

ENHANCED JOB CONTROL LANGUAGE PROCEDURES FOR THE SIMSYS2D

TWO-DIMENSIONAL WATER-QUALITY SIMULATION SYSTEM

By George A. Karavitis

U.S. GEOLOGICAL SURVEY

Water-Resources Investigations Report 84-4052

Tallahassee, Florida

1984



UNITED STATES DEPARTMENT OF THE INTERIOR

WILLIAM P. CLARK, Secretary

GEOLOGICAL SURVEY

Dallas L. Peck, Director

For additional information
write to:

District Chief
U.S. Geological Survey
Suite 3015
227 North Bronough Street
Tallahassee, Florida 32301

Copies of this report can be
purchased from:

Open-File Services Section
Western Distribution Branch
U.S. Geological Survey
Box 25425, Federal Center
Denver, Colorado 80225
(Telephone: (303) 234-5888)

CONTENTS

	Page
Abstract -----	1
Introduction -----	2
SIMSYS2D procedures -----	4
Procedural improvements -----	5
Processing considerations -----	5
Input data processor (IDP) -----	5
Mixer procedure -----	6
SIM2D procedure -----	6
References -----	7
Attachment A. Input data processor (IDP) procedure -----	8
Attachment B. Mixer procedure -----	11
Attachment C. SIM2D procedure -----	14
Attachment D. Disk to tape procedure -----	18
Attachment E. Tape to disk procedure -----	20

ILLUSTRATIONS

	Page
Figure 1. SIMSYS2D input-output file diagram -----	3

ENHANCED JOB CONTROL LANGUAGE PROCEDURES FOR THE SIMSYS2D
TWO-DIMENSIONAL WATER-QUALITY SIMULATION SYSTEM

By George A. Karavitis

ABSTRACT

The SIMSYS2D two-dimensional water-quality simulation system is a large-scale digital-modeling software system used to simulate flow and transport of solutes in freshwater and estuarine environments. Because of size, processing requirements, and complexity of the system, there is a need to easily move the system and its associated files between computer sites when required. A series of job control language (JCL) procedures were written to allow transfer between IBM and IBM-compatible computers.

INTRODUCTION

The SIMSYS2D two-dimensional water-quality simulation system is a large-scale digital-modeling software package designed for the study of shallow, well-mixed estuarine environments (Rand Corporation, 1977). The package was developed by the Rand Corporation in Santa Monica, Calif., under contract to the U.S. Geological Survey and the Rijkwaterstaat (state water agency) of the Netherlands. When supplied with the appropriate tidal, meteorological, and bathymetric information, SIMSYS2D is capable of calculating water transport, water level, and multiple pollutant concentration distributions within a study area. Simulation output takes the form of tabular listings, Benson-Lehner FR80¹/ plotter microfilm, Tektronix screen graphics, disk or tape data history files, and simulation restart tapes (Rand Corporation, 1978). The system is composed of a number of presimulation and postsimulation application programs that produce graphical output. At the center of the package is the simulation program that performs the actual predictive modeling of estuarine and shallow coastal environments.

The SIMSYS2D requires considerable computer resources for all but the smallest applications. A typical estuarine simulation of a 36-hour tidal period for a 100 x 100 computational grid with a 450-meter node interval and a single pollutant constituent would require approximately 1,400,000 bytes (1400K) of

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

main memory and 4 hours of Central Processing Unit (CPU) time on a mid-sized, general-purpose computer, such as an IBM 370/168II. In addition, large amounts of peripheral storage are needed to hold the output generated from each simulation. Management of input, output, graphics, restart, and program files can become a formidable task when a suite of simulations studying different environmental conditions is employed.

A number of operational data-processing problems can arise in implementation of SIMSYS2D. Data-processing installations may require advance notice or restrict processing to weekends for an application of this magnitude, since normal handling could severely affect computer operations. Another limitation is that few computer centers have enough online disk storage available to hold all the generated output files. This limitation would require the simultaneous use of a number of tape drives, which in turn would cause device allocation problems. These factors, in addition to high processing costs and the capability of a data-processing installation to support a project of this scale for a long term, suggest that the usefulness of SIMSYS2D can be enhanced if it is readily transferable between IBM or IBM compatible mainframes. In this way, simulation users could take advantage of increased computational speed, processor availability, or lower pricing rates at different sites. In addition, transferability would facilitate shifting of operations in the event an installation could no longer support the activity.

This report presents a series of enhanced job control language procedures designed to aid in both transferability between computer sites and ease of use by estuarine modelers.

SIMSYS2D PROCEDURES

The simulation system is divided into three distinct procedures called IDP, MIXER, and SIM2D. Figure 1 illustrates the relation of the three procedures and the files that they create. Each procedure consists of a main program (IDP, MIXER, or SIM2D) and several other utility programs that perform the following tasks. IDP (input data processor) receives the user-prepared input to the simulation in punched card image format and performs a variety of physical and logical tests to insure that no errors have been made. The power and importance of this step can be appreciated because a typical simulation can have 4,000 to 5,000 input card images with a myriad of formats, delimiters, and mandatory entries. It is necessary to run the IDP procedure until all errors in the input data have been found and the edit procedures do not produce any error messages. In addition to editing, IDP calculates the size of dimensioned variables used in the simulation. This procedure creates five online disk files for subsequent use and printed output that contains source data listings and editing messages.

The MIXER procedure takes the dimensioning information created by IDP and creates FORTRAN COMMON statements tailored precisely to the size of the computational grid and incorporates them in the simulation code. It then compiles the FORTRAN source code, invokes the linkage editor, produces a load module, and compresses the output load module library. At the end of this procedure, a fully executable load module, tailored to the specific problem area and conditions, exists to perform the modeling work.

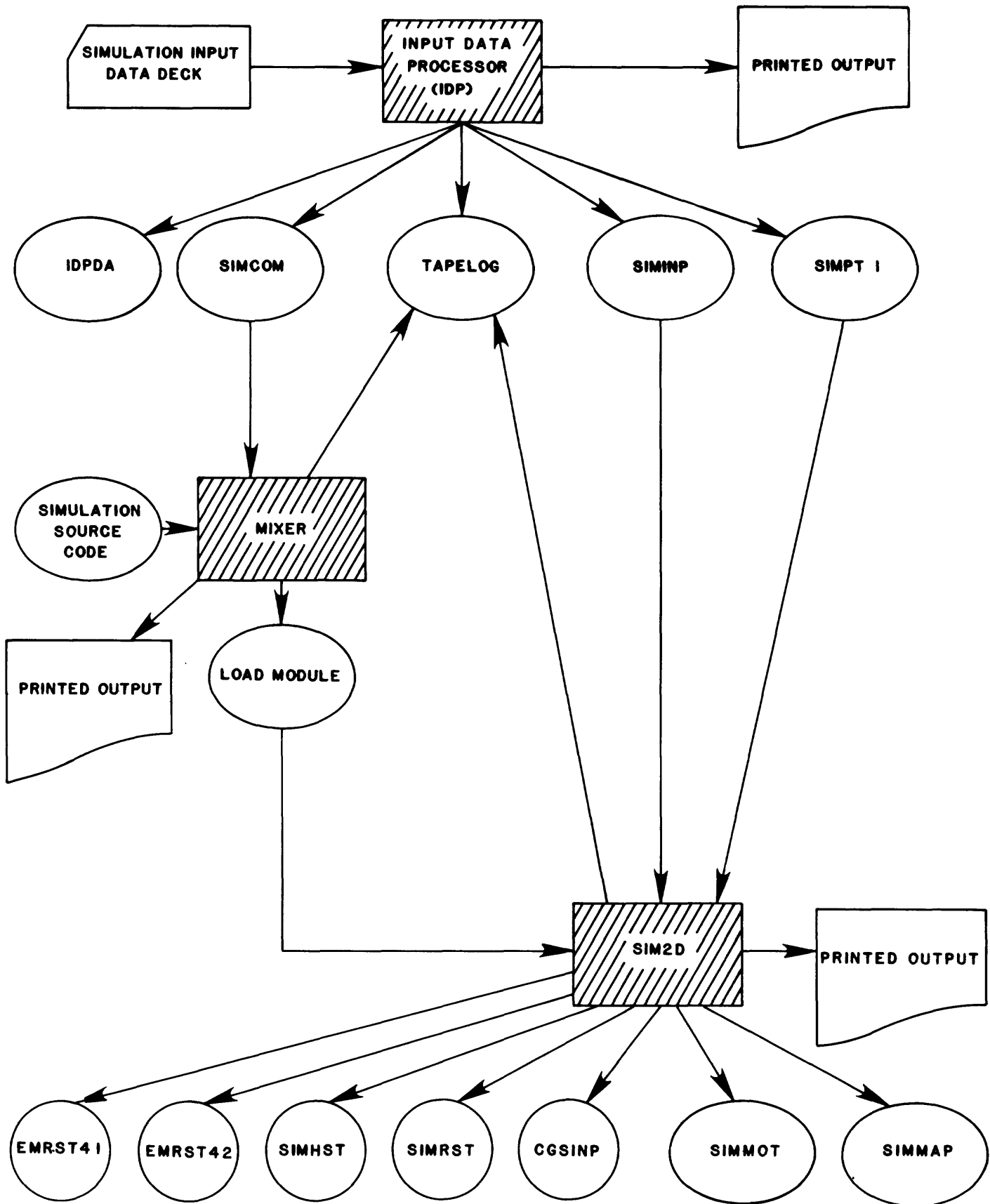


Figure 1.--SIMSYS2D input-output file diagram.

The SIM2D procedure can execute the simulation as initially defined in IDP or allow the user to change a limited number of processing parameters without re-running IDP and MIXER. The procedure produces seven output files and voluminous printed output that is user controlled to an extent. Three of the output files are restart tapes; one is a permanent file that contains all restart information written during execution of the simulation, and the other two are "emergency" restart files that contain only the calculated results for the current time step. The two emergency restarts are used in the event the system fails during the writing of restart information, so that no more than one time step will be lost. This can be an important consideration because about one-half CPU minute is expended for each time step.

PROCEDURAL IMPROVEMENTS

The original job control language (JCL) procedures developed by the Rand Corporation were written specifically for their IBM 370/155 computer and included programs and device group designations (esoteric names) that were not applicable to other computers. In addition, it was assumed that the candidate installation for SIMSYS2D would have the text editing system WYLBUR and access to the Benson-Lehner FR-80 micrographics system. While using SIMSYS2D, personnel of the U.S. Geological Survey found it desirable to move the modeling effort from one processor site to another to capitalize on price benefits. This required major modifications to the original Rand JCL furnished to the U.S. Geological Survey.

To make the model transferable, all installation dependent program processing or device group designations in the body of each procedure were changed to symbolic references that could be changed by the user in a predefined section of each procedure (attachments A, B, and C). Also, two utility procedures (attachments D and E) were created to allow loading or unloading of a magnetic tape with the required programs and files for transfer in a single batch job submission.

These improvements can drastically reduce the amount of time an estuarine modeler needs in modifying JCL when installing the SIMSYS2D system at a new site, or when changing file names and parameters during the modeling process.

The enhanced job control language for each of the SIMSYS2D procedures (IDP, MIXER, and SIM2D) is divided into two parts. The first part is the job control language statements needed to execute each program called for in the procedure. All nontemporary dataset names, region, time specifications, and esoteric device group designations are entered as symbolic parameters. Most other informational operands, such as data control block (DCB), SPACE, and DISPOSITION parameters (International Business Machines Corporation, 1977a) that have optimum or required values, are fixed.

At the end of each procedure is the second part where symbolics can be assigned or changed with each run. The user can change any of three types of symbolic parameters in this part. The first type is processing parameters such as region sizes, execution times, device group designations for temporary and permanent direct access devices, and parameters to pass to the compiler or linkage editor. The second type of symbolics is the library dataset names that

contain load modules needed for execution. Normally, the user would only have to specify these names the first time the procedure is tailored for a new installation. The third type of symbolics is the identifiers for individual datasets created by each run of the procedure.

All computer systems require that the user prefix a dataset name with a USERID, GROUPID/USERID, or some appropriate catalog pointer. No attempt has been made to make this dataset name prefix a symbolic parameter for two reasons; first, the procedures, programs, and datafiles can be more easily shared if only a single catalog pointer is used for shareable files such as libraries and graphics drivers. Individual users are responsible for adding their appropriate prefix to all output datasets. Secondly, some data-processing installations have a fixed limit on the absolute amount of disk storage space available to an individual user; the user then retains the flexibility of assigning different catalog pointers to files to allow input, output, and library datasets to be spread among a number of users to meet space allocation rules.

PROCESSING CONSIDERATIONS

Input Data Processor (IDP)

Care should be taken when editing the simulation override data contained in IDP part 1 (symbolic parameter &SIMPT1) for restart using WYLBUR; failure to replace this dataset with the proper blocking factor can cause problems. Because the &SIMPT1 file is concatenated to the &RVNSIM dataset in procedure SIM2D that contains the simulation variable names, the DCB attributes for the two datasets must be identical for the concatenation to work. If the attributes are different, the only diagnostic issued by the system is a cryptic error message "close failed for JS dataset" that does not reference which of the 22 files in the procedure caused the error. This problem arises because some implementations of WYLBUR, by default, replace card image datasets with a blocking factor of 39 (a BLKSIZE of 3120) (American Management Systems, Incorporated, 1975). To avoid this problem, check the DCB attributes of the dataset associated with the symbolic &RVNSIM; in TSO, this can be accomplished with a LISTDS command; under WYLBUR, it can be done with a "SHOW datasetname ALL" command. When the blocksize is known, the appropriate blocking factor can be added in parentheses to the CARD operand of the SAVE statement. Example--the &RVNSIM dataset has a blocksize of 800; therefore, it has a blocking factor of 10 because blocking factor = (BLOCKSIZE/LRECL). The proper WYLBUR command (Rand Computation Center, 1974) needed to save the IDP part 1 override data in a form that can be concatenated is:

```
SAVE * REP CARD(10)
```

A simple way to avoid the problem is to use the TSO editor only on the &SIMPT1 dataset. Even though it is a less powerful editor than WYLBUR, it generally leaves DCB attributes as it finds them when saving disk files.

Mixer Procedure

An important consideration in speed and cost of model execution is the choice of FORTRAN compilers used in the mixer procedure. The IBM FORTRAN H-Extended compiler (International Business Machines Corporation, 1974) with level 2 optimization can reduce the CPU time required for a simulation run by more than 30 percent over SIM2D code run under the FORTRAN H-level compiler. The H-Extended compiler (program IFEAABO) employs loop optimization techniques that are suited to iterative programs such as the SIMSY2D. The maximum optimization is specified in the symbolic parameter &PARMC in the mixer procedure by PARMC='OPT=2'.

SIM2D Procedure

The availability of magnetic tape transports during execution of the simulation is a major problem at some data-processing installations. Some processing centers may not have a sufficient number of 1600/6250 BPI tape drives to simultaneously load the two emergency restart tapes, the simulation restart tape, the course grid input tape, and any other files the user may wish to keep. A way to sidestep this problem is to define the large tape files as online temporary datasets that can be passed to additional steps at the end of the procedure for sequential loading to tape. This method has two advantages: first, the simulation I/O is performed at disk speeds, and second, the entire job can be executed with the allocation of a single high density tape drive. Similarly, restarts and coarse grid input tapes can be read into the procedure in sequential steps added to the front of the SIM2D procedure. To accomplish this, an execution of utility program IEBGENER (International Business Machines Corporation, 1977b) will be needed at the front of the procedure for each input dataset. The SYSUT1 DD statement would define the input tape file, whereas the SYSUT2 DD statement would be a temporary dataset (name beginning with two ampersands) with a disposition of "(new, pass)." At the end of the simulation, an IEBGENER step would be needed for each temporary dataset created by the simulation that is to be written to tape. Although this technique is not used in the procedure listed in attachment C, a version employing this method is available upon request from the author.

REFERENCES

- American Management Systems, Incorporated, 1975, WYLBUR Reference Manual.
- International Business Machines Corporation, 1974, IBM OS FORTRAN IV (H EXTENDED) Compiler Programmer's Guide: SC 28-6852-2.
- 1977a, OS/VS2 MVS JCL: GC 28-0692-3.
- 1977b, OS/VS2 MVS UTILITIES: GC 26-3902-0.
- Rand Computation Center, 1974, WYLBUR Learner's Guide: R-1555/6.
- Rand Corporation, 1977, SIMSYS2D: The two-dimensional water-quality simulation system: WN-9805-Neth/USGS.
- 1978, Course notes from "Estuarine Modeling with SIMSYS2D": sections 2-3.

ATTACHMENT A. INPUT DATA PROCESSOR (IDP) PROCEDURE

```

// ***** YOUR JOB CARD HERE *****
// *
// IDP      PROC ALLOC=ALLO30,REGG=350K,TIMEG=2,
//          TSOPAK=SYSTS,WORK=SYSDA,UTLIB=NULLFILE,
//          TAPELOG=NULLFILE,FR80LIB=NULLFILE,
//          SIMINP=NULLFILE,IDPDA=TEMPFILE,SIMCOM=NULLFILE,
//          SIMPT1=NULLFILE
// *-----*
// *
// * SAVED AS JCL.CNTL(IDP)
// *
// * PART OF SIMSYS2D SYSTEM FOR WATER QUALITY SIMULATION PROJECT.
// * INPUT DATA PROCESSOR - EDITS AND LISTS SIMULATION INPUT DATA.
// * THIS VERSION OF IDP WRITES SIMINP ON DISK; NO BACKUPS CREATED.
// *
// * M. C. FUJISAKI                      JUNE 30, 1978
// * REVISED BY P.H.CARRIGAN, JR  USGS    NOV. 13, 1978
// * REVISED BY GEORGE A. KARAVITIS USGS  JULY 17, 1980
// *-----*
// * THE FOLLOWING FILES, D1, D2, D3, D4, D5, ARE SPECIFIED HERE TO
// * DELETE AND UNCATALOG IF THEY EXIST - THEY ARE NOT USED BY THIS
// * JOBSTEP
// *
// DELETE EXEC PGM=IEFBRL4
// D1          DD DSN=&SIMPT1,DISP=(MOD,DELETE),
//              UNIT=&TSOPAK,SPACE=(TRK,(10,1))
// D2          DD DSN=&SIMCOM,DISP=(MOD,DELETE),
//              UNIT=&TSOPAK,SPACE=(TRK,(10,1))
// D3          DD DSN=&IDPDA,DISP=(MOD,DELETE),
//              UNIT=&TSOPAK,SPACE=(TRK,(10,1))
// D4          DD DSN=&SIMINP,DISP=(MOD,DELETE),
//              UNIT=&TSOPAK,SPACE=(TRK,(10,1))
// D5          DD DSN=&TAPELOG,DISP=(MOD,DELETE),
//              UNIT=&TSOPAK,SPACE=(TRK,(10,1))
// *-----*
// LOG1       EXEC PGM=LOG,
//              PARM=(1,IDP,'&WYLINP',1,'&IDPDA',2)
// STEPLIB    DD DSN=&UTLIB,DISP=SHR
// SYSPRINT   DD SYSOUT=A
// LOGFILE    DD DSN=&TAPELOG,DISP=(NEW,CATLG),UNIT=&TSOPAK,
//              DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120),
//              SPACE=(TRK,(4,4))
// *-----*
// LOG2       EXEC PGM=LOG,
//              PARM=('&SIMPT1',5,'&SIMCOM',4)
// STEPLIB    DD DSN=&UTLIB,DISP=SHR
// SYSPRINT   DD SYSOUT=A
// LOGFILE    DD DSN=&TAPELOG,DISP=(MOD,KEEP),
//              DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120),UNIT=&TSOPAK
// *-----*
// LOG3       EXEC PGM=LOG,
//              PARM=('&SIMINP',3)
// STEPLIB    DD DSN=&UTLIB,DISP=SHR
// SYSPRINT   DD SYSOUT=A
// LOGFILE    DD DSN=&TAPELOG,DISP=(MOD,KEEP),

```

```

//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120),UNIT=&TSOPAK
//*-----
//ALLOC EXEC PGM=&ALLOC
//STEPLIB DD DSN=&UTLIB,DISP=SHR
//FT05F001 DD DDNAME=SYSIN
//FT06F001 DD SYSOUT=A
//FT07F001 DD DUMMY
//FT11F001 DD DSN=&IDPDA,DISP=(NEW,CATLG),UNIT=&TSOPAK,
//          DCB=(RECFM=F,BLKSIZE=6440,BUFNO=1),
//          SPACE=(6440,(50,20))
//*-----

```

```

//GO EXEC PGM=IDP4,REGION=&REGG,TIME=&TIMEG
//STEPLIB DD DSN=&FR80LIB,DISP=SHR
//FT05F001 DD DSN=&WYLINP,DISP=(OLD,PASS)
//FT06F001 DD SYSOUT=A
//FT07F001 DD DSN=&SIMPT1,DISP=(NEW,CATLG),
//          DCB=(RECFM=F,LRECL=80,BLKSIZE=80),
//          UNIT=&TSOPAK,SPACE=(80,(50,100),RLSE)
//FT08F001 DD DSN=&SIMCOM,DISP=(NEW,CATLG),
//          DCB=(RECFM=F,LRECL=80,BLKSIZE=80),
//          UNIT=&TSOPAK,SPACE=(80,(50,100),RLSE)
//FT11F001 DD DSN=&IDPDA,DISP=OLD,UNIT=&TSOPAK,
//          DCB=(RECFM=F,BLKSIZE=6440,BUFNO=2)
//FT12F001 DD DUMMY
//FT13F001 DD SYSOUT=A,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=1330)
//FT14F001 DD DSN=&RRPLIB,DISP=SHR
//FT16F001 DD DSN=&&IN12,DISP=(NEW,DELETE),UNIT=&WORK,
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=800),
//          SPACE=(800,(5,100),RLSE)
//FT17F001 DD DSN=&&IN2A3,DISP=(NEW,DELETE),UNIT=&WORK,
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=800),
//          SPACE=(800,(5,100),RLSE)
//FT18F001 DD DSN=&&NEW2A3,DISP=(NEW,DELETE),UNIT=&WORK,
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=800),
//          SPACE=(800,(5,100),RLSE)
//FT19F001 DD DSN=&&ITPCOM,DISP=(NEW,DELETE),UNIT=&WORK,
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=800),
//          SPACE=(800,(5,100),RLSE)
//FT20F001 DD DSN=&SIMINP,DISP=(NEW,CATLG),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=800),
//          UNIT=&TSOPAK,SPACE=(TRK,(030,020),RLSE)
//FT21F001 DD DUMMY
//FT47F001 DD DSN=&TAPELOG,DISP=(MOD,KEEP),UNIT=&TSOPAK
//FT50F001 DD DSN=&&IFMT50,DISP=(NEW,DELETE),UNIT=&WORK,
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=800),
//          SPACE=(800,(5,100),RLSE)
//*-----

```

```

//PRLOG EXEC PGM=PRLOG
//STEPLIB DD DSN=&UTLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A
//LOGFILE DD DSN=&TAPELOG,DISP=SHR,
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120)
//ENDPROC PEND
//*-----

```

```

//IDPGO EXEC IDP,
//*****
//* USERS MAY CHANGE THE FOLLOWING PROCESSING PARAMETERS...
//*
//* REGG: MEMORY REQUIRED FOR IDP EXECUTION.
//* TIMEG: TIME IN MINUTES FOR IDP EXECUTION.
//* ALLOC: VERSION OF ALLOCATION PROGRAM TO EXECUTE.
//* TSOPAK: INSTALLATION DEVICE GROUP FOR TSO ACCESSABLE DISKS.
//* WORK: INSTALLATION DEVICE GROUP FOR TEMPORARY WORK SPACE.
//*
// REGG=350K,
// TIMEG=6,
// ALLOC=ALLO30,
// TSOPAK=SYSTS,
// WORK=SYSDA,
//*
//* USERS MUST INITIALLY PROVIDE THE FOLLOWING LIBRARY NAMES...
//*
//* UTLIB: RAND UTILITY LIBRARY.
//* FR80LIB: RAND GRAPHICS LIBRARY FOR FR80 PLOTTER.
//* RRPLIB: RAND RRP LIBRARY.
//*
// UTLIB='D0038.RMDUTIL',
// FR80LIB='D0038.RMDIDP@.PRODFR80',
// RRPLIB='D0038.RRPIDP@.V402V2',
//*
//* USERS MUST PROVIDE THE FOLLOWING DATASET NAMES EACH RUN.....
//*
//* WYLINP: INPUT IN PUNCH CARD FORMAT
//* IDPDA: IDP DISPLAY FILE, DIRECT ACCESS
//* SIMCOM: SIM2D COMMON - IDP DATA DEPENDENT
//* SIMPT1: SIM2D OVERRIDE DATA - IDP PART 1 (EXTRA COPY)
//* SIMINP: SIM2D INPUT DATA
//* TAPELOG: LOG OF SIM2D PROCESSING EVENTS
//*
// WYLINP='D0037.IDPINP.LOX71682',
// IDPDA='D0037.IDPDA.LOX71682',
// SIMCOM='D0037.SIMCOM.LOX71682',
// SIMPT1='D0037.SIMPT1.LOX71682',
// SIMINP='D0037.SIMINP.LOX71682',
// TAPELOG='D0037.TAPELOGG.LOX71682'
//*****
//

```

ATTACHMENT B. MIXER PROCEDURE

```

// ***** YOUR JOB CARD HERE *****
// *
// MIXER  PROC REGC=350K,TIMEC=4,PARMC=NOSOURCE,
//          TIMEL=1,PARML='MAP,LET,LIST,NCAL',
//          TSOPAK=SYSTS,WORK=SYSDA,REGL=150K,
//          REGCL=120K,TIMECL=1,FR80LIB=NULLFILE,
//          LIBL='SYS1.FORTLIB',TAPELOG=NULLFILE
// *-----
// *
// * SAVED AS JCL.CNTL(MIXER)
// *
// * PART OF SIMSYS2D SYSTEM FOR WATER QUALITY SIMULATION PROJECT
// * THIS PROCEDURE WILL COMBINE THE IDP DEPENDENT COMMON WITH THE
// * SOURCE, EXEC A FORTHCL TO CREATE A LOAD MODULE, AND EXEC PGM
// * IEBCOPY WITH ALTERNATE SYSIN CARD.
// *
// * M. C. FUJISAKI                JUNE 30, 1978
// * REVISED BY P.H.CARRIGAN JR    USGS    NOV. 17, 1978
// * REVISED BY GEORGE A. KARAVITIS USGS    DEC. 26, 1979
// *-----
// LOG1   EXEC PGM=LOG,
//          PARM=(1,MIXER,'&LOADDSN',17)
// STEPLIB DD DSN=&UTLIB,DISP=SHR
// SYSPRINT DD SYSOUT=A
// LOGFILE DD DSN=&TAPELOG,DISP=(MOD,KEEP),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120)
// *-----
// GO     EXEC PGM=MIXER
// STEPLIB DD DSN=&FR80LIB,DISP=SHR
// FT05F001 DD DDNAME=SYSIN
// FT06F001 DD SYSOUT=A
// FT08F001 DD DSN=&RANDSRC,DISP=(OLD,PASS)
// FT09F001 DD DSN=&RINMIX,DISP=(OLD,KEEP)
//          DD DSN=&SIMCOM,DISP=SHR,UNIT=&TSOPAK
// FT10F001 DD DSN=&&SOURCE,DISP=(NEW,PASS),UNIT=&WORK,
//          SPACE=(TRK,(10,5),RLSE),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=800)
// FT47F001 DD DSN=&TAPELOG,DISP=(MOD,KEEP),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120,BUFNO=2)
// *-----
// FORT   EXEC PGM=IFEAB,
//          REGION=&REGC,TIME=&TIMEC,PARM='&PARMC'
// SYSLIN  DD DSN=&&LOADSET,DISP=(MOD,PASS),
//          UNIT=&WORK,SPACE=(800,(40,80)),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=800)
// SYSPRINT DD SYSOUT=A
// SYSTEM DD SYSOUT=A
// SYSPUNCH DD SYSOUT=B
// SYSUT1  DD DSN=&&SYSUT1,UNIT=&WORK,SPACE=(1050,(10,20))
// SYSUT2  DD DSN=&&SYSUT2,UNIT=&WORK,SPACE=(1024,(10,20))
// SYSIN   DD DSN=&&SOURCE,DISP=(OLD,PASS)
// *-----
// LKED   EXEC PGM=IEWL,
//          REGION=&REGL,TIME=&TIMEL,PARM='&PARML',
//          COND=(4,LT,FORT)

```

```

//SYSLIB DD DSN=&LIBL,DISP=SHR
//SYSLIN DD DSN=&&LOADSET,DISP=(OLD,DELETE)
// DD DDNAME=SYSIN
//SYSLMOD DD DSN=&LOADDSN,DISP=(OLD,KEEP)
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=&&SYSUT1,UNIT=&WORK,SPACE=(1024,(50,20))
//*-----
//CLEAN EXEC PGM=IEBCOPY,REGION=&REGCL,TIME=&TIMECL
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=&LOADDSN,DISP=OLD,UNIT=&TSOPAK
//SYSUT3 DD DSN=&&SPILL3,DISP=(,DELETE,DELETE),UNIT=&WORK,
// SPACE=(TRK,(3,2))
//SYSUT4 DD DSN=&&SPILL4,DISP=(,DELETE,DELETE),UNIT=&WORK,
// SPACE=(TRK,(1,1))
//*-----
//PRLOG EXEC PGM=PRLOG
//STEPLIB DD DSN=&UTLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A
//LOGFILE DD DSN=&TAPELOG,DISP=SHR,
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120)
//ENDPROC PEND
//*-----
//MIXGO EXEC MIXER,
//*****
/** USERS MAY CHANGE THE FOLLOWING PROCESSING PARAMETERS...
/**
/** REGC: MEMORY REQUIRED FOR FORTRAN H-XTD LEVEL COMPILATION.
/** TIMEC: TIME REQUIRED FOR FORTRAN COMPILATION.
/** PARMC: FORTRAN H LEVEL COMPILER OPTIONS.
/** REGL: MEMORY REQUIRED FOR LINK EDIT STEP.
/** TIMEL: TIME IN MINUTES FOR LINK EDIT STEP.
/** PARML: LINKAGE EDITOR OPTIONS.
/** REGCL: MEMORY REQUIRED FOR IEBCOPY COMPRESS.
/** TIMECL: TIME IN MINUTES FOR IEBCOPY COMPRESS.
/** TSOPAK: INSTALLATION DEVICE GROUP FOR TSO ACCESSABLE DISKS.
/** WORK: INSTALLATION DEVICE GROUP FOR TEMPORARY WORK SPACE.
/**
/** REGC=350K,
/** TIMEC=4,
/** PARMC='NOSOURCE,OPTIMIZE(2)',
/** REGL=120K,
/** TIMEL=1,
/** PARML='MAP,LET,LIST,NCAL',
/** REGCL=120K,
/** TIMECL=1,
/** TSOPAK=SYSTS,
/** WORK=SYSDA,
/**
/**
/** USERS MUST INITIALLY PROVIDE THE FOLLOWING LIBRARY NAMES...
/**
/** LIBL: SYSTEM FORTRAN LIBRARY.
/** UTLIB: RAND UTILITY LIBRARY.
/** FR80LIB: RAND GRAPHICS LIBRARY FOR FR80 PLOTTER.
/** RANDSRC: RAND SIMULATION SOURCE CODE DATASET.

```

```

/**      RINMIX:   PERMANENT COMMON STMTS FOR SIM2D.
/**      LOADDSN:  LOAD MODULE LIBRARY TO CONTAIN OUPUT OF MIXER.
/**              (NOTE -- THE LIBRARY IS PERMANENT, BUT A NEW MEMBER
/**              NAME MUST BE SPECIFIED EACH RUN ON LIKAGE EDITOR
/**              NAME CONTROL STATEMENT BELOW.)
/**
/**              LIBL='SYS1.FORTLIB',
/**              UTLIB='D0038.RMDUTIL',
/**              FR80LIB='D0038.RMDIDP@@.PRODFR80',
/**              RANDSRC='D0038.RSCSIM2D.V404CARD',
/**              RINMIX='D0038.RINMIXER.CARDFMT',
/**              LOADDSN='D0037.LOAD3',
/**
/**
/**      USERS MUST PROVIDE THE FOLLOWING DATASET NAMES EACH RUN...
/**
/**      TAPELOG:  LOG OF SIM2D PROCESSING EVENTS.
/**      SIMCOM:   SIM2D COMMON INFORMATION CREATED BY IDP.
/**      USING THE LINKAGE EDITOR NAME CONTROL STATEMENT, NAME
/**      THE NEW LOAD MODULE TO BE CREATED BY THIS RUN.
/**
/**              TAPELOG='D0037.TAPELOGG.LOX71682',
/**              SIMCOM='D0037.SIMCOM.LOX71682'
//LKED.SYSIN DD *
ENTRY MAIN
NAME LOX71682                                SPECIFY MEMBER NAME HERE.
/*
//*****
//CLEAN.SYSIN DD *
C O=SYSUT2,I=SYSUT2
/*
//

```


ATTACHMENT C. SIM2D PROCEDURE

```

// ***** YOUR JOB CARD HERE *****
// *
// SIM2D  PROC TIMEL=2,PARML='MAP,LIST,SIZE=(350K)',
//          TAPELOG=NULLFILE,
//          REGG=1800K,TIMEG=95,PARMG=,
//          TSOPAK=SYSTS,WORK=SYSDA,
//          RSREEL1=,RSREEL2=,
//          REGL=450K,TDRIVE1=TAPE9,TDRIVE2=TAPE62,
//          EMRST41=NULLFILE,EMRST42=NULLFILE,
//          RSTDISP='NEW,CATLG',HSTDISP='NEW,CATLG',
//          SIMRST=NULLFILE,SIMHST=NULLFILE,CGSINP=NULLFILE,
//          SIMMOT=NULLFILE,SIMMAP=NULLFILE
// *-----
// *
// *   SAVED AS JCL.CNTL(SIM2D)
// *
// *   PART OF SIMSYS2D SYSTEM FOR WATER QUALITY SIMULATION PROJECT
// *   TWO-DIMENSIONAL WATER QUALITY SIMULATION PROGRAM
// *   THIS PROCEDURE WILL LINK EDIT THE LOAD MODULES AND EXEC THE
// *   SIM2D - COURSE GRID OUTPUT TO TAPE, NO EMERGENCY RESTART.
// *
// *   M. C. FUJISAKI                JULY  7, 1978
// *   REVISED BY P.H.CARRIGAN JR  USGS   NOV. 20, 1978
// *   REVISED BY GEORGE A. KARAVITIS USGS  DEC. 27, 1979
// *-----
// *   THE FOLLOWING FILES, D1, D2, D3, ARE SPECIFIED HERE TO
// *   DELETE AND UNCATALOG IF THEY EXIST - THEY ARE NOT USED BY THIS
// *   JOBSTEP
// *
// DELETE EXEC PGM=IEFBRI4
// D1          DD DSN=&SIMHST,DISP=(MOD,DELETE),
//              UNIT=&TSOPAK,SPACE=(TRK,(10,1))
// D2          DD DSN=&SIMMOT,DISP=(MOD,DELETE),
//              UNIT=&TSOPAK,SPACE=(TRK,(10,1))
// D3          DD DSN=&SIMMAP,DISP=(MOD,DELETE),
//              UNIT=&TSOPAK,SPACE=(TRK,(10,1))
// *-----
// LKED      EXEC PGM=IEWL,
//           REGION=&REGL,TIME=&TIMEL,PARM='&PARML'
// SYSLIB    DD DSN=&LIBL,DISP=SHR
//           DD DSN=&FR80LMD,DISP=SHR
// SYSLIN    DD DSN=&RLKSIM,DISP=SHR
//           DD DDNAME=SYSIN
// SYSLMOD   DD DSN=&&GOSET(GO),DISP=(NEW,PASS),
//           UNIT=&WORK,SPACE=(3072,(50,20,1))
// SYSPRINT  DD SYSOUT=A,DCB=BLKSIZE=1210
// SYSUT1    DD UNIT=(&WORK,SEP=(SYSLIB,SYSLMOD)),SPACE=(1024,(50,20))
// LOADDSN   DD DSN=&LOADDSN,DISP=SHR,UNIT=&TSOPAK
// SHARED    DD DSN=&SHARE,DISP=SHR
// OBJMOD    DD DSN=&RMDOBJ,DISP=SHR
// COMPOROY  DD DSN=&RMDSIM,DISP=SHR
// *-----
// LOG1      EXEC PGM=LOG,COND=(8,LT,LKED),
//           PARM=(1,SIM2D,'&SIMRST',7,'&SIMHST',8)
// STEPLIB   DD DSN=&UTLIB,DISP=SHR

```

```

//SYSPRINT DD SYSOUT=A
//LOGFILE DD DSN=&TAPELOG,DISP=(MOD,KEEP)
//*-----
//LOG2 EXEC PGM=LOG,COND=(8,LT,LKED),
// PARM=('&CGSINP',9,'&SIMMOT',10)
//STEPLIB DD DSN=&UTLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A
//LOGFILE DD DSN=&TAPELOG,DISP=(MOD,KEEP)
//*-----
//LOG3 EXEC PGM=LOG,COND=(8,LT,LKED),
// PARM('&SIMMAP',11)
//STEPLIB DD DSN=&UTLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A
//LOGFILE DD DSN=&TAPELOG,DISP=(MOD,KEEP)
//*-----
//LOG4 EXEC PGM=LOG,COND=(8,LT,LKED),
// PARM('&EMRST41',12,'&EMRST42',13)
//STEPLIB DD DSN=&UTLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A
//LOGFILE DD DSN=&TAPELOG,DISP=(MOD,KEEP)
//*-----
//GO EXEC PGM=*.LKED.SYSLMOD,COND=(8,LT,LKED),
// REGION=&REGG,TIME=&TIMEG,PARM='&PARMG'
//FT05F001 DD DSN=&RVNSIM,DISP=SHR
// DD DSN=&SIMPT1,DISP=SHR,UNIT=&TSOPAK
//FT06F001 DD SYSOUT=A
//FT07F001 DD SYSOUT=B
//FT08F001 DD DSN=&SIMRST,DISP=(&RSTDISP),UNIT=&TDRIVE1,
// DCB=(RECFM=VS,LRECL=X,BLKSIZE=800),
// VOL=SER=,LABEL=(01,SL,RETPD=365)
//FT09F001 DD DSN=&SIMHST,DISP=(&HSTDISP),
// DCB=(RECFM=VS,LRECL=X,BLKSIZE=800),
// UNIT=&TSOPAK,SPACE=(TRK,(10,5),RLSE)
//FT10F001 DD DSN=&SIMINP,DISP=SHR,UNIT=&TSOPAK
//FT12F001 DD DSN=&CGSINP,DISP=(NEW,KEEP),UNIT=&TDRIVE2,
// DCB=(RECFM=VS,LRECL=X,BLKSIZE=800),
// VOL=SER=,LABEL=(01,SL,RETPD=365)
//FT15F001 DD DSN=&SIMMOT,DISP=(NEW,CATLG),
// UNIT=&TSOPAK,SPACE=(TRK,(10,5),RLSE),
// DCB=(RECFM=VS,LRECL=X,BLKSIZE=800)
//FT40F001 DD DSN=&SIMMAP,DISP=(NEW,CATLG),UNIT=&TSOPAK,
// SPACE=(TRK,(5,2),RLSE),
// DCB=(RECFM=F,BLKSIZE=6440,LRECL=6440,BUFNO=2)
//FT41F001 DD DSN=&EMRST41,DISP=(NEW,KEEP),UNIT=&TDRIVE1,
// VOL=SER=&RSREEL1,DCB=(RECFM=F,BLKSIZE=822)
//FT42F001 DD DSN=&EMRST42,DISP=(NEW,KEEP),UNIT=&TDRIVE2,
// VOL=SER=&RSREEL2,DCB=(RECFM=F,BLKSIZE=822)
//FT47F001 DD DSN=&TAPELOG,DISP=(MOD,KEEP),
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120,BUFNO=2),UNIT=&TSOPAK
//FT50F001 DD DSN=&&IFMT50,DISP=(NEW,DELETE),UNIT=&WORK,
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=800),
// SPACE=(800,(5,100),RLSE)
//FT51F001 DD DUMMY
//FT52F001 DD DSN=&&SAVEGFL,UNIT=&WORK,
// DCB=(RECFM=F,LRECL=1536,

```

```

//          BLKSIZE=1536),SPACE=(1536,(10,10)),DISP=(NEW,PASS)
//*-----
//PRLOG EXEC PGM=PRLOG,COND=(8,LT,LKED)
//STEPLIB DD DSN=&UTLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A
//LOGFILE DD DSN=&TAPELOG,DISP=SHR
//ENDPROC PEND
//*-----
//SIMGO EXEC SIM2D,
//*****
//* USERS MAY CHANGE THE FOLLOWING PROCESSING PARAMETERS...
//*
//*          TIMEL:      TIME IN MINUTES FOR LINK EDIT STEP.
//*          REGL:      MEMORY REQUIRED FOR LINK EDIT STEP.
//*          PARML:     LINKAGE EDITOR OPTIONS.
//*          TIMEG:     TIME IN MINUTES FOR SIMULATION STEP.
//*          REGG:     MEMORY REQUIRED FOR SIMULATION STEP.
//*          TSOPAK:    INSTALLATION DEVICE GROUP FOR TSO ACCESSABLE DISK.
//*          WORK:     INSTALLATION DIVICE GROUP FOR TEMPORARY WORK SPACE.
//*          TDRIVE1:  INSTALLATION DEVICE GROUP FOR 9 TRK, 1600 BPI TAPE.
//*          TDRIVE2:  INSTALLATION DEVICE GROUP FOR 9 TRK, 6250 BPI TAPE.
//*          RSTDISP:  DISPOSITION OF SIMULATION RESTART DATASET.
//*                   ( NEW,CATLG FOR INITIAL RUN, OLD FOR RESTART)
//*          HSTDISP:  DISPOSITION OF SIMULATION HISTORY DATASET.
//*                   ( NEW,CATLG FOR INITAIL RUN, OLD FOR RESTART)
//*          RSREEL1:  VOLSER OF EMERGENCY RESTART TAPE #1.
//*          RSREEL2:  VOLSER OF EMERGENCY RESTART TAPE #2.
//*                   (IF THE EMERGENCY RESTART FACILITY IS NOT BEING
//*                   USED, OMIT THE VOLSER NUMBER AND CODE A COMMA.)
//*
//          TIMEL=2,
//          REGL=450K,
//          PARML='MAP,LIST,SIZE=(350K)',
//          TIMEG=95,
//          REGG=1800K,
//          TSOPAK=SYSTS,
//          WORK=SYSDA,
//          TDRIVE1=TAPE9,
//          TDRIVE2=TAPE62,
//          RSTDISP='NEW,KEEP',
//          HSTDISP='NEW,CATLG',
//          RSREEL1=,
//          RSREEL2=,
//*
//*
//* USERS MUST INITIALLY PROVIDE THE FOLLOWING LIBRARY NAMES...
//*
//*          LIBL:      SYSTEM FORTRAN LIBRARY.
//*          FR80LMD:   RAND FR80 PLOTTER LOAD MODULE LIBRARY.
//*          UTLLIB:    RAND UTILITY LIBRARY.
//*          RLKSIM:    LINKAGE EDITOR CONTROL STATEMENTS DATASET.
//*          SHARE:     SIM2D FUNCTIONS OBJECT MODULE LIBRARY.
//*          RMDOBJ:    SIM2D SUBROUTINE OBJECT MODULE LIBRARY.
//*          RMDSIM:    SIM2D COMMON STATEMENTS OBJECT MODULE LIBRARY.
//*          RVNSIM:    SIM2D VARIABLE NAMES DATASET.

```

```

/**          LOADDSN:  LOAD MODULE LIBRARY CONTAINING MEMBER CREATED
/**                      FOR THIS RUN BY MIXER.
/**
//          LIBL='SYS1.FORTLIB',
//          FR80LMD='D0038.FR80IGS.LMOD',
//          UTLIB='D0038.RMDUTIL',
//          RLKSIM='D0038.RLKSIM2D.INCLUDE',
//          SHARE='D0038.RMDSHARE',
//          RMDOBJ='D0038.RMDOBJ',
//          RMDSIM='D0038.RMDSIM2D',
//          RVNSIM='D0038.RVNSIM2D.V402',
//          LOADDSN='D0037.LOAD3',
/**
/**
/**  USERS MUST PROVIDE THE FOLLOWING DATASET NAMES EACH RUN...
/**
/**          EMRST41:  FIRST EMERGENCY RESTART TAPE DATASET.
/**          EMRST42:  SECOND EMERGENCY RESTART TAPE DATASET.
/**          SIMHST:  SIMULATION HISTORY FILE.
/**          SIMRST:  SIMULATION RESTART FILE.
/**          CGSINP:  COARSE GRID FILE.
/**          SIMMOT:  SIM2D MODEL OUTPUT FILE.
/**          SIMMAP:  SIM2D MAP FILE.
/**          SIMINP:  SIM2D INPUT DATA.
/**          SIMPT1:  SIM2D OVERRIDE DATA (IDP PART 1.).
/**          TAPELOG:  LOG OF SIM2D PROCESSING EVENTS.
/**          NAME MEMBER OF LOAD MODEL LIBRARY THAT WAS CREATED IN
/**          MIXER FOR THIS RUN IN THE LINKAGE EDITOR CONTROL
/**          STATEMENT BELOW.
/**
//          EMRST41=NULLFILE,
//          EMRST42=NULLFILE,
//          SIMHST='D0037.SIMHST.LOX71682',
//          SIMRST='UD0037.SIMRST.L11',
//          CGSINP='UD0037.CGSINP.L11',
//          SIMMOT='D0037.SIMMOT.LOX71682',
//          SIMMAP='D0037.SIMMAP.LOX71682',
//          SIMINP='D0037.SIMINP.LOX71682',
//          SIMPT1='D0037.SIMPT1.LOX71682',
//          TAPELOG='D0037.TAPELOGG.LOX71682'
//LKED.SYSIN DD *
//          INCLUDE LOADDSN(LOX71682)          SPECIFY MEMBER NAME HERE.
/**
/*******
//

```

ATTACHMENT D. DISK TO TAPE PROCEDURE

```

// ***** YOUR JOB CARD HERE *****
//*-----
//*
//* PROCEDURE TO LOAD ONLINE SIMSYS2D DISK FILES TO
//* MAGNETIC TAPE. THE SEQUENTIAL FILES ARE MOVED
//* FIRST, AND THE PARTITIONED FILES ARE MOVED LAST.
//*
//*
//* THE FOLLOWING PARAMETERS MUST BE SUPPLIED BY THE USER...
//*
//* TDRIVE1 INSTALLATION DEVICE GROUP FOR 6250 TAPE UNIT
//* TSOPAK INSTALLATION DEVICE GROUP FOR TSO ACCESSABLE DISK
//* WORK INSTALLATION DEVICE GROUP FOR TEMP WORK DISK DRIVE
//* TAPEVOL THE VOLUME SERIAL NUMBER OF THE MAG TAPE TO WRITE ON
//* USERID THE CATALOG POINTER OR USER IDENTIFICATION PREFIX
//* DSN THE NAME OF THE FILE (WITHOUT USERID)
//* FILE THE FILE NUMBER ON THE MAGNETIC TAPE
//*
//* GEORGE A. KARAVITIS USGS 7/26/82
//*-----
//LSEQ PROC TDRIVE1='TAPE62',TSOPAK='SYSTS',TAPEVOL='U90403',
// USERID='D0038',DSN=,FILE=
//GO EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT1 DD DSN=&USERID.&DSN,DISP=OLD
//SYSUT2 DD DSN=&DSN,UNIT=&TDRIVE1,VOL=(,RETAIN,SER=&TAPEVOL),
// DISP=NEW,LABEL=(&FILE,SL)
//ENDPROC PEND
//*-----
//FILE01 EXEC LSEQ,DSN='RINMIXER.CARDFMT',FILE=01
//FILE02 EXEC LSEQ,DSN='RRPIDP@@.V402V2',FILE=02
//FILE03 EXEC LSEQ,DSN='RSCSIM2D.V404CARD',FILE=03
//FILE04 EXEC LSEQ,DSN='RSYSIN1',FILE=04
//FILE05 EXEC LSEQ,DSN='RVNSIM2D.V402',FILE=05
//FILE06 EXEC LSEQ,DSN='RLKSIM2D.INCLUDE',FILE=06
//*-----
//LPDS PROC TDRIVE1='TAPE62',TSOPAK='SYSTS',TAPEVOL='U90403',
// USERID='D0038',WORK='SYSDA',DSN=,FILE=,
//GO EXEC PGM=IEBCOPY,REGION=80K
//SYSPRINT DD SYSOUT=A
//OUTPDS DD DSN=&DSN,DISP=NEW,UNIT=&TDRIVE1,
// VOL=(,RETAIN,=SER=&TAPEVOL),LABEL=(&FILE,SL)
//INPDS DD DSN=&USERID.&DSN,UNIT=&TSOPAK,DISP=OLD
//SYSUT3 DD UNIT=&WORK,DISP=(NEW,DELETE),SPACE=(80,(60,45))
//ENDPROC PEND
//*-----
//FILE07 EXEC LPDS DSN='RMDUTIL',FILE=07
//SYSIN DD *
// COPY OUTDD=OUTPDS,INDD=INPDS
//FILE08 EXEC LPDS DSN='RMDIDP@@.PRODFR80',FILE=08
//SYSIN DD *
// COPY OUTDD=OUTPDS,INDD=INPDS
//FILE09 EXEC LPDS DSN='FR80IGS.LMOD',FILE=09
//SYSIN DD *

```

```
                COPY OUTDD=OUTPDS,INDD=INPDS
//FILE10 EXEC LPDS DSN='RMDSHARE',FILE=10
//SYSIN      DD *
                COPY OUTDD=OUTPDS,INDD=INPDS
//FILE11 EXEC LPDS DSN='RMDOBJ',FILE=11
//SYSIN      DD *
                COPY OUTDD=OUTPDS,INDD=INPDS
//FILE12 EXEC LPDS DSN='RMDSIM2D',FILE=12
//SYSIN      DD *
                COPY OUTDD=OUTPDS,INDD=INPDS
//
```

ATTACHMENT E. TAPE TO DISK PROCEDURE

```

// ***** YOUR JOB CARD HERE *****
//*-----*
//*
//* PROCEDURE TO UNLOAD ONLINE SIMSYS2D TAPE FILES TO
//* ONLINE DISK. THE SEQUENTIAL FILES ARE MOVED
//* FIRST, AND THE PARTITIONED FILES ARE MOVED LAST.
//*
//*
//* THE FOLLOWING PARAMETERS MUST BE SUPPLIED BY THE USER...
//*
//* TDRIVE1 INSTALLATION DEVICE GROUP FOR 6250 TAPE UNIT
//* TSOPAK INSTALLATION DEVICE GROUP FOR TSO ACCESSABLE DISK
//* WORK INSTALLATION DEVICE GROUP FOR TEMP WORK DISK DRIVE
//* TAPEVOL THE VOLUME SERIAL NUMBER OF THE MAG TAPE TO WRITE ON
//* USERID THE CATALOG POINTER OR USER IDENTIFICATION PREFIX
//* DSN THE NAME OF THE FILE (WITHOUT USERID)
//* FILE THE FILE NUMBER ON THE MAGNETIC TAPE
//*
//* GEORGE A. KARAVITIS USGS 7/26/82
//*-----*
//USEQ PROC TDRIVE1='TAPE62',TSOPAK='SYSTS',TAPEVOL='U90403',
// USERID='D0038',DSN=,FILE=
//GO EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT1 DD DSN=&DSN,UNIT=&TDRIVE1,VOL=(,RETAIN,SER=&TAPEVOL),
// DISP=OLD,LABEL=(&FILE,SL)
//SYSUT2 DD DSN=&USERID.&DSN,DISP=(NEW,CATLG),UNIT=&TSOPAK,
// SPACE=(TRK,(50,15),RLSE)
//ENDPROC PEND
//*-----*
//FILE01 EXEC USEQ,DSN='RINMIXER.CARDFMT',FILE=01
//FILE02 EXEC USEQ,DSN='RRPIDP@@.V402V2',FILE=02
//FILE03 EXEC USEQ,DSN='RSCSIM2D.V404CARD',FILE=03
//FILE04 EXEC USEQ,DSN='RSYSIN1',FILE=04
//FILE05 EXEC USEQ,DSN='RVNSIM2D.V402',FILE=05
//FILE06 EXEC USEQ,DSN='RLKSIM2D.INCLUDE',FILE=06
//*-----*
//UPDS PROC TDRIVE1='TAPE62',TSOPAK='SYSTS',TAPEVOL='U90403',
// USERID='D0038',WORK='SYSDA',DSN=,FILE=,
//GO EXEC PGM=IEBCOPY,REGION=80K
//SYSPRINT DD SYSOUT=A
//INPDS DD DSN=&DSN,DISP=OLD,UNIT=&TDRIVE1,
// VOL=(,RETAIN,=SER=&TAPEVOL),LABEL=(&FILE,SL)
//OUTPDS DD DSN=&USERID.&DSN,UNIT=&TSOPAK,DISP=(NEW,CATLG)
//SYSUT3 DD UNIT=&WORK,DISP=(NEW,DELETE),SPACE=(80,(60,45))
//ENDPROC PEND
//*-----*
//FILE07 EXEC UPDS DSN='RMDUTIL',FILE=07
//SYSIN DD *
COPY OUTDD=OUTPDS,INDD=INPDS
//FILE08 EXEC UPDS DSN='RMDIDP@.PRODFR80',FILE=08
//SYSIN DD *
COPY OUTDD=OUTPDS,INDD=INPDS
//FILE09 EXEC UPDS DSN='FR80IGS.LMOD',FILE=09

```

```
//SYSIN      DD *
              COPY OUTDD=OUTPDS,INDD=INPDS
//FILE10 EXEC UPDS DSN='RMDSHARE',FILE=10
//SYSIN      DD *
              COPY OUTDD=OUTPDS,INDD=INPDS
//FILE11 EXEC UPDS DSN='RMDOBJ',FILE=11
//SYSIN      DD *
              COPY OUTDD=OUTPDS,INDD=INPDS
//FILE12 EXEC UPDS DSN='RMDSIM2D',FILE=12
//SYSIN      DD *
              COPY OUTDD=OUTPDS,INDD=INPDS
//
```