Technical Report No. 282

CHEEPELM, A SYSTEM FOR COMPILING, STORING, AND EXECUTING A LARGE SIMCOMP SIMULATION

G. W. Cole

Natural Resource Ecology Laboratory

Colorado State University

Fort Collins, Colorado 80523

GRASSLAND BIOME

U.S. International Biological Program

April 1975

TABLE OF CONTENTS

ABSTRACT

CHEEPELM is a set of three control card decks which allow the SIMCOMP user to separate his job into an initial compilation, execution, and correction phase. Only those subroutines which require corrections are recompiled. CHEEPELM is meant for large simulations where the cost for many small corrections can become prohibitive.

INTRODUCTION

During the initial building of the ELM simulation model (Innis 1975) it became obvious, due to its large size, that many of the problems of a small simulation must be dealt with in a different and generally more efficient manner. For example, card handling became a problem due to the large number of cards (about 4000). These were maintained on an UPDATE library so that only the changes had to be manually handled.

One problem which was never satisfactorily dealt with was that of compiling the program and then immediately recompiling the total program when a small execution error was detected by the modeler but not by the various compilers. This recompiling cost about $40 (291 CP seconds and 260 I/O seconds). In addition to the cost factor, the model, as it grew, exceeded daytime time limits at CSU for compilations and would generally have to be run in the evening. This resulted in one-day turnaround.

The reduction of turnaround time and cost has been the objective for the development of the CHEEPELM system. While this system is presently structured for the ELM74, it may be readily used for any SIMCOMP simulation. This modification is discussed later.

WHAT IS CHEEPELM?

Basically it is set of three control card decks and three small programs which amend the SIMCOMP 3.0 (Gustafson and Innis 1973) control cards to allow (i) the separation of the initial compilation and execution into three separately executed jobs and (ii) the recompilation of only those subroutines of the total simulation which are in error, or which are in need of a change.

The separation of the initial total compilation into three decks allows the compilation to be run within CSU daytime limits. Recompilation of only the subroutines containing the desired changes should reduce the costs. At this writing, little experience has been gained, consequently no dollar savings can be quoted.

CHEEPELM has a few other features worth mentioning:

(1) Previously the ELM program was compiled in one deck and executed in a second deck. This method precluded the execution of the SIMCOMP diagnostic mode during the execution, due to the lack of the DEBUG file and a loader map. This problem has now been remedied.

(2) The storing of the compiled program resident on a file called MAIN previously precluded storage on a tape. By storing the LGO file instead, the cost advantage of storage is still maintained and the LGO file can be successfully retrieved from tape.

(3) Should the initial compilation fail in the second deck, one only has to return to the beginning of Deck 2 rather than the beginning of Deck 1.

(4) Previously, when a single ELM compilation and execution run was made and the SIMCOMP DEBUG diagnostic operated, it would generate excessive volumes of paper that may or may not be needed. This need depended on how much information SIMCOMP DEBUG gleaned and presented on the first sheet of the diagnostic dump. CHEEPELM, allows the first sheet to be printed and the total DEBUG dump plus the loader map to be stored for one day on a permanent file. The loader map is no longer printed out but can be obtained if needed by the usage of Deck 5 (Appendix II).

Before discussing how to use this system, perhaps a word as to what CHEEPELM cannot do is appropriate. Unless provisions are made by initially allocating dummy storage variables, new subroutines cannot be used which depend on new variables in blank common. Also, new flows cannot be created.

CHEEPELM cannot substitute for sloppy permanent file management. CHEEPELM uses permanent files to communicate between the various decks. Some of these files are quite large and can become expensive if left on the system for an indefinite period. In particular, YOURNEWELMUPDATELIB and YOURDEBUG are estimated at $0.50 and $5.25, respectively, per day. PURGE these files when you are through. If you feel that these should be stored for future runs, store them on tape. Storage and retrieval of files from tape which really is not too difficult is illustrated by Decks 7-9 (Appendix II).

AN OVERVIEW OF CHEEPELM

Fig. 1 illustrates the sequence in the usage of the six basic decks required to utilize CHEEPELM. The details of the decks, for those interested, or wanting to make minor changes such as operating with a model other than ELM, will be found in a later section. Here we only discuss what is required to use CHEEPELM.

It is assumed that the reader is familar with the usage of the CDC Scope 3.3 file system and UPDATE system, particularly the latter since CHEEPELM utilizes UPDATE extensively. The knowledge required for use of the UPDATE system is minimal. The UPDATE features used here generally consist of a correction run, requiring only the usage of the *Insert and *Delete UPDATE directives (Control Data Corporation 1971).

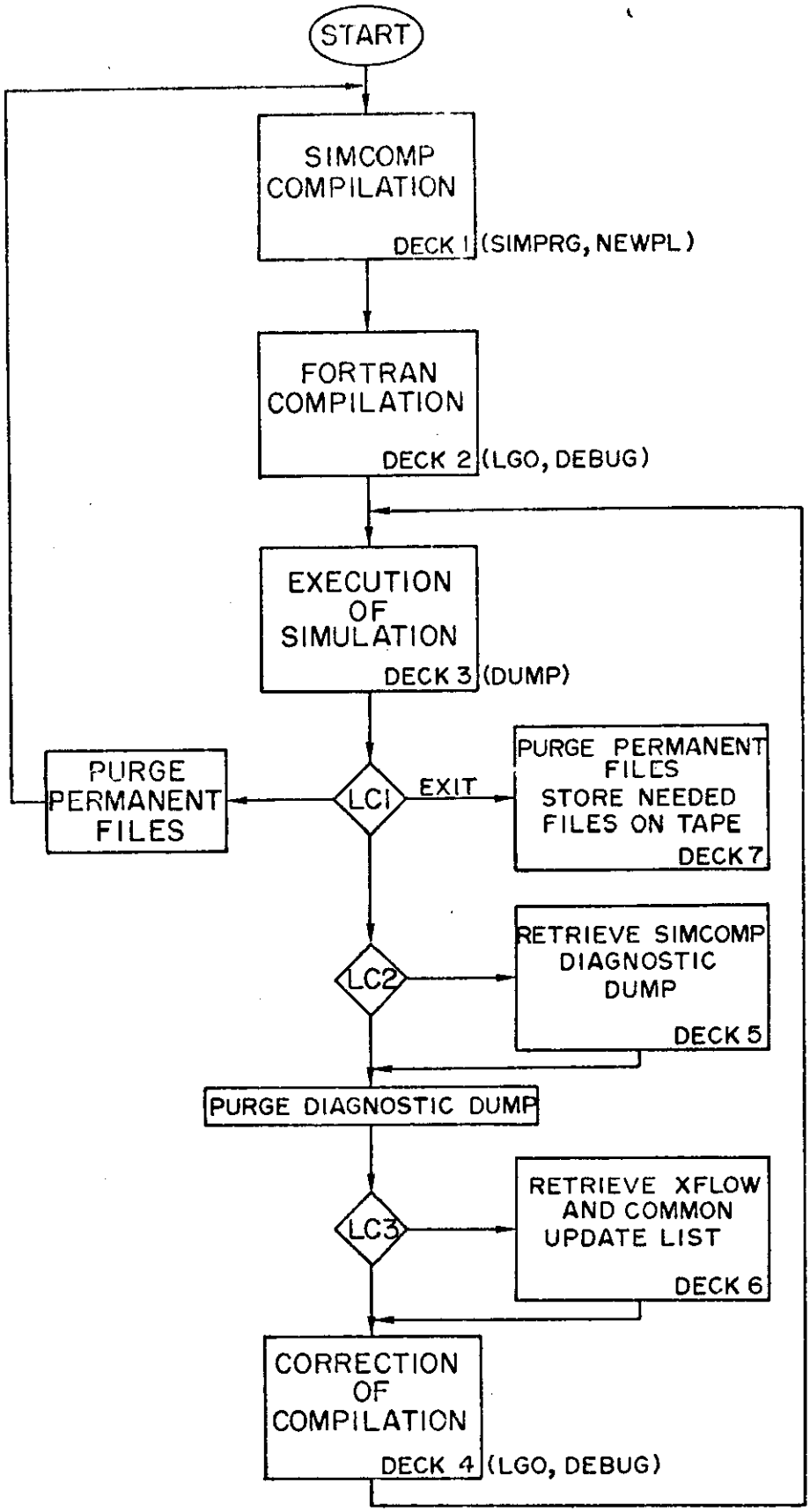

Fig. 1. The flow of execution of decks. Parenthetical names represent files cataloged for later use.

Returning to Fig. 1 as a reference to what follows, we note that there are two basic paths through the flow chart:

(1) The original compilation and execution mode, and

(2) The correction mode.

Starting with the former, we proceed by using Decks 1, 2, and 3 in that sequence. If the run has been successful and no corrections are needed (an unlikely case), we then EXIT Fig. 1 at the first logical choice (LC1), purge all unwanted files, and store the remaining files on tape.

The most likely path from LC1, however, is straight down, where LC2 is encountered. Here the modeler decides, after looking at the output from the simulation, if he requires information which may be stored in the diagnostic dump file. For very large simulations, such as ELM, the DEBUG diagnostic may be incomplete due to the limitations of the DEBUG system.

> WARNING: *The dump for ELM will be large, perhaps 75 pages. If you know what variables you are interested in, perhaps you could terminate the line printer when you have what you want.*

If the total dump has been printed out or if it can be determined that it is not needed, purge the file.

LC3 requires that a decision be made to obtain an UPDATE listing of the XFLOW and COMMON UPDATE decks stored on YOURNEWELMUPDATELIB. This update library was formed from the original update library plus the addition of two new decks. The first deck is called XFLOW and contains all the card images for subroutine XFLOW, a subroutine generated by SIMCOMP from your flow statements. The purpose of storing it is to prevent the need for regenerating subroutine XFLOW, should you require

changes to *EXISTING* flows. No new flows can be added. The listing of XFLOW will look strange at first; however, by studying it and your original SIMCOMP listing and/or flows in the original UPDATE listing, you should be able to distinguish where a flow begins and ends and what coding you generated vs. what SIMCOMP has added.

The second UPDATE deck is called COMMON and is required if a subroutine other than XFLOW is to be added or corrected during the correction mode. COMMON contains all the variables originally declared as STORAGE in the SIMCOMP source deck, plus a few others. Should a new subroutine(s) require the variables declared in STORAGE., then the common deck must be appended to the new subroutine by an update directive called *CALL COMMON directly following the subroutine name card. See Fig. 6 for an example.

Having decided whether or not to obtain a listing, one than proceeds to utilize Deck 4 to update the compiled file (LGO) and the debug file generated by FORTRAN (DEBUG) both of which were initially constructed and stored by the execution of Deck 2.

Having now corrected the appropriate files, one than returns to execute the simulation via Deck 3. This mode can be entered again by cycling through Decks 3 and 4 (also 5 and 6 perhaps) until all corrections are made; or if no further correction can be made via this method, then a return from LC1 to Deck 1 may be required. Ultimately it would be expected that all changes made by CHEEPELM would eventually have to be incorporated into the original source deck and the sequence of Decks 1-3 initiated. (Note: If one wants only a SIMCOMP listing, he need only execute Deck 1. Generally an UPDATE listing is just as satisfactory except for publication purposes).

HOW TO USE CHEEPELM

General Deck Structure

Due to the large number of control cards involved in Decks 1-4, it was decided to use the control card link utility (CCLINK) available at CSU. This allows one to put on file almost all control cards that are not expected to vary and to call this file into the control card stream via the CCLINK card. Further, to allow for changes to be readily made in the control cards which are on file, the cards are presently on an UPDATE library and can be changed at the time of Deck execution. Should either of these features not be desirable, they can be eliminated rather simply.

Fig. 2 illustrates the general structure of Decks 1-4 by an example of Deck 1. Decks 5-9 are too small to put on UPDATE.

The general strategy is to attach all special files necessary for the task at hand, then since cataloging can be done at the end, the last cards are catalogs (and purges if necessary). Since one cannot return to the previous control card stream after linkage via CCLINK, a second linkage is necessary to get to the catalog cards. This is provided by a CLCINK (CTLG) at the end of the file to which we linked originally. This then accounts for the catalog cards in the second record of the hands-on deck. (Hands-on deck, is used to differentiate between the deck which is on a permanent file and the deck which is read by the card reader. Where no confusion exists, no differentiation is made.)

The second record was chosen as a logical place to put the catalog cards, since this record follows immediately after the control card record. Due to this requirement, the movement of these cards is required prior to the first UPDATE, hence the COPYCR (INPUT, CTLG) control card.

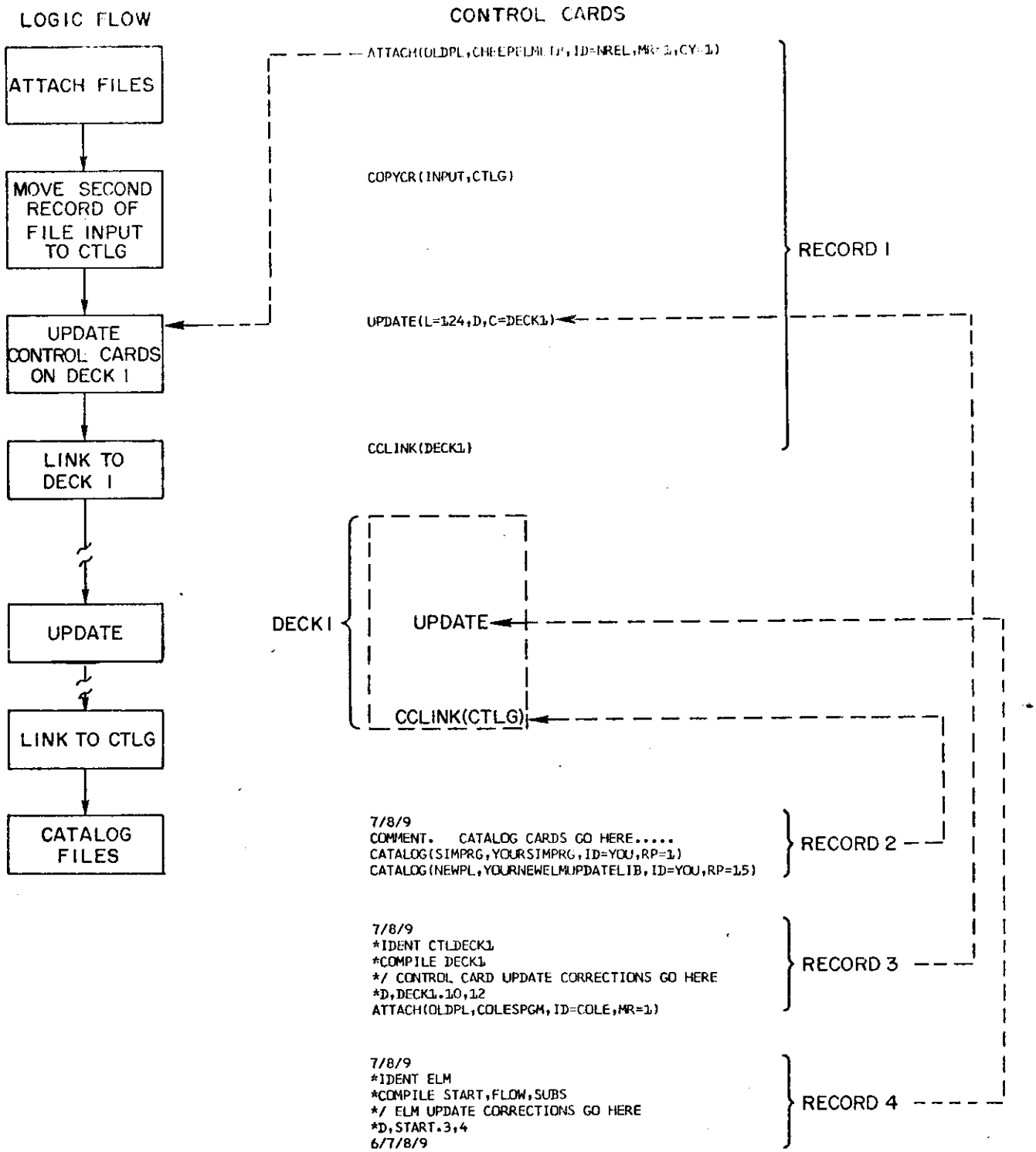

Fig. 2. A typical deck structure illustrating the use of the control card link and the control card update features. These common features are used in Decks 1-4. The dashed lines indicate which records are associated with which control cards.

Next the control cards stored in CHEEPELMLIB are updated (if needed) as illustrated by the example in Fig. 2, record 3. If no change is required, the *COMPILE DECK1 is still needed in the third record. This record is where the UPDATE insert and delete directives and new control cards are included. As illustrated we are deleting cards 10, 11, and 12 of Deck 1 and therefore are replacing the ELM74A UPDATE library file with another called COLESPGM.

The second UPDATE, which is initiated in the linked deck, is required to UPDATE the ELM74 (or other simulation) source deck. Record 4 contains the changes required for this particular simulation. Again the *COMPILE is required as a minimum, if no changes are required. The changes shown select three update decks called START, FLOW, and SUBS. Also we delete lines 3 and 4 from deck START.

Finally, we link back to file CTLG and perform the indicated catalogs.

> NOTE: *All permanent files have been given fictitious names and I.D.'s. Please do not use these names as they may conflict with others. As a minimum use your name as an I.D. The local file names must be used as in the examples.*

The details of all control cards in Decks 1-4 are portrayed in Fig. 3-6, respectively. All card images for the hands-on Decks 1-9 are to be found in Appendix II, DECK0, an update listing of the CHEEPELM library. To obtain the latest copy, perform an UPDATE correction run, on file CHEEPELMLIB cycle 1. For a list of the hands-on decks only, run an update on cycle 1 CHEEPELMLIB with a *COMPILE DECK0. Should the CHEEPELMLIB

not be on the system, it can be created by duplicating the cards in Appendices I and II, executing and storing in cycles 5 and 1.

Deck 1 Structure (Hands-on)

See Fig. 2 and 3 and the above General Deck Structure discussion for details.

Deck 2 Structure (Hands-on)

See Fig. 4. Deck 2 is essentially the same as Deck 1 without, however, any modification to a simulation source deck. We attach file SIMPRG (the output from SIMCOMP), which is the input for the FORTRAN compiler. We also attach as in Deck 1 the CHEEPELMLIB file. Files LGO and DEBUG are cataloged. LGO contains the FORTRAN compilation of the simulation and DEBUG contains the FORTRAN listing and reference maps needed for diagnostics in the case of a simulation error.

Deck 3 Structure (Hands-on)

See Fig. 5. Here we bring on the compiled file, DEBUG file, control deck, weather, and ELM data. The ELM data file is another UPDATE library containing the input variables required by ELM (in this case, the data for the Pawnee Site). We recommend creating such a file, since our experience with it has been very good. It represents "the" data deck with which others can be compared. It can be modified and still be available in original form. Best of all once it has been created, it is relatively insensitive to errors which can be made when handling the deck manually. Appendix III contains an example of this data file. Should this feature not be used, then the second UPDATE control card must be deleted and only the data cards are placed in the fourth record of

```
COMMENT. DECK1 JOB CARD T150 CM53000 MT1
COMMENT. DECK1 PERFORMS A SIMCOMP COMPILATION
ATTACH(OLDPL,CHEEPELMLIB,ID=NREL,MR=1,CY=1)
COPYCR(INPUT,CTLG)
UPDATE(L=124,D,C=DECK1)
CCLINK(DECK1)
            RFL(10000)
            RETURN(OLDPL)
            MAP(OFF)
            REWIND(CTLG)
            ATTACH(B,CHEEPELMLIB,ID=NREL,MR=1,CY=5)
            SKIPF(B,1,0,C)
            COPYCR(B,UDIN)
            REWIND(UDIN)
            REQUEST(ELMMIS,HY,ID-D1243,U,E,READ,MF)SWIFT
            LABEL(ELMUD,R,VSN=D1243,T=999,M=ELMMIS,L=ELM74AUPDATE )READ,SWIFT
            COPYBF(ELMUD,OLDPL)
            RFL(40000)
            UPDATE(D,L=124)
            REWIND(OUTPUT)
            COPYCF(OUTPUT,TEMP)
            REWIND(TEMP,OUTPUT)
            COPYSBF(TEMP,OUTPUT)
            RFL(10000)
            ATTACH(SIMCOM,SIMCOM3,CY=1,MR=1,ID=NREL)
            REQUEST(SIMPRG,*PF)
            RFL(53000)
            SIMCOM(COMPILE)
            RFL(10000)
            RETURN(COMPILE)
            COPYBR(B,SIFT2)
            REWIND(SIFT2)
            RFL(50000)
            SIFT2(SIMPRG,,XFLOW)
            RFL(10000)
            REWIND(XFLOW)
            COPYBR(B,SIFT1)
            REWIND(SIFT1)
            RFL(50000)
            SIFT1(XFLOW,,COMMON)
            REWIND(COMMON,XFLOW)
            UPDATE(N=MERGE,L=0,I=UDIN)
            RETURN(NEWPL)
            REQUEST(NEWPL,*PF)
            UPDATE(M=OLDPL,P=MERGE,N,L=0)
            RFL(20000)
            REWIND(SIMPRG)
            CCLINK(CTLG)
7/8/9
COMMENT.   CATALOG CARDS GO HERE.....
CATALOG(SIMPRG,YOURSIMPRG,ID=YOU,RP=1)
CATALOG(NEWPL,YOURNEWELMUPDATELIB,ID=YOU,RP=15)
TRANSF(TA***)     .... ADD THIS CARD ONLY IF NEEDED (SEE DECK 10)
7/8/9
*IDENT CTLDECK1
*COMPILE DECK1
*/ CONTROL CARD UPDATE CORRECTIONS GO HERE ... ONLY IF NEEDED
*D,DECK1.10,12
ATTACH(OLDPL,COLESPGM,ID=COLE,MR=1)
7/8/9
*IDENT ELM
*COMPILE START,FLOW,SUBS
*/  ELM UPDATE CORRECTIONS GO HERE
*D,START.3,4
6/7/8/9
```

Fig. 3. Total listing of Deck 1. Indented control cards are appended automatically to the non-indented control cards.

```
COMMENT. DECK2 JOB CARD  T210 CM53000
COMMENT. DECK2 PERFORMS A FORTRAN COMPILATION
ATTACH(SIMPRG,YOURSIMPRG,ID=YOU)
ATTACH(OLDPL,CHEEPELMLIB,ID=NREL,MR=1,CY=1)
COPYCR(INPUT,CTLG)
UPDATE(L=124,D,C=DECK2)
CCLINK(DECK2)
            RFL(10000)
            REWIND(CTLG)
            REQUEST(LGO,*PF)
            REQUEST(DEBUG,*PF)
            RFL(53000)
            FTN(I=SIMPRG,LN=DEBUG,R=1,A,S=0,ROUND=+-*/)
            RFL(20000)
            CCLINK(CTLG)
            EXIT,S.
            ATTACH(LIB,SIMCOM3,CY=3,MR=1,ID=NREL)
            SELECT(P=EDITOR,I=EDIT)
            EDIT.
7/8/9
COMMENT.  CATALOG CARDS FOR LGO AND DEBUG PLUS PURGE  SIMPRG GO HERE
CATALOG(DEBUG,YOURDEBUG,ID=YOU,RP=10)
CATALOG(LGO,YOURLGO,ID=YOU,RP=10)
PURGE(SIMPRG)
TRANSF(TA$$$)    ....  ADD THIS CARD ONLY IF NEEDED (SEE DECK 10)
7/8/9
*IDENT CTLDECK2
*COMPILE DECK2
*/ CONTROL CARD UPDATE CORRECTIONS GO HERE ... ONLY IF NEEDED
6/7/8/9
```

Fig. 4. Total listing of Deck 2. Indented control cards are appended automatically to the non-indented control cards.

```
COMMENT. DECK3 JOB CARD  T150  CM140000
COMMENT. DECK3 EXECUTES THE SIMULATION
ATTACH(EXFILE,YOURWEATHER,ID=YOU,MR=1)
ATTACH(BUG,YOURDEBUG,ID=YOU)
ATTACH(LLGO,YOURLGO,ID=YOU)
ATTACH(OLDPL1,ELM74PAWNEEDATA011475,ID=NREL,MR=1)
ATTACH(OLDPL,CHEEPELMLIB,ID=NREL,MR=1,CY=1)
COPYCR(INPUT,CTLG)
UPDATE(P=OLDPL,L=124,D,C=DECK3)
CCLINK(DECK3)
            RFL(10000)
            REWIND(CTLG)
            COPYBF(LLGO,LGO)
            ATTACH(FTNDUM,CHEEPELMLIB,CY=5,ID=NREL,MR=1)
            COPYCR(FTNDUM,FTNIN)
            REWIND(FTNIN)
            RFL(53000)
            FTN(I=FTNIN,LN=DEBUG,B=SIFT3)
            REWIND(SIFT3)
            RFL(10000)
            RETURN(DEBUG)
            COPYCF(BUG,DEBUG)
            ATTACH(LIB,SIMCOM3,CY=3,MR=1,ID=NREL)
            ATTACH(B,SIMCOM3,CY=2,MR=1,ID=NREL)
            RFL(40000)
            UPDATE(D,C=OUT,L=124,N,P=OLDPL1)
            RFL(10000)
            REWIND(OUTPUT)
            COPYCF(OUTPUT,FUD)
            REWIND(OUTPUT,FUD)
            RFL(15000)
            SELECT.
            COPYBF(B,LGO)
            MAP(PART)
            RFL(140000)
            LOAD(LGO)
            NOGO.
            RFL(10000)
            REWIND(OUTPUT)
            COPYCF(OUTPUT,SAVE)
            REWIND(SAVE,OUTPUT)
            COPYCF(SAVE,OUTPUT)
            REWIND(NEWT1)
            RFL(20000)
            SELECT(P=PRELOAD,I=PRELOAD)
            PRELOAD(NEWT1,MAIN)
            RFL(120000)
            MAIN(,OUT)
            RFL(10000)
            REWIND(OUTPUT)
            REQUEST(DUMP,*PF)
            COPYCF(OUTPUT,DUMP)
            COPYCF(OUTPUT,TEMP,10)
            REWIND(OUTPUT,TEMP)
            COPYSBF(FUD,OUTPUT)
            COPYCF(TEMP,OUTPUT)
            RFL(43000)
            MAP(OFF)
            SIFT3(TEMP)
            RFL(10000)
```

Fig. 5. Total listing of Deck 3. Indented control cards are appended automatically to the non-indented control cards.

```
            REWIND(TEMP)
            SKIPF(TEMP,1,17,C)
            COPYCF(TEMP,DUMP,10)
            RFL(20000)
            CCLINK(CTLG)
            EXIT.
            RFL(10000)
            REWIND(OUTPUT)
            REQUEST(DUMP,*PF)
            COPYCF(OUTPUT,DUMP)
            COPYCF(OUTPUT,TEMP,10)
            REWIND(OUTPUT,TEMP)
            COPYSBF(FUD,OUTPUT)
            COPYCF(TEMP,OUTPUT)
            RFL(43000)
            MAP(OFF)
            SIFT3(TEMP)
            RFL(10000)
            REWIND(TEMP)
            SKIPF(TEMP,1,17,C)
            COPYCF(TEMP,DUMP,10)
            RFL(20000)
            CCLINK(CTLG)
7/8/9
COMMENT.  CATALOG CARD FOR THE DIAGNOSTIC DUMP GOES HERE
CATALOG(DUMP,YOURDUMP,ID=YOU)
7/8/9
*IDENT CTLDECK3
*COMPILE DECK3
*/ CONTROL CARD UPDATE CORRECTIONS GO HERE ... ONLY IF NEEDED
*D,DECK3.26
RFL(65000)
*D.DECK3.38
RFL(65000)
7/8/9
*IDENT DATA
*COMPILE P72T1
*/ ELM DATA UPDATE
*D,P72T1.9
X(1)=5*0.0$
X(3)=400*0.$
*D,P72T1.31,398
            DT=1.$ TSTRT=0.$ TEND=5.$
DTPR=1.0$
PRINT. Y  ,X(1),X(2),X(3)
PLOT. (X(1))
6/7/8/9
```

Fig. 5. Continued.

Deck 3. For the details see item 2 of the "Simple Changes Which Can Be Made to CHEEPELM Decks" section.

Deck 4 Structure (Hands-on)

See Fig. 6. Deck 4 requires the attachment of the LGO, DEBUG, and YOURNEWUPDATELIB. The latter was created by Deck 1. In Deck 4 we allow for the correction of the compiled information on the LGO file and addition of corrected subroutines on the DEBUG file. In order to do this we must utilize the new update library created by Deck 1.

Two types of corrections can be handled. The first is related to corrections of existing subroutines. All subroutines in the original update library plus those that were "created" by SIMCOMP such as XFLOWS, START, CYCL1, CYCL2, and FINIS are correctable. The second update correction stream is in the fourth record (Fig. 6). It is identified as *IDENT ELM2. To utilize this correction mode, we must only include the subroutine(s) which is to be corrected. Therefore, all other subroutines must be excluded by *DELETE directives and/or the selective compile feature for UPDATE. For corrections to subroutine XFLOW, the selective compile feature is required since it is a separate deck. On the NEWELMUPDATELIB, there are two decks, ELM74 and XFLOW. In addition, an UPDATE common deck, appropriately called COMMON, is available and must be used when any subroutine other than XFLOW is being corrected since one of the tasks SIMCOMP performs is the amending to each subroutine of an unlabeled common block. This is accomplished by inserting a card *CALL COMMON directly after the card image which names the subroutine.

Deck 4 then processes the output from this UPDATE through the FORTRAN compiler and subsequently via the COPYL routine it replaces the old version of the subroutine on file LGO with the new version. This

```
COMMENT. DECK4 JOB CARD  T50?? CM53000
COMMENT. DECK4 PERFORMS CORRECTONS ON THE COMPILED SIMULATION
ATTACH(YBUG,YOURDEBUG,ID=YOU)
ATTACH(OLDLGO,YOURLGO,ID=YOU)
ATTACH(OLDPL1,YOURNEWELMUPDATELIB,ID=YOU,MR=1)
COPYCR(INPUT,CTLG)
ATTACH(OLDPL,CHEEPELMLIB,ID=NREL,MR=1,CY=1)
UPDATE(L=124,D,C=DECK4)
CCLINK(DECK4)
            *I,DECK4.41
             RFL(10000)
             MAP(OFF)
             REWIND(CTLG)
             RFL(53000)
             UPDATE(P=OLDPL1,L=124)
             FTN(I=COMPILE,B=OLDSUB,LN=DEBUG,R=1,A,S=0,ROUND=+-*/)
             RFL(10000)
             REWIND(OLDSUB)
             REQUEST(LGO,*PF)
             COPYL(OLDLGO,OLDSUB,LGO)
             REWIND(LGO,COMPILE)
             RFL(53000)
             UPDATE(P=OLDPL1,L=124)
             RFL(10000)
             REWIND(OUTPUT)
             COPYCF(OUTPUT,TEMP)
             REWIND(OUTPUT,TEMP)
             COPYSBF(TEMP,OUTPUT)
             RFL(53000)
             FTN(I=COMPILE,B=NEWSUB,LN=DEFOX,R=1,A,S=0,ROUND=+-*/)
             RFL(10000)
             REWIND(DEFOX)
             COPYSBF(DEFOX)
             REWIND(DEFOX)
             COPYCF(DEFOX,DEBUG)
             REWIND(DEBUG)
             REQUEST(NEWBUG,*PF)
             COPYCF(DEBUG,NEWBUG,100)
             COPYCF(YBUG,NEWBUG,100)
             RFL(15000)
             REWIND(NEWSUB)
             SELECT(B,I=NEWSUB)
             SELECT.
             RFL(20000)
             CCLINK(CTLG)
             EXIT,S.
             RETURN(LIB)
             ATTACH(LIB,SIMCOM3,CY=3,MR=1,ID=NREL)
             SELECT(P=EDITOR,I=EDIT)
             EDIT.
```

Fig. 6. Total listing of Deck 4. Indented control cards are appended automatically to the non-indented control cards.

```
7/8/9
COMMENT.PURGE AND CATALOG CARDS FOR YOURLGO AND YOURDEBUG GO HERE...
PURGE(OLDLGO)
CATALOG(LGO,YOURLGO,ID=YOU,RP=10)
PURGE(YBUG)
CATALOG(NEWBUG,YOURDEBUG,ID=YOU,RP=10)
7/8/9
*IDENT CTLDECK4
*/ CONTROL CARD UPDATE CORRECTIONS GO HERE ... ONLY IF NEEDED
*/ UPDATE NO. 1 IS TO CHANGE THE CONTROL CARDS IF YOU WISH.
*COMPILE DECK4
7/8/9
*IDENT ELM2
*/ UPDATE NO. 2 IS FOR ELM CORRECTIONS . DELETE ALL SUBROUTINES WHICH ARE NOT
*/ TO BE RECOMPILED
*COMPILE SUBS,XFLOW
*D,SUBS.2,5
*D,SUBS.7
*CALL COMMON
      Y=1000.
      CALL SMITH
*I,XFLOW.21
      CALL COLE
7/8/9
*IDENT NEWSUB
*/ UPDATE NO. 3 IS TO ADD A NEW SUBROUTINE IF AND ONLY IF IT IS REQUIRED BY
*/ THE CHANGE IN UPDATE  NO. 2 ABOVE.... COMPILE DUMMY IS NEEDED IF NO NEWSUBS
*/ ARE REQUIRED . IF ANY OF THE NEW SUBROUTINES REQUIRE CORRECTION ON A SUB-
*/ SEQUENT RUN ...MOVE  THESE SUBROUTINES INTO UPDATE 2 WITH THE *ADDFILE & *DECK
*/ NEW CARDS IN FRONT.
*ADDFILE INPUT
*DECK NEW
      SUBROUTINE SMITH
      END
      SUBROUTINE COLE
*CALL COMMON
      Y=3.**(Y)
      END
*COMPILE DUMMY
6/7/8/9
```

Fig. 6. Continued.

file is cataloged and stored for later use with Deck 3, the simulation execution deck.

In the previous paragraphs, we have discussed how to modify a subroutine which was in error. This modification can include the special case where one has compiled a subroutine which is not to be executed and he wishes to reduce the required core, since perhaps he has exceeded the computer's capability. For this situation we then delete all cards in the subroutine except the subroutine NAME and END cards. (Remember for functions, a dummy equivalence with the function name is required as part of the function.)

The second type of correction is handled by the next set of UPDATE directives in the hands-on deck, i.e., record 5 which starts with *IDENT NEWSUB. Here one adds a subroutine which is required by a change in the previous correction UPDATE. As a minimum, if no new subroutines are required, the *COMPILE DUMMY card is required, so that the subsequent FORTRAN compiler does not generate an error and cause the program to fail. It is advisable to leave this card in your deck as a separate card, even if you compile a new subroutine, since if the new subroutine must be corrected on a later pass through Deck 4, the *COMPILE DUMMY card will not be forgotten.

NOTE: *If a new subroutine must be corrected in a later pass through Deck 4, this new subroutine must now be corrected via UPDATE No. 2 since it will now be on the LGO file and must be replaced.*

This movement is accomplished as noted on the */ comment cards in the fifth record in Fig. 6.

Since the cards have been set up in Deck 3 to utilize the permanent file names selected for LGO and DEBUG, the equivalent files must be purged and recataloged. An example is shown in the second record of Deck 4 (Fig. 6).

Decks 5-9 Structure (Hands-on)

Since these decks are rather simple, reference to Appendix II, Deck 0, will show the deck structure. The comments therein are self-explanatory.

SYSTEM DESCRIPTION

The details of how CHEEPELM operates is encompassed in Fig. 7-10 and Appendices I and II. The figures have a flow chart type presentation and the appendix has the control cards for each deck with appropriate annotations. The figures indicate the functions being performed with the box symbol, and the lines interconnecting the boxes imply the generation and usage of the file(s) where names are shown on the line. The time sequence of events is not necessarily implied by the order of the boxes. For that information you must consult the listing of the specific decks in Appendix II. The dashed outlines in the figures imply that these functions are accomplished in the hands-on deck.

Deck 1

See Fig. 7. In Deck 1 the primary tasks are those of executing the SIMCOMP simulator to produce coding acceptable to the FORTRAN compiler and the storing of a new update library. This new update library will be used for making later correction if needed in Deck 4.

The creation of the FORTRAN input information is illustrated in Fig. 7 as the sequence from performing an UPDATE on the ELM74A update

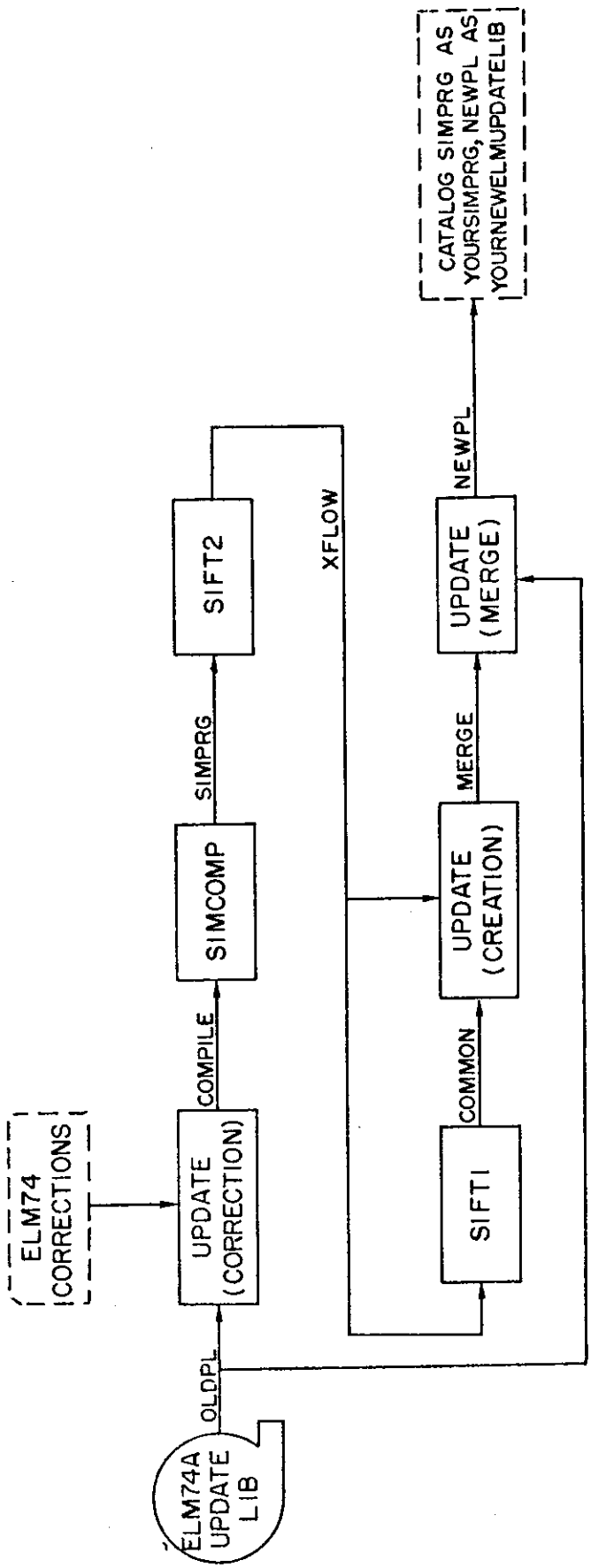

Fig. 7. Logic flow for Deck 1.

library (stored on tape) to the creation of file SIMPRG by SIMCOMP. SIMPRG is later stored on a permanent file in the hands-on deck.

The remainder of Fig. 7 depicts (i) the "sifting out" of subroutine XFLOW via program SIFT2 from file SIMPRG, (ii) the further "sifting out" of the COMMON statements from subroutine XFLOW (SIFT1), and (iii) the creation of a new UPDATE library NEWPL containing the contents of the original ELM74A update library in file OLDPL, plus two new decks called Deck XFLOW and COMDECK COMMON. The new update library is cataloged in the hands-on Deck 1.

One other function not shown is that of compacting the UPDATE output information on the output file.

The FORTRAN code for programs SIFT1 and SIFT2 can be found in Appendix I.

Deck 2

See Fig. 8. Deck 2 is the simplest of the four. The FORTRAN compiler compiles the program contained on file SIMPRG. FORTRAN produces the compiled file on LGO and the R=1 cross-reference map and listing on file DEBUG. Both files are stored on permanent files for usage by Decks 3 and 4. Since the output from FORTRAN has been shunted to file DEBUG, no error in compilation will be indicated on the printout, consequently SIMCOMP has a special program called EDIT which is executed if we obtain an error during the FORTRAN compilation.

Deck 3

See Fig. 9. Deck 3 performs the simulation. However, to accomplish this from the LGO file created in Deck 2 (or 4), the following functions must be performed:

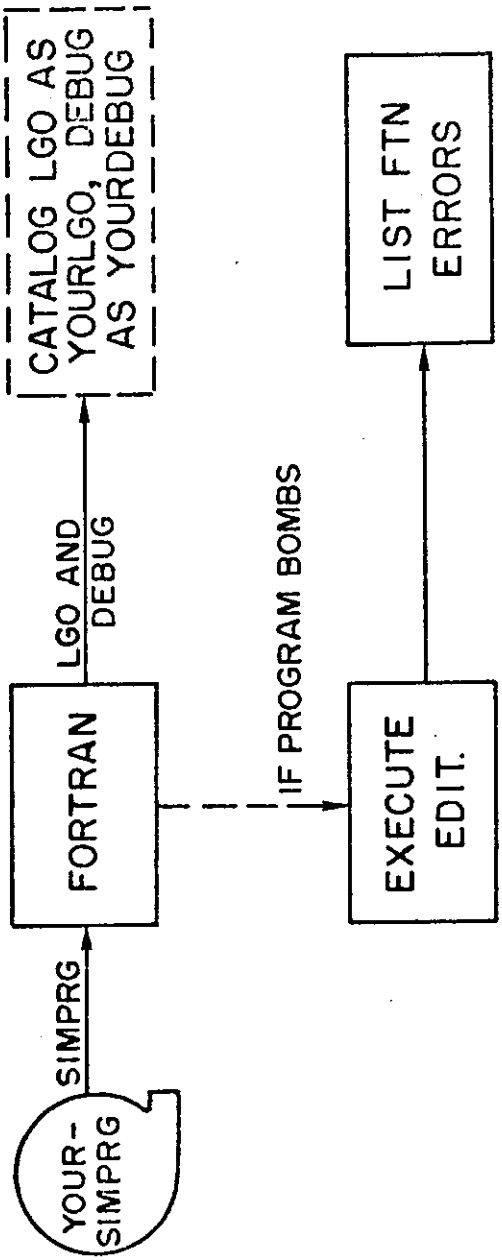

Fig. 8. Logic flow for Deck 2.

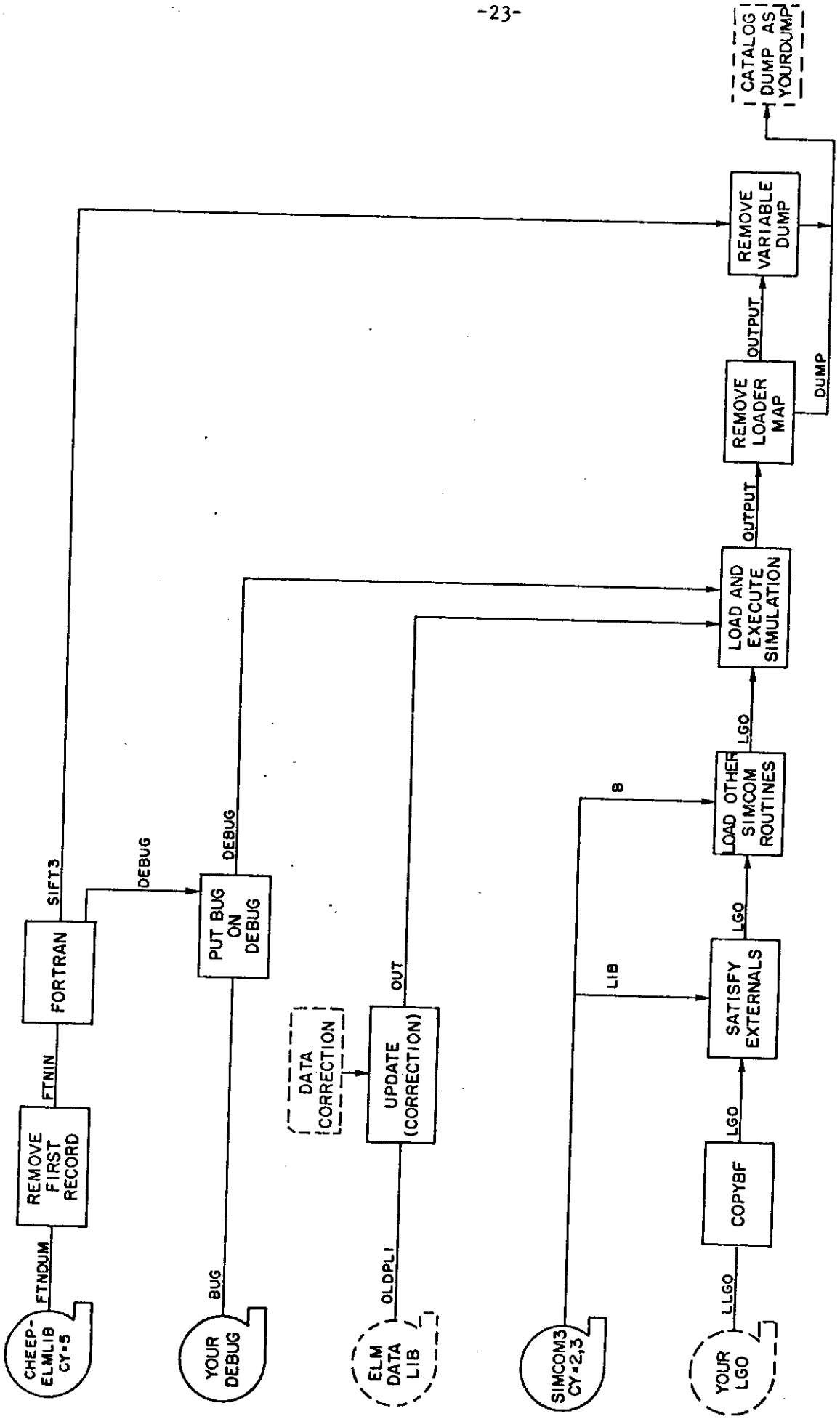

Fig. 9. Logic flow for Deck 3.

(1) Satisfying all unsatisfied externals.

(2) Loading the program.

(3) Supplying input data for the simulation.

Should the simulation "bomb-out," SIMCOMP has an error diagnostic capability. To accomplish this diagnosis, it needs the loader map generated by loading the program and the DEBUG file, which we stored after its generation by FORTRAN in Deck 2. With these, the diagnosis proceeds and if successful places on output a single sheet Diagnostic Dump and the Variable Dump which is multisheet and contains the contents of all variables in the simulation. The Variable Dump for an ELM simulation can be as much as 75 pages and may or may not be useful. Generally, only the Diagnostic Dump sheet is required, and the rest is useless paper and wasted printer time.

In Deck 3 then, we sort this information out and present as output only the absolute minimum to allow determination of the error, i.e., the Diagnostic Dump. The remainder, and largest amount of information by volume, is stored on a file called DUMP which is available for recall if the information originally printed out is not sufficient. In addition to the Variable Dump, we also store the Loader Map in the first record of the file. Both of these may be recalled by the use of Deck 5. If any plot requests were executed, these too will be on file DUMP and may or may not be useful.

Deck 4

See Fig. 10. This deck allows corrections of existing subroutines contained in the simulation on file OLDLGO and the addition of new subroutines which may be required by corrections to existing subroutines. To accomplish this, two CDC system utilities are utilized. For the

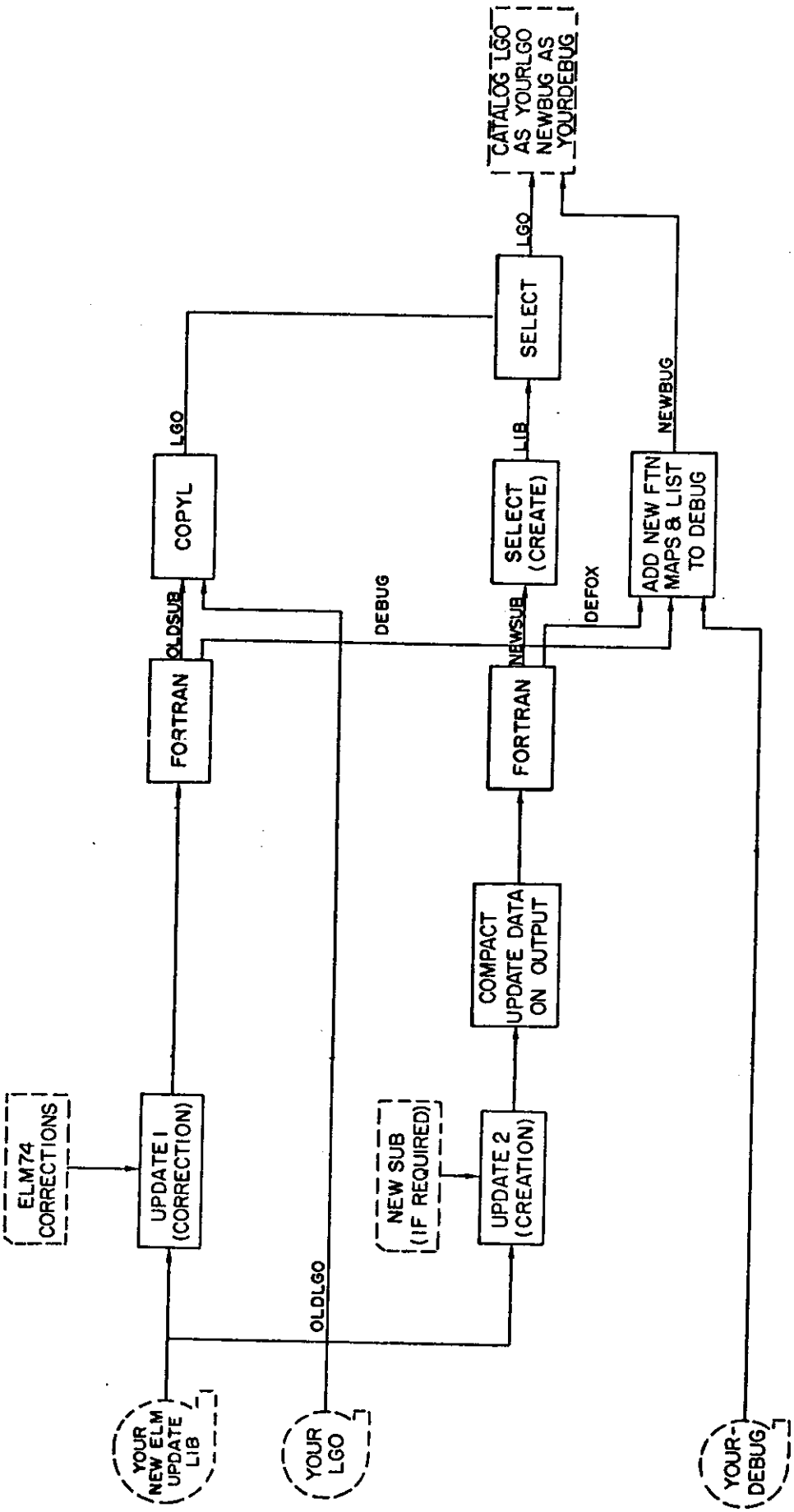

Fig. 10. Logic flow for Deck 4.

corrections we use COPYL which replaces compiled programs or subroutines with one of a similar name. For the addition to the compiled file we use the SELECT utility.

Since both utilities require compiled binary files, this implies that we must obtain the FORTRAN card images of the pertinent subroutines and recompile these via the FORTRAN compiler.

The reader will recall that in Deck 1 we created a new ELM74A update library. We use this program library to create the card images via UPDATE correction runs. The outputs from the UPDATES are then fed to the FORTRAN compiler. Finally the new LGO file which has been corrected is recataloged under its previous permanent file name, so that it can be utilized by Deck 3, the simulation execution.

One further function is also performed, that is the "correction" of the DEBUG file required by Deck 3 only if an error should occur in the simulation. The DEBUG file, created in Deck 2 contains the FORTRAN listings and R=1 cross-reference maps for all routines processed by FORTRAN in Deck 2. Here, in Deck 4, we have changed some of these routines. Consequently, new lists and maps have been generated. If we processed an error with a new compiled file but an old DEBUG file, it is not clear as to what the information from the Diagnostic Dump would mean.

As an attempt to overcome this, we amend the old DEBUG file by adding to the front of the file the new routines. This is done under the assumption that SIMCOMP, when searching the DEBUG file, will stop at the first routine with the required name. While this concept has not been exhaustively proven, the few cases tried with a simple program, appear to work.

We also, as in Deck 1, compact the update output information.

SIMPLE CHANGES WHICH CAN BE MADE TO CHEEPELM DECKS

These changes, which have been discussed previously in the text, represent typical changes which can be readily implemented. All references to UPDATE line numbers can be found in Appendix II.

Item 1: Replacement of ELM74 with Your Simulation

Assume that you have your SIMCOMP source deck on an update file. A typical example is already included in Appendix II. Reference DECK0 lines 40 and 41. Here we have a *DELETE directive removing lines 10, 11, and 12 of Deck 1. Reference to Deck 1 lines 10-12 will show that we are deleting the control cards which bring on the tape containing ELM74. We replace these with the ATTACH card shown on line Deck0.41. Since your simulation is on a permanent file, the job card should not request the mounting of a tape, i.e., MT1 is no longer needed (line Deck 0.25).

If your simulation is smaller than ELM74, you may not require as much core as is indicated in Deck 3 (line Deck0.76). The large values can be reduced from the present values of 140K for loading (line Deck3.26) and 120K for execution (line Deck3.38). Typical changes are shown between lines Deck0.93 and Deck0.96. Both the load and execution are changed to 65K.

Item 2: Deletion of Data File Input in Deck 3

Do not attach the data deck by not including the attached card indicated on line Deck0.81. After line Deck0.92 include the following three cards in the sequence shown:

```
*D,DECK3.16,18
*D,DECK3.39
MAIN.
```

Item 3: How to Submit Decks 1, 2, and 3 Simultaneously

Deck0 from lines Deck0.233 to Deck0.255 inclusive explain the required changes.

Item 4: How to Incorporate SIMCOMP 3.1

As presently implemented, CHEEPELM is implemented for SIMCOMP 3.0. The following changes allow usage of SIMCOMP 3.1, the data dump feature.

After line Deck0.39 include the following cards:

```
*D,DECK1.20
ATTACH(SIMCOM,SIM3P1,CY=1,MR=1,ID=NREL)
```

After line Deck0.92 include the following cards:

```
*D,DECK3.15
ATTACH(B,SIM3P1,CY=2,MR=1,ID=NREL)
REQUEST(SAVIT,*PF)
*I,DECK3.39
```

Include control for cataloging the data dump. See Stevens (1975).

LITERATURE CITED

Control Data Corporation. 1971. SCOPE reference manual, 6000 version 3.3, Chap. 10 Library preparation and maintenance, Revision F, Publication #60305200. Control Data Corporation, Sunnyvale, Calif.

Gustafson, J. D., and G. S. Innis. 1973. SIMCOMP version 3.0 user's manual. US/IBP Grassland Biome Tech. Rep. No. 218. Colorado State Univ., Fort Collins. 149 p.

Innis, G. S. [Coordinator]. 1975. ELM 73. US/IBP Grassland Biome Tech. Rep. Colorado State Univ., Fort Collins. (In prep.)

Stevens, K. 1974. SIMCOMP 3.1 (Memorandum dated September 5, 1974). Natural Resource Ecology Laboratory, Colorado State Univ., Fort Collins. 5 p.

APPENDIX I

CHEEPELMLIB Cycle 5

This appendix contains a listing of the control cards plus the source necessary to generate the permanent file CHEEPELMLIB cycle 5, and shows via the CATFILE output the names of the five records in the file.

```
TA123,T15,CM50000,ANMRA60Y.  COLE,GW.     $SAVE PUT SIFT1 & 2 ON FILE
REQUEST(X,*PF)
COPYCR(INPUT,X,2)
REWIND(INPUT)
COPYCR(INPUT,TEMP,3)
REWIND(TEMP)
COPYSBF(TEMP)
FTN.
FTN.
ATTACH(LIB,FTNLIB,ID=APPLE)
SELECT.
REWIND(LGO)
COPYCF(LGO,X)
CATALOG(X,CHEEPELMLIB,ID=NREL,CY=5,RP=999)
REWIND(X)
CATFILE,X)
       PROGRAM SIFT3 (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
C      SIFT3 READS INPUT FILE FOR THE DEBUG DIAGNOSTIC DUMP & PRINTS OUT THE FIRS
C      T SHEET.
       DIMENSION KARD(13),IVAR1(2)
       IVAR1(1)=10HARITHMETIC  $ IVAR1(2)=10H MODE ERRO
   50  READ(5,100)KARD
       IF(EOF(5).NE.0.)GO TO 26
       IF(CMPRF(IVAR1,1,KARD,2,20))50,20,50
  100  FORMAT(13A10)
   20  CONTINUE
       WRITE(6,100)KARD
       DO 25 I=1,47
       READ(5,100)KARD
       WRITE(6,100)KARD
   25  CONTINUE
   26  CONTINUE
       END
*COMDECK COMMON
*READ COMMON
*DECK XFLOW
*READ XFLOW
*DECK DUMMY
       PROGRAM  DUMMY  (OUTPUT,INPUT)
       Y=2.
        END
```

```
                      PROGRAM SIFT2(INPUT,OUTPUT,TAPE1,TAPE5=INPUT)
                C
                C.. THIS PGM SIFTS THROUGH THE OUTPUT OF A SIMCOMP COMPILATION AND WRITES
                C     SUBROUTINE XFLOWS TO TAPE1
                C
                      DIMENSION KARD(8),IVAR(2) ,IVAR1(2)
                      IVAR(1)=10HSUBROUTINE   $  IVAR(2)=10H XFLOWS
                      IVAR1(1)=10HCOMMON XEV$  IVAR1(2)=3HSTK
                      IVAR3=3HEND
                  50  READ(5,100) KARD
                 100  FORMAT(8A10)
                C     CHECK FOR START OF SUBROUTINE XFLOWS
                      IF(CMPRF(IVAR,1,KARD,7,17))50,20,50
                  20  WRITE(1,100)KARD
                  30  READ(5,100) KARD
                C     CHECK FOR LAST CARD
                  35  IF(CMPRF(IVAR3,1,KARD,7,3))37,40,37
                C     CHECK FOR GARBAGE SUCH AS PAGE HEADINGS
                  37  IF(CMPRF(IVAR,1,KARD,7,17))20,30,20
                  40  CONTINUE
                C     LAST CARD
                      WRITE(1,100)KARD
                      END
```

SYMBOLIC REFERENCE MAP

ENTRY POINTS
 6074 SIFT2

VARIABLES	SN	TYPE		RELOCATION				
6201 IVAR		INTEGER	ARRAY		6203	IVAR1	INTEGER	ARRAY
6170 IVAR3		INTEGER			6171	KARD	INTEGER	ARRAY

FILE NAMES	MODE									
0 INPUT		2022	OUTPUT		4044	TAPE1	FMT	0	TAPE5	FMT

EXTERNALS	TYPE	ARGS
CMPRF	REAL	5

STATEMENT LABELS

6114	20			6121	30			0	35	INACTIVE
0	37	INACTIVE		6133	40			6104	50	
6155	100	FMT								

STATISTICS

PROGRAM LENGTH	1178	79
BUFFER LENGTH	60668	3126

```
      PROGRAM SIFT1(INPUT,OUTPUT,TAPE1,TAPE5=INPUT)
C
C.. THIS PGM SIFTS THROUGH THE OUTPUT OF A SIMCOMP COMPILATION AND WRITES
C     THE COMMON CARDS FROM SUBROUTINE XFLOWS TO TAPE1 FOR STORING
C
      DIMENSION KARD(8),IVAR(2) ,IVAR1(2)
      IVAR(1)=10HSUBROUTINE    $  IVAR(2)=10H XFLOWS
      IVAR1(1)=10HCOMMON XEV$  IVAR1(2)=3HSTK
   50 READ(5,100) KARD
  100 FORMAT(8A10)
C     CHECK FOR START OF SUBROUTINE XFLOWS
      IF(CMPRF(IVAR,1,KARD,7,17))50,30,50
   20 WRITE(1,100)KARD
   30 READ(5,100) KARD
C     CHECK FOR LAST COMMON CARD
   35 IF(CMPRF(IVAR1,1,KARD,7,13))20,40,20
   40 CONTINUE
C     LAST COMMON CARD
      WRITE(1,100)KARD
      END
```

SYMBOLIC REFERENCE MAP

ENTRY POINTS
6074 SIFT1

VARIABLES	SN	TYPE		RELOCATION					
6173 IVAR		INTEGER	ARRAY		6175 IVAR1		INTEGER	ARRAY	
6163 KARD		INTEGER	ARRAY						

FILE NAMES	MODE								
0 INPUT		2022 OUTPUT			4044 TAPE1	FMT		0 TAPE5	FMT

EXTERNALS	TYPE	ARGS
CMPRF	REAL	5

STATEMENT LABELS										
6112 20			6117 30				0 35		INACTIVE	
0 40	INACTIVE		6102 50				6151 100	FMT		

STATISTICS

PROGRAM LENGTH	1118	73
BUFFER LENGTH	6066B	3126

CATALOG OF FILE X

RECORD NO.	LEVEL NO. OCTAL	LENGTH DECIMAL	LENGTH OCTAL(B)	PACKAGE	CHKSUM	DATE
1	0	62	76	*******	2433	
2	0	18	22	*******	6436	
3	0	158	236	SIFT2	7432	05/26/75
4	0	150	226	SIFT1	1304	05/26/75
5	0	61	75	CMPRF	5422	12/19/72
6	17	0	0		0	

LEVEL GROUP LENGTH IS 449/ 701B

***** END FILE *****

```
  05/26/75 *CSU SCOPE 3.3.14  B C01? C013 C140 C141 05/20/75
10.36.35.TA12360   FROM  AB       6A
10.38.36.TA123,T15,CM50000,ANMR****.  COLE,GW.
10.38.36. $SAVE PUT SIFT1 & 2 ON FILE
10.38.36.REQUEST(X,*PF)
10.38.37.COPYCR(INPUT,X,2)
10.39.03.REWIND(INPUT)
10.39.03.COPYCR(INPUT,TEMP,3)
10.39.17.REWIND(TEMP)
10.39.17.COPYSBF(TEMP)
10.39.17.FTN.
10.39.26.       .258 CP SECONDS COMPILATION TIME
10.39.26.FTN.
10.39.28.       .227 CP SECONDS COMPILATION TIME
10.39.28.ATTACH(LIB,FTNLIB,ID=APPLE)
10.39.30.CYCLE **, FTNLIB
10.39.30.PFN FOUND IN SD  024
10.39.30.CYCLE 01. FTNLIB
10.39.30.FILE HAS BEEN ATTACHED
10.39.30.SELECT.
10.39.30.FL= 023400 CP 00000.528SEC.  IO 00004.955SEC.
10.39.31. $S$BS$   CMPRF
10.39.31.FL= 000200 CP 00000.593SEC.  IO 00005.470SEC.
10.39.31.FL= 050000 CP 00000.594SEC.  IO 00005.470SEC.
10.39.31.REWIND(LGO)
10.39.32.COPYCF(LGO,X)
10.39.33.EOF/EOI ENCOUNTERED
10.39.33.CATALOG(X,CHEEPELMLIB,ID=NREL,CY=5,RP=99
10.39.33.9)
10.39.35.PFD 4/5 FULL
10.39.35.FILE CATALOGUED AS
10.39.35.CYCLE 05, CHEEPELMLIB
10.39.35.IN SD 023
10.39.35.REWIND(X)
10.39.35.CATFILE,X)
10.39.36.   CATFILE FINISHED
10.39.36.CP        .638 SEC.
10.39.36.PP      18.029 SEC.
10.39.36.IO       5.900 SEC.
```

APPENDIX II

CHEEPELMLIB Cycle 1

This appendix contains a copy of the listing generated when creating the CHEEPELMLIB cycle 1. The control cards are also shown.

```
*****    *DECK DECK0
*****    *TEXT


                    ***  CHEEPELM DECKS  ***

THIS UPDATE LIBRARY CONTAINS THE CONTROL CARDS FOR USING THE CHEEPELM SYSTEM.
(SEE NREL TECHNICAL REPORT #282   FOR DETAILS) IN *DECK DECK0 ARE THE "HANDS ON"
CONTROL CARDS AND TYPICAL UPDATE DIRECTIVES WHICH ARE FED INTO THE CARD READER.
ALL OTHER *DECK DECKS ARE THE DECKS WHICH THE HANDS ON DECKS LINK TO.  SOME OF
THE CARDS WHICH ARE STORED IN DECK0 ARE ONLY ILLUSTRATIVE AND SHOULD BE
REPLACED WITH YOUR OWN, SUCH AS SPECIFIC UPDATE CORRECTIONS FOR YOUR PROGRAM
AND ALL PERMANENT FILES PREFIXED WITH THE WORD "YOUR".
AS PRESENTLY STRUCTURED THIS DECK1 REQUIRES AN ELM74A UPDATE LIB ON TAPE,
HOWEVER BY CHANGING THESE DURING THE FIRST UPDATE OF DECK1, OTHER SIMCOMP
SIMULATIONS MAY BE RUN.  NOTE:  BY PERFORMING AN UPDATE ON THIS LIBRARY, ALL OR
PART OF THESE CARDS (EXCLUDING THE */ CARDS WHICH ARE NOT TEXT), CAN BE PUNCHED
OUT AUTOMATICALLY.
*/
*/
*/                      DECK 1
*/
*/
COMMENT.  DECK1 JOB CARD   T150  CM53000  MT1 (?? DXY00..SEE DECK10)
COMMENT. DECK1 PERFORMS A SIMCOMP COMPILATION
ATTACH(OLDPL,CHEEPELMLIB,ID=NREL,MR=1,CY=1)
COPYCR(INPUT,CTLG)
UPDATE(L=124,D,C=DECK1)
CCLINK(DECK1)
7/8/9
COMMENT.   CATALOG CARDS GO HERE.....
CATALOG(SIMPRG,YOURSIMPRG,ID=YOU,RP=1)
CATALOG(NEWPL,YOURNEWELMUPDATELIB,ID=YOU,RP=15)
TRANSF(TA***)          ....  ADD THIS CARD ONLY IF NEEDED (SEE DECK 10)
7/8/9
*IDENT CTLDECK1
*COMPILE DECK1
*/ CONTROL CARD UPDATE CORRECTIONS GO HERE ... ONLY IF NEEDED
*D,DECK1.10,12
ATTACH(OLDPL,COLESPGM,ID=COLE,MR=1)
7/8/9
*IDENT ELM
*COMPILE START,FLOW,SUBS
*/  ELM UPDATE CORRECTIONS GO HERE
*D,START.3,4
6/7/8/9
*/
*/
*/                      DECK 2
*/
*/
COMMENT.  DECK2 JOB CARD   T210  CM53000      (?? DXY01..SEE DECK10)
COMMENT. DECK2 PERFORMS A FORTRAN COMPILATION
ATTACH(SIMPRG,YOURSIMPRG,ID=YOU)
ATTACH(OLDPL,CHEEPELMLIB,ID=NREL,MR=1,CY=1)
```

```
DECK0     1
DECK0     2
DECK0     3
DECK0     4
DECK0     5
DECK0     6
DECK0     7
DECK0     8
DECK0     9
DECK0    10
DECK0    11
DECK0    12
DECK0    13
DECK0    14
DECK0    15
DECK0    16
DECK0    17
DECK0    18
DECK0    19
DECK0    20
DECK0    21
DECK0    22
DECK0    23
DECK0    24
DECK0    25
DECK0    26
DECK0    27
DECK0    28
DECK0    29
DECK0    30
DECK0    31
DECK0    32
DECK0    33
DECK0    34
DECK0    35
DECK0    36
DECK0    37
DECK0    38
DECK0    39
DECK0    40
DECK0    41
DECK0    42
DECK0    43
DECK0    44
DECK0    45
DECK0    46
DECK0    47
DECK0    48
DECK0    49
DECK0    50
DECK0    51
DECK0    52
DECK0    53
DECK0    54
DECK0    55
DECK0    56
```

```
COPYCR(INPUT,CTLG)                                                              DECK0      57
UPDATE(L=124,D,C=DECK2)                                                         DECK0      58
CCLINK(DECK2)                                                                   DECK0      59
7/8/9                                                                           DECK0      60
COMMENT.  CATALOG CARDS FOR LGO AND DEBUG PLUS PURGE  SIMPRG GO HERE            DECK0      61
CATALOG(DEBUG,YOURDEBUG,ID=YOU,RP=10)                                           DECK0      62
CATALOG(LGO,YOURLGO,ID=YOU,RP=10)                                               DECK0      63
PURGE(SIMPRG)                                                                   DECK0      64
TRANSF(TA$$$)          ....  ADD THIS CARD ONLY IF NEEDED (SEE DECK 10)         DECK0      65
7/8/9                                                                           DECK0      66
*IDENT CTLDECK2                                                                 DECK0      67
*COMPILE DECK2                                                                  DECK0      68
*/ CONTROL CARD UPDATE CORRECTIONS GO HERE ... ONLY IF NEEDED                   DECK0      69
6/7/8/9                                                                         DECK0      70
*/                                                                              DECK0      71
*/                                                                              DECK0      72
*/                        DECK 3                                                DECK0      73
*/                                                                              DECK0      74
*/                                                                              DECK0      75
COMMENT.  DECK3 JOB CARD   T150  CM140000      (?? DXY01..SEE DECK10)           DECK0      76
COMMENT. DECK3 EXECUTES THE SIMULATION                                          DECK0      77
ATTACH(EXFILE,YOURWEATHER,ID=YOU,MR=1)                                          DECK0      78
ATTACH(BUG,YOURDEBUG,ID=YOU)                                                    DECK0      79
ATTACH(LLGO,YOURLGO,ID=YOU)                                                     DECK0      80
ATTACH(OLDPL1,ELM74PAWNEEDATA011475,ID=NREL,MR=1)                               DECK0      81
ATTACH(OLDPL,CHEEPELMLIB,ID=NREL,MR=1,CY=1)                                     DECK0      82
COPYCR(INPUT,CTLG)                                                              DECK0      83
UPDATE(P=OLDPL,L=124,D,C=DECK3)                                                 DECK0      84
CCLINK(DECK3)                                                                   DECK0      85
7/8/9                                                                           DECK0      86
COMMENT.  CATALOG CARD FOR THE DIAGNOSTIC DUMP GOES HERE                        DECK0      87
CATALOG(DUMP,YOURDUMP,ID=YOU)                                                   DECK0      88
7/8/9                                                                           DECK0      89
*IDENT CTLDECK3                                                                 DECK0      90
*COMPILE DECK3                                                                  DECK0      91
*/ CONTROL CARD UPDATE CORRECTIONS GO HERE ... ONLY IF NEEDED                   DECK0      92
*D,DECK3,26                                                                     DECK0      93
RFL(65000)                                                                      DECK0      94
*D,DECK3,38                                                                     DECK0      95
RFL(65000)                                                                      DECK0      96
7/8/9                                                                           DECK0      97
*IDENT DATA                                                                     DECK0      98
*COMPILE P72T1                                                                  DECK0      99
*/ ELM DATA UPDATE                                                             DECK0     100
*D,P72T1,9                                                                      DECK0     101
X(1)=5*0.0$                                                                     DECK0     102
X(3)=430*0.$                                                                    DECK0     103
*D,P72T1.31,398                                                                 DECK0     104
          DT=1.$ TSTRT=0.$ TEND=5.$                                             DECK0     105
DTPR=1.0$                                                                       DECK0     106
PRINT. Y   ,X(1),X(2),X(3)                                                      DECK0     107
PLOT. (X(1))                                                                    DECK0     108
6/7/8/9                                                                         DECK0     109
*/                                                                              DECK0     110
*/                                                                              DECK0     111
*/                        DECK 4                                                DECK0     112
```

```
*/                                                                              DECK0   113
*/                                                                              DECK0   114
COMMENT.  DECK4 JOB CARD    T50?? CM53000                                       DECK0   115
COMMENT. DECK4 PERFORMS CORRECTONS ON THE COMPILED SIMULATION                   DECK0   116
COMMENT.ATTACH YOURNEWELMUPDATELIB,YOURLGO, & YOURDEBUG                         DECK0   117
ATTACH(YBUG,YOURDEBUG,ID=YOU)                                                   DECK0   118
ATTACH(OLDLGO,YOURLGO,ID=YOU)                                                   DECK0   119
ATTACH(OLDPL1,YOURNEWELMUPDATELIB,ID=YOU,MR=1)                                  DECK0   120
COPYCR(INPUT,CTLG)                                                              DECK0   121
ATTACH(OLDPL,CHEEPELMLIB,ID=NREL,MR=1,CY=1)                                     DECK0   122
UPDATE(L=124,D,C=DECK4)                                                         DECK0   123
CCLINK(DECK4)                                                                   DECK0   124
7/8/9                                                                           DECK0   125
COMMENT.PURGE AND CATALOG CARDS FOR YOURLGO AND YOURDEBUG GO HERE...            DECK0   126
PURGE(OLDLGO)                                                                   DECK0   127
CATALOG(LGO,YOURLGO,ID=YOU,RP=10)                                               DECK0   128
PURGE(YBUG)                                                                     DECK0   129
CATALOG(NEWBUG,YOURDEBUG,ID=YOU,RP=10)                                          DECK0   130
7/8/9                                                                           DECK0   131
*IDENT CTLDECK4                                                                 DECK0   132
*/ CONTROL CARD UPDATE CORRECTIONS GO HERE ... ONLY IF NEEDED                   DECK0   133
*/ UPDATE NO. 1 IS TO CHANGE THE CONTROL CARDS IF YOU WISH.                     DECK0   134
*COMPILE DECK4                                                                 DECK0   135
7/8/9                                                                           DECK0   136
*IDENT ELM2                                                                    DECK0   137
*/ UPDATE NO. 2 IS FOR ELM CORRECTIONS . DELETE ALL SUBROUTINES WHICH ARE NOT   DECK0   138
*/ TO BE RECOMPILED                                                            DECK0   139
*COMPILE SUBS,XFLOW                                                            DECK0   140
*D.SUBS.2,5                                                                    DECK0   141
*D.SUBS.7                                                                      DECK0   142
*CALL COMMON                                                                   DECK0   143
      Y=1000.                                                                   DECK0   144
      CALL SMITH                                                                DECK0   145
*I.XFLOW.21                                                                    DECK0   146
      CALL COLE                                                                 DECK0   147
7/8/9                                                                           DECK0   148
*IDENT NEWSUB                                                                  DECK0   149
*/ UPDATE NO. 3 IS TO ADD A NEW SUBROUTINE IF AND ONLY IF IT IS REQUIRED BY     DECK0   150
*/ THE CHANGE IN UPDATE  NO. 2 ABOVE.... COMPILE DUMMY IS NEEDED IF NO NEWSUBS  DECK0   151
*/ ARE REQUIRED . IF ANY OF THE NEW SUBROUTINES REQUIRE CORRECTION ON A SUB-    DECK0   152
*/ SEQUENT RUN ...MOVE THESE SUBROUTINES INTO UPDATE 2 WITH THE *ADDFILE & *DECK DECK0   153
*ADDFILE INPUT                                                                 DECK0   154
*DECK NEW                                                                      DECK0   155
      SUBROUTINE SMITH                                                          DECK0   156
      END                                                                       DECK0   157
      SUBROUTINE COLE                                                           DECK0   158
*CALL COMMON                                                                   DECK0   159
      Y=3.**(Y)                                                                 DECK0   160
      END                                                                       DECK0   161
*COMPILE DUMMY                                                                 DECK0   162
6/7/8/9                                                                         DECK0   163
*/                                                                              DECK0   164
*/                                                                              DECK0   165
*/                                                                              DECK0   166
*/                    DECK 5                                                    DECK0   167
                                                                                DECK0   168
```

```
*/                                                                              DECK0      169
COMMENT.  DECK5 JOB CARD   T2    CM10000                                        DECK0      170
COMMENT. DECK5 OBTAINS A LISTING OF THE LOADER MAP AND/OR THE DIAGNOSTIC DUMP   DECK0      171
ATTACH(A,YOURDUMP,ID=YOU)                                                       DECK0      172
COPYSBF(A)  GIVES YOU A LOADER MAP. IF NOT NEEDED USE SKIPF(A,1,17,C)           DECK0      173
COPYSBF(A)  GIVES YOU THE FULLLLLLLLL DIAGNOSTIC DUMP                           DECK0      174
6/7/8/9                                                                         DECK0      175
*/                                                                              DECK0      176
*/                                                                              DECK0      177
*/                      DECK 6                                                  DECK0      178
*/                                                                              DECK0      179
*/                                                                              DECK0      180
COMMENT.  DECK6 JOB CARD   T5    CM40000                                        DECK0      181
COMMENT. DECK6 OBTAINS AN UPDATE LIST OF THE COMMON AND XFLOWS                  DECK0      182
ATTACH(OLDPL,YOURNEWELMUPDATELIB,ID=YOU,MR=1)                                   DECK0      183
UPDATE(L=0)                                                                     DECK0      184
COMMENT.   THE XFLOW WILL BE A LONGGGGGGG LIST....                              DECK0      185
COPYSBF(COMPILE)                                                                DECK0      186
7/8/9                                                                           DECK0      187
*IDENT MAYBE                                                                    DECK0      188
*ADDFILE INPUT                                                                  DECK0      189
*DECK FUM?                                                                      DECK0      190
*CALL COMMON                                                                    DECK0      191
*COMPILE XFLOW                                                                  DECK0      192
6/7/8/9                                                                         DECK0      193
*/                                                                              DECK0      194
*/                                                                              DECK0      195
*/                      DECK 7                                                  DECK0      196
*/                                                                              DECK0      197
*/                                                                              DECK0      198
COMMENT.  DECK7 JOB CARD   T5    CM10000  MT1                                   DECK0      199
COMMENT. STORE A BINARY FILE ON LABELLED TAPE D1243                             DECK0      200
COMMENT. ASSUMING A FILE NAMED LGO IS TO BE STORED AND LABELED MYLGO            DECK0      201
REWIND(LGO)                                                                     DECK0      202
REQUEST(ELMMIS,HY,ID-D1243,U,E,WRITE,MF)SWIFT                                   DECK0      203
LABEL(FILE1,W,VSN=D1243,T=999,M=ELMMIS,L=MYLGO)WRITE,SWIFT                      DECK0      204
COPYBF(LGO,FILE1)                                                               DECK0      205
6/7/8/9                                                                         DECK0      206
*/                                                                              DECK0      207
*/                                                                              DECK0      208
*/                      DECK 8                                                  DECK0      209
*/                                                                              DECK0      210
*/                                                                              DECK0      211
COMMENT.  DECK8 JOB CARD   T5    CM10000  MT1                                   DECK0      212
COMMENT.   RETRIEVE A CODED FILE FROM LABELLED TAPE D1243                       DECK0      213
COMMENT. ASSUMING THAT WE RETRIVE FILE FOXXY AND PUT IN ON DOG                  DECK0      214
REQUEST(ELMMIS,HY,ID-D1243,U,E,READ,MF)SWIFT                                    DECK0      215
LABEL(FILEB,R,VSN=D1243,L=FOXXY,M=ELMMIS)READ,SWIFT                             DECK0      216
COPYCF(FILEB,DOG)                                                               DECK0      217
REWIND(DOG)                                                                     DECK0      218
6/7/8/9                                                                         DECK0      219
*/                                                                              DECK0      220
*/                                                                              DECK0      221
*/                      DECK 9                                                  DECK0      222
*/                                                                              DECK0      223
*/                                                                              DECK0      224
```

```
          COMMENT.  DECK9 JOB CARD   TS    CM10000  MT1                                                    DECK0     225
          COMMENT.  RETRIEVES ALL LABELS                                                                   DECK0     226
          REQUEST(ELMMIS,HY,ID-D1243,U,E,READ,MF)SWIFT                                                     DECK0     227
          LISTMF(M=ELMMIS)                                                                                 DECK0     228
          RETURN(ELMMIS)                                                                                   DECK0     229
          6/7/8/9                                                                                          DECK0     230
          */                                                                                               DECK0     231
          */                                                                                               DECK0     232
          */                          DECK 10                                                              DECK0     233
          */                                                                                               DECK0     234
          */                                                                                               DECK0     235
          DECK10 CONTAINS THE CONTROL CARD INFORMATION NECESSARY TO SUBMIT DECKS1,2, AND 3 SIMULTANEOUSLY, DECK0     236
          IF DESIRED.                                                                                      DECK0     237
                                                                                                           DECK0     238
               IN DECK1 : ADD THE ARGUMENT DXY00 TO THE JOB CARD. ADD A CONTROL CARD                       DECK0     239
               TRANSF(TA***)                                                                               DECK0     240
               AS THE LAST CARD IN THE SECOND RECORD OF THE DECK. THE *** IS THE JOB NUM.                  DECK0     241
               FOR DECK2.                                                                                  DECK0     242
                                                                                                           DECK0     243
               IN DECK2 : ADD THE ARGUMENT DXY01 TO THE JOB CARD. ADD A CONTROL CARD                       DECK0     244
               TRANSF(TA$$$)                                                                               DECK0     245
               AS THE LAST CARD IN THE SECOND RECORD OF THE DECK. THE $$$ IS THE JOB NUM.                  DECK0     246
               FOR DECK3                                                                                   DECK0     247
                                                                                                           DECK0     248
               IN DECK3 : ADD THE ARGUMENT DXY01 TO THE JOb CARD.                                          DECK0     249
                                                                                                           DECK0     250
                                                                                                           DECK0     251
               SHOULD THE FIRST OR SECOND JOB BOMB, IT MAY BE RESUBMITTED SINCE THE                        DECK0     252
               REMAINING JOBS WILL BE ON THE DEPENDENCY QUEUE AWAITING THE EXECUTION                       DECK0     253
               OF THE TRANSF(TA***) CONTROL CARDS.                                                         DECK0     254
                                                                                                           DECK0     255
               A SIMILAR GAME MAY BE PLAYED BETWEEN DECKS 3 AND 4                                          DECK0     256
*****     *ENDTEXT                                                                                         DECK0     257
*****     *DECK DECK1                                                                                      DECK1       1
*****     */                                                                                                         1
*****     */                                  DECK1 CREATES AND STORES A SIMCOM                                      1
*****     */                                  COMPILATION ON A NEW UPDATE LIBRARY                                    1
*****     */                                  CONTAINING XFLOW AND COMMON. THIS LATER                                1
*****     */                                  FILE IS USED TO MAKE LATER CORRECTIONS                                 1
          RFL(10000)                          WITHOUT RERUNNING THE TOTAL SIMULATION.                                1
          RETURN(OLDPL)                                                                            DECK1       2
          MAP(OFF)                                                                                 DECK1       3
          REWIND(CTLG)                                                                             DECK1       4
*****     */                                                                                       DECK1       5
*****     */                                  BRING ON 3R) UPDATE DIRECTIVES AND SIFT1,                          5
          ATTACH(B,CHEEPELMLIB,ID=NREL,MR=1,CY=5)  AND SIFT2.                                                    5
          SKIPF(B,1,0,C)                                                                           DECK1       6
          COPYCR(B,UDIN)                                                                           DECK1       7
          REWIND(UDIN)                                                                             DECK1       8
*****     */                                  BRING ON EL474 UPDATE LIBRARY.                       DECK1       9
          REQUEST(ELMMIS,HY,ID-D1243,U,E,READ,MF)SWIFT                                                       9
          LABEL(ELMUD,R,VSN=D1243,T=999,M=ELMMIS,L=ELM74AUPDATE )READ,SWIFT                        DECK1      10
          COPYBF(ELMUD,OLDPL)                                                                      DECK1      11
          RFL(40000)                                                                               DECK1      12
*****     */                                  UPDATE1 FOR ELM                                      DECK1      13
          UPDATE(D,L=124)                                                                                    13
                                                                                                   DECK1      14
```

```
CREATION   N            CARDS ENCOUNTERED IN INPUT            UPDATE V1.2        05/26/75  10.41.07.      PA(    6
*****   */                                 COMPACT UPDATE INFO.                                  14
        REWIND(OUTPUT)                                                               DECK1      15
        COPYCF(OUTPUT,TEMP)                                                          DECK1      16
        REWIND(TEMP,OUTPUT)                                                          DECK1      17
        COPYSBF(TEMP,OUTPUT)                                                         DECK1      18
        RFL(10000)                                                                   DECK1      19
        ATTACH(SIMCOM,SIMCOM3,CY=1,MR=1,ID=NREL)                                     DECK1      20
        REQUEST(SIMPRG,*PF)                                                          DECK1      21
        RFL(53000)                                                                   DECK1      22
*****   */                                 OBTAIN SIMCOM COMPILATION                            22
        SIMCOM(COMPILE)                                                              DECK1      23
        RFL(10000)                                                                   DECK1      24
        RETURN(COMPILE)                                                              DECK1      25
*****   */                                 USE SIFT2 TO GET ONLY SUBROUTINE XFLOWS.             25
        COPYBR(B,SIFT2)                                                              DECK1      26
        REWIND(SIFT2)                                                                DECK1      27
        RFL(50000)                                                                   DECK1      28
        SIFT2(SIMPRG,,XFLOW)                                                         DECK1      29
        RFL(10000)                                                                   DECK1      30
        REWIND(XFLOW)                                                                DECK1      31
*****   */                                 USE SIFT1 TO GET COMMON FROM XFLOW.                  31
        COPYBR(B,SIFT1)                                                              DECK1      32
        REWIND(SIFT1)                                                                DECK1      33
        RFL(50000)                                                                   DECK1      34
        SIFT1(XFLOW,,COMMON)                                                         DECK1      35
        REWIND(COMMON,XFLOW)                                                         DECK1      36
*****   */                                 UPDATE 2 PUTS COMMON AND XFLOW ON AN                 36
*****   */                                 UPDATE LIB.                                          36
        UPDATE(N=MERGE,L=0,I=UDIN)                                                   DECK1      37
        RETURN(NEWPL)                                                                DECK1      38
        REQUEST(NEWPL,*PF)                                                           DECK1      39
*****   */                                 UPDATE 3 MERGES THE ORGINAL ELM/4A AND               39
*****   */                                 COMMON AND XFLOWS AND CREATES A NEW                  39
*****   */                                 PROGRAM LIBRARY.                                     39
        UPDATE(M=OLDPL,P=MERGE,N,L=0)                                                DECK1      40
        RFL(20000)                                                                   DECK1      41
        REWIND(SIMPRG)                                                               DECK1      42
        CCLINK(CTLG)                                                                 DECK1      43
*****   *DECK DECK2                                                                  DECK2       1
*****   */                                 DECK2 CREATES AND STORES LGO AND DEBUG AS             1
*****   */                                 PERMANENT FILES FOR USE IN DECK3.                     1
        RFL(10000)                                                                   DECK2       2
        REWIND(CTLG)                                                                 DECK2       3
        REQUEST(LGO,*PF)                                                             DECK2       4
        REQUEST(DEBUG,*PF)                                                           DECK2       5
        RFL(53000)                                                                   DECK2       6
        FTN(I=SIMPRG,LN=DEBUG,R=1,A,S=0,ROUND=+-*/)                                  DECK2       7
        RFL(20000)                                                                   DECK2       8
        CCLINK(CTLG)                                                                 DECK2       9
        EXIT,S.                                                                      DECK2      10
        ATTACH(LIB,SIMCOM3,CY=3,MR=1,ID=NREL)                                        DECK2      11
        SELECT(P=EDITOR,I=EDIT)                                                      DECK2      12
        EDIT.                                                                        DECK2      13
*****   *COMDECK D3A                                                                 D3A         1
        RFL(10000)                                                                   D3A         2
        REWIND(OUTPUT)                                                               D3A         3
```

```
      REQUEST(DUMP,*PF)                                                          D3A        4
***** */                                                                                    4
      COPYCF(OUTPUT,DUMP)                REMOVE LOADER MAP.                      D3A        5
***** */                                                                                    5
      COPYCF(OUTPUT,TEMP,10)             REST OF OUTPUT TO TEMP.                 D3A        6
      REWIND(OUTPUT,TEMP)                                                        D3A        7
***** */                                                                                    7
      COPYSBF(FUD,OUTPUT)                REPLACE UPDATE INFO ON OUTPUT.          D3A        8
***** */                                                                                    8
      COPYCF(TEMP,OUTPUT)                REPLACE SIMULATION RESULTS ON OUTPUT.   D3A        9
***** */                                                                                    9
      RFL(43000)                         REPLACE FIRST SHEET OF DEBUG DIAGNOSTIC D3A       10
      MAP(OFF)                                                                   D3A       11
      SIFT3(TEMP)                                                                D3A       12
      RFL(10000)                                                                 D3A       13
      REWIND(TEMP)                                                               D3A       14
      SKIPF(TEMP,1,17,C)                                                         D3A       15
***** */                                                                                    15
      COPYCF(TEMP,DUMP,10)               STORE DIAGNOSTIC DUMP                   D3A       16
      RFL(20000)                                                                 D3A       17
      CCLINK(CTLG)                                                               D3A       18
***** *DECK DECK3                                                                 DECK3      1
***** */                                                                                    1
***** */                               DECK3 RUNS THE SIMULATION FROM THE FTN.              1
***** */                               LGO FILE STORED IN DECK2.  IT ALSO                   1
***** */                               ACCEPTS THE DEBUG FILE FOR DEBUGGING IF              1
***** */                               NEEDED.  THE DEBUG DUMP OF ALL VARIABLES             1
***** */                               AND THE LOADER MAP ARE REMOVED FROM                  1
***** */                               OUTPUT AND CAN BE STORED FOR LATER                   1
      RFL(10000)                       RETRIEVAL.                                        1
      REWIND(CTLG)                                                               DECK3      2
      COPYBF(LLGO,LGO)                                                           DECK3      3
***** */                                                                        DECK3      4
***** */                               FTN. IS NEEDED TO FOOL SIMCOMP SO THAT IT            4
***** */                               THINKS THAT DEBUG FILE HAS BEEN CREATED.             4
***** */                               PGM. SIFT3 IS LATER USED TO REMOVE THE               4
      ATTACH(FTNDUM,CHEEPELMLIB,CY=5,ID=NREL,MR=1)  THE REGISTER DUMP OF DEBUG DIAGNOSTIC. 4
      COPYCR(FTNDUM,FTNIN)                                                       DECK3      5
      REWIND(FTNIN)                                                              DECK3      6
      RFL(53000)                                                                 DECK3      7
      FTN(I=FTNIN,LN=DEBUG,B=SIFT3)                                              DECK3      8
      REWIND(SIFT3)                                                              DECK3      9
      RFL(10000)                                                                 DECK3     10
      RETURN(DEBUG)                                                              DECK3     11
***** */                               BRING ON REAL DEBUG.                     DECK3     12
      COPYCF(BUG,DEBUG)                                                                    12
      ATTACH(LIB,SIMCOM3,CY=3,MR=1,ID=NREL)                                      DECK3     13
      ATTACH(B,SIMCOM3,CY=2,MR=1,ID=NREL)                                        DECK3     14
      RFL(40000)                                                                 DECK3     15
***** */                               FIRST UPDATE IS FOR THE ELM DATA INPUT.  DECK3     16
      UPDATE(D,C=OUT,L=124,N,P=OLDPL1)                                                     16
***** */                                                                        DECK3     17
***** */                               REMOVE AND STORE OUTPUT PRIOR TO LOADER              17
      RFL(10000)                       MAPS, IT WILL BE PUT BACK LATER.                    17
      REWIND(OUTPUT)                                                             DECK3     18
      COPYCF(OUTPUT,FUD)                                                         DECK3     19
                                                                                 DECK3     20
```

```
      REWIND(OUTPUT,FUO)                                                    DECK3    21
      RFL(15000)                                                            DECK3    22
*****/                                                                              22
      SELECT.                                SATISFY EXTERNALS ON LGO.
      COPYBF(B,LGO)                                                         DECK3    23
      MAP(PART)                                                             DECK3    24
      RFL(140000)                                                           DECK3    25
      LOAD(LGO)                                                             DECK3    26
      NOGO.                                                                 DECK3    27
*****/                                                                      DECK3    28
*****/                                 PLACE AN EOR(17) MARK ON OUTPUT SO THAT        28
*****/                                 WE CAN FIND THE MAPS LATER FOR REMOVAL         28
      RFL(10000)                       FROM OUTPUT                                   28
      REWIND(OUTPUT)                                                        DECK3    29
      COPYCF(OUTPUT,SAVE)                                                   DECK3    30
      REWIND(SAVE,OUTPUT)                                                   DECK3    31
      COPYCF(SAVE,OUTPUT)                                                   DECK3    32
      REWIND(NEWT1)                                                         DECK3    33
      RFL(20000)                                                            DECK3    34
      SELECT(P=PRELOAD,I=PRELOAD)                                           DECK3    35
      PRELOAD(NEWT1,MAIN)                                                   DECK3    36
      RFL(120000)                                                           DECK3    37
*****/                                                                      DECK3    38
      MAIN(,OUT)                       EXECUTE SIMULATION WITH DATA ON FILE OUT       38
*****/                                                                      DECK3    39
*****/                                 THE REMAINDER OF THESE CARDS UP TO THE         39
*****/                                 EXIT. CARD, REMOVES THE UNWANTED OUTPUT        39
*****/                                 TO A FILE CALLED DUMP FOR LATER RETRIEVAL      39
*****/                                 IF NEEDED FOR CATALOGING.                      39
*****/*CALL D3A                                                                     DECK3    40
      EXIT.                                                                 DECK3    41
*****/                                 THE REMAINDER OF THE CARDS DUPLICATE THE       41
*****/                                 ABOVE SHOULD WE DUMP ON AN ERROR MODE          41
*****/                                 THAT SIMCOMP DOESN*T RECOVER FROM.             41
*****/*CALL D3A                                                                     DECK3    42
*****/*DECK DECK4                                                                  DECK4     1
*****/                                 DECK4 CORRECTS AN EXISTING SIMCOMP SIM.        1
*****/                                 LGO FILE WITHOUT A COMPLETE RERUN              1
      RFL(10000)                                                            DECK4    2
      MAP(OFF)                                                              DECK4    3
      REWIND(CTLG)                                                          DECK4    4
      RFL(53000)                                                            DECK4    5
*****/                                 UPDATE1 MODIFIES A PGM. ON YOUR LGO FILE       5
      UPDATE(P=OLDPL1,L=124)                                                DECK4    6
      FTN(I=COMPILE,B=OLDSUB,LN=DEBUG,R=1,A,S=0,ROUND=+-*/.                 DECK4    7
      RFL(10000)                                                            DECK4    8
      REWIND(OLDSUB)                                                        DECK4    9
      REQUEST(LGO,*PF)                                                      DECK4   10
*****/                                                                              10
*****/                                  COPYL REPLACES THE OLD SUB WITH THE           10
      COPYL(OLDLGO,OLDSUB,LGO)         CORRECTED OLD SUB                             10
      REWIND(LGO,COMPILE)                                                   DECK4    11
      RFL(53000)                                                            DECK4    12
*****/                                                                      DECK4    13
*****/                                 UPDATE2 ADDS A NEW SUB. REQUIRED BY            13
      UPDATE(P=OLDPL1,L=124)           UPDATE1                                       13
      RFL(10000)                                                            DECK4    14
                                                                            DECK4    15
```

```
*****     */                                  COMPACT UPDATE OUTPUT                                  15
          REWIND(OUTPUT)                                                              DECK4        16
          COPYCF(OUTPUT,TEMP)                                                         DECK4        17
          REWIND(OUTPUT,TEMP)                                                         DECK4        18
          COPYSBF(TEMP,OUTPUT)                                                        DECK4        19
          RFL(53000)                                                                  DECK4        20
          FTN(I=COMPILE,B=NEWSUB,LN=DEFOX,R=1,A,S=0,ROUND=+-*/)                       DECK4        21
*****     */                                  PUT FTN LISTING ON OUTPUT                              21
          RFL(10000)                                                                  DECK4        22
          REWIND(DEFOX)                                                               DECK4        23
          COPYSBF(DEFOX)                                                              DECK4        24
          REWIND(DEFOX)                                                               DECK4        25
          COPYCF(DEFOX,DEBUG)                                                         DECK4        26
*****     */                                  ADD NEW DEBUG DATA TO YOURDEBUG FILE                   26
          REWIND(DEBUG)                                                               DECK4        27
          REQUEST(NEWBUG,*PF)                                                         DECK4        28
          COPYCF(DEBUG,NEWBUG,100)                                                    DECK4        29
          COPYCF(YBUG,NEWBUG,100)                                                     DECK4        30
          PFL(15000)                                                                  DECK4        31
          REWIND(NEWSUB)                                                              DECK4        32
*****     */                                  CREATE A SELCT LIB. TO ALLOW THE NEW SUB.              32
*****     */                                  TO BE SELECTED IF NEEDED                               32
          SELECT(B,I=NEWSUB)                                                          DECK4        33
          SELECT.                                                                     DECK4        34
          RFL(20000)                                                                  DECK4        35
          CCLINK(CTLG)                                                                DECK4        36
          EXIT.S.                                                                     DECK4        37
          RETURN(LIB)                                                                 DECK4        38
          ATTACH(LIB,SIMCOM3,CY=3,MR=1,ID=NREL)                                       DECK4        39
          SELECT(P=EDITOR,I=EDIT)                                                     DECK4        40
          EDIT.                                                                       DECK4        41
```

CORRECTION IDENTS ARE LISTED IN CHRONOLOGICAL ORDER OF INSERTION

DECK0 DECK1 DECK2 D3A DECK3 DECK4

DECKS ARE LISTED IN THE ORDER OF THEIR OCCURRENCE ON A NEW PROGRAM LIBRARY IF ONE IS CREATED BY THIS UPDATE

YANK$$$ DECK0 DECK1 DECK2 D3A DECK3 DECK4

DECKS WRITTEN TO COMPILE FILE

DECK0 DECK1 DECK2 DECK3 DECK4

THIS UPDATE REQUIRED 33400B WORDS OF CORE.

```
  05/26/75 *CSU SCOPE 3.3.14  B C012 C013 C140 C141 05/20/75
10.40.57.TA123GQ    FROM  AB       13A
10.40.57.TA123,T2,CM40000,ANM****Y.  COLE,GW.  $S
10.40.57.AVE  CREATE CHEEPELMLIB CY=1
10.40.58.REQUEST(NEWPL,*PF)
10.40.58.UPDATE(L=A124,D,N)
10.41.07.READING INPUT
10.41.07.UPDATE CREATION RUN
10.41.08.CREATING NEW PROGRAM LIBRARY
10.41.12.UPDATE COMPLETE
10.41.12.CATALOG(NEWPL,CHEEPELMLIB,ID=NREL,CY=1,R
10.41.12.P=999)
10.41.14.CYCLE **, CHEEPELMLIB
10.41.14.PFN FOUND IN SD  023
10.41.14.FILE CATALOGUED AS
10.41.14.CYCLE 01, CHEEPELMLIB
10.41.14.IN SD 023
10.41.14.CP       1.018 SEC.
10.41.14.PP       5.293 SEC.
10.41.14.IO       1.775 SEC.
```

APPENDIX III

ELM74PAWNEEDATA

This appendix contains a partial listing of the ELM74PAWNEEDATA update library file used in Deck 3.

```
TIME=0. $                                                                P72T1     2
TSTRT=0. $                                                               P72T1     3
TEND=2. $                                                                P72T1     4
DT=2. $                                                                  P72T1     5
DTPR=364. $                                                              P72T1     6
DTPL=3.64 $                                                              P72T1     7
DTFL=14. $                                                               P72T1     8
X(1)=999*0. $                                                            P72T1     9
X(200)=0.,0.,0.,0.,10.$                                                  P72T1    10
X(210)=10.32,3.87 ,6.02 ,3.44 ,18.92$                                    P72T1    11
X(220)=22.5,1.4,.55,6.4,1.3$                                             P72T1    12
X(230)=5*.4$                                                             P72T1    13
X(240)=47.87, 17.95,27.92,15.96,87.75$                                   P72T1    14
X(280)=80.,9.$                                                           P72T1    15
X(290)=66.,2.,63.,2.,112.,4.$                                            P72T1    16
X(470)=1.21,.63,1.21,.63,3*0.,139.$                                      P72T1    17
X(498)=4.,11*0.$                                                         P72T1    18
X(510)=0.,.001$                                                          P72T1    19
X(610)=416.,693.,4510.$                                                  P72T1    20
X(616)=0.,2*4.,2.$                                                       P72T1    21
X(651)=.03,2*.07,.03$                                                    P72T1    22
X(661)=3.2,2*6.4,3.4$                                                    P72T1    23
X(811)=25.,2*5.0,3.0$                                                    P72T1    24
X(821)=15.,6.,4.,2.$                                                     P72T1    25
X(891)=1800.,900.,600.,400.$                                             P72T1    26
X(901)=.00268,.813,4*0.,.048,1.83,.2$                                    P72T1    27
X(911)=.00268,.956,4*0.,.095,3.13,.4$                                    P72T1    28
X(921)=.00536,.605,4*0.,.095,4.89,.4$                                    P72T1    29
X(931)=.00536,.365,4*0.,.062,6.7,.58$                                    P72T1    30
ACSIO=2.5 $                                                              P72T1    31
ACLD(1)=58., 56., 58., 55., 61., 55., 47., 43., 46., 45., 51., 56. $     P72T1    32
ACLIT=0. $                                                               P72T1    33
ACOND=7.E-04 $                                                           P72T1    34
ACONV=100. $                                                             P72T1    35
ACPIR(1)=2*-1., .8, 2*0. $                                               P72T1    36
ACSC=0. $                                                                P72T1    37
ADAMO=4. $                                                               P72T1    38
ADEN=1.4 $                                                               P72T1    39
ADEP(1)=1.5, 2.5, 11., 4*15., 5*0. $                                     P72T1    40
ADPM=75. $                                                               P72T1    41
ADRD=1. $                                                                P72T1    42
AEV=15. $                                                                P72T1    43
AFAIN=0 $                                                                P72T1    44
AFIEL(1)=.16, .18, .25, 12*.29 $                                         P72T1    45
AFILI(1)=.16, .18, .25, 4*.29, 4*0. $                                    P72T1    46
AFIPR=3.3E-04 $                                                          P72T1    47
AHHC=.5$                                                                 P72T1    48
AHHD=.3$                                                                 P72T1    49
AHHH=150.$                                                               P72T1    50
AIRRI=0 $                                                                P72T1    51
AMIRR(1)=2*1., .7, 2*0. $                                                P72T1    52
```