USER'S GUIDE TO REVISED METHOD-OF-CHARACTERISTICS SOLUTE-TRANSPORT MODEL (*MOC--Version 3.1*)

U.S. GEOLOGICAL SURVEY

Water-Resources Investigations Report 94-4115

USER'S GUIDE TO REVISED METHOD-OF-CHARACTERISTICS SOLUTE-TRANSPORT MODEL *(MOC--Version 3.1)*

By Leonard F. Konikow, Gregory E. Granato, and George Z. Hornberger

---

U.S. GEOLOGICAL SURVEY

Water-Resources Investigations Report 94-4115

Reston, VA
1994

U.S. DEPARTMENT OF THE INTERIOR

BRUCE BABBITT, Secretary

U.S. GEOLOGICAL SURVEY

Gordon P. Eaton, Director

For additional information write to:

U. S. Geological Survey
431 National Center
Reston, VA 22092

Copies of this report can be purchased from:

U.S. Geological Survey
Earth Science Information Center
Books and Open-File Reports Section
Box 25286, Mail Stop 517
Federal Center
Denver, CO 80225-0046

CONTENTS

ILLUSTRATIONS

TABLES

USER'S GUIDE TO REVISED METHOD-OF-CHARACTERISTICS SOLUTE-TRANSPORT MODEL *(MOC--Version 3.1)*

By Leonard F. Konikow, Gregory E. Granato, and
George Z. Hornberger

ABSTRACT

The U.S. Geological Survey computer model to simulate two-dimensional solute transport and dispersion in ground water (Konikow and Bredehoeft, 1978; Goode and Konikow, 1989) has been modified to improve management of input and output data and to provide progressive run-time information. All opening and closing of files are now done automatically by the program. Names of input data files are entered either interactively or using a batch-mode script file. Names of output files, created automatically by the program, are based on the name of the input file. In the interactive mode, messages are written to the screen during execution to allow the user to monitor the status and progress of the simulation and to anticipate total running time. Information reported and updated during a simulation include the current pumping period and time step, number of particle moves, and percentage completion of the current time step. The batch mode enables a user to run a series of simulations consecutively, without additional control. A report of the model's activity in the batch mode is written to a separate output file, allowing later review. The user has several options for creating separate output files for different types of data. The formats are compatible with many commercially available applications, which facilitates graphical postprocessing of model results.

This page intentionally blank

INTRODUCTION

The method of characteristics (MOC) program simulates two-dimensional ground-water flow and solute transport through porous media. The computer model, written in FORTRAN, uses the finite-difference method and the method of characteristics to solve for flow and transport, respectively. The most recent set of revisions (Version 3.1), detailed in this report, expands the program's run-time capabilities and improves output formats and options.

This report describes the changes that have been made since the last update of MOC (Goode and Konikow, 1989) and provides user instructions for MOC Version 3.1. Those not familiar with the program should refer to the original MOC documentation (Konikow and Bredehoeft, 1978). Although many changes have been made since MOC's first release, data files that ran with previous MOC versions are still compatible with MOC 3.1.

The MOC program can now be executed in either interactive or batch (or background) modes. The normal or default mode of operation is interactive. In this mode, the user is prompted for the name of the input data file and, if necessary, is asked questions relating to overwriting of existing files. In previous versions of MOC, the user had to open input and output files before executing the model. Naming of output files is accomplished automatically in both modes, which facilitates the creation of multiple output files from a single input file and reduces the amount of data entry required to use MOC.

In the interactive mode, status messages and graphical displays allow the user to monitor the progress of the simulation and make a rough estimate of the remaining run time. If the on-screen displays indicate an unexpectedly large run time, it may indicate an error in the input data or in the problem conceptualization, and the user can terminate the simulation and examine the preliminary output to ascertain the cause of the lengthy execution time. This may eliminate a great deal of useless computational effort.

If a user does not wish to monitor simulation progress or wishes to automatically run a series of simulations with multiple input data files, then

MOC Version 3.1 can be run in batch mode. The user must create a text file containing the names of a series of input data files to be run consecutively when MOC is executed. On some machines this may be used in conjunction with a command to run the program in the background; in that case the program will continue running after the user has logged out. In the batch mode, normal output to the screen is eliminated.

Through the selection of new options in the input data file, MOC now has the ability to generate additional output files that contain specific types of data. MOC can create ASCII text files for selected parameters; these files are written in formats designed to be read directly by commonly available independent graphics, spreadsheet, and contour plotting applications. Input options available for these optional output files are explained in Appendix A. PREMOC (Granato and others, 1993), a preprocessor for MOC, has been updated (to its Version 4.1) to accommodate the new options available in MOC.

The format of the standard MOC output file has been altered to improve its readability when viewed on a standard computer monitor. Most carriage control characters have been eliminated and most output is written in the first 80 columns of a line, which are visible on most monitors.

INTERACTIVE MODE

MOC 3.1 is implemented in the interactive mode by entering into the computer the name of the executable code. Upon the start of execution, a title screen is written to the monitor and a prompt is given for the name of the input file (see Appendix A for precise instructions on the proper formats for the input data file). If the file requested is not found, then the program queries the user for a new disk directory and (or) file name.

Names for MOC 3.1 output files are created automatically by the program (see details in the section *OUTPUT FILE FORMATS*). If a file of the same name as a potential output file exists in the directory specified, the user is prompted to decide whether or not to overwrite the older file to prevent an

inadvertent loss of data (this choice is not available in batch mode). If the user chooses to not overwrite an existing file, execution of the program is terminated.

During execution a progress report is printed to the screen. The current and total number of pumping (stress) periods, time steps (for solving the flow equation), and particle moves (time increments for solving the transport equation) in the current time step are indicated. Asterisks are printed along a graded line to show the percent of particle moves completed; the particle moves are typically the most computationally intensive part of a simulation. This provides a real-time diagnostic tool for the simulation and an estimate of the total run time. An example of a progress report printed to the screen during execution of MOC 3.1 in its interactive mode is shown in figure 1. This progress report indicates that the program is in the first and only pumping period and time step. There are 20 particle moves required, and the program has stepped through 10 percent of them.

```
 sample.dat
FILE EXISTS
PUMPING PERIOD:                                  1  OF    1
TIME STEP:                                       1  OF    1
NUMBER OF PARTICLE MOVES:        20

0.....10.....20.....30.....40.....50.....60.....70.....80.....90....100
*******
```

Figure 1.--Example of screen output showing progress report.

BATCH MODE

The batch mode is designed to run multiple input files consecutively without user or operator interference. This mode would be of value typically during model calibration or sensitivity testing, when a large number of simulations are made while varying values of key parameters. Batch mode is initiated automatically if a file named *MOCBAT.DAT* (upper or lower case) exists in the directory in which MOC 3.1 is run. The program searches for the *MOCBAT.DAT* file only in the directory from which the command was issued.

MOCBAT.DAT is an ASCII file created by the user in a specific format. The first line must contain an integer denoting the number of files in the batch (which equals the number of subsequent lines in the file). The second and following lines each contain a file name (and path, if pertinent) of input data files for MOC. The integer will be read in free format and the file names in a character format (A40). *MOCBAT.DAT* can be created by any ASCII text editor. Normal screen output is turned off in this mode.

An example of a representative *MOCBAT.DAT* file is shown in figure 2. This example file indicates that the MOC model should be run four consecutive times using the listed input data files in order: *sample.dat*, *test1.dat*, *test2.dat*, and *\home\data\test3.dat*.

```
4
sample.dat
test1.dat
test2.dat
\home\data\test3.dat
```

Figure 2.--Example of an input file for the batch mode.

When MOC 3.1 is run in batch mode, it will generate a file, named *MOCBAT.RPT*, which is a report of the program's activity while running in batch mode. Files that are created by the process of running MOC and any errors or problems encountered with input files are recorded. Figure 3 presents an example of part of a report created when MOC 3.1 was executed in the batch mode in the same directory as the file *MOCBAT.DAT* shown in figure 2. Because the file "test1.dat" did not exist, an error message was written to the report file and execution progressed to the next file in the list.

In order to run MOC interactively after being executed in batch mode, the file *MOCBAT.DAT* must be deleted or renamed.

```
 USGS MOC model, Version 3.1 (BATCH MODE)

sample.dat
FILE EXISTS
U.S.G.S. Method-of-characteristics model for solute transport in ground water

Sample data

The following OUTPUT files will be generated:

     standard MOC output data ......... sample.out

     observation well data ............ sample.obs

PUMPING PERIOD:                        1  OF    1

TIME STEP:                             1  OF    1

NUMBER OF PARTICLE MOVES:       20

test1.dat
Input file specified in mocbat.dat does not exist.
Processing will continue with next input file.
...
...
```

Figure 3.--An example MOCBAT.RPT file, created when MOC 3.1 was executed in the same directory as the file MOCBAT.DAT in Figure 2. (Only part of the file is shown.)

OUTPUT FILES

The standard MOC output file lists input parameters, head and concentration distributions, stability criteria, and the fluid and chemical mass balances. In addition to the standard output, MOC 3.1 can also create several separate output files containing data describing a single parameter, variable, or "records" at observation wells. Options to create these files aid in the ability to import the data into graphing and contouring programs. These options for output files are accessed using numerical codes (which act as flags or toggles) in the input data file, as described in Appendix A. The user activates the options by setting NOUTFL, the last parameter on Line 2 of the input data set, equal to 1. This flag causes the program to read Line 3.2, on which all of the specifications for optional output files are defined.

All data are written to these optional files using an exponential FORTRAN format of E10.3. Spaces are used as delimiters between adjacent

data values, and double quotes are used around any header information; these conventions are consistent with the input requirements of many commonly available graphical software applications.

Names for MOC 3.1 output files are based on the name of the input data file. The output file names are automatically created by truncating the extension (or suffix) of the input file name, if it exists, and inserting (or appending) an appropriate new file name extension. If an extension exists, it is recognized by a *period* followed by one to three characters at the end of the file name (for example, in the file name *example.dat*, the characters *"dat"* would be recognized as the suffix). The naming conventions are described in table 1.

The MOC 3.1 file naming system is compatible with the file naming nomenclature and standards of MS-DOS[1], Macintosh, UNIX, and other operating systems. Complete pathnames of files should be specified when the input file resides in another directory. All output files are written to the directory in which each input file is located.

Standard MOC Output Data

Name.....*filename.*out

This file is the primary and comprehensive output file for MOC 3.1. An example is presented in Appendix C. Input data parameters are printed in the beginning of the file and calculated values later in the file. Compared to the previous version of MOC, output of carriage control characters is eliminated and the amount of output written in the first 80 columns of a line is maximized to facilitate on-screen viewing of output files.

[1]The use of brand/firm/trade names in this report is for identification purposes only and does not constitute endorsement by the U.S. Geological Survey.

Table 1.--Naming convention for all output files, showing suffixes identifying optional output files when the original input data file was named "filename.dat" (variable segments of file names are shown in italics).

File	Description
filename.out	standard MOC output data
filename.obs *or* *filename*.o*N*	observation well head and concentration data; all wells included in one file (.obs), or each well has its own file (.o*N* for *N* wells)
filename.vel	velocity data
filename.hd0	initial hydraulic head data
filename.hd1	final hydraulic head data
filename.cn0	initial concentration data
filename.cn1	final concentration data
filename.trn	transmissivity data
filename.thk	saturated thickness data
filename.rec	diffuse recharge/discharge data
filename.prm	hydraulic conductivity data

Observation Well Data

Name.....*filename*.obs

This file includes the calculated heads and concentrations at up to five selected observation wells in an ASCII format. The file is generated by the program when NOBSO is set equal to 1. The file has a three-line heading in double quotes (a common portable heading delimiter). The second header line includes the nodal location of the observation wells. Data for up to a maximum of five wells are written in each row following the header lines. The output format is illustrated in Figure 4.

```
                                "OBSERVATION WELL DATA"
"NODE  (I,J):  (  4,  6)                ( 3,  8)              ( 5,  9)   "
"TIME(YRS)      HEAD        CONC.       HEAD        CONC.      HEAD       CONC.    "
0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00 0.0000E+00 0.0000E+00
8.3333E-01  2.0648E+02  0.1100E+01  0.2034E+02  0.1000E+01 0.0000E+00 0.0000E+00
...
...
```

Figure 4.--An example observation well data output file (of the ".obs" type) for a case in which three observation wells are specified. (Only the top part of the file is shown.)

Name.....*filename.oN*

Optionally, data for each observation well can be written to a separate file when NOBSO is set equal to 2. Each file name includes a number at the end of its suffix denoting which observation well is associated with that file (for example, *sample.*o3 is the file containing the data for observation well 3, specified in the input data file named *sample.dat*). In this file, the format is the same as the first three columns of the ".obs" file above (also see example in Appendix C).

Velocity Data

Name.....*filename*.vel

The file containing velocity information includes one header line. The four columns below it list the x-coordinate, the y-coordinate, the component of the velocity in the x-direction, and the component of the velocity in the y-direction for each node in the grid. Data from the first and last column of nodes and first and last row of nodes are not printed, as these are always no-flow boundary cells.

The coordinates for each finite-difference cell are listed in map units (that is, the size of the grid spacing, XDEL and YDEL, multiplied by the number of nodes in the respective direction in the grid). However, the finite-difference grid for the MOC model assumes that node (1,1) is in the upper left corner of the grid, whereas most plotting packages assume a normal

Cartesian coordinate system in which the origin (where x = 0 and y = 0) is located in the lower left corner of the plotting space. Therefore, to ensure that values are plotted in their proper geometric locations and that the plotted map will correspond with the model grid, the y-coordinates are transformed in this file so that the lower left corner of the model grid has the spatial (x,y) location of (0,0). A sample velocity data file is shown is Figure 5.

```
" VELOCITIES; the four columns = X, Y, VX, VY"
3.000E+03   2.300E+03    0.000E+00   0.000E+00
5.000E+03   2.300E+03    0.000E+00   0.000E+00
7.000E+03   2.300E+03    0.111E+00   0.121E+00
3.000E+03   2.100E+03   -0.341E+00   0.009E+00
5.000E+03   2.100E+03    0.000E+00   0.000E+00
7.000E+03   2.100E+03    0.111E+00  -0.380E+00
3.000E+03   1.900E+03   -0.341E+00   0.009E+00
...
...
```

Figure 5.--An example velocity data output file. (Only the top part of the file is shown.)

Hydraulic Head Data

Names.....*filename*.hd0, *filname*.hd1

These files can be used to map or contour initial and (or) final groundwater heads. Two options exist for the format of the output data: column or matrix style. Matrix format is a two-dimensional array of values containing a head value at each node; each value is located at the same respective row and column in the array as is the node in the model grid where the value was calculated. In the column format, the three columns are the x-coordinate, y-coordinate, and head, and the y-coordinates are transformed as described for the velocity data file. Coordinates are listed in map units. Data from the first and last columns of nodes and first and last rows of nodes are not printed for either format.

Concentration Data

Names.....*filename*.cn0, *filename*.cn1

These files include the initial and (or) final concentration data at each node. The formatting options are identical to those described above for the data files for hydraulic head. If the subgrid option is implemented for the solution to the transport equation, concentration values are only available for the nodes within the subgrid. Examples of output files for final concentrations in both column and matrix formats are included in Appendix C.

Other Output Files

Transmissivity data ..... *filename*.trn
Saturated thickness data ..... *filename*.thk
Diffuse recharge/discharge data ..... *filename*.rec
Hydraulic conductivity data ..... *filename*.prm

The user has the option to write either all or none of these four separate output files. These files also may be created in either a column or matrix style.

IMPLEMENTATION OF REVISED CODE

Detailed instruction on how to convert the most recent previous version of the MOC model (Goode and Konikow, 1989) to the present Version 3.1 are documented in Appendix D. The source code for the MOC model is written in standard FORTRAN. Implementation of the revisions requires deleting specified lines from the previous version and inserting new lines of code in the appropriate locations, as specified in Appendix D.

After the source code has been modified, it must be recompiled using a FORTRAN compiler. The new code has been successfully compiled on IBM compatible machines using the Microsoft FORTRAN compiler (Version 5.0) and the Microsoft FORTRAN Powerstation Optimizing Compiler (Version 1.0)

for use with extended memory. It has also been complied successfully on a UNIX-based Data General workstation using the Green Hills FORTRAN-88000 compiler (using the "-novms" compiling option).

SUMMARY

MOC Version 3.1 implements several new input and output options to improve the clarity of the calculated information and the usability of the model. The format of the main output file is changed, and options to create separate, output files for specific parameters and variables are added to facilitate postprocessing of data. This new version of MOC can be run in a batch mode, making it possible to run multiple data sets from one set of commands. When the program is run interactively, the progress of the simulation is displayed graphically on the screen. This provides information on simulation progress and for run-time estimation. These changes allow the user of the MOC solute-transport simulation model to monitor and process the program's output more easily and effectively than before.

REFERENCES

Goode, D.J., and Konikow, L.F., 1989, Modification of a method-of-characteristics solute-transport model to incorporate decay and equilibrium-controlled sorption or ion exchange: U. S. Geol. Survey Water-Res. Inv. Report 89-4030, 65 p.

Granato, G.E., Konikow, L.F., and Srinivasan, P., 1993, PREMOC version 4.0, A preprocessor for the two-dimensional method of characteristics (MOC) solute-transport model: International Ground Water Modeling Center, Golden, CO, Report IGWMC - FOS 23, 22 p.

Konikow, L.F., and Bredehoeft, J.D., 1978, Computer model of two-dimensional solute transport and dispersion in ground water: U.S. Geol. Survey Techniques of Water-Resources Investigations, Book 7, Chapter C2, 90 p.

APPENDIX 1: DATA INPUT FORMATS FOR *MOC (3.1)*

Line	Column	Format	Variable	Definition
1	1-80	10A8	TITLE	Description of problem
2	1-4	I4	NTIM	Maximum number of time steps in a pumping (stress) period for transient flow (limit=100)*. If S=0, pumping period (PINT) is divided into NTIM intervals.
	5-8	I4	NPMP	Number of pumping periods. Note that if NPMP>1, then data set 10 must be completed.
	9-12	I4	NX	Number of nodes in x direction (limit=40)*. To select optional use of subgrid for transport, specify NX as a negative value and complete line 2.1.
	13-16	I4	NY	Number of nodes in y direction (limit=40)*.
	21-24	I4	NPNT	Time-step interval for printing hydraulic and chemical output data.
	25-28	I4	NITP	Number of iteration parameters (usually $4 \leq NITP \leq 7$ in ADIP). For SIP, set NITP=10.
	29-32	I4	NUMOBS	Number of observation points to be specified in data set 1 (limit=5)*.
	33-36	I4	ITMAX	Maximum allowable number of iterations for flow equation (usually $100 \leq ITMAX \leq 200$).
	37-40	I4	NREC	Number of pumping or injection wells to be specified in data set 2.
	41-44	I4	NPTPND	Initial number of particles per node (options=1, 4, 5, 8, 9, 16).
	45-48	I4	NCODES	Number of node identification codes to be specified in data set 7 (limit=10)*.
	49-52	I4	NPNTMV	Particle movement interval (IMOV) for printing chemical output data. (Specify 0 to print only at end of time steps.)
	53-56	I4	NPNTVL	Option for printing computed velocities (0=do not print; -1=1st time step; -2=last time step; n>0=every nth time step).
	57-60	I4	NPNTD	Option for printing computed dispersion equation coefficients (0=never; 1=first time step; 2=all time steps).
	61-64	I4	NPDELC	Option for printing computed changes in concentration (0=do not print; 1=print whenever chemical output is printed).
	65-68	I4	NPNCHV	Option to write velocity data to a separate output file (0=do not write; -1=1st time step; -2=last time step; n>0=every nth time step)**.
	69-72	I4	IREACT	Option to specify type of reaction (-1=decay only; 0=no reaction; 1=linear sorption; 2=Freundlich sorption; 3=Langmuir sorption; 4=monovalent exchange; 5=divalent exchange; 6=mono-divalent exchange; 7=di-monovalent exchange). If IREACT≠0, line 3.1 is required.
	73-74	I2	NOUTFL	Flag to allow specifications for creation of additional output files (0=do not allow; 1=read line 3.2).
2.1 (optional; include only if NX<0)	***	free (integers)	MX	x-coordinate of upper-left node of transport subgrid.
			MY	y-coordinate of upper-left node of transport subgrid.
			MMX	x-coordinate of lower-right node of transport subgrid.
			MMY	y-coordinate of lower-right node of transport subgrid.

See footnotes at end of table.

Data input formats -- Continued

Line	Column	Format	Variable	Definition
3	1-5	G5.0	PINT	Pumping period (or stress period), in years.
	6-10	G5.0	TOL	Convergence criteria for iterative solution to flow equation (usually TOL≤0.01).
	11-15	G5.0	POROS	Effective porosity.
	16-20	G5.0	BETA	Longitudinal dispersivity.
	21-25	G5.0	S	Storage coefficient (set S=0 for steady flow problems).
	26-30	G5.0	TIMX	Time increment multiplier for transient flow problems. TIMX is disregarded if S=0.
	31-35	G5.0	TINIT	Size of initial time step in seconds. TINIT is disregarded if S=0.
	36-40	G5.0	XDEL	Width of finite-difference cell in x direction.
	41-45	G5.0	YDEL	Width in finite-difference cell in y direction.
	46-50	G5.0	DLTRAT	Ratio of transverse to longitudinal dispersivity.
	51-55	G5.0	CELDIS	Maximum cell distance per particle move (value between 0 and 1.0).
	56-60	G5.0	ANFCTR	Ratio of T_{yy} to T_{xx}.
3.1# (optional; include only if IREACT≠0)	***	free# (real)	THALF	Decay half-life of solute, in seconds (if no decay, specify THALF=0.0).
			DK	Linear sorption distribution coefficient (L^3M^{-1}).
			RHOB	Aquifer bulk density, mass of solid per unit volume of aquifer (ML^{-3}).
			EKF	Freundlich sorption coefficient, units depend on XNF.
			XNF	Freundlich sorption exponent, dimensionless.
			EKL	Langmuir sorption coefficient (L^3M^{-1}).
			CEC	Maximum sorption capacity or ion-exchange capacity (MM^{-1}).
			EK	Ion-exchange selectivity coefficient, dimensionless.
			CTOT	Total solution concentration of two exchanging ions (equivalents/L^3).
3.2 (optional; include only if NOUTFL=1)	1-2	I2	NOBSO	Write observation-point data to separate output file (0=do not write; 1=write all data to one file; 2=write data in NUMOBS files, one file for each observation point)†.
	3-4	I2	NHEADO	Write heads to separate output file (0=do not write; -1=initial head; 1=final head; 2=initial and final head).
	5-6	I2	NCONCO	Write concentration data to separate output file (0=do not write; -1=initial concentration; 1= final concentration; 2=initial and final concentration).
	7-8	I2	NPARMO	Write transmissivity, saturated thickness, diffuse recharge/discharge, and hydraulic conductivity data to separate files (0=do not write; 1=write).
	9-10	I2	IFMT	Option to specify format for output files generated by NHEADO, NCONCO, and NPARMO (0=column; 1=matrix)††.

See footnotes at end of table.

Data input formats -- Continued

Data set	Number of lines	Format	Variable	Definition
1	Value of NUMOBS (limit=5)*	2I2	IXOBS, IYOBS	x and y coordinates of observation points. This data set is eliminated if NUMOBS=0.
2	Value of NREC	2I2, 2G8.2	IX, IY, REC, CNRECH	x and y coordinates of pumping (+) or injection (-) wells, volumetric rate in L^3/second, and if an injection well, the concentration of injected water. This data set is eliminated if NREC=0.
3	a. 1 b. Value of NY (limit=40)*§	I1, G10.0 20G4.1	INPUT, FCTR VPRM	Parameter line‡ for transmissivity. Array for temporary storage of transmissivity data, in L^2/second. For an anisotropic aquifer, read in values of T_{xx} and the program will adjust for anisotropy by multiplying T_{yy} by ANFCTR.
4	a. 1 b. Value of NY (limit=40)*§	I1, G10.0 20G3.0	INPUT, FCTR THCK	Parameter line‡ for THCK. Saturated thickness of aquifer.
5	a. 1 b. Value of NY (limit=40)*§	I1, G10.0 20G4.1	INPUT, FCTR RECH	Parameter line‡ for RECH. Diffuse recharge (-) or discharge (+), in L/second.
6	a. 1 b. Value of NY (limit=40)*	I1, G10.0 40I1	INPUT, FCTR NODEID	Parameter line‡ for NODEID. Node identification matrix (used to define constant-head nodes or other boundary conditions and stresses).
7	Value of NCODES (limit=10)*	I2,3G10.2, I2	ICODE, FCTR1, FCTR2, FCTR3, OVERRD	Instructions for using NODEID array. Where NODEID=ICODE, the program sets leakance=FCTR1, CNRECH=FCTR2, and if OVERRD is nonzero, RECH=FCTR3. Set OVERRD=0 to preserve values of RECH specified in data set 5.
8	a. 1 b. Value of NY (limit=40)*§	I1, G10.0 20G4.0	INPUT, FCTR WT	Parameter line‡ for WT. Initial water table or potentiometric elevation, or constant head in stream or source bed.
9	a. 1 b. Value of NY (limit=20)*	I1, G10.0 20G4.0	INPUT, FCTR, CONC	Parameter line‡ for CONC. Initial concentration in aquifer.
10 (optional; only include if NPMP>1)				This data set allows time-step and iteration parameters, print options, and stress data to be revised for each pumping period of the simulation. The sequence of lines in data set 10 must be repeated (NPMP-1) times (that is, data set 10 is required for each pumping period after the first).

See footnotes at end of table.

Data input formats -- Continued

Data set	Number of lines	Format	Variable	Definition
10 (continued)				
	a. 1	I1	ICHK	Parameter to check whether any revisions are desired. Set ICHK=1 if data are to be revised, and then complete data set 10b and c. Set ICHK=0 if data are not to be revised for the next pumping period, and skip rest of data set 10.
	b. 1	10I4, 3G5.0	NTIM, NPNT, NITP, ITMAX, NREC, NPNTMV, NPNTVL, NPNTD, NPDELC, NPNCHV, PINT, TIMX, TINIT	Thirteen parameters to be revised for next pumping period; the parameters were previously defined in the description of data lines 2 and 3. Only include this line if ICHK=1 in previous part a.
	c. Value of NREC	2I2, 2G8.2	IX, IY, REC, CNRECH	Revision of previously defined data set 2. Include part c only if ICHK=1 in previous part a and if NREC>0 in previous part b.

* These limits can be modified if necessary by changing the corresponding array dimensions in the COMMON statements of the program.

** File will be named *filename*.vel, where *filename* is the prefix part of the name of the input data. Each row contains the *x* coordinate, the *y* coordinate, the component of the velocity in the *x* direction, and the component of the velocity in the *y* direction. Time step is noted in the header.

*** Free format: leave a space between each entry.

Variables used on line 3.1 will depend on value of IREACT:

IREACT	Variables, in this order
-1	THALF
0	none
1	DK, RHOB, THALF
2	RHOB, EKF, XNF, THALF
3	RHOB, EKL, CEC, THALF
4-7	RHOB, EK, CEC, CTOT, THALF

† File will be named *filename*.obs if NOBSO=1, or multiple files will be named *filename*.obN, where N is the number of the observation well, if NOBSO=2. Output is in column format, where column 1 is time (in years), column 2 is head, and column 3 is concentration. If NOBSO=1, additional columns are added in sequence for head and concentration at next observation points.

†† Files will be named *filename.xxx*, where *xxx* is a suffix that will be generated (*.hd0* for initial heads; *.hd1* for final heads; *.cn0* for initial concentrations; *.cn1* for final concentrations; *.trn* for transmissivity; *.thk* for saturated thickness; *.rec* for diffuse recharge/discharge; *.prm* for hydraulic conductivity). If matrix format is selected, values will be printed in a matrix having NX columns and NY rows of values in their respective grid locations. If column format is selected, each row (or line) will include the *x* coordinate, *y* coordinate, and the value of the parameter at that location.

§ If NX>20, the number of lines doubles because format limits data to 20 values per line.

‡ The parameter line must be the first line of the indicated data sets. It is used to specify whether the parameter is constant and uniform, and can be defined by one value, or whether it varies in space and must be defined at each node. If INPUT=0, the data set has a constant value, which is defined by FCTR. If INPUT=1, the data set is read next as described by part b. Then FCTR is a multiplication factor for the values read in the data set.

APPENDIX 2: INPUT DATA FOR SAMPLE PROBLEM

INPUT FILE: prob3.dat

```
Column Labels
          1         2         3         4         5         6         7
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345
          ↓         ↓         ↓         ↓         ↓         ↓         ↓

Line 1    Problem 3 (Steady flow, 1 well, constant-head boundaries, linear sorption)
Line 2       1   1  -9  10       1   7   2 100   1   9   2  10   1   0   0   0   1 1
Line 2.1              3           2           7           8
Line 3    2.500.0001.3000100.00.0000.0000.000900.0900.0.3000.50001.000
Line 3.1     1.000000      0.2000000      0.0000000E+00
Line 3.2     2 0 2 0 1
Data Set 1   5 4
             5 7
Data Set 2   4 7 1.0    0.00E+00
Data Set 3  0 0.100
Data Set 4  0  20.0
Data Set 5  0 0.000E+00
Data Set 6  1  1.00
            000000000
            022111220
            000000000
            000000000
            000000000
            000000000
            000000000
            000000000
            022222220
            000000000
Data Set 7   2  1.00      0.000E+00 0.000E+00 0
             1  1.00       100.     0.000E+00 0
Data Set 8  1  1.00
             0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
             0.0100.100.100.100.100.100. 0.0
             0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
             0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
             0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
             0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
             0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
             0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
             0.075.075.075.075.075.075.075.0 0.0
             0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
Data Set 9  0 0.000E+00
```

APPENDIX 3: SELECTED OUTPUT FOR SAMPLE PROBLEM

MAIN OUTPUT FILE: prob3.out

```
U.S.G.S. Method-of-characteristics model for solute transport in ground water

 Problem 3 (Steady flow, 1 well, constant-head boundaries, linear sorption)



          I N P U T     D A T A

            GRID DESCRIPTORS

NX     (Number of columns)              =     9
NY     (Number of rows)                 =    10
XDEL   (Grid spacing in X-direction)    =   900.0
YDEL   (Grid spacing in Y-direction)    =   900.0


      SECONDARY SUBGRID FOR TRANSPORT

   NMX    (NUMBER OF COLUMNS)  =     5
   NMY    (NUMBER OF ROWS)     =     7

   CROSS-REF. TO PRIMARY GRID      IX  IY
                                  --- ---
     FIRST NODE (UPPER LEFT) AT:   3   2
     LAST NODE (LOWER RIGHT) AT:   7   8


           TIME  PARAMETERS

NTIM    (Max. no. of time steps)      =         1
NPMP    (No. of pumping periods)      =         1
PINT    (Pumping period in years)     =         2.500
TIMX    (Time step multiplier--flow)  =         0.00
TINIT   (Initial time step in sec.)   =         0.


 HYDROLOGIC AND CHEMICAL PARAMETERS

S       (Storage coefficient)         =      0.000000
POROS   (Effective porosity)          =      0.300
BETA    (Longitudinal dispersivity)   =    100.0
DLTRAT  (Ratio of transverse to
        longitudinal dispersivity)    =      0.30
ANFCTR  (Ratio of Tyy to Txx)         =      1.000000


         EXECUTION PARAMETERS

NITP    (No. of iteration parameters)    =    7
TOL     (Convergence criteria--flow)     =    0.10E-03
ITMAX   (Max. no. of iterations--flow)   =  100
CELDIS  (Max. cell distance per move
           of particles--M.O.C.)         =    0.500
NPTPND  (No. particles per node)         =    9
```

PROGRAM OPTIONS

```
NPNT   (Time step interval for complete printout)          =   1
NPNTMV (Move interval for chem. concentration printout)    =  10
NPNTVL (Time step interval for velocity printout:
        0=Never; -1=First time step; -2=Last time step;
        N>0=Every Nth time step)                           =   1
NPNTD  (Print option--dispersion-equation coefficients:
        0=Never; 1=First time step; 2=All time steps)      =   0
NUMOBS (No. of observation wells for hydrograph printout)  =   2
NREC   (No. of pumping wells)                              =   1
NCODES (No. of codes used for node identification)         =   2
NPDELC (Print option--conc. change:  0=NO; 1=YES)          =   0
IREACT (Reaction specifier)                                =   1
NOUTFL (Extra output file specifications: 0=NO; 1=YES)     =   1
```

REACTION - LINEAR SORPTION

```
RHOB    (Bulk density)                =  2.00000E-01
DK      (Distribution coefficient)    =  1.00000E+00
RF      (Retardation factor)          =  1.66667E+00
```

OPTIONS TO CREATE ADDITIONAL OUTPUT FILES

```
NOBSO    Write observation well data   (0: do not write; 1: write all
         data to one file; 2: write data to NUMOBS files--one file for
         each observation point)                                        =  2

NHEADO   Write head data (0: do not write; -1: write initial head only;
         1: write final head only;  2: write initial and final head)    =  0

NPNCHV   Write velocity data (-1:initial velocity; 0: do not write;
         1: write final velocity; N>0: every Nth time step)             =  0

NCONCO   Write concentration data (0: do not write; -1: write initial
         conc.; 1: write final conc.; 2: write initial and final conc.) =  2

NPARMO   Write transmissivity, sat. thickness, diffuse recharge/dis-
         charge, & hydraulic conductivity data to separate files using
         values from input data  (0: do not write any; 1: write all)    =  0

IFMT     Write data in column or matrix format (0: column; 1: matrix)   =  1
```

THE FOLLOWING ADDITIONAL OUTPUT FILES WILL BE CREATED:

```
observation well data ..... prob3.o1
observation well data ..... prob3.o2
initial concentration data ..... prob3.cn0
final concentration data ..... prob3.cn1
```

STEADY-STATE FLOW

TIME INTERVALS (IN SEC) FOR SOLUTE-TRANSPORT SIMULATION
0.78894E+08

LOCATION OF OBSERVATION WELLS

NO.	X	Y
1	5	4
2	5	7

LOCATION OF PUMPING WELLS

X	Y	RATE(L**3/SEC)	CONC.
4	7	1.0000	0.00

AREA OF ONE CELL = 0.8100E+06

X-Y SPACING:
900.00
900.00

TRANSMISSIVITY MAP (L**2/SEC)

0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
0.00E+00	1.00E-01	1.00E-01	1.00E-01	1.00E-01	1.00E-01	1.00E-01	1.00E-01	0.00E+00
0.00E+00	1.00E-01	1.00E-01	1.00E-01	1.00E-01	1.00E-01	1.00E-01	1.00E-01	0.00E+00
0.00E+00	1.00E-01	1.00E-01	1.00E-01	1.00E-01	1.00E-01	1.00E-01	1.00E-01	0.00E+00
0.00E+00	1.00E-01	1.00E-01	1.00E-01	1.00E-01	1.00E-01	1.00E-01	1.00E-01	0.00E+00
0.00E+00	1.00E-01	1.00E-01	1.00E-01	1.00E-01	1.00E-01	1.00E-01	1.00E-01	0.00E+00
0.00E+00	1.00E-01	1.00E-01	1.00E-01	1.00E-01	1.00E-01	1.00E-01	1.00E-01	0.00E+00
0.00E+00	1.00E-01	1.00E-01	1.00E-01	1.00E-01	1.00E-01	1.00E-01	1.00E-01	0.00E+00
0.00E+00	1.00E-01	1.00E-01	1.00E-01	1.00E-01	1.00E-01	1.00E-01	1.00E-01	0.00E+00
0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00

AQUIFER THICKNESS (L)

0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	20.0	20.0	20.0	20.0	20.0	20.0	20.0	0.0
0.0	20.0	20.0	20.0	20.0	20.0	20.0	20.0	0.0
0.0	20.0	20.0	20.0	20.0	20.0	20.0	20.0	0.0
0.0	20.0	20.0	20.0	20.0	20.0	20.0	20.0	0.0
0.0	20.0	20.0	20.0	20.0	20.0	20.0	20.0	0.0
0.0	20.0	20.0	20.0	20.0	20.0	20.0	20.0	0.0
0.0	20.0	20.0	20.0	20.0	20.0	20.0	20.0	0.0
0.0	20.0	20.0	20.0	20.0	20.0	20.0	20.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

HYDRAULIC CONDUCTIVITY MAP (L/SEC)

0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
0.00E+00	5.00E-03	5.00E-03	5.00E-03	5.00E-03	5.00E-03	5.00E-03	5.00E-03	0.00E+00
0.00E+00	5.00E-03	5.00E-03	5.00E-03	5.00E-03	5.00E-03	5.00E-03	5.00E-03	0.00E+00
0.00E+00	5.00E-03	5.00E-03	5.00E-03	5.00E-03	5.00E-03	5.00E-03	5.00E-03	0.00E+00
0.00E+00	5.00E-03	5.00E-03	5.00E-03	5.00E-03	5.00E-03	5.00E-03	5.00E-03	0.00E+00
0.00E+00	5.00E-03	5.00E-03	5.00E-03	5.00E-03	5.00E-03	5.00E-03	5.00E-03	0.00E+00
0.00E+00	5.00E-03	5.00E-03	5.00E-03	5.00E-03	5.00E-03	5.00E-03	5.00E-03	0.00E+00
0.00E+00	5.00E-03	5.00E-03	5.00E-03	5.00E-03	5.00E-03	5.00E-03	5.00E-03	0.00E+00
0.00E+00	5.00E-03	5.00E-03	5.00E-03	5.00E-03	5.00E-03	5.00E-03	5.00E-03	0.00E+00
0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00

FLOW MODEL (PRIMARY GRID):

NO. OF FINITE-DIFFERENCE CELLS IN AQUIFER = 56
AREA OF AQUIFER IN MODEL = 0.45360E+08 (L**2)

TRANSPORT SUBGRID:

NO. OF FINITE-DIFFERENCE CELLS IN AQUIFER = 35
AREA OF AQUIFER IN MODEL = 0.28350E+08 (L**2)

NZCRIT (Max. no. of cells that can be void of particles; if exceeded, particles are regenerated) = 1

NODE IDENTIFICATION MAP

0	0	0	0	0	0	0	0	0
0	2	2	1	1	1	2	2	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	2	2	2	2	2	2	2	0
0	0	0	0	0	0	0	0	0

NO. OF NODE IDENT. CODES SPECIFIED = 2

THE FOLLOWING ASSIGNMENTS HAVE BEEN MADE:

CODE NO.	LEAKANCE	SOURCE CONC.	RECHARGE
2	0.100E+01	0.00	
1	0.100E+01	100.00	

LEAKANCE COEF. = VERTICAL HYDRAULIC COND./THICKNESS (L/(L*SEC))

0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
0.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	0.00E+00
0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
0.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	0.00E+00
0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00

DIFFUSE RECHARGE AND DISCHARGE (L/SEC)

0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00

```
INITIAL HEADS AND SPECIFIED BOUNDARY HEADS

    0.    0.    0.    0.    0.    0.    0.    0.    0.
    0.  100.  100.  100.  100.  100.  100.  100.    0.
    0.    0.    0.    0.    0.    0.    0.    0.    0.
    0.    0.    0.    0.    0.    0.    0.    0.    0.
    0.    0.    0.    0.    0.    0.    0.    0.    0.
    0.    0.    0.    0.    0.    0.    0.    0.    0.
    0.    0.    0.    0.    0.    0.    0.    0.    0.
    0.    0.    0.    0.    0.    0.    0.    0.    0.
    0.   75.   75.   75.   75.   75.   75.   75.    0.
    0.    0.    0.    0.    0.    0.    0.    0.    0.
```

```
HEAD DISTRIBUTION (BY ROW)

Number of time steps =      0
      Time (seconds) =  0.00000E+00
      Time (days)    =  0.00000E+00
      Time (years)   =  0.00000E+00

 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000
 0.0000000 100.0000000 100.0000000 100.0000000 100.0000000 100.0000000 100.0000000 100.0000000   0.0000000
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000
 0.0000000  75.0000000  75.0000000  75.0000000  75.0000000  75.0000000  75.0000000  75.0000000   0.0000000
 0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000   0.0000000
```

```
CONCENTRATION

Number of time steps =      0
      Time (seconds) =  0.00000E+00
 Chem.time (seconds) =  0.00000E+00
 Chem.time (days)    =  0.00000E+00
      Time (years)   =  0.00000E+00
 Chem.time (years)   =  0.00000E+00
 No. moves completed =      0


    0    0    0    0    0

    0    0    0    0    0

    0    0    0    0    0

    0    0    0    0    0

    0    0    0    0    0

    0    0    0    0    0

    0    0    0    0    0


  N =     1
  NUMBER OF ITERATIONS =    20
```

HEAD DISTRIBUTION (BY ROW)

Number of time steps = 1
Time (seconds) = 0.78894E+08
Time (days) = 0.91313E+03
Time (years) = 0.25000E+01

0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
0.0000000	99.9999995	99.9999995	99.9999995	99.9999995	99.9999995	99.9999995	99.9999995	0.0000000
0.0000000	95.9387858	95.9346978	95.9468712	95.9958792	96.0611455	96.1171357	96.1482887	0.0000000
0.0000000	91.8816815	91.8531641	91.8569301	91.9755221	92.1315893	92.2591385	92.3277521	0.0000000
0.0000000	87.8530673	87.7393101	87.6521342	87.9176617	88.2305223	88.4600398	88.5758019	0.0000000
0.0000000	83.9382225	83.5988908	83.0946482	83.8124811	84.4128118	84.7747128	84.9396259	0.0000000
0.0000000	80.3627221	79.6233998	77.3151005	79.8248158	80.8335448	81.2863911	81.4683757	0.0000000
0.0000000	77.5265176	77.2168501	76.7175099	77.3381095	77.8101322	78.0688950	78.1790838	0.0000000
0.0000000	75.0000003	75.0000003	75.0000002	75.0000003	75.0000003	75.0000004	75.0000004	0.0000000
0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000

HEAD DISTRIBUTION (BY ROW)

Number of time steps = 1
Time (seconds) = 0.78894E+08
Time (days) = 0.91313E+03
Time (years) = 0.25000E+01

0	0	0	0	0	0	0	0	0
0	100	100	100	100	100	100	100	0
0	96	96	96	96	96	96	96	0
0	92	92	92	92	92	92	92	0
0	88	88	88	88	88	88	89	0
0	84	84	83	84	84	85	85	0
0	80	80	77	80	81	81	81	0
0	78	77	77	77	78	78	78	0
0	75	75	75	75	75	75	75	0
0	0	0	0	0	0	0	0	0

Cumulative mass balance -- (IN L**3)

Recharge and injection = 0.00000E+00
Pumpage and E-T withdrawal = -0.78894E+08
Cumulative net pumpage = -0.78894E+08
Water release from storage = 0.00000E+00
Leakage into aquifer = 0.21978E+09
Leakage out of aquifer = -0.14088E+09
Cumulative net leakage = 0.78895E+08

Mass balance residual = 763.91
Error (as percent) = 0.34759E-03

RATE MASS BALANCE -- (IN L**3/SEC)

Leakage into aquifer = 0.27857E+01
Leakage out of aquifer = -0.17857E+01
Net leakage (QNET) = 0.10000E+01
Recharge and injection = 0.00000E+00
Pumpage and E-T withdrawal = -0.10000E+01
Net withdrawal (TPUM) = -0.10000E+01

X VELOCITIES

AT NODES

0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00
0.000E+00	4.673E-15	-9.243E-15	-6.994E-14	-1.306E-13	-1.386E-13	-9.961E-14	-3.561E-14	0.000E+00
0.000E+00	3.785E-08	-7.486E-08	-5.665E-07	-1.058E-06	-1.123E-06	-8.069E-07	-2.885E-07	0.000E+00
0.000E+00	2.640E-07	2.292E-07	-1.133E-06	-2.543E-06	-2.626E-06	-1.816E-06	-6.353E-07	0.000E+00
0.000E+00	1.053E-06	1.860E-06	-1.651E-06	-5.355E-06	-5.022E-06	-3.197E-06	-1.072E-06	0.000E+00
0.000E+00	3.142E-06	7.811E-06	-1.978E-06	-1.221E-05	-8.910E-06	-4.878E-06	-1.527E-06	0.000E+00
0.000E+00	6.846E-06	2.822E-05	-1.865E-06	-3.258E-05	-1.353E-05	-5.878E-06	-1.685E-06	0.000E+00
0.000E+00	2.867E-06	7.491E-06	-1.123E-06	-1.012E-05	-6.767E-06	-3.416E-06	-1.020E-06	0.000E+00
0.000E+00	3.540E-13	9.248E-13	-1.386E-13	-1.249E-12	-8.354E-13	-4.218E-13	-1.260E-13	0.000E+00
0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00

Y VELOCITIES

AT NODES

0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00
0.000E+00	3.760E-05	3.764E-05	3.753E-05	3.708E-05	3.647E-05	3.595E-05	3.566E-05	0.000E+00
0.000E+00	7.517E-05	7.543E-05	7.540E-05	7.430E-05	7.286E-05	7.167E-05	7.104E-05	0.000E+00
0.000E+00	7.487E-05	7.588E-05	7.680E-05	7.480E-05	7.251E-05	7.090E-05	7.012E-05	0.000E+00
0.000E+00	7.355E-05	7.643E-05	8.113E-05	7.558E-05	7.147E-05	6.930E-05	6.841E-05	0.000E+00
0.000E+00	6.936E-05	7.515E-05	9.571E-05	7.493E-05	6.849E-05	6.642E-05	6.581E-05	0.000E+00
0.000E+00	5.937E-05	5.909E-05	5.905E-05	5.995E-05	6.114E-05	6.209E-05	6.260E-05	0.000E+00
0.000E+00	4.965E-05	4.281E-05	2.144E-05	4.467E-05	5.401E-05	5.821E-05	5.989E-05	0.000E+00
0.000E+00	2.339E-05	2.053E-05	1.590E-05	2.165E-05	2.602E-05	2.842E-05	2.944E-05	0.000E+00
0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00

STABILITY CRITERIA --- M.O.C.

Maximum fluid velocities (L/T): X-VEL = 4.65E-05 Y-VEL = 1.07E-04

Maximum effective solute velocities: X-VEL = 2.79E-05 Y-VEL = 6.42E-05

TMV (MAX. INJ.) = 0.19925E+08
TIMV (CELDIS) = 0.70075E+07

TIMV = 7.01E+06 NTIMV = 11 NMOV = 12

TIM (N) = 0.78894E+08
TIMEVELO = 0.65745E+07
TIMEDISP = 0.45091E+08

TIMV = 6.57E+06 NTIMD = 1 NMOV = 12

THE LIMITING STABILITY CRITERION IS CELDIS
MAX. Y-VEL. IS CONSTRAINT AND OCCURS BETWEEN NODES (4, 6) AND (4, 7)

NO. OF PARTICLE MOVES REQUIRED TO COMPLETE THIS TIME STEP = 12

NP = 315 IMOV = 1
TIM(N) = 0.78894E+08 TIMV = 0.65745E+07 SUMTCH = 0.65745E+07

NP = 322 IMOV = 2
TIM(N) = 0.78894E+08 TIMV = 0.65745E+07 SUMTCH = 0.13149E+08

NP = 338 IMOV = 3
TIM(N) = 0.78894E+08 TIMV = 0.65745E+07 SUMTCH = 0.19724E+08

NP = 338 IMOV = 4
TIM(N) = 0.78894E+08 TIMV = 0.65745E+07 SUMTCH = 0.26298E+08

•
•
•

```
                                        .
                                        .
                                        .

  NP       =          387          IMOV     =           10
  TIM(N)   =  0.78894E+08          TIMV     =  0.65745E+07          SUMTCH =  0.65745E+08


CONCENTRATION

Number of time steps =     1
      Delta t        =  0.78894E+08
      Time (seconds) =  0.78894E+08
 Chem.time (seconds) =  0.65745E+08
 Chem.time (days)    =  0.76094E+03
      Time (years)   =  0.25000E+01
 Chem.time (years)   =  0.20833E+01
 No. moves completed =    10


    2   96   97   95    1

    3   90   93   90    3

    4   76   80   75    4

    3   49   51   38    3

    1    9    9    6    0

    0    2    1    1    0

    0    0    0    0    0

CHEMICAL MASS BALANCE

       Mass in boundaries     =  0.78884E+10
       Mass out boundaries    = -0.18207E+06
       Mass pumped in         =  0.00000E+00
       Mass pumped out        = -0.11296E+08
       Mass lost by decay     =  0.00000E+00
       Mass adsorbed on solids=  0.31867E+10
       Initial mass adsorbed  =  0.00000E+00
       Inflow minus outflow   =  0.78769E+10
       Initial mass dissolved =  0.00000E+00
       Present mass dissolved =  0.47801E+10
       Change mass dissolved  =  0.47801E+10
       Change totl.mass stored=  0.79668E+10
    COMPARE RESIDUAL WITH NET FLUX AND MASS ACCUMULATION:
       Mass balance residual  = -0.89924E+08
       Error  (as percent)    = -0.11400E+01

  NP       =          387          IMOV     =           11
  TIM(N)   =  0.78894E+08          TIMV     =  0.65745E+07          SUMTCH =  0.72320E+08

  NP       =          403          IMOV     =           12
  TIM(N)   =  0.78894E+08          TIMV     =  0.65745E+07          SUMTCH =  0.78894E+08
```

```
CONCENTRATION

Number of time steps =     1
      Delta t        =  0.78894E+08
      Time (seconds) =  0.78894E+08
 Chem.time (seconds) =  0.78894E+08
 Chem.time (days)    =  0.91313E+03
      Time (years)   =  0.25000E+01
 Chem.time (years)   =  0.25000E+01
 No. moves completed =    12


    2   97   98   97    2

    3   94   97   93    3

    5   84   88   82    5

    6   67   72   62    5

    3   34   38   24    2

    1    7    7    3    0

    0    1    0    0    0

CHEMICAL MASS BALANCE

       Mass in boundaries     =  0.94684E+10
       Mass out boundaries    = -0.16040E+07
       Mass pumped in         =  0.00000E+00
       Mass pumped out        = -0.61552E+08
       Mass lost by decay     =  0.00000E+00
       Mass adsorbed on solids=  0.38305E+10
       Initial mass adsorbed  =  0.00000E+00
       Inflow minus outflow   =  0.94053E+10
       Initial mass dissolved =  0.00000E+00
       Present mass dissolved =  0.57457E+10
       Change mass dissolved  =  0.57457E+10
       Change totl.mass stored=  0.95762E+10
     COMPARE RESIDUAL WITH NET FLUX AND MASS ACCUMULATION:
       Mass balance residual  = -0.17093E+09
       Error  (as percent)    = -0.18053E+01
```

Problem 3 (Steady flow, 1 well, constant-head boundaries, linear sorption)

Time versus head and concentration at selected observation points

PUMPING PERIOD NO. 1

STEADY-STATE SOLUTION

OBS.WELL NO.	X	Y	N	HEAD	CONCENTRATION	TIME (YEARS)
1	5	4				
			0	0.0	0.0	0.000
			1	92.0	0.0	0.208
			2	92.0	0.2	0.417
			3	92.0	1.2	0.625
			4	92.0	2.7	0.833
			5	92.0	14.4	1.042
			6	92.0	36.5	1.250
			7	92.0	51.2	1.458
			8	92.0	60.2	1.667
			9	92.0	73.8	1.875
			10	92.0	80.2	2.083
			11	92.0	83.2	2.292
			12	92.0	88.4	2.500
OBS.WELL NO.	X	Y	N	HEAD	CONCENTRATION	TIME (YEARS)
2	5	7				
			0	0.0	0.0	0.000
			1	79.8	0.0	0.208
			2	79.8	0.0	0.417
			3	79.8	0.0	0.625
			4	79.8	0.0	0.833
			5	79.8	0.0	1.042
			6	79.8	0.0	1.250
			7	79.8	0.1	1.458
			8	79.8	0.2	1.667
			9	79.8	0.5	1.875
			10	79.8	1.5	2.083
			11	79.8	3.6	2.292
			12	79.8	6.8	2.500

OPTIONAL OUTPUT FILES

prob3.o1

```
      "OBSERVATION WELL DATA"
"NODE  (I,J):  (  5,  4)
"TIME(YRS)       HEAD         CONC."
 0.0000E+00  0.0000E+00  0.0000E+00
 2.0833E-01  9.1976E+01  0.0000E+00
 4.1667E-01  9.1976E+01  1.9696E-01
 6.2500E-01  9.1976E+01  1.2238E+00
 8.3333E-01  9.1976E+01  2.7345E+00
 1.0417E+00  9.1976E+01  1.4441E+01
 1.2500E+00  9.1976E+01  3.6470E+01
 1.4583E+00  9.1976E+01  5.1200E+01
 1.6667E+00  9.1976E+01  6.0175E+01
 1.8750E+00  9.1976E+01  7.3812E+01
 2.0833E+00  9.1976E+01  8.0172E+01
 2.2917E+00  9.1976E+01  8.3248E+01
 2.5000E+00  9.1976E+01  8.8431E+01
```

prob3.o2

```
    "OBSERVATION WELL DATA"
"NODE  (I,J):  (  5,  7)
"TIME(YRS)       HEAD         CONC."
 0.0000E+00  0.0000E+00  0.0000E+00
 2.0833E-01  7.9825E+01  0.0000E+00
 4.1667E-01  7.9825E+01  0.0000E+00
 6.2500E-01  7.9825E+01  0.0000E+00
 8.3333E-01  7.9825E+01  3.7578E-04
 1.0417E+00  7.9825E+01  3.3789E-03
 1.2500E+00  7.9825E+01  1.8039E-02
 1.4583E+00  7.9825E+01  6.7213E-02
 1.6667E+00  7.9825E+01  1.7876E-01
 1.8750E+00  7.9825E+01  5.1765E-01
 2.0833E+00  7.9825E+01  1.4993E+00
 2.2917E+00  7.9825E+01  3.5846E+00
 2.5000E+00  7.9825E+01  6.7915E+00
```

prob3.cn1 (matrix format; IFMT = 1)

```
 1.630E+00 9.697E+01 9.848E+01 9.687E+01 1.617E+00
 3.275E+00 9.357E+01 9.656E+01 9.344E+01 3.172E+00
 5.410E+00 8.366E+01 8.843E+01 8.249E+01 5.166E+00
 5.633E+00 6.690E+01 7.173E+01 6.152E+01 4.709E+00
 3.244E+00 3.421E+01 3.803E+01 2.442E+01 2.027E+00
 5.132E-01 6.839E+00 6.791E+00 3.212E+00 3.127E-01
 3.178E-02 5.257E-01 4.388E-01 3.816E-01 2.891E-02
```

prob3.cn1 (column format; IFMT = 0)

```
2.250E+03   7.650E+03   1.630E+00
3.150E+03   7.650E+03   9.697E+01
4.050E+03   7.650E+03   9.848E+01
4.950E+03   7.650E+03   9.687E+01
5.850E+03   7.650E+03   1.617E+00
2.250E+03   6.750E+03   3.275E+00
3.150E+03   6.750E+03   9.357E+01
4.050E+03   6.750E+03   9.656E+01
4.950E+03   6.750E+03   9.344E+01
5.850E+03   6.750E+03   3.172E+00
2.250E+03   5.850E+03   5.410E+00
3.150E+03   5.850E+03   8.366E+01
4.050E+03   5.850E+03   8.843E+01
4.950E+03   5.850E+03   8.249E+01
5.850E+03   5.850E+03   5.166E+00
2.250E+03   4.950E+03   5.633E+00
3.150E+03   4.950E+03   6.690E+01
4.050E+03   4.950E+03   7.173E+01
4.950E+03   4.950E+03   6.152E+01
5.850E+03   4.950E+03   4.709E+00
2.250E+03   4.050E+03   3.244E+00
3.150E+03   4.050E+03   3.421E+01
4.050E+03   4.050E+03   3.803E+01
4.950E+03   4.050E+03   2.442E+01
5.850E+03   4.050E+03   2.027E+00
2.250E+03   3.150E+03   5.132E-01
3.150E+03   3.150E+03   6.839E+00
4.050E+03   3.150E+03   6.791E+00
4.950E+03   3.150E+03   3.212E+00
5.850E+03   3.150E+03   3.127E-01
2.250E+03   2.250E+03   3.178E-02
3.150E+03   2.250E+03   5.257E-01
4.050E+03   2.250E+03   4.388E-01
4.950E+03   2.250E+03   3.816E-01
5.850E+03   2.250E+03   2.891E-02
```

APPENDIX 4: COMPUTER PROGRAM MODIFICATIONS

Following are the changes (deletions and insertions) that must be made to update MOC to Version 3.1. These instructions assume a starting point of the previous update of March, 1989, as documented by Goode and Konikow (1989). Line numbers appear in columns 73-80 for reference.

FILE A: Previous version updated through July 20, 1989
FILE B: New modified version

```
INSERT
      B:  C     *  INPUT/OUTPUT REVISIONS 6/92-2/93 (L.KONIKOW AND G.GRANATO)  *     A  69R
      B:  C     *  INPUT/OUTPUT REVISIONS 6/93-7/93 (L.KONIKOW AND S.LESSOFF)  *     A  69S
      B:  C     *   I/O REVISIONS 9/93-2/94 (G.Z.Hornberger & L.F. KONIKOW)    *     A  69T
      B:  C     *                                                              *     A  69U

DELETE
      A:        DOUBLE PRECISION DMIN1,DEXP,DLOG,DABS                                A  80

INSERT
      B:        LOGICAL EXISTS                                                       A  75

INSERT
      B:        CHARACTER*43 OPF                                                     A 117B
      B:        CHARACTER*36 OPE                                                     A 117C

REPLACE
      A:        line A 142
WITH
      B:       2              NPNTMV,NPNTVL,NPNTD,NPNCHV,NPDELC,ICHK,IFMT          A 142a
      B:       3              ,NOBSO,NHEADO,NCONCO,NPARMO,NOUTFL                   A 143

REPLACE
      A:        line A 239
WITH
      B:        COMMON /DIFUS/ DISP(020,020,4)                                      A 239A
      B:        COMMON /PRPT/ GX1,GX2,GX3,GEGT                                      A 239B
      B:        COMMON /FLMGR/IOF(32),ICFLAG,IVIN,IRPT                              A 239C
      B:        COMMON /BOUT/ICNFILE,NUMFILE                                        A 239D
      B:        COMMON /FLNMS/OPF(31),OPE(31)                                       A 239E
      B:  C    ****************************************************************     A 239F
      B:        CALL CLEAR                                                          A 239G
      B:        CALL HOME                                                           A 239H
      B:  C    IF FILE MOCBAT.DAT EXISTS THIS PROGRAM WILL RUN IN BATCH MODE. THE   A 239I
      B:  C    FILE HAS A HEADER WITH THE NUMBER OF INPUT FILES TO FOLLOW,          A 239J
      B:  C    THEREFORE IT HAS THE REQUIRED NUMBER OF INPUT FILE NAMES             A 239K
      B:  C                                                                        A 239L
      B:        NUMFILE=0                                                          A 239M
      B:        ICNFILE=0                                                          A 239N
      B:        ICAP=0                                                             A 2390
      B:  C     UNIT 6 SHOULD BE PRINTING TO SCREEN: "*" IS GENERALLY PRINT TO     A 239P
      B:  C     SCREEN. IRPT=6 PRINTS TO SCREEN INTERACTIVELY OR TO MOCBAT.RPT     A 239Q
      B:  C     IN BATCH MODE.                                                     A 239R
      B:        IRPT=6                                                             A 239S
      B:        INQUIRE (FILE='MOCBAT.DAT', EXIST=EXISTS)                          A 239T
      B:        IF (EXISTS) ICAP=1                                                 A 239U
      B:        INQUIRE (FILE='mocbat.dat', EXIST=EXISTS)                          A 239V
      B:        IF (EXISTS) ICAP=2                                                 A 239W
      B:        IF (ICAP.NE.0) THEN                                                A 239X
      B:         ICFLAG=1                                                          A 239Y
      B:         IF (ICAP.EQ.1)                                                    A 239Z
```

```
      B:       + OPEN(UNIT=45,FILE='MOCBAT.DAT',ACCESS='SEQUENTIAL',STATUS='OLD')   A 239A
      B:         IF (ICAP.EQ.2)                                                     A 241B
      B:       + OPEN(UNIT=45,FILE='mocbat.dat',ACCESS='SEQUENTIAL',STATUS='OLD')   A 241C
      B:         OPEN(UNIT=46,FILE='mocbat.rpt')                                    A 241D
      B:         IRPT=46                                                            A 241E
      B:         READ(45,*) NUMFILE                                                 A 241F
      B:  C      STATUS MESSAGE FOR PC                                              A 241G
      B:         WRITE(IRPT,*) 'USGS MOC model, Version 3.1 (BATCH MODE)'           A 241H
      B:         WRITE(*,*) 'USGS MOC model, Version 3.1 (RUNNING BATCH MODE)'      A 241I
      B:         WRITE (*,750)                                                      A 241J
      B:         WRITE (*,731)                                                      A 241K
      B:        ELSE                                                                A 241L
      B:         ICFLAG=0                                                           A 241M
      B:         NUMFILE=1                                                          A 241N
      B:        END IF                                                              A 241O
      B:    100 IF (ICNFILE.EQ.NUMFILE) GOTO 1000                                   A 241P
      B:  C     RESET PUMPING PERIOD BETWEEN INFILES                                A 241Q
      B:  C     CALL SUBROUTINE TO PRINT SCREENS AND SET UP FILE INTERFACES         A 241R
      B:        CALL FILMGR                                                         A 241S
      B:        IF (ICNFILE.EQ.NUMFILE) GOTO 1000                                   A 241T

REPLACE
      A:        lines A 702 - A 705
WITH
      B:        IF (NPNCHV.EQ.0) GO TO 155                                          A 703a
      B:    155 CONTINUE                                                          A 705a
      B:        ICNFILE=ICNFILE+1                                                   A 706
      B:        CLOSE (IVIN)                                                        A 707
      B:        GOTO 100                                                            A 708A
      B:   1000 IF (ICFLAG.EQ.1) CLOSE (46)                                         A 708B
      B:        IF (ICFLAG.EQ.1) GOTO 494                                           A 708C
      B:        WRITE (*,550)                                                       A 708D
      B:    494 CONTINUE                                                          A 709

INSERT
      B:    550 FORMAT(//,4X,'Normal completion of calculations for MOC model. '  A 724
      B:       1///)                                                              A 726
      B:    731 FORMAT (/,' Batch mode report written to "mocbat.rpt"')           A 726A
      B:    750 FORMAT(/01X,77HU.S.G.S. Method-of-characteristics model for solute A 727
      B:       1 transport in ground water)                                       A 728

DELETE
      A:        DOUBLE PRECISION DMIN1,DEXP,DLOG,DABS                             B  20

INSERT
      B:        CHARACTER*43 OPF                                                  B  21
      B:        CHARACTER*36 OPE                                                  B  22
      B:        CHARACTER*24 GSRNG                                                B  23
      B:        CHARACTER*2 PSRNG                                                 B  24

REPLACE
      A:        line B  92
WITH
      B:       2              NPNTMV,NPNTVL,NPNTD,NPNCHV,NPDELC,ICHK,IFMT        B  92a
      B:       3              ,NOBSO,NHEADO,NCONCO,NPARMO,NOUTFL                 B  93

INSERT
      B:        COMMON /FLMGR/IOF(32),ICFLAG,IVIN,IRPT                            B 212A
      B:        COMMON /FLNMS/OPF(31),OPE(31)                                     B 212B

REPLACE
      A:        lines B 240 - B 260
WITH
      B:        READ (IVIN,720) TITLE                                             B 250a
```

```
INSERT
      B:        NPMAX=6400                                                          B 285

REPLACE
      A:        lines B 420 - B 431A
WITH
      B:        READ (IVIN,740) NTIM,NPMP,NX,NY,NPNT,NITP,NUMOBS,ITMAX,NREC,        B 420N
      B:       1NPTPND,NCODES,NPNTMV,NPNTVL,NPNTD,NPDELC,NPNCHV,IREACT,NOUTFL       B 431a
      B: C      OPEN FILES AND BEGIN WRITING OUTPUT AS READ INPUT                   B 431b

REPLACE
      A:        line SB 438D
WITH
      B:        READ (IVIN,*) MX,MY,MMX,MMY                                        SB 438d

REPLACE
      A:        lines B 440 - B 450
WITH
      B:        READ (IVIN,800) PINT,TOL,POROS,BETA,S,TIMX,TINIT,XDEL,YDEL,DLTRAT,  B 441
      B:       1CELDIS,ANFCTR                                                      B 451
      B:        YMAX=(NY*YDEL)-YDEL                                                B 452

REPLACE
      A:        line B 456R
WITH
      B:        IF (IREACT.EQ.1) READ (IVIN,*) DK,RHOB,THALF                       B 456r

REPLACE
      A:        lines B 458R - B 462R
WITH
      B:        IF (IREACT.EQ.-1) READ (IVIN,*) THALF                              B 458r
      B:        IF (IREACT.EQ.2) READ (IVIN,*) RHOB,EKF,XNF,THALF                  B 459r
      B:        IF (IREACT.EQ.3) READ (IVIN,*) RHOB,EKL,CEC,THALF                  B 461r
      B:        IF (IREACT.GE.4) READ (IVIN,*) RHOB,EK,CEC,CTOT,THALF              B 462r

INSERT
      B: C         ---READ OPTIONS FOR OUTPUT FILES IF NOUTFL.GT.0                  B 535
      B:        IF (NOUTFL.EQ.0) GO TO 6                                           B 536
      B:        READ (IVIN,741) NOBSO,NHEADO,NCONCO,NPARMO,IFMT                    B 537
      B:      6 CONTINUE                                                           B 538
      B:        WRITE (IRPT,750)                                                   B 539A
      B:        WRITE (IRPT,730) TITLE                                             B 539B
      B:        CALL OFLMGR                                                        B 539C
      B:        WRITE (IOF(1),750)                                                 B 539D
      B:        WRITE (IOF(1),730) TITLE                                           B 539E

REPLACE
      A:        lines SB 545 - B 590
WITH
      B:        WRITE (IOF(1),755)                                                SB 545a
      B:        WRITE (IOF(1),760)                                                 B 550a
      B:        WRITE (IOF(1),770) NX,NY,XDEL,YDEL                                 B 560a
      B:        IF (MCHK.GT.0) WRITE (IOF(1),775) NMX,NMY,MX,MY,MMX,MMY           SB 565a
      B:        WRITE (IOF(1),780) NTIM,NPMP,PINT,TIMX,TINIT                       B 570a
      B:        WRITE (IOF(1),790) S,POROS,BETA,DLTRAT,ANFCTR                      B 580a
      B:        WRITE (IOF(1),870) NITP,TOL,ITMAX,CELDIS,NPTPND                    B 590a

REPLACE
      A:        lines B 612 - B 631R
WITH
      B:       1.AND.NPTPND.NE.16.AND.NPTPND.NE.1) WRITE(IOF(1),880)               B 612a
      B:        IF (NPTPND.EQ.1) WRITE (IOF(1),882)                                B 614a
      B:        IF((NX.EQ.3.OR.NY.EQ.3).AND.NPTPND.NE.1) WRITE(IOF(1),883)         B 614b
      B:        IF (NITP.LE.0) WRITE (IOF(1),885)                                  B 615a
      B:        WRITE (IOF(1),890) NPNT,NPNTMV,NPNTVL,NPNTD,NUMOBS,NREC,NCODES,    B 621
```

```
      B:      1NPDELC,IREACT,NOUTFL                                                B 631r

REPLACE
      A:      line B 632B
WITH
      B:       WRITE (IOF(1),891) REACTN(9)                                        B 632b

REPLACE
      A:      line B 632E
WITH
      B:       WRITE (IOF(1),891) REACTN(IREACT+1)                                 B 632e

REPLACE
      A:      line B 634R
WITH
      B:       WRITE (IOF(1),892) RHOB                                             B 634r

REPLACE
      A:      line B 637R
WITH
      B:        WRITE (IOF(1),893) DK,RF                                           B 637r

REPLACE
      A:      lines B 639R - B 643R
WITH
      B:        IF (IREACT.EQ.2) WRITE (IOF(1),894) EKF,XNF                        B 639r
      B:        IF (IREACT.EQ.3) WRITE (IOF(1),895) EKL,CEC                        B 641r
      B:        IF (IREACT.GE.4) WRITE (IOF(1),896) EK,CEC,CTOT                    B 642r
      B:        IF (BETA.EQ.0.0) WRITE (IOF(1),897)                                B 643r

REPLACE
      A:      line B 646R
WITH
      B:      IF (DECAY.NE.0) WRITE (IOF(1),898) THALF,DECAY                       B 646r

INSERT
      B:      WRITE (IOF(1),1190)                                                  B 647A
      B:      WRITE (IOF(1),1140) NOBSO                                            B 647B
      B:      WRITE (IOF(1),1150) NHEADO                                           B 647C
      B:      WRITE (IOF(1),1160) NPNCHV                                           B 647D
      B:      WRITE (IOF(1),1170) NCONCO                                           B 647E
      B:      WRITE (IOF(1),1180) NPARMO                                           B 647F
      B:      WRITE (IOF(1),1185) IFMT                                             B 647G
      B:      IF (NOBSO.NE.0.OR.NHEADO.NE.0.OR.NCONCO.NE.0.OR.NPARMO.NE.0          B 647H
      B:     1.OR.NPNCHV.NE.0) THEN                                                B 647I
      B:       WRITE (IOF(1),1130)                                                B 647J
      B:       IF (NOBSO.EQ.1) WRITE (IOF(1),1120) OPE(2),OPF(2)                  B 647K
      B:       IF (NOBSO.EQ.2) THEN                                               B 647L
      B:         DO 51 J=1,NUMOBS                                                 B 647M
      B:           IFL=(11+J)                                                     B 647N
      B:           WRITE (IOF(1),1120) OPE(2),OPF(IFL)                            B 647O
      B:   51    CONTINUE                                                         B 647P
      B:         END IF                                                           B 647Q
      B:       IF (NHEADO.EQ.-1.OR.NHEADO.EQ.2)                                   B 648A
      B:      1   WRITE (IOF(1),1120) OPE(3),OPF(3)                              B 648B
      B:       IF (NHEADO.EQ.1.OR.NHEADO.EQ.2)                                    B 648C
      B:      1   WRITE (IOF(1),1120) OPE(10),OPF(10)                            B 648D
      B:       IF (NCONCO.EQ.-1.OR.NCONCO.EQ.2)                                   B 648E
      B:      1   WRITE (IOF(1),1120) OPE(5),OPF(5)                              B 648F
      B:       IF (NCONCO.EQ.1.OR.NCONCO.EQ.2)                                    B 648G
      B:      1   WRITE (IOF(1),1120) OPE(11),OPF(11)                            B 648H
      B:       IF (NPNCHV.NE.0) WRITE (IOF(1),1120) OPE(4),OPF(4)                 B 648I
      B:       IF (NPARMO.NE.0) THEN                                              B 648J
      B:        DO 7 I=6,9                                                        B 648K
      B:         WRITE (IOF(1),1120) OPE(I),OPF(I)                               B 648L
```

```
      B:     7    CONTINUE                                                    B 648M
      B:           END IF                                                     B 648N
      B:          END IF                                                      B 648O

REPLACE
      A:          lines B 690 - B 695
WITH
      B:     10 READ (IVIN,1060) ICHK                                         B 690a
      B:          IF (ICHK.LE.0) WRITE (IOF(1),1110) INT                     B 695a

REPLACE
      A:          lines B 710 - B 750
WITH
      B:          READ (IVIN,1070) NTIM,NPNT,NITP,ITMAX,NREC,NPNTMV,NPNTVL,NPNTD,NPD  B 710a
      B:         1ELC,NPNCHV,PINT,TIMX,TINIT                                 B 720a
      B:          WRITE (IOF(1),1080) INT                                    B 730a
      B:          WRITE (IOF(1),1090) NTIM,NPNT,NITP,ITMAX,NREC,NPNTMV,NPNTVL,NPNTD,  B 740a
      B:         1NPDELC,NPNCHV,PINT,TIMX,TINIT                              B 750a

INSERT
      B:          TTOT=TINIT                                                 B 812

REPLACE
      A:          lines B 840 - B 865
WITH
      B:          TIM(K)=TIMX*TIM(K-1)                                       B 840a
      B:     40 TTOT=(TTOT+TIM(K))                                           B 845
      B:          WRITE (IOF(1),470)                                         B 850a
      B:          WRITE (IOF(1),490) (TIM(K),K=1,NTIM)                       B 861a
      B:          IF (TINIT.GT.PYR) WRITE (IOF(1),475)                       B 865a
      B:          IF (TTOT.LT.PYR) WRITE (IOF(1),477)                        B 867

REPLACE
      A:          lines B 891 - B 892
WITH
      B:           WRITE (IOF(1),480)                                        B 891a
      B:           WRITE (IOF(1),490) (TIM(K),K=1,NTIM)                      B 892a

REPLACE
      A:          lines B1195 - B1200
WITH
      B:           WRITE (IOF(1),755)                                        B1195a
      B:          WRITE (IOF(1),900)                                         B1200a

REPLACE
      A:          lines B1220 - B1230
WITH
      B:          READ (IVIN,700) IX,IY                                      B1220a
      B:          WRITE (IOF(1),810) J,IX,IY                                 B1230a

REPLACE
      A:          line B1310
WITH
      B:  C       ---READ PUMPAGE DATA--(X-Y COORDINATES AND RATE IN L**3/SEC)---  B1311

REPLACE
      A:          line B1346B
WITH
      B:           IF (IREACT.GE.2.AND.IREACT.LE.7) WRITE (IOF(1),899) RF2MIN  B1346b

REPLACE
      A:          lines B1347 - B1350
WITH
      B:          WRITE (IOF(1),755)                                         B1347a
      B:          WRITE (IOF(1),910)                                         B1350a
```

```
REPLACE
       A:        line B1370
WITH
       B:        READ (IVIN,710) IX,IY,FCTR,CNREC                                  B1370a

REPLACE
       A:        lines SB1382 - B1384R
WITH
       B:        IF (FCTR.LE.0.0) CNRECH(JX,JY)=CNREC                              SB1382
       B:        IF (FCTR.LE.0.0.AND.CNREC.GT.CINMAX) CINMAX=CNREC                 B1384R

 REPLACE
       A:        line SB1402
WITH
       B:    110 WRITE (IOF(1),820) IX,IY,REC(IX,IY),CNREC                         SB1402a

REPLACE
       A:        line B1422R
WITH
       B:         IF (IREACT.GE.2.AND.IREACT.LE.7) WRITE (IOF(1),899) RF2MIN       B1422r

REPLACE
       A:        lines B1435 - B1470
WITH
       B:        WRITE (IOF(1),755)                                                B1435a
       B:        WRITE (IOF(1),690) AREA                                           B1440a
       B:        WRITE (IOF(1),600)                                                B1450a
       B:        WRITE (IOF(1),610) XDEL                                           B1460a
       B:        WRITE (IOF(1),610) YDEL                                           B1470a

REPLACE
       A:        lines B1490 - B1520
WITH
       B:  C     ---READ TRANSMISSIVITY IN L**2/SEC INTO VPRM ARRAY---             B1491
       B:  C        ---FCTR = TRANSMISSIVITY MULTIPLIER  --->  L**2/SEC---         B1501
       B:        WRITE (IOF(1),530)                                                B1510a
       B:        WRITE (IOF(1),755)                                                B1515a
       B:        READ (IVIN,550) INPUT,FCTR                                        B1520a

REPLACE
       A:        line B1540
WITH
       B:        YYY=(IY*YDEL)-1.5*YDEL                                            B1535
       B:        IF (INPUT.EQ.1) READ (IVIN,560) (VPRM(IX,IY),IX=1,NX)             B1540a

INSERT
       B:        XXX=(IX*XDEL)-.5*XDEL                                             B1615A
       B:        IF (NPARMO.NE.0) THEN                                             B1615B
       B:         IF (IX.GT.1.AND.IX.LT.NX.AND.IY.GT.1.AND.IY.LT.NY) THEN          B1615C
       B:            IF (IFMT.EQ.0) THEN                                           B1615D
       B:               WRITE (IOF(6),841) XXX,(YMAX-YYY),VPRM(IX,IY)              B1615E
       B:            ELSE                                                          B1615F
       B:               WRITE (IOF(6),842) VPRM(IX,IY)                             B1615G
       B:                 IF (IX.EQ.(NX-1)) THEN                                   B1615H
       B:               WRITE (IOF(6),843)                                         B1615I
       B:                 END IF                                                   B1615J
       B:            END IF                                                        B1615K
       B:         END IF                                                           B1615L
       B:        END IF                                                            B1615M

REPLACE
       A:        line B1631
WITH
       B:    160 WRITE (IOF(1),840) (VPRM(IX,IY),IX=1,NX)                          B1631a
```

```
REPLACE
      A:        line B1680
WITH
      B:        WRITE (IOF(1),1050)                                          B1680a

REPLACE
      A:        lines B1940 - B1950
WITH
      B:        WRITE (IOF(1),510)                                           B1940a
      B:        WRITE (IOF(1),755)                                           B1945a
      B:        READ (IVIN,550) INPUT,FCTR                                   B1950a

REPLACE
      A:        line B1970
WITH
      B:        YYY=(IY*YDEL)-1.5*YDEL                                       B1965
      B:        IF (INPUT.EQ.1) READ (IVIN,540) (THCK(IX,IY),IX=1,NX)        B1970a

INSERT
      B:        XXX=(IX*XDEL)-.5*XDEL                                        B1985

REPLACE
      A:        line B2010
WITH
      B:        GO TO 199                                                    B2010a

INSERT
      B:    199 IF (NPARMO.NE.0) THEN                                        B2025A
      B:         IF (IX.GT.1.AND.IY.GT.1.AND.IX.LT.NX.AND.IY.LT.NY) THEN     B2025B
      B:            IF (IFMT.EQ.0) THEN                                     B2025C
      B:               WRITE (IOF(7),841) XXX,(YMAX-YYY),THCK(IX,IY)        B2025D
      B:            ELSE                                                    B2025E
      B:               WRITE (IOF(7),842) THCK(IX,IY)                       B2025F
      B:                 IF (IX.EQ.(NX-1)) THEN                             B2025G
      B:               WRITE (IOF(7),843)                                   B2025H
      B:                 END IF                                             B2025I
      B:            END IF                                                  B2025J
      B:         END IF                                                     B2025K
      B:        END IF                                                      B2025L

REPLACE
      A:        line B2041
WITH
      B:        WRITE (IOF(1),500) (THCK(IX,IY),IX=1,NX)                     B2041a

REPLACE
      A:        lines B2070 - B2080
WITH
      B:        READ (IVIN,550) INPUT,FCTR                                   B2080a

REPLACE
      A:        line B2100
WITH
      B:        IF (INPUT.EQ.1) READ (IVIN,560) (RECH(IX,IY),IX=1,NX)        B2100a

REPLACE
      A:        line B2170
WITH
      B:    240 CONTINUE                                                     B2171

REPLACE
      A:        line B2190
WITH
      B:  C     ---COMPUTE HYDRAULIC CONDUCTIVITY FROM TRANSMISSIVITY---     B2191
```

```
REPLACE
      A:        lines B2310 - B2315
WITH
      B:        WRITE (IOF(1),620)                                                    B2310a
      B:        WRITE (IOF(1),755)                                                    B2315a

REPLACE
      A:        line B2322
WITH
      B:        YYY=(IY*YDEL)-1.5*YDEL                                                B2321
      B:        WRITE (IOF(1),840) (VPRM(IX,IY),IX=1,NX)                              B2322a
      B:        IF (IY.GT.1.AND.IY.LT.NY) THEN                                        B2323A
      B:        DO 259 IX=2,(NX-1)                                                    B2323B
      B:         XXX=(IX*XDEL)-.5*XDEL                                                B2323C
      B:         IF (IFMT.EQ.0) THEN                                                  B2323D
      B:          IF (NPARMO.NE.0) WRITE (IOF(9),841) XXX,(YMAX-YYY),VPRM(IX,IY)     B2323E
      B:          ELSE                                                               B2323F
      B:            WRITE (IOF(9),842) VPRM(IX,IY)                                   B2323G
      B:              IF (IX.EQ.(NX-1)) THEN                                         B2323H
      B:               WRITE (IOF(9),843)                                            B2323I
      B:              END IF                                                         B2323J
      B:            END IF                                                           B2323K
      B:    259 CONTINUE                                                             B2323L
      B:         END IF                                                              B2323M

REPLACE
      A:        lines SB2334 - SB2337
WITH
      B:        WRITE (IOF(1),633)                                                   SB2334a
      B:        WRITE (IOF(1),635) NCA,AAQ                                           SB2335a
      B:        WRITE (IOF(1),634)                                                   SB2336a
      B:        WRITE (IOF(1),630) NCA2,AAQ2,NZCRIT                                  SB2337a

REPLACE
      A:        line B2340
WITH
      B:        WRITE (IOF(1),630) NCA,AAQ,NZCRIT                                     B2340a

REPLACE
      A:        lines B2390 - B2400
WITH
      B:        WRITE (IOF(1),570)                                                    B2390a
      B:        READ (IVIN,550) INPUT,FCTR                                            B2400a

REPLACE
      A:        line B2420
WITH
      B:        IF (INPUT.EQ.1) READ (IVIN,640) (NODEID(IX,IY),IX=1,NX)               B2420a

REPLACE
      A:        lines B2450 - B2460
WITH
      B:    280 WRITE (IOF(1),580) (NODEID(IX,IY),IX=1,NX)                            B2450a
      B:        WRITE (IOF(1),920) NCODES                                             B2460a

REPLACE
      A:        line B2480
WITH
      B:        WRITE (IOF(1),930)                                                    B2480a

REPLACE
      A:        line B2500
WITH
      B:        READ (IVIN,850) ICODE,FCTR1,FCTR2,FCTR3,OVERRD                        B2500a
```

```
REPLACE
      A:        lines B2580 - B2615
WITH
      B:        WRITE (IOF(1),860) ICODE,FCTR1,FCTR2                          B2580a
      B:    300 IF (OVERRD.NE.0) WRITE (IOF(1),1100) FCTR3                    B2590a
      B:    310 WRITE (IOF(1),590)                                            B2600a
      B:        WRITE (IOF(1),755)                                            B2615a

REPLACE
      A:        line B2621
WITH
      B:    320 WRITE (IOF(1),840) (VPRM(IX,IY),IX=1,NX)                      B2621a

INSERT
      B:  C     ---PRINT DIFFUSE RECHARGE AND DISCHARGE---                    B2631
      B:        WRITE (IOF(1),830)                                            B2632
      B:        WRITE (IOF(1),755)                                            B2633
      B:        DO 323 IY=1,NY                                                B2634
      B:        IF (NPARMO.NE.0) THEN                                         B2634A
      B:         IF (INT.LE.1) THEN                                           B2634B
      B:         IF (IY.GT.1.AND.IY.LT.NY) THEN                               B2634C
      B:          YYY=(IY*YDEL)-1.5*YDEL                                      B2634D
      B:          DO 325 IX=2,(NX-1)                                          B2634E
      B:          XXX=(IX*XDEL)-.5*XDEL                                       B2634F
      B:            IF (IFMT.EQ.0) THEN                                       B2634G
      B:               WRITE (IOF(8),841) XXX,(YMAX-YYY),RECH(IX,IY)          B2634H
      B:            ELSE                                                      B2634I
      B:               WRITE (IOF(8),842) RECH(IX,IY)                         B2634J
      B:                 IF (IX.EQ.(NX-1)) THEN                               B2634K
      B:               WRITE (IOF(8),843)                                     B2634L
      B:                 END IF                                               B2634M
      B:             END IF                                                   B2634N
      B:    325 CONTINUE                                                      B2634O
      B:         ENDIF                                                        B2634P
      B:         ENDIF                                                        B2634Q
      B:        ENDIF                                                         B2634R
      B:    323 WRITE (IOF(1),840) (RECH(IX,IY),IX=1,NX)                      B2635
      B:  C     ************************************************************  B2639

REPLACE
      A:        lines B2650 - B2660
WITH
      B:        WRITE (IOF(1),670)                                            B2650a
      B:        WRITE (IOF(1),755)                                            B2655a
      B:        READ (IVIN,550) INPUT,FCTR                                    B2660a

REPLACE
      A:        line B2680
WITH
      B:        YYY=(IY*YDEL)-1.5*YDEL                                        B2675
      B:        IF (INPUT.EQ.1) READ (IVIN,660) (WT(IX,IY),IX=1,NX)           B2680a

REPLACE
      A:        line B2720
WITH
      B:        GO TO 335                                                     B2720a

INSERT
      B:    335 IF (IX.GT.1.AND.IX.LT.NX.AND.IY.GT.1.AND.IY.LT.NY) THEN       B2735A
      B:         IF (((NHEADO.EQ.-1).OR.(NHEADO.EQ.2)).AND.INT.LT.1) THEN     B2735B
      B:          XXX=(IX*XDEL)-.5*XDEL                                       B2735C
      B:            IF (IFMT.EQ.0) THEN                                       B2735D
      B:               WRITE (IOF(3),841) XXX,(YMAX-YYY),WT(IX,IY)            B2735E
      B:            ELSE                                                      B2735F
      B:               WRITE (IOF(3),842) WT(IX,IY)                           B2735G
```

```
      B:                IF (IX.EQ.(NX-1)) THEN                                B2735H
      B:              WRITE (IOF(3),843)                                      B2735I
      B:                END IF                                                B2735J
      B:            END IF                                                    B2735K
      B:         END IF      .                                                B2735L
      B:        END IF                                                        B2735M

REPLACE
      A:        line B2750
WITH
      B:    350 WRITE (IOF(1),680) (WT(IX,IY),IX=1,NX)                        B2750a

REPLACE
      A:        lines B2970 - B2981
WITH
      B:  C     (DECOMMENT FOLLOWING 2 LINES TO PRINT ITERATION PARAMETERS)   B2965
      B:  C     WRITE (IOF(1),450)                                            B2970a
      B:  C     WRITE (IOF(1),460) (AOPT(IP),IP=1,NITP)                       B2981a

REPLACE
      A:        line B3010
WITH
      B:        READ (IVIN,550) INPUT,FCTR                                    B3010a

REPLACE
      A:        line SB3031
WITH
      B:        YYY=JY*YDEL-1.5*YDEL                                          B3027
      B:        IF (INPUT.EQ.1) READ (IVIN,660) (CONC(IX,IY),IX=1,NMX)       SB3031a

REPLACE
      A:        line B3070
WITH
      B:        GO TO 395                                                     B3070a

INSERT
      B:    395 IF (JX.GT.1.AND.JX.LT.NX.AND.JY.GT.1.AND.JY.LT.NY) THEN      B3085A
      B:         IF (((NCONCO.EQ.-1).OR.(NCONCO.EQ.2)).AND.INT.LT.1) THEN     B3085B
      B:          XXX=(JX*XDEL)-.5*XDEL                                       B3085C
      B:            IF (IFMT.EQ.0) THEN                                       B3085D
      B:               WRITE (IOF(5),841) XXX,(YMAX-YYY),CONC(IX,IY)          B3085E
      B:            ELSE                                                      B3085F
      B:               WRITE (IOF(5),842) CONC(IX,IY)                         B3085G
      B:               NNN=NMX                                                B3085g
      B:                 IF (MCHK.GT.0) NNN=(NMX+1)                           B3085H
      B:               IF (IX.EQ.(NNN-1)) THEN                                B3085I
      B:                  WRITE (IOF(5),843)                                  B3085J
      B:               END IF                                                 B3085K
      B:            END IF                                                    B3085L
      B:          END IF                                                      B3085M
      B:        END IF                                                        B3085N

INSERT
      B:        PSRNG=' ('                                                    B3111B
      B:        IF (NOBSO.EQ.1) THEN                                          B3111C
      B:          WRITE (IOF(2),441)                                          B3111D
      B:          WRITE (IOF(2),442) (PSRNG,IXOBS(J),IYOBS(J),J=1,NUMOBS)     B3111E
      B:          WRITE (IOF(2),447)                                          B3111e
      B:          WRITE (IOF(2),446)                                          B3111F
      B:         DO 41 J=1,NUMOBS                                             B3111G
      B:          WRITE (IOF(2),445)                                          B3111H
      B:     41 CONTINUE                                                      B3111I
      B:          WRITE (IOF(2),447)                                          B3111J
      B:          WRITE (IOF(2),448) 0.,(WT(IXOBS(J),IYOBS(J)),CONINT(IXOBS(J) B3111K
      B:       1,IYOBS(J)), J=1,NUMOBS)                                       B3111L
```

```
      B:        END IF                                                          B3111M
      B:        IF (NOBSO.EQ.2) THEN                                            B3111N
      B:          DO 421 J=1,NUMOBS                                             B3111O
      B:           IFL=(11+J)                                                   B3111P
      B:           WRITE (IOF(IFL),443)                                         B3111Q
      B:           WRITE (IOF(IFL),444) PSRNG, IXOBS(J), IYOBS(J)               B3111R
      B:           WRITE (IOF(IFL),446)                                         B3111S
      B:           WRITE (IOF(IFL),445)                                         B3111T
      B:           WRITE (IOF(IFL),447)                                         B3111U
      B:           WRITE (IOF(IFL),448) 0.,WT(IXOBS(J),IYOBS(J)),CONINT(IXOBS(J B3111V
      B:         1),IYOBS(J))                                                   B3111W
      B:      421 CONTINUE                                                      B3111X
      B:        END IF                                                          B3111Y

REPLACE
      A:        line B3114R
WITH
      B:        IF (IREACT.GE.2.AND.IREACT.LE.7) WRITE (IOF(1),899) RF2MIN      B3114r

REPLACE
      A:        line B3163
WITH
      B:       1     WRITE (IOF(1),935) IX,IY                                   B3163a

REPLACE
      A:        lines B3170 - B3180
WITH
      B:        IF (TMRX(IX,IY,1).GT.0.0) WRITE (IOF(1),940) IX,IY            B3170a
      B:        IF (TMRX(IX,IY,2).GT.0.0) WRITE (IOF(1),950) IX,IY            B3180a

REPLACE
      A:        line B3183
WITH
      B:         IF (TMRX(IX-1,IY,1).GT.0.0) WRITE (IOF(1),940) IX,IY         B3183a

REPLACE
      A:        line B3186

WITH
      B:         IF (TMRX(IX,IY-1,2).GT.0.0) WRITE (IOF(1),950) IX,IY        B3186a

REPLACE
      A:        lines B3190 - B3220
WITH
      B:        IF (NODEID(IX,IY).GT.0) WRITE (IOF(1),960) IX,IY              B3190a
      B:        IF (WT(IX,IY).NE.0.0) WRITE (IOF(1),970) IX,IY                B3200a
      B:        IF (RECH(IX,IY).NE.0.0) WRITE (IOF(1),980) IX,IY              B3210a
      B:        IF (REC(IX,IY).NE.0.0) WRITE (IOF(1),990) IX,IY               B3220a

REPLACE
      A:        lines B3240 - B3280
WITH
      B:        IF (NODEID(IX,IY).GT.0.0) WRITE (IOF(1),1000) IX,IY           B3240a
      B:        IF (WT(IX,IY).NE.0.0) WRITE (IOF(1),1010) IX,IY               B3250a
      B:        IF (RECH(IX,IY).NE.0.0) WRITE (IOF(1),1020) IX,IY             B3260a
      B:        IF (REC(IX,IY).NE.0.0) WRITE (IOF(1),1030) IX,IY              B3270a
      B:        IF (THCK(IX,IY).GT.0.0) WRITE (IOF(1),1040) IX,IY             B3280a

REPLACE
      A:        line B3360
WITH
      B:      443 FORMAT (6X,23H"OBSERVATION WELL DATA")                      B3323
      B:      441 FORMAT (50X,23H"OBSERVATION WELL DATA")                     B3325
      B:      442 FORMAT (14H"NODE  (I,J): ,A2,I3,1H,,I3,13H)                 B3327
      B:         14(A2,I3,1H,,I3,15H)              ,$))                       B3335
```

```
      B:    444 FORMAT (14H"NODE  (I,J): ,A2,I3,1H,,I3,6H)     ")                  B3327
      B:    445 FORMAT (24H    HEAD        CONC.    ,$)                           B3339
      B:    446 FORMAT (12H"TIME(YRS)  ,$)                                        B3340
      B:    447 FORMAT (1H")                                                      B3342
      B:    448 FORMAT (11(1PE11.4,1X))                                           B3345
      B:    450 FORMAT (//1H ,20HITERATION PARAMETERS)                            B3361

REPLACE
      A:        lines B3380 - B3384
WITH
      B:    470 FORMAT (//1H ,27HTIME INTERVALS (IN SECONDS))                     B3381
      B:    475 FORMAT(/1H ,5X,65H*** WARNING ***   INITIAL TIME STEP IS LONGER TH  B3384

REPLACE
      A:        lines B3391 - B3401
WITH
      B:    477 FORMAT(/1H ,5X,'*** WARNING ***   SPECIFIED VALUES FOR NTIM AND TI  B3387
      B:       1MX WILL NOT ALLOW'/32X' SIMULATION TIME TO REACH PINT')            B3390
      B:    480 FORMAT (1H ,///16X,17HSTEADY-STATE FLOW//5X,55HTIME INTERVALS (IN   B3392
      B:       1SEC) FOR SOLUTE-TRANSPORT SIMULATION)                             B3402

REPLACE
      A:        lines B3430 - B3450
WITH
      B:    510 FORMAT (///,22HAQUIFER THICKNESS (L) )                            B3431
      B:    530 FORMAT (///,30HTRANSMISSIVITY MAP (L**2/SEC) )                    B3451

REPLACE
      A:        line B3490
WITH
      B:    570 FORMAT (///,23HNODE IDENTIFICATION MAP//)                         B3491

REPLACE
      A:        lines B3510 - B3520

WITH
      B:    590 FORMAT (///,//2X,16HLEAKANCE COEF. =,1X                           B3512
      B:       1,48HVERTICAL HYDRAULIC COND./THICKNESS (L/(L*SEC))  )             B3513
      B:    600 FORMAT (/1H ,9X,12HX-Y SPACING:)                                  B3521

REPLACE
      A:        lines B3540 - SB3585
WITH
      B:    620 FORMAT (///,'HYDRAULIC CONDUCTIVITY MAP (L/SEC)')                 B3541
      B:    630 FORMAT (//1H ,4X,5X,44HNO. OF FINITE-DIFFERENCE CELLS IN AQUIFER = SB3551a
      B:       1 ,I4//10X,28HAREA OF AQUIFER IN MODEL  = ,G12.5,10H   (L**2) ////1  B3561
      B:       20X,47HNZCRIT   (Max. no. of cells that can be void of/20X,56Hparti  B3570a
      B:       3cles;  if exceeded, particles are regenerated)   = ,I4/)          B3580a
      B:    633 FORMAT (1H ,////2X,26HFLOW MODEL (PRIMARY GRID):/)                SB3582a
      B:    634 FORMAT (1H ,///2X,18HTRANSPORT SUBGRID:/)                         SB3583a
      B:    635 FORMAT (1H ,04X,05X,44HNO. OF FINITE-DIFFERENCE CELLS IN AQUIFER = SB3584a
      B:       1 ,I4//10X,28HAREA OF AQUIFER IN MODEL  = ,G12.5,10H   (L**2) /)   SB3586

REPLACE
      A:        line B3620
WITH
      B:    670 FORMAT (///,42HINITIAL HEADS AND SPECIFIED BOUNDARY HEADS)         B3621

REPLACE
      A:        line B3640
WITH
      B:    690 FORMAT (/1H ,9X,19HAREA OF ONE CELL = ,G12.4)                      B3641

REPLACE
      A:        lines B3680 - B3710
```

```
WITH
      B:    730 FORMAT (/,' ',10A8,//)                                              B3680a
      B:    731 FORMAT (/,' Batch mode report written to "mocbat.rpt"')             B3685
      B:    740 FORMAT (4I4,4X,13I4,I2)                                             B3691R
      B:    741 FORMAT (5I2)                                                        B3695
      B:    750 FORMAT (01X,77HU.S.G.S. Method-of-characteristics model for solute  B3701
      B:       1 transport in ground water)                                         B3710a

REPLACE
      A:        lines B3720 - SB3763
WITH
      B:    760 FORMAT (//1H ,21X,21HI N P U T     D A T A)                         B3722
      B:    770 FORMAT (/1H ,23X,16HGRID DESCRIPTORS//13X,41HNX    (Number of colu  B3731
      B:      `1mns)            =  ,I4/13X,39HNY    (Number of rows)               B3741
      B:       2   =,I6/13X,40HXDEL  (Grid spacing in X-direction)   = ,F7.1/13X,4  B3745
      B:       30HYDEL  (Grid spacing in Y-direction)   = ,F7.1)                    B3749
      B:    775 FORMAT (//,018X,31HSECONDARY SUBGRID FOR TRANSPORT//16X,30HNMX   ( SB3763a

REPLACE
      A:        lines B3770 - B3860
WITH
      B:    780 FORMAT (//,023X,16HTIME  PARAMETERS//13X,41HNTIM   (Max. no. of ti  B3771
      B:       1me steps)     =    ,I6/13X,41HNPMP   (No. of pumping periods)      B3780a
      B:       2=    ,I6/13X,40HPINT   (Pumping period in years)    =   ,F11.3/13X  B3791a
      B:       340HTIMX   (Time step multiplier--flow) =   ,F10.2/13X,38HTINIT  (I  B3800a
      B:       4nitial time step in sec.)  = ,F10.0)                               B3810a
      B:    790 FORMAT (//,014X,34HHYDROLOGIC AND CHEMICAL PARAMETERS//13X,1HS,7X,  B3820a
      B:       129H(Storage coefficient)       =,5X,F9.6/13X,28HPOROS   (Effective  B3830a
      B:       2 porosity),8X,3H=  ,F9.3/13X,39HBETA    (Longitudinal dispersivity SB3841a
      B:       3) =  ,F7.1/13X,31HDLTRAT  (Ratio of transverse to/21X,30Hlongitudi SB3851a
      B:       4nal dispersivity)  = ,F9.2/13X,39HANFCTR  (Ratio of Tyy to Txx)    B3860a

REPLACE
      A:        line B3910
WITH
      B:    830 FORMAT (///,39HDIFFUSE RECHARGE AND DISCHARGE (L/SEC) )             B3911

INSERT
      B:    841 FORMAT (1P3(E10.3,2X))                                              B3923
      B:    842 FORMAT (1PE10.3,$)                                                  B3925
      B:    843 FORMAT (/,$)                                                        B3927

REPLACE
      A:        lines B3940 - B4012A
WITH
      B:    860 FORMAT (/1H ,7X,I2,7X,E10.3,5X,F9.2)                                B3941
      B:    870 FORMAT (//,021X,20HEXECUTION PARAMETERS//13X,41HNITP   (No. of ite  B3951
      B:       1ration parameters)   = ,I4/13X,43HTOL    (Convergence criteria--fl  B3960a
      B:       2ow)    =    ,E9.2/13X,42HITMAX  (Max. no. of iterations--flow)  =   B3971a
      B:       3 ,I3/13X,35HCELDIS (Max. cell distance per move/24X,30Hof particle  B3980a
      B:       4s--M.O.C.)       = ,F8.3/,13X,31HNPTPND (No. particles per node),6  B3990a
      B:       5X,4H  = ,I4)                                                       B4000a
      B:    880 FORMAT (//,05X,47H*** WARNING *** NPTPND MUST =  1,4,5,8,9, OR 16)  B4011a
      B:    882 FORMAT (//,05X,58H*** CAUTION *** USE OF NPTPND=1 MAY CAUSE LOSS O  B4012a

REPLACE
      A:        line B4013A
WITH
      B:    883 FORMAT (//,05X,56H*** ONE-DIMENSIONAL *** WILL USE ONLY 1 ROW OF P  B4013a

REPLACE
      A:        line B4015 - B4128J
WITH
      B:    885 FORMAT (/1H ,5X,38H*** WARNING ***  NITP MUST BE POSITIVE)          B4015a
      B:    890 FORMAT (//034X,'PROGRAM OPTIONS',//013X,'NPNT   (Time step interva  B4021
```

```
      B:       11 for complete printout)         =',I4/13X,'NPNTMV (Move interval   B4030a
      B:       2for chem. concentration printout)   =',I4/13X,'NPNTVL (Time step i  B4035
      B:       3nterval for velocity printout:'/21X,'0=Never; -1=First time step;   B4040
      B:       4-2=Last time step;'/21X,'N>0=Every Nth time step)'24X,'  ='         B4045a
      B:       6,I4/                                                               B4051a
      B:       7 13X,'NPNTD  (Print option--dispersion-equation coefficients:'     B4055a
      B:       8 /21X,'0=Never; 1=First time step; 2=All time steps)     =',I4     B4061a
      B:       $ /13X,'NUMOBS (No. of observation wells for hydrograph printout) =  B4065a
      B:       2 ',I3/                                                             B4071a
      B:       3 13X,'NREC   (No. of pumping wells)',29X,'=',I4/                   B4075a
      B:       4 13X,'NCODES (No. of codes used for node identification)',8X,'=',  B4077
      B:       4 I4/                                                               B4081a
      B:       5 13X,'NPDELC (Print option--conc. change:  0=NO; 1=YES) '          B4090
      B:       6 ,8X,'='                                                           B4118r
      B:       7 ,I4/,13X,'IREACT (Reaction specifier)',31X,'=',I4/                B4121
      B:       8 13X,'NOUTFL (Extra output file specifications: 0=NO; 1=YES)',     B4124
      B:       9 4X,'=',I4/)                                                       B4126r
      B:     891 FORMAT (//13X,'REACTION - ',A26//)                                B4127S
      B:     892 FORMAT (13X,'RHOB     (Bulk density)                = ',1PE12.5)  B4128a
      B:     893 FORMAT (13X,'DK       (Distribution coefficient)    = ',1PE12.5/  B4128b
      B:        1        13X,'RF       (Retardation factor)          = ',1PE12.5)  B4128c
      B:     894 FORMAT (13X,'EKF      (Freundlich sorption constant)= ',1PE12.5/  B4128d
      B:        1        13X,'XNF      (Freundlich sorption slope)   = ',1PE12.5)  B4128e
      B:     895 FORMAT (13X,'EKL      (Langmuir sorption constant)  = ',1PE12.5/  B4128f
      B:        1        13X,'CEC      (Cation exchange capacity)    = ',1PE12.5)  B4128g
      B:     896 FORMAT (13X,'EK       (Ion exchange constant)       = ',1PE12.5/  B4128h
      B:        1        13X,'CEC      (Cation exchange capacity)    = ',1PE12.5/  B4128i
      B:        2        13X,'CTOT     (Total solute concentration)  = ',1PE12.5)  B4128j

REPLACE
      A:        lines B4129C - B4130
WITH
      B:     898 FORMAT (/13X,'THALF    (Half life of decay, in sec)  = ',1PE12.5/  B4129c
      B:        1         13X,'DECAY    (Decay constant=ln2/THALF)    = ',1PE12.5)  B4129d
      B:     899 FORMAT (1H ,' Minimum nonlinear retardation factor for this',     B4129e
      B:        1            ' pumping period'//10X,' RF2MIN = ',1PE12.4)         B4129f
      B:     900 FORMAT (/1H ,9X,29HLOCATION OF OBSERVATION WELLS//17X,3HNO.,5X,1HX  B4131

REPLACE
      A:        lines B4150 - B4180
WITH
      B:     910 FORMAT (/1H ,9X,28HLOCATION  OF  PUMPING  WELLS//11X,28HX   Y   RA  B4151
      B:        1TE(L**3/SEC) CONC./)                                               B4161
      B:     920 FORMAT (////,5X,37HNO. OF NODE IDENT. CODES SPECIFIED = ,I2)       B4171
      B:     930 FORMAT (//,10X,41HTHE FOLLOWING ASSIGNMENTS HAVE BEEN MADE://5X,51  B4181

REPLACE
      A:        line B4420
WITH
      B:    1050 FORMAT (1H ,5X,45H*** WARNING ***   ANFCTR WAS SPECIFIED AS 0.0/23  B4421

REPLACE
      A:        lines B4450 - B4480
WITH
      B:    1070 FORMAT (10I4,3G5.0)                                                B4450a
      B:    1080 FORMAT (////5X,25HSTART PUMPING PERIOD NO. ,I2//2X,75HThe followin  B4461
      B:        1g time step, pumpage, and print parameters have been redefined:/)  B4470a
      B:    1090 FORMAT (//,015X,9HNTIM   = ,I4/15X,9HNPNT   = ,I4/15X,9HNITP   = ,  B4481

REPLACE
      A:        lines B4532 - B4533
WITH
      B:    1110 FORMAT (////5X,25HSTART PUMPING PERIOD NO. ,I2//2X,23HNo parameter  B4532a
      B:        1s redefined/)                                                      B4533a
```

```
INSERT
      B:   1120  FORMAT (1X,1A36,1A43)                                                B4534A
      B:   1130  FORMAT (//04X,'THE FOLLOWING ADDITIONAL OUTPUT FILES WILL BE CREA  B4534B
      B:       1TED: '/)                                                            B4524C
      B:   1140 FORMAT (/' NOBSO     Write observation well data   (0: do not writ  B4524D
      B:       1e; 1: write all'/'          data to one file; 2: write data to NU  B4525A
      B:       2MOBS files--one file for '/'          each observation point)      B4525B
      B:       3                                  =',I3)                            B4525C
      B:   1150 FORMAT (/' NHEADO    Write head data (0: do not write; -1: write i  B4525D
      B:       1nitial head only;'/'          1: write final head only;  2: write  B4526A
      B:       1 initial and final head)    =',I3)                                 B4526B
      B:   1160 FORMAT (/' NPNCHV    Write velocity data (-1:initial velocity; 0:   B4526C
      B:       1do not write;'/'          -2: write final velocity; N>0: every Nt  B4526D
      B:       2h time step)',13X,'=',I3)                                          B4527A
      B:   1170 FORMAT (/' NCONCO    Write concentration data (0: do not write; -1  B4527B
      B:       1: write initial  '/'          conc.; 1: write final conc.; 2: wri  B4527C
      B:       2te initial and final conc.) =',I3)                                 B4527D
      B:   1180 FORMAT (/' NPARMO    Write transmissivity, sat. thickness, diffuse  B4528A
      B:       1 recharge/dis-  '/'          charge, & hydraulic conductivity dat  B4528B
      B:       2a to separate files using '/'          values from input data  (0  B4528C
      B:       3: do not write any; 1: write all)    =',I3)                        B4528D
      B:   1185 FORMAT (/' IFMT      Write data in column or matrix format (0: col  B4529A
      B:       1umn; 1: matrix)   =',I3)                                          B4529B
      B:   1190 FORMAT (//,19X,'OPTIONS TO CREATE ADDITIONAL OUTPUT FILES  '/)     B4529C
DELETE
      A:        DOUBLE PRECISION DMIN1,DEXP,DLOG,DABS                             C  20
REPLACE
      A:        line C  82
WITH
      B:       2              NPNTMV,NPNTVL,NPNTD,NPNCHV,NPDELC,ICHK,IFMT        C  82a
      B:       3              ,NOBSO,NHEADO,NCONCO,NPARMO,NOUTFL                 C  93
INSERT
      B:        COMMON /FLMGR/IOF(32),ICFLAG,IVIN,IRPT                          SC 166
REPLACE
      A:        line C1050
WITH
      B:    100 WRITE (IOF(1),160)                                              C1050a
REPLACE
      A:        line C1157B
WITH
      B:         WRITE (IOF(1),170) IX,IY                                       C1157b
REPLACE
      A:        line C1183
WITH
      B:        PQIN=PQIN-REC(IX,IY)                                            C1183a
REPLACE
      A:        line C1185
WITH
      B:     32 PQOUT=PQOUT-REC(IX,IY)                                          C1185a
REPLACE
      A:        line C1187
WITH
      B:        PQIN=PQIN-RECH(IX,IY)*AREA                                      C1187a
REPLACE
      A:        line C1189
WITH
      B:     36 PQOUT=PQOUT-RECH(IX,IY)*AREA                                    C1189a
```

```
REPLACE
       A:        line C1250
WITH
       B:        WRITE (IOF(1),140) N                                          C1250a

REPLACE
       A:        line C1260
WITH
       B:        WRITE (IOF(1),150) KOUNT                                      C1260a

REPLACE
       A:        line C1265
WITH
       B:         WRITE (IOF(1),151)                                           C1265a

REPLACE
       A:        line C1330
WITH
       B:    140 FORMAT (1H ///3X,4HN = ,1I4)                                  C1331

REPLACE
       A:        line C1350
WITH
       B:    160 FORMAT (////5X,64H***   EXECUTION TERMINATED -- MAX. NO. ITERATION  C1351

REPLACE
       A:        line D  62
WITH
       B:       2              NPNTMV,NPNTVL,NPNTD,NPNCHV,NPDELC,ICHK,IFMT     D  62a
       B:       3              ,NOBSO,NHEADO,NCONCO,NPARMO,NOUTFL              D  65

INSERT
       B:        COMMON /FLMGR/IOF(32),ICFLAG,IVIN,IRPT                       SD 159

DELETE
       A:        DOUBLE PRECISION DMIN1,DEXP,DLOG,DABS                         E  20

REPLACE
       A:        line E  82
WITH
       B:       2              NPNTMV,NPNTVL,NPNTD,NPNCHV,NPDELC,ICHK,IFMT     E  82a
       B:       3              ,NOBSO,NHEADO,NCONCO,NPARMO,NOUTFL              E  93

INSERT
       B:        COMMON /FLMGR/IOF(32),ICFLAG,IVIN,IRPT                       SE 209

INSERT
       B:        IF (NPNTVL.GT.0) THEN                                         E 783

INSERT
       B:        END IF                                                        E 787

REPLACE
       A:        lines E 801 - E 810
WITH
       B:        WRITE (IOF(1),320)                                            E 801a
       B:        WRITE (IOF(1),330)                                            E 810a

REPLACE
       A:        line E 830
WITH
       B:     40 WRITE (IOF(1),350) (VX(IX,IY),IX=1,NX)                        E 830a

REPLACE
       A:        line E 840
WITH
       B:  C     WRITE (IOF(1),340)                                            E 840a
```

```
REPLACE
      A:        line E 860 - E 880
WITH
      B:  C  50 WRITE (IOF(1),350) (VXBDY(IX,IY),IX=1,NX)                    E 860a
      B:        WRITE (IOF(1),360)                                           E 870a
      B:        WRITE (IOF(1),330)                                           E 880a

REPLACE
      A:        line E 900
WITH
      B:     60 WRITE (IOF(1),350) (VY(IX,IY),IX=1,NX)                       E 900a

REPLACE
      A:        line E 910

WITH
      B:  C     WRITE (IOF(1),340)                                           E 910a

REPLACE
      A:        line E 930
WITH
      B:  C  70 WRITE (IOF(1),350) (VYBDY(IX,IY),IX=1,NX)                    E 930a

REPLACE
      A:        lines E 961 - E 975
WITH
      B:         IF (NPNCHV.EQ.-2.AND.((N.EQ.NTIM).OR.IPRNT.EQ.1)) GO TO 90  E 961a
      B:         IF (NPNCHV.EQ.-1.AND.N.EQ.1) GO TO 90                       E 971a
      B:         IF (MOD(N,NPNCHV).EQ.0.AND (NPNCHV.NE.-2)) GO TO 90         E 975a
      B:        IF (NPNCHV.EQ.1.AND.(IPRNT.EQ.1.OR.N.EQ.NTIM)                E 976
      B:       1.AND.INT.EQ.NPMP) GO TO 90                                   E 978

REPLACE
      A:        line E 991
WITH
      B:        WRITE (IOF(4),361) N, INT                                    E 992
      B:        YMAX=NY*YDEL-YDEL                                            E 995

DELETE
      A:        WRITE (7,520) (VX(IX,IY),IX=1,NX)                            E1010
      A:    100 WRITE (7,520) (VY(IX,IY),IX=1,NX)                            E1020

INSERT
      B:        YYY=(IY*YDEL)-1.5*YDEL                                       E1002
      B:        DO 101 IX=1,NX                                               E1003
      B:        XXX=(IX*XDEL)-.5*XDEL                                        E1004
      B:        IF (IX.GT.1.AND.IX.LT.NX.AND.IY.GT.1.AND.IY.LT.NY) WRITE     E1005
      B:       1(IOF(4),842) XXX,(YMAX-YYY),VX(IX,IY),VY(IX,IY)              E1007
      B:    101 CONTINUE                                                     E1009
      B:    100 CONTINUE                                                     E1015

REPLACE
      A:        lines E1051 - E1070
WITH
      B:        WRITE (IOF(1),390)                                           E1051a
      B:        WRITE (IOF(1),410) VMXBD,VMYBD                               E1070a

REPLACE
      A:        line E1079R
WITH
      B:        WRITE (IOF(1),394) VMXBD,VMYBD                               E1079r

REPLACE
      A:        lines E1126 - E1130
WITH
      B:        IF (AMAX1(VMXBD,VMYBD).LE.1.0E-10) WRITE(IOF(1),570)         E1126a
      B:        WRITE (IOF(1),310) TMV,TIMV                                  E1130a
```

```
REPLACE
       A:        line E1210
WITH
       B:        WRITE (IOF(1),420) TIMV,NTIMV,NMOV                              E1210a

REPLACE
       A:        lines E1230 - E1240
WITH
       B:        WRITE (IOF(1),370) TIM(N)                                       E1230a
       B:        WRITE (IOF(1),380) TIMV                                         E1240a

REPLACE
       A:        line E1860
WITH
       B:        WRITE (IOF(1),440) TIMDC                                        E1860a

REPLACE
       A:        line E1930
WITH
       B:    180 WRITE (IOF(1),430) TIMV,NTIMD,NMOV                              E1930a

REPLACE
       A:        line E2060
WITH
       B:    210 WRITE (IOF(1),530)                                              E2060a

REPLACE
       A:        line E2064
WITH
       B:           WRITE (IOF(1),534) MAXVYI,MAXVYJ,MAXVYI,MJ                   E2064a

REPLACE
       A:        line E2067
WITH
       B:           WRITE (IOF(1),535) MAXVXI,MAXVXJ,MI,MAXVXJ                   E2067a

REPLACE
       A:        line E2080
WITH
       B:    220 WRITE (IOF(1),540)                                              E2080a

REPLACE
       A:        lines E2100 - E2102
WITH
       B:    230 WRITE (IOF(1),550)                                              E2100a
       B:        WRITE (IOF(1),560) MAXX,MAXY                                    E2102a

REPLACE
       A:        line E2106
WITH
       B:    235 WRITE (IOF(1),580)                                              E2106a

REPLACE
       A:        lines E2170 -  E2180
WITH
       B:    250 WRITE (IOF(1),450)                                              E2170a
       B:        WRITE (IOF(1),460)                                              E2180a

REPLACE
       A:        lines SE2201 - E2210
WITH
       B:    260 WRITE (IOF(1),500) (DISP(IX,IY,1),IX=1,NMX)                     SE2201a
       B:        WRITE (IOF(1),470)                                              E2210a

REPLACE
       A:        lines SE2231 - E2240
```

```
WITH
      B:    270 WRITE (IOF(1),500) (DISP(IX,IY,2),IX=1,NMX)                    SE2231a
      B:        WRITE (IOF(1),480)                                              E2240a

REPLACE
      A:        lines SE2261 - E2270
WITH
      B:    280 WRITE (IOF(1),500) (DISP(IX,IY,3),IX=1,NMX)                    SE2261a
      B:        WRITE (IOF(1),490)                                              E2270a

REPLACE
      A:        line SE2291
WITH
      B:    290 WRITE (IOF(1),500) (DISP(IX,IY,4),IX=1,NMX)                    SE2291a

REPLACE
      A:        line E2361R
WITH
      B:    310 FORMAT (/1X,19H TMV (MAX. INJ.) = ,G12.5/20H  TIMV (CELDIS)   = ,G  E2361r

REPLACE
      A:        line E2380
WITH
      B:    320 FORMAT (////,2X,12HX VELOCITIES)                                E2382

REPLACE
      A:        line E2401
WITH
      B:  C 340 FORMAT (/1X,24X,41HON BOUNDARIES (USING INTERFACE THICKNESS)/)   E2401a

REPLACE
      A:        line E2420
WITH
      B:    360 FORMAT (1H /,2X,12HY VELOCITIES)                                E2422
      B:    361 FORMAT (46H" VELOCITIES; the four columns = X, Y, VX, VY ",/,13H"  E2425
     +Time step = ,I4,', Pumping period = ',I4,1H")                             E2427

REPLACE
      A:        lines E2450 - E2453R
WITH
      B:    390 FORMAT (//1H ,9X,29HSTABILITY CRITERIA --- M.O.C.//)            E2451
      B:    394 FORMAT (/1X,4X,46HMaximum effective solute velocities:  X-VEL = ,  E2453r

REPLACE
      A:        lines E2471 - E2490
WITH
      B:    410 FORMAT (/1X,4X,41HMaximum fluid velocities (L/T):  X-VEL = ,    E2471A
      B:       11PE9.2,5X,8HY-VEL = ,1PE9.2)                                    E2472A
      B:    420 FORMAT (/1X,8H TIMV = ,1PE9.2,5X,8HNTIMV = ,I5,5X,7HNMOV = ,I5/)  E2480a
      B:    430 FORMAT (/1X,8H TIMV = ,1PE9.2,5X,8HNTIMD = ,I5,5X,7HNMOV = ,I5)   E2490a

REPLACE
      A:        line E2511
WITH
      B:    450 FORMAT (///,32HDISPERSION EQUATION COEFFICIENTS,10X,33H=(D-IJ)*(B)  E2511a

REPLACE
      A:        lines E2531 - E2561
WITH
      B:    460 FORMAT (/1X,35X,14HXX COEFFICIENT/)                             E2531a
      B:    470 FORMAT (/1X,35X,14HYY COEFFICIENT/)                             E2541a
      B:    480 FORMAT (/1X,35X,14HXY COEFFICIENT/)                             E2551a
      B:    490 FORMAT (/1X,35X,14HYX COEFFICIENT/)                             E2561a

REPLACE
      A:        line E2600
```

```
WITH
      B:    530 FORMAT (/1X,09X,42HTHE LIMITING STABILITY CRITERION IS CELDIS)      E2600a
REPLACE
      A:        lines E2610 - E2620
WITH
      B:    540 FORMAT (/1X,09X,40HTHE LIMITING STABILITY CRITERION IS BETA)        E2610a
      B:    550 FORMAT (/1X,09X,58HTHE LIMITING STABILITY CRITERION IS MAXIMUM INJ  E2620a
REPLACE
      A:        lines E2637 - E2638
WITH
      B:    570 FORMAT (/1X,4X,47H*** WARNING ***  DECREASE CRITERIA IN E 230-260)  E2637a
      B:    580 FORMAT (/1X,09X,63H*TIME INCREMENT FOR SOLUTE TRANSPORT EQUALS TIM  E2638a
INSERT
      B:    842 FORMAT (1P4(E10.3,2X))                                              E2639B
REPLACE
      A:        line F  62
WITH
      B:       2              NPNTMV,NPNTVL,NPNTD,NPNCHV,NPDELC,ICHK,IFMT           F  62a
      B:       3              ,NOBSO,NHEADO,NCONCO,NPARMO,NOUTFL                    F  65
INSERT
      B:        COMMON /PRPT/ GX1,GX2,GX3,GEGT                                     F 180
      B:        COMMON /FLMGR/IOF(32),ICFLAG,IVIN,IRPT                            SF 185
REPLACE
      A:        lines F 200 - F 205R
WITH
      B:        WRITE (IOF(1),680) NMOV                                           F 200a
      B:        IF (THALF.GT.0.0.AND.THALF.LT.TIMV) WRITE (IOF(1),685)            F 205r
INSERT
      B:        CALL PGRPT1                                                       F 275
INSERT
      B:         IF (ICFLAG.EQ.1) GOTO 10                                         F 281
      B:  C     ****************************************************************  F 282
      B:  C     THIS Section PRINTS % PROGRESS AS PARTICLES ARE MOVED             F 283
      B:  C     NOTE:                                                             F 284A
      B:  C     FORMAT 998 USES NONSTANDARD '$' FORMAT SO ASTERICKS WILL PRINT    F 284B
      B:  C     ON SAME LINE. THOUGH THIS IS OFTEN SUPPORTED, IF PROBLEMS ARISE   F 284C
      B:  C     CONSULT YOUR FORTRAN COMPILER MANUAL.                             F 284D
      B:        GX2=AINT(GEGT*IMOV)                                               F 284E
      B:        GX3=GX2-GX1                                                       F 284F
      B:        IF (GX3.GT.0) THEN                                                F 285
      B:         DO 997 JJ=1,GX3                                                  F 286
      B:        WRITE (IRPT,998)                                                  F 287
      B:    997  CONTINUE                                                         F 288
      B:         GX1=GX2                                                          F 289A
      B:        END IF                                                            F 289B
      B:    998 FORMAT('*',$)                                                     F 289C
      B:  C     ***********END OF PROGRESS PRINTING FUNCTION*******************   F 289D
REPLACE
      A:        line F1730
WITH
      B:        WRITE (IOF(1),690) IN,IX,IY                                       F1730a
REPLACE
      A:        lines F1830 - F1871
WITH
      B:  C     (FOLLOWING 3 CHANGES ARE EQUIVALENT BUT MORE EFFICIENT STATEMENTS)  F1825
      B:  C     VXN=VXNW*(1.0-CELDX)+VXNE*CELDX                                   F1830a
      B:        VXN=VXNW+(VXNE-VXNW)*CELDX                                        F1832
```

```
      B:  C     VXS=VXSW*(1.0-CELDX)+VXSE*CELDX                                  F1850a
      B:        VXS=VXSW+(VXSE-VXSW)*CELDX                                       F1852
      B:  C     XVEL=(VXN*(1.0-CELDY)+VXS*CELDY)/THCK(IX,IY)                     F1871a
      B:        XVEL=(VXN+(VXS-VXN)*CELDY)/THCK(IX,IY)                           F1873

REPLACE
      A:        lines F1931 - F1971
WITH
      B:  C     (FOLLOWING 3 CHANGES ARE EQUIVALENT BUT MORE EFFICIENT STATEMENTS)  F1925
      B:  C     VYW=VYNW*(1.0-CELYD)+VYSW*CELYD                                  F1931a
      B:        VYW=VYNW+(VYSW-VYNW)*CELYD                                       F1933
      B:  C     VYE=VYNE*(1.0-CELYD)+VYSE*CELYD                                  F1951a
      B:        VYE=VYNE+(VYSE-VYNE)*CELYD                                       F1953
      B:  C     YVEL=(VYW*(1.0-CELXD)+VYE*CELXD)/THCK(IX,IY)                     F1971a
      B:        YVEL=(VYW+((VYE-VYW)*CELXD))/THCK(IX,IY)                         F1973

DELETE
      A:  C     IF (REC(IX,IY).LT.-0.1) GO TO 350                                FXXX
      A:  C     IF (REC(IX,IY).GT.0.1) GO TO 360                                 FXXX
      A:  C     IF (REC(INX,INY).GT.0.1) GO TO 560                               FXXX

REPLACE
      A:        line F3800
WITH
      B:   600  WRITE (IOF(1),700) IMOV,IN                                       F3800a

REPLACE
      A:        line F3930
WITH
      B:        WRITE (IOF(1),670) NP,IMOV                                       F3930a

REPLACE
      A:        lines F4150 - F4160
WITH
      B:   670  FORMAT (1H ,2X,2HNP,7X,2H= ,8X,I4,10X,11HIMOV     = ,8X,I4)      F4151
      B:   680  FORMAT (/1H ,9X,61HNO. OF PARTICLE MOVES REQUIRED TO COMPLETE THIS  F4161

REPLACE
      A:        line F4174R
WITH
      B:   685  FORMAT (//,05X,51H*** CAUTION ***  DECAY HALF-LIFE IS LESS THAN TI  F4174r

REPLACE
      A:        line F4180
WITH
      B:   690  FORMAT (//,05X,53H*** WARNING ***     QUADRANT NOT LOCATED FOR PT.  F4181

REPLACE
      A:        line F4200
WITH
      B:   700  FORMAT (//,05X,17H ***   NOTE   ***,10X,23HNPTM.EQ.NPMAX --- IMOV=  F4201

REPLACE
      A:        line G  72
WITH
      B:       2              NPNTMV,NPNTVL,NPNTD,NPNCHV,NPDELC,ICHK,IFMT          G  72a
      B:       3              ,NOBSO,NHEADO,NCONCO,NPARMO,NOUTFL                   G  75

INSERT
      B:        COMMON /FLMGR/IOF(32),ICFLAG,IVIN,IRPT                           SG 200

REPLACE
      A:        IF (ABS(C1).LT.1.0E-20) C1=0.0                                   MULTICS
WITH
      B:        IF (ABS(C1).LT.1.0E-20) C1=0.0                                    G 402R
```

```
REPLACE
      A:       line G 960
WITH
      B:       IF (NZERO.GT.0) WRITE (IOF(1),290) NZERO,IMOV                    G 960a

REPLACE
      A:       lines G 990 - G1000
WITH
      B:       WRITE (IOF(1),300)                                               G 990a
      B:       WRITE (IOF(1),320)                                               G1000a

REPLACE
      A:       line SG1021
WITH
      B:   100 WRITE (IOF(1),330) (NPCELL(IX,IY),IX=1,NMX)                      SG1021a

REPLACE
      A:       line SG1161
WITH
      B:   120 IF (CONC(IX,IY).GT.0.0) WRITE (IOF(1),310) JX,JY,CONC(IX,IY)     SG1161a

REPLACE
      A:       line G1370
WITH
      B:       WRITE (IOF(1),280) TIM(N),TIMV,SUMTCH                            G1370a

REPLACE
      A:       lines G1860 - G1880
WITH
      B:   290 FORMAT (05X,042H  NUMBER OF CELLS WITH ZERO PARTICLES  =  ,I4,5X,9 G1863
      B:      1HIMOV  =  ,I4)                                                   G1863
      B:   300 FORMAT (//,05X,44H***   NZCRIT EXCEEDED  ---  CALL GENPT   ***/)  G1881

REPLACE
      A:       line G1910
WITH
      B:   320 FORMAT (/1H ,2X,6HNPCELL/)                                       G1911

REPLACE
      A:       line H  31A
WITH
      B:      1              HMIN,PYR,ANFCTR,YMAX                               H  31a

REPLACE
      A:       lines H  40 - H  61
WITH
      B:       COMMON /PRMJ/ NTIM,NPMP,NPNT,NITP,N,NX,NY,NP,NREC,INT,NNX,NNY,    H  41
      B:      1              NUMOBS,NMOV,IMOV,NPMAX,ITMAX,NZCRIT,IPRNT,NPTPND,  H  51
      B:      2              NPNTMV,NPNTVL,NPNTD,NPNCHV,NPDELC,ICHK,IFMT        H  62
      B:      3              ,NOBSO,NHEADO,NCONCO,NPARMO,NOUTFL                 H  63

INSERT
      B:       COMMON /FLMGR/IOF(32),ICFLAG,IVIN,IRPT                           H 132

INSERT
      B:       YMAX=NY*YDEL-YDEL                                                H 155

REPLACE
      A:       lines H 190 - H 240
WITH
      B:       WRITE (IOF(1),120)                                               H 190a
      B:       WRITE (IOF(1),130) N                                             H 200a
      B:       WRITE (IOF(1),140) SUMT                                          H 210a
      B:       WRITE (IOF(1),150) TIMD                                          H 220a
      B:       WRITE (IOF(1),160) TIMY                                          H 230a
      B:       WRITE (IOF(1),170)                                               H 240a
```

```
REPLACE
      A:       line H 260
WITH
      B:    10 WRITE (IOF(1),180) (HK(IX,IY),IX=1,NX)                          H 260a

REPLACE
      A:       lines H 300 - H 350
WITH
      B:       WRITE (IOF(1),120)                                               H 300a
      B:       WRITE (IOF(1),130) N                                             H 310a
      B:       WRITE (IOF(1),140) SUMT                                          H 320a
      B:       WRITE (IOF(1),150) TIMD                                          H 330a
      B:       WRITE (IOF(1),160) TIMY                                          H 340a
      B:       WRITE (IOF(1),170)                                               H 350a

INSERT
      B:       YYY=(IY*YDEL)-1.5*YDEL                                           H 365

INSERT
      B:       IF ((NHEADO.EQ.1).OR.(NHEADO.EQ.2)) THEN                         H 371
      B:        IF ((IPRNT.EQ.1.OR.N.EQ.NTIM).AND.(INT.EQ.NPMP)) THEN           H 372
      B:         IF ((IX.GT.1).AND.(IX.LT.NX).AND.(IY.GT.1).AND.(IY.LT.NY))    H 374
      B:       1 THEN                                                           H 375A
      B:           XXX=(IX*XDEL)-.5*XDEL                                        H 375B
      B:            IF (IFMT.EQ.0) THEN                                         H 375C
      B:               WRITE (IOF(10),841) XXX,(YMAX-YYY),HK(IX,IY)             H 375D
      B:            ELSE                                                        H 375E
      B:               WRITE (IOF(10),842) HK(IX,IY)                            H 375F
      B:                 IF (IX.EQ.(NX-1)) THEN                                 H 375G
      B:               WRITE (IOF(10),843)                                      H 375H
      B:                 END IF                                                 H 375I
      B:            END IF                                                      H 375J
      B:          END IF                                                        H 376
      B:         END IF                                                         H 378
      B:       END IF                                                           H 380

REPLACE
      A:       line H 390
WITH
      B:    30 WRITE (IOF(1),190) (IH(ID),ID=1,NX)                             H 390a

REPLACE
      A:       line H 510

WITH
      B:       IF (S.GT.0.0) WRITE (IOF(1),290)                                 H 511

REPLACE
      A:       line H 564
WITH
      B:       PQIN=PQIN-REC(IX,IY)                                             H 564a

REPLACE
      A:       line H 568
WITH
      B:    32 PQOUT=PQOUT-REC(IX,IY)                                          H 568a

REPLACE
      A:       line H 574
WITH
      B:       PQIN=PQIN-RECH(IX,IY)*AREA                                       H 574a

REPLACE
      A:       line H 578
WITH
      B:    36 PQOUT=PQOUT-RECH(IX,IY)*AREA                                    H 578a
```

```
REPLACE
      A:        line H 700
WITH
      B:        IF (S.GT.0.0) WRITE (IOF(1),300) (IH(IX),IX=1,NX)                    H 701

REPLACE
      A:        lines H 840 - H 910
WITH
      B:    100 WRITE (IOF(1),240)                                                   H 840a
      B:        WRITE (IOF(1),211) TPIN                                              H 842a
      B:        WRITE (IOF(1),212) TPOUT                                             H 844a
      B:        WRITE (IOF(1),250) PUMP                                              H 850a
      B:        WRITE (IOF(1),230) QSTR                                              H 860a
      B:        WRITE (IOF(1),202) TOTLQI                                            H 862a
      B:        WRITE (IOF(1),203) TOTLQ                                             H 864a
      B:        WRITE (IOF(1),260) TOTLQN                                            H 866a
      B:        WRITE (IOF(1),270) ERRMB                                             H 880a
      B:        WRITE (IOF(1),280) PCTERR                                            H 883a
      B:        WRITE (IOF(1),201)                                                   H 886a
      B:        WRITE (IOF(1),202) QIN                                               H 889a
      B:        WRITE (IOF(1),203) QOUT                                              H 892a
      B:        WRITE (IOF(1),204) QNET                                              H 895a
      B:        WRITE (IOF(1),211) PQIN                                              H 898a
      B:        WRITE (IOF(1),212) PQOUT                                             H 901a
      B:        WRITE (IOF(1),210) TPUM                                              H 910a

REPLACE
      A:        lines H1000 - H1040
WITH
      B:    120 FORMAT (1H ,//1X,26HHEAD DISTRIBUTION (BY ROW)/)                     H1001
      B:    130 FORMAT (1X,23HNumber of time steps = ,1I5)                           H1010a
      B:    140 FORMAT (7X,17HTime (seconds) = ,1G12.5)                              H1020a
      B:    150 FORMAT (7X,17HTime (days)    = ,1E12.5)                              H1030a
      B:    160 FORMAT (7X,17HTime (years)   = ,1E12.5)                              H1040a

REPLACE
      A:        lines H1055 - H1180
WITH
      B:    180 FORMAT (1H ,10F12.7)                                                 H1056
      B:    190 FORMAT (/1H ,30I4)                                                  SH1071a
      B:    201 FORMAT (//1H ,1X,35HRATE MASS BALANCE -- (IN L**3/SEC) //)           H1074
      B:    202 FORMAT (4X,29HLeakage into aquifer       = ,E12.5)                   H1076a
      B:    203 FORMAT (4X,29HLeakage out of aquifer     = ,E12.5)                   H1083a
      B:    204 FORMAT (4X,29HNet leakage    (QNET)      = ,E12.5)                   H1086a
      B:    210 FORMAT (4X,29HNet withdrawal   (TPUM)    = ,E12.5///)                H1093a
      B:    211 FORMAT (4X,29HRecharge and injection     = ,E12.5)                   H1096a
      B:    212 FORMAT (4X,29HPumpage and E-T withdrawal = ,E12.5)                   H1103a
      B:    230 FORMAT (4X,29HWater release from storage = ,1E12.5)                  H1120a
      B:    240 FORMAT (//1H ,1X,37HCumulative mass balance -- (IN L**3) //)        H1125a
      B:    250 FORMAT (4X,29HCumulative  net  pumpage   = ,1E12.5)                  H1140a
      B:    260 FORMAT (4X,29HCumulative  net  leakage   = ,1E12.5)                  H1150a
      B:    270 FORMAT (/1H ,7X,25HMass balance residual  = ,G12.5)                  H1161
      B:    280 FORMAT (1H ,7X,25HError  (as percent)    = ,G12.5/)                  H1170a
      B:    290 FORMAT (///1H ,8HDRAWDOWN)                                           H1181

INSERT
      B:    841 FORMAT (1P,3(E10.3,2X))                                              H1195
      B:    842 FORMAT (1PE10.3,$)                                                   H1196
      B:    843 FORMAT (/,$)                                                         H1199

REPLACE
      A:        line I  62
WITH
      B:      2             NPNTMV,NPNTVL,NPNTD,NPNCHV,NPDELC,ICHK,IFMT            I  62a
      B:      3             ,NOBSO,NHEADO,NCONCO,NPARMO,NOUTFL                     I  65
```

```
INSERT
      B:        COMMON /FLMGR/IOF(32),ICFLAG,IVIN,IRPT                          SI 159
INSERT
      B:        YMAX=NY*YDEL-YDEL                                                I 185
INSERT
      B:        IF ((NPNTMV.EQ.0).OR.(NPNT.EQ.0)) GOTO 8                         I 217
REPLACE
      A:        lines I 250 - SI 351
WITH
      B:      8 WRITE (IOF(1),160)                                               I 250a
      B:        WRITE (IOF(1),170) N                                             I 260a
      B:        IF (N.GT.0) WRITE (IOF(1),180) TIM(N)                            I 270a
      B:        WRITE (IOF(1),190) SUMT                                          I 280a
      B:        WRITE (IOF(1),450) SUMTCH                                        I 290a
      B:        WRITE (IOF(1),200) TCHD                                          I 300a
      B:        WRITE (IOF(1),210) TMYR                                          I 310a
      B:        WRITE (IOF(1),460) TCHYR                                         I 320a
      B:        WRITE (IOF(1),380) IMOV                                          I 330a
      B:        WRITE (IOF(1),220)                                               I 340a
      B:     11 DO 20 IY=1,NMY                                                  SI 351a
      B:        JY=IY+MY-1                                                       I 356
      B:        YYY=(JY*YDEL)-1.5*YDEL                                           I 357
INSERT
      B:        JX=IX+MX-1                                                       I 363A
      B:        IF (NCONCO.EQ.0) GOTO 10                                         I 363B
      B:        IF (NCONCO.EQ.1.OR.NCONCO.EQ.2) THEN                             I 363C
      B:         IF ((IPRNT.EQ.1.OR.N.EQ.NTIM)                                   I 363D
      B:       1.AND.IMOV.EQ.NMOV.AND.INT.EQ.NPMP) THEN                          I 363E
      B:           IF (JX.GT.1.AND.JX.LT.NX.AND.JY.GT.1.AND.JY.LT.NY) THEN       I 363F
      B:            XXX=(JX*XDEL)-.5*XDEL                                        I 363G
      B:             IF (IFMT.EQ.0) THEN                                         I 363H
      B:                WRITE (IOF(11),841) XXX,(YMAX-YYY),CONC(IX,IY)           I 363I
      B:             ELSE                                                        I 363J
      B:                WRITE (IOF(11),842) CONC(IX,IY)                          I 363K
      B:                NNN=NMX                                                  I 363L
      B:                IF (MCHK.GT.0) NNN=(NMX+1)                               I 3631
      B:                  IF (IX.EQ.(NNN-1)) THEN                                I 363M
      B:                WRITE (IOF(11),843)                                      I 363N
      B:                  END IF                                                 I 363O
      B:              END IF                                                     I 363P
      B:            END IF                                                       I 364
      B:           END IF                                                        I 366
      B:          END IF                                                         I 368
REPLACE
      A:        line SI 381
WITH
      B:     20 WRITE (IOF(1),240) (IC(IX),IX=1,NMX)                            SI 381a
REPLACE
      A:        lines I 440 - I 530
WITH
      B:        WRITE (IOF(1),230)                                               I 440a
      B:        WRITE (IOF(1),170) N                                             I 450a
      B:        WRITE (IOF(1),180) TIM(N)                                        I 460a
      B:        WRITE (IOF(1),190) SUMT                                          I 470a
      B:        WRITE (IOF(1),450) SUMTCH                                        I 480a
      B:        WRITE (IOF(1),200) TCHD                                          I 490a
      B:        WRITE (IOF(1),210) TMYR                                          I 500a
      B:        WRITE (IOF(1),460) TCHYR                                         I 510a
      B:        WRITE (IOF(1),380) IMOV                                          I 520a
      B:        WRITE (IOF(1),220)                                               I 530a
```

```
REPLACE
      A:        line SI 581
WITH
      B:     40 WRITE (IOF(1),240) (IC(IX),IX=1,NMX)                    SI 581a
REPLACE
      A:        lines I 670 - I 710
WITH
      B:     70 WRITE (IOF(1),220)                                      I 670a
      B:        WRITE (IOF(1),250)                                      I 680a
      B:        WRITE (IOF(1),220)                                      I 690a
      B:        WRITE (IOF(1),260) FLMIN                                I 700a
      B:        WRITE (IOF(1),270) FLMOT                                I 710a
REPLACE
      A:        line I 740 - I 830
WITH
      B:        WRITE (IOF(1),290) RECIN                                I 740a
      B:        WRITE (IOF(1),280) RECOUT                               I 750a
      B:        WRITE (IOF(1),295) DMASS1                               I 751r
      B:        WRITE (IOF(1),296) ADSORB                               I 752r
      B:        WRITE (IOF(1),298) SORBI                                I 753r
      B:        WRITE (IOF(1),300) SUMIO                                I 760a
      B:        WRITE (IOF(1),310) STORMI                               I 770a
      B:        WRITE (IOF(1),320) STORM                                I 780a
      B:        WRITE (IOF(1),330) CSTORM                               I 790a
      B:        WRITE (IOF(1),332) CSTM2                                I 795r
      B:        WRITE (IOF(1),340)                                      I 810a
      B:        WRITE (IOF(1),350) RESID                                I 820a
      B:        WRITE (IOF(1),360) ERR1                                 I 830a
REPLACE
      A:        lines I 850 - I 860
WITH
      B:        WRITE (IOF(1),370)                                      I 850a
      B:        WRITE (IOF(1),360) ERR3                                 I 860a
      B:    100 WRITE (IOF(1),220)                                      I 863
REPLACE
      A:        line I 921
WITH
      B:        IF (NUMOBS.LE.0) GO TO 150                              I 922
REPLACE
      A:        lines I 931 - I 960
WITH
      B:        WRITE (IOF(1),390) TITLE                                I 931a
      B:        WRITE (IOF(1),400) INT                                  I 940a
      B:        IF (S.GT.0.0) WRITE (IOF(1),410)                        I 950a
      B:        IF (S.EQ.0.0) WRITE (IOF(1),420)                        I 960a
INSERT
      B:  C     FILENAME.OBS                                            I1055
REPLACE
      A:        line I1080
WITH
      B:        WRITE (IOF(1),430) J,IXOBS(J),IYOBS(J)                  I1080a
REPLACE
      A:        lines SI1088 - SI1091
WITH
      B:    125 WRITE (IOF(1),435)                                      SI1088a
      B:    127 WRITE (IOF(1),440) MOZ,WT(JX,JY),C1INT,                 SI1091a
REPLACE
      A:        line I1130
```

```
WITH
      B:    130 WRITE (IOF(1),440) M,TMWL(J,M),TMCN(J,M),TMYR                          I1130a
INSERT
      B:  C     ************************************************************          I1141
      B:        DO 132 M=1,NTO                                                       I1142
      B:        TMYR=TMOBS(M)/TMFY                                                   I1143
      B:        IF (NOBSO.EQ.1) THEN                                                 I1145A
      B:        WRITE (IOF(2),438) TMYR,(TMWL(J,M),TMCN(J,M), J=1,NUMOBS)            I1145B
      B:        END IF                                                               I1145C
      B:        IF (NOBSO.EQ.2) THEN                                                 I1145D
      B:          DO 142 J=1,NUMOBS                                                  I1145E
      B:            IFL=(11+J)                                                       I1145F
      B:            WRITE (IOF(IFL),438) TMYR, TMWL(J,M), TMCN(J,M)                  I1145G
      B:    142 CONTINUE                                                             I1145H
      B:        END IF                                                               I1145I
      B:    132 CONTINUE                                                             I1147
REPLACE
      A:        lines I1210 - I1260
WITH
      B:    160 FORMAT (///1X,13HCONCENTRATION/)                                     I1211
      B:    170 FORMAT (1X,23HNumber of time steps = ,1I5)                           I1220a
      B:    180 FORMAT (7X,17HDelta t        = ,1G12.5)                              I1230a
      B:    190 FORMAT (7X,17HTime (seconds) = ,1G12.5)                              I1240a
      B:    200 FORMAT (2X,22HChem.time (days)    = ,1E12.5)                         I1250a
      B:    210 FORMAT (7X,17HTime (years)   = ,1E12.5)                              I1260a
REPLACE
      A:        lines I1280 - I1290
WITH
      B:    230 FORMAT (///1H ,23HCHANGE IN CONCENTRATION/)                          I1281
      B:    240 FORMAT (/1H ,20I5)                                                   I1291
REPLACE
      A:        lines I1310 - I1382R
WITH
      B:    260 FORMAT (8X,25HMass in boundaries     = ,1E12.5)                      I1310a
      B:    270 FORMAT (8X,25HMass out boundaries    = ,1E12.5)                      I1320a
      B:    280 FORMAT (8X,25HMass pumped out        = ,1E12.5)                      I1330a
      B:    290 FORMAT (8X,25HMass pumped in         = ,1E12.5)                      I1340a
      B:    295 FORMAT (8X,25HMass lost by decay     = ,1E12.5)                      I1341r
      B:    296 FORMAT (8X,25HMass adsorbed on solids= ,1E12.5)                      I1342r
      B:    298 FORMAT (8X,25HInitial mass adsorbed  = ,1E12.5)                      I1343r
      B:    300 FORMAT (8X,25HInflow minus outflow   = ,1E12.5)                      I1350a
      B:    310 FORMAT (8X,25HInitial mass dissolved = ,1E12.5)                      I1361r
      B:    320 FORMAT (8X,25HPresent mass dissolved = ,1E12.5)                      I1371r
      B:    330 FORMAT (8X,25HChange mass dissolved  = ,1E12.5)                      I1381r
      B:    332 FORMAT (8X,25HChange totl.mass stored= ,1E12.5)                      I1382r
REPLACE
      A:        lines I1410 - I1420
WITH
      B:    350 FORMAT (8X,25HMass balance residual  = ,1E12.5)                      I1410a
      B:    360 FORMAT (8X,25HError  (as percent)    = ,1E12.5)                      I1420a
      B:    361 FORMAT (8X,27H"Error  (as percent)    = ",1E12.5)                    I1421
REPLACE
      A:        lines I1450 - I1520
WITH
      B:    380 FORMAT (1X,23H No. moves completed = ,1I5)                           I1450a
      B:    390 FORMAT (////////10A8//)                                              I1462
      B:    400 FORMAT (/1H ,5X,65HTime versus head and concentration at selected    I1470a
      B:       1observation points//15X,19HPUMPING PERIOD NO. ,I4///)                I1481
      B:    410 FORMAT (/1H ,16X,19HTRANSIENT  SOLUTION//)                           I1491
      B:    420 FORMAT (/1H ,15X,21HSTEADY-STATE SOLUTION//)                         I1501
```

```
      B:    430 FORMAT (/1H ,2X,22HOBS.WELL NO.    X    Y,05X,1HN,6X,40H  HEAD      I1512
      B:      1  CONCENTRATION   TIME (YEARS)//06X,I3,9X,I2,3X,I2/)                  I1522

REPLACE
      A:        lines I1531 - I1550
WITH
      B:    438 FORMAT (11(1PE11.4,1X))                                              I1530
      B:    440 FORMAT (1H ,28X,I2,6X,F7.1,8X,F7.1,8X,F7.3)                          I1532
      B:    450 FORMAT (1H ,1X,22HChem.time (seconds) = ,E12.5)                      I1540a
      B:    460 FORMAT (1H ,1X,22HChem.time (years)   = ,E12.5)                      I1550a
      B:    841 FORMAT (1P,3(E10.3,2X))                                              I1555
      B:    842 FORMAT (1PE10.3,$)                                                   I1556
      B:    843 FORMAT (/,$)                                                         I1558

INSERT
      B:        SUBROUTINE FILMGR                                                    L  10
      B:  C     THIS SUBROUTINE RECEIVES KEYBOARD INPUT AND SETS UP INPUT            L  20
      B:  C         FILE NAMES AND UNIT NUMBERS.                                     L  30
      B:  C     -- FILENAME VARIABLES SET TO 40-CHARACTERS FOR IBM PC-VERSION --     L  40
      B:        CHARACTER*40 INFILE                                                  L  50
      B:        COMMON /FLMGR/IOF(32),ICFLAG,IVIN,IRPT                               L  60
      B:        COMMON /NAME/INFILE                                                  L  70
      B:        COMMON /BOUT/ICNFILE,NUMFILE                                         L  80
      B:        LOGICAL EXISTS                                                       L  90
      B:  C     ***************************************************************      L 100
      B:  C     NOTE:                                                                L 110
      B:  C     FORMAT 52 USES NONSTANARD '$' FORMAT SO READ INPUT WILL PRINT        L 120
      B:  C     ON SAME LINE. THOUGH THIS IS OFTEN SUPPORTED, OTHERS USE '/'         L 130
      B:  C     IF PROBLEMS ARISE, CONSULT YOUR FORTRAN COMPILER MANUAL.             L 140
      B:  C                                                                          L 150
      B:        IVIN=7                                                               L 160
      B:        IF (ICFLAG.EQ.0) THEN                                                L 170
      B:  C                                                                          L 180
      B:  C     INTRODUCTORY SCREEN AND PROMPTS FOR INPUT FILE                       L 190
      B:  C     FIRST SCREEN TAKES MOC INPUT FILE NAME                               L 200
      B:        WRITE (*,*) '  '                                                     L 210
      B:        WRITE (*,*) '  '                                                     L 220
      B:        WRITE (*,*) '  '                                                     L 230
      B:        WRITE (*,*) '  '                                                     L 240
      B:        WRITE (*,*) '  '                                                     L 250
      B:        WRITE (*,*) '  '                                                     L 260
      B:        WRITE (*,*) '  '                                                     L 270
      B:        WRITE (*,51)                                                         L 280
      B:    51  FORMAT (23X,'USGS MOC model (Version 3.1) ')                         L 290
      B:        WRITE (*,*) '  '                                                     L 300
      B:        WRITE (*,508)                                                        L 310
      B:    508 FORMAT (5X,'Please provide the name of the file containing the inp   L 320
      B:       1ut data for the ',/,' MOC model.  If the file is not in the curren   L 330
      B:       2t working directory, specify',/,' the complete pathname (directory   L 340
      B:       3 and subdirectories) and disk drive (if ',/,' different).  Example   L 350
      B:       4s include:',40H    MOCINP.DAT or A: \DATADIR\MOCINP.DAT )            L 360
      B:    48  WRITE (*,*) '  '                                                     L 370
      B:        WRITE (*,52)                                                         L 380
      B:        WRITE (*,53)                                                         L 385
      B:        READ  (*,58) INFILE                                                  L 390
      B:        IF (INFILE.eq."QUIT".or.INFILE.eq."quit") STOP                       L 392
      B:    52  FORMAT (' ',//,3X,' ENTER NAME OF INPUT FILE')                       L 395
      B:    53  FORMAT (' ',3x,'(or type "QUIT" to stop): ',$)                       L 400
      B:        INQUIRE (FILE=INFILE, EXIST=EXISTS)                                  L 410
      B:        IF (EXISTS) THEN                                                     L 420
      B:          CONTINUE                                                           L 430
      B:           WRITE(*,*) ' '                                                    L 440
      B:           WRITE(*,*) 'FILE EXISTS; program execution is continuing.'        L 450
      B:           WRITE(*,*) ' '                                                    L 460
      B:           WRITE(*,*) ' '                                                    L 470
```

```
B:           WRITE(*,*) ' '                                                  L 480
B:          ELSE                                                             L 490
B:           WRITE(*,56)                                                     L 500
B:           GOTO 48                                                         L 510
B:          END IF                                                           L 520
B:    56   FORMAT(' FILE OF THIS NAME DOES NOT EXIST!  USE ANOTHER NAME OR DI  L 530
B:       +RECTORY')                                                          L 540
B:        ELSE                                                             L 550
B:    25   READ (45,57) INFILE                                             L 560
B:  C       FOR PC WRITE INFILE NAMES TO TRACK WORK PERFORMED               L 570
B:          WRITE (IRPT,97) INFILE                                          L 580
B:          INQUIRE (FILE=INFILE, EXIST=EXISTS)                             L 590
B:          IF (EXISTS) THEN                                                L 600
B:            WRITE(IRPT,*) 'FILE EXISTS'                                   L 610
B:            CONTINUE                                                      L 620
B:          ELSE                                                            L 630
B:         WRITE(IRPT,*) 'Input file specified in mocbat.dat does not exist.' L 640
B:         WRITE(IRPT,*) 'Processing will continue with next input file.'   L 650
B:           ICNFILE=ICNFILE+1                                             L 660
B:           IF (ICNFILE.EQ.NUMFILE) THEN                                  L 670
B:            RETURN                                                       L 680
B:           ELSE                                                          L 690
B:            GOTO 25                                                      L 700
B:           END IF                                                        L 710
B:          END IF                                                         L 720
B:         END IF                                                          L 730
B:         OPEN(UNIT=IVIN,FILE=INFILE,ACCESS='SEQUENTIAL',STATUS='OLD')    L 740
B:    57   FORMAT(A40)                                                     L 750
B:    97   FORMAT(/1x,A40)                                                L 760
B:    58   FORMAT(A40)                                                     L 770
B:         RETURN                                                          L 780
B:         END                                                             L 790-
B:         SUBROUTINE OFLMGR                                               M  10
B:  C      THIS SUBROUTINE SETS UP OUTPUT FILE NAMES AND UNIT NUMBERS.     M  20
B:  C      -- FILENAME VARIABLES SET TO 40-CHARACTERS FOR IBM PC-VERSION -- M  30
B:         CHARACTER*40 OUTFILE,INFILE                                     M  40
B:         CHARACTER*43 OPF                                                M  50
B:         CHARACTER*36 OPE                                                M  60
B:         COMMON /PRMJ/ NTIM,NPMP,NPNT,NITP,N,NX,NY,NP,NREC,INT,NNX,NNY,  M  70
B:        1              NUMOBS,NMOV,IMOV,NPMAX,ITMAX,NZCRIT,IPRNT,NPTPND,  M  80
B:        2              NPNTMV,NPNTVL,NPNTD,NPNCHV,NPDELC,ICHK,IFMT       M  90
B:        3              ,NOBSO,NHEADO,NCONCO,NPARMO,NOUTFL                M 100
B:         COMMON /NAME/INFILE                                             M 110
B:         COMMON /FLMGR/IOF(32),ICFLAG,IVIN,IRPT                          M 120
B:         COMMON /FLNMS/ OPF(31),OPE(31)                                  M 130
B:          CHARACTER*2 A                                                  M 140
B:         LOGICAL EXISTS                                                  M 150
B:  C     ***************************************************************  M 160
B:  C      NOTE:                                                           M 170
B:  C      FORMAT 54 USES NONSTANDARD '$' FORMAT SO READ INPUT WILL PRINT  M 180
B:  C      ON SAME LINE. THOUGH THIS IS OFTEN SUPPORTED, OTHERS USE '/'    M 190
B:  C      IF PROBLEMS ARISE, CONSULT YOUR FORTRAN COMPILER MANUAL.        M 200
B:  C                                                                      M 210
B:         WRITE(IRPT,*) 'The following OUTPUT files will be generated: '  M 220
B:         OPE(1)=' standard MOC output data ......... '                   M 230
B:         OPE(2)=' observation well data ............ '                   M 240
B:         OPE(3)=' initial hydraulic head data ...... '                   M 250
B:         OPE(10)=' final hydraulic head data ........ '                  M 260
B:         OPE(4)=' velocity data ..................... '                  M 270
B:         OPE(5)=' initial concentration data ....... '                   M 280
B:         OPE(11)=' final concentration data ......... '                  M 290
B:         OPE(6)=' transmissivity data .............. '                   M 300
B:         OPE(7)=' saturated thickness data ......... '                   M 310
B:         OPE(8)=' diffuse recharge/discharge data .. '                   M 320
```

```
B:        OPE(9)=' hydraulic conductivity data ..... '                    M 330
B:  C     STRIP EXTENSION OFF INPUT FILENAME FOR OUTPUT FILENAMES         M 340
B:        DO 10 NC= 40,1,-1                                               M 350
B:         IF (INFILE(NC:NC).EQ.'.') GOTO 20                              M 360
B:     10 CONTINUE                                                        M 370
B:     20 IF (NC.LE.1) THEN                                               M 380
B:         DO 30 IEND=1,40                                                M 390
B:          IF(INFILE(IEND:IEND).EQ.' ') THEN                             M 400
B:           NC=IEND+1                                                    M 410
B:           GOTO 40                                                      M 420
B:          ELSE                                                          M 430
B:           NC=40                                                        M 440
B:          END IF                                                        M 450
B:     30  CONTINUE                                                       M 460
B:        END IF                                                          M 470
B:     40 OUTFILE=INFILE(1:NC-1)                                          M 480
B:        IEND=INDEX(OUTFILE,' ')                                         M 490
B:        OPF(1)=OUTFILE(1:(IEND-1))//'.out'                              M 500
B:        WRITE (IRPT,*) '  '                                             M 510
B:         WRITE (IRPT,35) OPE(1),OPF(1)                                  M 520
B:         IF (ICFLAG.EQ.0) THEN                                          M 530
B:         INQUIRE (FILE=OPF(1), EXIST=EXISTS)                            M 540
B:          IF (EXISTS) THEN                                              M 550
B:           WRITE(*,36)                                                  M 560
B:           WRITE(*,37)                                                  M 570
B:           READ (*,38) IWRITE                                           M 580
B:           IF (IWRITE.EQ.1) STOP                                        M 590
B:          ENDIF                                                         M 600
B:         ENDIF                                                          M 610
B:        WRITE (IRPT,*) '  '                                             M 620
B:  C      OBS. WELL FILE                                                 M 630
B:         IF (NOBSO.EQ.1) THEN                                           M 640
B:          OPF(2)=OUTFILE(1:(IEND-1))//'.obs'                            M 650
B:         WRITE (IRPT,35) OPE(2),OPF(2)                                  M 660
B:          IF (ICFLAG.EQ.0) THEN                                         M 670
B:           INQUIRE (FILE=OPF(2), EXIST=EXISTS)                          M 680
B:           IF (EXISTS) THEN                                             M 690
B:            WRITE(*,36)                                                 M 700
B:            WRITE(*,37)                                                 M 710
B:            READ (*,38) IWRITE                                          M 720
B:            WRITE (*,*) '  '                                            M 730
B:           IF (IWRITE.EQ.1) STOP                                        M 740
B:           ENDIF                                                        M 750
B:          ENDIF                                                         M 760
B:         ENDIF                                                          M 770
B:         IF (NOBSO.EQ.2) THEN                                           M 780
B:          DO 44 J=1,(NUMOBS)                                            M 790
B:            IFL=(11+J)                                                  M 800
B:             IF (J.LT.10) THEN                                          M 810
B:               WRITE(A,'(I1)') J                                        M 820
B:               OPF(IFL)=outfile(1:(IEND-1))//'.o'//A                    M 830
B:             ELSE IF (J.GT.9) THEN                                      M 840
B:               WRITE(A,'(I2)') J                                        M 850
B:               OPF(IFL)=outfile(1:(IEND-1))//'.o'//A                    M 860
B:             END IF                                                     M 870
B:            WRITE (IRPT,35) OPE(2),OPF(IFL)                             M 880
B:           IF (ICFLAG.EQ.0) THEN                                        M 890
B:            INQUIRE (FILE=OPF(IFL), EXIST=EXISTS)                       M 900
B:            IF (EXISTS) THEN                                            M 910
B:             WRITE(*,36)                                                M 920
B:             WRITE(*,37)                                                M 930
B:             READ (*,38) IWRITE                                         M 940
B:             WRITE (*,*) '  '                                           M 950
B:             IF (IWRITE.EQ.1) STOP                                      M 960
```

```
B:             ENDIF                                                    M 970
B:            ENDIF                                                     M 980
B:     44    CONTINUE                                                  M 990
B:          ENDIF                                                      M1000
B:  C       HEAD DATA FILE                                             M1010
B:          IF ((NHEADO.EQ.-1).OR.(NHEADO.EQ.2)) THEN                  M1020
B:           OPF(3)=OUTFILE(1:(IEND-1))//'.hd0'                        M1030
B:           WRITE (IRPT,35) OPE(3),OPF(3)                             M1040
B:            IF (ICFLAG.EQ.0) THEN                                    M1050
B:             INQUIRE (FILE=OPF(3), EXIST=EXISTS)                     M1060
B:             IF (EXISTS) THEN                                        M1070
B:              WRITE(*,36)                                            M1080
B:              WRITE(*,37)                                            M1090
B:              READ (*,38) IWRITE                                     M1100
B:              WRITE (*,*) '  '                                       M1110
B:              IF (IWRITE.EQ.1) STOP                                  M1120
B:             ENDIF                                                   M1130
B:            ENDIF                                                    M1140
B:          ENDIF                                                      M1150
B:          IF ((NHEADO.EQ.1).OR.(NHEADO.EQ.2)) THEN                   M1160
B:           OPF(10)=OUTFILE(1:(IEND-1))//'.hd1'                       M1170
B:           WRITE (IRPT,35) OPE(10),OPF(10)                           M1180
B:           IF (ICFLAG.EQ.0) THEN                                     M1190
B:            INQUIRE (FILE=OPF(10), EXIST=EXISTS)                     M1200
B:            IF (EXISTS) THEN                                         M1210
B:             WRITE(*,36)                                             M1220
B:             WRITE(*,37)                                             M1230
B:             READ (*,38) IWRITE                                      M1240
B:             WRITE (*,*) '  '                                        M1250
B:             IF (IWRITE.EQ.1) STOP                                   M1260
B:            ENDIF                                                    M1270
B:           ENDIF                                                     M1280
B:          ENDIF                                                      M1290
B:  C       VELOCITY DATA FILE                                         M1300
B:          IF (NPNCHV.NE.0) THEN                                      M1310
B:           OPF(4)=OUTFILE(1:(IEND-1))//'.vel'                        M1320
B:           WRITE (IRPT,35) OPE(4),OPF(4)                             M1330
B:           IF (ICFLAG.EQ.0) THEN                                     M1340
B:            INQUIRE (FILE=OPF(4), EXIST=EXISTS)                      M1350
B:            IF (EXISTS) THEN                                         M1360
B:             WRITE(*,36)                                             M1370
B:             WRITE(*,37)                                             M1380
B:             READ (*,38) IWRITE                                      M1390
B:             WRITE (*,*) '  '                                        M1400
B:             IF (IWRITE.EQ.1) STOP                                   M1410
B:            ENDIF                                                    M1420
B:           ENDIF                                                     M1430
B:          ENDIF                                                      M1440
B:  C       CONCENTRATION DATA FILES                                   M1450
B:          IF ((NCONCO.EQ.-1).OR.(NCONCO.EQ.2)) THEN                  M1460
B:           OPF(5)=OUTFILE(1:(IEND-1))//'.cn0'                        M1470
B:           WRITE (IRPT,35) OPE(5),OPF(5)                             M1480
B:           IF (ICFLAG.EQ.0) THEN                                     M1490
B:            INQUIRE (FILE=OPF(5), EXIST=EXISTS)                      M1500
B:            IF (EXISTS) THEN                                         M1510
B:             WRITE(*,36)                                             M1520
B:             WRITE(*,37)                                             M1530
B:             READ (*,38) IWRITE                                      M1540
B:             WRITE (*,*) '  '                                        M1550
B:             IF (IWRITE.EQ.1) STOP                                   M1560
B:            ENDIF                                                    M1570
B:           ENDIF                                                     M1580
B:          ENDIF                                                      M1590
B:          IF ((NCONCO.EQ.2).OR.(NCONCO.EQ.1)) THEN                   M1600
```

```
B:          OPF(11)=OUTFILE(1:(IEND-1))//'.cn1'                                    M1610
B:          WRITE (IRPT,35) OPE(11),OPF(11)                                        M1620
B:          IF (ICFLAG.EQ.0) THEN                                                  M1630
B:           INQUIRE (FILE=OPF(11), EXIST=EXISTS)                                  M1640
B:           IF (EXISTS) THEN                                                      M1650
B:            WRITE(*,36)                                                          M1660
B:            WRITE(*,37)                                                          M1670
B:            READ (*,38) IWRITE                                                   M1680
B:            WRITE (*,*) '  '                                                     M1690
B:            IF (IWRITE.EQ.1) STOP                                                M1700
B:           ENDIF                                                                 M1710
B:          ENDIF                                                                  M1720
B:         ENDIF                                                                   M1730
B:  C      INPUT PARAMETER DATA FILES                                             M1740
B:         IF (NPARMO.NE.0) THEN                                                  M1750
B:          OPF(6)=OUTFILE(1:(IEND-1))//'.trn'                                    M1760
B:          OPF(7)=OUTFILE(1:(IEND-1))//'.thk'                                    M1770
B:          OPF(8)=OUTFILE(1:(IEND-1))//'.rec'                                    M1780
B:          OPF(9)=OUTFILE(1:(IEND-1))//'.prm'                                    M1790
B:          DO 69 IJG=6,9                                                         M1800
B:          WRITE (IRPT,35) OPE(IJG),OPF(IJG)                                     M1810
B:          IF (ICFLAG.EQ.0) THEN                                                 M1820
B:           INQUIRE (FILE=OPF(IJG), EXIST=EXISTS)                                M1830
B:           IF (EXISTS) THEN                                                     M1840
B:            WRITE(*,36)                                                         M1850
B:            WRITE(*,37)                                                         M1860
B:            READ (*,38) IWRITE                                                  M1870
B:            WRITE (*,*) '  '                                                    M1880
B:            IF (IWRITE.EQ.1) STOP                                               M1890
B:           ENDIF                                                                M1900
B:          ENDIF                                                                 M1910
B:     69   CONTINUE                                                             M1920
B:         ENDIF                                                                 M1930
B:          DO 70 IJG=1,32                                                       M1940
B:           IOF(IJG)=IJG+7                                                      M1950
B:     70   CONTINUE                                                            M1960
B:        OPEN(UNIT=IOF(1),FILE=OPF(1),ACCESS='SEQUENTIAL')                      M1970
B:        IF (NOBSO.EQ.1) OPEN(UNIT=IOF(2),FILE=OPF(2),                          M1980
B:       1ACCESS='SEQUENTIAL')                                                   M1990
B:        IF (NOBSO.EQ.2) THEN                                                   M2000
B:         DO 75 J=1,NUMOBS                                                      M2010
B:           IFL=(11+J)                                                          M2020
B:         OPEN(UNIT=IOF(IFL),FILE=OPF(IFL),ACCESS='SEQUENTIAL')                 M2030
B:     75 CONTINUE                                                             M2040
B:        END IF                                                                 M2050
B:        IF ((NHEADO.EQ.-1).OR.(NHEADO.EQ.2)) THEN                              M2060
B:         OPEN(UNIT=IOF(3),FILE=OPF(3),ACCESS='SEQUENTIAL')                     M2070
B:        END IF                                                                 M2080
B:        IF ((NHEADO.EQ.2).OR.(NHEADO.EQ.1)) THEN                               M2090
B:         OPEN(UNIT=IOF(10),FILE=OPF(10),ACCESS='SEQUENTIAL')                   M2100
B:        END IF                                                                 M2110
B:        IF (NPNCHV.NE.0) OPEN(UNIT=IOF(4),FILE=OPF(4),ACCESS='SEQUENTIAL')     M2120
B:        IF ((NCONCO.EQ.-1).OR.(NCONCO.EQ.2)) THEN                              M2130
B:         OPEN(UNIT=IOF(5),FILE=OPF(5),ACCESS='SEQUENTIAL')                     M2140
B:        END IF                                                                 M2150
B:        IF ((NCONCO.EQ.2).OR.(NCONCO.EQ.1)) THEN                               M2160
B:         OPEN(UNIT=IOF(11),FILE=OPF(11),ACCESS='SEQUENTIAL')                   M2170
B:        END IF                                                                 M2180
B:        IF (NPARMO.NE.0) THEN                                                  M2190
B:         OPEN(UNIT=IOF(6),FILE=OPF(6),ACCESS='SEQUENTIAL')                     M2200
B:         OPEN(UNIT=IOF(7),FILE=OPF(7),ACCESS='SEQUENTIAL')                     M2210
B:         OPEN(UNIT=IOF(8),FILE=OPF(8),ACCESS='SEQUENTIAL')                     M2220
B:         OPEN(UNIT=IOF(9),FILE=OPF(9),ACCESS='SEQUENTIAL')                     M2230
B:        END IF                                                                 M2240
```


*U.S. G.P.O.:1994-301-077:00004

```
B:     35  FORMAT (1X,1A36,1A43)                                                M2250
B:     36  FORMAT (' A FILE HAVING THIS NAME EXISTS IN WORKING DIRECTORY.')     M2260
B:     37  FORMAT (' DO YOU WANT TO OVER-WRITE EXISTING FILE?  ',/,             M2270
B:        15X,' IF NOT, TYPE "1"; IF OK, HIT "ENTER/RETURN":',$)               M2280
B:     38  FORMAT (I1)                                                          M2290
B:         RETURN                                                              M2300
B:         END                                                                 M2310-
B:         SUBROUTINE PGRPT1                                                   N  10
B:         COMMON /PRMJ/ NTIM,NPMP,NPNT,NITP,N,NX,NY,NP,NREC,INT,NNX,NNY,      N  20
B:        1              NUMOBS,NMOV,IMOV,NPMAX,ITMAX,NZCRIT,IPRNT,NPTPND,    N  30
B:        2              NPNTMV,NPNTVL,NPNTD,NPNCHV,NPDELC,ICHK,IFMT          N  40
B:        3              ,NOBSO,NHEADO,NCONCO,NPARMO,NOUTFL                   N  50
B:         COMMON /PRPT/ GX1,GX2,GX3,GEGT                                      N  60
B:         COMMON /FLMGR/IOF(32),ICFLAG,IVIN,IRPT                              N  70
B:  C      THIS SUBROUTINE PRINTS A PROGRESS REPORT WHEN MOVE IS CALLED        N  80
B:         WRITE(IRPT,*) ' '                                                   N  90
B:         WRITE(IRPT,*) ' '                                                   N 100
B:         WRITE (IRPT,134) INT,NPMP                                           N 110
B:    134  FORMAT ('   PUMPING PERIOD:',08X,I5,'   OF',I4)                     N 120
B:         WRITE(IRPT,*) ' '                                                   N 130
B:         WRITE (IRPT,135) N,NTIM                                             N 140
B:    135  FORMAT ('   TIME STEP:',13X,I5,'   OF',I4)                         N 150
B:         WRITE(IRPT,*) ' '                                                   N 160
B:         WRITE (IRPT,136) NMOV                                               N 170
B:    136  FORMAT ('   NUMBER OF PARTICLE MOVES:',I8)                         N 180
B:         WRITE(IRPT,*) ' '                                                   N 190
B:         WRITE(IRPT,*) ' '                                                   N 200
B:         IF (ICFLAG.EQ.0) WRITE (IRPT,137) N                                 N 210
B:    137  FORMAT (10x,' PERCENT COMPLETION OF MOVES DURING TIME STEP',        N 220
B:        1I5,':',/)                                                          N 230
B:         IF (ICFLAG.EQ.0) WRITE (IRPT,999)                                   N 240
B:    999  FORMAT (80H 0.....10......20......30......40......50......60.....  N 250
B:        +70......80......90....100)                                         N 260
B:         WRITE(IRPT,'(1H ,$)')                                               N 270
B:         GX1=0                                                               N 280
B:         GX2=0                                                               N 290
B:         GX3=0                                                               N 300
B:         IF (NMOV.GT.0) THEN                                                 N 310
B:          GEGT=(1.0000/(FLOAT(NMOV)))*79.00000                               N 320
B:         ELSE                                                                N 330
B:          GEGT=0.00000                                                       N 340
B:         END IF                                                              N 350
B:         RETURN                                                              N 360
B:         END                                                                 N 370-
B:         SUBROUTINE CLEAR                                                    O  10
B:         CHARACTER NUL,ESC                                                   O  20
B:         NUL=CHAR(0)                                                         O  30
B:         ESC=CHAR(27)                                                        O  40
B:         WRITE (*,10) NUL, ESC, NUL, ESC                                     O  50
B:     10  FORMAT (1X,2A1,'[0m',2A1,'[2J' )                                    O  60
B:         RETURN                                                              O  70
B:         END                                                                 O  80-
B:         SUBROUTINE HOME                                                     P  10
B:         CHARACTER NUL,ESC                                                   P  20
B:         NUL=CHAR(0)                                                         P  30
B:         ESC=CHAR(27)                                                        P  40
B:         WRITE (*,10) NUL, ESC, NUL, ESC                                     P  50
B:     10  FORMAT (1X,2A1,'[0;40;36m',2A1,'[H')                                P  60
B:         RETURN                                                              P  70
B:         END                                                                 P  80-
```


*U.S. G.P.O.:1994-301-077:4