESTERENDE KANAALNUMMERS ZIJN: ">);
219000         GO EX;
220000      END;
221000      FOR K:=1,2,3 DO KANID[KANNR,K]:=TRC[I+K];                      3 216000
222000      IF KANID[KANNR,2]<0 OR KANID[KANNR,2]>25 THEN
223000      BEGIN
224000         WRITE(LP,</,"WRONG IDENTIFICATION:",2I9>,                   3
225000         KANNR,KANID[KANNR,2]);
226000         GO EX;
227000      END;
228000      WRITE(LP,<I5>,KANNR);                                          3 223000
229000EX:
230000   END;                                                               2 198000
231000
232000   WRITE(LP[SKIP 1]);
233000   FIRST:=TRUE;
234000   NULSTAR;
235000HERH:
236000   READ(TRIV,22,TR[*])[EOF];
237000   VULTRC;
238000   IF SJAAR=1977 AND SDAGNR=257 AND SUUR=5 AND SMIN>30 THEN GO EOF;
239000   %FOUTE TAPE,SAMPLENR LOOPT DOOR TOT 8000
240000   FOR KANAAL:=1 STEP 1 UNTIL 63 DO
```

```
241000      BEGIN                                                                2
242000         INDEX:=KANAAL-1;
243000         INSTR:=KANID[KANAAL,2];
244000         IF INSTR<1 OR INSTR>25 THEN GO EXT;
245000         %25 IS EEN BEREKEND TEMPERATUURKANAAL,OOK BEWERKEN
246000         IF TRC[INDEX]=9999 THEN GO EXT;
247000         INTEL[KANAAL,1]:=*+TRC[INDEX]; %SOM
248000         INTEL[KANAAL,2]:=*+1; %AANTAL
249000         INTEL[KANAAL,3]:=*+TRC[INDEX]*TRC[INDEX]; %SOM V.O. KWADR.
250000         IF TRC[INDEX]>INTEL[KANAAL,4] THEN
251000         BEGIN                                                             3
252000            INTEL[KANAAL,4]:=TRC[INDEX]; %MAXIMUM
253000            INTEL[KANAAL,5]:=TRC[63]; %SAMPLENUMMER
254000         END;                                                              3 251000
255000         IF TRC[INDEX]<INTEL[KANAAL,6] THEN
256000         BEGIN                                                             3
257000            INTEL[KANAAL,6]:=TRC[INDEX]; %MINIMUM
258000            INTEL[KANAAL,7]:=TRC[63]; %SAMPLENUMMER
259000         END;                                                              3 256000
260000   EXT:
261000      END;                                                                 2 241000
262000
263000      WRITE(DSK,64,TRC[*]);
264000      IF FIRST THEN
265000      BEGIN                                                               2
266000         FIRST:=FALSE; OTEL:=TRC[63];
267000         WRITE(LP,</,"EERSTE SAMPLENR IS: ",I7>,OTEL);
268000      END;                                                                2 265000
269000      IF TRC[63]=OTEL THEN
270000      BEGIN                                                               2
271000         OTEL:=OTEL+1;
272000         IF TRC[63]>2999 THEN
273000         BEGIN                                                            3
274000            BEREKEN; FIRST:=TRUE; GO VERDER
275000         END;                                                             3 273000
276000      END                                                                 2 270000
277000      ELSE
278000      BEGIN                                                               2
279000         IF OTEL>TRC[63] THEN
280000         BEGIN                                                            3
281000            BEREKEN; FIRST:=TRUE; GO VERDER
282000         END;                                                             3 280000
283000         WRITE(LP,</,"VERMISTE SAMPLE(S) VANAF: ",I5>,OTEL);
284000         WRITE(LP,</,"TOT: ",I5>,TRC[63]);
285000         OTEL:=TRC[63]+1;
286000      END;                                                                2 278000
287000   VERDER:
288000      GO HERH;
289000   EOF:
290000      BEREKEN;
291000      CLOSE(TRIV);
292000   END.                                                                   1 1000
293000
```

```
 1000  BEGIN                                                                  1
 2000      COMMENT TRIVOM- KNMI GOEK 12 OKTOBER 1977,BONNR MOB77208.
 3000      WIJZIGINGEN TBV TEMPERATUURBEREKENINGEN UIT 3 THERMOKOPPEL-
 4000      SPANNINGEN DOORGEVOERD MEI 1978.
 5000      DE DIGITALE MEETWAARDEN WORDEN OMGEREKEND D.M.V. IJKFACTOREN.
 6000      TEVENS WORDT ER EEN ELEVATIECORRECTIE UITGEVOERD.
 7000      DE TIJD KOMT IN KANAAL NR 50 EN 51.
 8000      DE TE LEZEN TAPES KOMEN VAN OMPTRIV.
 9000      DE IJKFACTOREN WORDEN INGEVOERD VIA KAARTEN.
10000      ER WORDT VAN DE NIEUWE BEREKENDE WAARDEN WEER EEN TAPE AANGEMAAKT;
11000      INTEGER KX;
12000
13000      REAL K,I,J,GETAL,JRDG, UURMIN,HZ,OPH,BLOKGR,KANAAL,OSAMPLE,SAMPLE,
14000      X,Y,STARTTIJD,LTIJD,TMSEC,TSEC,TMIN,TUUR,TDAGNR,JRDAG,INDEX,
15000      F,TEKEN1,TEKEN2,TFOUT,ST,STST,GT,RT,LT,GRENS;
16000      EBCDIC ARRAY CH[0:255];
17000      LABEL EOK,EOF,H,HH,HHH;
18000      ARRAY TR[0:21],TRC[0:51],KANID[0:3,1:40],M[0:330],
19000      N[0:9],COR[0:8,0:40],TRW[0:25];
20000      PROCEDURE PRTRC; FORWARD;
21000
22000      FILE LP(KIND=PRINTER,MAXRECSIZE=132,UNITS=CHARACTERS,FILETYPE=3);
23000      FILE LPR(KIND=PRINTER,MAXRECSIZE=132,UNITS=CHARACTERS,FILETYPE=3);
24000      FILE TRIV(KIND=PETAPE,BLOCKSIZE=1100,MAXRECSIZE=22);
25000      FILE KAART(KIND=READER,BLOCKSIZE=80,UNITS=CHARACTERS);
26000      FILE TRIVOM(KIND=PETAPE,BLOCKSIZE=2600,MAXRECSIZE=26,
27000      SAVEFACTOR=999);
28000
29000      PROCEDURE PR(X); REAL X;
30000      BEGIN                                                              2
31000         RT:=*+1; IF RT GEQ 12 THEN WRITE(LPR [SPACE 1]);
32000         WRITE(LPR,<I6,"*">,X);
33000      END;                                                               2 30000
34000
35000      PROCEDURE PRG(X); REAL X;
36000      BEGIN                                                              2
37000         LT:=*+1; IF LT GEQ 12 THEN WRITE(LPR [SPACE 1]);
38000         WRITE(LPR,<I6,"GTR">,X);
39000      END;                                                               2 36000
40000
41000      PROCEDURE VULTRC;
42000      BEGIN                                                              2
43000         FILL TRC[*] WITH 52(99999);
44000         FOR I:=0 STEP 1 UNTIL 12 DO
45000         BEGIN                                                           3
46000            FOR J:=0,1,2 DO
47000            BEGIN                                                        4
48000               GETAL:=TR[I].[47-J*16:16];
49000               IF GETAL=9999 THEN GETAL:=99999 ELSE
50000               IF KANID[2,I*3+J+1]=25 THEN%OMGEREKENDE TEMP.
51000               BEGIN                                                    5
52000                  GETAL:=TR[I].[46-J*16:15]*(IF TR[I].[47-J*16:1]=1
53000                  THEN -1 ELSE 1);
54000               END                                                      5 51000
55000               ELSE
56000               BEGIN                                                    5
57000                  IF GETAL>2047 THEN GETAL:=GETAL-4096;
58000               END;                                                     5 56000
59000               TRC[J+I*3]:=GETAL;
60000            END;                                                        4 47000
61000         END;                                                           3 45000
62000         TRC[39]:=TR[13].[47:16]; %KANAAL 40
63000         IF TRC[39]=9999 THEN TRC[39]:=99999 ELSE
64000         IF KANID[2,40] NEQ 25 THEN
65000         BEGIN                                                          3
66000            IF TRC[39]>2047 THEN TRC[39]:=TRC[39]-4096;
67000         END;                                                           3 65000
68000         TRC[51]:=TR[21]; %SAMPLENUMMER
69000         FOR I:=0 STEP 1 UNTIL 39 DO
70000         BEGIN                                                          3
71000            IF KANID[2,I+1] NEQ 25 THEN
72000            BEGIN                                                       4
73000               IF TRC[I] NEQ 99999 THEN TRC[I]:=TRC[I]*0.005
74000            END;                                                        4 72000
75000         END;                                                           3 70000
76000
77000      END;                                                              2 42000
78000
79000      PROCEDURE PRTRC;
80000      BEGIN                                                             2
```

```
81000        I:=-1;
82000        THRU 4 DO WRITE(LP,</,10R11.3>,THRU 10 DO TRC[I:=*+1]);
83000        WRITE(LP,</,9R11.3,3I11>,THRU 12 DO TRC[I:=*+1]);
84000     END;                                                          2 80000
85000
86000     PROCEDURE CORRIN;
87000     BEGIN                                                         2
88000        READ(KAART,<10I8>,N)[EOK];
89000        IF KANAAL NEQ N[0] THEN
90000        BEGIN                                                      3
91000           WRITE(LP,</,"KANAAL=",I5>,KANAAL);
92000           WRITE(LP,</,9I8>,N[*]);
93000           GO EOK;
94000        END;                                                       3 90000
95000        COR[0,KANAAL]:=N[0];
96000        COR[1,KANAAL]:=N[1]*4-5;
97000        COR[2,KANAAL]:=N[2]*2-4;
98000        COR[3,KANAAL]:=N[3]*2-2;
99000        COR[4,KANAAL]:=N[4];
100000       COR[5,KANAAL]:=N[5];
101000       COR[6,KANAAL]:=N[6]*2-1;
102000       COR[7,KANAAL]:=N[7];
103000       COR[8,KANAAL]:=N[8];
104000       I:=-1;
105000       WRITE(LP,</,I6,3R11.5,2I8,R9.1,2I9>,THRU 9 DO COR[I:=*+1,KANAAL]);
106000    END;                                                           2 87000
107000
108000    F:=ARCTAN(1)/45;
109000    READ(KAART,<10I8>,JRDG,UURMIN,HZ)[EOF];
110000    WRITE(LP [SKIP 1]);
111000    WRITE(LP,<"TITLE VAN TAPE   FREKWENTIE">);
112000    WRITE(LP,</,"TRIVAAN",I9,X6,I2>,JRDG*10**4+UURMIN,HZ);
113000    IF HZ=10 THEN OPH:=1 ELSE OPH:=2;
114000    IF HZ=10 THEN BLOKGR:=6000 ELSE BLOKGR:=3000;
115000    REPLACE CH[0] BY "TRIVAAN",JRDG FOR 5 DIGITS,
116000    UURMIN FOR 4 DIGITS,".";
117000    REPLACE TRIV.TITLE BY CH;
118000    REPLACE CH[0] BY "TRIVOM",JRDG FOR 5 DIGITS,
119000    UURMIN FOR 4 DIGITS,".";
120000    REPLACE TRIVOM.TITLE BY CH;
121000    WRITE(LP,</,"KANAAL      A          B          C      ",
122000    "T-START  T-EIND     TETA    PHI-1    PHI-2">);
123000    FOR KANAAL:=1 STEP 1 UNTIL 40 DO CORRIN;
124000    WRITE(LP[SKIP 1]);
125000    WRITE(LP,<"KANAALIDENTIFICATIE">);
126000    FILL M[*] WITH 331(99);
127000    FOR I:=0 STEP 1 UNTIL 4  DO
128000    BEGIN                                                          2
129000       READ(TRIV,22,TR)[EOF]; %IDENTIFICATIEBLOKKEN LEZEN
130000       FOR K:=0 STEP 1 UNTIL 21 DO
131000       FOR J:=0,1,2 DO M[I*66+K*3+J]:=TR[K].[47-J*16:16];
132000    END;                                                           2 128000
133000    J:=0;
134000    FOR INDEX:=12 STEP 4 UNTIL 168 DO
135000    BEGIN                                                          2
136000       IF M[INDEX] NEQ J:=*+1 THEN
137000       BEGIN                                                       3
138000          FOR I:=326 STEP -1 UNTIL INDEX DO M[I+4]:=M[I];
139000          M[INDEX]:=J; %KANAALNUMMER
140000          IF J=22 THEN   %FOUTJE IN HP-TAPE
141000          BEGIN                                                    4
142000             M[INDEX+2]:=1; M[INDEX+3]:=23
143000          END                                                      4 141000
144000          ELSE
145000          M[INDEX+2]:=M[INDEX+3]:=99;
146000          M[INDEX+1]:=99;
147000       END;                                                        3 137000
148000    END;                                                           2 135000
149000    FOR I:=0 STEP 1 UNTIL 3 DO
150000    BEGIN                                                          2
151000       FOR K:=0 STEP 1 UNTIL 25 DO
152000       BEGIN                                                       3
153000          INDEX:=I*52+K*2;
154000          IF INDEX >171 THEN M[INDEX]:=M[INDEX+1]:=99999;
155000          TRW[K]:=0 & M[INDEX][47:24] & M[INDEX+1][23:24];
156000       END;                                                        3 152000
157000       WRITE(TRIVOM,26,TRW); %4 IDENTIFICATIEBLOKKEN WEGSCHRIJVEN
158000    END;                                                           2 150000
159000    FOR KANAAL:=1 STEP 1 UNTIL 40 DO
160000    BEGIN                                                          2
```

```
161000        FOR I:=0,1,2,3 DO
162000        KANID[I,KANAAL]:=M[(KANAAL-1)*4+I+12];
163000        J:=-1;
164000        WRITE(LP,</,I2,3I8>,"THRU 4 DO KANID[J:=J+1,KANAAL]);
165000     END;                                                          2 160000
166000
167000  H:
168000     READ(TRIV,22,TR)[EOF];
169000     VULTRC;
170000     IF JRDG=77247 OR (JRDG=77215 AND UURMIN NEQ 940) THEN
171000     BEGIN                                                         2
172000        IF TFOUT:=TFOUT+1 LEQ 100 THEN
173000        BEGIN                                                      3
174000           IF TFOUT=1 OR TFOUT=100 THEN
175000           WRITE(LPR,</,"DUBBEL ID BLOK SKIPPEN",3I6>,
176000           JRDG,UURMIN,TFOUT);
177000           GO H;
178000        END;                                                       3 173000
179000     END;                                                          2 171000
180000     %96 BLOKKEN EN DUBBEL IDENTIFICATIEBLOK SKIPPEN,FOUTE TAPE
181000     SAMPLE:=TRC[51]; OSAMPLE:=SAMPLE-1;
182000     IF SAMPLE>(BLOKGR-100) THEN
183000     BEGIN                                                         2
184000        PR(SAMPLE); GO H
185000     END;                                                          2 183000
186000     IF JRDG=77213 AND TRC[0]=99999 THEN  %BLOK NULLEN SKIPPEN
187000     BEGIN                                                         2
188000        WRITE(LPR,</,"BLOK NULLEN SKIPPEN",I6>,JRDG);
189000        GO H;
190000     END;                                                          2 187000
191000     TMIN:=UURMIN MOD 100;
192000     TUUR:=UURMIN DIV 100;
193000     TDAGNR:=JRDG MOD 1000;
194000
195000  HH:
196000     IF JRDG=77257 AND TUUR=5 AND TMIN GEQ 40 THEN
197000     BEGIN                                                         2
198000        WRITE(LPR,<"AFGEBROKEN 77257">); GO EOF
199000     END;                                                          2 197000
200000     %FOUTE TAPE,SAMPLENR LOOPT DOOR TOT 8000
201000     IF TRC[51]>BLOKGR THEN GO HHH;
202000     %SAMPLENRS>3000 OF >6000 SKIPPEN
203000     IF SAMPLE:=TRC[51]<OSAMPLE THEN GRENS:=BLOKGR-OSAMPLE+SAMPLE
204000     ELSE GRENS:=SAMPLE-OSAMPLE;
205000     FOR I:=1 STEP 1 UNTIL GRENS DO
206000     BEGIN                                                         2
207000        TMSEC:=TMSEC+OPH;
208000        IF TMSEC GEQ 10 THEN
209000        BEGIN                                                      3
210000           TMSEC:=0; TSEC:=T+1
211000        END;                                                       3 209000
212000        IF TSEC GEQ 60 THEN
213000        BEGIN                                                      3
214000           TSEC:=0; TMIN:=T+1
215000        END;                                                       3 213000
216000        IF TMIN GEQ 60 THEN
217000        BEGIN                                                      3
218000           TMIN:=0; TUUR:=T+1
219000        END;                                                       3 217000
220000        IF TUUR GEQ 24 THEN
221000        BEGIN                                                      3
222000           TUUR:=0; TDAGNR:=T+1
223000        END;                                                       3 221000
224000     END;                                                          2 206000
225000     JRDAG:=(JRDG DIV 1000)*1000 +TDAGNR;
226000     LTIJD:=TUUR*10**5+TMIN*1000+TSEC*10+TMSEC;
227000     TRC[49]:=JRDAG;
228000     TRC[50]:=LTIJD;
229000     OSAMPLE:=SAMPLE;
230000     IF SAMPLE=1 OR SAMPLE=BLOKGR THEN PRTRC;
231000
232000     FOR KANAAL:=1 STEP 1 UNTIL 40 DO
233000     BEGIN                                                         2
234000        IF TDAGNR*10000+TUUR*100+TMIN>COR[5,KANAAL] THEN CORRIN;
235000        IF X:=TRC[KANAAL-1] NEQ 99999 THEN
236000        X:=(COR[1,KANAAL]*TRC[KANAAL-1]**2+
237000        COR[2,KANAAL]*TRC[KANAAL-1]+COR[3,KANAAL]);
238000        IF KANID[2,KANAAL]=4 THEN
239000        BEGIN                                                      3
240000           IF X NEQ 99999 THEN
```

```
241000          Y:=ARCSIN(COS(COR[6,KANAAL]*F)*SIN(X*F)-SIN(COR[6,KANAAL]*F)*
242000        COS(X*F)*SIN(((TRC[KANAAL-2])/100+COR[7,KANAAL]-COR[8,KANAAL])*F));
243000          IF SAMPLE=1 OR SAMPLE=BLOKGR THEN
244000          WRITE(LP,</,2R12.3>,X*100,Y*100/F);
245000          X:=Y/F;
246000        END;                                                          3 239000
247000        XX:=IF KANID[2,KANAAL] NEQ 25 THEN
248000        X*100 ELSE TRC[KANAAL-1];
249000        IF X=99999 THEN TRC[KANAAL-1]:=X ELSE TRC[KANAAL-1]:=XX;
250000      END;                                                            2 233000
251000      J:=-1;
252000      FOR I:=0 STEP 2 UNTIL 50 DO
253000      BEGIN                                                           2
254000        TEKEN1:=TEKEN2:=1;
255000        IF TRC[I]>0 THEN TEKEN1:=0;
256000        IF TRC[I+1]>0 THEN TEKEN2:=0;
257000        TRW[J:=J+1]:=0 & TEKEN1[47:1]& ABS(TRC[I])[46:23] &
258000        TEKEN2[23:1]& ABS(TRC[I+1])[22:23];
259000      END;                                                            2 253000
260000      WRITE(TRIVOM,26,TRW);
261000      IF SAMPLE=1 OR SAMPLE=BLOKGR THEN PRTRC;
262000  HHH:
263000      READ(TRIV,22,TR)[EOF];
264000      VULTRC;
265000      GO HH;
266000  EOK:
267000  EOF:
268000      CLOSE(TRIV); CLOSE(TRIVOM); CLOSE(KAART);
269000      THRU 20 DO
270000      BEGIN                                                           2
271000        READ(TRIVOM,26,TRW);
272000        FOR I:=0 STEP 1 UNTIL 25 DO
273000        BEGIN                                                         3
274000          TEKEN1:=TRW[I].[47:1];
275000          TRC[I*2]:=TRW[I].[46:23];
276000          TEKEN2:=TRW[I].[23:1];
277000          TRC[I*2+1]:=TRW[I].[22:23];
278000          IF TEKEN1=1 THEN TRC[I*2]:=-TRC[I*2];
279000          IF TEKEN2=1 THEN TRC[I*2+1]:=-TRC[I*2+1];
280000        END;                                                          3 273000
281000        PRTRC;
282000      END;                                                            2 270000
283000      LOCK(TRIVOM);
284000  END.                                                                1 1000
285000
```

```
 1000  BEGIN                                                                  1
 2000     COMMENT TRIVBER-KNMI-GOEK-8 NOVEMBER 1977.
 3000     DE TITLE VAN DE TE LEZEN TAPE WORDT OP EEN KAART MEEGEGEVEN.
 4000     PROGRAMMA LEEST TAPE TRIVOM000000000 AANGEMAAKT DOOR TRIVOM,
 5000     EN BEREKENT:GEM,STDEV,MAX,SAMPLE,MIN,SAMPLE,SCHEEFHEID,KURTOSIS,
 6000     B EN N*A;
 7000
 8000     PROCEDURE KOP; FORWARD;
 9000     PROCEDURE NULSTAR; FORWARD;
10000
11000     BOOLEAN FIRST;
12000     BOOLEAN ARRAY EERSTE[0:40];
13000     REAL KANNR,I,J,AANTALK,KANAAL,SJAAR,SDAGNR,SUUR,SMIN,
14000     EJAAR,EDAGNR,EUUR,EMIN,INSTR,OTEL,TELLER,
15000     GETAL,K,MAXM,MINM,LASTND,BLOKNO,SAMP,START,EIND,SKREC,
16000     TEKEN1,TEKEN2,DTG,INTERVAL,HULP;
17000     DOUBLE INDEX,N,SOM,SOMKW,STDEV,GEM,SOMSAMP,SCHEEFH,KURT,
18000     B,NMA,A;
19000     ARRAY TR[0:109],TRC[0:51],KANID[0:40,0:3],
20000     VORIGERI[0:40],M[0:207];
21000     DOUBLE ARRAY INTEL[0:40,1:15];
22000     EBCDIC ARRAY CH[0:255];
23000     LABEL EOF,EX,EXT,HERH;
24000
25000     FILE LP(KIND=PRINTER,MAXRECSIZE=132,FILETYPE=3,UNITS=CHARACTERS);
26000     FILE TRIV(KIND=PETAPE,BLOCKSIZE=2600,MAXRECSIZE=26);
27000     FILE KAART(KIND=READER,UNITS=CHARACTERS,MAXRECSIZE=80);
28000
29000     PROCEDURE BEREKEN;
30000     BEGIN                                                              2
31000        LABEL NOBER;
32000        WRITE(LP,</,"  STARTIJD BLOK     EINDTIJD BLOK">);
33000        WRITE(LP,</,"  DAG UUR MIN       DAG UUR MIN    ">);
34000        WRITE(LP,</,3I4,>,SDAGNR,SUUR,SMIN);
35000        IF SMIN:=SMIN+INTERVAL GEQ 60 THEN
36000        BEGIN                                                           3
37000           SMIN:=*-60; SUUR:=SUUR+1
38000        END;                                                           3 36000
39000        IF SUUR GEQ 24 THEN
40000        BEGIN                                                           3
41000           SUUR:=0; SDAGNR:=*+1
42000        END;                                                           3 40000
43000        EMIN:=SMIN;
44000        IF EMIN:=*+INTERVAL GEQ 60 THEN
45000        BEGIN                                                           3
46000           EMIN:=*-60;
47000           EUUR:=SUUR+1;
48000           IF EUUR GEQ 24 THEN
49000           BEGIN                                                        4
50000              EUUR:=0; EDAGNR:=*+1
51000           END;                                                        4 49000
52000        END                                                            3 45000
53000        ELSE
54000        BEGIN                                                           3
55000           EMIN:=SMIN+INTERVAL;
56000           EUUR:=SUUR; EDAGNR:=SDAGNR;
57000        END;                                                           3 54000
58000        OTEL:=EDAGNR*10000000+EUUR*100000+EMIN*1000;
59000        WRITE(LP,<X6,3I4,>,SDAGNR,SUUR,SMIN);
60000        WRITE(LP,</,"BLOKNO : ",I5>,BLOKNO:=*+1);
61000        KOP;
62000        FOR KANAAL:=1 STEP 1 UNTIL 40 DO
63000        BEGIN                                                           3
64000           IF KANID[KANAAL,2] NEQ 0 THEN
65000           BEGIN                                                        4
66000              SOM:=INTEL[KANAAL,1];
67000              N:=INTEL[KANAAL,2];
68000              SOMKW:=INTEL[KANAAL,3];
69000              GEM:=IF N NEQ 0 THEN SOM/N ELSE 0;
70000              INTEL[KANAAL,8]:=GEM;
71000              IF KANID[KANAAL,2]=9 OR KANID[KANAAL,2]=3 THEN
72000              BEGIN                                                    5
73000                 WHILE INTEL[KANAAL,8]>360 DO INTEL[KANAAL,8]:=*-360;
74000                 WHILE INTEL[KANAAL,8]<0 DO INTEL[KANAAL,8]:=*+360;
75000              END;                                                    5 72000
76000              STDEV:=IF N<2 THEN 0 ELSE
77000              SQRT(ABS((SOMKW-SOM*SOM/N)/(N-1)));
78000              INTEL[KANAAL,9]:=STDEV;
79000              IF STDEV=0 THEN GO NOBER; %NIETS MEER BEREKENEN
80000              SCHEEFH:=0; IF STDEV NEQ 0 THEN
```

```
81000              SCHEEFH:=((INTEL[KANAAL,12]+2*N*GEM**3-3*GEM*
82000              SOMKW)/
83000              (N*STDEV**3));
84000              INTEL[KANAAL,11]:=SCHEEFH;
85000              KURT:=0; IF STDEV NEQ 0 THEN
86000              KURT:=((INTEL[KANAAL,13]-4*GEM*INTEL[KANAAL,12]-
87000              3*N*GEM**4+6*GEM**2*SOMKW)/
88000              (N*STDEV**4));
89000              INTEL[KANAAL,12]:=KURT;
90000              SOMSAMP:=INTEL[KANAAL,14];
91000              A:=((INTEL[KANAAL,10]/N)-GEM*SOMSAMP/N)/
92000              ((INTEL[KANAAL,15]-N*(SOMSAMP/N)**2)/N);
93000              B:=GEM-A*SOMSAMP/N;
94000              INTEL[KANAAL,10]:=B;
95000              INTEL[KANAAL,13]:=N*A;
96000  NOBER:
97000           END;                                                    4 65000
98000        END;                                                       3 63000
99000        FOR KANAAL:=1 STEP 1 UNTIL 40 DO
100000       BEGIN                                                      3
101000          WRITE(LP,</,I3,I7,2F9.2>,KANAAL,SINGLE(INTEL[KANAAL,2]),
102000          SINGLE(INTEL[KANAAL,8]),SINGLE(INTEL[KANAAL,9]));
103000          FOR J:=4,6 DO WRITE(LP,<F7.2,I7>,SINGLE(INTEL[KANAAL,J]),
104000          SINGLE(INTEL[KANAAL,J+1]));
105000          FOR J:=11,12,10,13 DO WRITE(LP,<X1,F9.2>,
106000          SINGLE(INTEL[KANAAL,J]));
107000       END;                                                       3 100000
108000       NULSTAR;
109000       WRITE(LP[SKIP 1]);
110000    END;                                                          2 30000
111000
112000    PROCEDURE PRTRC;
113000    BEGIN                                                         2
114000       J:=-1;
115000       FOR I:=0,1,2 DO
116000       WRITE(LP,</,16I8>,THRU 16 DO TRC[J:=J+1]);
117000       WRITE(LP,</,4I8>,THRU 4 DO TRC[J:=J+1]);
118000    END;                                                          2 113000
119000
120000    PROCEDURE NULSTAR;
121000    BEGIN                                                         2
122000       FOR KANAAL:=0 STEP 1 UNTIL 40 DO
123000       BEGIN                                                      3
124000          FOR J:=1 STEP 1 UNTIL 15 DO INTEL[KANAAL,J]:=0;
125000          INTEL[KANAAL,4]:=-99999;
126000          INTEL[KANAAL,6]:=99999;
127000          EERSTE[KANAAL]:=TRUE;
128000          VORIGERI[KANAAL]:=0;
129000       END;                                                       3 123000
130000    END;                                                          2 121000
131000
132000    PROCEDURE VULTRC;
133000    BEGIN                                                         2
134000       FOR I:=0 STEP 1 UNTIL 25 DO
135000       BEGIN                                                      3
136000          TEKEN1:=TR[I].[47:1];
137000          TRC[I*2]:=TR[I].[46:23];
138000          TEKEN2:=TR[I].[23:1];
139000          TRC[I*2+1]:=TR[I].[22:23];
140000          IF TEKEN1=1 THEN TRC[I*2]:=-TRC[I*2];
141000          IF TEKEN2=1 THEN TRC[I*2+1]:=-TRC[I*2+1];
142000       END;                                                       3 135000
143000    END;                                                          2 133000
144000
145000    PROCEDURE KOP;
146000    BEGIN                                                         2
147000       WRITE(LP,</,"KAN AANTAL       GEM     STDEV    ",
148000       "MAX  SAMPLE    MIN  SAMPLE ",
149000       "SCHEEFHEID   KURTOSIS       B        N*A">);
150000    END;                                                          2 146000
151000
152000    READ(KAART,<2I9>,DTG,INTERVAL)[EOF];
153000    READ(KAART,<2I9>,START,EIND)[EOF];
154000    WRITE(LP [SKIP 1]);
155000    WRITE(LP,<"TITLE LEESTAPE   STARTTIJD  EINDTIJD   INTERVAL">);
156000    WRITE(LP,</,"TRIVOM",I9,2I11,I7>,DTG,START,EIND,INTERVAL);
157000    REPLACE CH[0] BY "TRIVOM",DTG FOR 9 DIGITS,".";
158000    REPLACE TRIV.TITLE BY CH;
159000    FOR K:=0,1,2,3 DO
160000    BEGIN                                                         2
```

```
161000        READ(TRIV,26,TR(*))[EOF]; %LEES EN PRINT 4 IDENTIFICATIEBLOKKEN
162000        FOR I:=0 STEP 1 UNTIL 25 DO
163000        BEGIN                                                          3
164000           M[K*52+I*2]:=TR[I].[46:23];
165000           M[K*52+I*2+1]:=TR[I].[22:23];
166000        END;                                                           3 163000
167000     END;                                                              2 160000
168000     SJAAR:=M[0]; SDAGNR:=M[1]; SUUR:=START DIV 100; SMIN:=START MOD 100;
169000     START:=START*1000; EIND:=EIND*1000;
170000     EDAGNR:=SDAGNR;
171000
172000     J:=-1;
173000     WRITE(LP,</>,"STARTTIJD:",6I6>,THRU 6 DO M[J:=J+1]);
174000     WRITE(LP,</>,"EINDTIJD :",6I6>,THRU 6 DO M[J:=J+1]);
175000     WRITE(LP,<///,"KANAALIDENTIFICATIE">);
176000     FOR I:=12 STEP 4 UNTIL 168 DO
177000     BEGIN                                                             2
178000        KANNR:=M[I];
179000        IF KANNR=99999 THEN GO EX;
180000        IF KANNR<0 OR KANNR>40 THEN
181000        BEGIN                                                          3
182000           WRITE(LP,</>,"FOUT KANAALNUMMER: ",I9>,KANNR);
183000           WRITE(LP,</>,"RESTERENDE KANAALNUMMERS ZIJN: ">);
184000           GO EX;
185000        END;                                                           3 181000
186000        FOR J:=0,1,2,3 DO KANID[KANNR,J]:=M[I+J];
187000        IF KANID[KANNR,2]<0 OR KANID[KANNR,2]>25 THEN
188000        BEGIN                                                          3
189000           WRITE(LP,</>,"SOORT INSTRUMENT FOUT: ",I9>,KANID[KANNR,2]);
190000           WRITE(LP,</>,"RESTERENDE KANAALNUMMERS: ">);
191000           GO EX;
192000        END;                                                           3 188000
193000        J:=-1;
194000        WRITE(LP,</>,4I5>,THRU 4 DO KANID[KANNR,J:=J+1]);
195000  EX:
196000     END;                                                              2 177000
197000
198000     J:=0; IF EIND<START THEN J:=1; %TOT IN VOLGENDE DAG
199000     EIND:=EIND+(SDAGNR+J)*10000000;
200000     START:=START+SDAGNR*10000000;
201000     WRITE(LP[SKIP 1]);
202000     OTEL:=START+INTERVAL*1000;
203000     J:=-1; THRU 41 DO EERSTE[J:=J+1]:=TRUE;
204000     NULSTAR;
205000     EJAAR:=0;
206000
207000  HERH:
208000     READ(TRIV,26,TR(*))[EOF];
209000     VULTRC;
210000     IF (TRC[49] MOD 1000)*10000000+TRC[50]<START THEN
211000     BEGIN                                                             2
212000        SKREC:=SKREC+1; GO HERH
213000     END;                                                              2 211000
214000     IF SKREC NEQ 0 THEN
215000     BEGIN                                                             2
216000        WRITE(LP,</>,"GESKIPT",I8," RECORDS">,SKREC); SKREC:=0
217000     END;                                                              2 215000
218000
219000     IF (TRC[49] MOD 1000)*10000000+TRC[50] GEQ OTEL THEN BEREKEN;
220000     FOR KANAAL:=1 STEP 1 UNTIL 40 DO
221000     BEGIN                                                             2
222000        INDEX:=TRC[KANAAL-1];
223000        IF INDEX=99999 THEN GO EXT ELSE INDEX:=INDEX/100;
224000        IF KANID[KANAAL,2]=9 THEN %WINDRICHTING CONTINU MAKEN
225000        BEGIN                                                          3
226000           IF EERSTE[KANAAL-1] THEN
227000           BEGIN                                                       4
228000              VORIGERI[KANAAL-1]:=INDEX; EERSTE[KANAAL-1]:=FALSE
229000           END                                                         4 227000
230000           ELSE
231000           BEGIN                                                       4
232000              IF ABS(INDEX-VORIGERI[KANAAL-1])>90
233000              AND ABS(INDEX-VORIGERI[KANAAL-1])<270 THEN
234000              TRC[KANAAL-1]:=VORIGERI[KANAAL-1]*100;
235000              HULP:=TRC[KANAAL-1]/100;
236000              IF HULP-VORIGERI[KANAAL-1]<-180 THEN INDEX:=HULP+360 ELSE
237000              IF HULP-VORIGERI[KANAAL-1]>180 THEN INDEX:=HULP-360;
238000              VORIGERI[KANAAL-1]:=INDEX;
239000           END;                                                        4 231000
240000        END;                                                           3 225000
```

```
241000
242000        INTEL[KANAAL,1]:=*+INDEX; %SOM
243000        INTEL[KANAAL,2]:=*+1; %AANTAL SAMPLES
244000        INTEL[KANAAL,3]:=*+INDEX**2; %SOM VAN DE KWADRATEN
245000        IF INDEX>INTEL[KANAAL,4] THEN
246000        BEGIN                                                          3
247000           INTEL[KANAAL,4]:=INDEX; %MAXIMUM
248000           INTEL[KANAAL,5]:=TRC[51]; %SAMPLENUMMER
249000        END;                                                           3 246000
250000        IF INDEX<INTEL[KANAAL,6] THEN
251000        BEGIN                                                          3
252000           INTEL[KANAAL,6]:=INDEX; %MINIMUM
253000           INTEL[KANAAL,7]:=TRC[51]; %SAMPLENUMMER
254000        END;                                                           3 251000
255000        INTEL[KANAAL,10]:=*+INTEL[KANAAL,2]*INDEX; %SAMPLENR*WAARDE
256000        INTEL[KANAAL,12]:=*+INDEX**3; %SOM V.D.3E MACHTEN
257000        INTEL[KANAAL,13]:=*+INDEX**4; %SOM V.D.4E MACHTEN
258000        INTEL[KANAAL,14]:=*+INTEL[KANAAL,2]; %SOM VAN DE SAMPLENRS
259000        INTEL[KANAAL,15]:=*+INTEL[KANAAL,2]**2; %SOM VAN DE SAMPLENRS**2
260000  EXT:
261000     END;                                                              2 221000
262000
263000     IF (TRC[49] MOD 1000)*10000000+TRC[50]<EIND THEN GO HERH;
264000  EOF:
265000     BEREKEN;
266000     CLOSE(TRIV);
267000  END.                                                                 1 1000
268000
```

```
1000   $SET LIST
2000   $SET LINEINFO INSTALLATION
3000   $INCLUDE "PLOTPAK/DRIE."
4000  BEGIN
5000   $SET OMIT                                                              1
6000     ************************************************************************
7000     *                                                                      *
8000     *                    T R I V P L O T                                   *
9000     *                                                                      *
10000    *  UIT DE MAGNEETBAND FILES MET TRIVAANMETINGEN KUNNEN EEN AANTAL      *
11000    *  KANALEN (MAXIMAAL 10) WORDEN GESELEKTEERD VAN IEDERE GEWENSTE       *
12000    *  PERIODE.DEZE REEKS KAN NA EVENTUELE REDUKTIE WORDEN GEPLOT.         *
13000    *                                                                      *
14000    *  PROGRAMMEUR :P.A.T. NIEUWENDIJK                                     *
15000    *                                                                      *
16000    *  WIJZIGINGEN :                                                       *
17000    *  -------------                                                       *
18000    *  21 MAART 1979.                                                      *
19000    *  ENKELE KLEINE DETAIL WIJZIGINGEN TER OPTIMALISATIE.                 *
20000    *                                                                      *
21000    ************************************************************************
22000  $POP OMIT
23000  $PAGE
24000    % D E C L A R A T  I E S
25000
26000    FILE TAPE(
27000    KIND=PETAPE
28000    ,UNITS=WORDS
29000    ,BLOCKSIZE=2600
30000    ,MAXRECSIZE=26)
31000    ,CARD (
32000    KIND=READER
33000    ,UNITS=CHARACTERS)
34000    ,PRFI (
35000    KIND=PRINTER
36000    ,UNITS=CHARACTERS);
37000
38000    BOOLEAN ARRAY AANWEZIG[0:51];
39000    BOOLEAN EOF;
40000
41000    ARRAY          ARCHIEF[0:25]
42000    ,VDD[0:51]
43000    ,SPATIE[0:0]
44000    ,IDENT[0:63,0:3]
45000    ,BEGINTYD,EINDTYD[0:5]
46000    ,KANAALNUMMERS[0:9]
47000    ,TEXT[0:22]
48000    ,PLOTAR[0:9,0:4095]
49000    ,XAR[0:4095]
50000    ,NULPUNI[0:9],WAARDE[0:51];
51000    LABEL          EOC,EOT;
52000    ARRAY REFERENCE YAR[0];
53000
54000    INTEGER        I,TE,TO,K,J,AANTKAN,CHECK,CHECKO,REDUKTIE,HZ;
55000
56000    POINTER        PTR;
57000    EBCDIC ARRAY   TITEL[0:99],TXT[0]=TEXT;
58000    REAL           TX,TY,T,LXAS,LYAS,MAXIMUM,MINIMUM,R,Y;
59000
60000
61000    DEFINE ONTBREKEND=99999#,
62000    DIGITSIN(X)=(FIRSTONE(SCALERIGHTF(X,12))-1) DIV 4 +1#,
63000    DIG(X,I)   =Y:=X FOR MIN(DIGITSIN(Y),I) DIGITS#,
64000    F(X,W,D)   =DIG((X DIV POTL[D]),W-1-D),".",
65000    (INTEGER(X) MOD POTL[D]) FOR D DIGITS#,
66000    SPACES(X,W,D)=" " FOR (W-2-D-DIGITSIN(X DIV POTL[D])),
67000    CASE REAL(X<0) OF (" ","-").[7:48] FOR 1#,
68000    FS(X,W,D)  =SPACES(X,W,D),F(X,W,D)#,
69000    IFS(X,W)   =SPACES(X,W,0),DIG(X,W)#,
70000    NOTDEFINED=987654321#;
71000
72000    % P R O C E D U R E S
73000
74000    PROCEDURE PAKUIT;
75000    BEGIN                                                                  2
76000       INTEGER I,J,K;
77000       FOR I:=0 STEP 1 UNTIL 51 DO
78000       IF AANWEZIG[I] THEN
79000       WAARDE[I]:=IF ARCHIEF[K:=I.[47:47]].[J:=(47-I.[0:1]*24):1]=1 THEN
80000       -ARCHIEF[K].[J-1:23] ELSE ARCHIEF[K].[J-1:23];
```

```
81000     END                                                          2 75000
82000     PAKUIT;
83000   $INCLUDE "HEADING."
84000
85000
86000
87000     REAL PROCEDURE RESTORE;
88000     BEGIN                                                        2
89000        OWN INTEGER I;
90000        RESTORE:=WAARDE[I];
91000        IF I:=I+1 > 51 THEN I:=0;
92000     END;                                                         2 88000
93000
94000
95000     % I N I T I A L I S E R I N G
96000
97000     HEADING(PRFI,"TRIVAAN     ");
98000     NAME   (PRFI,"PLOTS       ");
99000
100000    % I N V O E R  P L O T  P A R A M E T E R S
101000
102000    READ(CARD,<2I9,2F4.2,I3>,
103000    TO                    %BEGINDATUM
104000    ,TE                    %EINDDATUM
105000    ,LXAS                  %LENGTE X-AS IN MIN/CM
106000    ,LYAS                  %LENGTE Y-AS IN SCH/CM
107000    ,REDUKTIE);            %# SAMPLES REDUKTIE
108000    I:=0;
109000    WHILE I<10 DO
110000    BEGIN                                                        2
111000       READ(CARD,<I2,F5.2>,KANAALNUMMERS[I] %TE PLOTTEN KAN#
112000       ,NULPUNT[I])[EOC]; %MINIMUM VAN EEN KANAAL
113000       I:=I+1;
114000    END;                                                         2 110000
115000  EOC:
116000    AANTKAN:=I;
117000    I:=J:=-1;
118000
119000    % P R I N T  P A R A M E T E R S
120000
121000    WRITE(PRFI,<"BEGINDATUM              ",I9,/,
122000    "EINDDATUM                ",I9,/,
123000    "LENGTE X-AS              ",F9.2," MIN/CM",/,
124000    "LENGTE Y-AS              ",F9.2," SCH/CM",/,
125000    "REDUKTIE FACTOR          ",I9,/,
126000    "TE PLOTTEN KANALEN       ",*I9,/,
127000    "BIJ BEHORENDE NULPUNTEN ",*F9.2>,
128000    TO,TE,LXAS,LYAS,REDUKTIE,
129000    AANTKAN,THRU AANTKAN DO KANAALNUMMERS[I:=I+1],
130000    AANTKAN,THRU AANTKAN DO NULPUNT[J:=J+1]);
131000
132000    % I N V O E R  I D E N T I F I K A T I E  M A G N E E T B A N D
133000
134000    FOR I:=0 STEP 1 UNTIL 51 DO AANWEZIG[I]:=TRUE;
135000    READ(TAPE,26,ARCHIEF);
136000    PAKUIT;
137000    FOR I:=0 STEP 1 UNTIL 5 DO
138000    BEGINTYD[I]:=RESTORE;
139000    FOR I:=0 STEP 1 UNTIL 5 DO
140000    EINDTYD[I]:=RESTORE;
141000    FOR I:=0 STEP 1 UNTIL 9 DO
142000    BEGIN                                                        2
143000       J:=RESTORE-1;
144000       IDENT[J,0]:=J+1;
145000       IDENT[J,1]:=RESTORE;
146000       IDENT[J,2]:=RESTORE;
147000       IDENT[J,3]:=RESTORE;
148000    END;                                                         2 142000
149000    THRU 3 DO
150000    BEGIN                                                        2
151000       READ(TAPE,26,ARCHIEF);
152000       PAKUIT;
153000       FOR I:=0 STEP 1 UNTIL 12 DO
154000       BEGIN                                                     3
155000          J:=RESTORE-1;
156000          IF J>0 AND J<64 THEN
157000          BEGIN                                                  4
158000             IDENT[J,0]:=J+1;
159000             IDENT[J,1]:=RESTORE;
160000             IDENT[J,2]:=RESTORE;
```

```
161000              IDENT[J,3]:=RESTORE;
162000            END                                                    4 157000
163000          END                                                      3 154000
164000      END;                                                         2 150000
165000
166000      % P R I N T  M A G N E E T B A N D I D E N T I F I K A T I E
167000
168000      WRITE(PRFI,<"I D E N T I F I K A T I E",/>);
169000      FOR I:=0 STEP 1 UNTIL 63 DO
170000      IF IDENT[I,0] NEQ 0 THEN
171000      WRITE(PRFI,<I2" KAN "I2" UITHOUDER "I2" INST. "I2" PL "J5>,
172000      I+1,IDENT[I,0],IDENT[I,1],IDENT[I,2],IDENT[I,3]);
173000
174000      % ZOEK STARTPUNT VERWERKING
175000
176000      DO
177000      BEGIN                                                        2
178000          READ(TAPE,26,ARCHIEF);
179000          WAARDE[49]:=ARCHIEF[24].[22:23];
180000          WAARDE[50]:=ARCHIEF[25].[46:23];
181000          CHECK:=WAARDE[49]*10000+WAARDE[50] DIV 1000;
182000      END                                                          2 177000
183000      UNTIL CHECK>=T0;
184000      CHECK:=T0;
185000
186000      % SELEKTEREN TE PLOTTEN KANALEN
187000
188000      MAXIMUM:=-1000000;
189000      K:=0;
190000      FOR J:=0 STEP 1 UNTIL 51 DO
191000      AANWEZIG[J]:=FALSE;
192000      AANWEZIG[49]:=AANWEZIG[50]:=TRUE;
193000      FOR J:=0,1+J WHILE J<AANTKAN DO
194000      AANWEZIG[KANAALNUMMERS[J]-1]:=TRUE;
195000      PAKUIT;
196000
197000      % CONTINUE MAKEN WINDRICHTING VOOR DE HOOGTE
198000
199000      FOR J:=0,1+J WHILE J<AANTKAN DO
200000      VDD[J]:=WAARDE[KANAALNUMMERS[J]-1]/100;
201000      MINIMUM:=VDD[0];
202000      FOR J:=0,1+J WHILE J<AANTKAN DO
203000      MINIMUM:=MIN(MINIMUM,VDD[J]);
204000      FOR J:=0,J+1 WHILE J<AANTKAN DO
205000      BEGIN                                                        2
206000          IF VDD[J]-MINIMUM>180 THEN VDD[J]:=*-360 ELSE
207000          IF VDD[J]-MINIMUM<=-180 THEN VDD[J]:=*+360;
208000          MINIMUM:=VDD[J];
209000      END;                                                         2 205000
210000
211000      % REDUCEREN VAN DE TE PLOTTEN REEKS
212000
213000      DO
214000      BEGIN                                                        2
215000          FOR I:=1 STEP 1 UNTIL REDUKTIE DO
216000          BEGIN                                                    3
217000              FOR J:=0,1+J WHILE J<AANTKAN DO
218000              IF IDENT[KANAALNUMMERS[J]-1,2]=9 OR
219000              IDENT[KANAALNUMMERS[J]-1,2]=3 THEN
220000              %INDIEN KANAAL IS WINDRICHTING REEKS CONTINU MAKEN
221000              BEGIN                                                4
222000                  R:=WAARDE[KANAALNUMMERS[J]-1]/100;
223000                  IF R-VDD[J]>180 THEN R:=*-360 ELSE
224000                  IF R-VDD[J]<=-180 THEN R:=*+360;
225000                  VDD[J]:=R;
226000                  PLOTAR[J,K]:=*+R;
227000              END                                                  4 221000
228000              ELSE
229000              PLOTAR[J,K]:=*+WAARDE[KANAALNUMMERS[J]-1]/100;
230000              READ(TAPE,26,ARCHIEF)[EOT];
231000              PAKUIT;
232000              CHECK:=WAARDE[49]*10000+WAARDE[50] DIV 1000;
233000          END;                                                     3 216000
234000
235000          % OPVANG END-OF-TAPE
236000
237000          IF FALSE THEN
238000          BEGIN                                                    3
239000  EOT:
240000              TE:=CHECK; CHECK:=*+1;
```

```
241000        END;                                                          3 238000
242000
243000        % REDUCEREN PLOT ARRAY
244000
245000        FOR J:=0,J+1 WHILE J<AANTKAN DO
246000        BEGIN                                                         3
247000           PLOTAR[J,K]:=*/REDUKTIE;
248000           MAXIMUM:=MAX(PLOTAR[J,K],MAXIMUM);
249000           IF PLOTAR[J,K]<NULPUNT[J] THEN
250000           PLOTAR[J,K]:=NOTDEFINED;
251000        END;                                                          3 246000
252000        K:=*+1;
253000     END                                                              2 214000
254000     UNTIL CHECK>TE;
255000
256000     % I N I T I A L I S E R E N P L O T T E N
257000
258000     MINIMUM:=NULPUNT[0];
259000     FOR I:=0,1+I WHILE I<AANTKAN DO
260000     MINIMUM:=MIN(MINIMUM,NULPUNT[I]);
261000     T:=(TE MOD 100 -TO MOD 100)+
262000     ((TE DIV 100) MOD 100-(TO DIV 100) MOD 100)*60+
263000     ((TE DIV 10000) MOD 1000-(TO DIV 10000) MOD 1000)*1440;
264000
265000     TX:=MAX(T/LXAS,15); % LENGTE X-AS IN CM MINIMAAL 15 CM
266000     TY:=((ENTIER(MAXIMUM)+SIGN(MAXIMUM))-
267000     (ENTIER(MINIMUM)+SIGN(MINIMUM)))/LYAS; % LENGTE Y-AS CM
268000
269000     WRITE(PRFI,<
270000     "MAXIMUM(SCALED)        ",F10.2,/,
271000     "MINIMUM                ",F10.2,/,
272000     "LENGTE X-AS            ",F10.2," CM",/,
273000     "LENGTE Y-AS            ",F10.2," CM">,
274000     MAXIMUM,MINIMUM,TX,TY);
275000     REPLACE POINTER(SPATIE[0]) BY " " FOR 6;
276000     REPLACE PTR:TITEL[0] BY
277000     "TRPL/",TO FOR 9 DIGITS,"/";
278000     FOR I:=0,I+1 WHILE I<AANTKAN DO
279000     REPLACE PTR:PTR BY KANAALNUMMERS[I] FOR 2DIGITS;
280000     REPLACE PTR:PTR BY "/",REDUKTIE FOR 2 DIGITS,".";
281000     REPLACE PLOTFILE.TITLE BY TITEL;
282000     FACTOR(0.3937);
283000     DIMTAB(TX+10,TY+10,0);
284000
285000     % PLOTTEN KADER T.B.V. DE ONDERGRENZEN
286000
287000     PLOT(.5,.5,3);
288000     PLOT(4.5,.5,2);
289000     PLOT(4.5,5.5,2);
290000     PLOT(5.5,6.5,2);
291000     PLOT(4.5,5.5,3);
292000     PLOT(.5,5.5,2);
293000     PLOT(.5,.5,2);
294000     REPLACE POINTER(TEXT[0]) BY "         LOWER";
295000     SYMBOL(.6,5.1,.25,TEXT,0,13);
296000     REPLACE POINTER(TEXT[0]) BY "CHANNEL BOUNDS";
297000     SYMBOL(.6,4.7,.25,TEXT,0,14);
298000     REPLACE POINTER(TEXT[0]) BY "         (UNITS)";
299000     SYMBOL(.6,4.3,.25,TEXT,0,15);
300000     FOR I:=0 STEP 1 WHILE I<AANTKAN DO
301000     BEGIN                                                            2
302000        REPLACE TXT BY IFS(KANAALNUMMERS[I],4)," " FOR 3,
303000        FS(NULPUNT[I]*100,6,2);
304000        SYMBOL(.8,3.8-I*0.35,.25,TEXT,0,13);
305000     END;                                                             2 301000
306000
307000     REPLACE POINTER(TEXT[0]) BY "TIME      1 CM=",
308000     F(LXAS*100,5,2)," MINUTES  PERIOD ",
309000     TO FOR 9 DIGITS,"-",TE FOR 9 DIGITS," " FOR 5;
310000     SYMBOL(6,4.5,.3,TEXT,0,57);
311000     AXIS(5.5,6.5,SPATIE,-1,TX,0,0.2,0,LXAS);
312000     REPLACE POINTER(TEXT[0]) BY "1 CM = ",
313000     F(LYAS*100,5,2)," UNITS      ";
314000     SYMBOL(3,7,.3,TEXT,90,20);
315000
316000     %PLOTTEN Y-AS
317000
318000     PLOT(5.5,6.5,3);
319000     FOR R:=0 STEP 1 WHILE R<TY DO
320000     BEGIN                                                            2
```

```
321000        PLOT(5.5,6.5+R,2);
322000        IF R MOD 5=0 THEN PLOT(5.0,6.5+R,2) ELSE PLOT(5.3,6.5+R,2);
323000        PLOT(5.5,6.5+R,3);
324000     END;                                                          2 320000
325000     % PLOTTEN Y-AS RECHTS
326000
327000     PLOT(5.5+TX,6.5,3);
328000     FOR R:=0,1+R WHILE R<TY DO
329000     BEGIN                                                         2
330000        PLOT(5.5+TX,6.5+R,2);
331000        IF R MOD 5 = 0 THEN
332000        PLOT(6.0+TX,6.5+R,2) ELSE PLOT(5.7+TX,6.5+R,2);
333000        PLOT(5.5+TX,6.5+R,3);
334000     END;                                                          2 329000
335000
336000     PLOT(5.5,6.5,-3);
337000     FOR I:=0,I+1 WHILE I<K DO XAR[I]:=I;
338000     XAR[K]:=0;
339000     XAR[K+1]:=LXAS*K/T;
340000     FOR I:=0,I+1 WHILE I<AANTKAN DO
341000     BEGIN                                                         2
342000        YAR:=PLOTAR[I,*];
343000        YAR[K]:=NULPUNT[I];
344000        YAR[K+1]:=LYAS;
345000        LINE(XAR,YAR,K,1,.2,0,0);
346000        NUMBER((XAR[0]-XAR[K])/XAR[K+1],
347000        ((YAR[0]-YAR[K])/YAR[K+1])+.3,
348000        .2,KANAALNUMMERS[I],0,-1);
349000        NUMBER(((XAR[K-1]-XAR[K])/XAR[K+1])+.3,
350000        (YAR[K-1]-YAR[K])/YAR[K+1],
351000        .2,KANAALNUMMERS[I],0,-1);
352000     END;                                                          2 341000
353000     PLOT(0,0,5);
354000     PLOT(0,0,999);
355000  END.                                                             1 4000
356000
357000  PLOT(0,0,999);
358000  END.                                                             0 344000
359000
```

```
 1000  BEGIN                                                                  1
 2000     COMMENT TRIVFLUX-KNMI-GOEK-12 DECEMBER 1977.
 3000     DE TITLE VAN DE TE LEZEN TAPE WORDT OP EEN KAART MEEGEGEVEN.
 4000     TEVENS WORDT OP KAART MEEGEGEVEN STARTTIJD,EINDTIJD,INTERVAL,
 5000     PAREN KANALEN WAARVOOR CORRELATIES BEREKEND MOETEN WORDEN EN
 6000     DE KANAALNUMMERS WAARVOOR GEM. EN STANDAARDEVIATIE BEREKEND MOETEN
 7000     WORDEN. SLUITKAART NA CORRELATIEGEGEVENS EN NA KANAALNRS. =999.
 8000     PROGRAMMA LEEST TAPE TRIVOMOOOOOOOOO AANGEMAAKT DOOR TRIVOM,
 9000     EN BEREKENT FLUXEN UIT F,E,T GEGEVENS;
10000
11000     PROCEDURE KOP; FORWARD;
12000     PROCEDURE HEAD; FORWARD;
13000     PROCEDURE NULSTAR; FORWARD;
14000     BOOLEAN FIRST;
15000     REAL KANNR,I,J,AANTALK,KANAAL,SJAAR,SDAGNR,SUUR,SMIN,SSEC,
16000     SMSEC,EJAAR,EDAGNR,EUUR,EMIN,ESEC,EMSEC,INSTR,OTEL,
17000     GETAL,INDEX,K,VAR,N,SOM,SOMKW,STDEV,GEM,START,EIND,PR1,PR2,
18000     TEKEN1,TEKEN2,OTG,INTERVAL,HULP,BLOKNO,KAN1,KAN2,PI,SKREC;
19000     ARRAY TR[0:109],TRC[0:51],KANID[0:40,0:3],INTEL[1:40,1:3],
20000     KOR[0:100,0:1],M[0:207],TEST[0:40],R[0:100,0:3],
21000     VORIGERI[0:40];
22000     BOOLEAN ARRAY EERSTE[0:40];
23000     EBCDIC ARRAY CH[0:255];
24000     LABEL EOF,EX,EXT,HERH,EOK,LL,H1,H2;
25000
26000     FILE LP(KIND=PRINTER,MAXRECSIZE=132,FILETYPE=3,UNITS=CHARACTERS);
27000     FILE TRIV(KIND=PETAPE,BLOCKSIZE=2600,MAXRECSIZE=26);
28000     FILE KAART(KIND=READER,UNITS=CHARACTERS,MAXRECSIZE=80);
29000
30000     PROCEDURE BEREKEN;
31000     BEGIN                                                              2
32000        HEAD;
33000        SMIN:=(START DIV 1000) MOD 100;
34000        SUUR:=(START DIV 100000) MOD 100;
35000        WRITE(LP,</>," DAG UUR MIN SEC">);
36000        WRITE(LP,</>,4I4,>,SDAGNR,SUUR,SMIN,SSEC);
37000        WRITE(LP,</>,"BLOKNO : ",I5>,BLOKNO:=*+1);
38000        WRITE(LP[SPACE 1]);
39000        START:=OTEL;
40000        IF SMIN:=SMIN+INTERVAL GEQ 60 THEN
41000        BEGIN                                                          3
42000           SMIN:=SMIN-60; SUUR:=SUUR+1
43000        END;                                                           3 41000
44000        IF SUUR GEQ 24 THEN
45000        BEGIN                                                          3
46000           SUUR:=0; SDAGNR:=SDAGNR+1
47000        END;                                                           3 45000
48000        EDAGNR:=SDAGNR; EUUR:=SUUR; EMIN:=SMIN;
49000        IF EMIN:=*+INTERVAL GEQ 60 THEN
50000        BEGIN                                                          3
51000           EMIN:=*-60; EUUR:=*+1;
52000           IF EUUR GEQ 24 THEN
53000           BEGIN                                                       4
54000              EUUR:=0; EDAGNR:=*+1
55000           END;                                                        4 53000
56000        END;                                                           3 50000
57000        OTEL:=EDAGNR*10000000+EUUR*100000+EMIN*1000:
58000        KOP;
59000        I:=-1; J:=0;
60000        WHILE KANAAL:=TEST[I:=*+1] NEQ 999 DO
61000        BEGIN                                                          3
62000           GEM:=STDEV:=0;
63000           IF KANID[KANAAL,2] NEQ 0 THEN
64000           BEGIN                                                       4
65000              SOM:=INTEL[KANAAL,1];
66000              N:=INTEL[KANAAL,2];
67000              SOMKW:=INTEL[KANAAL,3];
68000              GEM:=IF N NEQ 0 THEN SOM/N ELSE 0;
69000              IF KANID[KANAAL,2]=9 THEN
70000              BEGIN                                                    5
71000                 WHILE GEM>360 DO GEM:=GEM-360;
72000                 WHILE GEM<0 DO GEM:=GEM+360;
73000              END;                                                     5 70000
74000              STDEV:=IF N<2 THEN 0 ELSE
75000              SQRT((SOMKW-SOM*SOM/N)/(N-1));
76000           END;                                                        4 64000
77000           J:=J+1; IF J>4 THEN
78000           BEGIN                                                       4
79000              J:=1; WRITE(LP[SPACE 1])
80000           END;                                                        4 78000
```

```
81000          WRITE(LP,<I3,I5,F8.2,X1,F7.2,X5>,KANAAL,N,
82000          SINGLE(GEM),SINGLE(STDEV));
83000       END;                                                    3 61000
84000       WRITE(LP,</,/,"KORRELATIES:">);
85000       WRITE(LP,</,"KANAAL KANAAL KORRELATIE">);
86000       I:=-1;
87000       WHILE KOR[I:=*+1,0] NEQ 999 DO
88000       BEGIN                                                   3
89000          N:=R[I,3];
90000          IF N=0 THEN VAR:=0 ELSE
91000          VAR:=R[I,0]/N-(R[I,1]/N)*(R[I,2]/N);
92000          IF KANID[KOR[I,0],2]=4 OR KANID[KOR[I,1],2]=4 THEN
93000          VAR:=VAR*PI/180;
94000          WRITE(LP,</,2I6,F8.4>,KOR[I,0],KOR[I,1],
95000          SINGLE(VAR));
96000       END;                                                    3 88000
97000       NULSTAR;
98000    END;                                                       2 31000
99000
100000   PROCEDURE PRTRC;
101000   BEGIN                                                      2
102000      J:=-1;
103000      FOR I:=0,1,2 DO
104000      WRITE(LP,</,16I8>,THRU 16 DO TRC[J:=*+1]);
105000      WRITE(LP,</,4I8>,THRU 4 DO TRC[J:=*+1]);
106000   END;                                                       2 101000
107000
108000   PROCEDURE NULSTAR;
109000   BEGIN                                                      2
110000      FOR KANAAL:=1 STEP 1 UNTIL 40 DO
111000      BEGIN                                                   3
112000         FOR J:=1 STEP 1 UNTIL 3 DO
113000         INTEL[KANAAL,J]:=0;
114000         EERSTE[KANAAL]:=TRUE;
115000         VORIGERI[KANAAL]:=0;
116000      END;                                                    3 111000
117000      FOR KANAAL:=0 STEP 1 UNTIL 100 DO
118000      FOR J:=0,1,2,3 DO R[KANAAL,J]:=0;
119000   END;                                                       2 109000
120000
121000   PROCEDURE VULTRC;
122000   BEGIN                                                      2
123000      FOR I:=0 STEP 1 UNTIL 25 DO
124000      BEGIN                                                   3
125000         TEKEN1:=TR[I].[47:1];
126000         TRC[I*2]:=TR[I].[46:23];
127000         TEKEN2:=TR[I].[23:1];
128000         TRC[I*2+1]:=TR[I].[22:23];
129000         IF TEKEN1=1 THEN TRC[I*2]:=-TRC[I*2];
130000         IF TEKEN2=1 THEN TRC[I*2+1]:=-TRC[I*2+1];
131000      END;                                                    3 124000
132000   END;                                                       2 122000
133000
134000   PROCEDURE CONTINURI;
135000   BEGIN                                                      2
136000      IF KANID[KANAAL,2]=9 THEN %WINDRICHTING CONTINU MAKEN
137000      BEGIN                                                   3
138000         IF EERSTE[KANAAL-1] THEN
139000         BEGIN                                                4
140000            VORIGERI[KANAAL-1]:=INDEX; EERSTE[KANAAL-1]:=FALSE
141000         END                                                  4 139000
142000         ELSE
143000         BEGIN                                                4
144000            IF ABS(INDEX-VORIGERI[KANAAL-1])>90
145000            AND ABS(INDEX-VORIGERI[KANAAL-1])<270 THEN
146000            TRC[KANAAL-1]:=VORIGERI[KANAAL-1]*100;
147000            HULP:=TRC[KANAAL-1]/100;
148000            IF HULP-VORIGERI[KANAAL-1]<-180 THEN INDEX:=HULP+360 ELSE
149000            IF HULP-VORIGERI[KANAAL-1]>180 THEN INDEX:=HULP-360;
150000            VORIGERI[KANAAL-1]:=INDEX;
151000         END;                                                 4 143000
152000      END;                                                    3 137000
153000   END;                                                       2 135000
154000   PROCEDURE HEAD;
155000   BEGIN                                                      2
156000      WRITE(LP[SKIP 1]);
157000      WRITE(LP,<"TITLE LEESTAPE   STARTTIJD  EINDTIJD   INTERVAL">);
158000      WRITE(LP,</,"TRIVOM",I9,2I11,I7>,DTG,START,EIND,INTERVAL);
159000      WRITE(LP[SPACE 1]);
160000   END;                                                       2 155000
```

```
161000
162000      PROCEDURE KOP;
163000      THRU 4 DO WRITE(LP,<"KAN     N      GEM    STDEV      ">);
164000
165000      PI:=22/7;
166000      READ(KAART,<2I9>,DTG,INTERVAL)[EOF];
167000      READ(KAART,<2I9>,START,EIND)[EOF];
168000      HEAD;
169000      START:=START*1000; EIND:=EIND*1000;
170000      WRITE(LP,</,"KANALEN WAARVOOR KORRELATIEBEREKENING WORDT ",
171000      "UITGEVOERD">);
172000      I:=-1;
173000  H1:
174000      I:=I+1;
175000      READ(KAART,<2I4>,KOR[I,0],KOR[I,1])[EOK];
176000      WRITE(LP,</,2I4>,KOR[I,0],KOR[I,1]);
177000      IF KOR[I,0] NEQ 999 THEN GO H1;
178000      WRITE(LP,</,"KANALEN WAARVOOR GEM EN STDEV BEREKEND MOETEN WORDEN">);
179000      I:=-1;
180000  H2:
181000      I:=I+1;
182000      READ(KAART,<I4>,TEST[I])[EOK];
183000      WRITE(LP,</,I4>,TEST[I]);
184000      IF TEST[I] NEQ 999 THEN GO H2;
185000      WRITE(LP [SKIP 1]);
186000      REPLACE CH[0] BY "TRIVOM",DTG FOR 9 DIGITS,".";
187000      REPLACE TRIV.TITLE BY CH;
188000      FOR K:=0,1,2,3 DO
189000      BEGIN                                                          2
190000         READ(TRIV,26,TR[*])[EOF]; %LEES EN PRINT 4 IDENTIFICATIEBLOKKEN
191000         FOR I:=0 STEP 1 UNTIL 25 DO
192000         BEGIN                                                       3
193000            M[K*52+I*2]:=TR[I].[46:23];
194000            M[K*52+I*2+1]:=TR[I].[22:23];
195000         END;                                                        3 192000
196000      END;                                                           2 189000
197000      SJAAR:=M[0]; SDAGNR:=M[1]; SUUR:=M[2];
198000      SMIN:=M[3]; SSEC:=M[4]; SMSEC:=M[5];
199000      J:=-1;
200000      WRITE(LP,</,"STARTTIJD:",6I6>,THRU 6 DO M[J:=J+1]);
201000      WRITE(LP,</,"EINDTIJD :",6I6>,THRU 6 DO M[J:=J+1]);
202000      WRITE(LP,</,/,"KANAALIDENTIFICATIE">);
203000      FOR I:=12 STEP 4 UNTIL 168 DO
204000      BEGIN                                                          2
205000         KANNR:=M[I];
206000         IF KANNR=99999 THEN GO EX;
207000         IF KANNR<0 OR KANNR>40 THEN
208000         BEGIN                                                       3
209000            WRITE(LP,</,"FOUT KANAALNUMMER: ",I9>,KANNR);
210000            WRITE(LP,</,"RESTERENDE KANAALNUMMERS ZIJN: ">);
211000            GO EX;
212000         END;                                                        3 208000
213000         FOR J:=0,1,2,3 DO KANID[KANNR,J]:=M[I+J];
214000         IF KANID[KANNR,2]<0 OR KANID[KANNR,2]>25 THEN
215000         BEGIN                                                       3
216000            WRITE(LP,</,"SOORT INSTRUMENT FOUT: ",I9>,KANID[KANNR,2]);
217000            WRITE(LP,</,"RESTERENDE KANAALNUMMERS: ">);
218000            GO EX;
219000         END;                                                        3 215000
220000         J:=-1;
221000         WRITE(LP,</,4I5>,THRU 4 DO KANID[KANNR,J:=J+1]);
222000  EX:
223000      END;                                                           2 204000
224000
225000      J:=0; IF EIND<START THEN J:=1; %TOT IN VOLGENDE DAG
226000      EIND:=EIND+(SDAGNR+J)*10000000;
227000      START:=START+SDAGNR*10000000;
228000      WRITE(LP[SKIP 1]);
229000      TEL:=START+INTERVAL*1000;
230000      NULSTAR;
231000  HERH:
232000      READ(TRIV,26,TR[*])[EOF];
233000      VULTRC;
234000      IF (TRC[49] MOD 1000)*10000000+TRC[50]<START THEN
235000      BEGIN                                                          2
236000         SKREC:=SKREC+1; GO HERH
237000      END;                                                           2 235000
238000      IF SKREC NEQ 0 THEN
239000      BEGIN                                                          2
240000         WRITE(LP,</,"GESKIPT",I8," RECORDS">,SKREC); SKREC:=0
```

```
241000     END;                                                          2 239000
242000     FOR KANAAL:=1 STEP 1 UNTIL 40 DO
243000     BEGIN                                                         2
244000        INDEX:=TRC[KANAAL-1];
245000        IF INDEX=99999 THEN GO EXT ELSE INDEX:=INDEX/100;
246000        CONTINUE;
247000        INTEL[KANAAL,1]:=*+INDEX; %SOM
248000        INTEL[KANAAL,2]:=*+1; %AANTAL
249000        INTEL[KANAAL,3]:=*+INDEX*INDEX; %SOM V.D. KWADR.
250000  EXT:
251000     END;                                                          2 243000
252000
253000     I:=-1;
254000     WHILE KOR[I:=*+1,0] NEQ 999 DO
255000     BEGIN                                                         2
256000        KAN1:=KOR[I,0]; KAN2:=KOR[I,1];
257000        IF TRC[KAN1]=99999 THEN GO LL ELSE INDEX:=TRC[KAN1]/100;
258000        KANAAL:=KAN1;
259000        CONTINUE; PR1:=INDEX;
260000        IF TRC[KAN2]=99999 THEN GO LL ELSE INDEX:=TRC[KAN2]/100;
261000        KANAAL:=KAN2;
262000        CONTINUE; PR2:=INDEX;
263000        R[I,0]:=*+PR1*PR2;
264000        R[I,1]:=*+PR1;
265000        R[I,2]:=*+PR2;
266000        R[I,3]:=*+1;
267000  LL:
268000     END;                                                          2 255000
269000     IF (TRC[49] MOD 1000)*10000000+TRC[50]>OTEL THEN BEREKEN;
270000     IF (TRC[49] MOD 1000)*10000000+TRC[50]<EIND THEN GO HERH;
271000  EOF:
272000     BEREKEN;
273000     CLOSE(TRIV);
274000  EOK:
275000  END.                                                             1 1000
276000
```

```
1000   $SET LINEINFO
2000   $SET INSTALLATION
3000   BEGIN                                                                  1
4000    $SET OMIT
5000      ##################################################################
6000      #                                                                #
7000      #  OMREKENING VAM POOLKOORDINATEN NAAR CARTESISCHE KOORDINATEN   #
8000      #                                                                #
9000      #  DE INVOER BESTAAT UIT DE VOLGENDE KAARTEN                     #
10000     #  1  <I9>      FILE IDENTIFIER  VARIABELE :IFILE                #
11000     #  2  <6I2>     KANAALNUMMERS VAN DE TE GEBRUIKEN KANALEN        #
12000     #               1  WINDSNELHEID                                  #
13000     #               2  AZIMUTH                                       #
14000     #               3  ELEVATIE                                      #
15000     #               4  TEMPERATUUR(DROGE BOL)                        #
16000     #               5  TEMAPERATUUR(NATTE BOL)                       #
17000     #               6  Q(VOCHTIGHEID)                                #
18000     #                                                                #
19000     #  PROGRAMMEUR :P.NIEUWENDIJK                                    #
20000     #  CREATIONDATE : 771128                                         #
21000     #                                                                #
22000     ##################################################################
23000   $POP OMIT
24000   $PAGE
25000     ARRAY   ARCHIEF[0:25],BEGINTYD,EINDTYD,KANNR[0:5],
26000     WAARDE[0:51],IDENT[0:63,0:3],KANWERK[0:6,0:5];
27000
28000     INTEGER IFILE,I,J,ATRIV,BLOK,IQ,ITD,ITN,K;
29000
30000     REAL    FF,AZ,EL,U,V,W,Q,R;
31000
32000     BOOLEAN FOUT;
33000
34000     LABEL   EOC,EOF;
35000
36000     EBCDIC ARRAY TITEL[0:131];
37000
38000     FILE CARD(KIND=READER,
39000     UNITS=CHARACTERS)
40000     ,TAPE(KIND=PETAPE,
41000     UNITS=WORDS,
42000     BLOCKSIZE=2600,
43000     MAXRECSIZE=26)
44000     ,OUT (KIND=PETAPE,
45000     SAVEFACTOR=999,
46000     BLOCKSIZE=2600,
47000     MAXRECSIZE=26,
48000     MYUSE=OUT)
49000     ,LINE(KIND=PRINTER,
50000     UNITS=CHARACTERS);
51000
52000     DEFINE ONTBREKEND=99999#,
53000     PAKUIT=FOR I:=0 STEP 1 UNTIL 51 DO
54000     WAARDE[I]:=
55000     IF ARCHIEF[K:=I.[47:47]].[J:=(47-I.[0:1]*24):1]=1 THEN
56000     -ARCHIEF[K].[J-1:23] ELSE
57000     ARCHIEF[K].[J-1:23]#
58000     ,PAKIN =FOR I:=0 STEP 1 UNTIL 51 DO
59000     BEGIN                                                              2
60000        ARCHIEF[K:=I.[47:47]].[J:=(47-I.[0:1]*24):1]:=
61000        REAL(WAARDE[I]<0);
62000        ARCHIEF[K].[J-1:23]:=INTEGER(ABS(WAARDE[I]));
63000     END#;                                                              2 59000
64000
65000   $INCLUDE "HEADING."
66000
67000     REAL PROCEDURE RESTORE;
68000     BEGIN                                                              2
69000        OWN INTEGER I;
70000        RESTORE:=WAARDE[I];
71000        IF I:=I+1 > 51 THEN I:=0;
72000     END;                                                               2 68000
73000
74000     PROCEDURE ERROR(X);
75000     VALUE X;
76000     INTEGER X;
77000     BEGIN                                                              2
78000        EBCDIC ARRAY STRING[0:131];
79000        DEFINE REP=REPLACE STRING BY #;
80000
```

```
81000       REP " " FOR 132;
82000       CASE X OF
83000       BEGIN                                                    3
84000       1:
85000          REP "FILE IDENTIFIER FOUT";
86000       2:
87000          REP "FOUTE TRIVAAN KAART INVOER";
88000       END;                                                     3 83000
89000       WRITE(LINE,132,STRING);
90000       MYSELF.STATUS:=-1;
91000    END                                                         2 77000
92000    ERROR;
93000
94000    % I N I T I A L I S A T I E
95000
96000    HEADING(LINE,"OMREKENING");
97000    NAME(LINE,    "POOLKOORDINATEN");
98000    NAME(LINE,    "NAAR CARTESISCHE");
99000    NAME(LINE,    "KOORDINATEN");
100000   WRITE(LINE(SKIP 1));
101000
102000   % I N V O E R F I L E I D E N T I F I E R
103000
104000   READ(CARD,<I9>,IFILE);
105000   I:=IFILE DIV 10000000;
106000   IF I<77 OR I>80 THEN FOUT:=TRUE;
107000   I:=(J:=IFILE MOD 10000000) DIV 10000;
108000   IF I<1 OR I>365 THEN FOUT:=TRUE;
109000   I:=(J:=J MOD 10000) DIV 100;
110000   IF I<0 OR I>23 THEN FOUT:=TRUE;
111000   I:=IFILE MOD 100;
112000   IF I<0 OR I>59 THEN FOUT:=TRUE;
113000   IF FOUT THEN ERROR(1);
114000   REPLACE TITEL BY " " FOR 132;
115000   REPLACE TITEL BY "TRIVOM",IFILE FOR 9 DIGITS,".";
116000   REPLACE TAPE.TITLE BY TITEL;
117000   TAPE.OPEN:=TRUE;
118000   WRITE(LINE,<" INPUT FILE TITLE ",A16," SERIALNO ",A6>,
119000   TITEL,TAPE.SERIALNO);
120000   REPLACE TITEL BY "TRICAR",IFILE FOR 9 DIGITS,".";
121000   REPLACE OUT.TITLE BY TITEL;
122000   OUT.OPEN:=TRUE;
123000   WRITE(LINE,<"OUTPUT FILE TITLE ",A16," SERIALNO ",A6,//>,
124000   TITEL,OUT.SERIALNO);
125000
126000   % I N V O E R  I D E N T I F I K A T I E  M A G N E E T B A N D
127000
128000   READ(TAPE,26,ARCHIEF);
129000   WRITE(OUT,26,ARCHIEF);
130000   PAKUIT;
131000   FOR I:=0 STEP 1 UNTIL 5 DO
132000   BEGINTYD[I]:=RESTORE;
133000   FOR I:=0 STEP 1 UNTIL 5 DO
134000   EINDTYD[I]:=RESTORE;
135000   FOR I:=0 STEP 1 UNTIL 9 DO
136000   BEGIN                                                    2
137000      J:=RESTORE-1;
138000      IDENT[J,0]:=J+1;
139000      IDENT[J,1]:=RESTORE;
140000      IDENT[J,2]:=RESTORE;
141000      IDENT[J,3]:=RESTORE;
142000   END;                                                     2 136000
143000   THRU 3 DO
144000   BEGIN                                                    2
145000      READ(TAPE,26,ARCHIEF);
146000      WRITE(OUT,26,ARCHIEF);
147000      PAKUIT;
148000      FOR I:=0 STEP 1 UNTIL 12 DO
149000      BEGIN                                                 3
150000         J:=RESTORE-1;
151000         IF J>0 AND J<64 THEN
152000         BEGIN                                              4
153000            IDENT[J,0]:=J+1;
154000            IDENT[J,1]:=RESTORE;
155000            IDENT[J,2]:=RESTORE;
156000            IDENT[J,3]:=RESTORE;
157000         END                                                4 152000
158000      END                                                   3 149000
159000   END;                                                     2 144000
160000
```

```
161000     % P R I N T  M A G N E E T B A N D I D E N T I F I K A T I E
162000
163000     FOR I:=0 STEP 1 UNTIL 63 DO
164000     IF IDENT[I,0] NEQ 0 THEN
165000     WRITE(LINE,<I2" KAN "I2" UITHOUDER "I2" INST. "I2" PL "J5>,
166000     I+1,IDENT[I,0],IDENT[I,1],IDENT[I,2],IDENT[I,3]);
167000
168000     % I N V O E R  T R I V A A N K A N A A L N U M M E R S
169000
170000     WHILE TRUE DO
171000     BEGIN                                                          2
172000        READ(CARD,<6I2>,KANNR)[EOC];
173000        FOUT:=FALSE;
174000        FOR I:=0 STEP 1 UNTIL 4 DO
175000        IF KANNR[I]<1 AND KANNR[I]>40 THEN
176000        IF I>2 AND KANNR[I]=0 THEN ELSE FOUT:=TRUE;
177000        IF FOUT THEN ERROR(2);
178000        REPLACE KANWERK[ATRIV,0] BY KANNR FOR 6 WORDS;
179000        ATRIV:=ATRIV+1;
180000     END;                                                           2 171000
181000
182000     % P R I N T E N  V A N  K A A R T I N V O E R
183000
184000  EOC:
185000     WRITE(LINE,<//"AANTAL TRIVANEN "J2,/>,ATRIV);
186000     FOR I:=0 STEP 1 UNTIL ATRIV-1 DO
187000     IF KANWERK[I,0] NEQ 0 THEN
188000     WRITE(LINE,<"TRIVAAN "I2,/,
189000     " FF KAN "I2,/,
190000     " AZ KAN "I2,/,
191000     " EL KAN "I2,/,
192000     " TD KAN "I2,/,
193000     " TN KAN "I2,/,
194000     " Q  KAN "I2,//>,I+1,KANWERK[I,*]);
195000     FOR I:=0 STEP 1 UNTIL ATRIV-1 DO
196000     FOR J:=0,1,2,3,4,5 DO KANWERK[I,J]:=*-1;
197000
198000     % V E R W E R K I N G
199000
200000     WRITE(LINE[SKIP 1]);
201000     NAME(LINE,"RESULTATEN");
202000     WHILE TRUE DO
203000     BEGIN                                                          2
204000        READ(TAPE,26,ARCHIEF)[EOF];
205000        PAKUIT;
206000        BLOK:=BLOK+1;
207000        IF BLOK=3000 THEN
208000        WRITE(LINE,<"VOOR OMREKENING",/,
209000        "DATUM "I6" TIJD "I8" SAMPLE "I6,/,
210000        7(7(" K="I2" W="I8),/)>,
211000        WAARDE[49],WAARDE[50],WAARDE[51],
212000        FOR I:=0 STEP 1 UNTIL 48 DO
213000        (I+1,WAARDE[I]));
214000        FOR I:=0 STEP 1 UNTIL ATRIV-1 DO
215000        BEGIN                                                       3
216000           FF:=
217000           IF KANWERK[I,0]<0 THEN ONTBREKEND ELSE WAARDE[KANWERK[I,0]];
218000           AZ:=
219000           IF KANWERK[I,1]<0 THEN ONTBREKEND ELSE WAARDE[KANWERK[I,1]];
220000           EL:=
221000           IF KANWERK[I,2]<0 THEN ONTBREKEND ELSE WAARDE[KANWERK[I,2]];
222000           IF FF=ONTBREKEND  OR EL=ONTBREKEND THEN U:=V:=W:=ONTBREKEND
223000           ELSE
224000           BEGIN                                                    4
225000              EL:=(EL/100)*0.01745329252;
226000              W:= FF * SIN(EL);
227000              IF AZ=ONTBREKEND THEN U:=V:=ONTBREKEND ELSE
228000              BEGIN                                                 5
229000                 U:=
230000                 (R:=FF*(EL:=COS(EL)))*SIN(AZ:=(AZ/100-180)*0.01745);
231000                 V:=R*COS(AZ);
232000              END;                                                  5 228000
233000           END;                                                     4 224000
234000           IF KANWERK[I,0]>=0 THEN WAARDE[KANWERK[I,0]]:=U;
235000           IF KANWERK[I,1]>=0 THEN WAARDE[KANWERK[I,1]]:=V;
236000           IF KANWERK[I,2]>=0 THEN WAARDE[KANWERK[I,2]]:=W;
237000           ITD:=KANWERK[I,3];
238000           ITN:=KANWERK[I,4];
239000           IQ:=KANWERK[I,5];
240000           IF ITD >= 0 AND ITN >= 0 AND IQ >= 0 THEN
```

```
241000            Q:=(.616
242000               *10**(7.5*(WAARDE[ITN]/100)/(237.3+(WAARDE[ITN]/100))+.78571)
243000               -.41*(WAARDE[ITD]-WAARDE[ITN])/100)*100  ELSE
244000               Q:=99999;
245000               IF IQ >= 0 THEN WAARDE[IQ]:=Q;
246000            END;
247000            IF BLOK=3000 THEN                                              3 215000
248000            BEGIN
249000               BLOK:=0;                                                    3
250000               WRITE(LINE,</"NA OMREKENING"/"DATUM "I6" TIJD "I8,
251000               " SAMPLE "I8,/,
252000               7(7(" K="I2" W="I8),/)>,
253000               WAARDE[49],WAARDE[50],WAARDE[51],
254000               FOR I:=0 STEP 1 UNTIL 48 DO
255000               (I+1,WAARDE[I]));
256000            END;
257000            PAKIN;
258000            WRITE(OUT,26,ARCHIEF);                                         3 248000
259000         END;
260000
261000         % A F H A N D E L I N G                                           2 203000
262000
263000   EOF:
264000         WRITE(LINE,<//,"GELEZEN RECORDS          ",I10,/,
265000         "GESCHREVEN RECORDS       ",I10>,
266000         TAPE.RECORD+1,OUT.RECORD+1);
267000         LOCK(OUT);
268000   END.
269000
270000                                                                          1 3000
271000         WRITE(LINE,<//,"GELEZEN RECORDS          ",I10,/,
272000         "GESCHREVEN RECORDS       ",I10>,
273000         TAPE.RECORD+1,OUT.RECORD+1);
274000         LOCK(OUT);
275000   END.
                                                                               1 3000
```

```
1000   $SET VECTORMODE
2000   $SET LINEINFO
3000   $SET LISTINCL
4000   $SET INSTALLATION
5000  BEGIN                                                              1
6000   $SET OMIT
7000     ##################################################################
8000     #                                                                #
9000     #  BEREKENEN VAN GEMIDDELDEN,TREND,KORRELATIES,TRIPEL-           #
10000    #  KORRELATIES EN AFGELEIDE GROOTHEDEN.                          #
11000    #                                                                #
12000    #  DE INVOER BESTAAT UIT DE VOLGENDE KAARTEN                     #
13000    #  1  <I9>      FILE IDENTIFIER  VARIABELE :IFILE                #
14000    #  2  <2I9,I2>  BEGINDATUM,EINDDATUM EN INTERVAL VAN DE TE       #
15000    #               VERWERKEN PERIODE.                               #
16000    #  3  <6I2>     PER TRIVAAN DE                                   #
17000    #               KANAALNUMMERS VAN DE TE GEBRUIKEN KANALEN        #
18000    #               1  WINDSNELHEID                                  #
19000    #               2  AZIMUTH                                       #
20000    #               3  ELEVATIE                                      #
21000    #               4  TEMPERATUUR(DROGE BOL)                       #
22000    #               5  TEMAPERATUUR(NATTE BOL)                      #
23000    #               6  Q(VOCHTIGHEID)                               #
24000    #                                                                #
25000    #  PROGRAMMEUR :P.NIEUWENDIJK                                    #
26000    #  CREATIONDATE : 780316                                         #
27000    #                                                                #
28000    #  UITBREIDING :1 MAART 1979.                                    #
29000    #  M.B.V. DE TASKVALUE KAN WORDEN OPGEGEVEN OF EEN SELEKTIE      #
30000    #  VAN DE BEREKENDE RESULTATEN MOET WORDEN GEARCHIVEERD.         #
31000    #  (TASKVALUE=1).                                                #
32000    #                                                                #
33000    ##################################################################
34000   $POP OMIT
35000   $PAGE
36000     ARRAY    ARCHIEF[0:25],BEGINTYD,EINDTYD,KANNR[0:5],
37000     ASA,SCANAR,MAARDE[0:51],
38000     IDENT[0:63,0:3],KANWERK[0:6,0:5],
39000     SAVEAR[0:208],
40000     SOM,SSA,SKA,LIA,LIB,PRD[0:51],
41000     ATRIP,TRIPEL[0:6,0:5,0:5,0:5],
42000     RR,FLUX,FLUXT,AFLUX,FLUXLOK[0:6,0:5,0:5],
43000     SIGMA,DJ[0:6,0:5],
44000     USTER,TSTER,LL,FI,DD,FF,TRFF1,TRA1[0:6],UF[0:51];
45000
46000     INTEGER IFILE,I,J,ATRIV,BLOK,IQ,ITD,ITN,K,
47000     IU,IV,IW,NRRECS,L,M,I1,I2,I3,
48000     VALJOE,INDEX,
49000     BEGINPERIODE,EINDPERIODE,INTERVAL,CHECK,CHECK0,CHECK1;
50000
51000     REAL  U,V,W,Q,RAD,PI,R,UN,VN,FN,FO,PI2,HPI,PIHPI;
52000
53000     BOOLEAN FOUT,ENDOFFILE;
54000     BOOLEAN ARRAY AANWEZIG[0:51];
55000     ARRAY REFERENCE REF[0],RT3[0,0,0],RT2[0,0],RT1[0],
56000     RAT3[0,0,0],RAT2[0,0],RAT1[0],
57000     RFL2[0,0],RAFL2[0,0],RFL1[0],RAFL1[0],
58000     RFLOK[0],RFLUXT[0];
59000
60000     LABEL   EOC,EOF,L1,L4,L5,L6,L7,L8,L9,L10,L11,L12,L13;
61000
62000     EBCDIC ARRAY TITEL[0:131],RULE[0:59,0:131];
63000     POINTER PTR1,PTR2;
64000
65000     FILE CARD(KIND=READER,
66000     UNITS=CHARACTERS)
67000     ,TAPE(KIND=PETAPE,
68000     UNITS=WORDS,
69000     BLOCKSIZE=2600,
70000     MAXRECSIZE=26)
71000     ,OUT (KIND=DISK,XXXXXXX    TEMPORARY WORKFILE  XXXXXXXXXXX
72000     BLOCKSIZE=3120,
73000     MAXRECSIZE=52,
74000     FLEXIBLE,
75000     AREAS=300,
76000     AREASIZE=60)
77000     ,SAVE(KIND=DISK,XXXXXXX    OPTIONELE FILE VOOR RESULTATEN  XXXX
78000     MAXRECSIZE=209,
79000     UNITS=WORDS,
80000     AREASIZE=1,
```

```
81000      AREAS=72)
82000      ,LINE(KIND=PRINTER,
83000      UNITS=CHARACTERS);
84000
85000      DEFINE ONTBREKEND=99999#,
86000      SAVERESULTS = VALJOE=1#,
87000      DIGITSIN(X)=(FIRSTONE(SCALERIGHTF(X,12))-1) DIV 4 +1#,
88000      PAKUIT=FOR I:=0 STEP 1 UNTIL 51 DO
89000      IF AANWEZIG[I] THEN
90000      WAARDE[I]:=
91000      IF ARCHIEF[K:=I.[47:47]].[J:=(47-I.[0:1]*24):1]=1 THEN
92000      -ARCHIEF[K].[J-1:23] ELSE
93000      ARCHIEF[K].[J-1:23]#
94000      ,PAKIN =FOR I:=0 STEP 1 UNTIL 51 DO
95000      BEGIN                                                          2
96000         ARCHIEF[K:=I.[47:47]].[J:=(47-I.[0:1]*24):1]:=
97000         REAL(WAARDE[I]<0);
98000         ARCHIEF[K].[J-1:23]:=INTEGER(ABS(WAARDE[I]));
99000      END#;                                                          2 95000
100000
101000   $INCLUDE "HEADING."
102000
103000     REAL PROCEDURE RESTORE;
104000     BEGIN                                                          2
105000        OWN INTEGER I;
106000        RESTORE:=WAARDE[I];
107000        IF I:=I+1 > 51 THEN I:=0;
108000     END;                                                           2 104000
109000
110000
111000     % P R O C .  T O  U P D A T E  D A T E  W I T H  I N T E R V A L
112000
113000     INTEGER PROCEDURE UPDATE(CHECK,INTERVAL);
114000     VALUE    CHECK,INTERVAL;
115000     INTEGER  CHECK,INTERVAL;
116000     BEGIN                                                          2
117000        INTEGER M,U,D,J,DUMMY;
118000
119000        M:=CHECK MOD 100;
120000        U:=(DUMMY:=CHECK DIV 100) MOD 100;
121000        D:=(DUMMY:= * DIV 100) MOD 1000;
122000        J:=CHECK DIV 10000000;
123000
124000        IF M:=M+INTERVAL >= 60 THEN
125000        BEGIN                                                       3
126000           M:=M-60;
127000           IF U:=U+1 > 23 THEN
128000           IF D:=U+1+U:=0 > 365+REAL(J MOD 4 = 0) THEN J:=J+D:=1;
129000        END;                                                        3 125000
130000
131000        UPDATE:=((J*1000+D)*100+U)*100+M;
132000
133000     END                                                            2 116000
134000     UPDATE;
135000
136000     % E R R O R  A F H A N D E L I N G
137000
138000     PROCEDURE ERROR(X);
139000     VALUE X;
140000     INTEGER X;
141000     BEGIN                                                          2
142000        EBCDIC ARRAY STRING[0:131];
143000        DEFINE REP=REPLACE STRING BY #;
144000
145000        REP " " FOR 132;
146000        CASE X OF
147000        BEGIN                                                       3
148000        1:
149000           REP "FILE IDENTIFIER FOUT";
150000        2:
151000           REP "FOUTE TRIVAAN KAART INVOER";
152000        3:
153000           REP "FOUTE OPGAAF BEGINPERIODE ";
154000        4:
155000           REP "TE VERWERKEN PERIODE FOUT";
156000        END;                                                        3 147000
157000        WRITE(LINE,132,STRING);
158000        MYSELF.STATUS:=-1;
159000     END                                                            2 141000
160000     ERROR;
```

```
161000
162000      % F O R M A T  P R O C E D U R E
163000
164000      REAL PROCEDURE FORM(R,I1,I2);
165000      VALUE R,I1,I2;
166000      REAL  R,I1,I2;
167000      BEGIN                                                          2
168000         ARRAY A[0:0];
169000         REAL Y;
170000         INTEGER I;
171000         POINTER PA;
172000         DEFINE DIGITSIN(X)=(FIRSTONE(SCALERIGHTF(X,12))-1) DIV 4 +1#,
173000         DIG(X,I)    =Y:=X FOR MIN(DIGITSIN(Y),I) DIGITS#,
174000         F(X,W,D)    =DIG((X DIV POTL[D]),W-1-D),".",
175000         (INTEGER(X) MOD POTL[D]) FOR D DIGITS#;
176000         PA:=POINTER(A[0]);
177000         REPLACE PA BY " " FOR 6;
178000         IF R=ONTBREKEND THEN
179000         REPLACE PA BY " " FOR 5-I1-I2,"+" FOR I1+I2 ELSE
180000         BEGIN                                                       3
181000            I:=DIGITSIN(R);
182000            IF I>I1 OR I+I2+REAL(R<0)>5 OR R>=POTL[I1+1] THEN
183000            REPLACE PA BY R FOR 6 NUMERIC ELSE
184000            BEGIN                                                    4
185000               PA:=PA+(5-I2-I-REAL(R<0));
186000               IF R<0 THEN REPLACE PA:PA BY "-";
187000               R:=INTEGER(ABS(R)*POTL[I2]);
188000               REPLACE PA BY F(R,(I1+I2+1),I2);
189000            END;                                                     4 184000
190000         END;                                                        3 180000
191000         FORM:=A[0];
192000      END;                                                           2 167000
193000
194000
195000      % I N I T I A L I S A T I E
196000
197000
198000      % RESULTATEN ARCHIVEREN
199000
200000      VALJOE:=MYSELF.TASKVALUE;
201000
202000      HEADING(LINE,"BEREKENING");
203000      NAME(LINE,   "TREND");
204000      NAME(LINE,   "KORRELATIE");
205000      NAME(LINE,   "EN AFGELEIDE");
206000      NAME(LINE,   "GROOTHEDEN");
207000      WRITE(LINE[SKIP 1]);
208000
209000      % I N V O E R  F I L E  I D E N T I F I E R
210000
211000      READ(CARD,<I9>,IFILE);
212000      I:=IFILE DIV 10000000;
213000      IF I<77 OR I>80 THEN FOUT:=TRUE;
214000      I:=(J:=IFILE MOD 10000000) DIV 10000;
215000      IF I<1 OR I>365 THEN FOUT:=TRUE;
216000      I:=(J:=J MOD 10000) DIV 100;
217000      IF I<0 OR I>23 THEN FOUT:=TRUE;
218000      I:=IFILE MOD 100;
219000      IF I<0 OR I>59 THEN FOUT:=TRUE;
220000      IF FOUT THEN ERROR(1);
221000      REPLACE TITEL BY " " FOR 132;
222000      REPLACE TITEL BY "TRICAR",IFILE FOR 9 DIGITS,".";
223000      REPLACE TAPE.TITLE BY TITEL;
224000      TAPE.OPEN:=TRUE;
225000      WRITE(LINE,<" INPUT FILE TITLE ",A16," SERIALNO ",A6,//>,
226000      TITEL,TAPE.SERIALNO);
227000
228000      % I N V O E R  I D E N T I F I K A T I E  M A G N E E T B A N D
229000
230000      FOR I:=0 STEP 1 UNTIL 51 DO AANWEZIG[I]:=TRUE;
231000      READ(TAPE,26,ARCHIEF);
232000      PAKUIT;
233000      FOR I:=0 STEP 1 UNTIL 5 DO
234000      BEGINTYD[I]:=RESTORE;
235000      FOR I:=0 STEP 1 UNTIL 5 DO
236000      EINDTYD[I]:=RESTORE;
237000      FOR I:=0 STEP 1 UNTIL 9 DO
238000      BEGIN                                                          2
239000         J:=RESTORE-1;
240000         IDENT[J,0]:=J+1;
```

```
241000        IDENT[J,1]:=RESTORE;
242000        IDENT[J,2]:=RESTORE;
243000        IDENT[J,3]:=RESTORE;
244000     END;                                                    2 238000
245000     THRU 3 DO
246000     BEGIN                                                   2
247000        READ(TAPE,26,ARCHIEF);
248000        PAKUIT;
249000        FOR I:=0 STEP 1 UNTIL 12 DO
250000        BEGIN                                                3
251000           J:=RESTORE-1;
252000           IF J>0 AND J<64 THEN
253000           BEGIN                                             4
254000              IDENT[J,0]:=J+1;
255000              IDENT[J,1]:=RESTORE;
256000              IDENT[J,2]:=RESTORE;
257000              IDENT[J,3]:=RESTORE;
258000           END                                               4 253000
259000        END                                                  3 250000
260000     END;                                                    2 246000
261000
262000     % P R I N T  M A G N E E T B A N D I D E N T I F I K A T I E
263000
264000     WRITE(LINE,<"I D E N T I F I K A T I E",/>);
265000     FOR I:=0 STEP 1 UNTIL 63 DO
266000     IF IDENT[I,0] NEQ 0 THEN
267000     WRITE(LINE,<I2" KAN "I2" UITHOUDER "I2" INST. "I2" PL "J5>,
268000     I+1,IDENT[I,0],IDENT[I,1],IDENT[I,2],IDENT[I,3]);
269000
270000     % I N V O E R  T E  V E R W E R K E N  P E R I O D E
271000
272000     READ(CARD,<2I9,I2>,BEGINPERIODE
273000     ,EINDPERIODE
274000     ,INTERVAL);
275000     IF EINDPERIODE < BEGINPERIODE THEN ERROR(4);
276000     WRITE(LINE,</"TE VERWERKEN PERIODE:"I9" T/M "I9,
277000     " GEBRUIKTE INTERVAL:"I2" MINUTEN.">,
278000     BEGINPERIODE,EINDPERIODE,INTERVAL);
279000     OUT(AREAS=INTEGER(600*INTERVAL/30));
280000
281000     %INDIEN ARCHIVEREN INITIALISEREN DISKFILE
282000
283000     IF SAVERESULTS THEN
284000     BEGIN                                                   2
285000        REPLACE TITEL BY "KOR/",BEGINPERIODE FOR 9 DIGITS,
286000        "/",EINDPERIODE FOR 9 DIGITS,"/",INTERVAL FOR 2 DIGITS,".";
287000        REPLACE SAVE.TITLE BY TITEL;
288000        WRITE(LINE,<"RESULTATEN GEARCHIVEERD IN FILE:",A27>,TITEL);
289000     END;                                                    2 284000
290000
291000
292000     % I N V O E R  T R I V A A N K A N A A L N U M M E R S
293000
294000     WHILE TRUE DO
295000     BEGIN                                                   2
296000        READ(CARD,<6I2>,KANNR)[EOC];
297000        FOUT:=FALSE;
298000        FOR I:=0 STEP 1 UNTIL 4 DO
299000        IF KANNR[I]<1 OR KANNR[I]>40 THEN
300000        IF I>2 AND KANNR[I]=0 THEN ELSE FOUT:=TRUE;
301000        IF FOUT THEN ERROR(2);
302000        REPLACE KANWERK[ATRIV,0] BY KANNR FOR 6 WORDS;
303000        ATRIV:=*+1;
304000     END;                                                    2 295000
305000
306000     % P R I N T E N  V A N  K A A R T I N V O E R
307000
308000  EOC:
309000     WRITE(LINE,<//"AANTAL TRIVANEN "J2,/>,ATRIV);
310000     FOR I:=0 STEP 1 UNTIL ATRIV-1 DO
311000     IF KANWERK[I,0] NEQ 0 THEN
312000     WRITE(LINE,<"TRIVAAN "I2,/,
313000     " FF KAN "I2,/,
314000     " AZ KAN "I2,/,
315000     " EL KAN "I2,/,
316000     " TD KAN "I2,/,
317000     " TN KAN "I2,/,
318000     " Q  KAN "I2,//>,I+1,KANWERK[I,*]);
319000     ATRIV:=*-1;
320000     FOR I:=0 STEP 1 UNTIL ATRIV DO
```

```
321000      FOR J:=0,1,2,3,4,5 DO KANWERK[I,J]:=*-1;
322000      WRITE(LINE[SKIP 1]);
323000
324000      % I N I T I A L I S A T I E  C O N S T A N T E N
325000
326000      RAD:=180/(PI:=ARCTAN(1)*4);
327000      PI2:=PI*2;
328000      HPI:=PI/2;
329000      PIHPI:=PI+HPI;
330000      FOR I:=0 STEP 1 UNTIL 48 DO AANWEZIG[I]:=FALSE;
331000      FOR I:=0 STEP 1 UNTIL ATRIV DO
332000      FOR J:=0,1,2,3,4,5 DO
333000      BEGIN                                                          2
334000         RR[I,J,J]:=1;
335000         IF K:=KANWERK[I,J]>=0 THEN AANWEZIG[K]:=TRUE;
336000      END;                                                           2 333000
337000
338000      % Z O E K  S T A R T P U N T  O P  M A G N E E T B A N D
339000
340000      DO
341000      BEGIN                                                          2
342000         READ(TAPE,26,ARCHIEF);
343000         PAKUIT;
344000         CHECK:=WAARDE[49]*10000+WAARDE[50] DIV 1000;
345000      END                                                            2 341000
346000      UNTIL CHECK>=BEGINPERIODE;
347000      CHECK:=BEGINPERIODE;
348000
349000      % V E R W E R K I N G
350000
351000      DO
352000      BEGIN                                                          2
353000         CHECK:=UPDATE(CHECKO:=CHECK,INTERVAL);
354000
355000         % I N I T I A L I S E R E N  A R R A Y S  S O M M A T I E S
356000
357000         DO VECTORMODE(SOM,PRD,ASA,FOR 52)
358000         BEGIN                                                       3
359000            SOM:=0;
360000            PRD:=0;
361000            ASA:=0;
362000            INCREMENT SOM,PRD,ASA;
363000         END;                                                        3 358000
364000
365000         % I N T E L L E N  E N  D U M P E N  N A A R  W O R K F I L E
366000
367000   L1:
368000         DO
369000         BEGIN                                                       3
370000            FOR I:=0 STEP 1 UNTIL 48 DO
371000            IF WAARDE[I] = ONTBREKEND OR NOT AANWEZIG[I] THEN ELSE
372000            BEGIN                                                    4
373000               R:=(WAARDE[I]:=*/100);
374000               SOM[I]:=*+R;
375000               ASA[I]:=*+1;
376000               PRD[I]:=*+R*ASA[I];
377000            END;                                                     4 372000
378000            WRITE(OUT,52,WAARDE);
379000            READ(TAPE,26,ARCHIEF)[EOF];
380000            PAKUIT;
381000            CHECK1:=WAARDE[49]*10000+WAARDE[50] DIV 1000;
382000         END                                                         3 369000
383000         UNTIL CHECK<=CHECK1;
384000    $SET OMIT=NOT TESTUITVOER
385000         WRITE(LINE,<//,"VALUES NA INTELLEN">);
386000         WRITE(LINE,[12]/[10],SOM);
387000         WRITE(LINE,[12]/[10],ASA);
388000         WRITE(LINE,[12]/[10],PRD);
389000    $POP OMIT
390000
391000         % B E R E K E N E N  E E R S T E  R E S U L T A T E N
392000
393000         IF FALSE THEN
394000         BEGIN                                                       3
395000   EOF:
396000            ENDOFFILE:=TRUE;
397000         END;                                                        3 394000
398000         FOR I:=0 STEP 1 UNTIL 48 DO
399000         IF ASA[I] > 1 THEN
400000         BEGIN                                                       3
```

```
401000          SSA[I]:=(0.5*(ASA[I]+1));
402000          SKA[I]:=((ASA[I]**2-1)/12);
403000          SOM[I]:=**/ASA[I];
404000          LIA[I]:=PRD[I]/ASA[I] - SSA[I]*SOM[I];
405000          LIA[I]:=**/SKA[I];
406000          LIB[I]:=SOM[I]-LIA[I]*SSA[I];
407000       END                                                    3 400000
408000       ELSE SOM[I]:=LIA[I]:=LIB[I]:=ONTBREKEND;
409000  $SET OMIT=NOT TESTUITVOER
410000       WRITE(LINE,[12]/[10],ASA);
411000       WRITE(LINE,[12]/[10],SSA);
412000       WRITE(LINE,[12]/[10],SKA);
413000       WRITE(LINE,[12]/[10],SOM);
414000       WRITE(LINE,[12]/[10],LIA);
415000       WRITE(LINE,[12]/[10],LIB);
416000       WRITE(LINE[SKIP 1]);
417000  $POP OMIT
418000
419000       % P E R  T R I V A A N  1E  B E R E K E N I N G E N
420000  L4:
421000
422000       FOR I:=0 STEP 1 UNTIL ATRIV DO
423000       BEGIN                                                  3
424000          REF:=KANWERK[I,*];
425000          U:=IF REF[0]<0 THEN ONTBREKEND ELSE SOM[REF[0]];
426000          V:=IF REF[1]<0 THEN ONTBREKEND ELSE SOM[REF[1]];
427000          IF U=0 OR  U=ONTBREKEND OR V=ONTBREKEND THEN
428000          DD[I]:=FF[I]:=FI[I]:=
429000          TRFF1[I]:=TRA1[I]:=ONTBREKEND ELSE
430000          BEGIN                                               4
431000             IU:=REF[0];
432000             IV:=REF[1];
433000             FI[I]:=ARCTAN(V/U);
434000             IF V<0 THEN
435000             IF U<0 THEN FI[I]:=*+PI ELSE FI[I]:=*+PI2
436000             ELSE
437000             IF U<0 THEN FI[I]:=*+PI ELSE;
438000             IF DD[I]:=PIHPI-FI[I]<0 THEN DD[I]:=*+PI2;
439000             FF[I]:=SQRT(U**2+V**2);
440000             UN:=LIA[IU]*ASA[IU]+LIB[IU];
441000             VN:=LIA[IV]*ASA[IV]+LIB[IV];
442000             FN:=SQRT(UN**2+VN**2);
443000             FO:=SQRT(LIB[IU]**2+LIB[IV]**2);
444000             TRFF1[I]:=(FN-FO);
445000             FN:=ARCTAN(VN/UN);
446000             IF VN<0 THEN
447000             IF UN<0 THEN FN:=*+PI ELSE FN:=*+PI2
448000             ELSE
449000             IF UN<0 THEN FN:=*+PI ELSE;
450000             FO:=ARCTAN(LIB[IV]/LIB[IU]);
451000             IF LIB[IV]<0 THEN
452000             IF LIB[IU]<0 THEN FO:=*+PI ELSE FO:=*+PI2
453000             ELSE
454000             IF LIB[IU]<0 THEN FO:=*+PI ELSE;
455000             IF FO:=PIHPI-FO<0 THEN FO:=*+PI2;
456000             IF FN:=PIHPI-FN<0 THEN FN:=*+PI2;
457000             TRA1[I]:=(FN-FO)*RAD;
458000             FOR J:=0,1,2,3,4,5 DO
459000             DJ[I,J]:=IF I1:=REF[J]<0 THEN ONTBREKEND ELSE
460000             LIA[I1]*ASA[I1];
461000          END                                                4 430000
462000       END;                                                  3 423000
463000
464000       % I N I T I A L I S A T I E  2E  B E R E K E N I N G E N
465000
466000  L5:
467000       NRRECS:=OUT.RECORD+1;
468000       SEEK(OUT[0]);
469000
470000       FOR I:=0 STEP 1 UNTIL 6 DO
471000       FOR J:=0 STEP 1 UNTIL 5 DO
472000       BEGIN                                                  3
473000          REPLACE FLUX[I,J,*] BY 0 FOR 6 WORDS;
474000          REPLACE AFLUX[I,J,*] BY 0 FOR 6 WORDS;
475000          REPLACE FLUXT[I,J,*] BY 0 FOR 6 WORDS;
476000          REPLACE FLUXLOK[I,J,*] BY 0 FOR 6 WORDS;
477000          FOR K:=0 STEP 1 UNTIL 5 DO
478000          BEGIN                                              4
479000             REPLACE TRIPEL[I,J,K,*] BY 0 FOR 6 WORDS;
480000             REPLACE ATRIP[I,J,K,*] BY 0 FOR 6 WORDS;
```

```
            END;                                                        4 478000
         END;                                                           3 472000

         %T R E N D C O R R E C T I E + I N T E L L E N   K O R R E L A T I

   L6:
         FOR M:=1 STEP 1 UNTIL NRRECS DO
         BEGIN                                                          3
            READ(OUT,52,SCANAR);
            FOR I:=0 STEP 1 UNTIL 48 DO
            IF SCANAR[I] NEQ ONTBREKEND AND AANWEZIG[I] THEN
            UF[I]:=(SCANAR[I])-(LIA[I]*M+LIB[I]) ELSE
            UF[I]:=ONTBREKEND;
            FOR I:=0 STEP 1 UNTIL ATRIV DO
            BEGIN                                                       4
               REF:=KANWERK[I,*];
               RT3:=TRIPEL[I,*,*,*];
               RAT3:=ATRIP[I,*,*,*];
               RFL2:=FLUX[I,*,*];
               RAFL2:=AFLUX[I,*,*];
               FOR J:=0 STEP 1 WHILE J<6 DO
               BEGIN                                                    5
                  RT2:=RT3[J,*,*];
                  RAT2:=RAT3[J,*,*];
                  RFL1:=RFL2[J,*];
                  RAFL1:=RAFL2[J,*];
                  FOR K:=J STEP 1 WHILE K<6 DO
                  IF I1:=REF[J]>=0 AND I2:=REF[K]>=0 THEN
                  BEGIN                                                 6
                     IF UF[I1] NEQ ONTBREKEND AND
                     UF[I2] NEQ ONTBREKEND THEN
                     BEGIN                                              7
                        RAFL1[K]:=*+1;
                        RFL1[K]:=*+R:=UF[I1]*UF[I2];
                        RT1:=RT2[K,*];
                        RAT1:=RAT2[K,*];
                        FOR L:=K STEP 1 WHILE L<6 DO
                        IF I3:=REF[L]>=0 THEN
                        IF UF[I3] NEQ ONTBREKEND THEN
                        BEGIN                                           8
                           RT1[L]:=*+R*UF[I3];
                           RAT1[L]:=*+1;
                        END;                                            8 520000
                     END;                                               7 512000
                  END                                                   6 509000
               END                                                      5 502000
            END                                                         4 495000
         END;                                                           3 488000

         SEEK(OUT[0]);
         %P E R   T R I V A A N   2E   B E R E K E N I N G

   L7:
         FOR I:=0 STEP 1 UNTIL ATRIV DO
         BEGIN                                                          3
            REF:=KANWERK[I,*];
            RR[I,0,0]:=RR[I,1,1]:=COS(FI[I]);
            RR[I,1,0]:=-RR[I,0,1]:=SIN(FI[I]);
            FI[I]:=* * RAD;
            DD[I]:=* * RAD;
            RFL2:=FLUX[I,*,*];
            RAFL2:=AFLUX[I,*,*];
            FOR J:=0 STEP 1 WHILE J<6 DO
            FOR K:=0 STEP 1 WHILE K<6 DO
            IF K>=J THEN
            RFL2[J,K]:=IF REF[K]<0 OR REF[J]<0
            THEN ONTBREKEND ELSE RFL2[J,K]/RAFL2[J,K]
            ELSE RFL2[J,K]:=RFL2[K,J];

            % B E R E K E N I N G E N   G E D R A A I D   A S S E N S T E L S

   L8:
            RAT3:=ATRIP[I,*,*,*];
            RT3:=TRIPEL[I,*,*,*];
            FOR J:=0 STEP 1 WHILE J<6 DO
            BEGIN                                                       4
               RT2:=RT3[J,*,*];
               RAT2:=RAT3[J,*,*];
               RFLOK:=FLUXLOK[I,J,*];
               RFLUXT:=FLUXT[I,J,*];
```

```
561000              FOR K:=0 STEP 1 WHILE K<6 DO
562000              BEGIN                                                    5
563000                 RT1:=RT2[K,*];
564000                 RAT1:=RAT2[K,*];
565000                 FOR L:=0 STEP 1 WHILE L<6 DO
566000                 BEGIN                                                 6
567000                    RFL1:=RFL2[L,*];
568000                    IF L>=K AND K>=J THEN
569000                    RT1[L]:=IF REF[J]<0 OR REF[K]<0 OR REF[L]<0 THEN
570000                    ONTBREKEND ELSE
571000                    RT1[L]/RAT1[L];
572000  L9:
573000                    FOR M:=0 STEP 1 WHILE M<6 DO
574000                    IF RFL1[M] NEQ ONTBREKEND THEN
575000                    BEGIN                                              7
576000
577000                       % L O K A A L   A S S E N S T E L S E L
578000
579000                       RFLOK[K]:=+RFL1[M]*RR[I,J,L]*RR[I,K,M];
580000
581000                       % A S S E N S T E L S E L   G E D R A A I D   O V E R
582000
583000                       RFLUXT[K]:=+RFL1[M]*RR[0,J,L]*RR[0,K,M];
584000                    END                                                7 575000
585000                 END                                                   6 566000
586000              END;                                                     5 562000
587000           END;                                                        4 556000
588000
589000           % C O N T R O L E   A A N W E Z I G H E I D   K A N A L E N
590000
591000  L10:
592000           FOR J:=0 STEP 1 WHILE J<6 DO
593000           FOR K:=J STEP 1 UNTIL 5 DO
594000           IF REF[K]<0 THEN FLUXLOK[I,J,K]:=FLUXT[I,J,K]:=ONTBREKEND;
595000
596000           % A F G E L E I D E   G R O O T H E D E N
597000
598000           USTER[I]:=IF FLUXLOK[I,0,2]=ONTBREKEND THEN ONTBREKEND
599000           ELSE SQRT(ABS(FLUXLOK[I,0,2]));
600000           IF USTER[I] NEQ 0 THEN
601000           TSTER[I]:=IF FLUX[I,2,3]=ONTBREKEND OR USTER[I]=ONTBREKEND
602000           THEN ONTBREKEND ELSE
603000           -FLUXLOK[I,2,3]/USTER[I];
604000           IF USTER[I]=ONTBREKEND OR R:=REF[3]<0 OR
605000           TSTER[I]=ONTBREKEND THEN LL[I]:=ONTBREKEND
606000           ELSE
607000           BEGIN                                                      4
608000              LL[I]:=USTER[I]**2*(SOM[R]+273)/(.35*TSTER[I]*9.81);
609000              IF ABS(LL[I])>5000 THEN LL[I]:=5000*SIGN(LL[I]);
610000           END;                                                       4 607000
611000           FOR J:=0,1,2,3,4,5 DO
612000           SIGMA[I,J]:=IF FLUXLOK[I,J,J]=ONTBREKEND THEN
613000           ONTBREKEND ELSE SQRT(FLUXLOK[I,J,J]);
614000        END;                                                          3 535000
615000
616000        % P R I N T E R U I T V O E R
617000
618000  L11:
619000        FOR I:=0 STEP 1 UNTIL ATRIV DO
620000        BEGIN                                                         3
621000
622000           % PER INTERVAL INDIEN GEWENST ALLE TRIVANEN ARCHIVEREN
623000
624000           IF SAVERESULTS THEN
625000           BEGIN                                                      4
626000              REF:=KANWERK[I,*];
627000              RT2:=FLUXT[I,*,*];
628000              INDEX:=I*34;
629000              SAVEAR[0]:=CHECK0; %                BEGINTIJD
630000              SAVEAR[1]:=CHECK; %                 EINDTIJD
631000              SAVEAR[2]:=NRRECS; %                AANTAL SAMPLES
632000              SAVEAR[INDEX+3]:=IDENT[REF[0],3] DIV 10; % HOOGTE
633000              SAVEAR[INDEX+4]:=IDENT[REF[0],1]; %  UITHOUDER
634000              SAVEAR[INDEX+5]:=FF[I]; %           WINDSNELHEID
635000              SAVEAR[INDEX+6]:=DD[I]; %           WINDRICHTING
636000              SAVEAR[INDEX+7]:=          %       DROGE BOL TEMPERATUUR
637000              IF REF[3]<0 THEN ONTBREKEND ELSE SOM[REF[3]];
638000              SAVEAR[INDEX+8]:=          %       NATTE BOL TEMPERATUUR
639000              IF REF[4]<0 THEN ONTBREKEND ELSE SOM[REF[4]];
640000              SAVEAR[INDEX+9]:=          %       REL. VOCHTIGHEID
```

```
641000            IF REF[5]<0 THEN ONTBREKEND ELSE SOM[REF[5]];
642000            FOR J:=0,1+J WHILE J<6 DO%          STANDAARDOEVIATIES
643000            SAVEAR[INDEX+10+J]:=SIGMA[I,J]; %    U,V,W,TD,TN, Q
644000            J:=INDEX+15;
645000            FOR K:=0,1+K WHILE K<6 DO%          FLUXEN IN GRONDWIND
646000            FOR L:=K,1+L WHILE L<6 DO
647000            SAVEAR[J:=J+1]:=RT2[K,L]; %          STELSEL(21 WAARDEN)
648000            IF I=ATRIV THEN
649000            WRITE(SAVE,209,SAVEAR);
650000          END;                                                    4 625000
651000
652000          % I N I T I A L I S A T I E  P R I N T  A R R A Y
653000
654000  L13:
655000          REF:=KANWERK[I,*];
656000          RT3:=TRIPEL[I,*,*,*];
657000          RFL2:=FLUX[I,*,*];
658000          RAFL2:=FLUXLOK[I,*,*];
659000          RT2:=FLUXT[I,*,*];
660000          IF (R:=I MOD 3)=0 THEN
661000          BEGIN                                                   4
662000             FOR J:=0 STEP 1 UNTIL 59 DO
663000             REPLACE RULE[J,0] BY " " FOR 132;
664000             REPLACE RULE[0,0] BY "TIJD ",CHECKO FOR 9 DIGITS,
665000             "-",CHECK FOR 9 DIGITS,
666000             "  SAMPLES ",NRRECS FOR DIGITSIN(NRRECS) DIGITS;
667000          END;                                                    4 661000
668000          % R E G E L  2
669000          REPLACE RULE[2,R:=R+44] BY "HOOGTE (",
670000          IDENT[REF[0],3] DIV 10 FOR 3 DIGITS,
671000          ") UITHOUDER (",IDENT[REF[0],1] FOR 2 DIGITS,")";
672000          % R E G E L  3
673000          REPLACE RULE[3,R] BY "GEMIDDELDEN  TREND";
674000          % R E G E L  5 - 10
675000          REPLACE RULE[5,R] BY "FF  ",FORM(FF[I],2,2) FOR 6,
676000          "  ",FORM(TRFF1[I],2,2) FOR 6;
677000          REPLACE RULE[6,R] BY "AZ  ",FORM(DD[I],3,2) FOR 6,
678000          "  ",FORM(TRA1[I],3,2) FOR 6;
679000          U:=IF REF[2]<0 THEN ONTBREKEND ELSE SOM[REF[2]];
680000          REPLACE RULE[7,R] BY
681000          " W  ",FORM(U,2,2) FOR 6,"  ",FORM(DJ[I,2],2,2) FOR 6;
682000          % D R O G E  B O L
683000          U:=IF REF[3]<0 THEN ONTBREKEND ELSE SOM[REF[3]];
684000          REPLACE RULE[8,R] BY
685000          "TD  ",FORM(U,2,2) FOR 6,"  ",FORM(DJ[I,3],2,2) FOR 6;
686000          % N A T T E  B O L
687000          U:=IF REF[4]<0 THEN ONTBREKEND ELSE SOM[REF[4]];
688000          REPLACE RULE[9,R] BY
689000          "TN  ",FORM(U,2,2) FOR 6,"  ",FORM(DJ[I,4],2,2) FOR 6;
690000          % R E L A T I E V E  V O C H T I G H E I D
691000          U:=IF REF[5]<0 THEN ONTBREKEND ELSE SOM[REF[5]];
692000          REPLACE RULE[10,R] BY
693000          " Q  ",FORM(U,2,2) FOR 6,"  ",FORM(DJ[I,5],2,2) FOR 6;
694000          % S T O E V
695000          REPLACE RULE[13,R+7] BY
696000          "HOREV HORLR  VERT   TD    TN    Q";
697000          PTR1:=RULE[14,R];
698000          REPLACE  PTR1:PTR1 BY "STOEV  ";
699000          FOR J:=0 STEP 1 WHILE J<6 DO
700000          REPLACE PTR1:PTR1 BY FORM(SIGMA[I,J],2,2) FOR 6;
701000          % L  T*  U*
702000          REPLACE RULE[15,R] BY
703000          "L  ",CASE REAL(LL[I]<0) OF (" ","-").[7:48] FOR 1,
704000          LL[I] FOR 4 DIGITS,"   ",
705000          "U* ",FORM(USTER[I],2,2) FOR 6,"   ",
706000          "T* ",FORM(TSTER[I],2,2) FOR 6;
707000          % F L U X  I N  O N-S T E L S E L
708000          REPLACE RULE[18,R] BY "FLUXEN IN ON STELSEL";
709000          I1:=19;
710000          FOR J:=0 STEP 1 WHILE J<6 DO
711000          BEGIN                                                   4
712000             I1:=I1+1;
713000             FOR K:=J STEP 1 UNTIL 5 DO
714000             BEGIN                                                5
715000                PTR1:=RULE[I1,R+(K*7 +REAL(K>0))];
716000                REPLACE PTR1:PTR1 BY FORM(RFL2[J,K],1,3) FOR 6;
717000                IF K<5 THEN REPLACE PTR1:PTR1 BY " ";
718000             END                                                  5 714000
719000          END;                                                    4 711000
720000          % F L U X  L O K A A L  S T E L S E L
```

```
721000          REPLACE RULE[28,R] BY "FLUXEN IN LOKAAL STELSEL";
722000          I1:=30;
723000          FOR J:=0 STEP 1 WHILE J<6 DO
724000          BEGIN                                                          4
725000             I1:=I1+1;
726000             FOR K:=J STEP 1 UNTIL 5 DO
727000             BEGIN                                                       5
728000                PTR1:=RULE[I1,R+(K*7 +REAL(K>0))];
729000                REPLACE PTR1:PTR1 BY FORM(RAFL2[J,K],1,3) FOR 6;
730000                IF K<5 THEN REPLACE PTR1:PTR1 BY " ";
731000             END                                                         5 727000
732000          END;                                                           4 724000
733000          % F L U X  I N  G R O N D W I N D  S T E L S E L
734000          REPLACE RULE[37,R] BY "FLUXEN IN GRONDWIND STELSEL";
735000          I1:=39;
736000          FOR J:=0 STEP 1 WHILE J<6 DO
737000          BEGIN                                                          4
738000             I1:=I1+1;
739000             FOR K:=J STEP 1 UNTIL 5 DO
740000             BEGIN                                                       5
741000                PTR1:=RULE[I1,R+(K*7 +REAL(K>0))];
742000                REPLACE PTR1:PTR1 BY FORM(RT2[J,K],1,3) FOR 6;
743000                IF K<5 THEN REPLACE PTR1:PTR1 BY " ";
744000             END                                                         5 740000
745000          END;                                                           4 737000
746000          % T R I P E L  K O R R E L A T I E S
747000          REPLACE RULE[46,R] BY "TRIPEL KORRELATIES";
748000          PTR1:=RULE[48,R];
749000          REPLACE PTR1:PTR1 BY "111    ";
750000          IF RT3[0,0,0]=ONTBREKEND THEN
751000          REPLACE PTR1:PTR1 BY "+" FOR 9 ELSE
752000          REPLACE PTR1:PTR1 BY RT3[0,0,0] FOR 9 NUMERIC;
753000          REPLACE PTR1:PTR1 BY " 222    ";
754000          IF RT3[1,1,1]=ONTBREKEND THEN
755000          REPLACE PTR1:PTR1 BY "+" FOR 9 ELSE
756000          REPLACE PTR1:PTR1 BY RT3[1,1,1] FOR 9 NUMERIC;
757000          PTR1:=RULE[49,R];
758000          REPLACE PTR1:PTR1 BY "112    ";
759000          IF RT3[0,0,1]=ONTBREKEND THEN
760000          REPLACE PTR1:PTR1 BY "+" FOR 9 ELSE
761000          REPLACE PTR1:PTR1 BY RT3[0,0,1] FOR 9 NUMERIC;
762000          REPLACE PTR1:PTR1 BY " 223    ";
763000          IF RT3[1,1,2]=ONTBREKEND THEN
764000          REPLACE PTR1:PTR1 BY "+" FOR 9 ELSE
765000          REPLACE PTR1:PTR1 BY RT3[1,1,2] FOR 9 NUMERIC;
766000          PTR1:=RULE[50,R];
767000          REPLACE PTR1:PTR1 BY "113    ";
768000          IF RT3[0,0,2]=ONTBREKEND THEN
769000          REPLACE PTR1:PTR1 BY "+" FOR 9 ELSE
770000          REPLACE PTR1:PTR1 BY RT3[0,0,2] FOR 9 NUMERIC;
771000          REPLACE PTR1:PTR1 BY " 224    ";
772000          IF RT3[1,1,1]=ONTBREKEND THEN
773000          REPLACE PTR1:PTR1 BY "+" FOR 9 ELSE
774000          REPLACE PTR1:PTR1 BY RT3[1,1,3] FOR 9 NUMERIC;
775000          PTR1:=RULE[51,R];
776000          REPLACE PTR1:PTR1 BY "114    ";
777000          IF RT3[0,0,3]=ONTBREKEND THEN
778000          REPLACE PTR1:PTR1 BY "+" FOR 9 ELSE
779000          REPLACE PTR1:PTR1 BY RT3[0,0,3] FOR 9 NUMERIC;
780000          REPLACE PTR1:PTR1 BY " 233    ";
781000          IF RT3[1,2,2]=ONTBREKEND THEN
782000          REPLACE PTR1:PTR1 BY "+" FOR 9 ELSE
783000          REPLACE PTR1:PTR1 BY RT3[1,2,2] FOR 9 NUMERIC;
784000          PTR1:=RULE[52,R];
785000          REPLACE PTR1:PTR1 BY "122    ";
786000          IF RT3[0,1,1]=ONTBREKEND THEN
787000          REPLACE PTR1:PTR1 BY "+" FOR 9 ELSE
788000          REPLACE PTR1:PTR1 BY RT3[0,1,1] FOR 9 NUMERIC;
789000          REPLACE PTR1:PTR1 BY " 234    ";
790000          IF RT3[1,2,3]=ONTBREKEND THEN
791000          REPLACE PTR1:PTR1 BY "+" FOR 9 ELSE
792000          REPLACE PTR1:PTR1 BY RT3[1,2,3] FOR 9 NUMERIC;
793000          PTR1:=RULE[53,R];
794000          REPLACE PTR1:PTR1 BY "123    ";
795000          IF RT3[0,1,2]=ONTBREKEND THEN
796000          REPLACE PTR1:PTR1 BY "+" FOR 9 ELSE
797000          REPLACE PTR1:PTR1 BY RT3[0,1,2] FOR 9 NUMERIC;
798000          REPLACE PTR1:PTR1 BY " 244    ";
799000          IF RT3[1,3,3]=ONTBREKEND THEN
800000          REPLACE PTR1:PTR1 BY "+" FOR 9 ELSE
```

```
801000          REPLACE PTR1:PTR1 BY RT3[1,3,3] FOR 9 NUMERIC;
802000          PTR1:=RULE[54,R];
803000          REPLACE PTR1:PTR1 BY "124     ";
804000          IF RT3[0,1,3]=ONTBREKEND THEN
805000          REPLACE PTR1:PTR1 BY "+" FOR 9 ELSE
806000          REPLACE PTR1:PTR1 BY RT3[0,1,3] FOR 9 NUMERIC;
807000          REPLACE PTR1:PTR1 BY " 333     ";
808000          IF RT3[2,2,2]=ONTBREKEND THEN
809000          REPLACE PTR1:PTR1 BY "+" FOR 9 ELSE
810000          REPLACE PTR1:PTR1 BY RT3[2,2,2] FOR 9 NUMERIC;
811000          PTR1:=RULE[55,R];
812000          REPLACE PTR1:PTR1 BY "133     ";
813000          IF RT3[0,2,2]=ONTBREKEND THEN
814000          REPLACE PTR1:PTR1 BY "+" FOR 9 ELSE
815000          REPLACE PTR1:PTR1 BY RT3[0,2,2] FOR 9 NUMERIC;
816000          REPLACE PTR1:PTR1 BY " 334     ";
817000          IF RT3[2,2,3]=ONTBREKEND THEN
818000          REPLACE PTR1:PTR1 BY "+" FOR 9 ELSE
819000          REPLACE PTR1:PTR1 BY RT3[2,2,3] FOR 9 NUMERIC;
820000          PTR1:=RULE[56,R];
821000          REPLACE PTR1:PTR1 BY "134     ";
822000          IF RT3[0,2,3]=ONTBREKEND THEN
823000          REPLACE PTR1:PTR1 BY "+" FOR 9 ELSE
824000          REPLACE PTR1:PTR1 BY RT3[0,2,3] FOR 9 NUMERIC;
825000          REPLACE PTR1:PTR1 BY " 344     ";
826000          IF RT3[2,3,3]=ONTBREKEND THEN
827000          REPLACE PTR1:PTR1 BY "+" FOR 9 ELSE
828000          REPLACE PTR1:PTR1 BY RT3[2,3,3] FOR 9 NUMERIC;
829000          PTR1:=RULE[57,R];
830000          REPLACE PTR1:PTR1 BY "144     ";
831000          IF RT3[0,3,3]=ONTBREKEND THEN
832000          REPLACE PTR1:PTR1 BY "+" FOR 9 ELSE
833000          REPLACE PTR1:PTR1 BY RT3[0,3,3] FOR 9 NUMERIC;
834000          REPLACE PTR1:PTR1 BY " 444     ";
835000          IF RT3[3,3,3]=ONTBREKEND THEN
836000          REPLACE PTR1:PTR1 BY "+" FOR 9 ELSE
837000          REPLACE PTR1:PTR1 BY RT3[3,3,3] FOR 9 NUMERIC;
838000          PTR1:=RULE[58,R+16];
839000          REPLACE PTR1:PTR1 BY " 346     ";
840000          IF RT3[2,3,5]=ONTBREKEND THEN
841000          REPLACE PTR1:PTR1 BY "+" FOR 9 ELSE
842000          REPLACE PTR1:PTR1 BY RT3[2,3,5] FOR 9 NUMERIC;
843000          IF I MOD 3 = 2 OR I=ATRIV THEN
844000          FOR J:=0 STEP 1 UNTIL 59 DO
845000          WRITE(LINE,132,RULE[J,*]);
846000          WRITE(LINE[SKIP 1]);
847000       END;                                                    3 620000
848000    END                                                        2 352000
849000    UNTIL CHECK>=EINDPERIODE OR ENDOFFILE; %
850000    CLOSE (OUT,PURGE);
851000    IF SAVERESULTS THEN LOCK(SAVE,CRUNCH);
852000 END.                                                        1 5000
853000
854000          WRITE(LINE[SKIP 1]);
855000       END;                                                    3 628000
856000    END                                                        2 360000
857000    UNTIL CHECK>=EINDPERIODE OR ENDOFFILE; %
858000    CLOSE (OUT,PURGE);
859000    IF SAVERESULTS THEN LOCK(SAVE,CRUNCH);
860000 END.                                                        1 5000
861000
```

```
1000   $SET STATISTICS
2000   $SET LINEINFO
3000   $SET INSTALLATION
4000   BEGIN                                                                1
5000    $SET OMIT
6000      ##################################################################
7000      #                                                                #
8000      # DIT PROGRAMMA SELEKTEERT DE GEGEVENS VAN IEDER PERIODE UIT HET  #
9000      # TRIVAANBESTAND EN MAAKT NA BEWERKING(VOORKOKEN) VAN DE INGEVOERDE #
10000     # GEGEVENS EEN FILE (TIMESERIE) AAN WELKE ALS INVOERFILE WORDT GE- #
11000     # BRUIKT DOOR HET BINNEN DIT PROGRAMMA GESCHEDULDE PROGRAMMA COVSPEK#
13000     # VAN VERE. DE DOOR DIT PROGRAMMA AANGEMAAKTE FILES COVFILE EN    #
14000     # SPEKFILE WORDEN GEPRINT.                                        #
15000     #                                                                 #
16000     # PROGRAMMANAAM :     P R E P C O V          (CODE)              #
17000     #                     S Y M B O L / P R E P C O V (SYMBOLIC)      #
18000     # PROGRAMMEUR   :     P.NIEUWENDIJK                               #
19000     # DATUM         :     31 OKTOBER   1978                           #
20000     # CHARGECODE    :     MOB78267                                    #
21000     #                                                                 #
22000     ##################################################################
23000   $POP OMIT
24000   $PAGE
25000     % D E C L A R A T  I E S
26000
27000     FILE TAPE(
28000     KIND=PETAPE
29000     ,UNITS=WORDS
30000     ,BLOCKSIZE=2600
31000     ,MAXRECSIZE=26)
32000     ,OUT (
33000     KIND=DISK
34000     ,FLEXIBLE
35000     ,AREAS=600
36000     ,AREASIZE=30
37000     ,TITLE="WORKFILE.")
38000     ,TIMESERIE(
39000     KIND=DISK)
40000     ,CARD (
41000     KIND=READER
42000     ,UNITS=CHARACTERS)
43000     ,LINE (
44000     KIND=PRINTER
45000     ,UNITS=CHARACTERS);
46000
47000     BOOLEAN ARRAY AANWEZIG[0:51];
48000     BOOLEAN EOF;
49000     POINTER PRULE;
50000
51000     ARRAY          ARCHIEF[0:25]
52000     ,TOBE[0]=AANWEZIG
53000     ,IDENT[0:63,0:3]
54000     ,BEGINTYD,EINDTYD[0:5]
55000     ,COVSPEKNRS,KANAALNUMMERS[0:29]
56000     ,STORE[0:0],FH,FI[0:4]
57000     ,WAARDE,SOM,TEL[0:51];
58000
59000     EBCDIC ARRAY   RULE[0:599];
60000     TASK           COVSPEK;
61000
62000     INTEGER        I,DTG,TE,TO,K,J,AANTKAN,CHECK,CHECKO,IST,DIM,N,NN,%
63000     MEAN,IU,IV,IW,MM,HZ;
64000     REAL           PI,HPI,PI2,U,V,W,DELTA,INTEGRAL,PIHPI;
65000
66000
67000     DEFINE ONTBREKEND=99999#,
68000     DIGITSIN(X)=(FIRSTONE(SCALERIGHTF(X,12))-1) DIV 4 +1#,
69000     PAKUIT=FOR I:=0 STEP 1 UNTIL 51 DO
70000     IF AANWEZIG[I] THEN
71000     WAARDE[I]:=
72000     IF ARCHIEF[K:=I.[47:47]].[J:=(47-I.[0:1]*24):1]=1 THEN
73000     -ARCHIEF[K].[J-1:23] ELSE
74000     ARCHIEF[K].[J-1:23]#;
75000
76000     % P R O C E D U R E S
77000
78000   $INCLUDE "HEADING."
79000
80000     PROCEDURE COVARIANTIESPEKTRUM(DIM,MM,WINDOW,N,START);
81000     VALUE      DIM,MM,WINDOW,N,START;
```

```
82000     INTEGER   DIM,MM,WINDOW,N,START;
83000     EXTERNAL;
84000
85000
86000     REAL PROCEDURE RESTORE;
87000     BEGIN                                                          2
88000        OWN INTEGER I;
89000        RESTORE:=WAARDE[I];
90000        IF I:=I+1 > 51 THEN I:=0;
91000     END;                                                           2 87000
92000
93000
94000     % M A I N  P R O G R A M
95000
96000     % INITIALISERING 1
97000
98000     HEADING(LINE,"PREPCOV");
99000     NAME(LINE,"PRAMS");
100000    READ(CARD,<I9>,DTG);
101000    READ(CARD,<2I9,I2>,T0,TE,HZ);
102000    READ(CARD,<30I2>,KANAALNUMMERS);
103000
104000    HPI:=.5*PI:=4*ARCTAN(1);
105000    PI2:=2*PI;
106000    PIHPI:=PI+HPI;
107000    DELTA:=1/HZ;
108000
109000    %OPENEN INPUT MT FILE
110000
111000    REPLACE RULE BY "TRICAR",DTG FOR 9 DIGITS,".";
112000    REPLACE TAPE.TITLE BY RULE;
113000    TAPE.OPEN:=TRUE;
114000    WRITE(LINE,<"INPUT FILE TITLE ",A16," SERIALNO ",A6>,
115000    RULE,TAPE.SERIALNO);
116000
117000
118000    % I N V O E R  I D E N T I F I K A T I E  M A G N E E T B A N D
119000
120000    FOR I:=0 STEP 1 UNTIL 51 DO AANWEZIG[I]:=TRUE;
121000    READ(TAPE,26,ARCHIEF);
122000    PAKUIT;
123000    FOR I:=0 STEP 1 UNTIL 5 DO
124000    BEGINTYD[I]:=RESTORE;
125000    FOR I:=0 STEP 1 UNTIL 5 DO
126000    EINDTYD[I]:=RESTORE;
127000    FOR I:=0 STEP 1 UNTIL 9 DO
128000    BEGIN                                                          2
129000       J:=RESTORE-1;
130000       IDENT[J,0]:=J+1;
131000       IDENT[J,1]:=RESTORE;
132000       IDENT[J,2]:=RESTORE;
133000       IDENT[J,3]:=RESTORE;
134000    END;                                                           2 128000
135000    THRU 3 DO
136000    BEGIN                                                          2
137000       READ(TAPE,26,ARCHIEF);
138000       PAKUIT;
139000       FOR I:=0 STEP 1 UNTIL 12 DO
140000       BEGIN                                                       3
141000          J:=RESTORE-1;
142000          IF J>0 AND J<64 THEN
143000          BEGIN                                                    4
144000             IDENT[J,0]:=J+1;
145000             IDENT[J,1]:=RESTORE;
146000             IDENT[J,2]:=RESTORE;
147000             IDENT[J,3]:=RESTORE;
148000          END                                                      4 143000
149000       END                                                         3 140000
150000    END;                                                           2 136000
151000
152000    % P R I N T  M A G N E E T B A N D I D E N T I F I K A T I E
153000
154000    WRITE(LINE,<"I D E N T I F I K A T I E",/>);
155000    FOR I:=0 STEP 1 UNTIL 63 DO
156000    IF IDENT[I,0] NEQ 0 THEN
157000    WRITE(LINE,<I2" KAN "I2" UITHOUDER "I2" INST. "I2" PL "J5>,
158000    I+1,IDENT[I,0],IDENT[I,1],IDENT[I,2],IDENT[I,3]);
159000
160000    % PRINT INFO
161000
```

```
162000      FOR I:=0 STEP 1 UNTIL 48 DO AANWEZIG[I]:=FALSE;
163000      FOR I:=0 STEP 1 UNTIL 29 DO
164000      IF KANAALNUMMERS[I]>0 THEN
165000      BEGIN                                                          2
166000         AANTKAN:=*+1;
167000         AANWEZIG[KANAALNUMMERS[I]-1]:=TRUE;
168000      END;                                                           2 165000
169000
170000      J:=-1;
171000      WRITE(LINE,<//,"KANAALNUMMERS PER TRIVAAN",/,
172000      5(I2,6I3,/)>,
173000      FOR I:=0 STEP 6 UNTIL 29 DO
174000      (I DIV 6 + 1,THRU 6 DO KANAALNUMMERS[J:=*+1]));
175000      WRITE(LINE,<"TE VERWERKEN PERIODE ",I9," T/M ",I9>,
176000      TO,TE);
177000
178000      % CREEREN WORKFILE
179000
180000      REPLACE OUT.TITLE BY "WORKFILE.";
181000      OUT(AREAS=30,AREASIZE=AANTKAN*300,
182000      MAXRECSIZE=AANTKAN,BLOCKSIZE=AANTKAN*300);
183000      RESIZE(STORE,AANTKAN);
184000      FOR I:=0 STEP 1 UNTIL 29 DO
185000      KANAALNUMMERS[I]:=*-1;
186000
187000      % ZOEK STARTPUNT VERWERKING
188000
189000      DO
190000      BEGIN                                                          2
191000         READ(TAPE,26,ARCHIEF);
192000         PAKUIT;
193000         CHECK:=WAARDE[49]*10000+WAARDE[50] DIV 1000;
194000      END                                                            2 190000
195000      UNTIL CHECK>=TO;
196000      CHECK:=TO;
197000
198000      % VOORBEWERKING 1
199000
200000      DO
201000      BEGIN                                                          2
202000         IST:=-1;
203000         FOR I:=0 STEP 1 UNTIL 29 DO
204000         IF J:=KANAALNUMMERS[I]>=0 THEN
205000         BEGIN                                                       3
206000            IF WAARDE[J]=ONTBREKEND OR NOT AANWEZIG[J] THEN ELSE
207000            BEGIN                                                   4
208000               STORE[IST:=*+1]:=WAARDE[J]/100;
209000               TEL[IST]:=*+1;
210000               SOM[IST]:=*+WAARDE[J]/100;
211000            END                                                     4 207000
212000         END;                                                       3 205000
213000         WRITE(OUT,AANTKAN,STORE);
214000         EOF:=READ(TAPE,26,ARCHIEF);
215000         PAKUIT;
216000         CHECK:=WAARDE[49]*10000+WAARDE[50] DIV 1000;
217000      END                                                            2 201000
218000      UNTIL CHECK>=TE OR EOF;
219000
220000      % BEREKENINGEN 1
221000
222000      FOR I:=0,I+1 WHILE I<AANTKAN DO
223000      SOM[I]:=IF TEL[I]>0 THEN SOM[I]/TEL[I] ELSE ONTBREKEND;
224000
225000      FOR I:=0,I+6 WHILE I<AANTKAN DO
226000      IF TEL[I]>0 AND TEL[I+1]>0 THEN
227000      BEGIN                                                          2
228000         J:=I DIV 6;
229000         FH[J]:=SQRT(SOM[I]**2+SOM[I+1]**2);
230000         FI[J]:=ARCTAN(SOM[I+1]/SOM[I]);
231000         IF SOM[I+1]<0 THEN
232000         FI[J]:=IF SOM[I]<0 THEN FI[J]+PI ELSE FI[J]+PI2
233000         ELSE
234000         IF SOM[I]<0 THEN FI[J]:=*+PI;
235000         FI[J]:=* *180/PI;
236000      END                                                            2 227000
237000      ELSE FH[I DIV 6]:=FI[I DIV 6]:=ONTBREKEND;
238000
239000      % INVOER 3
240000
241000      IST:=-1;
```

```
242000      FOR I:=0 STEP 1 UNTIL 29 DO
243000      IF J:=KANAALNUMMERS[I]>=0 THEN
244000      TOBE[J]:=* &(IST:=*+1)  [10:8];
245000      I:=-1;
246000      READ(CARD,<I1,*I2,I2>,
247000      DIM,DIM,THRU DIM DO COVSPEKNRS[I:=*+1],MEAN);
248000      CLOSE(CARD);
249000
250000      NN:=ENTIER((OUT.RECORD+1)/MEAN);
251000      MM:=ENTIER(LN(.5*NN)/LN(2));
252000
253000      % CREEER TIMESERIE
254000
255000      REPLACE RULE BY " " FOR 600;
256000      REPLACE PRULE:RULE BY
257000      "TIMESERIE/",TO FOR 9 DIGITS,"/",
258000      TE FOR 9 DIGITS,"/";
259000      FOR I:=0,I+1 WHILE I<DIM DO
260000      REPLACE PRULE:PRULE BY COVSPEKNRS[I] FOR 2 DIGITS;
261000      REPLACE PRULE BY "/",MEAN FOR 2 DIGITS,".";
262000
263000      REPLACE TIMESERIE.TITLE BY RULE;
264000      TIMESERIE(
265000      MAXRECSIZE=DIM
266000      ,BLOCKSIZE=DIM*2**MM
267000      ,AREAS=NN DIV 2**MM + 1
268000      ,AREASIZE=DIM*2**MM);
269000
270000      % PRINT INFO
271000
272000      I:=-1;
273000      WRITE(LINE,<"KANAALNUMMERS VOOR BEREKENING "
274000      ,"COVARIANTIE EN SPECTRA :",*I3>,
275000      DIM,THRU DIM DO COVSPEKNRS[I:=*+1]);
276000      WRITE(LINE,<"DIM                    :",I6,/,
277000      "#SAMPLES REDUCTIE     :",I6,/,
278000      "SAMPLE FREQUENTIE(HZ):",I6,/,
279000      "LENGTE REEKS          :",I6,/,
280000      "MAXIMUM LAG           :",I6,/,
281000      "FH PER TRIVAAN        :",5F8.3,/,
282000      "FI PER TRIVAAN        :",5F8.3,/
283000      >,
284000      DIM,MEAN,HZ,NN,2**MM,FH,FI);
285000      SEEK(OUT[0]);
286000      WRITE(LINE,<"TITLE TIMESERIE       :",A60>,RULE);
287000
288000      % INITIALISERING 2
289000
290000      LOCK(TAPE);
291000      FOR I:=0  STEP 1 UNTIL 63 DO DEALLOCATE(IDENT[I,*]);
292000      DEALLOCATE(BEGINTYD);
293000      DEALLOCATE(EINDTYD);
294000      DEALLOCATE(TEL);
295000      RESIZE(ARCHIEF,DIM);
296000      RESIZE(WAARDE,AANTKAN);
297000
298000      % VOORBEWERKING 2
299000
300000      FOR I:=0,I+1 WHILE I<NN DO
301000      BEGIN                                                    2
302000         REPLACE WAARDE[*] BY 0 FOR AANTKAN WORDS;
303000         FOR J:=0,J+1 WHILE J<MEAN DO
304000         BEGIN                                                 3
305000            READ(OUT,AANTKAN,STORE);
306000            FOR K:=0 STEP 6 UNTIL 29 DO
307000            BEGIN                                             4
308000               IU:=KANAALNUMMERS[K];
309000               IV:=KANAALNUMMERS[K+1];
310000               IW:=K DIV 6;
311000               IF IU<0 OR IV<0 THEN ELSE
312000               BEGIN                                          5
313000                  IU:=TOBE[IU].[10:8];
314000                  IV:=TOBE[IV].[10:8];
315000                  U:=(SOM[IU]*STORE[IU]+SOM[IV]*STORE[IV])/
316000                  FH[IW]-FH[IW];
317000                  V:=(-SOM[IV]*STORE[IU]+STORE[IV]*SOM[IU])/
318000                  FH[IW];
319000                  WAARDE[IU]:=*+U;
320000                  WAARDE[IV]:=*+V;
321000               END;                                          5 312000
```

```
322000              FOR IW:=KANAALNUMMERS[K+2]
323000              ,KANAALNUMMERS[K+3]
324000              ,KANAALNUMMERS[K+4]
325000              ,KANAALNUMMERS[K+5]
326000              DO
327000              IF IW>=0 THEN
328000              BEGIN                                                    5
329000                 IW:=TOBE[IW].[10:8];
330000                 W:=STORE[IW]-SOM[IW];
331000                 WAARDE[IW]:=*+W;
332000              END                                                      5 328000
333000           END                                                         4 307000
334000        END;                                                           3 304000
335000        FOR K:=0,1+K WHILE K<AANTKAN DO
336000        WAARDE[K]:=*/MEAN;
337000        FOR K:=0,1+K WHILE K<DIM DO
338000        ARCHIEF[K]:=WAARDE[TOBE[COVSPEKNRS[K]-1].[10:8]];
339000        WRITE(TIMESERIE,DIM,ARCHIEF);
340000     END;                                                              2 301000
341000
342000     % INITIALISERING 3
343000
344000     LOCK(TIMESERIE,CRUNCH);
345000     CLOSE(OUT,PURGE);
346000     DEALLOCATE(WAARDE);
347000     DEALLOCATE(SOM);
348000     DEALLOCATE(KANAALNUMMERS);
349000
350000     % SCHEDULING COVSPEK
351000
352000     % INTER PROGRAM COMMUNICATION COVSPEK
353000
354000     REPLACE COVSPEK.NAME BY "COVSPEK.";
355000     REPLACE PRULE:RULE BY
356000     "FILE TIMESERIE(KIND=DISK,TITLE=TIMESERIE/",
357000     TO FOR 9 DIGITS,"/",TE FOR 9 DIGITS,"/";
358000     FOR I:=0,I+1 WHILE I<DIM DO
359000     REPLACE PRULE:PRULE BY COVSPEKNRS[I] FOR 2 DIGITS;
360000     REPLACE PRULE:PRULE BY
361000     "/",MEAN FOR 2 DIGITS,");FILE COVFILE(TITLE=COVFILE/",
362000     RULE[41] FOR 23+DIM*2,");FILE SPEKFILE(TITLE=SPEKFILE/",
363000     RULE[41] FOR 23+DIM*2,");",48"00";
364000
365000     REPLACE COVSPEK.FILECARDS BY RULE;
366000     CALL COVARIANTIESPEKTRUM(DIM,MM,1,NN,0)[COVSPEK];
367000
368000     %PRINT COVFILE
369000
370000     K:=2**MM+1; RESIZE(ARCHIEF,K);
371000     REPLACE RULE[0] BY "COVFILE/",RULE[41]FOR 23+DIM*2,".";
372000     REPLACE OUT.TITLE BY RULE;
373000     OUT(BLOCKSIZE=K,MAXRECSIZE=K,MYUSE=IN);
374000     OUT.OPEN:=TRUE;
375000     %
376000     % PRINT COVFILE LAGS
377000     %
378000     WRITE(LINE,<"COVFILE LAGS"/,
379000     *(10F10.3,/)>,
380000     K DIV 10+1,
381000     FOR I:=0,I+1 WHILE I<K DO I*DELTA*MEAN);
382000
383000     WHILE NOT READ(OUT,K,ARCHIEF) DO
384000     BEGIN                                                            2
385000        WRITE(LINE,<"COVFILE BLOCK ",I3>,OUT.BLOCK+1);
386000        WRITE(LINE,<*(10F10.3,/)>,K DIV 10+1,ARCHIEF);
387000     END;                                                             2 384000
388000
389000     %PRINT SPEKFILE
390000
391000     CLOSE(OUT);
392000     REPLACE RULE BY "SPEKFILE/",RULE[41] FOR 23+DIM*2,".";
393000     REPLACE OUT.TITLE BY RULE;
394000     OUT(BLOCKSIZE=K,MAXRECSIZE=K,MYUSE=IN);
395000     OUT.OPEN:=TRUE;
396000
397000     %
398000     % PRINTEN FREQUENTIE SPEKFILE
399000     %
400000     WRITE(LINE,<"SPEKFILE FREQUENTIES"/,
401000     *(10F10.3,/)>,
```

```
402000      K DIV 10+1,
403000      FOR I:=0,1+I WHILE I<K DO
404000      I/((K-1)*2*MEAN*DELTA));
405000      WHILE NOT READ(OUT,K,ARCHIEF) DO
406000      BEGIN                                                          2
407000         INTEGRAL:=0;
408000         FOR I:=1,I+1 WHILE I<K DO INTEGRAL:=*+ARCHIEF[I];
409000         INTEGRAL:=(0.5*ARCHIEF[0]+INTEGRAL)/(2**MM);
410000         WRITE(LINE,<"SPEKFILE BLOCK ",I3," INTEGRAL ",F10.3>
411000         ,OUT.BLOCK+1,INTEGRAL);
412000         WRITE(LINE,<*(10F10.3,/)>,K DIV 10+1,ARCHIEF);
413000      END;                                                           2 406000
414000
415000   END.                                                             1 4000
416000
```

```
 1000   $ SET LIST LINEINFO
 2000   $ LEVEL 2
 3000  PROCEDURE COVSPEK(DIM,MM,WINDOW,N,NSTART);
 4000  VALUE DIM,MM,WINDOW,N,NSTART; INTEGER DIM,MM,WINDOW,N,NSTART;
 5000  BEGIN                                                                  1
 6000     INTEGER P,M,NW,NBLOK,KK,OPTIES;
 7000     FILE TIMESERIE(KIND=DISK,MAXRECSIZE=DIM,BLOCKSIZE=(2**MM)*DIM);
 8000     FILE COVFILE(KIND=DISK,MAXRECSIZE=2**MM+1,AREASIZE=DIM**2);
 9000     FILE SPEKFILE(KIND=DISK,MAXRECSIZE=2**MM+1,AREASIZE=DIM**2);
10000     FILE LP(KIND=PRINTER,FILETYPE=3);
11000     PROCEDURE FFT4(A,B,N,M,KS);
12000     VALUE N,M,KS; INTEGER N,M,KS; ARRAY A,B[0];
13000     BEGIN                                                               2
14000        INTEGER K0,K1,K2,K3,K,SPAN;
15000        REAL A0,A1,A2,A3,B0,B1,B2,B3,RE,IM,RAD,DC,DS,C1,C2,C3,S1,S2,S3;
16000        LABEL LA,LB;
17000        SPAN:=KS; KS:=2**M; RAD:= 4.0*ARCTAN(1.0)/KS;
18000        KS:=SPAN DIV KS; N:=N-1; K:=M;
19000        FOR M:=M-2 WHILE M GEQ 0 DO
20000        BEGIN                                                            3
21000           C1:=1.0; S1:=0; K0:=0; K:=KS; DC:=2.0*SIN(RAD)**2;
22000           RAD:=RAD+RAD; DS:=SIN(RAD);
23000           RAD:=RAD+RAD; SPAN:=SPAN DIV 4;
24000  LA:
25000           K1:=K0+SPAN; K2:=K1+SPAN; K3:=K2+SPAN;
26000           A0:=A[K0]; B0:=B[K0];
27000           A1:=A[K1]; B1:=B[K1];
28000           A2:=A[K2]; B2:=B[K2];
29000           A3:=A[K3]; B3:=B[K3];
30000           A[K0]:=A0+A1+A2+A3;
31000           B[K0]:=B0+B1+B2+B3;
32000           IF S1=0 THEN
33000           BEGIN                                                        4
34000              A[K1]:=A0+A2-A1-A3; B[K1]:=B0+B2-B1-B3;
35000              A[K2]:=A0-A2-B1+B3; B[K2]:=B0-B2+A1-A3;
36000              A[K3]:=A0-A2+B1-B3; B[K3]:=B0-B2-A1+A3
37000           END                                                          4 33000
38000           ELSE
39000           BEGIN                                                        4
40000              RE:=A0+A2-A1-A3; IM:=B0+B2-B1-B3;
41000              A[K1]:=RE*C2-IM*S2; B[K1]:=RE*S2+IM*C2;
42000              RE:=A0-A2-B1+B3; IM:=B0-B2+A1-A3;
43000              A[K2]:=RE*C1-IM*S1; B[K2]:=RE*S1+IM*C1;
44000              RE:=A0-A2+B1-B3; IM:=B0-B2-A1+A3;
45000              A[K3]:=RE*C3-IM*S3; B[K3]:=RE*S3+IM*C3
46000           END;                                                         4 39000
47000           K0:=K3+SPAN; IF K0<N THEN GO TO LA;
48000           K0:=K0-N; IF K0 NEQ K THEN GO TO LA;
49000           IF K0 NEQ SPAN THEN
50000           BEGIN                                                        4
51000              C2:=C1-(DC*C1+DS*S1);
52000              S1:=(DS*C1-DC*S1)+S1; C1:=C2;
53000              C2:=C1**2-S1**2; S2:=2.0*C1*S1;
54000              C3:=C2*C1-S2*S1; S3:=C2*S1+S2*C1;
55000              K:=K+KS; GO TO LA;
56000           END;                                                         4 50000
57000           K:=M
58000        END;                                                            3 20000
59000        IF K NEQ 0 THEN
60000        BEGIN                                                           3
61000           SPAN:=SPAN DIV 2; K0:=0;
62000  LB:
63000           K2:=K0+SPAN; A0:=A[K2]; B0:=B[K2];
64000           A[K2]:=A[K0]-A0; A[K0]:=A[K0]+A0;
65000           B[K2]:=B[K0]-B0; B[K0]:=B[K0]+B0;
66000           K0:=K2+SPAN; IF K0<N THEN GO TO LB;
67000           K0:=K0-N; IF K0 NEQ SPAN THEN GO TO LB
68000        END                                                             3 60000
69000     END                                                                2 13000
70000     FFT4;
71000     PROCEDURE REVFFT4(A,B,N,M,KS);
72000     VALUE N,M,KS; INTEGER N,M,KS; ARRAY A,B[0];
73000     BEGIN                                                              2
74000        INTEGER K0,K1,K2,K3,K,SPAN;
75000        REAL A0,A1,A2,A3,B0,B1,B2,B3,RAD,DC,DS,C1,C2,C3,S1,S2,S3;
76000        LABEL LA,LB;
77000        RAD:=4.0*ARCTAN(1.0); N:=N-1; K0:=0; SPAN:=KS;
78000        IF M MOD 2 = 1 THEN
79000        BEGIN                                                           3
80000  LA:
```

```
81000           K2:=K0+SPAN; A0:=A[K2]; B0:=B[K2];
82000           A[K2]:=A[K0]-A0; A[K0]:=A[K0]+A0;
83000           B[K2]:=B[K0]-B0; B[K0]:=B[K0]+B0;
84000           K0:=K2+SPAN; IF K0<N THEN GO TO LA;
85000           K0:=K0-N; IF K0 NEQ SPAN THEN GO TO LA;
86000           SPAN:=SPAN+SPAN; RAD:=0.5*RAD
87000         END;                                                    3 79000
88000         FOR M:=M-2 WHILE M GEQ 0 DO
89000         BEGIN                                                   3
90000           C1:=1.0; S1:=0; K0:=0; RAD:=0.25*RAD;
91000           DC:=2.0*SIN(RAD)**2; DS:=SIN(RAD+RAD); K:=KS;
92000   LB:
93000           K1:=K0+SPAN; K2:=K1+SPAN; K3:=K2+SPAN;
94000           A0:=A[K0]; B0:=B[K0];
95000           IF S1=0 THEN
96000           BEGIN                                                 4
97000             A2:=A[K1]; B2:=B[K1];
98000             A3:=A[K3]; B3:=B[K3];
99000             A1:=A[K2]; B1:=B[K2]
100000          END                                                   4 96000
101000          ELSE
102000          BEGIN                                                 4
103000            A2:=A[K1]*C2-B[K1]*S2; B2:=A[K1]*S2+B[K1]*C2;
104000            A1:=A[K2]*C1-B[K2]*S1; B1:=A[K2]*S1+B[K2]*C1;
105000            A3:=A[K3]*C3-B[K3]*S3; B3:=A[K3]*S3+B[K3]*C3
106000          END;                                                  4 102000
107000          A[K0]:=A0+A2+A1+A3; B[K0]:=B0+B2+B1+B3;
108000          A[K1]:=A0-A2-B1+B3; B[K1]:=B0-B2+A1-A3;
109000          A[K2]:=A0+A2-A1-A3; B[K2]:=B0+B2-B1-B3;
110000          A[K3]:=A0-A2+B1-B3; B[K3]:=B0-B2-A1+A3;
111000          K0:=K3+SPAN; IF K0<N THEN GO TO LB;
112000          K0:=K0-N; IF K0 NEQ K THEN GO TO LB;
113000          IF K0 NEQ SPAN THEN
114000          BEGIN                                                 4
115000            C2:=C1-(DC*C1+DS*S1);
116000            S1:=(DS*C1-DC*S1)+S1; C1:=C2;
117000            C2:=C1**2-S1**2; S2:=2.0*C1*S1;
118000            C3:=C2*C1-S2*S1; S3:=C2*S1+S2*C1;
119000            K:=K+KS; GO TO LB
120000          END;                                                  4 114000
121000          SPAN:=4*SPAN
122000        END                                                     3 89000
123000      END                                                       2 73000
124000      REVFFT4;
125000      PROCEDURE UNSCRAMBEL (R,I,N,NF,EVALUATE);
126000      VALUE N,NF,EVALUATE; INTEGER N,NF; BOOLEAN EVALUATE; ARRAY R,I[0];
127000      BEGIN                                                     2
128000        INTEGER NH,J,K1,K2;
129000        REAL  AA,AB,BA,BB,RE,IM,CK,SK,DC,DS;
130000        LABEL L;
131000        NH:=N DIV 2; DS:=2.0*NF*ARCTAN(1.0)/N; CK:=1.0;
132000        DC:=2.0*SIN(DS)**2; DS:=SIN(DS+DS); SK:=0;
133000        IF EVALUATE THEN
134000        BEGIN                                                   3
135000          CK:=-CK; DS:=-DS
136000        END                                                     3 134000
137000        ELSE
138000        BEGIN                                                   3
139000          FOR J:=0,J+1 WHILE J<NF DO
140000          BEGIN                                                 4
141000            R[N+J]:=R[J]; I[N+J]:=I[J]
142000          END                                                   4 140000
143000        END;                                                    3 138000
144000        K1:=J:=0; K2:=N;
145000  L:
146000        AA:=R[K1]+R[K2]; AB:=R[K1]-R[K2];
147000        BA:=I[K1]+I[K2]; BB:=I[K1]-I[K2];
148000        RE:=CK*BA+SK*AB; IM:=SK*BA-CK*AB;
149000        I[K2]:=IM-BB; I[K1]:=IM+BB;
150000        R[K2]:=AA-RE; R[K1]:=AA+RE;
151000        K2:=K2+1; K1:=K1+1; J:=J+1;
152000        IF J<NF THEN GO TO L; J:=0;
153000        AA:=CK-(DC*CK+DS*SK);
154000        SK:=(DS*CK-DC*SK)+SK;
155000        CK:=AA; K2:=N-K1;
156000        IF K1 LEQ NH THEN GO TO L
157000      END                                                       2 127000
158000      UNSCRAMBEL;
159000      INTEGER I,J,K,I1,I2,K1,KD0,KD1,KD2,KD3,LL,MH,MD,D2,MD2,D4,ND,
160000      BI,F1,F2,NN,ST,LIM,M1D,MD21;
```

```
161000      REAL R10,R11,R12,R13,R20,R21,R22,R23,QE,QO,FI,
162000      I10,I11,I12,I13,I20,I21,I22,I23,FE,FO,FAK,T;
163000      INTEGER ARRAY BITREV[0:2**MM-1];
164000      BOOLEAN BOL,FIRST,BEZIG,IND,EVEN;
165000      ARRAY A,B,C,D[0:(2**MM+1)*DIM-1],RR,II[0:(2**MM+1)*DIM*DIM-1];
166000      LABEL S;
167000      DEFINE SL(R1,R2,I1,I2)=
168000      RR[J]:=R1*R2+I1*I2+RR[J];
169000      II[J]:=R1*I2-I1*R2+II[J];
170000      #;
171000      PROCEDURE OPSLAG(A,B,C,D); ARRAY A,B,C,D[0];
172000      BEGIN                                                          2
173000         LABEL L0,L1,L2,L3,E0,E1,E2,E3,V0,V1,V2,V3,00,01,02,03;
174000         SWITCH L:=L0,L1,L2,L3; SWITCH E:=E0,E1,E2,E3;
175000         SWITCH V:=V0,V1,V2,V3; SWITCH O:=00,01,02,03;
176000         EVEN:=FALSE;
177000         FOR K:=0,K+1 WHILE K LEQ M DO
178000         BEGIN                                                       3
179000            KD0:=K*DIM; K1:=KD0*DIM; EVEN:=NOT EVEN;
180000            GO TO L[DIM];
181000            L3:KD3:=KD0+3; R13:=A[KD3]; I13:=B[KD3];
182000            L2:KD2:=KD0+2; R12:=A[KD2]; I12:=B[KD2];
183000            L1:KD1:=KD0+1; R11:=A[KD1]; I11:=B[KD1];
184000      L0:
185000            R10:=A[KD0]; I10:=B[KD0];
186000            IF EVEN THEN GO TO E[DIM] ELSE GO TO O[DIM];
187000            E3:R23:=C[KD3]+R13; I23:=D[KD3]+I13;
188000            E2:R22:=C[KD2]+R12; I22:=D[KD2]+I12;
189000            E1:R21:=C[KD1]+R11; I21:=D[KD1]+I11;
190000            E0:R20:=C[KD0]+R10; I20:=D[KD0]+I10; GO TO V[DIM];
191000            03:R23:=R13-C[KD3]; I23:=I13-D[KD3];
192000            02:R22:=R12-C[KD2]; I22:=I12-D[KD2];
193000            01:R21:=R11-C[KD1]; I21:=I11-D[KD1];
194000            00:R20:=R10-C[KD0]; I20:=I10-D[KD0]; GO TO V[DIM];
195000            V3:J:=K1+3;  SL(R10,R23,I10,I23);
196000            J:=J+DIM; SL(R11,R23,I11,I23);
197000            J:=J+DIM; SL(R12,R23,I12,I23);
198000            J:=J+DIM; SL(R13,R23,I13,I23);
199000            J:=J-1;   SL(R13,R22,I13,I22);
200000            J:=J-1;   SL(R13,R21,I13,I21);
201000            J:=J-1;   SL(R13,R20,I13,I20);
202000            V2:J:=K1+2;  SL(R10,R22,I10,I22);
203000            J:=J+DIM; SL(R11,R22,I11,I22);
204000            J:=J+DIM; SL(R12,R22,I12,I22);
205000            J:=J-1;   SL(R12,R21,I12,I21);
206000            J:=J-1;   SL(R12,R20,I12,I20);
207000            V1:J:=K1+1;  SL(R10,R21,I10,I21);
208000            J:=J+DIM; SL(R11,R21,I11,I21);
209000            J:=J-1;   SL(R11,R20,I11,I20);
210000            V0:J:=K1;    SL(R10,R20,I10,I20);
211000         END                                                        3 178000
212000      END                                                           2 172000
213000      OPSLAG;
214000      PROCEDURE VERWERK(A,B,C,D); ARRAY A,B,C,D[0];
215000      BEGIN                                                          2
216000         LABEL EOF;
217000         REPLACE POINTER(A) BY 0 FOR M1D WORDS;
218000         REPLACE POINTER(B) BY 0 FOR M1D WORDS;
219000         IF KK<NBLOK THEN
220000         FOR I:=0,I+1 WHILE I<MH DO
221000         BEGIN                                                       3
222000            J:=BITREV[I]*DIM;
223000            READ(TIMESERIE,DIM,A[J]);
224000            NW:=*+1; IF NW EQL N THEN GO TO EOF;
225000            READ(TIMESERIE,DIM,B[J]);
226000            NW:=*+1; IF NW EQL N THEN GO TO EOF;
227000         END;                                                       3 221000
228000      EOF:
229000         REVFFT4(A,B,MD,MM,DIM);
230000         UNSCRAMBEL(A,B,MD,DIM,FALSE);
231000         IF KK NEQ 0 THEN OPSLAG(C,D,A,B);
232000      END                                                           2 215000
233000      VERWERK;
234000      M:=2**MM; OPTIES:=MYSELF.TASKVALUE;
235000      IF NSTART NEQ 0 THEN READ(TIMESERIE[NSTART-1],DIM,A[0]);
236000      MH:=M DIV 2; MD:=M*DIM; D2:=DIM*DIM; MD2:=M*D2; D4:=D2*2;
237000      MD21:=MD2+D2;
238000      M1D:=(M+1)*DIM; NBLOK:=((N-1) DIV M)+1; NW:=0;
239000      FOR K:=0,K+1 WHILE K<M DO
240000      BEGIN                                                          2
```

```
241000          T:=0;
242000          FOR NN:=0,NN+1 WHILE NN<MM DO
243000          T.[NN:1]:=K.[MM-NN-1:1];
244000          BITREV[K]:=T;
245000       END;                                                    2 240000
246000       FOR KK:=0 STEP 1 UNTIL NBLOK DO
247000       IF KK MOD 2 = 0 THEN VERWERK(A,B,C,D)
248000       ELSE VERWERK(C,D,A,B);
249000       UNSCRAMBEL(RR,II,MD2,D2,TRUE);
250000       FOR I:=0,I+1 WHILE I<MD21 DO II[I]:=-II[I];
251000       FFT4(RR,II,MD2,MM,MD2);
252000       FAK:=1/(8*M*N);
253000       FOR I:=0,I+1 WHILE I<MD21 DO
254000       BEGIN                                                   2
255000          RR[I]:= RR[I]*FAK;
256000          II[I]:=-II[I]*FAK
257000       END;                                                    2 254000
258000       FOR I:=0,I+1 WHILE I<D2 DO
259000       WRITE(COVFILE,*,FOR J:=0,J+1 WHILE J<MM DO
260000       [RR[K:=(BITREV[J]*D2+I)],II[K]],RR[D2+I]);
261000       LOCK(COVFILE);
262000       FOR I:=0,I+1 WHILE I<D2 DO RR[I]:=RR[I]*0.5;
263000       FOR I:=D2,I+D4 WHILE I<MD2 DO
264000       FOR J:=0,J+1 WHILE J<D2 DO
265000       RR[I+J]:=II[I+J]:=0;
266000       FI:=ARCTAN(1.0)/MM;
267000       FOR I:=0,I+2 WHILE I<M DO
268000       BEGIN                                                   2
269000          I1:=I*D2; BI:=BITREV[I]*2;
270000          CASE WINDOW OF
271000          BEGIN                                                3
272000          0:
273000             FE:=FO:=1;
274000          1:
275000             FE:=COS(BI*FI)**2; FO:=COS((BI+1)*FI)**2;
276000          2:
277000             FE:=1-BI/M; FO:=1-(BI+1)/M;
278000          3:
279000             QE:=BI/M; QO:=(BI+1)/M;
280000             FE:=IF BI<MM THEN (QE-1)*QE*QE*6+1 ELSE 2*(1-QE)**3;
281000             FO:=IF BI<MM THEN (QO-1)*QO*QO*6+1 ELSE 2*(1-QO)**3
282000          END;                                                 3 271000
283000          THRU D2 DO
284000          BEGIN                                                3
285000             RR[I1]:=RR[I1]*FE;
286000             II[I1]:=II[I1]*FO;
287000             I1:=I1+1
288000          END                                                  3 284000
289000       END;                                                    2 268000
290000       REVFFT4(RR,II,MD2,MM,D2);
291000       UNSCRAMBEL(RR,II,MD2,D2,FALSE);
292000       FOR I:=0,I+D2 WHILE I LEQ MD2 DO
293000       FOR F1:=0,F1+1 WHILE F1<DIM DO
294000       FOR F2:=F1,F2+1 WHILE F2<DIM DO
295000       BEGIN                                                   2
296000          I1:=I+F1*DIM+F2;
297000          I2:=I+F2*DIM+F1;
298000          RR[I1]:=(RR[I1]+RR[I2])/2;
299000          IF I1 NEQ I2 THEN
300000          RR[I2]:=(II[I1]-II[I2])/2
301000       END;                                                    2 295000
302000       FOR I:=0,I+1 WHILE I<D2 DO
303000       WRITE(SPEKFILE,*,FOR J:=I,J+D2 WHILE J<MD21 DO RR[J]);
304000       LOCK(SPEKFILE);
305000    END.                                                       1 5000
306000
```