INTERNAL REPORTS IN

SIMULATION, OPTIMIZATION AND CONTROL

No. SOC-302

DDATA - A FORTRAN PACKAGE OF DATA HANDLING ROUTINES

J.W. Bandler and W.M. Zuberek

September 1982

FACULTY OF ENGINEERING
McMASTER UNIVERSITY
HAMILTON, ONTARIO, CANADA

DDATA - A FORTRAN PACKAGE OF DATA HANDLING ROUTINES

J.W. Bandler and W.M. Zuberek

Abstract

DDATA is a package of subroutines for accessing data with different physical representations. The package performs all the required data conversions and communicates with the user's programs on the level of binary information. The description of a particular representation of the data must be supplied in the form of data descriptors that constitute the initial section of the data file. The format of data descriptors is fixed. Some simple elements of data preprocessing (e.g., scaling, default values) are included in the package. The package and documentation have been developed for the CDC 170/730 system with the NOS 1.4 operating system and the Fortran 4.8508 compiler. Examples are given for 23-bus and 26-bus test power systems, illustrating how data can be read, printed and stored in a descripted data file created after solving a load flow problem.

This work was supported by the Natural Sciences and Engineering Research Council of Canada under Grant G0647.

J.W. Bandler and W.M. Zuberek are with the Group on Simulation, Optimization and Control, and the Department of Electrical and Computer Engineering, McMaster University, Hamilton, Canada L8S 4L7.

W.M. Zuberek is on leave from the Institute of Computer Science, Technical University of Warsaw, Warsaw, Poland.

I. INTRODUCTION

One of the common requirements of "application programs" [1] is to accept as large a variety of input data as possible. This requirement is, however, conflicting with another common requirement of portability [2], which results in using one of the popular high-level programming languages (in scientific applications it is usually Fortran or its subset) in which most of the input/output is strictly formatted. The development of flexible facilities for handling input data with different physical representations can be quite complex and rather artificial with respect to the standard constructs of the selected programming language. A solution which offers a reasonable compromise between the requirements is to separate the input and/or output operations from the user's programs and to provide a set of standardized data handling functions in the form of a library of routines linked with the user's programs by appropriate calling sequences [3].

There are several advantages of such a solution. Flexibility of the user's programs is one of them, potential efficiency (if the package is implemented in a low level language) is another one, but extensibility which allows one to add new data handling facilities and new functions without modification of the existing programs may appear to be the most important feature when the class of data representations is not known in advance and/or may change from time to time.

The DDATA package has been developed as a set of Fortran IV subroutines for the CDC 170/730 system. It has been assumed that the main interface with the user's (or application) programs is on the level of binary information, i.e., that all the data conversions from the

physical, external representation to the internal, binary form are performed within the package. The package provides operations on data files, data records and data items. In the current implementation only sequential files can be handled. The physical representation of the data can be quite arbitrary, and its specification (for the DDATA package) is required in the fixed form of data descriptors which constitute the initial section of the data file, and are "invisible" to the user's programs.

At McMaster University the package is available in the form of a library of compiled subroutines in the group indirect file LIBSPWR. The general sequence of NOS commands to use the package may be as follows:

/GET (LIBSPWR/GR) - fetch the library,

/LIBRARY (LIBSPWR) - indicate the library to the loader,

/FTN (..., GO) - compile, load and execute the program.

The package can be used to convert data to any previously established format as required by existing programs as well as for direct accessing of original data. Several data files for power systems analysis and optimization have been created in the described form, and some higher level application-oriented packages have been developed to simplify the interaction of the user's programs with the basic functions of the package.

II. GENERAL INFORMATION

The DDATA package is a set of subroutines that form the software interface between the user's programs and the physical data files providing uniform access to the data. There are 3 levels of operations

(or functions) performed by the package:

- data file operations,
- data record operations,
- data item operations.

Data items are the smallest, indivisible units of information such as numbers or character strings. It is assumed that the data items (or, more precisely, values or instances of data items) are transferred between the package and the user's programs in binary form. There are 4 types of data items in the current version of the package:

- integer numbers,
- real numbers,
- complex numbers,
- character strings.

All the data items which are handled by the DDATA package are uniquely identified by their names. The data names are fixed for a particular application (e.g., analysis and optimization of power systems) and are specified within data descriptors.

Data items are grouped into logical data records, i.e., into conceptual sequences of information units which are required by the user's program (e.g., in power systems a description of a bus may contain bus identification which usually is an integer number, bus type which determines whether the bus is load, generator or slack, bus voltage, bus injected power, etc.). Logical data records are defined by the user's programs in the form of sequences of data names corresponding to consecutive data items of the logical data record. Logical data records may significantly differ from the physical data records, i.e., the records that represent the information in the (physical) data file.

It is the function of the DDATA package to map the physical data records into the logical ones, as defined by the user's program. Consequently, the user's programs need not be concerned with all the details of physical representation as well as ordering of data items provided the relevant information is available in the data file. Moreover, the same user's program can use the data files with different representations of data, which can be very useful and quite important when an interchange of information between different programs or systems is required.

All the logical data records which are handled by the DDATA package are uniquely identified by user-defined indexes, and there is a bound on the number of simulatenously handled logical records. In the current version of the package the bound is equal to 10. The indexes of the logical data records can be, however, redefined, and the same index may identify different data structures in disjoint periods of time.

The logical data records have a static structure, i.e., the ordering of data items that compose the logical record is fixed and cannot depend on the values of data. There is, however, a very simple mechanism that allows us to redefine (or switch) the record structure if dynamic, data-depending structuring is required.

Data records are grouped into data files. The DDATA package can process sequential files only, however, each data file can contain a number of subfiles with different data records, and the only requirement is that the (physical) data records must be uniquely assigned to the subfiles corresponding to the record's structure. Therefore, each physical data record contains a header with corresponding subfile identification (in the current implementation the record header contains 3 characters). Each subfile is terminated by the end-of-subfile record

that contains the header only.

Each data file that is to be processed by the DDATA package must contain the data descriptor subfile. The format of this subfile is fixed, and the contents of the subfile must precisely describe the physical representation of the data in the remaining subfiles of the data file. Data descriptors are copied from the data file to a set of internal tables of the DDATA package and control all the processing of data performed by the package.

III. DATA DESCRIPTORS

Data descriptors are required as the initial subfile of the descripted data file in the following sequence:

- data header record,
- data descriptor records,
- end-of-descriptor-subfile record.

The format of all these records is fixed and is composed of the fields shown in Table I.

For the data header record the fields parameter 1, parameter 2 and parameter 3 are application-dependent and may be defined in any convenient way. In power systems data files they are used to represent the number of buses, the index of the slack bus, and the number of transmission lines of the system, respectively. The information code is an integer value which is provided to describe the ordering of data subfiles within the data file, and can be used in different ways for different applications. In power systems data files it is interpreted as a string of decimal digits

TABLE I

FORMAT OF RECORDS

Character Position	Data Header Record	Data Descriptor Records
1	character 0	character 0
2	character 0	subfile marker
3	character 0	character 0
4-8	parameter 1	data type
9-13	parameter 2	data position
14-18	parameter 3	data length
20-29	internal name of the data file	data name
30-34	not used	processing code
35-44	information code	multiplier
45-54	not used	default value

$$\ldots d_4\ d_3\ d_2\ d_1\ d_0$$

where d_0 is equal to the number of data subfiles in the data file, d_1 corresponds to the bus data subfile, d_2 to the transmission line subfile, and d_3 to the fuel cost data subfile. Moreover, if $d_i > d_j$, $i, j = 1,2,\ldots$, the d_i subfile follows the d_j one, if $d_i < d_j$, the d_i subfile precedes the d_j one, and otherwise the records of the two subfiles are unordered.

For the data descriptor records the subfile marker is the character that identifies the corresponding data subfile, i.e., it must be equal to the first character in the header of those data records, in which the data item is located. The subfile marker must be different from 0, that

is reserved for the data descriptor subfile, and from *, which is reserved for end-of-subfile records as well as for "comment" records, which are ignored by the package.

The data type is an integer value describing the physical representation of data. In the current version of the package the following data types are implemented:

1 - formatted integer data (as for Fortran I specification),

2 - formatted real data (as for Fortran F specification),

3 - formatted complex data (as for Fortran F specification),

4 - formatted real data (as for Fortran E specification),

5 - formatted complex data (as for Fortran E specification),

7 - binary real data (in the form of character strings),

8 - binary complex data (in the form of character strings),

9 - character strings.

The data position is an integer value, which determines the position of the first character of the corresponding data item in the physical data records, and the data length specifies the number of characters provided for the physical representation of the data.

The data name is the unique name of the data item. It must be left justified and must not be longer than 10 characters. The data names used in the described power systems data files are given in Appendix 1.

The processing code is not used in the current version of the DDATA package because the existing preprocessing facilities are very simple. It can be used, in some future extensions, to indicate the required initial transformations of the data.

The value of the multiplier is used to scale original data values, and the default value replaces the data value when the corresponding

data field is blank.

The two examples of (physical) descripted data files shown on pp. 10-13 correspond to the test 23-bus power system [4,5] (internal file name DATA-23) and the 26-bus power system [5-7] (internal file name DATA-26). It should be observed that the physical representations of data are similar but differ in several details.

```
000   23   23   30  DATA-23                    122
***  1111 2222 3333 ********** 4444 555555555 666666666
010    1    4    4  BUSNR
010    9   11   10  BUSNAME
010    1    9    1  BUSTYPE
010    2   20    5  BUSBASEV          1.00       1.00
010    2   25    5  BUSVMOD           1.00       1.00
010    2   30    5  BUSVARG           1.00       0.00
010    2   36    6  BUSGP             1.00       0.00
010    2   42    6  BUSGQ             1.00       0.00
010    2   48    6  BUSLP             1.00       0.00
010    2   54    6  BUSLQ             1.00       0.00
010    2   70    6  BUSSTL            1.00       0.00
010    2   60    5  BUSGQMAX          1.00       5.00
010    2   65    5  BUSGQMIN          1.00      -5.00
010    3   36    6  BUSGPQ            1.00       0.00
010    3   48    6  BUSLPQ            1.00       0.00
020    3   14    6  LINEINPA          1.00       0.00
020    3   26    6  LINERX            1.00       1.00
020    3   38    6  LINEOUTA          1.00       0.00
020    1    4    4  LINEBINP
020    1    8    4  LINEBOUT
020    1   12    1  LINECNR
020    2   14    6  LINEINPC          1.00       0.00
020    2   20    6  LINEINPS          1.00       0.00
020    2   26    6  LINER             1.00       1.00
020    2   32    6  LINEX             1.00       1.00
020    2   38    6  LINEOUTC          1.00       0.00
020    2   44    6  LINEOUTS          1.00       0.00
020    2   71    5  LINERAT           1.00       1.00
020    2   76    5  LINEBASE          1.00       1.00
020    1   50    1  LINETRT
020    2   51    5  LINETAP           1.00       1.00
020    2   56    5  LINETMAX          1.00       1.10
020    2   61    5  LINETMIN          1.00       0.90
020    2   66    5  LINETSH           1.00       0.00
0*0
200   23   11       .005900.02420.0540       .00590
200   23   21       .007550.03090.0693       .00755
200   18   31       .009850.04040.0888       .00985
200    6   31       .007850.03250.0709       .00785
200   18   51       .017100.06150.1620       .01710
200    1   41       .016000.05760.1520       .01600
200    2   71       .007400.02660.0700       .00740
200    7   51       .005600.02290.0504       .00560
200    6   41       .010900.04460.1003       .01090
200   19   31       .022800.02330.0514       .02280
200    6   31       .014550.05970.1315       .01455
200    7   31       .014550.05970.1315       .01455
200   10  201       .113650.00430.0351       .11365
200   20   91       .113650.00430.0351       .11365
200   11   91       .103900.00380.0307       .10390
200   14  111       .097550.00350.0228       .09755
200   22  101       .243550.00890.0726       .24355
200   12  131       .027150.00100.0080       .02715
200   13  141       .056650.00210.0167       .05665
200   15  141       .043100.00160.0127       .04310
200   21  151       .122550.00450.0362       .12255
200   17  141       .064900.00240.0192       .06490
200   21  161       .052290.00190.0156       .05229
200   16  171       .033500.00140.0114       .03350
200   22  121       .055450.00200.0164       .05545
200    9   61       .0     0.00230.0839       .0     11.04
200   10   61       .0     0.00230.0839       .0     11.03
200    9   71       .0     0.00190.1300       .0     11.05
```

```
200  10   71        .0     0.00230.0839       .0     11.06
200  23  181        .0     0.00250.2000       .0
2*0
100 001 0 BUS-001         1.00 0.00   0.000 0.000 0.000 0.000 0.00 0.00
100 002 0 BUS-002         1.00 0.00   0.000 0.000 0.470 0.120 0.00 0.00
100 003 0 BUS-003         1.00 0.00   0.000 0.000 0.510 0.130 0.00 0.00
100 004 0 BUS-004         1.00 0.00   0.000 0.000 0.410 0.100 0.00 0.00
100 005 0 BUS-005         1.00 0.00   0.000 0.000 0.480 -0.12 0.00 0.00
100 006 0 BUS-006         1.00 0.00   0.000 0.000 0.010 0.000 0.00 0.00
100 007 0 BUS-007         1.00 0.00   0.000 0.000 1.500 0.380 0.00 0.00
100 008 0 BUS-008         1.00 0.00   0.000 0.000 1.770 0.440 0.00 0.00
100 009 0 BUS-009         1.00 0.00   0.000 0.000 0.060 0.000 0.00 0.00
100 010 0 BUS-010         1.00 0.00   0.000 0.000 -0.04 0.000 0.00 0.00
100 011 0 BUS-011         1.00 0.00   0.000 0.000 2.010 0.500 0.00 0.00
100 012 0 BUS-012         1.00 0.00   0.000 0.000 1.320 0.330 0.00 0.00
100 013 0 BUS-013         1.00 0.00   0.000 0.000 3.440 0.860 0.00 0.00
100 014 0 BUS-014         1.00 0.00   0.000 0.000 1.040 0.260 0.00 0.00
100 015 0 BUS-015         1.00 0.00   0.000 0.000 3.760 0.940 0.00 0.00
100 016 0 BUS-016         1.00 0.00   0.000 0.000 3.750 0.940 0.00 0.00
100 017 0 BUS-017         1.00 0.00   0.000 0.000 -2.10 -0.52 0.00 0.00
100 018 1 BUS-018         1.03 0.00   0.260 0.000
100 019 1 BUS-019         1.05 0.00   0.890 0.000
100 020 1 BUS-020         1.05 0.00   0.200 0.000
100 021 1 BUS-021         1.05 0.00   9.030 0.000
100 022 1 BUS-022         1.05 0.00   9.230 0.000
100 023 2 BUS-023         1.04 0.00
1*0
```

```
000   26   26   32  DATA-26                        122
*** 1111 2222 3333  ********** 4444 555555555 666666666
010    1    4    4  BUSNR
010    9   11   10  BUSNAME
010    1    9    1  BUSTYPE
010    2   20    5  BUSBASEV          1.00        1.00
010    2   25    5  BUSVMOD           1.00        1.00
010    2   30    5  BUSVARG           1.00        0.00
010    2   36    6  BUSGP             1.00        0.00
010    2   42    6  BUSGQ             1.00        0.00
010    2   48    6  BUSLP             1.00        0.00
010    2   54    6  BUSLQ             1.00        0.00
010    2   60    6  BUSGQMAX          1.00        5.00
010    2   66    6  BUSGQMIN          1.00       -5.00
010    2   72    6  BUSSTL            1.00        0.00
010    3   36    6  BUSGPQ            1.00        0.00
010    3   48    6  BUSLPQ            1.00        0.00
020    3   14    6  LINEINPA          1.00        0.00
020    3   26    6  LINERX            1.00        1.00
020    3   38    6  LINEOUTA          1.00        0.00
020    1    4    4  LINEBINP
020    1    8    4  LINEBOUT
020    1   12    1  LINECNR
020    2   14    6  LINEINPC          1.00        0.00
020    2   20    6  LINEINPS          1.00        0.00
020    2   26    6  LINER             1.00        1.00
020    2   32    6  LINEX             1.00        1.00
020    2   38    6  LINEOUTC          1.00        0.00
020    2   44    6  LINEOUTS          1.00        0.00
020    2   50    5  LINERAT           1.00        1.00
020    2   55    5  LINEBASE          1.00        1.00
020    1   60    1  LINETRT
020    2   61    5  LINETAP           1.00        1.00
020    2   66    5  LINETMAX          1.00        1.10
020    2   71    5  LINETMIN          1.00        0.90
020    2   76    5  LINETSH           1.00        0.00
0*0
200   13  261       0.00000.000 0.0131       0.0000          11.03
200   26  161       0.00000.000 0.0392       0.0000          10.96
200   16  231       0.00000.000 0.4320       0.0000
200   23  261       0.00000.000 0.3140       0.0000
200    2  101       0.00000.000 0.0150       0.0000          11.03
200    9  101       0.41200.14940.3392       0.4120
200    9  121       0.01320.06520.1494       0.0132
200   12  261       0.01470.05330.1210       0.0147
200    9  141       0.03190.06180.2397       0.0319
200   11  141       0.03490.06760.2620       0.0349
200   19  261       0.02950.06100.2521       0.0295
200    6  251       0.02650.05130.1986       0.0265
200    6  191       0.00740.01290.0532       0.0074
200    7  191       0.04370.09060.3742       0.0437
200    6   71       0.04750.09210.3569       0.0475
200   11  221       0.02480.05130.2118       0.0248
200    8  111       0.04470.08650.3355       0.0447
200   17  221       0.02370.02810.1069       0.0237
200    8  211       0.03790.07350.2847       0.0379
200   17  211       0.03370.04590.3955       0.0337
200    1   41       0.03190.06190.2401       0.0319
200    4  211       0.03150.06100.2365       0.0315
200   20  211       0.00000.000 0.0205       0.0000          10.97
200   15   11       0.00000.000 0.0147       0.0000          10.89
200    2  131       0.39170.00360.0797       0.3917
200    1   71       0.04040.01990.0785       0.0404
200   15  201       0.44710.01070.0617       0.4471
200    2  181       0.25930.00740.0608       0.2593
```

```
200    1   31        0.00000.000 0.0392       0.0000           10.98
200   24   31        0.00000.000 0.1450       0.0000           10.98
200    5  211        0.00000.000 0.1750       0.0000           10.99
200    5  251        0.00000.000 0.1540       0.0000           11.03
2*0
100 001 0 BUS 001        1.00 0.00  0.000 0.000 0.82  0.21  0.000 0.000
100 002 0 BUS 002        1.00 0.00  0.000 0.000 0.00  0.00  0.000 0.000
100 003 0 BUS 003        1.00 0.00  0.000 0.000 0.57  0.17  0.000 0.000
100 004 0 BUS 004        1.00 0.00  0.000 0.000 0.48  0.21  0.000 0.000
100 005 0 BUS 005        1.00 0.00  0.000 0.000 0.43  0.11  0.000 0.000
100 006 0 BUS 006        1.00 0.00  0.000 0.000 0.40  0.10  0.000 0.000
100 007 0 BUS 007        1.00 0.00  0.000 0.000 1.11  0.27  0.000 0.000
100 008 0 BUS 008        1.00 0.00  0.000 0.000 0.23  0.06  0.000 0.000
100 009 0 BUS 009        1.00 0.00  0.000 0.000 0.67  0.21  0.000 0.000
100 010 0 BUS 010        1.00 0.00  0.000 0.000 1.02  0.27  0.000 0.000
100 011 0 BUS 011        1.00 0.00  0.000 0.000 0.43  0.14  0.000 0.000
100 012 0 BUS 012        1.00 0.00  0.000 0.000 0.43  0.12  0.000 0.000
100 013 0 BUS 013        1.00 0.00  0.000 0.000 0.00  0.00  0.000 0.000
100 014 0 BUS 014        1.00 0.00  0.000 0.000 0.00  0.00  0.000 0.000
100 015 0 BUS 015        1.00 0.00  0.000 0.000 0.00  0.00  0.000 0.000
100 016 0 BUS 016        1.00 0.00  0.000 0.000 1.31  0.30  0.000 0.000
100 017 0 BUS 017        1.00 0.00  0.000 0.000 0.03  0.01  0.000 0.000
100 018 1 BUS 018        1.07 0.00  2.800 1.070
100 019 1 BUS 019        1.05 0.00  1.450 1.050
100 020 1 BUS 020        1.00 0.00  2.200 1.000
100 021 1 BUS 021        1.02 0.00  1.100 1.020
100 022 1 BUS 022        0.89 0.00 -0.56  0.890
100 023 1 BUS 023        1.00 0.00 -0.04  1.000
100 024 1 BUS 024        1.00 0.00 -0.05  1.000
100 025 1 BUS 025        1.00 0.00  0.630 1.000
100 026 2 BUS 026        1.01 0.00
1*0
```

IV. DATA OPERATIONS

All the data operations are performed as a result of appropriate subroutine calls. The following operations are available in the current version of the DDATA package (for each operation the name of the corresponding subroutine, an indication whether the operation is used for reading R, writing W, or reading and writing RW the data, and a brief description are given).

The data file operations:

- DD00DF/R - define the input data file; in effect the data header record is transferred from the data file and the parameters describing the file are returned to the calling program.
- DD00WF/W - define the output data file and write data header record; in effect the data header record with the parameters supplied by the DD00WF call is sent to the output file.
- DD00DN/R - define all the data names which are to be used when accessing the file defined by DD00DF (DD00DN call must follow the DD00DF call); in effect all the data descriptor records are transferred to the package, and those which match the data names indicated in the DD00DN call, are stored in the internal tables of the package.
- DD00DD/W - define data descriptors and write data descriptor records (DD00DD call must follow the DD00WF call); in effect the data descriptors submitted to the package as the arguments of the DD00DD call are entered into internal tables of the package and then are sent to the output file defined by the DD00WF call.
- DD00ES/W - close the subfile; in effect the end-of-subfile record

is sent to the output file.

- DD00RF/RW - reset the file; in effect the input or output file is positioned at the beginning of the information.

The data record operations:

- DD11DR/RW - define the logical record (the DD11DR call must follow the DD00DN or DD00DD call or another DD11DR call); in effect the data descriptors corresponding to the data names indicated by the DD11DR arguments are linked to represent the defined record structure (all these data names must be defined by the preceeding DD00DN or DD00DD operation).
- DD11ER/RW - erase the logical record definition; in effect the previously defined logical record structure is removed from the internal package tables.
- DD11GS/R - transfer subsequent physical record from the data file; in effect the physical record is stored in the internal buffer in the package, and consecutive elements of the defined logical records can be accessed by the corresponding data item operations.
- DD11SW/R - switch the logical record definition; the operation may be used when data-dependent structuring is required, and it results in substitution of the current logical record definition by the new one indicated.
- DD22SR/W - initialize the new output record; in effect the record header is created accordingly to the indicated logical data record, and consecutive elements of the record can be set up in the buffer by the corresponding data item operations.
- DD22PR/W - transfer a physical record from the internal buffer to the output file; in effect the record created by a sequence of

preceding data item operations is sent to the output file defined earlier for the record.

Data item operations:

- DD11IN/R - get an integer value;
- DD11RN/R - get a real value;
- DD11CN/R - get a complex value;
- DD11ST/R - get a character string.

All these operations must be performed after a DD11GS call, and their order must correspond to the definition of the logical record indicated in the DD11GS call. The result of each of these operations is a data item value that is returned to the calling program, after conversions performed by the package.

- DD22IN/W - place an integer value;
- DD22RN/W - place a real value;
- DD22CN/W - place a complex value;
- DD22ST/W - place a character string.

All these operations must be performed after a DD22SR call, and their order must correspond to the definition of the logical record indicated in the DD22SR call. Each of these operations results in setting up in the internal package buffer the converted representation of the submitted data value.

There are several restrictions imposed on the ordering of data operations:

(1) The sequence of DD00DF, DD00DN and all the DD11DR calls referring to the file defined by DD00DF must not be interposed by any other data operation.

(2) The sequence of DD00WF, DD00DD and all the DD11DR calls referring

to the file defined by DD00WF must not be interposed by any other data operation.

(3) The sequence of DD11GS and corresponding DD11IN, DD11RN, DD11CN and DD11ST calls must not be interposed by any other data operation except of DD11SW.

(4) The sequence of DD22SR, corresponding DD22IN, DD22RN, DD22CN, DD22ST, and DD22PR calls must not be interposed by any other data operation.

Therefore, it is recommended to implement the required sequences of data operations in the form of separate subroutines that can be used as more powerful, higher-level data operations. Examples 1 and 2 in Section VI indicate typical organization of such subroutines.

V. DESCRIPTION OF SUBROUTINES

Subroutine DD00DD

This subroutine defines the data descriptors for the output data file defined by the preceding DD00WF call. All the data descriptors are stored in internal tables of the package (to be used in subsequent definitions of logical data records by appropriate DD11DR calls), and then are written to the output file in the form of a data descriptor subfile.

The subroutine call is

CALL DD00DD (TN,KA,LT,RT,LR,IFLAG)

and the arguments are as follows.

TN is a REAL array containing the data names. Data names must be left justified and stored in the form of character strings, one

name in one element of the array.

KA is an INTEGER array containing data attributes. Each element of KA, as a string of decimal digits, has the structure

abccddeeff

where

a is a subfile index that indicates all the data elements grouped within the same data subfile,

b is the data type (see Data Descriptors),

cc is the index (in the TN array) of another data element in the case of data redefinitions, e.g., if the same data element is described as a REAL value and as an element of a COMPLEX value, one of these definitions must refer (as a "redefinition") to the "original" (or first) data definition,

dd is the data length, i.e., the number of characters in the external (physical) representation of data values,

ee if not equal to 00 it is the index (in the auxiliary array RT) of the REAL multiplier used in scaling of data values,

ff if not equal to 00 it is the index (in the auxiliary array RT) of the default value.

LT is an INTEGER argument that must be set to the number of elements in the TN and KA arrays.

RT is a REAL array that contains the values of multipliers and default values required by the data attributes.

LR is an INTEGER argument that must be set to the length of the array RT.

IFLAG is an INTEGER variable that is used as a return flag:

-4 incorrect data attributes,

-3 insufficient workspace of the package (e.g., too many data names are defined),

-2 incorrect value of argument LT (e.g., LT $\leq$ 0),

-1 incorrect use of the subroutine,

0 normal return.

Subroutine DD00DF

This subroutine defines the input data file, reads the data header record and returns the parameters describing the file. The subroutine call is

CALL DD00DF (NF,N1,N2,N3,DN,IFLAG)

and the arguments are as follows.

NF is an INTEGER argument that must be set to the unit (or channel) number of the data file. It must be positive and less than 100.

N1 is an INTEGER variable that returns the first parameter of the data header record (for power systems data it is the number of buses).

N2 is an INTEGER variable that returns the second parameter of the data header record (for power systems data it is the index of the slack bus).

N3 is an INTEGER variable that returns the third parameter of the data header record (for power systems data it is the number of transmission lines).

DN is a REAL variable that returns the internal name of the data file.

IFLAG is an INTEGER variable that is used as a return flag:

-2 incorrect structure of the data file (e.g., empty file),

-1 incorrect value of NF argument (e.g., NF $\leq$ 0),

$\geq$0 normal return; IFLAG is equal to the information code that describes the ordering of data subfiles within the data file.

Subroutine DD00DN

This subroutine defines all the data names which are to be used when accessing the data file defined by corresponding DD00DF call. The subroutine must be called after the DD00DF call and before any other data handling subroutine.

The subroutine call is

CALL DD00DN (TN,LT,IFLAG)

and the arguments are as follows.

TN is a REAL array containing the data names. Data names must be left justified and stored in the form of character strings, one name in one element of the array.

LT is an INTEGER argument that must be set to the number of data names in array TN.

IFLAG is an INTEGER variable that is used as a return flag:

-4 incorrect structure of the data file (e.g., incorrect data descriptors),

-3 insufficient workspace of the package (e.g., too many data names are defined),

-2 incorrect value of LT argument (e.g., LT $\leq$ 0),

-1 incorrect use of the subroutine,

0 normal return,

>0 some data names are incorrect (the value of IFLAG is equal to the number of incorrect data names). All the data names which do not correspond to the valid data names (i.e., the names in data descriptor records) are replaced in array TN by the string of * characters.

Subroutine DD00ES

This subroutine writes the end-of-subfile record that closes the data subfile corresponding to the logical data record indicated in the subroutine call.

The subroutine call is

CALL DD00ES (NR,IFLAG)

and the arguments are as follows.

NR is an INTEGER argument that must be set to the identifier of a logical data record.

IFLAG is an INTEGER variable that is used as a return flag:

-3 argument NR corresponds to an undefined logical record,

-2 incorrect value of NR argument (e.g., NR $\leq$ 0),

-1 incorrect use of the subroutine,

$\geq$0 normal return; the value of IFLAG is equal to the number of records in the data subfile.

Subroutine DD00RF

This subroutine resets the file indicated in the subroutine call, i.e., it positions the file at the beginning of information, and initializes some indicators within the package.

The subroutine call is

CALL DD00RF (NF,IFLAG)

and the arguments are as follows.

NF is an INTEGER argument that must be set to the unit (or channel) number of the file.

IFLAG is an INTEGER variable that is used as a return flag:

-1 incorrect value of argument NF (e.g., NF $\leq$ 0),

0 normal return.

Subroutine DD00WF

This subroutine defines the output data file and writes the data header record.

The subroutine call is

CALL DD00WF (NF,N1,N2,N3,DN,IC,IFLAG)

and the arguments are as follows.

NF is an INTEGER argument that must be set to the unit (or channel) number of the output file. It must be positive and less than 100.

N1 is an INTEGER argument that must be set to the first parameter of the data header record.

N2 is an INTEGER argument that must be set to the second parameter of the data header record.

N3 is an INTEGER argument that must be set to the third parameter of the data header record.

DN is a REAL argument that must be set to the internal name of the data file.

IC is an INTEGER argument that must be set to the value describing the ordering of data subfiles within the file.

IFLAG is an INTEGER variable that is used as a return flag:

-1 incorrect value of NF argument (e.g., NF $\leq$ 0),

0 normal return.

Subroutine DD11CN

This subroutine retrieves a complex value from the physical record, performs data conversions, and returns the value in the binary form.

The subroutine call is

CALL DD11CN (Z,IFLAG)

and the arguments are as follows.

Z is a COMPLEX variable that returns the data value.

IFLAG is an INTEGER variable that is used as a return flag:

-2 expected data type is different than COMPLEX,

-1 incorrect use of the subroutine,

0 normal return,

1 default value assigned to argument Z.

Subroutine DD11DR

This subroutine defines the logical data record as a sequence of data names and assigns an integer identifier to it. The subroutine must be called after corresponding DD00DN call, and must use only the data names which have been defined by DD00DN call. Several logical data records may be defined for the same data file by a series of DD11DR calls with appropriate arguments.

The subroutine call is

CALL DD11DR (NR,TN,LT,IFLAG)

and the arguments are as follows.

NR is an INTEGER argument that must be set to the unique identifier of the record; it must be positive and not greater than the limit of simultaneously defined logical records.

TN is a REAL array containing the data names composing the logical data record. Data names must be left justified and stored in the form of character strings, one name in one element of the array.

LT is an INTEGER argument that must be set to the number of data names in array TN.

IFLAG is an INTEGER variable that is used as a return flag:

- -6 some data names are incorrect, i.e., they have not been defined by preceding DD00DN call; all the incorrect data names are replaced in array TN by the string of * characters,
- -5 the identifier NR is not unique and has already been defined,
- -4 incorrect value of NR argument (e.g., NR $\leq$ 0),
- -3 insufficient workspace of the package (e.g., too many data names are defined),
- -2 incorrect value of LT argument (e.g., LT $\leq$ 0),
- -1 incorrect use of the subroutine,
- 0 normal return.

Subroutine DD11ER

This subroutine erases the definition of a logical data record from the internal tables of the package and allows a new logical record to be defined (by the subroutine DD11DR) with the same logical record

identifier.

The subroutine call is

CALL DD11ER (NR,IFLAG)

and the arguments are as follows.

NR is an INTEGER argument that must be set to the identifier of a logical data record.

IFLAG is an INTEGER variable that is used as a return flag:

-2 incorrect argument NR (e.g., NR $\leq$ 0),

-1 incorrect use of the subroutine,

0 normal return.

Subroutine DD11GS

This subroutine retrieves from the input data file the consecutive data record that corresponds to the indicated logical data record.

The subroutine call is

CALL DD11GS (NR,IFLAG)

and the arguments are as follows.

NR is an INTEGER argument that must be set to the idenifier of a logical data record.

IFLAG is an INTEGER variable that is used as a return flag:

-3 incorrect structure of the data file (e.g., end of file encountered),

-2 incorrect value of NR argument (e.g., NR $\leq$ 0),

-1 incorrect use of the subroutine (e.g., NR corresponds to undefined logical data record),

0 end-of-subfile encountered which indicates that there are no more data records corresponding to the indicated logical

data record,

>0 normal return; the value of IFLAG is equal to the consecutive number of the data record.

The elements of the retrieved data record (for the normal return) can be accessed by appropriate calls of subroutines

DD11IN - for INTEGER values,

DD11RN - for REAL values,

DD11CN - for COMPLEX values,

DD11ST - for character strings.

The order of accessing elements of the record must exactly correspond to ordering of data names in the definition (subroutine DD11DR) of the logical data record indicated by argument NR.

Subroutine DD11IN

This subroutine retrieves an integer value from the physical record, performs data conversions, and returns the value in binary form.

The subroutine call is

CALL DD11IN (N,IFLAG)

and the arguments are as follows.

N is an INTEGER variable that returns the data value.

IFLAG is an INTEGER variable that is used as a return flag:

-2 expected data type is different than INTEGER,

-1 incorrect use of the subroutine,

0 normal return,

1 default value assigned to argument N.

Subroutine DD11RN

This subroutine retrieves a real value from the physical record, performs data conversions, and returns the value in the binary form.

The subroutine call is

CALL DD11RN (X,IFLAG)

and the arguments are as follows.

X is a REAL variable that returns the data value.

IFLAG is an INTEGER variable that is used as a return flag:

-2 expected data type is different than REAL,

-1 incorrect use of the subroutine,

0 normal return,

1 default value assigned to argument X.

Subroutine DD11ST

This subroutine retrieves a character string from the physical record and returns it after justification.

The subroutine call is

CALL DD11ST (S,IFLAG)

and the arguments are as follows.

S is a REAL variable or a REAL array that returns the character string left justified and stored 10 characters per word.

IFLAG is an INTEGER variable that is used as a return flag:

-2 expected data type is different than character string,

-1 incorrect use of the subroutine,

0 normal return.

Subroutine DD11SW

This subroutine replaces the current logical data record definition by an alternative one provided both the logical data records have identical initial parts. The subroutine may be used when data-depending structuring is required.

The subroutine call is

CALL DD11SW (NR,IFLAG)

and the arguments are as follows.

NR is an INTEGER argument that must be set to the identifier of the alternative logical data record.

IFLAG is an INTEGER variable that is used as a return flag:

- -3 inconsistent logical record definitions,
- -2 incorrect value of argument NR (e.g., NR $\leq$ 0),
- -1 incorrect use of the subroutine,
- 0 normal return.

Subroutine DD22CN

This subroutine converts a complex value to the required physical representation and places the converted value in a physical record.

The subroutine call is

CALL DD22CN (Z,IFLAG)

and the arguments are as follows.

Z is a COMPLEX argument that submits the data value.

IFLAG is an INTEGER variable that is used as a return flag:

- -2 expected data type is different than COMPLEX,
- -1 incorrect use of the subroutine,
- 0 normal return.

Subroutine DD22IN

This subroutine converts an integer value to the required physical representation and places the converted value in a physical record.

The subroutine call is

CALL DD22IN (N,IFLAG)

and the arguments are as follows.

N is an INTEGER argument that submits the data value.

IFLAG is an INTEGER variable that is used as a return flag:

- -2 expected data type is different than INTEGER,
- -1 incorrect use of the subroutine,
- 0 normal return.

Subroutine DD22PR

This subroutine transfers a physical record from the buffer (in which it is completed from data values) to the output file.

The subroutine call is

CALL DD22PR (NR,IFLAG)

and the arguments are as follows.

NR is an INTEGER argument that must be set to the identifier of a logical data record.

IFLAG is an INTEGER variable that is used as a return flag:

- -3 the physical record in the buffer does not correspond to argument NR,
- -2 incorrect argument NR (e.g., NR $\leq$ 0),
- -1 incorrect use of the subroutine,
- >0 normal return; the value of IFLAG is equal to the consecutive number of the data record.

Subroutine DD22RN

This subroutine converts a real value to the required physical representation and places the converted value in the physical record.

The subroutine call is

CALL DD22RN (X,IFLAG)

and the arguments are as follows.

X is a REAL argument that submits the data value.

IFLAG is an INTEGER variable that is used as a return flag:

-2 expected data type is different than REAL,

-1 incorrect use of the subroutine,

0 normal return.

Subroutine DD22SR

This subroutine initializes a new physical data record that corresponds to the logical data record indicated in the subroutine call. The data values are set up in the record by a sequence of DD22CN, DD22IN, DD22RN and/or DD22ST calls that must follow the DD22SR call, and must correspond to the definition of indicated logical record.

The subroutine call is

CALL DD22SR (NR,IFLAG)

and the arguments are as follows.

NR is an INTEGER argument that must be set to the identifier of a logical data record.

IFLAG is an INTEGER variable that is used as a return flag:

-3 undefined logical record indicated by argument NR,

-2 incorrect value of argument NR (e.g., NR $\leq$ 0),

-1 incorrect use of the subroutine,

0 normal return.

Subroutine DD22ST

This subroutine places a character string in a physical record.

The subroutine call is

CALL DD22ST (S,IFLAG)

and the arguments are as follows.

S is a REAL argument or a REAL array that submits the string of characters; it is assumed that the string is left justified and stored 10 characters per word.

IFLAG is an INTEGER variable that is used as a return flag:

-2 expected data type is different than character string,

-1 incorrect use of the subroutine,

0 normal return.

VI. EXAMPLES

Example 1

The first program, T1DPWR, reads and prints basic power systems data from a descripted data file with the local name DATA. The data operations are grouped in 3 subroutines:

- subroutine PWRDD that defines the data file (DD00DF), all the data names (DD00DN), and two logical data records (DD11DR) corresponding to transmission line records and bus records with identifiers 1 and 2, respectively,
- subroutine PWRDD1 that reads one transmission line data record,
- subroutine PWRDD2 that reads one bus data record.

It should be noted that the 3 subroutines compose the PWRDD package [8] that is available in compiled form in the same library as the DDATA package.

The results are shown for the data file describing the 26-bus test power system [5-7].

```
      PROGRAM T1DPWR (DATA,OUTPUT,TAPE1=DATA,TAPE6=OUTPUT)                  000001
C                                                                           000002
C     THIS PROGRAM READS AND PRINTS DESCRIPTED DATA FILES FOR               000003
C     POWER SYSTEMS ANALYSIS.                                               000004
C                                                                           000005
      COMPLEX YL,YS1,YS2,V,S                                                000006
C                                                                           000007
C       READ DATA HEADER                                                    000008
C                                                                           000009
      CALL PWRDD(NB,NS,NT,DNAME,ICD,1,6,IRET)                               000010
      IF(IRET.LT.0) GOTO 90                                                 000011
      WRITE(6,111) DNAME,NB,NS,NT,IRET                                      000012
  111 FORMAT(/"0DATA-NAME : ",A10                                           000013
     1       /" NUMBER OF BUSES :              ",I4                         000014
     2       /" SLACK-BUS INDEX :              ",I4                         000015
     3       /" NUMBER OF TRANSMISSION-LINES :",I4                          000016
     4       /" RETURN FLAG :                  ",I4)                        000017
C                                                                           000018
C       READ TRANSMISSION-LINE DATA                                         000019
C                                                                           000020
      WRITE(6,222)                                                          000021
  222 FORMAT(//"  L I N E      LINE ADMITTANCE      LINE INPUT SHUNT",       000022
     1          "   LINE OUTPUT SHUNT    TAP"/)                            000023
   10 CALL PWRDD1(L1,L2,YL,YS1,YS2,TR,6,IRET)                               000024
      IF(IRET.LT.0) GOTO 90                                                 000025
      IF(IRET.EQ.0) GOTO 20                                                 000026
      WRITE(6,333) L1,L2,YL,YS1,YS2,TR                                      000027
  333 FORMAT(1X,2I4,2X,2F10.5,2(2X,2F9.5),2X,F5.2)                          000028
      GOTO 10                                                               000029
C                                                                           000030
C       RESET DATA FILE IF REQUIRED                                         000031
C                                                                           000032
   20 IF(MOD(ICD/10,10).LE.MOD(ICD/100,10)) CALL DD00RF(1,IRET)             000033
      IF(IRET.NE.0) GOTO 90                                                 000034
C                                                                           000035
C       READ BUS DATA                                                       000036
C                                                                           000037
      WRITE(6,444)                                                          000038
  444 FORMAT(//6X,"BUS T",7X,"BUS VOLTAGE",10X,"BUS POWER",7X,"ST-LOAD"     000039
     1 /)                                                                   000040
   30 CALL PWRDD2(N,K,V,S,SL,6,IRET)                                        000041
      IF(IRET.LT.0) GOTO 90                                                 000042
      IF(IRET.EQ.0) STOP                                                    000043
      WRITE(6,555) N,K,V,S,SL                                               000044
  555 FORMAT(5X,I4,I2,2(2X,2F9.5),2X,F9.5)                                  000045
      GOTO 30                                                               000046
C                                                                           000047
C       INCORRECT TERMINATION                                               000048
C                                                                           000049
   90 WRITE(6,999) IRET                                                     000050
  999 FORMAT(//" ERROR RETURN :",I3)                                        000051
      STOP 77777                                                            000052
      END                                                                   000053
C                                                                           000054
C                                                                           000055
      SUBROUTINE PWRDD (NB,NS,NL,NT,IK,LDT,LCH,IRET)                        000056
C                                                                           000057
C     THIS SUBROUTINE DEFINES LOGICAL DATA STRUCTURES AND READS DATA        000058
C     DESCRIPTORS.                                                          000059
C                                                                           000060
C     DATE : 82.08.12 (W.M.ZUBEREK)                                         000061
C                                                                           000062
      DIMENSION RLB(14),RL(7),RB(7)                                         000063
      EQUIVALENCE (RLB(1),RL(1)),(RLB(8),RB(1))                             000064
      DATA RLB/"LINEBINP","LINEBOUT","LINEINPA","LINERX","LINEOUTA",        000065
```

```
     1          "LINETRT","LINETAP",                                          000066
     2          "BUSNR","BUSTYPE","BUSVMOD","BUSVARG","BUSSTL",               000067
     3          "BUSGPQ","BUSLPQ"/                                            000068
C                                                                            000069
C       *RL* DESCRIBES LOGICAL TRANSMISSION-LINE RECORD                      000070
C       *RB* DESCRIBES LOGICAL BUS RECORD                                    000071
C                                                                            000072
      IRET=-1                                                                000073
      CALL DD00DF(LDT,NB,NS,NL,TN,IK)                                        000074
      IF(IK.LT.0.AND.LCH.GT.0) WRITE(LCH,111) IK                             000075
  111 FORMAT(/" DD00DF ERROR RETURN :",I3)                                   000076
      IF(IK.LT.0) RETURN                                                     000077
      IRET=-2                                                                000078
      CALL DD00DN(RLB,14,IER)                                                000079
      IF(IER.NE.0.AND.LCH.GT.0) WRITE(LCH,222) IER                           000080
  222 FORMAT(/" DD00DN ERROR RETURN :",I3)                                   000081
      IF(IER.LT.0) RETURN                                                    000082
      IF(IER.EQ.0) GOTO 10                                                   000083
      IF(LCH.GT.0) WRITE(LCH,333) (RLB(I),I=1,14)                            000084
  333 FORMAT(/" DATA-NAMES : ",A10/(14X,A10))                                000085
      RETURN                                                                 000086
   10 IRET=-3                                                                000087
      CALL DD11DR(1,RL,7,IER)                                                000088
      IF(IER.NE.0.AND.LCH.GT.0) WRITE(LCH,444) IER                           000089
  444 FORMAT(/" DD11DR/1 ERROR RETURN :",I3)                                 000090
      IF(IER.EQ.0) GOTO 20                                                   000091
      IF(LCH.GT.0) WRITE(LCH,333) (RL(I),I=1,7)                              000092
      RETURN                                                                 000093
   20 CALL DD11DR(2,RB,7,IER)                                                000094
      IF(IER.NE.0.AND.LCH.GT.0) WRITE(LCH,555) IER                           000095
  555 FORMAT(/" DD11DR/2 ERROR RETURN :",I3)                                 000096
      IF(IER.EQ.0) GOTO 30                                                   000097
      IF(LCH.GT.0) WRITE(LCH,333) (RB(I),I=1,7)                              000098
      RETURN                                                                 000099
   30 IRET=0                                                                 000100
      IK=10*(IK/10)                                                          000101
      RETURN                                                                 000102
      END                                                                    000103
C                                                                            000104
C                                                                            000105
      SUBROUTINE PWRDD1 (K1,K2,YA,Y1,Y2,T,LCH,IER)                           000106
      COMPLEX YA,Y1,Y2                                                       000107
C                                                                            000108
C     THIS SUBROUTINE READS ONE TRANSMISSION-LINE RECORD AND RETURNS.        000109
C                                                                            000110
C     DATE : 82.06.29 (W.M.ZUBEREK)                                          000111
C                                                                            000112
      CALL DD11GS(1,IER)                                                     000113
      IF(IER.LT.0.AND.LCH.GT.0) WRITE(LCH,111) IER                           000114
  111 FORMAT(/" DD11GS ERROR RETURN :",I3)                                   000115
      IF(IER.LE.0) RETURN                                                    000116
      CALL DD11IN(K1,IE)                                                     000117
      IF(IE.LT.0) IER=-4                                                     000118
      CALL DD11IN(K2,IE)                                                     000119
      IF(IE.LT.0) IER=-4                                                     000120
      CALL DD11CN(Y1,IE)                                                     000121
      IF(IE.LT.0) IER=-4                                                     000122
      CALL DD11CN(YA,IE)                                                     000123
      IF(IE.LT.0) IER=-4                                                     000124
      CALL DD11CN(Y2,IE)                                                     000125
      IF(IE.LT.0) IER=-4                                                     000126
      CALL DD11IN(K,IE)                                                      000127
      IF(IE.LT.0) IER=-4                                                     000128
      CALL DD11RN(T,IE)                                                      000129
      IF(IE.LT.0) IER=-4                                                     000130
```

```
      IF(K.EQ.0) T=1.0                                                  000131
      IF(K.EQ.5) T=1.0/T                                                000132
      YA=1.0/YA                                                         000133
      RETURN                                                            000134
      END                                                               000135
C                                                                       000136
C                                                                       000137
      SUBROUTINE PWRDD2 (N,K,V,S,SL,LCH,IER)                            000138
      COMPLEX V,S                                                       000139
C                                                                       000140
C     THIS SUBROUTINE READS ONE BUS DESCRIPTION RECORD AND RETURNS.     000141
C                                                                       000142
C     DATE : 82.06.29 (W.M.ZUBEREK)                                     000143
C                                                                       000144
      COMPLEX GPQ,HPQ                                                   000145
      FACT=3.14159265/180.0                                             000146
      CALL DD11GS(2,IER)                                                000147
      IF(IER.LT.0.AND.LCH.GT.0) WRITE(LCH,111) IER                      000148
  111 FORMAT(/" DD11GS ERROR RETURN :",I3)                              000149
      IF(IER.LE.0) RETURN                                               000150
      CALL DD11IN(N,IE)                                                 000151
      IF(IE.LT.0) IER=-4                                                000152
      CALL DD11IN(K,IE)                                                 000153
      IF(IE.LT.0) IER=-4                                                000154
      CALL DD11RN(VM,IE)                                                000155
      IF(IE.LT.0) IER=-4                                                000156
      CALL DD11RN(VA,IE)                                                000157
      IF(IE.LT.0) IER=-4                                                000158
      CALL DD11RN(SL,IE)                                                000159
      IF(IE.LT.0) IER=-4                                                000160
      CALL DD11CN(GPQ,IE)                                               000161
      IF(IE.LT.0) IER=-4                                                000162
      CALL DD11CN(HPQ,IE)                                               000163
      IF(IE.LT.0) IER=-4                                                000164
      S=GPQ-HPQ                                                         000165
      VA=FACT*VA                                                        000166
      V=CMPLX(VM*COS(VA),VM*SIN(VA))                                    000167
      RETURN                                                            000168
      END                                                               000169
```

DATA-NAME : DATA-26
NUMBER OF BUSES : 26
SLACK-BUS INDEX : 26
NUMBER OF TRANSMISSION-LINES : 32
RETURN FLAG : 0

L I N E		LINE ADMITTANCE		LINE INPUT SHUNT		LINE OUTPUT SHUNT		TAP
13	26	0.00000	-76.33528	0.00000	0.00000	0.00000	0.00000	1.03
26	16	0.00000	-25.51020	0.00000	0.00000	0.00000	0.00000	.96
16	23	0.00000	-2.31481	0.00000	0.00000	0.00000	0.00000	1.00
23	26	0.00000	-3.18471	0.00000	0.00000	0.00000	0.00000	1.00
2	10	0.00000	-66.66667	0.00000	0.00000	0.00000	0.00000	1.03
9	10	1.08752	-2.46912	0.00000	.41200	0.00000	.41200	1.00
9	12	2.46904	-5.60600	0.00000	.01820	0.00000	.01820	1.00
12	26	3.04887	-6.92145	0.00000	.01470	0.00000	.01470	1.00
9	14	1.00856	-3.91185	0.00000	.03190	0.00000	.03190	1.00
11	14	.92332	-3.57856	0.00000	.03490	0.00000	.03490	1.00
19	26	.90672	-3.74728	0.00000	.02950	0.00000	.02950	1.00
6	26	1.21929	-4.72029	0.00000	.02650	0.00000	.02650	1.00
6	19	4.30481	-17.75316	0.00000	.00740	0.00000	.00740	1.00
7	19	.61120	-2.52439	0.00000	.04370	0.00000	.04370	1.00
6	7	.67790	-2.62697	0.00000	.04750	0.00000	.04750	1.00
11	22	1.02021	-4.45980	0.00000	.02480	0.00000	.02480	1.00
8	11	.72053	-2.79484	0.00000	.04470	0.00000	.04470	1.00
17	22	.78665	-5.28218	0.00000	.02370	0.00000	.02370	1.00
8	21	.85014	-3.29299	0.00000	.03790	0.00000	.03790	1.00
17	21	.42095	-3.20106	0.00000	.03870	0.00000	.03870	1.00
1	4	1.00684	-3.90536	0.00000	.03190	0.00000	.03190	1.00
4	21	1.02258	-3.96458	0.00000	.03150	0.00000	.03150	1.00
20	21	0.00000	-32.78689	0.00000	0.00000	0.00000	0.00000	.97
15	1	0.00000	-68.02721	0.00000	0.00000	0.00000	0.00000	.89
2	13	1.69543	-13.93204	0.00000	.30170	0.00000	.30170	1.00
1	7	3.03434	-11.96964	0.00000	.04040	0.00000	.04040	1.00
15	20	2.72863	-15.73426	0.00000	.44710	0.00000	.44710	1.00
2	18	1.97260	-16.20728	0.00000	.25930	0.00000	.25930	1.00
1	3	0.00000	-25.51020	0.00000	0.00000	0.00000	0.00000	.98
24	3	0.00000	-6.89655	0.00000	0.00000	0.00000	0.00000	.98
5	21	0.00000	-5.71429	0.00000	0.00000	0.00000	0.00000	.99
5	25	0.00000	-6.49351	0.00000	0.00000	0.00000	0.00000	1.03

BUS	T	BUS VOLTAGE		BUS POWER		ST-LOAD
1	0	1.00000	0.00000	-.32000	-.21000	0.00000
2	0	1.00000	0.00000	0.00000	0.00000	0.00000
3	0	1.00000	0.00000	-.57000	-.17000	0.00000
4	0	1.00000	0.00000	-.42000	-.21000	0.00000
5	0	1.00000	0.00000	-.42000	-.11000	0.00000
6	0	1.00000	0.00000	-.40000	-.10000	0.00000
7	0	1.00000	0.00000	-1.11000	-.27000	0.00000
8	0	1.00000	0.00000	-.22000	-.06000	0.00000
9	0	1.00000	0.00000	-.67000	-.21000	0.00000
10	0	1.00000	0.00000	-1.02000	-.27000	0.00000
11	0	1.00000	0.00000	-.48000	-.14000	0.00000
12	0	1.00000	0.00000	-.48000	-.12000	0.00000
13	0	1.00000	0.00000	0.00000	0.00000	0.00000
14	0	1.00000	0.00000	0.00000	0.00000	0.00000
15	0	1.00000	0.00000	0.00000	0.00000	0.00000
16	0	1.00000	0.00000	-1.31000	-.30000	0.00000
17	0	1.00000	0.00000	-.08000	-.01000	0.00000
18	1	1.07000	0.00000	2.80000	1.07000	0.00000

19	1	1.05000	0.00000	1.45000	1.05000	0.00000
20	1	1.00000	0.00000	2.80000	1.00000	0.00000
21	1	1.02000	0.00000	1.10000	1.02000	0.00000
22	1	.89000	0.00000	-.56000	.89000	0.00000
23	1	1.00000	0.00000	-.04000	1.00000	0.00000
24	1	1.00000	0.00000	-.05000	1.00000	0.00000
25	1	1.00000	0.00000	.63000	1.00000	0.00000
26	2	1.01000	0.00000	0.00000	0.00000	0.00000

Example 2

The second program, T2DPWR, reads power systems data, solves the load flow problem using sparse matrix techniques (Harwell package MA28 [9]) and the fast decoupled method (LFLFD package [10]), and stores the solution in a new descripted data file with the local name SDATA. Reading and initial processing of data is performed by the PWRDS package [11] with PWRDS2 entry. Storing the results in a new data file is controlled by STRRES subroutine that calls:

- subroutine PWRDF to define the output file (DD00WF) and the data descriptors (DD00DD) as well as two logical data records corresponding to transmission line records and bus records with identifiers 10 and 9, respectively,
- subroutine PWRDF1 to write one transmission line data record,
- subroutine PWRDF2 to write one bus data record.

The subroutine PWRDF, PWRDF1 and PWRDF2 perform functions that are "symmetrical" to those of the package PWRDD, and also are available in compiled form in the same library as the DDATA package.

The results of the program T2DPWR are shown for the 23-bus test power system [4,5]. Moreover, the contents of the created data file are presented in addition to the output of the program T1DPWR (Example 1) executed for the data file already created.

```
      PROGRAM T2DPWR (DATA,SDATA,OUTPUT,TAPE1=DATA,TAPE2=SDATA,          000001
     1                TAPE6=OUTPUT)                                      000002
C                                                                        000003
C  THIS PROGRAM SOLVES THE LOAD FLOW PROBLEM USING SPARSE MATRIX         000004
C  TECHNIQUES (HARWELL PACKAGE MA28) AND THE FAST-DECOUPLED METHOD       000005
C  (PACKAGE LFLFD) AND STORES THE RESULTS AS A DESCRIPTED DATA FILE.     000006
C                                                                        000007
      DIMENSION W(3000)                                                  000008
      EXTERNAL FLOW                                                      000009
      CALL SECOND(TIME1)                                                 000010
      CALL PWRDS2(FLOW,1,6,W,3000,IRET)                                  000011
      IF(IRET.NE.0) WRITE(6,111) IRET                                    000012
  111 FORMAT(/" PWRDS1 RETURN FLAG :",I3)                                000013
      CALL SECOND(TIME2)                                                 000014
      EXTIME=TIME2-TIME1                                                 000015
      WRITE(6,222) EXTIME                                                000016
  222 FORMAT(/" TOTAL EXECUTION TIME :",F7.3," SECONDS")                 000017
      STOP                                                               000018
      END                                                                000019
C                                                                        000020
C                                                                        000021
      SUBROUTINE FLOW (DN,NBS,NS,NTL,NB,NLB,NZ,NLZ,INDR,INDC,IBT,        000022
     1                 Y,YS,V,S,STL,LB1,LB2,YL,YS1,YS2,TRT,W,LW,LCH,     000023
     2                 IFLAG)                                            000024
      DIMENSION INDR(NB),INDC(NZ),IBT(NB),STL(NTL),LB1(NTL),LB2(NTL),    000025
     1          TRT(NTL),W(LW)                                           000026
      COMPLEX Y(NZ),YS(NBS),YL(NTL),YS1(NTL),YS2(NTL),V(NBS),S(NBS)      000027
C                                                                        000028
      IF(LCH.GT.0) WRITE(LCH,111) DN,NBS,NS,NTL,NZ,NLB,NLZ,LW            000029
  111 FORMAT(/"0DATA-NAME : ",A10                                        000030
     1       /" NUMBER OF BUSES :                ",I4                  000031
     2       /" SLACK-BUS INDEX :                ",I4                  000032
     3       /" NUMBER OF TRANSMISSION LINES :",I4                      000033
     4       /" NUMBER OF NON-ZEROS :            ",I4                  000034
     5       /" NUMBER OF LOAD-BUSES :           ",I4                  000035
     6       /" NUMBER OF LOAD NON-ZEROS :       ",I4                  000036
     7       /" REMAINING WORKSPACE :          ",I5/)                   000037
      IFLAG=-5                                                           000038
      ITL=20                                                             000039
      VEPS=1.E-6                                                         000040
      MODE=0                                                             000041
      TIMEX=2.5                                                          000042
      CALL LFLFD1(NB,NLB,NZ,NLZ,INDR,INDC,IBT,Y,YS,V,S,W,LW,ITL,VEPS,    000043
     1            TIMEX,MODE,IRET)                                       000044
      IF(IRET.LT.0) RETURN                                               000045
      IF(LCH.GT.0) WRITE(LCH,222) IRET,ITL,VEPS,TIMEX                    000046
  222 FORMAT(/" RETURN FLAG :                     ",I4                   000047
     1       /" NUMBER OF ITERATIONS :            ",I4                  000048
     2       /" ACCURACY OBTAINED :      ",1PE10.3                      000049
     3       /" SOLUTION TIME :                ",0PF6.3," SECONDS")     000050
      CALL PRTRES(NB,NS,INDR,INDC,IBT,Y,YS,V,S)                          000051
      IFLAG=-6                                                           000052
      CALL STRRES(DN,NBS,NS,NTL,LB1,LB2,YL,YS1,YS2,TRT,IBT,V,S,STL,      000053
     1            2,LCH,IRET)                                            000054
      IF(IRET.LT.0) RETURN                                               000055
      IFLAG=0                                                            000056
      RETURN                                                             000057
      END                                                                000058
C                                                                        000059
C                                                                        000060
      SUBROUTINE PRTRES(NB,NL,INDR,INDC,IBT,Y,YS,V,S)                    000061
      DIMENSION INDR(1),INDC(1),IBT(1)                                   000062
      COMPLEX Y(1),YS(1),V(1),S(1)                                       000063
C                                                                        000064
C     THIS SUBROUTINE PRINTS FINAL RESULTS OF THE LOAD FLOW SOLUTION     000065
```

```
C     (AND STORES REACTIVE POWERS FOR GENERATOR BUSES).                      000066
C                                                                            000067
C     NB    - NUMBER OF BUSES (EXCLUDING THE SLACK BUS),                     000068
C     NL    - INDEX OF THE SLACK BUS,                                        000069
C     INDR  - ROW INDEX OF THE SPARSE BUS-ADMITTANCE MATRIX,                 000070
C     INDC  - COLUMN INDEX OF THE SPARSE BUS-ADMITTANCE MATRIX,              000071
C     IBT   - VECTOR OF BUS TYPES (0 - LOAD, 1 - GENERATOR),                 000072
C     Y     - SPARSE COMPLEX BUS-ADMITTANCE MATRIX,                          000073
C     YS    - COMPLEX VECTOR OF SLACK ADMITTANCES,                           000074
C     V     - COMPLEX VECTOR OF BUS-VOLTAGES (RECTANGULAR MODE).             000075
C                                                                            000076
      COMPLEX CC,PW                                                          000077
      FACT=180.0/3.14159265                                                  000078
      NBS=NB+1                                                               000079
      WRITE(6,900) NBS                                                       000080
  900 FORMAT(//10X,"LOAD FLOW SOLUTION BY FAST-DECOUPLED METHOD",            000081
     1 //20X,I4,"-BUS POWER SYSTEM"///51X,"GENERATOR"/"  BUS",               000082
     2 "    COMPLEX BUS VOLTAGE    POLAR BUS VOLTAGE  REACTIVE POWER"/)      000083
      DO 98 I=1,NB                                                           000084
      II=I                                                                   000085
      IF(II.GE.NL) II=II+1                                                   000086
      V1=CABS(V(I))                                                          000087
      V2=ATAN2(AIMAG(V(I)),REAL(V(I)))*FACT                                  000088
      IF(IBT(I).EQ.0) GOTO 97                                                000089
      J1=1                                                                   000090
      IF(I.GT.1) J1=INDR(I-1)+1                                              000091
      J2=INDR(I)                                                             000092
      CC=YS(I)*V(NBS)                                                        000093
      DO 96 J=J1,J2                                                          000094
      K=INDC(J)                                                              000095
   96 CC=CC+Y(J)*V(K)                                                        000096
      Q=AIMAG(V(I)*CONJG(CC))                                                000097
      S(I)=CMPLX(REAL(S(I)),Q)                                               000098
      WRITE(6,902) II,V(I),V1,V2,Q                                           000099
  902 FORMAT(1X,I4,2X,2F9.5,2H*J,2X,F9.5,F9.2,3X,F10.5)                      000100
      GOTO 98                                                                000101
   97 WRITE(6,902) II,V(I),V1,V2                                             000102
   98 CONTINUE                                                               000103
      CC=(0.0,0.0)                                                           000104
      DO 99 I=1,NBS                                                          000105
   99 CC=CC+YS(I)*V(I)                                                       000106
      PW=V(NBS)*CONJG(CC)                                                    000107
      S(NBS)=PW                                                              000108
      WRITE(6,903) PW                                                        000109
  903 FORMAT(/" COMPLEX SLACK BUS POWER :",3X,2F9.5,2H*J)                    000110
      RETURN                                                                 000111
      END                                                                    000112
C                                                                            000113
C                                                                            000114
      SUBROUTINE STRRES (DN,NBS,NS,NT,L1,L2,YL,YS1,YS2,TRT,IBT,V,S,          000115
     1                   STL,LDF,LCH,IRET)                                   000116
      DIMENSION L1(NT),L2(NT),TRT(NT),IBT(NBS),STL(NBS)                      000117
      COMPLEX YL(NT),YS1(NT),YS2(NT),V(NBS),S(NBS)                           000118
C                                                                            000119
C     THIS SUBROUTINE STORES THE RESULTS IN A DESCRIPTED DATA FILE.          000120
C                                                                            000121
      ENCODE(10,111,DNAME) DN                                                000122
  111 FORMAT(2HS-,A8)                                                        000123
      CALL PWRDF(NBS,NS,NT,DNAME,122,LDF,LCH,IRET)                           000124
      IF(IRET.NE.0.AND.LCH.GT.0) WRITE(LCH,222) IRET                         000125
  222 FORMAT(/" PWRDF  ERROR RETURN :",I4)                                   000126
      IF(IRET.LT.0) RETURN                                                   000127
      DO 10 I=1,NT                                                           000128
      CALL PWRDF1(L1(I),L2(I),YL(I),YS1(I),YS2(I),TRT(I),LCH,IRET)           000129
      IF(IRET.LT.0.AND.LCH.GT.0) WRITE(LCH,444) IRET                         000130
```

```
  444 FORMAT(/" PWRDF1 ERROR RETURN :",I4)                              000131
      IF(IRET.LT.0) RETURN                                              000132
   10 CONTINUE                                                          000133
      DO 20 I=1,NBS                                                     000134
      CALL PWRDF2(I,IBT(I),V(I),S(I),STL(I),LCH,IRET)                   000135
      IF(IRET.LT.0.AND.LCH.GT.0) WRITE(LCH,666) IRET                    000136
  666 FORMAT(/" PWRDF2 ERROR RETURN :",I4)                              000137
      IF(IRET.LT.0) RETURN                                              000138
   20 CONTINUE                                                          000139
   30 CALL DD00RF(LDF,IRET)                                             000140
      IF(IRET.NE.0.AND.LCH.GT.0) WRITE(LCH,999) IRET                    000141
  999 FORMAT(/" DD00RF ERROR RETURN :",I4/)                             000142
      RETURN                                                            000143
      END                                                               000144
C                                                                       000145
C                                                                       000146
      SUBROUTINE PWRDF (NB,NS,NT,DNM,ICD,LFL,LCH,IFLAG)                 000147
C                                                                       000148
C     THIS SUBROUTINE DEFINES THE OUTPUT FILE, DATA DESCRIPTORS,        000149
C     AND LOGICAL DATA RECORDS FOR TRANSMISSION-LINE DATA AND           000150
C     BUS DATA.                                                         000151
C                                                                       000152
C     DATE : 82.08.18 (W.M.ZUBEREK)                                     000153
C                                                                       000154
      DIMENSION RB(7),RL(7),DN(24),KD(24),TRV(1)                        000155
      COMMON /PWRDF0/ NBUS,NLINE                                        000156
      DATA LT/1/,LD/24/                                                 000157
      DATA RB/8HBUSNR   ,8HBUSTYPE ,8HBUSVMOD ,8HBUSVARG ,8HBUSGPQ   ,  000158
     1        8HBUSLPQ  ,8HBUSSTL  /                                     000159
      DATA RL/8HLINEBINP,8HLINEBOUT,8HLINERX  ,8HLINEINPA,8HLINEOUTA,   000160
     1        8HLINETRT ,8HLINETAP /                                     000161
      DATA DN/8HBUSNR   ,8HBUSTYPE ,8HBUSVMOD ,8HBUSVARG ,8HBUSGP     ,  000162
     1        8HBUSGQ   ,8HBUSLP   ,8HBUSLQ   ,8HBUSSTL  ,8HBUSGPQ   ,  000163
     2        8HBUSLPQ  ,8HLINEBINP,8HLINEBOUT,8HLINEINPC,8HLINEINPS,   000164
     3        8HLINER   ,8HLINEX   ,8HLINEOUTC,8HLINEOUTS,8HLINETRT ,   000165
     4        8HLINETAP ,8HLINEINPA,8HLINERX  ,8HLINEOUTA/              000166
      DATA KD/1100050000,1100010000,1200100101,1200100100,1200060100,   000167
     1        1200060100,1200060100,1200060100,1200060100,1305060100,   000168
     2        1307060100,2100050000,2100050000,2200060100,2200060100,   000169
     3        2200060101,2200060101,2200060100,2200060100,2100010001,   000170
     4        2200050101,2314060100,2316060101,2312060100/              000171
      DATA TRV/1.0/                                                     000172
      NBUS=NB                                                           000173
      NLINE=NT                                                          000174
      IFLAG=-1                                                          000175
      CALL DD00WF(LFL,NB,NS,NT,DNM,ICD,IE)                              000176
      IF(IE.NE.0.AND.LCH.GT.0) WRITE(LCH,111) IE                        000177
  111 FORMAT(/" DD00WF ERROR RETURN :",I4)                              000178
      IF(IE.LT.0) RETURN                                                000179
      CALL DD00DD(DN,KD,LD,TRV,LT,IE)                                   000180
      IF(IE.NE.0.AND.LCH.GT.0) WRITE(LCH,222) IE                        000181
  222 FORMAT(/" DD00DD ERROR RETURN :",I4)                              000182
      IF(IE.LT.0) RETURN                                                000183
      CALL DD11DR(10,RL,7,IE)                                           000184
      IF(IE.NE.0.AND.LCH.GT.0) WRITE(LCH,333) IE,(RL(I),I=1,7)          000185
  333 FORMAT(/" DD11DR ERROR RETURN :",I4//(10X,A10))                   000186
      IF(IE.LT.0) RETURN                                                000187
      CALL DD11DR(9,RB,7,IE)                                            000188
      IF(IE.NE.0.AND.LCH.GT.0) WRITE(LCH,333) IE,(RB(I),I=1,7)          000189
      IF(IE.LT.0) RETURN                                                000190
      IFLAG=0                                                           000191
      RETURN                                                            000192
      END                                                               000193
C                                                                       000194
C                                                                       000195
```

```
      SUBROUTINE PWRDF1 (L1,L2,YL,YS1,YS2,TR,LCH,IFLAG)                 000196
      COMPLEX ZL,YL,YS1,YS2                                             000197
C                                                                       000198
C     THIS SUBROUTINE WRITES ONE TRANSMISSION-LINE RECORD AND RETURNS   000199
C     (END-OF-SUBFILE RECORD IS WRITTEN AFTER THE LAST DATA RECORD).    000200
C                                                                       000201
C     DATE : 82.08.18 (W.M.ZUBEREK)                                     000202
C                                                                       000203
      COMMON /PWRDF0/ NBUS,NLINE                                        000204
      IFLAG=-4                                                          000205
      IF(NLINE.LE.0) RETURN                                             000206
      IFLAG=-1                                                          000207
      ZL=1.0/YL                                                         000208
      CALL DD22SR(10,IE)                                                000209
      IF(IE.NE.0.AND.LCH.GT.0) WRITE(LCH,111) IE                        000210
  111 FORMAT(/" DD22SR ERROR RETURN :",I4)                              000211
      IF(IE.LT.0) RETURN                                                000212
      IFLAG=0                                                           000213
      CALL DD22IN(L1,IE)                                                000214
      IF(IE.NE.0) IFLAG=-2                                              000215
      CALL DD22IN(L2,IE)                                                000216
      IF(IE.NE.0) IFLAG=-2                                              000217
      CALL DD22CN(ZL,IE)                                                000218
      IF(IE.NE.0) IFLAG=-2                                              000219
      CALL DD22CN(YS1,IE)                                               000220
      IF(IE.NE.0) IFLAG=-2                                              000221
      CALL DD22CN(YS2,IE)                                               000222
      IF(IE.NE.0) IFLAG=-2                                              000223
      IF(TR.EQ.1.0) GOTO 10                                             000224
      CALL DD22IN(1,IE)                                                 000225
      IF(IE.NE.0) IFLAG=-2                                              000226
      CALL DD22RN(TR,IE)                                                000227
      IF(IE.NE.0) IFLAG=-2                                              000228
   10 IF(IFLAG.LT.0.AND.LCH.GT.0) WRITE(LCH,222)                        000229
  222 FORMAT(/" DD22IN/DD22RN/DD22CN ERROR RETURN")                     000230
      IF(IFLAG.LT.0) RETURN                                             000231
      IFLAG=-3                                                          000232
      CALL DD22PR(10,IE)                                                000233
      IF(IE.LT.0.AND.LCH.GT.0) WRITE(LCH,333) IE                        000234
  333 FORMAT(/" DD22PR ERROR RETURN :",I4)                              000235
      IF(IE.LT.0) RETURN                                                000236
      NLINE=NLINE-1                                                     000237
      IFLAG=0                                                           000238
      IF(NLINE.NE.0) RETURN                                             000239
      CALL DD00ES(10,IE)                                                000240
      IF(IE.LT.0.AND.LCH.GT.0) WRITE(LCH,999) IE                        000241
  999 FORMAT(/" DD00ES ERROR RETURN :",I4)                              000242
      RETURN                                                            000243
      END                                                               000244
C                                                                       000245
C                                                                       000246
      SUBROUTINE PWRDF2 (N,K,V,S,STL,LCH,IFLAG)                         000247
      COMPLEX V,S,SG,SL                                                 000248
C                                                                       000249
C     THIS SUBROUTINE WRITES ONE BUS DATA RECORD AND RETURNS            000250
C     (END-OF-SUBFILE RECORD IS WRITTEN AFTER THE LAST DATA RECORD).    000251
C                                                                       000252
C     DATE : 82.08.18 (W.M.ZUBEREK)                                     000253
C                                                                       000254
      COMMON /PWRDF0/ NBUS,NLINE                                        000255
      FACT=180.0/3.14159265                                             000256
      IFLAG=-4                                                          000257
      IF(NBUS.LE.0) RETURN                                              000258
      IFLAG=-1                                                          000259
      CALL DD22SR(9,IE)                                                 000260
```

```
      IF(IE.NE.0.AND.LCH.GT.0) WRITE(LCH,111) IE                        000261
  111 FORMAT(/" DD22SR ERROR RETURN :",I4)                              000262
      IF(IE.LT.0) RETURN                                                000263
      IFLAG=0                                                           000264
      CALL DD22IN(N,IE)                                                 000265
      IF(IE.NE.0) IFLAG=-2                                              000266
      CALL DD22IN(K,IE)                                                 000267
      IF(IE.NE.0) IFLAG=-2                                              000268
      SG=(0.0,0.0)                                                      000269
      SL=(0.0,0.0)                                                      000270
      IF(K.EQ.0) SL=-S                                                  000271
      IF(K.NE.0) SG=S                                                   000272
      VM=CABS(V)                                                        000273
      VA=ATAN2(AIMAG(V),REAL(V))*FACT                                   000274
      CALL DD22RN(VM,IE)                                                000275
      IF(IE.NE.0) IFLAG=-2                                              000276
      CALL DD22RN(VA,IE)                                                000277
      IF(IE.NE.0) IFLAG=-2                                              000278
      CALL DD22CN(SG,IE)                                                000279
      IF(IE.NE.0) IFLAG=-2                                              000280
      CALL DD22CN(SL,IE)                                                000281
      IF(IE.NE.0) IFLAG=-2                                              000282
      IF(STL.EQ.0.0) GOTO 10                                            000283
      CALL DD22RN(STL,IE)                                               000284
      IF(IE.NE.0) IFLAG=-2                                              000285
   10 IF(IFLAG.LT.0.AND.LCH.GT.0) WRITE(LCH,222)                        000286
  222 FORMAT(/" DD22IN/DD22RN/DD22CN ERROR RETURN")                     000287
      IF(IFLAG.LT.0) RETURN                                             000288
      IFLAG=-3                                                          000289
      CALL DD22PR(9,IE)                                                 000290
      IF(IE.LT.0.AND.LCH.GT.0) WRITE(LCH,333) IE                        000291
  333 FORMAT(/" DD22PR ERROR RETURN :",I4)                              000292
      IFLAG=0                                                           000293
      NBUS=NBUS-1                                                       000294
      IF(NBUS.NE.0) RETURN                                              000295
      CALL DD00ES(9,IE)                                                 000296
      IF(IE.LT.0.AND.LCH.GT.0) WRITE(LCH,999) IE                        000297
  999 FORMAT(/" DD00ES ERROR RETURN :",I4)                              000298
      RETURN                                                            000299
      END                                                               000300
```

DATA-NAME : DATA-23
NUMBER OF BUSES : 23
SLACK-BUS INDEX : 23
NUMBER OF TRANSMISSION LINES : 30
NUMBER OF NON-ZEROS : 76
NUMBER OF LOAD-BUSES : 17
NUMBER OF LOAD NON-ZEROS : 53
REMAINING WORKSPACE : 2277

RETURN FLAG : 0
NUMBER OF ITERATIONS : 13
ACCURACY OBTAINED : 2.529E-07
SOLUTION TIME : .195 SECONDS

LOAD FLOW SOLUTION BY FAST-DECOUPLED METHOD

23-BUS POWER SYSTEM

BUS	COMPLEX BUS VOLTAGE		POLAR BUS VOLTAGE		GENERATOR REACTIVE POWER
1	1.03142	.01794*J	1.03157	1.00	
2	1.00589	.02628*J	1.00623	1.50	
3	1.00396	.08640*J	1.00767	4.92	
4	1.00152	.06700*J	1.00376	3.83	
5	.99740	.05462*J	.99890	3.14	
6	1.00615	.14056*J	1.01592	7.95	
7	.98975	.08071*J	.99304	4.66	
8	.99314	.04138*J	.99400	2.39	
9	1.01123	.21371*J	1.03357	11.93	
10	1.01228	.24728*J	1.04205	13.73	
11	.98065	.24858*J	1.01166	14.22	
12	.94305	.38733*J	1.01950	22.33	
13	.94647	.35230*J	1.01008	20.44	
14	.95293	.33946*J	1.01159	19.61	
15	.94772	.34178*J	1.00746	19.83	
16	.94076	.41251*J	1.02723	23.68	
17	.94554	.40069*J	1.02693	22.97	
18	1.02816	.06154*J	1.03000	3.43	.42038
19	1.04752	.07216*J	1.05000	3.94	.72284
20	1.02234	.22509*J	1.05000	12.94	.45105
21	.93006	.48733*J	1.05000	27.65	1.90158
22	.93399	.47975*J	1.05000	27.19	1.25886

COMPLEX SLACK BUS POWER : -.68393 .39132*J

TOTAL EXECUTION TIME : 1.762 SECONDS

```
000   23   23   30  S-DATA-23              122 (V:82.08) 82/09/11. 14.06.18.
0A0    1    4    5  BUSNR          .000000000.000000000.
0A0    1    9    1  BUSTYPE        .000000000.000000000.
0A0    2   10   10  BUSVMOD        1.000000001.000000000
0A0    2   20   10  BUSVARG        1.000000000.000000000.
0A0    2   30    6  BUSGP          1.000000000.000000000.
0A0    2   36    6  BUSGQ          1.000000000.000000000.
0A0    2   42    6  BUSLP          1.000000000.000000000.
0A0    2   48    6  BUSLQ          1.000000000.000000000.
0A0    2   54    6  BUSSTL         1.000000000.000000000.
0A0    3   30    6  BUSGPQ         1.000000000.000000000.
0A0    3   42    6  BUSLPQ         1.000000000.000000000.
0B0    1    4    5  LINEBINP       .000000000.000000000.
0B0    1    9    5  LINEBOUT       .000000000.000000000.
0B0    2   14    6  LINEINPC       1.000000000.000000000.
0B0    2   20    6  LINEINPS       1.000000000.000000000.
0B0    2   26    6  LINER          1.000000001.000000000
0B0    2   32    6  LINEX          1.000000001.000000000
0B0    2   38    6  LINEOUTC       1.000000000.000000000.
0B0    2   44    6  LINEOUTS       1.000000000.000000000.
0B0    1   50    1  LINETRT        .000000001.000000000
0B0    2   51    5  LINETAP        1.000000001.000000000
0B0    3   14    6  LINEINPA       1.000000000.000000000.
0B0    3   26    6  LINERX         1.000000001.000000000
0B0    3   38    6  LINEOUTA       1.000000000.000000000.
0*0
B00   23    1.00000.00590.02420.05400.00000.00590
B00   23    2.00000.00755.03090.06930.00000.00755
B00   18    3.00000.00985.04040.08850.00000.00985
B00    6    3.00000.00785.03250.07090.00000.00785
B00   18    5.00000.01710.06150.16200.00000.01710
B00    1    4.00000.01600.05750.15200.00000.01600
B00    2    7.00000.00740.02560.07000.00000.00740
B00    7    5.00000.00560.02290.05040.00000.00560
B00    6    4.00000.01090.04460.10030.00000.01090
B00   19    8.00000.02280.02330.05140.00000.02280
B00    6    8.00000.01455.05970.13150.00000.01455
B00    7    8.00000.01455.05970.13150.00000.01455
B00   10   20.00000.11865.00430.03510.00000.11865
B00   20    9.00000.11865.00430.03510.00000.11865
B00   11    9.00000.10390.00380.03070.00000.10390
B00   14   11.00000.09755.00350.02880.00000.09755
B00   22   10.00000.24355.00890.07260.00000.24355
B00   12   13.00000.02715.00100.00800.00000.02715
B00   13   14.00000.05665.00210.01670.00000.05665
B00   15   14.00000.04310.00160.01270.00000.04310
B00   21   15.00000.12255.00450.03620.00000.12255
B00   17   14.00000.06490.00240.01920.00000.06490
B00   21   16.00000.05280.00190.01560.00000.05280
B00   16   17.00000.03850.00140.01140.00000.03850
B00   22   12.00000.05545.00200.01640.00000.05545
B00    9    6.00000.00000.00220.00890.00000.0000011.040
B00   10    6.00000.00000.00220.00890.00000.0000011.030
B00    9    7.00000.00000.00190.15000.00000.0000011.050
B00   10    7.00000.00000.00220.00890.00000.0000011.060
B00   23   13.00000.00000.00250.20000.00000.00000
B*0
A00    101.03157413.996741923.00000.00000.00000.00000
A00    201.006234191.49666891.00000.00000.47000.12000
A00    301.007671194.91347637.00000.00000.51000.13000
A00    401.003757073.82734406.00000.00000.41000.10000
A00    50.9988980543.13535156.00000.00000.48000-.1200
A00    601.015916177.95273304.00000.00000.01000.00000
A00    70.9930093834.66209503.00000.000001.5000.38000
A00    80.9940914032.38594732.00000.000001.7700.44000
```

```
A00     901.0335666611.9330784.00000.00000.06000.00000
A00    1001.0420485413.7275731.00000.00000-.0400.00000
A00    1101.0116626414.2238314.00000.000002.0100.50000
A00    1201.0194977222.3289136.00000.000001.3200.33000
A00    1301.0100814120.4431622.00000.000003.4400.85000
A00    1401.0115888119.6071152.00000.000001.0400.26000
A00    1501.0074608019.8310952.00000.000003.7600.94000
A00    1601.0272276723.6764869.00000.000003.7500.94000
A00    1701.0269318922.9656076.00000.00000-2.100-.5200
A00    1811.030000003.42544759.26000.42038.00000.00000
A00    1911.050000003.94071744.89000.72284.00000.00000
A00    2011.0500000012.9379946.20000.45105.00000.00000
A00    2111.0500000027.65340759.03001.9016.00000.00000
A00    2211.0500000027.18733339.23001.2589.00000.00000
A00    2321.04000000.000000000-.6839.89132.00000.00000
A*0
```

DATA-NAME : S-DATA-23
NUMBER OF BUSES : 23
SLACK-BUS INDEX : 23
NUMBER OF TRANSMISSION-LINES : 30
RETURN FLAG : 0

LINE		LINE ADMITTANCE		LINE INPUT SHUNT		LINE OUTPUT SHUNT		TAP
23	1	6.91105	-15.42135	0.00000	.00590	0.00000	.00590	1.00
23	2	5.36710	-12.03689	0.00000	.00755	0.00000	.00755	1.00
18	3	4.24477	-9.33008	0.00000	.00985	0.00000	.00985	1.00
6	3	5.34271	-11.65532	0.00000	.00785	0.00000	.00785	1.00
18	5	2.04821	-5.39528	0.00000	.01710	0.00000	.01710	1.00
1	4	2.18002	-5.75283	0.00000	.01600	0.00000	.01600	1.00
2	7	4.74360	-12.48315	0.00000	.00740	0.00000	.00740	1.00
7	5	7.47250	-16.44603	0.00000	.00560	0.00000	.00560	1.00
6	4	3.70148	-8.32417	0.00000	.01090	0.00000	.01090	1.00
19	8	7.31589	-16.18891	0.00000	.02280	0.00000	.02280	1.00
6	8	2.86244	-6.30504	0.00000	.01455	0.00000	.01455	1.00
7	8	2.86244	-6.30504	0.00000	.01455	0.00000	.01455	1.00
10	20	3.43862	-28.06877	0.00000	.11865	0.00000	.11865	1.00
20	9	3.43862	-28.06877	0.00000	.11865	0.00000	.11865	1.00
11	9	3.97103	-32.08176	0.00000	.10390	0.00000	.10390	1.00
14	11	4.15830	-34.21687	0.00000	.09755	0.00000	.09755	1.00
22	10	1.66356	-13.57017	0.00000	.24355	0.00000	.24355	1.00
12	13	15.38462	-123.07692	0.00000	.02715	0.00000	.02715	1.00
13	14	7.41264	-58.94811	0.00000	.05665	0.00000	.05665	1.00
15	14	9.76503	-77.50992	0.00000	.04310	0.00000	.04310	1.00
21	15	3.38170	-27.20393	0.00000	.12255	0.00000	.12255	1.00
17	14	6.41026	-51.20205	0.00000	.06490	0.00000	.06490	1.00
21	16	7.69324	-63.16557	0.00000	.05280	0.00000	.05280	1.00
16	17	10.61249	-86.41601	0.00000	.03850	0.00000	.03850	1.00
22	12	7.32708	-60.08206	0.00000	.05545	0.00000	.05545	1.00
9	6	.32650	-11.91000	0.00000	0.00000	0.00000	0.00000	1.04
10	6	.32650	-11.91000	0.00000	0.00000	0.00000	0.00000	1.03
9	7	.11240	-7.69066	0.00000	0.00000	0.00000	0.00000	1.05
10	7	.32650	-11.91000	0.00000	0.00000	0.00000	0.00000	1.06
23	13	.06249	-4.99922	0.00000	0.00000	0.00000	0.00000	1.00

BUS	T	BUS VOLTAGE		BUS POWER		ST-LOAD
1	0	1.03142	.01794	0.00000	0.00000	0.00000
2	0	1.00589	.02628	-.47000	-.12000	0.00000
3	0	1.00896	.00640	-.51000	-.13000	0.00000
4	0	1.00152	.06700	-.41000	-.10000	0.00000
5	0	.99740	.05463	-.42000	.12000	0.00000
6	0	1.00615	.14056	-.01000	0.00000	0.00000
7	0	.98975	.08971	-1.50000	-.38000	0.00000
8	0	.99314	.04138	-1.77000	-.44000	0.00000
9	0	1.01123	.21371	-.06000	0.00000	0.00000
10	0	1.01228	.24728	.04000	0.00000	0.00000
11	0	.98065	.24853	-2.01000	-.50000	0.00000
12	0	.94895	.38733	-1.32000	-.33000	0.00000
13	0	.94647	.35220	-3.44000	-.86000	0.00000
14	0	.95293	.38946	-1.04000	-.26000	0.00000
15	0	.94772	.34173	-3.76000	-.94000	0.00000
16	0	.94076	.41251	-3.76000	-.94000	0.00000
17	0	.94554	.40069	2.16000	.52000	0.00000
18	1	1.02816	.06154	.26000	.42088	0.00000
19	1	1.04752	.07216	.89000	.72284	0.00000
20	1	1.02884	.28309	.28000	.45105	0.00000
21	1	.98896	.48733	9.08000	1.90160	0.00000

22	1	.98399	.47975	9.23000	1.25890	0.00000
23	2	1.04000	0.00000	-.68390	.89132	0.00000

IV. REFERENCES

[1] T.D. Sterling, "Guidelines for humanizing computer information sytems - a report from Stanley House", Comm. ACM, Vol. 17, 1974, pp. 609-613.

[2] R.G. Hamlet and R.M. Haralick, "Transportable package software", Software Practice & Experience, vol. 10, 1980, pp. 1009-1027.

[3] T. Winograd, "Beyond programming languages", Comm. ACM, vol. 22, 1979, pp. 391-401.

[4] T.S. Dillon, "Rescheduling, constrained participation factors and parameter sensitivity in the optimal power flow problem", IEEE Trans. Power Apparatus and Systems, vol. PAS-100, 1981, pp. 2628-2634.

[5] J.W. Bandler and M.A. El-Kady, "The adjoint network approach to power flow solution and sensitivities of test power systems: data and results", Faculty of Engineering, McMaster University, Hamilton, Canada, Repot SOC-255, 1980.

[6] M.S. Sachdev and S.A. Ibrahim, "A fast approximate technique for outage studies in power system planning and operation", IEEE Trans. Power Apparatus and Systems, vol. PAS-93, 1974, pp. 1133-1142.

[7] J.W. Bandler and M.A. El-Kady, "A new method for computerized solution of power flow equations", IEEE Trans. Power Apparatus and Systems, vol. PAS-101, 1982, pp. 1-10.

[8] "Document SOC-D9: PWRDD, reading data for power systems analysis", Group on Simulation, Optimization and Control, Faculty of Engineering, McMaster University, Hamilton, Canada, June 1982.

[9] "Document SOC-D1: MA28, sparse matrix techniques", Group on Simulation, Optimization and Control, Faculty of Engineering, McMaster University, Hamilton, Canada, December 1981. Origin of the package: Harwell Subroutine Library, AERE, Harwell, Oxfordshire, England.

[10] J.W. Bandler and W.M. Zuberek, "LFLFD - a Fortran implementation of the fast decoupled load flow technique", Faculty of Engineering, McMaster University, Hamilton, Canada, Report SOC-296, 1982.

[11] "Document SOC-D10: PWRDS, reading and preprocessing power systems data", Group on Simulation, Optimization and Control, Faculty of Engineering, McMaster University, Hamilton, Canada, June 1982.

APPENDIX 1

LIST OF DATA NAMES IN POWER SYSTEMS DATA FILES

BUSBASEV	real	bus base voltage
BUSGP	real	bus generated active power
BUSGPQ	complex	bus generated power (BUSGP + jBUSGQ)
BUSGQ	real	bus generated reactive power
BUSGQMAX	real	maximum generated reactive power
BUSGQMIN	real	minimum generated reactive power
BUSLP	real	bus load active power
BUSLPQ	complex	bus load power (BUSLP + jBUSLQ)
BUSLQ	real	bus load reactive power
BUSNAME	character	bus name
BUSNR	integer	bus index
BUSSTL	real	bus static load
BUSTYPE	integer	bus type (0-load, 1-generator, 2-slack)
BUSVARG	real	bus voltage argument (in degrees)
BUSVMOD	real	bus voltage magnitude
COSTBNR	integer	bus index
COSTFA	real	fuel cost factor a
COSTFB	real	fuel cost factor b
COSTFC	real	fuel cost factor c
COSTPMAX	real	maximum generated active power
COSTPMIN	real	minimum generated active power
LINEBASE	real	line base voltage
LINEBINP	integer	line input bus index
LINEBOUT	integer	line output bus index
LINECNR	integer	line circuit number

LINEINPA	complex	line input shunt admittance
LINEINPC	real	line input shunt conductance
LINEINPS	real	line input shunt susceptance
LINEOUTA	complex	line output shunt admittance
LINEOUTC	real	line output shunt conductance
LINEOUTS	real	line output shunt susceptance
LINER	real	line resistance
LINERAT	real	maximum line rating
LINERX	complex	line impedance (LINER + jLINEX)
LINETAP	real	line transformer tap
LINETMAX	real	maximum transformer tap
LINETMIN	real	minimum transformer tap
LINETRT	integer	line transformer type
LINETSH	real	line phase shifter
LINEX	real	line reactance

APPENDIX 2

LISTING OF THE PACKAGE

Subroutine	Number of Lines (source text)	Number of Words (compiled code)	Listing from Page
DD00CL	29	104	53
DD00DD	111	442	53
DD00DF	31	131	55
DD00DN	87	263	55
DD00ES	27	72	57
DD00RF	25	67	57
DD00WF	34	121	57
DD11CN	40	234	58
DD11DR	74	155	59
DD11ER	20	52	60
DD11GS	36	144	60
DD11IN	29	102	61
DD11RN	32	143	61
DD11ST	28	52	62
DD11SW	35	74	62
DD22CN	44	205	63
DD22IN	29	75	63
DD22PR	26	102	64
DD22RN	31	131	64
DD22SR	32	72	65
DD22ST	30	72	65
DD99BC	19	37	66
DD99BF	18	41	66
DD99EC	25	140	67
DD99EF	26	145	67
DD99FC	25	140	67
DD99FF	38	205	68
DD99IC	25	137	68
DD99IF	22	125	69
DD99RL	27	65	69
DD99SF	13	52	70
DD99ST	13	70	70

```
      SUBROUTINE DD00CL                                                 000001
C                                                                       000002
C     THIS SUBROUTINE INITIALIZES THE WORKSPACE *KW* AND *W* AS         000003
C     A LINKED LIST OF 3-ELEMENT CELLS POINTED BY *MFREE*, AND          000004
C     ERASES THE RECORD DEFINITIONS *MREC*.                             000005
C                                                                       000006
C     DATE : 82.08.10 (W.M.ZUBEREK)                                     000007
C                                                                       000008
      DIMENSION KW(450),W(450)                                          000009
      EQUIVALENCE(KW(1),W(1))                                           000010
      COMMON /DD00CM/ LW,KW                                             000011
      COMMON /DD11CM/ LR,MREC(10),RMARK(10),MCHAN(10),MRNR(10)          000012
      COMMON /DD22CM/ MFREE,NAMES,MADDR,NCHAN,KCODE                     000013
      COMMON /DD33CM/ LB,A(10),B(150)                                   000014
      COMMON /DD99CM/ SP,DIG(10)                                        000015
      DATA LW/450/,LR/10/,LB/150/,KCODE/-1/                             000016
      DATA SP/1H /,DIG/1H0,1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9/         000017
      DO 10 I=1,LR                                                      000018
      MCHAN(I)=0                                                        000019
   10 MREC(I)=0                                                         000020
      DO 20 I=1,LW,3                                                    000021
      J=I                                                               000022
   20 KW(I)=I+3                                                         000023
      KW(J)=0                                                           000024
      MFREE=1                                                           000025
      KCODE=0                                                           000026
      NAMES=0                                                           000027
      RETURN                                                            000028
      END                                                               000029
C                                                                       000030
C                                                                       000031
      SUBROUTINE DD00DD (DN,KD,LD,RT,LT,IE)                             000032
      DIMENSION DN(LD),KD(LD),RT(1)                                     000033
C                                                                       000034
C     THIS SUBROUTINE DEFINES THE DATA DESCRIPTORS. A LINKED LIST       000035
C     OF DATA NAMES IS CREATED AND POINTED BY *NAMES*, AND THE          000036
C     ENCODED DATA ATTRIBUTES ARE LINKED WITH THE APPROPRIATE           000037
C     DATA NAME CELLS.                                                  000038
C                                                                       000039
C     DN - REAL VECTOR OF DATA NAMES,                                   000040
C     KD - INTEGER VECTOR OF DATA ATTRIBUTES,                           000041
C     LD - THE LENGTH OF *DN* AND *KD*,                                 000042
C     RT - AUXILIARY REAL VECTOR OF MULTIPLIERS AND DEFAULT VALUES,     000043
C     LT - THE LENGTH OF *RT*,                                          000044
C     IE - RETURN FLAG.                                                 000045
C                                                                       000046
C     DATE : 82.08.16 (W.M.ZUBEREK)                                     000047
C                                                                       000048
      DIMENSION KW(450),W(450)                                          000049
      EQUIVALENCE (KW(1),W(1))                                          000050
      COMMON /DD00CM/ LW,KW                                             000051
      COMMON /DD22CM/ MFREE,NAMES,MADDR,NCHAN,KCODE                     000052
      COMMON /DD33CM/ LB,KA(10),B(150)                                  000053
      DATA ZS/3H0*0/,Z0/1H0/                                            000054
      IE=-1                                                             000055
      IF(KCODE.NE.4) RETURN                                             000056
      IE=-2                                                             000057
      IF(LD.LE.0) RETURN                                                000058
      IE=-3                                                             000059
      MADDR=0                                                           000060
      IF(NAMES.NE.0) CALL DD99RL(NAMES)                                 000061
      DO 60 I=1,10                                                      000062
   60 KA(I)=4                                                           000063
C                                                                       000064
C       CREATE LIST OF DATA NAMES AND LINK DATA ATTRIBUTES              000065
```

```
C                                                                       000066
      DO 10 I=1,LD                                                      000067
      L=MFREE                                                           000068
      IF(L.LE.0) RETURN                                                 000069
      MFREE=KW(L)                                                       000070
      KW(L)=0                                                           000071
      IF(MADDR.EQ.0) NAMES=L                                            000072
      IF(MADDR.NE.0) KW(MADDR)=L                                        000073
      MADDR=L                                                           000074
      W(L+1)=DN(I)                                                      000075
      K=KD(I)                                                           000076
      J=MOD(K,100)                                                      000077
      X=0.0                                                             000078
      IF(J.NE.0.AND.J.LE.LT) X=RT(J)                                    000079
      K=K/100                                                           000080
      J=MOD(K,100)                                                      000081
      Y=0.0                                                             000082
      IF(J.NE.0.AND.J.LE.LT) Y=RT(J)                                    000083
      K=K/100                                                           000084
      J=MOD(K,100)                                                      000085
      K=K/100                                                           000086
      M=MOD(K,100)                                                      000087
      K=K/100                                                           000088
      N=MOD(K,10)                                                       000089
      K=K/10                                                            000090
      K=MOD(K,10)                                                       000091
      IF(K.EQ.0) GOTO 90                                                000092
      DECODE(10,111,KD Z                                                000093
  111 FORMAT(9X,A1)                                                     000094
      IF(M.EQ.0) GOTO 20                                                000095
      IF(M.GT.I) GOTO 90                                                000096
      ZZ=DN(M)                                                          000097
      M=NAMES                                                           000098
   30 IF(M.EQ.0) GOTO 90                                                000099
      IF(ZZ.EQ.W(M+1)) GOTO 40                                          000100
      M=KW(M)                                                           000101
      GOTO 30                                                           000102
   40 M=KW(M+2)                                                         000103
      M=AND(KW(M),255)                                                  000104
      GOTO 25                                                           000105
   20 M=KA(KD                                                           000106
   25 KA(KD=M+J                                                         000107
      K=MFREE                                                           000108
      IF(K.LE.0) RETURN                                                 000109
      MFREE=KW(K)                                                       000110
      KW(L+2)=K                                                         000111
      W(K+1)=Y                                                          000112
      W(K+2)=X                                                          000113
      W(KD=OR(AND(Z,MASK(6)),M,SHIFT(J,8),SHIFT(N,16))                  000114
   10 CONTINUE                                                          000115
C                                                                       000116
C       WRITE DATA DESCRIPTOR RECORDS                                   000117
C                                                                       000118
      I=NAMES                                                           000119
   50 IF(I.EQ.0) GOTO 80                                                000120
      J=KW(I+2)                                                         000121
      X=W(J)                                                            000122
      N2=AND(X,255)                                                     000123
      N3=AND(SHIFT(X,-8),255)                                           000124
      N1=AND(SHIFT(X,-16),15)                                           000125
      CALL DD99FF(B,KA,1,10,W(J+1),IERR)                                000126
      CALL DD99FF(B,KA,11,10,W(J+2),IERR)                               000127
      WRITE(NCHAN,222) Z0,W(J),Z0,N1,N2,N3,W(I+1),(B(KD,K=1,20)         000128
  222 FORMAT(3A1,3I5,2X,A10,5X,20A1)                                    000129
      I=KW(I)                                                           000130
```

```
      GOTO 50                                                           000131
C                                                                       000132
C       WRITE END-OF-SUBFILE RECORD!                                    000133
C                                                                       000134
   80 WRITE(NCHAN,333) ZS                                               000135
  333 FORMAT(A3)                                                        000136
      IE=0                                                              000137
      KCODE=2                                                           000138
      RETURN                                                            000139
   90 IE=-4                                                             000140
      RETURN                                                            000141
      END                                                               000142
C                                                                       000143
C                                                                       000144
      SUBROUTINE DD00DF (LF,NB,NS,NT,DN,IE)                             000145
C                                                                       000146
C     THIS SUBROUTINE DEFINES THE INPUT DATA FILE AND READS THE DATA    000147
C     HEADER RECORD.                                                    000148
C                                                                       000149
C     LF - THE UNIT (CHANNEL) NUMBER OF THE FILE,                       000150
C     NB - FIRST DATA PARAMETER,                                        000151
C     NS - SECOND DATA PARAMETER,                                       000152
C     NT - THIRD DATA PARAMETER,                                        000153
C     DN - INTERNAL FILE NAME,                                          000154
C     IE - RETURN FLAG / INFORMATION CODE.                              000155
C                                                                       000156
C     DATE : 82.08.11 (W.M.ZUBEREK)                                     000157
C                                                                       000158
      COMMON /DD22CM/ MFREE,NAMES,MADDR,NCHAN,KCODE                     000159
      DATA ZS/1H*/,Z0/1H0/                                              000160
      IF(KCODE.LT.0) CALL DD00CL                                        000161
      IE=-1                                                             000162
      IF(LF.LE.0.OR.LF.GT.99) RETURN                                    000163
      IE=-2                                                             000164
      NCHAN=LF                                                          000165
      REWIND LF                                                         000166
   10 READ(LF,111) X,Y,Z,NB,NS,NT,DN,KF,ICD                             000167
  111 FORMAT(3A1,3I5,2X,A10,I5,I10)                                     000168
      IF(EOF(LF).NE.0) RETURN                                           000169
      IF(X.EQ.ZS) GOTO 10                                               000170
      IF(X.NE.Z0.OR.Y.NE.Z0.OR.Z.NE.Z0) RETURN                          000171
      IE=ICD                                                            000172
      KCODE=1                                                           000173
      RETURN                                                            000174
      END                                                               000175
C                                                                       000176
C                                                                       000177
      SUBROUTINE DD00DN (TN,LT,IE)                                      000178
      DIMENSION TN(LT)                                                  000179
C                                                                       000180
C     THIS SUBROUTINE DEFINES ALL THE DATA NAMES WHICH ARE TO BE USED   000181
C     FOR THE INPUT FILE INDICATED BY *DD00DF*, READS THE DATA DESCRIP- 000182
C     TOR RECORDS AND STORES THE DATA DESCRIPTORS THAT CORRESPOND TO    000183
C     THE SUBMITTED DATA NAMES.                                         000184
C                                                                       000185
C     TN - REAL VECTOR OF DATA NAMES,                                   000186
C     LT - THE LENGTH OF *TN*,                                          000187
C     IE - RETURN FLAG.                                                 000188
C                                                                       000189
C     DATE : 82.08.11 (W.M.ZUBEREK)                                     000190
C                                                                       000191
      DIMENSION KW(450),W(450)                                          000192
      EQUIVALENCE (KW(1),W(1))                                          000193
      COMMON /DD00CM/ LW,KW                                             000194
      COMMON /DD22CM/ MFREE,NAMES,MADDR,NCHAN,KCODE                     000195
```

```
      COMMON /DD33CM/ LB,A(10),B(150)                                    000196
      DATA STS/10H**********/,ZS/1H*/,Z0/1H0/                            000197
      IE=-1                                                              000198
      IF(KCODE.LE.0) RETURN                                              000199
      IE=-2                                                              000200
      IF(LT.LE.0) RETURN                                                 000201
      IF(NAMES.EQ.0) GOTO 10                                             000202
      CALL DD99RL(NAMES)                                                 000203
C                                                                        000204
C       CREATE LIST OF DATA NAMES                                        000205
C                                                                        000206
   10 DO 20 I=1,LT                                                       000207
      L=MFREE                                                            000208
      IF(L.LE.0) GOTO 90                                                 000209
      MFREE=KW(L)                                                        000210
      KW(L)=NAMES                                                        000211
      NAMES=L                                                            000212
      W(L+1)=TN(I)                                                       000213
      KW(L+2)=0                                                          000214
   20 CONTINUE                                                           000215
C                                                                        000216
C       READ DATA DESCRIPTORS AND COMPARE DATA NAMES                     000217
C                                                                        000218
      IE=-4                                                              000219
   30 READ(NCHAN,111) X,Y,Z,N1,N2,N3,DN,KE,R,V                           000220
  111 FORMAT(3A1,3I5,2X,A10,I5,2F10.8)                                   000221
      IF(EOF(NCHAN).NE.0) RETURN                                         000222
      IF(X.EQ.ZS) GOTO 30                                                000223
      IF(X.NE.Z0.OR.Z.NE.Z0) RETURN                                      000224
      IF(Y.EQ.ZS) GOTO 60                                                000225
      I=NAMES                                                            000226
   40 IF(I.EQ.0) GOTO 30                                                 000227
      IF(W(I+1).EQ.DN) GOTO 50                                           000228
      I=KW(I)                                                            000229
      GOTO 40                                                            000230
   50 IF(KW(I+2).NE.0) RETURN                                            000231
      IF(N2+N3-1.GT.LB) RETURN                                           000232
C                                                                        000233
C       ENCODE AND STORE DATA ATTRIBUTES                                 000234
C                                                                        000235
      L=MFREE                                                            000236
      IF(L.LE.0) GOTO 90                                                 000237
      MFREE=KW(L)                                                        000238
      KW(I+2)=L                                                          000239
      W(L+1)=R                                                           000240
      W(L+2)=V                                                           000241
      Y=AND(Y,MASK(6))                                                   000242
      W(L)=OR(Y,N2,SHIFT(N3,8),SHIFT(N1,16),SHIFT(AND(KE,1023),20))      000243
      GOTO 30                                                            000244
C                                                                        000245
C       CHECK UNMATCHED DATA NAMES                                       000246
C                                                                        000247
   60 IE=0                                                               000248
      KCODE=2                                                            000249
      I=NAMES                                                            000250
   70 IF(I.EQ.0) RETURN                                                  000251
      IF(KW(I+2).NE.0) GOTO 85                                           000252
      X=W(I+1)                                                           000253
      DO 80 J=1,LT                                                       000254
      IF(X.NE.TN(J)) GOTO 80                                             000255
      TN(J)=STS                                                          000256
      IE=IE+1                                                            000257
      GOTO 85                                                            000258
   80 CONTINUE                                                           000259
   85 I=KW(I)                                                            000260
```

```
      GOTO 70                                                          000261
   90 IE=-3                                                            000262
      RETURN                                                           000263
      END                                                              000264
C                                                                      000265
C                                                                      000266
      SUBROUTINE DD00ES (NR,IE)                                        000267
C                                                                      000268
C     THIS SUBROUTINE CLOSES THE DATA SUBFILE WRITING END-OF-SUBFILE   000269
C     RECORD.                                                          000270
C                                                                      000271
C     NR - INTEGER RECORD IDENTIFIER,                                  000272
C     IE - RETURN FLAG.                                                000273
C                                                                      000274
C     DATE : 82.08.12 (W.M.ZUBEREK)                                    000275
C                                                                      000276
      COMMON /DD11CM/ LR,MREC(10),RMARK(10),MCHAN(10),MRNR(10)         000277
      COMMON /DD22CM/ MFREE,NAMES,MADDR,NCHAN,KCODE                    000278
      DATA Z0/2H*0/                                                    000279
      IE=-2                                                            000280
      IF(NR.LE.0.OR.NR.GT.LR) RETURN                                   000281
      IE=-1                                                            000282
      IF(KCODE.LE.0) RETURN                                            000283
      LCH=MCHAN(NR)                                                    000284
      IF(MREC(NR).LE.0.OR.LCH.LE.0) RETURN                             000285
      IE=-3                                                            000286
      IF(MREC(NR).LE.0) RETURN                                         000287
      IE=MRNR(NR)                                                      000288
      MRNR(NR)=-1                                                      000289
      WRITE(LCH,111) RMARK(NR),Z0                                      000290
  111 FORMAT(A1,A2)                                                    000291
      RETURN                                                           000292
      END                                                              000293
C                                                                      000294
C                                                                      000295
      SUBROUTINE DD00RF (NF,IE)                                        000296
C                                                                      000297
C     THIS SUBROUTINE RESETS THE DATA FILE.                            000298
C                                                                      000299
C     NF - THE UNIT (CHANNEL) NUMBER OF THE FILE,                      000300
C     IE - RETURN FLAG.                                                000301
C                                                                      000302
C     DATE : 82.08.12 (W.M.ZUBEREK)                                    000303
C                                                                      000304
      LOGICAL FILE                                                     000305
      COMMON /DD11CM/ LR,MREC(10),RMARK(10),MCHAN(10),MRNR(10)         000306
      COMMON /DD22CM/ MFREE,NAMES,MADDR,NCHAN,KCODE                    000307
      IF(KCODE.LT.0) CALL DD00CL                                       000308
      IE=-1                                                            000309
      IF(NF.LE.0.OR.NF.GT.99) RETURN                                   000310
      FILE=.FALSE.                                                     000311
      DO 10 I=1,LR                                                     000312
      IF(NF.NE.MCHAN(I)) GOTO 10                                       000313
      FILE=.TRUE.                                                      000314
      MRNR(I)=0                                                        000315
   10 CONTINUE                                                         000316
      IF(FILE) REWIND NF                                               000317
      IE=0                                                             000318
      RETURN                                                           000319
      END                                                              000320
C                                                                      000321
C                                                                      000322
      SUBROUTINE DD00WF (LD,NB,NS,NT,DN,IC,IE)                         000323
C                                                                      000324
C     THIS SUBROUTINE DEFINES THE OUTPUT DATA FILE AND WRITES THE      000325
```

```
C     DATA HEADER RECORD.                                                    000326
C                                                                            000327
C     LD - THE UNIT (CHANNEL) NUMBER OF THE FILE,                            000328
C     NB - FIRST DATA PARAMETER,                                             000329
C     NS - SECOND DATA PARAMETER,                                            000330
C     NT - THIRD DATA PARAMETER,                                             000331
C     DN - INTERNAL FILE NAME,                                               000332
C     IC - INFORMATION CODE,                                                 000333
C     IE - RETURN FLAG.                                                      000334
C                                                                            000335
C     DATE : 82.08.16 (W.M.ZUBEREK)                                          000336
C                                                                            000337
      DIMENSION KW(450),W(450)                                               000338
      EQUIVALENCE (KW(1),W(1))                                               000339
      COMMON /DD00CM/ LW,KW(1)                                               000340
      COMMON /DD22CM/ MFREE,NAMES,MADDR,NCHAN,KCODE                          000341
      COMMON /DD33CM/ LB,A(10),B(150)                                        000342
      DATA X/3H000/,Y/3H0*0/                                                 000343
      IF(KCODE.LT.0) CALL DD00CL                                             000344
      IE=-1                                                                  000345
      IF(LD.LE.0.OR.LD.GT.99) RETURN                                         000346
      NCHAN=LD                                                               000347
      REWIND LD                                                              000348
      CALL DATE(DAT)                                                         000349
      CALL TIME(TIM)                                                         000350
      WRITE(LD,111) X,NB,NS,NT,DN,IC,DAT,TIM                                 000351
  111 FORMAT(A3,3I5,2X,A10,5X,I10,10H (V:82.08),2A10)                        000352
      IE=0                                                                   000353
      KCODE=4                                                                000354
      RETURN                                                                 000355
      END                                                                    000356
C                                                                            000357
C                                                                            000358
      SUBROUTINE DD11CM (Z,IE)                                               000359
      COMPLEX Z                                                              000360
C                                                                            000361
C     THIS SUBROUTINE GETS A COMPLEX VALUE FROM THE BUFFER.                  000362
C                                                                            000363
C     Z  - COMPLEX VARIABLE,                                                 000364
C     IE - RETURN FLAG.                                                      000365
C                                                                            000366
C     DATE : 82.08.17 (W.M.ZUBEREK)                                          000367
C                                                                            000368
      DIMENSION KW(450),W(450)                                               000369
      EQUIVALENCE (KW(1),W(1))                                               000370
      COMMON /DD00CM/ LW,KW                                                  000371
      COMMON /DD22CM/ MFREE,NAMES,MADDR,NCHAN,KCODE                          000372
      COMMON /DD33CM/ LB,A(10),B(150)                                        000373
      IE=-1                                                                  000374
      IF(KCODE.NE.5.OR.MADDR.LE.0) RETURN                                    000375
      IE=-2                                                                  000376
      K=MADDR                                                                000377
      MADDR=KW(K)                                                            000378
      J=KW(K+2)                                                              000379
      M=KW(J)                                                                000380
      K=AND(SHIFT(M,-16),15)                                                 000381
      I=AND(M,255)                                                           000382
      L=AND(SHIFT(M,-8),255)                                                 000383
      IF(K.EQ.3) CALL DD99FC(B,A,I,L,X,IE)                                   000384
      IF(K.EQ.5) CALL DD99EC(B,A,I,L,X,IE)                                   000385
      IF(K.EQ.8) CALL DD99BC(B,I,L,X,IE)                                     000386
      IF(IE.LT.0) RETURN                                                     000387
      IF(IE.EQ.0) X=X*W(J+1)                                                 000388
      IF(IE.EQ.1) X=W(J+2)                                                   000389
      IF(K.EQ.3) CALL DD99FC(B,A,I+L,L,Y,IE)                                 000390
```

```
      IF(K.EQ.5) CALL DD99EC(B,A,I+L,L,Y,IE)                            000391
      IF(K.EQ.8) CALL DD99BC(B,I+L,L,Y,IE)                              000392
      IF(IE.EQ.0) Y=Y*W(J+1)                                            000393
      IF(IE.EQ.1) Y=W(J+2)                                              000394
      IF(IE.LT.0) RETURN                                                000395
      Z=CMPLX(X,Y)                                                      000396
      RETURN                                                            000397
      END                                                               000398
C                                                                       000399
C                                                                       000400
      SUBROUTINE DD11DR (NR,TN,LT,IE)                                   000401
      DIMENSION TN(LT)                                                  000402
C                                                                       000403
C     THIS SUBROUTINE DEFINES THE LOGICAL DATA RECORD AS A SEQUENCE OF  000404
C     DATA NAMES PROVIDED ALL THE DATA NAMES HAVE BEEN DEFINED BY       000405
C     *DD00DN*. A LINKED LIST CORRESPONDING TO THE RECORD STRUCTURE IS  000406
C     CREATED AND POINTED BY *MREC(NR)*.                                000407
C                                                                       000408
C     NR - INTEGER RECORD IDENTIFIER,                                   000409
C     TN - REAL VECTOR OF DATA NAMES,                                   000410
C     LT - THE LENGTH OF *TN*,                                          000411
C     IE - RETURN FLAG.                                                 000412
C                                                                       000413
C     DATE : 82.08.11 (W.M.ZUBEREK)                                     000414
C                                                                       000415
      DIMENSION KW(450),W(450)                                          000416
      EQUIVALENCE (KW(1),W(1))                                          000417
      COMMON /DD00CM/ LW,KW                                             000418
      COMMON /DD11CM/ LR,MREC(10),RMARK(10),MCHAN(10),MRNR(10)          000419
      COMMON /DD22CM/ MFREE,NAMES,MADDR,NCHAN,KCODE                     000420
      DATA STS/10H**********/,COL/1H:/                                  000421
      IE=-1                                                             000422
      IF(KCODE.NE.2) RETURN                                             000423
      IE=-2                                                             000424
      IF(LT.LE.0) RETURN                                                000425
      IE=-4                                                             000426
      IF(NR.LE.0.OR.NR.GT.LR) RETURN                                    000427
      IE=-5                                                             000428
      IF(MREC(NR).NE.0) RETURN                                          000429
      MCHAN(NR)=NCHAN                                                   000430
      MRNR(NR)=0                                                        000431
      RMARK(NR)=STS                                                     000432
      IE=0                                                              000433
C                                                                       000434
C       RECORD STRUCTURE IS ANALYZED FROM THE END                       000435
C                                                                       000436
      DO 40 I=1,LT                                                      000437
      J=LT+1-I                                                          000438
      X=TN(J)                                                           000439
C                                                                       000440
C       SEARCH THE LIST OF DATA NAMES                                   000441
C                                                                       000442
      K=NAMES                                                           000443
   10 IF(K.LE.0) GOTO 30                                                000444
      IF(X.EQ.W(K+1)) GOTO 20                                           000445
      K=KW(K)                                                           000446
      GOTO 10                                                           000447
   20 L=MFREE                                                           000448
      IF(L.LE.0) GOTO 90                                                000449
      MFREE=KW(L)                                                       000450
      KW(L)=MREC(NR)                                                    000451
      MREC(NR)=L                                                        000452
      KW(L+1)=NR                                                        000453
C                                                                       000454
C       INCREASE THE REFERENCE COUNT                                    000455
```

```
C                                                                       000456
      X=SHIFT(63,30)                                                    000457
      K=KW(K+2)                                                         000458
      KW(L+2)=K                                                         000459
      N=AND(W(KD),X)+SHIFT(1,30)                                        000460
      W(KD)=OR(AND(W(KD),COMPL(XD)),N)                                  000461
      X=OR(AND(W(KD),MASK(6)),COL)                                      000462
C                                                                       000463
C       CHECK DATA CONSISTENCY                                          000464
C                                                                       000465
      IF(RMARK(NR).EQ.STS) RMARK(NR)=X                                  000466
      IF(RMARK(NR).EQ.X) GOTO 40                                        000467
   30 TN(J)=STS                                                         000468
      IE=-6                                                             000469
   40 CONTINUE                                                          000470
      RETURN                                                            000471
   90 IE=-3                                                             000472
      RETURN                                                            000473
      END                                                               000474
C                                                                       000475
C                                                                       000476
      SUBROUTINE DD11ER (NR,IE)                                         000477
C                                                                       000478
C     THIS SUBROUTINE ERASES THE LOGICAL RECORD DEFINITION.             000479
C                                                                       000480
C     NR - INTEGER RECORD IDENTIFIER,                                   000481
C     IE - RETURN FLAG.                                                 000482
C                                                                       000483
C     DATA : 82.07.30 (W.M.ZUBEREK)                                     000484
C                                                                       000485
      COMMON /DD11CM/ LR,MREC(10),RMARK(10),MCHAN(10),MRNR(10)          000486
      COMMON /DD22CM/ MFREE,NAMES,NADDR,NCHAN,KCODE                     000487
      IE=-1                                                             000488
      IF(KCODE.LE.0) RETURN                                             000489
      IE=-2                                                             000490
      IF(NR.LE.0.OR.NR.GT.LR) RETURN                                    000491
      IF(MREC(NR).LE.0) RETURN                                          000492
      CALL DD99RL(MREC(NR))                                             000493
      IE=0                                                              000494
      RETURN                                                            000495
      END                                                               000496
C                                                                       000497
C                                                                       000498
      SUBROUTINE DD11GS (NR,IE)                                         000499
C                                                                       000500
C     THIS SUBROUTINE READS CONSECUTIVE PHYSICAL RECORD CORRESPONDING   000501
C     TO THE INDICATED LOGICAL DATA RECORD AND INITIALIZES THE RECORD.  000502
C                                                                       000503
C     NR - INTEGER RECORD IDENTIFIER,                                   000504
C     IE - RETURN FLAG.                                                 000505
C                                                                       000506
C     DATE : 82.08.11 (W.M.ZUBEREK)                                     000507
C                                                                       000508
      COMMON /DD11CM/ LR,MREC(10),RMARK(10),MCHAN(10),MRNR(10)          000509
      COMMON /DD22CM/ MFREE,NAMES,MADDR,NCHAN,KCODE                     000510
      COMMON /DD33CM/ LB,A(10),B(150)                                   000511
      DATA ZS/1H*/                                                      000512
      IE=-2                                                             000513
      IF(NR.LE.0.OR.NR.GT.LR) RETURN                                    000514
      IE=-1                                                             000515
      IF(KCODE.LE.0.OR.MREC(NR).LE.0) RETURN                            000516
      LCH=MCHAN(NR)                                                     000517
      IE=-3                                                             000518
   10 READ(LCH,111) (B(I),I=1,LB)                                       000519
  111 FORMAT(150A1)                                                     000520
```

```
      IF(EOF(LCH).NE.0) RETURN                                          000521
      IF(B(1).EQ.ZS) GOTO 10                                            000522
      IF(B(1).NE.RMARK(NR)) GOTO 10                                     000523
      IF(B(2).EQ.ZS) GOTO 90                                            000524
      IE=MRNR(NR)+1                                                     000525
      MRNR(NR)=IE                                                       000526
      MADDR=MREC(NR)                                                    000527
      KCODE=5                                                           000528
      RETURN                                                            000529
   90 IE=0                                                              000530
      MRNR(NR)=-1                                                       000531
      KCODE=3                                                           000532
      RETURN                                                            000533
      END                                                               000534
C                                                                       000535
C                                                                       000536
      SUBROUTINE DD11IN (N,IE)                                          000537
C                                                                       000538
C     THIS SUBROUTINE GETS AN INTEGER VALUE.                            000539
C                                                                       000540
C     N  - INTEGER VARIABLE,                                            000541
C     IE - RETURN FLAG.                                                 000542
C                                                                       000543
C     DATE : 82.07.20 (W.M.ZUBEREK)                                     000544
C                                                                       000545
      DIMENSION KW(450),W(450)                                          000546
      EQUIVALENCE (KW(1),W(1))                                          000547
      COMMON /DD00CM/ LW,KW                                             000548
      COMMON /DD22CM/ MFREE,NAMES,MADDR,NCHAN,KCODE                     000549
      COMMON /DD33CM/ LB,A(10),B(150)                                   000550
      IE=-1                                                             000551
      IF(KCODE.NE.5.OR.MADDR.LE.0) RETURN                               000552
      IE=-2                                                             000553
      K=MADDR                                                           000554
      MADDR=KW(K)                                                       000555
      J=KW(K+2)                                                         000556
      M=KW(J)                                                           000557
      K=AND(SHIFT(M,-16),15)                                            000558
      IF(K.NE.1) RETURN                                                 000559
      I=AND(M,255)                                                      000560
      L=AND(SHIFT(M,-8),255)                                            000561
      CALL DD99IC(B,A,I,L,N,IE)                                         000562
      IF(IE.EQ.1) N=IFIX(W(J+2))                                        000563
      RETURN                                                            000564
      END                                                               000565
C                                                                       000566
C                                                                       000567
      SUBROUTINE DD11RN (X,IE)                                          000568
C                                                                       000569
C     THIS SUBROUTINE GETS A REAL VALUE FROM THE BUFFER.                000570
C                                                                       000571
C     X  - REAL VARIABLE,                                               000572
C     IE - RETURN FLAG.                                                 000573
C                                                                       000574
C     DATE : 82.08.17 (W.M.ZUBEREK)                                     000575
C                                                                       000576
      DIMENSION KW(450),W(450)                                          000577
      EQUIVALENCE (KW(1),W(1))                                          000578
      COMMON /DD00CM/ LW,KW                                             000579
      COMMON /DD22CM/ MFREE,NAMES,MADDR,NCHAN,KCODE                     000580
      COMMON /DD33CM/ LB,A(10),B(150)                                   000581
      IE=-1                                                             000582
      IF(KCODE.NE.5.OR.MADDR.LE.0) RETURN                               000583
      IE=-2                                                             000584
      K=MADDR                                                           000585
```

```
      MADDR=KW(K)                                                       000586
      J=KW(K+2)                                                         000587
      M=KW(J)                                                           000588
      K=AND(SHIFT(M,-16),15)                                            000589
      I=AND(M,255)                                                      000590
      L=AND(SHIFT(M,-8),255)                                            000591
      IF(K.EQ.2) CALL DD99FC(B,A,I,L,X,IE)                              000592
      IF(K.EQ.4) CALL DD99EC(B,A,I,L,X,IE)                              000593
      IF(K.EQ.7) CALL DD99BC(B,I,L,X,IE)                                000594
      IF(IE.LT.0) RETURN                                                000595
      IF(IE.EQ.0) X=X*W(J+1)                                            000596
      IF(IE.EQ.1) X=W(J+2)                                              000597
      RETURN                                                            000598
      END                                                               000599
C                                                                       000600
C                                                                       000601
      SUBROUTINE DD11ST (S,IE)                                          000602
      DIMENSION S(1)                                                    000603
C                                                                       000604
C     THIS SUBROUTINE GETS A CHARACTER STRING FROM THE BUFFER.          000605
C                                                                       000606
C     S  - REAL VARIABLE OR ARRAY,                                      000607
C     IE - RETURN FLAG.                                                 000608
C                                                                       000609
C     DATE : 82.08.11 (W.M.ZUBEREK)                                     000610
C                                                                       000611
      DIMENSION KW(450),W(450)                                          000612
      EQUIVALENCE (KW(1),W(1))                                          000613
      COMMON /DD00CM/ LW,KW                                             000614
      COMMON /DD22CM/ MFREE,NAMES,MADDR,NCHAN,KCODE                     000615
      COMMON /DD33CM/ LB,A(10),B(150)                                   000616
      IE=-1                                                             000617
      IF(KCODE.NE.5.OR.MADDR.LE.0) RETURN                               000618
      IE=-2                                                             000619
      K=MADDR                                                           000620
      MADDR=KW(K)                                                       000621
      J=KW(K+2)                                                         000622
      M=KW(J)                                                           000623
      K=AND(SHIFT(M,-16),15)                                            000624
      I=AND(M,255)                                                      000625
      L=AND(SHIFT(M,-8),255)                                            000626
      IF(K.EQ.9) CALL DD99ST(B,I,L,S,IE)                                000627
      RETURN                                                            000628
      END                                                               000629
C                                                                       000630
C                                                                       000631
      SUBROUTINE DD11SW (NR,IE)                                         000632
C                                                                       000633
C     THIS SUBROUTINE REPLACES THE CURRENT LOGICAL RECORD STRUCTURE     000634
C     BY AN ALTERNATIVE ONE PROVIDED BOTH THE RECORDS HAVE IDENTICAL    000635
C     INITIAL PARTS.                                                    000636
C                                                                       000637
C     NR - INTEGER RECORD IDENTIFIER,                                   000638
C     IE - RETURN FLAG.                                                 000639
C                                                                       000640
C     DATE : 82.07.30 (W.M.ZUBEREK)                                     000641
C                                                                       000642
      DIMENSION KW(450),W(450)                                          000643
      EQUIVALENCE (KW(1),W(1))                                          000644
      COMMON /DD00CM/ LW,KW                                             000645
      COMMON /DD11CM/ LR,MREC(10),RMARK(10),MCHAN(10),MRNR(10)          000646
      COMMON /DD22CM/ MFREE,NAMES,MADDR,NCHAN,KCODE                     000647
      IE=-1                                                             000648
      IF(KCODE.NE.5.OR.MADDR.LE.0) RETURN                               000649
      IE=-2                                                             000650
```

```
      IF(NR.LE.0.OR.NR.GT.LR) RETURN                                    000651
      IF(MREC(NR).LE.0) RETURN                                          000652
      IE=-3                                                             000653
      K=KW(MADDR+1)                                                     000654
      K=MREC(K)                                                         000655
      L=MREC(NR)                                                        000656
   10 IF(KW(K).EQ.MADDR) GOTO 20                                        000657
      IF(KW(K+2).NE.KW(L+2)) RETURN                                     000658
      K=KW(K)                                                           000659
      L=KW(L)                                                           000660
      IF(K.LE.0.OR.L.LE.0) RETURN                                       000661
      GOTO 10                                                           000662
   20 IE=0                                                              000663
      MADDR=KW(L)                                                       000664
      RETURN                                                            000665
      END                                                               000666
C                                                                       000667
C                                                                       000668
      SUBROUTINE DD22CN (Z,IE)                                          000669
      COMPLEX Z                                                         000670
C                                                                       000671
C     THIS SUBROUTINE PUTS A COMPLEX VALUE INTO THE BUFFER.             000672
C                                                                       000673
C     Z  - A COMPLEX EXPRESSION,                                        000674
C     IE - RETURN FLAG.                                                 000675
C                                                                       000676
C     DATE : 82.08.17 (W.M.ZUBEREK)                                     000677
C                                                                       000678
      DIMENSION KW(450),W(450)                                          000679
      EQUIVALENCE (KW(1),W(1))                                          000680
      COMMON /DD00CM/ LW,KW                                             000681
      COMMON /DD22CM/ MFREE,NAMES,MADDR,NCHAN,KCODE                     000682
      COMMON /DD33CM/ LB,RB(10),B(150)                                  000683
      IE=-1                                                             000684
      IF(KCODE.NE.6.OR.MADDR.LE.0) RETURN                               000685
      IE=-2                                                             000686
      K=MADDR                                                           000687
      MADDR=KW(K)                                                       000688
      J=KW(K+2)                                                         000689
      X=W(J+1)                                                          000690
      Z1=REAL(Z)/X                                                      000691
      Z2=AIMAG(Z)/X                                                     000692
      Y=W(J)                                                            000693
      I=AND(Y,255)                                                      000694
      L=AND(SHIFT(Y,-8),255)                                            000695
      M=AND(SHIFT(Y,-16),15)                                            000696
      J=I+L-1                                                           000697
      IF(M.NE.8) J=J+L                                                  000698
      NCHAN=MAX0(NCHAN,J)                                               000699
      IF(M.NE.3) GOTO 10                                                000700
      CALL DD99FF(B,RB,I,L,Z1,IE)                                       000701
      CALL DD99FF(B,RB,I+L,L,Z2,IE)                                     000702
      GOTO 90                                                           000703
   10 IF(M.NE.5) GOTO 20                                                000704
      CALL DD99EF(B,RB,I,L,Z1,IE)                                       000705
      CALL DD99EF(B,RB,I+L,L,Z2,IE)                                     000706
      GOTO 90                                                           000707
   20 IF(M.NE.8) RETURN                                                 000708
      CALL DD99BF(B,I,L,Z1,IE)                                          000709
      CALL DD99BF(B,I+L,L,Z2,IE)                                        000710
   90 RETURN                                                            000711
      END                                                               000712
C                                                                       000713
C                                                                       000714
      SUBROUTINE DD22IN (N,IE)                                          000715
```

```
C                                                                       000716
C     THIS SUBROUTINE PUTS AN INTEGER VALUE INTO THE BUFFER.            000717
C                                                                       000718
C     N  - AN INTEGER EXPRESSION,                                       000719
C     IE - RETURN FLAG.                                                 000720
C                                                                       000721
C     DATE : 82.08.16 (W.M.ZUBEREK)                                     000722
C                                                                       000723
      DIMENSION KW(450),W(450)                                          000724
      EQUIVALENCE (KW(1),W(1))                                          000725
      COMMON /DD00CM/ LW,KW                                             000726
      COMMON /DD22CM/ MFREE,NAMES,MADDR,NCHAN,KCODE                     000727
      COMMON /DD33CM/ LB,RB(10),B(150)                                  000728
      IE=-1                                                             000729
      IF(KCODE.NE.6.OR.MADDR.LE.0) RETURN                               000730
      IE=-2                                                             000731
      K=MADDR                                                           000732
      MADDR=KW(K)                                                       000733
      J=KW(K+2)                                                         000734
      X=W(J)                                                            000735
      K=AND(SHIFT(X,-16),15)                                            000736
      I=AND(X,255)                                                      000737
      L=AND(SHIFT(X,-8),255)                                            000738
      IF(K.NE.1) RETURN                                                 000739
      NCHAN=MAX0(NCHAN,I+L-1)                                           000740
      CALL DD99IF(B,RB,I,L,N,IE)                                        000741
      RETURN                                                            000742
      END                                                               000743
C                                                                       000744
C                                                                       000745
      SUBROUTINE DD22PR (NR,IE)                                         000746
C                                                                       000747
C     THIS SUBROUTINE TRANSFERS THE RECORD TO THE OUTPUT FILE.          000748
C                                                                       000749
C     NR - INTEGER RECORD IDENTIFIER,                                   000750
C     IE - RETURN FLAG.                                                 000751
C                                                                       000752
C     DATE : 82.08.17 (W.M.ZUBEREK)                                     000753
C                                                                       000754
      COMMON /DD11CM/ LR,MREC(10),RMARK(10),MCHAN(10),MRNR(10)          000755
      COMMON /DD22CM/ MFREE,NAMES,MADDR,NCHAN,KCODE                     000756
      COMMON /DD33CM/ LB,RB(10),B(150)                                  000757
      IE=-2                                                             000758
      IF(NR.LE.0.OR.NR.GT.LR) RETURN                                    000759
      IE=-1                                                             000760
      IF(KCODE.NE.6.OR.MRNR(NR).LT.0) RETURN                            000761
      LCH=MCHAN(NR)                                                     000762
      IF(MREC(NR).LE.0.OR.LCH.LE.0) RETURN                              000763
      IE=-3                                                             000764
      IF(B(1).NE.RMARK(NR)) RETURN                                      000765
      WRITE(LCH,111) (B(I),I=1,NCHAN)                                   000766
  111 FORMAT(150A1)                                                     000767
      IE=MRNR(NR)+1                                                     000768
      MRNR(NR)=IE                                                       000769
      RETURN                                                            000770
      END                                                               000771
C                                                                       000772
C                                                                       000773
      SUBROUTINE DD22RN (X,IE)                                          000774
C                                                                       000775
C     THIS SUBROUTINE PUTS A REAL VALUE INTO THE BUFFER.                000776
C                                                                       000777
C     X  - A REAL EXPRESSION,                                           000778
C     IE - RETURN FLAG.                                                 000779
C                                                                       000780
```

```
C     DATE : 82.08.17 (W.M.ZUBEREK)                                    000781
C                                                                      000782
      DIMENSION KW(450),W(450)                                         000783
      EQUIVALENCE (KW(1),W(1))                                         000784
      COMMON /DD99CM/ LW,KW                                            000785
      COMMON /DD22CM/ MFREE,NAMES,MADDR,NCHAN,KCODE                    000786
      COMMON /DD33CM/ LB,RB(10),B(150)                                 000787
      IE=-1                                                            000788
      IF(KCODE.NE.6.OR.MADDR.LE.0) RETURN                              000789
      IE=-2                                                            000790
      K=MADDR                                                          000791
      MADDR=KW(K)                                                      000792
      J=KW(K+2)                                                        000793
      Z=K/W(J+1)                                                       000794
      Y=W(J)                                                           000795
      I=AND(Y,255)                                                     000796
      L=AND(SHIFT(Y,-8),255)                                           000797
      M=AND(SHIFT(Y,-16),15)                                           000798
      NCHAN=MAX0(NCHAN,I+L-1)                                          000799
      IF(M.EQ.2) CALL DD99FF(B,RB,I,L,Z,IE)                            000800
      IF(M.EQ.4) CALL DD99EF(B,RB,I,L,Z,IE)                            000801
      IF(M.EQ.7) CALL DD99BF(B,I,L,Z,IE)                               000802
      RETURN                                                           000803
      END                                                              000804
C                                                                      000805
C                                                                      000806
      SUBROUTINE DD22SR (NR,IE)                                        000807
C                                                                      000808
C     THIS SUBROUTINE INITIALIZES THE NEW OUTPUT RECORD.               000809
C                                                                      000810
C     NR - INTEGER RECORD IDENTIFIER,                                  000811
C     IE - RETURN FLAG.                                                000812
C                                                                      000813
C     DATE : 82.08.12 (W.M.ZUBEREK)                                    000814
C                                                                      000815
      COMMON /DD11CM/ LR,MREC(10),RMARK(10),MCHAN(10),MRNR(10)         000816
      COMMON /DD22CM/ MFREE,MDESC,MADDR,NCHAN,KCODE                    000817
      COMMON /DD33CM/ LB,RB(10),B(150)                                 000818
      DATA SP/1H /,Z0/1E0/                                             000819
      IE=-2                                                            000820
      IF(NR.LE.0.OR.NR.GT.LR) RETURN                                   000821
      IE=-1                                                            000822
      IF(KCODE.LE.2) RETURN                                            000823
      K=MREC(NR)                                                       000824
      IF(K.LE.0) RETURN                                                000825
      MADDR=K                                                          000826
      IE=-3                                                            000827
      IF(MRNR(NR).LT.0) RETURN                                         000828
      DO 10 I=4,150                                                    000829
   10 B(I)=SP                                                          000830
      B(1)=RMARK(NR)                                                   000831
      B(2)=Z0                                                          000832
      B(3)=Z0                                                          000833
      NCHAN=3                                                          000834
      IE=0                                                             000835
      KCODE=6                                                          000836
      RETURN                                                           000837
      END                                                              000838
C                                                                      000839
C                                                                      000840
      SUBROUTINE DD22ST (S,IE)                                         000841
      DIMENSION S(1)                                                   000842
C                                                                      000843
C     THIS SUBROUTINE PUTS A CHARACTER STRING INTO THE BUFFER.         000844
C                                                                      000845
```

```
C     S  - REAL VARIABLE OR ARRAY,                                      000846
C     IE - RETURN FLAG.                                                 000847
C                                                                       000848
C     DATE : 82.08.16 (W.M.ZUBEREK)                                     000849
C                                                                       000850
      DIMENSION KW(450),W(450)                                          000851
      EQUIVALENCE (KW(1),W(1))                                          000852
      COMMON /DD00CM/ LW,KW                                             000853
      COMMON /DD22CM/ MFREE,NAMES,MADDR,NCHAN,KCODE                     000854
      COMMON /DD33CM/ LB,A(10),B(150)                                   000855
      IE=-1                                                             000856
      IF(KCODE.NE.6.OR.MADDR.LE.0) RETURN                               000857
      IE=-2                                                             000858
      K=MADDR                                                           000859
      MADDR=KW(K)                                                       000860
      J=KW(K+2)                                                         000861
      X=W(J)                                                            000862
      K=AND(SHIFT(X,-16),15)                                            000863
      I=AND(X,255)                                                      000864
      L=AND(SHIFT(X,-8),255)                                            000865
      IF(K.NE.9) RETURN                                                 000866
      NCHAN=MAX0(NCHAN,I+L-1)                                           000867
      CALL DD99SF(B,I,L,S,IE)                                           000868
      RETURN                                                            000869
      END                                                               000870
C                                                                       000871
C                                                                       000872
      SUBROUTINE DD99BC (R,K,L,X,IE)                                    000873
      DIMENSION R(1)                                                    000874
C                                                                       000875
C     THIS SUBROUTINE PERFORMS INPUT BINARY CONVERSION.                 000876
C                                                                       000877
C     DATE : 82.08.17 (W.M.ZUBEREK)                                     000878
C                                                                       000879
      X=0.0                                                             000880
      Y=MASK(5)                                                         000881
      Z=COMPL(Y)                                                        000882
      M=MIN0(12,L)                                                      000883
      J=K+M                                                             000884
      DO 10 I=1,M                                                       000885
      J=J-1                                                             000886
      X=OR(AND(R(J),Y),AND(SHIFT(X,-5),Z))                              000887
   10 CONTINUE                                                          000888
      IE=0                                                              000889
      RETURN                                                            000890
      END                                                               000891
C                                                                       000892
C                                                                       000893
      SUBROUTINE DD99BF (R,X,L,K,IE)                                    000894
      DIMENSION R(1)                                                    000895
C                                                                       000896
C     THIS SUBROUTINE PERFORMS OUTPUT BINARY CONVERSION.                000897
C                                                                       000898
C     DATE : 82.08.17 (W.M.ZUBEREK)                                     000899
C                                                                       000900
      DATA Z/1HA/                                                       000901
      Y=X                                                               000902
      J=K                                                               000903
      DO 10 I=1,L                                                       000904
      R(J)=OR(AND(Y,MASK(5)),Z)                                         000905
      J=J+1                                                             000906
      Y=SHIFT(Y,5)                                                      000907
   10 CONTINUE                                                          000908
      IE=0                                                              000909
      RETURN                                                            000910
```

```
      END                                                               000911
C                                                                       000912
C                                                                       000913
      SUBROUTINE DD99EC (B,A,K,L,X,IE)                                  000914
      DIMENSION A(1),B(1)                                               000915
C                                                                       000916
C     THIS SUBROUTINE PERFORMS INPUT "E" CONVERSION.                    000917
C                                                                       000918
C     DATE : 82.07.26 (W.M.ZUBEREKO                                     000919
C                                                                       000920
      COMMON /DD99CM/ SP,D(10)                                          000921
      DATA T1,T2/2H(E,3H.0)/                                            000922
      J=K+L-1                                                           000923
      DO 10 I=K,J                                                       000924
      IF(B(I).NE.SP) GOTO 20                                            000925
   10 CONTINUE                                                          000926
      IE=1                                                              000927
      RETURN                                                            000928
   20 IE=0                                                              000929
      J2=MOD(L,10)+1                                                    000930
      J1=L/10+1                                                         000931
      ENCODE(7,111,F) T1,D(J1),D(J2),T2                                 000932
  111 FORMAT(A2,2A1,A3)                                                 000933
      ENCODE(L,222,A) (B(I),I=K,J)                                      000934
  222 FORMAT(150A1)                                                     000935
      DECODE(L,F,A) X                                                   000936
      RETURN                                                            000937
      END                                                               000938
C                                                                       000939
C                                                                       000940
      SUBROUTINE DD99EF (B,A,K,L,X,IE)                                  000941
      DIMENSION B(1),A(1)                                               000942
C                                                                       000943
C     THIS SUBROUTINE PERFORMS OUTPUT "E" CONVERSION.                   000944
C                                                                       000945
C     DATE : 82.07.25 (W.M.ZUBEREKO                                     000946
C                                                                       000947
      COMMON /DD99CM/ SPACE,D(10)                                       000948
      DATA T1,T2,T3/4H(1PE,1H.,1H)/                                     000949
      IE=-1                                                             000950
      IF(L.GE.100) RETURN                                               000951
      LF=L-6                                                            000952
      IF(LF.LE.0) RETURN                                                000953
      J2=MOD(L,10)+1                                                    000954
      J1=L/10+1                                                         000955
      J4=MOD(LF,10)+1                                                   000956
      J3=LF/10+1                                                        000957
      ENCODE(10,111,F) T1,D(J1),D(J2),T2,D(J3),D(J4),T3                 000958
  111 FORMAT(A4,6A1)                                                    000959
      J=K+L-1                                                           000960
      ENCODE(L,F,A) X                                                   000961
      DECODE(L,222,A) (B(I),I=K,J)                                      000962
  222 FORMAT(150A1)                                                     000963
      IE=0                                                              000964
      RETURN                                                            000965
      END                                                               000966
C                                                                       000967
C                                                                       000968
      SUBROUTINE DD99FC (B,A,K,L,X,IE)                                  000969
      DIMENSION A(1),B(1)                                               000970
C                                                                       000971
C     THIS SUBROUTINE PERFORMS INPUT "F" CONVERSION.                    000972
C                                                                       000973
C     DATE : 82.07.25 (W.M.ZUBEREKO                                     000974
C                                                                       000975
```

```
      COMMON /DD99CM/ SP,D(10)                                          000976
      DATA T1,T2/2H(F,3H.0)/                                            000977
      J=K+L-1                                                           000978
      DO 10 I=K,J                                                       000979
      IF(B(I).NE.SP) GOTO 20                                            000980
   10 CONTINUE                                                          000981
      IE=1                                                              000982
      RETURN                                                            000983
   20 IE=0                                                              000984
      J2=MOD(L,10)+1                                                    000985
      J1=L/10+1                                                         000986
      ENCODE(7,111,F) T1,D(J1),D(J2),T2                                 000987
  111 FORMAT(A2,2A1,A3)                                                 000988
      ENCODE(L,222,A) (B(I),I=K,J)                                      000989
  222 FORMAT(150A1)                                                     000990
      DECODE(L,F,A) X                                                   000991
      RETURN                                                            000992
      END                                                               000993
C                                                                       000994
C                                                                       000995
      SUBROUTINE DD99FF (B,A,K,L,X,IE)                                  000996
      DIMENSION B(1),A(1)                                               000997
C                                                                       000998
C     THIS SUBROUTINE PERFORMS OUTPUT "F" CONVERSION.                   000999
C                                                                       001000
C     DATE : 82.07.25 (W.M.ZUBEREK)                                     001001
C                                                                       001002
      COMMON /DD99CM/ SPACE,D(10)                                       001003
      DATA T1,T2,T3/2H(F,1H.,1H)/                                       001004
      IE=-1                                                             001005
      IF(L.GE.100) RETURN                                               001006
      LS=1                                                              001007
      IF(X.LT.0.0) LS=2                                                 001008
      Y=ABS(X)                                                          001009
      Z=1.0                                                             001010
   10 IF(Y.LT.Z) GOTO 20                                                001011
      LS=LS+1                                                           001012
      Z=Z*10.0                                                          001013
      GOTO 10                                                           001014
   20 IF(LS.GE.L) RETURN                                                001015
      LF=L-LS                                                           001016
      Y=Y+0.5/10.0**LF                                                  001017
      IF(Y.LT.Z) GOTO 30                                                001018
      LF=LF-1                                                           001019
      IF(LF.LE.0) RETURN                                                001020
   30 J2=MOD(L,10)+1                                                    001021
      J1=L/10+1                                                         001022
      J4=MOD(LF,10)+1                                                   001023
      J3=LF/10+1                                                        001024
      ENCODE(8,111,F) T1,D(J1),D(J2),T2,D(J3),D(J4),T3                  001025
  111 FORMAT(A2,6A1)                                                    001026
      ENCODE(L,F,A) X                                                   001027
      J=K+L-1                                                           001028
      DECODE(L,222,A) (B(I),I=K,J)                                      001029
  222 FORMAT(150A1)                                                     001030
      IE=0                                                              001031
      RETURN                                                            001032
      END                                                               001033
C                                                                       001034
C                                                                       001035
      SUBROUTINE DD99IC (B,A,K,L,N,IE)                                  001036
      DIMENSION B(1),A(1)                                               001037
C                                                                       001038
C     THIS SUBROUTINE PERFORMS INPUT "I" CONVERSION.                    001039
C                                                                       001040
```

```
C     DATE : 82.07.25 (W.M.ZUBEREK)                                    001041
C                                                                      001042
      COMMON /DD99CM/ SP,D(10)                                         001043
      DATA T1,T2/2H( I,1H)/                                            001044
      J=K+L-1                                                          001045
      DO 10 I=K,J                                                      001046
      IF(B(I).NE.SP) GOTO 20                                           001047
   10 CONTINUE                                                         001048
      IE=1                                                             001049
      RETURN                                                           001050
   20 IE=0                                                             001051
      J2=MOD(L,10)+1                                                   001052
      J1=L/10+1                                                        001053
      ENCODE(5,111,F) T1,D(J1),D(J2),T2                                001054
  111 FORMAT(A2,3A1)                                                   001055
      ENCODE(L,222,A) (B(I),I=K,J)                                     001056
  222 FORMAT(150A1)                                                    001057
      DECODE(L,F,A) N                                                  001058
      RETURN                                                           001059
      END                                                              001060
C                                                                      001061
C                                                                      001062
      SUBROUTINE DD99IF (B,A,K,L,N,IE)                                 001063
      DIMENSION B(1),A(1)                                              001064
C                                                                      001065
C     THIS SUBROUTINE PERFORMS OUTPUT "I" CONVERSION.                  001066
C                                                                      001067
C     DATE : 82.07.20 (W.M.ZUBEREK)                                    001068
C                                                                      001069
      COMMON /DD99CM/ SPACE,D(10)                                      001070
      DATA T1,T2/2H( I,1H)/                                            001071
      IE=-1                                                            001072
      IF(L.GE.100) RETURN                                              001073
      J2=MOD(L,10)+1                                                   001074
      J1=L/10+1                                                        001075
      ENCODE(5,111,F) T1,D(J1),D(J2),T2                                001076
  111 FORMAT(A2,3A1)                                                   001077
      J=K+L-1                                                          001078
      ENCODE(L,F,A) N                                                  001079
      DECODE(L,222,A) (B(I),I=K,J)                                     001080
  222 FORMAT(150A1)                                                    001081
      IE=0                                                             001082
      RETURN                                                           001083
      END                                                              001084
C                                                                      001085
C                                                                      001086
      SUBROUTINE DD99RL (K)                                            001087
C                                                                      001088
C     THIS SUBROUTINE RELEASES THE UNUSED LIST STRUCTURES AND LINKS    001089
C     THE RELEASED CELLS TO THE FREE LIST POINTED BY *MFREE*.          001090
C                                                                      001091
C     DATE : 82.07.15 (W.M.ZUBEREK)                                    001092
C                                                                      001093
      DIMENSION KW(450),W(450)                                         001094
      EQUIVALENCE (KW(1),W(1))                                         001095
      COMMON /DD00CM/ LW,KW                                            001096
   10 IF(K.LE.0) RETURN                                                001097
      L=KW(K+2)                                                        001098
      IF(L.LE.0) GOTO 30                                               001099
      K=SHIFT(63,30)                                                   001100
      N=AND(W(L),K)                                                    001101
      IF(N.EQ.0) GOTO 20                                               001102
      N=N-SHIFT(1,30)                                                  001103
      W(L)=OR(AND(W(L),COMPL(K)),N)                                    001104
      GOTO 30                                                          001105
```

```
   20 KW(L)=MFREE                                                    001106
      MFREE=L                                                        001107
   30 L=KW(K)                                                        001108
      KW(K)=MFREE                                                    001109
      MFREE=K                                                        001110
      K=L                                                            001111
      GOTO 10                                                        001112
      END                                                            001113
C                                                                    001114
C                                                                    001115
      SUBROUTINE DD99SF (R,K,L,X,IE)                                 001116
      DIMENSION R(1),X(1)                                            001117
C                                                                    001118
C     THIS SUBROUTINE PERFORMS CHARACTER STRING ENCODING.            001119
C                                                                    001120
C     DATE : 82.07.23 (W.M.ZUBEREK)                                  001121
C                                                                    001122
      J=K+L-1                                                        001123
      DECODE(L,111,X) (R(I),I=K,J)                                   001124
  111 FORMAT(150A1)                                                  001125
      IE=0                                                           001126
      RETURN                                                         001127
      END                                                            001128
C                                                                    001129
C                                                                    001130
      SUBROUTINE DD99ST (B,K,L,X,IE)                                 001131
      DIMENSION B(1),X(1)                                            001132
C                                                                    001133
C     THIS SUBROUTINE PERFORMS INPUT CHARACTER TRANSFER.             001134
C                                                                    001135
C     DATE : 82.07.25 (W.M.ZUBEREK)                                  001136
C                                                                    001137
      IE=0                                                           001138
      J=K+L-1                                                        001139
      ENCODE(L,111,X) (B(I),I=K,J)                                   001140
  111 FORMAT(150A1)                                                  001141
      RETURN                                                         001142
      END                                                            001143
```

SOC-302

DDATA - A FORTRAN PACKAGE OF DATA HANDLING ROUTINES

J.W. Bandler and W.M. Zuberek

September 1982, No. of Pages: 70

Revised:

Key Words: Data handling, data conversion, input/output routines, operations on data files

Abstract: DDATA is a package of subroutines for accessing data with different physical representations. The package performs all the required data conversions and communicates with the user's programs on the level of binary information. The description of a particular representation of the data must be supplied in the form of data descriptors that constitute the initial section of the data file. The format of data descriptors is fixed. Some simple elements of data preprocessing (e.g., scaling, default values) are included in the package. The package and documentation have been developed for the CDC 170/730 system with the NOS 1.4 operating system and the Fortran 4.8508 compiler. Examples are given for 23-bus and 26-bus test power systems, illustrating how data can be read, printed and stored in a descripted data file created after solving a load flow problem.

Description: Contains Fortran listing, user's manual. The listing contains 1143 lines, of which 348 are comments.

Related Work: SOC-255, SOC-296.

Price: $50.00. Source deck or magnetic tape: $150.00. Availability subject to signed author-purchaser agreement.